

# Celestial Road

Videojuego de laberintos en  
Realidad Virtual con Unity



Grado en Ingeniería Multimedia

Trabajo Fin de Grado

Autor:  
Alba González Aller

Tutor/es:  
Mireia Luisa Sempere Tortosa

Septiembre 2019



Universitat d'Alacant  
Universidad de Alicante



## Resumen

Celestial Road es un juego de Realidad Virtual, desarrollado con el motor de Unity, que plantea retos en forma de laberintos a sus jugadores. El jugador tendrá que encontrar el camino correcto, recoger los objetos escondidos y llegar a la salida en el menor tiempo posible para lograr desbloquear el siguiente nivel y avanzar en el juego.

Además, el uso de la Realidad Virtual aporta a Celestial Road un aspecto diferenciador, dando al usuario la oportunidad de adentrarse en un nuevo mundo y disfrutar de estos retos en primera persona. Para ello, el usuario solo necesitará disponer de un smartphone con Android y unas gafas de Realidad Virtual, como pueden ser las Cardboard.

## Abstract

Celestial Road is a Virtual Reality game, developed with Unity engine, which poses challenges in the form of mazes to its players. The player will have to find the right path, collect the hidden objects and reach the exit in the shortest possible time to unlock the next level and advance in the game.

In addition, the use of Virtual Reality gives Celestial Road a differentiating aspect, giving the user the opportunity to enter a new world and enjoy these challenges in the first person. For this, the user will only need to have an Android smartphone and Virtual Reality glasses, such as Cardboards.



# Motivación, justificación y objetivo general

La idea de este trabajo de fin de grado nace de la curiosidad por aprender nuevas tecnologías, a la vez que se pone en práctica lo aprendido durante toda la carrera de *Ingeniería Multimedia*. Aplicando los conocimientos adquiridos de ámbitos como la gestión de proyectos, diseño de videojuegos y programación. También se incluyen elementos de tecnologías cada día más importantes como es, en este caso, la realidad virtual.

El campo de la realidad virtual y aumentada crece día a día, ofreciendo a los usuarios nuevas experiencias imposibles de vivir en la vida real, ya sea por una incapacidad, tiempo, dinero o simplemente porque son en lugares fantásticos e imaginarios.

Por todo ello, siempre ha llamado mi atención y por tanto la motivación de este proyecto se trata de una motivación personal por aprender más sobre esta tecnología. Por ello, a pesar de haber optado por la rama de gestión de contenidos en el cuarto curso de Ingeniería Multimedia, he considerado esta una gran oportunidad para conocer cómo funciona la realidad virtual, cómo se puede desarrollar para esta tecnología y qué aplicaciones puede tener tanto en el presente como en el futuro.

De esta forma, este proyecto busca crear un entorno virtual en el que aquellas personas apasionadas de los puzzles y laberintos, como es mi caso, puedan disfrutar de ellos de una forma más inmersiva e interactiva, dándole la oportunidad de vivirlos en primera persona desde cualquier lugar y en cualquier momento. Para cumplir esta última característica, decidí realizar este juego para dispositivos móviles haciendo uso de las *Google Cardboard*, para la realidad virtual.

Además, este trabajo también me permite estudiar y practicar otros de los campos que desde el inicio de la carrera llamaban mi atención, como son el diseño y el modelado. Por todo ello, creo que este trabajo me ha ayudado a conocer muchos campos nuevos realmente interesantes y apasionantes, de los que seguiré aprendiendo en los años venideros.

# Agradecimientos

Para comenzar me gustaría agradecer el apoyo de todas las personas que me han acompañado durante estos años, que han estado a mi lado tanto en los buenos momentos como en los de agobio, tristeza, etc.

También me gustaría agradecerlo a mis padres por haberme permitido estudiar la carrera que elegí, aunque supusiera que me alejara de casa y me fuera hasta Alicante. Y sobre todo a mi madre por haberme escuchado siempre que lo he necesitado y haberme animado siempre a seguir adelante sin dudar de que lo conseguiría.

En segundo lugar, quiero dar las gracias a mis amigos y a mi pareja, por estar siempre apoyándome y a mi lado, por haberme sacado de casa cuando sabían que lo necesitaba, haber soportado mis momentos de tristeza y haber celebrado conmigo todos los éxitos; por todo ello, solo puedo decir *gracias*.

Por último, agradecer a todos los profesores que me han enseñado y apoyado en estos cuatro años, sobre todo a los profesores de cuarto curso que se han implicado muchísimo con todos los alumnos y nos han ayudado a salir adelante y disfrutar de este último año de carrera.

## Citas

*“La innovación distingue a los líderes de los seguidores”*

- Steve Jobs

*“Creo que es posible para la gente normal elegir ser extraordinaria”*

- Elon Musk

*“No hay nada más peligroso que una idea cuando solo se tiene una”*

- Alain

*“No he fallado. Simplemente he encontrado formas que no funcionan”*

- Thomas Edison.





# Índice de contenidos

Resumen.....	2
Abstract .....	2
Motivación, justificación y objetivo general .....	4
Agradecimientos .....	5
Citas.....	6
Índice de figuras .....	12
Índice de tablas .....	15
1. Introducción .....	16
2. Planificación .....	18
3. Estado del arte. ....	19
3.1. Realidad Virtual .....	19
3.1.1. ¿Qué es la realidad virtual?.....	19
3.1.2. Evolución de la realidad virtual .....	20
3.1.3. Dispositivos actuales de Realidad Virtual.....	24
3.1.4. Conclusión .....	29
3.2. Análisis de mercado .....	29
3.2.1. Principales juegos y aplicaciones .....	30
3.2.2. Competencia y referentes .....	36
4. Herramientas utilizadas .....	39
4.1. Motores de videojuegos.....	39
4.1.1. ¿Qué es un motor de videojuegos? .....	39
4.1.2. Principales motores en la actualidad .....	40
4.1.3. Conclusión .....	42
4.2. Software de modelado .....	43
4.2.1. Blender .....	43
4.2.2. Autodesk 3ds Max .....	44

4.2.3.	Conclusión .....	44
4.3.	Herramientas de diseño y texturizado .....	45
4.3.1.	Adobe Photoshop .....	45
4.3.2.	Adobe Illustrator .....	45
5.	Objetivos .....	47
6.	Metodología .....	48
6.1.	Metodología de desarrollo .....	48
6.2.	Gestión del proyecto .....	49
6.3.	Control de versiones .....	50
7.	Game Design Document (GDD) .....	51
7.1.	Concepto y características básicas .....	51
7.2.	Jugabilidad y mecánicas .....	52
7.3.	Estilo y ambientación .....	56
7.3.1.	Guía de estilo .....	57
7.3.1.1.	Misión .....	57
7.3.1.2.	Filosofía de diseño .....	57
7.3.1.3.	Logo y colores .....	58
7.3.2.	Música .....	59
7.4.	Estados del juego .....	60
7.5.	Interfaces .....	61
7.6.	Niveles .....	66
8.	Requisitos funcionales y no funcionales .....	75
8.1.	Requisitos funcionales .....	75
8.2.	Requisitos no funcionales .....	79
9.	Fases de la implementación .....	81
9.1.	Diseño de los mapas e interfaces .....	81
9.2.	Modelado .....	83
9.3.	Creación de texturas .....	85

9.4.	Importación a Unity.....	87
9.5.	Programación y scripts necesarios.....	89
10.	Pruebas y validación.....	91
10.1.	Validación.....	92
11.	Conclusiones.....	94
11.1.	Estado actual del juego.....	94
11.2.	Mejoras y ampliaciones.....	94
11.3.	Nociones aprendidas.....	95
11.4.	Resultado y conclusión final.....	97
	Bibliografía.....	98
	Anexo.....	102
A.	Objetos modelados.....	102



# Índice de figuras

Figura 1. Infografía de informe anual de la GSMA.....	16
Figura 2. Ejemplo de un estereoscopio.....	21
Figura 3. Sensorama.....	21
Figura 4. Espada de Damocles, primer HMD.....	22
Figura 5. VIVED - 1984.....	23
Figura 6. Consola Virtual Boy de Nintendo.....	23
Figura 7. Oculus Rift.....	25
Figura 8. Gafas HTC Vive.....	26
Figura 9. HTC Vive Pro.....	26
Figura 10. Gafas Sony PlayStation VR.....	27
Figura 11. Samsung Gear VR.....	28
Figura 12. Controles mando Gear VR.....	28
Figura 13. Google Cardboard.....	29
Figura 14. Imágenes de la aplicación Sites in VR.....	30
Figura 15. Imágenes de InMind VR.....	31
Figura 16. Imágenes de la aplicación El cuerpo humano en 3D.....	32
Figura 17. Imágenes de la aplicación Jurassic VR – Google Cardboard.....	33
Figura 18. Capturas de la aplicación de Google Spotlight Stories.....	34
Figura 19. Capturas de la aplicación Within.....	34
Figura 20. Imágenes de VR Cave.....	35
Figura 21. Imágenes de VR Thrills: Roller Coaster 360.....	36
Figura 22. Imágenes de VR Roller Coaster Temple Rider.....	36
Figura 23. Imágenes de GooglePlay de “Laberinto VR Maze” y “Maze VR – Cardboard” respectivamente.....	37
Figura 24. Imágenes de GooglePlay de “3D Maze (The Labyrinth)” y “Maze World 3D” respectivamente.....	38
Figura 25. Logo de Unreal Engine y vista de su interfaz.....	40
Figura 26. Logo de Unity y vista de su interfaz.....	41
Figura 27. Logo de CryEngine y vista de su interfaz.....	42
Figura 28. Logo de Blender y vista de su interfaz.....	43
Figura 29. Logo de 3DS Max y vista de su interfaz.....	44
Figura 30. Logo Photoshop y vista de su interfaz.....	45

Figura 31. Logo Illustrator y vista de su interfaz .....	46
Figura 32. Esquema metodología SCRUM. Fuente [42] .....	48
Figura 33. Tablero de Trello .....	50
Figura 34. Punto de mira reticular cerrado.....	54
Figura 35. Punto de mira reticular abierto.....	54
Figura 36. Círculo de carga casi completo.....	54
Figura 37. Círculo para salir del laberinto y superar el nivel.....	54
Figura 38. Interfaz de fin de nivel con distintas puntuaciones .....	55
Figura 39. Comparación del primer y último nivel respectivamente.....	56
Figura 40. Logo completo sobre fondo claro.....	58
Figura 41. Símbolo o icono del juego .....	58
Figura 42. Logo completo en blanco y negro .....	58
Figura 43. Logo completo sobre fondo oscuro .....	58
Figura 44. Esquema de flujo de estados .....	61
Figura 45. Boceto pantalla de inicio.....	61
Figura 46. Interfaz pantalla de inicio.....	61
Figura 47. Boceto menú principal.....	62
Figura 48. Interfaz menú principal.....	62
Figura 49. Boceto H.U.D. ....	63
Figura 50. Interfaz del H.U.D. de los niveles sin VR.....	63
Figura 51. Boceto menú de opciones.....	63
Figura 52. Interfaz menú de opciones en el modo no VR .....	63
Figura 53. Interfaz menú de opciones en el modo VR .....	63
Figura 54. Boceto menú de pausa.....	64
Figura 55. Interfaz menú de pausa.....	64
Figura 56. Boceto fin de nivel .....	64
Figura 57. Interfaz fin de nivel.....	64
Figura 58. Boceto pantalla de carga.....	65
Figura 59. Interfaz pantalla de carga.....	65
Figura 60. Interfaz escena créditos .....	65
Figura 61. Boceto laberinto nivel 1.....	66
Figura 62. Vista aérea nivel 1.....	66
Figura 63. Vista inicial del nivel 1 con el panel de inicio .....	67
Figura 64. Vista inicial del nivel 1 sin panel.....	67

Figura 65. Boceto laberinto nivel 2.....	68
Figura 66. Vista aérea nivel 2.....	68
Figura 67. Vista inicial del nivel 2. Panel de inicio.....	69
Figura 68. Vista inicial del nivel 2 sin panel de inicio .....	69
Figura 69. Boceto laberinto nivel 3.....	70
Figura 70. Vista aérea nivel 3.....	70
Figura 71. Vista inicial del nivel 3 sin panel de inicio .....	70
Figura 72. Boceto laberinto nivel 4.....	71
Figura 73. Vista aérea nivel 4.....	71
Figura 74. Vista inicial del nivel 4 sin panel de inicio .....	71
Figura 75. Boceto laberinto nivel 5 .....	72
Figura 76. Vista aérea nivel 5 .....	72
Figura 77. Vista inicial del nivel 5 sin panel de inicio .....	73
Figura 78. Boceto laberinto nivel 6.....	73
Figura 79. Vista aérea del nivel6.....	73
Figura 80. Vista inicial del nivel 6 sin panel de inicio .....	74
Figura 81. Captura del nivel 6 desde móvil con el modo VR activado .....	74
Figura 82. Bocetos de objetos que forman el laberinto.....	82
Figura 83. Diseño de botones en Adobe Illustrator .....	82
Figura 84. Modelado de objetos en Blender.....	83
Figura 85. Reducción de caras de un objeto haciendo uso del operador Decimate. ....	83
Figura 86. Vista de Blender en el modo UV Editing. ....	84
Figura 87. Vista de Blender en modo Texture Paint .....	85
Figura 88. Creación del mapa de emisión en Photoshop.....	86
Figura 89. Modelo incluido en una escena en Unity.....	87
Figura 90. Detalles de un material en Unity.....	88
Figura 91. Método que actualiza la posición del menú de opciones.....	89
Figura 92. Función para guardar y desbloquear niveles .....	90
Figura 93. Objetos iniciales y comunes entre niveles .....	102
Figura 94. Objetos y adornos de los últimos niveles.....	103
Figura 95. Círculo de fin de nivel.....	103
Figura 96. Piedras del suelo del laberinto .....	104
Figura 97. Piedras que flotan sobre el laberinto.....	104

# Índice de tablas

Tabla 1. Planificación temporal TFG.....	18
Tabla 2. RF – 1. Seleccionar nivel .....	75
Tabla 3. RF – 2. Mirar o girar la cámara .....	75
Tabla 4. RF – 3. Caminar .....	75
Tabla 5. RF – 4. Parar de caminar.....	75
Tabla 6. RF – 5. Pausar juego.....	76
Tabla 7. RF – 6. Reiniciar el nivel .....	76
Tabla 8. RF – 7. Volver al nivel.....	76
Tabla 9. RF – 8. Salir de un nivel.....	76
Tabla 10. RF – 9. Avanzar al siguiente nivel .....	77
Tabla 11. RF – 10. Repetir nivel.....	77
Tabla 12. RF – 11. Ir al menú principal desde la pantalla de fin de nivel .....	77
Tabla 13. RF – 12. Desbloquear niveles.....	77
Tabla 14. RF – 13. Guardar partida .....	77
Tabla 15. RF – 14. Cargar partida .....	78
Tabla 16. RF – 15. Valorar nivel.....	78
Tabla 17. RF – 16. Cronometrar nivel.....	78
Tabla 18. RF – 17. Coger objetos.....	78
Tabla 19. RF – 18. Activar y desactivar realidad virtual .....	78
Tabla 20. RF – 19. Activar y desactivar sonido .....	79
Tabla 21. RNF – 1. Lenguaje de programación.....	79
Tabla 22. RNF – 2. Idioma del juego.....	79
Tabla 23. RNF – 3. Controles .....	79
Tabla 24. RNF – 4. Versión de Android.....	80
Tabla 25. RNF – 5. Tiempo real .....	80
Tabla 26. RNF – 6. Sencillo e intuitivo .....	80



# 1. Introducción

En la actualidad vivimos en una era cada vez más digital, donde la tecnología está presente en casi todos los aspectos de nuestro día a día. Todo el mundo dispone de un smartphone pero, es más, existen muchos otros objetos que podemos programar y nos acompañan en nuestra vida diaria, algunos ejemplos de estos podrían ser relojes inteligentes, electrodomésticos como lavadoras, arroceras, aspiradoras o bombillas capaces de cambiar de color e intensidad.

Por ello, según los datos del informe anual *Mobile Economy de la GSMA* [1,2], el número de conexiones móviles de tarjetas SIM alcanzó un número cercano a los 7.900 millones en 2018, llegando a superar a la población mundial. Y se prevé que este número no haga nada más que aumentar y más con la nueva incorporación de la tecnología 5G, llegando a hasta 9.200 millones de conexiones de tarjetas móviles y 25.200 millones de conexiones producidas por otros dispositivos, *Internet of Things* (IoT).

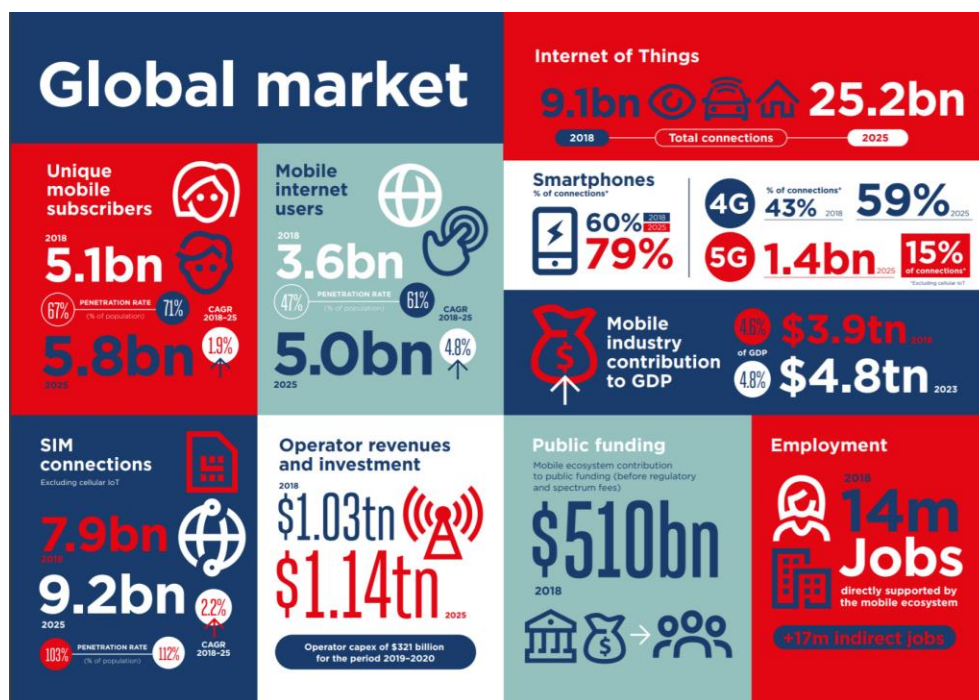


Figura 1. Infografía de informe anual de la GSMA

Por tanto, como no puede ser de otra forma, el mercado de las aplicaciones y los juegos para smartphones tampoco ha dejado de aumentar en estos años. A día de hoy, existen aplicaciones que nos ayudan a llevar a cabo todo tipo de tareas como puede ser cocinar, comprar, llevar

rutinas de ejercicio, aprender idiomas o guiarnos por calles y carreteras; algunas de ellas incluso han llegado a considerarse imprescindibles en la vida diaria de muchas personas.

Igualmente, existen numerosos juegos que buscan entretener y divertir a las personas, y es justo en esta categoría donde se incluirá este trabajo. Aunque también añade otra gran tendencia actual, la *Realidad Virtual (RV o VR en inglés)*.

Hoy en día, la realidad virtual empieza a entrar en nuestras vidas, dándonos la oportunidad de vivir experiencias hasta el momento imposibles y aportándonos nuevas formas de aprender y jugar. Existen numerosos ejemplos de juegos y aplicaciones que usan esta tecnología como comentaremos en el siguiente apartado de este documento.

Celestial Road es un juego para dispositivos móviles que plantea retos en forma de laberintos y puzzles a sus jugadores, incluyendo un aspecto diferenciador, dará al usuario la opción de disfrutar de estos retos en primera persona haciendo uso de realidad virtual. Para ello, el usuario solo necesitará disponer de unas gafas Cardboard u otro tipo de visor de realidad virtual para poder disfrutar de esta experiencia. De esta forma, se crea un entorno virtual en el que el usuario pueda disfrutar de estos retos de una forma más inmersiva e interactiva.

La decisión de desarrollar para móviles y hacer uso de las Cardboard, se tomó intentando llegar al mayor público posible ya que, otros dispositivos mejores de realidad virtual, como las *Oculus Rift* o las *HTC Vive* (explicadas en el apartado 3.1.3 del documento), son menos asequibles para la mayoría de las personas debido a su elevado coste.

El público objetivo al que se dirige este proyecto es bastante amplio ya que se componen de personas, mayores de 6 años debido al uso de la realidad virtual, a las que les gusten los laberintos o simplemente estén interesados en probar nuevas experiencias en realidad virtual.

## 2. Planificación

Para lograr llevar a cabo este trabajo y realizar su defensa en la convocatoria elegida, en este caso la C4, septiembre, se realizó una planificación en la que se establecieron los plazos de tiempo disponibles para cada tarea, así como las fechas límites en que estas se debían finalizar para lograr completar el proyecto de forma exitosa.

En la *Tabla 1* se puede observar la planificación con la que se ha trabajado, teniendo en cuenta que este trabajo contaba con un plazo de aproximadamente tres meses, 14 semanas para ser más exactos, y se dedicó una media de 25 horas semanales de trabajo.

Con todo ello, el trabajo se ha conseguido finalizar de forma exitosa según la planificación prevista y el tiempo dedicado ha cumplido y superado el especificado en el plan de estudios de esta carrera, Ingeniería Multimedia, en el que se estiman una cantidad de 12 *ECTS* para este trabajo, lo que se corresponde con aproximadamente 300 horas de trabajo.

*Tabla 1. Planificación temporal TFG*

<b>Contenidos</b>	<b>Tiempo total</b>	<b>Fecha límite fin</b>
Motivación, justificación, objetivo general Introducción Planificación Objetivos	15 días	12 junio
Estado del arte Herramientas utilizadas	20 días	2 julio
Metodología Documento de diseño del videojuego (GDD) Requisitos funcionales y no funcionales	23 días	25 julio
Implementación	25 días	20 agosto
Pruebas y validación Conclusiones Trabajo futuro Referencias, bibliografía y apéndices Agradecimientos, citas, índices	12 días	2 septiembre

## 3. Estado del arte.

En este apartado se explicarán los antecedentes tecnológicos del proyecto, como la aparición de la realidad virtual, los dispositivos utilizados actualmente y algunos juegos y aplicaciones que hacen uso de realidad virtual y Cardboard para ofrecer nuevas experiencias o conocimientos a sus usuarios. Además, se comentarán los principales motores gráficos y las herramientas que se utilizarán en este proyecto.

### 3.1. Realidad Virtual

Para comenzar hablaremos de la realidad virtual y de cómo ha avanzado y evolucionado a lo largo de los años.

#### 3.1.1. ¿Qué es la realidad virtual?

La realidad virtual es la representación de escenas o imágenes de objetos producida por un sistema informático, que aporta al usuario la sensación de que su existencia es real. Este entorno es contemplado por el usuario a través de un dispositivo conocido como gafas o casco de realidad virtual que puede complementarse con otros dispositivos como mandos, guantes o trajes especiales que permiten mayor interacción con el entorno. [3]

El objetivo de la realidad virtual es engañar a los sentidos del usuario, para ser percibido como un mundo real, para lo cual es necesario:

- **Existencia de un entorno virtual**, normalmente tridimensional, que posea una colección de objetos y un conjunto de reglas y relaciones que controlen el comportamiento de dichos objetos.
- **Simulación en tiempo real**, se requiere una interacción y retroalimentación inmediatas, lo que requiere unos gastos computacionales muy altos, debido a que se debe muestrear en todo momento las acciones, la posición y orientación del usuario para recalcular todo el mundo según las nuevas condiciones y generar la salida necesaria para todos los dispositivos.
- **Interacción a través de diferentes canales sensoriales**, el usuario debe poder navegar por el mundo virtual, aunque esté sujeto a las restricciones definidas por el creador del

mundo. Es importante evitar la sensación de mareo que se produce si el mundo se mueve sin que se mueva el usuario.

- **Inmersión física o psicológica**, hace creer al usuario que forma parte de la experiencia virtual, para ello, hace falta enfocar la atención del usuario en el mundo virtual y conseguir darle sensación de presencia.
- **Retroalimentación sensorial continua**, el sistema debe aportar respuestas a los sentidos del usuario cada vez que este se mueva y realice alguna acción, como pueden ser sonidos producidos por acciones o simplemente seguir la posición y orientación de la cabeza del usuario con el tracking de la cabeza.

Lo principales **usos** de la realidad virtual van mucho más lejos de lo que mucha gente se cree, no solamente busca el entretenimiento y el disfrute personal, también se usa para otras finalidades como por ejemplo la educación, entrenamiento (por ejemplo, para: soldados, conductores, estudiantes de medicina...), para el tratamiento de fobias, educación especial para niños con características especiales, rehabilitación o visitar sitios que en condiciones normales no sería posible.

### 3.1.2. Evolución de la realidad virtual

En la actualidad todos conocemos qué es la realidad virtual, o por lo menos la relacionamos con algunas de sus características básicas, como la necesidad de un entorno o mundo virtual que busca hacernos creer que es real y con el que podemos interactuar. Pero esto no siempre fue así.

Estos conocimientos y nociones han llegado a nosotros mediante muchas obras de ciencia ficción, en las cuales la realidad virtual está claramente presente y es un tema recurrente; y por las noticias de avances de grandes compañías en este sector, unidas a los nuevos productos lanzados al mercado durante los últimos años.

Sin embargo, la historia de la realidad virtual se remonta a mucho antes. Uno de los primeros hitos de esta historia consta en **1836**, cuando Charles **Wheatstone** inventó el **estereoscopio** (figura 2), un instrumento que permitía crear una ilusión de profundidad en una imagen a partir de dos fotografías prácticamente idénticas, que al ser observadas una por cada ojo de forma independiente conseguían generar una sola imagen estereoscópica en nuestro cerebro y por tanto la ilusión de profundidad. Y aunque dista mucho de ser considerado un sistema de realidad virtual, fue el primer patrón para el diseño de los primeros visores de Realidad Virtual.



Figura 2. Ejemplo de un estereoscopio

Tras esto, aunque se desarrollaron algunos simuladores de uso militar para entrenar soldados como fue el Blue Box, no hubo auténticos avances públicos hasta **1962** cuando **Morton Heilig** estrena su obra *“experiencia teatral”*, en la que intenta abarcar todos los sentidos del espectador y atraer su atención a la pantalla.

Para lograr este objetivo crea un simulador multisensorial llamado **Sensorama** (figura 3), en este simulador se podían visualizar películas mientras se estimulaban el resto de los sentidos mediante la proyección de la película en color, el uso de sonidos estéreos, olores, viento y vibraciones. Aunque todavía no era interactivo y por tanto no es considerado un simulador de realidad virtual. [4]

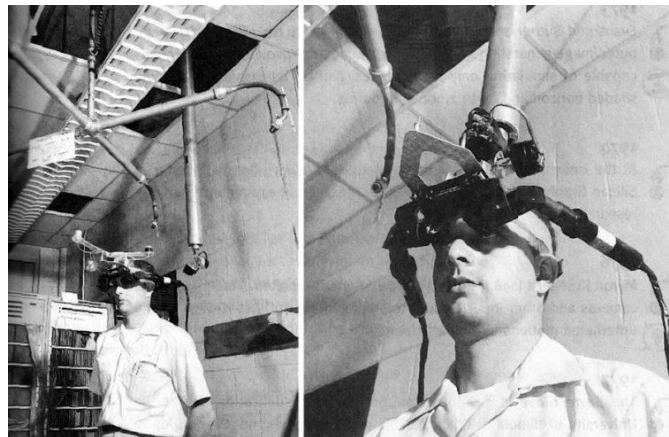


Figura 3. Sensorama

Los siguientes hitos que marcaron la evolución de esta tecnología fueron protagonizados por **Ivan Sutherland**, quien en **1965** propuso la solución definitiva para crear realidad virtual, **“The Ultimate Display”**, un concepto que construía el mundo real de forma artificial con gráficos interactivos y retroalimentación para el usuario. Y en **1966** creó **“La Espada de Damocles”** (figura 4) el **primer HMD (Head Mounted Display)**, estaba formado por tubos de rayos catódicos en un armazón de alambre que pendía del techo.

Esta pantalla que se colocaba sobre la cabeza disponía de seguimiento en tiempo real que se actualizaba correctamente de acuerdo con la posición y orientación de la cabeza de usuario. Dado que superponía imágenes sobre un fondo real también sentó las bases de la realidad aumentada.

Además, en 1968 Sutherland y Evans crean el primer generador de escenarios **3D** y fundan su propia sociedad.



*Figura 4. Espada de Damocles, primer HMD*

Durante los siguientes años van surgiendo muchos avances en los campos de la realidad virtual, como la aparición de *Sketchpad*, primer programa de *CAD*, o la creación de simuladores y nuevos aparatos de VR como guantes y visores.

Entre estos avances podemos destacar el simulador de vuelo contenido en un casco de **Thomas Furness, The Super Cockpit (1980)**, que permitía controlar un avión mediante gestos, palabras o movimientos oculares, además de proporcionar datos de aeronáutica en un espacio tridimensional en tiempo real. [5]

**VIVED** (*Virtual Visual Environment Display*) construido en la **NASA** en **1984**, con la tecnología *off-the-shelf*, se trataba de un HMD monocromo estereoscópico que incorporaba control por voz y un sistema de reconocimiento de gestos por guantes (figura 5). [7]



Figura 5. VIVED - 1984

En **1992** se presenta el sistema **CAVE** en el **SIGGRAPH**, conferencia organizada por el grupo de computación gráfica **SIGGRAPH**, este dispositivo era un sistema de visualización científica combinada con VR. En lugar de usar un HMD, proyecta imágenes estereoscópicas en las paredes de una habitación. [6]

Poco a poco las grandes empresas de videojuegos se unen a esta nueva tendencia tecnológica, **Sega** intentó lanzar **Sega VR**, pero nunca llegó al mercado. Por otro lado, **Nintendo** en **1995** lanza la "**Virtual Boy**" (figura 6) con gráficos 3D de 32 bits en colores rojo y negro que, aunque si vio la luz también fue un fracaso debido a que era demasiado voluminoso y frágil. Con el tiempo también sacaron sus productos otras empresas de videojuegos como **PlayStation**, aunque fue poco conocido, y **Xbox** que lanzó el **Kinect**.



Figura 6. Consola Virtual Boy de Nintendo



Ya en **2012** aparecieron las gafas de realidad virtual, **Oculus Rift**, que lograban un ángulo de visión de 90°, esto llamó la atención de *Facebook* quien en 2014 compró la empresa y ha seguido con su desarrollo hasta la actualidad. Y aunque surgieron otros dispositivos como el *Leap Motion* o las *Google's Glass*, fue gracias a las *Oculus* que la realidad virtual volvió a ser tendencia y desde entonces muchas empresas han creado sus propios dispositivos como comentamos en el siguiente apartado de este trabajo.

### 3.1.3. Dispositivos actuales de Realidad Virtual

Como se ha comentado, a lo largo de los años han existido muchos intentos de desarrollo de sistemas de VR, pero siempre han terminado en fracaso. Sin embargo, gracias a los avances tecnológicos se han conseguido crear nuevos dispositivos más inmersivos y con mayor atractivo para el público.

Gracias a estos avances, así como al apoyo por parte del mundo cinematográfico desde hace algunos años con películas como *Matrix* o más reciente *Ready Player One*, el mundo de la realidad virtual no deja de avanzar y grandes compañías están apostando para lograr hacerse con el liderazgo.

Actualmente hay una gran cantidad de alternativas que se adaptan al consumidor tanto por calidad, precio y plataforma. A continuación, se explican algunos de los principales dispositivos de VR disponibles en la actualidad.

#### ***Oculus Rift***

Las gafas de *Oculus* [8] poseen dos pantallas **OLED** de **1080x1200** lo que hace que la realidad virtual sea totalmente inmersiva con muy buena visualización. Disponen de auriculares que proporcionan un **sonido envolvente 3D**.

Para conseguir saber la posición del usuario en todo momento, el casco y los mandos tienen una serie de sensores que llaman “constelación”, debido a que son muchas luces infrarrojas colocadas alrededor del visor y los mandos. Estos envían información a otros sensores que se encuentran conectados al ordenador, donde se procesan los datos.

Dependiendo del número de sensores el espacio de juego varía, con dos sensores el jugador se puede mover por un espacio de 1.5m<sup>2</sup>, si dispone de tres el espacio aumenta a 2.5m<sup>2</sup>.

Estas gafas inicialmente **necesitan** un **ordenador** potente para su funcionamiento, pero tras un tiempo han ido lanzando nuevas **alternativas** que no necesitan un pc para su funcionamiento como **Oculus GO** u **Oculus Quest**, más potente que el anterior.

Las *Oculus* son compatibles con los mandos *Oculus Touch*, estos están compuestos por un *joystick* analógico, dos botones en la parte superior y dos gatillos que simulan a la perfección la acción de coger algo. Además, poseen unos sensores que detectan cuando levantas los dedos de encima y se ve como abres las manos en realidad virtual. [9]

La versión para el consumidor se lanzó al mercado entre los meses de marzo-abril de 2016 con un precio de 599\$ en EE. UU. y con un precio de 699€ en Europa. Aunque actualmente se pueden conseguir junto con los *Touch Controllers* o **Oculus Touch** (Figura 7) a un precio de **449€**.



Figura 7. Oculus Rift

### **HTC Vive**

Estas gafas han sido desarrolladas por **HTC** junto con la desarrolladora de juegos **Valve**. Existen dos modelos ya que se ha lanzado una versión *Pro* que mejora algunas características. Ambas **necesitan un ordenador** para su funcionamiento, con unas características algo superiores a las necesarias para el uso de las *Oculus*.

La versión normal de estas gafas (figura 8), al igual que las anteriores, disponen de dos pantallas **OLED** con la resolución **1080x1200**, aunque la versión **Pro** mejora esta resolución con pantallas de **1440x1600** consiguiendo la mejor resolución de todas las gafas de realidad virtual que se encuentran en el mercado.

Por otro lado, el tracking que ofrecen las *HTC Vive* [10] proporcionan un espacio mayor de juego de hasta 4.5m2.

Los mandos o controladores cuentan con 24 sensores para saber su posición en todo momento, panel táctil multifunción, un gatillo y aporta retroalimentación táctil HD con cada acción.

Con un precio aproximado de 581€ solo las gafas y llegando a una cifra de **1357€** la versión *HTC Vive Pro* con sus accesorios (Figura 9).



Figura 8. Gafas HTC Vive



Figura 9. HTC Vive Pro

### ***Sony PlayStation VR***

**Sony** también quiso unirse a esta nueva tendencia y con ello ofrecer una nueva forma de jugar, el resultado de este trabajo fue *PlayStation VR* (figura 10) [11] un ligero y cómodo casco que

funciona **conectado** a la consola **PS4**. El principal propósito de estas gafas era permitir a sus usuarios entrar en los videojuegos y poder ser el auténtico protagonista de estos.

Estas gafas aportan una **visión** de **360** grados y un **sonido envolvente** en 3D para poder crear la mejor sensación de inmersión. Además dispone de micrófono para poder comunicarse con otros jugadores.

Para completar la experiencia se puede usar los Mandos *PlayStation Move* para poder incluir las manos en el juego o el *Aim Controller* para juegos tipo *Shooter*. Además, permiten tanto jugar los juegos adaptados como los no adaptados visualizando una pantalla a través del casco.

Si se dispone de una *PS4*, este casco es muy buena opción ya que aporta nuevas formas de jugar a un precio no demasiado elevado, entorno a **240€**.



Figura 10. Gafas Sony PlayStation VR

### ***Samsung Gear VR***

Se trata de un dispositivo que ha sido **resultado** de la **colaboración** entre **Oculus** y el fabricante **Samsung**. Es un dispositivo tipo carcasa (figura 11) que gracias a los botones y sensores que incorpora permite una experiencia virtual única. **No** tiene **pantalla** como los anteriores, por lo que para su uso es **necesario** disponer de un **teléfono Samsung Galaxy** a partir del modelo S6 que sirva como pantalla.

Las *Gear VR* [12] están construidas en plástico de alta resistencia, dispone de un par de botones en el lateral derecho y un panel táctil para la navegación. Aunque con la introducción del mando

en las últimas versiones esto ya no es necesario (Figura 12). En la parte superior está la rueda que aleja o acerca el móvil a tus ojos para lograr la imagen lo más nítida posible para cada usuario.



Figura 11. Samsung Gear VR

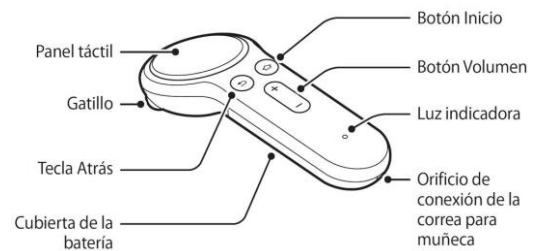


Figura 12. Controles mando Gear VR

## **Google Cardboard**

Se trata de una **carcasa de cartón** (figura 13) que, **junto** con un **Smartphone**, permite conseguir una realidad virtual doméstica bastante auténtica. Su mayor punto fuerte es que son muy **baratas** y han servido de inspiración para la creación de muchas carcasas de realidad virtual que actualmente se pueden comprar en cualquier tienda.

Por otro lado, tienen una **funcionalidad** muy **limitada** que se reduce a poder ver el mundo en realidad virtual, el tracking de la cabeza y poder hacer clic moviendo ligeramente los imanes que tienen incorporados.

Las gafas diseñadas por **Google** están compuestas por un trozo de cartón troquelado, un par de velcros para sujetar la parte frontal, dos imanes (que sirven como activador o botón) y dos lentes de 40 mm de distancia focal. En algunos modelos se incluyen una cinta para sujetarlas a nuestra cabeza, y unas pegatinas *NFC*. [13]



Figura 13. Google Cardboard

### 3.1.4. Conclusión

En este trabajo hemos decidido **centrarnos** en este último dispositivo, las **Cardboard**, debido a su bajo coste y a que para su uso solo necesitas tener las gafas y un smartphone, no es necesario ni un ordenador ni una consola. En la actualidad casi todo el mundo tiene ambas cosas a su **disposición** y por tanto creo que, de esta forma, un mayor número de personas podrán disfrutar de este juego.

## 3.2. Análisis de mercado

En la actualidad existen una gran cantidad de **aplicaciones** para móvil que hacen **uso** de la **realidad virtual** para ofrecer nuevas experiencias a los usuarios.

Para poder disfrutar de ellas, solo es **necesario** disponer de un **teléfono** móvil con una versión relativamente actual de Android o IOS, tener instalada la tienda de aplicaciones (GooglePlay o App Store) desde la que poder descargarse las aplicaciones y a ser posible que el smartphone disponga de giroscopio para poder llevar a cabo todas las acciones disponibles en los juegos y aplicaciones; además, es necesario un **visor** con el que poder ver la realidad virtual, **como** pueden ser las **Cardboard**.

Las aplicaciones actuales de la realidad virtual abarcan numerosos campos distintos desde el entretenimiento, al turismo, la educación e incluso la medicina. Por ello, no podían faltar aplicaciones ni juegos que cubran todas estas posibilidades.

A continuación, se citan algunos **ejemplos** de aplicaciones y juegos que hacen uso de la realidad virtual y han triunfado y recibido gran aceptación entre los usuarios. Para este trabajo se han

probado todas ellas para buscar inspiración y para observar que tipo de interacciones proponían al usuario.

### 3.2.1. Principales juegos y aplicaciones

#### ***Sites in VR***

Esta aplicación [14] permite a los usuarios visitar distintos sitios y monumentos del mundo de forma inmersiva e interactiva sin necesidad de desplazarse.

De esta forma, da la oportunidad de **conocer** nuevos **lugares** y **monumentos** de forma rápida y sencilla, ayudando tanto a personas que no puedan o quieran desplazarse hasta ellos como a aquellas que quieran estudiar o examinar la arquitectura de algún monumento o la flora de los parques naturales, entre muchas otras opciones.

*Sites in VR* ha conseguido más de un millón de descargas y mantiene una puntuación de 4,5 sobre 5 en *GooglePlay*, lo que demuestra que ha tenido gran aceptación entre sus usuarios y ofrece muy buenas experiencias.

Cuenta con una gran cantidad de lugares para visitar y da al usuario la opción de elegir el visor que quiere utilizar para ajustar la imagen de la visita a este y que ofrecer la mejor experiencia posible.

Su interacción es bastante simple ya que solo debes mirar hacia un lado y otro para poder observar la escena que te rodea (figura 14).



Figura 14. Imágenes de la aplicación Sites in VR

## ***InMind VR***

*InMind* [15] es una breve **aventura** diseñada para *Google Cardboard* que permite al jugador experimentar un **viaje al cerebro** del paciente en busca de neuronas que causan algún trastorno mental. Ha conseguido más de un millón de descargas y mantiene una puntuación de 4,2 sobre 5 en la tienda de *GooglePlay*.

Es un juego simple pero muy entretenido, orientado a un público infantil o joven, que a la vez que enseña como es el espacio cerebral de las neuronas, busca entretener al jugador haciéndole disparar a aquellas neuronas que causan algún trastorno mental al paciente.

Tiene unos gráficos bastante buenos y la **interacción** con la realidad virtual funciona muy bien, consiguiendo una sensación de inmersión buena sin causar mareos al jugador a pesar de estar **moviendo** la **cabeza** para disparar a las neuronas.

Su interacción es simple e intuitiva, solo debes mirar hacia los lados para observar las neuronas y mantener la vista en las neuronas “malas” para conseguir eliminarlas (figura 15).



Figura 15. Imágenes de *InMind VR*

## ***El cuerpo humano en 3D***

Siguiendo con otra **aplicación educativa**, nos encontramos con *El cuerpo humano en 3D* [16], una aplicación diseñada para un público de entre 8 y 18 años y disponible en 17 idiomas diferentes.

Esta aplicación permite **ver** el **cuerpo humano**, en este caso el masculino, en **3D** y en realidad virtual. Dando la oportunidad de ver y conocer todas las partes del cuerpo desde la piel a los músculos, huesos, sistemas digestivos, respiratorio, circulatorio, etc. (figura 16).

Por otro lado, tiene **ejercicios para que el usuario** pueda relacionar los músculos, huesos y órganos con sus nombres, lo que la convierte en una aplicación muy completa y educativa. Esto



se ve reflejado en sus valoraciones ya que cuenta con una puntuación de un 4,7 sobre 5 en la tienda de *GooglePlay* y muy buenos comentarios entre sus más de 25 mil valoraciones. Además, de un gran número de descargas que sobrepasa el millón.

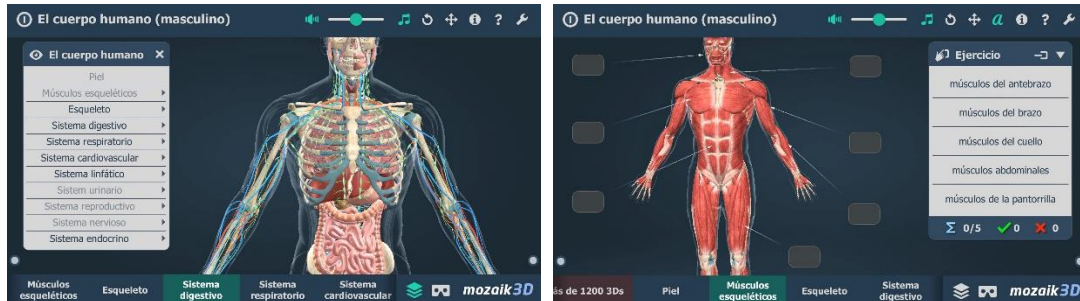


Figura 16. Imágenes de la aplicación *El cuerpo humano en 3D*

### ***Jurassic VR – Google Cardboard***

*Jurassic VR* [17] crea una nueva experiencia donde el jugador tendrá la oportunidad de sumergirse en un nuevo **entorno** donde enormes **dinosaurios** cobran vida y caminan junto a ti.

Esta aplicación permite ver muchas razas distintas de dinosaurios y seguir lo que habría sido su día a día, desde pasear a cazar a sus presas. Aunque está disponible para todos los públicos se recomienda que solo la usen mayores de 12 años debido al realismo y violencia de algunos momentos.

Los **controles** son muy sencillos ya que solo hace **uso** del **giroscopio** para mirar y caminar hacia la dirección en la que se está mirando. Con estos simples controles ya puedes recorrer el parque y observar los dinosaurios en su hábitat.

Esta aplicación ha recibido más de 5 millones de descargas ya que aporta una nueva experiencia y una forma sencilla de conocer a estos seres vivos ya desaparecidos. Cuenta con una puntuación de 3,7 sobre 5 en la tienda de *GooglePlay* con más de 16 mil valoraciones.



Figura 17. Imágenes de la aplicación Jurassic VR – Google Cardboard

### ***Google Spotlight Stories***

Pasando más a la parte de entretenimiento y dejando a un lado el sector educativo, nos encontramos con infinidad de aplicaciones que buscan entretener y divertir al usuario. A continuación, se comentarán algunas que llamaron mi atención y la de otros usuarios teniendo en cuenta sus puntuaciones y número de descargas, comenzando por *Google Spotlight Stories* [18].

*Google Spotlight Stories* (figura 18) es una **plataforma de historias envolventes**, videos 360º, la mayoría disponibles en realidad virtual para visualizarse con un visor como las Cardboard.

Esta aplicación es ganadora del premio *Emmy* a la innovación en programación interactiva de 2017 y ha sido nominada a dos premios *Emmy* más y a los *Óscar*. Cuenta con más de un millón de descargas y mantiene una puntuación de 4,2 sobre 5 en la tienda de *GooglePlay*.

Cuenta con todo tipo de historias y animaciones, desde videos musicales animados hasta cortos para los más pequeños. Para mi gusto una gran aplicación con videos muy bonitos y bien trabajados.

En *web* de la plataforma [19] se puede ver el resumen de muchas de sus historias y el enlace al video, en caso de que se encuentre disponible en *YouTube*.

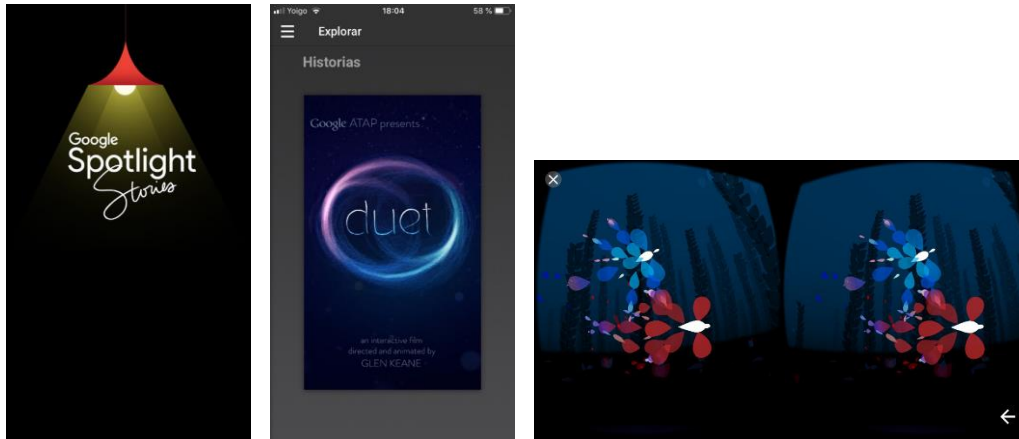


Figura 18. Capturas de la aplicación de Google Spotlight Stories

### ***Within***

*Within* [20] (figura 19) al igual que la aplicación anterior es una plataforma de **historias** y videos **inmersivos**, pero en esta todos los videos están disponibles para VR. Contiene desde relatos emocionantes en mundos de fantasía hasta documentales y noticias.

Tiene más de un millón de descargas y una puntuación de 4 sobre 5 en la tienda de aplicaciones, *GooglePlay*.

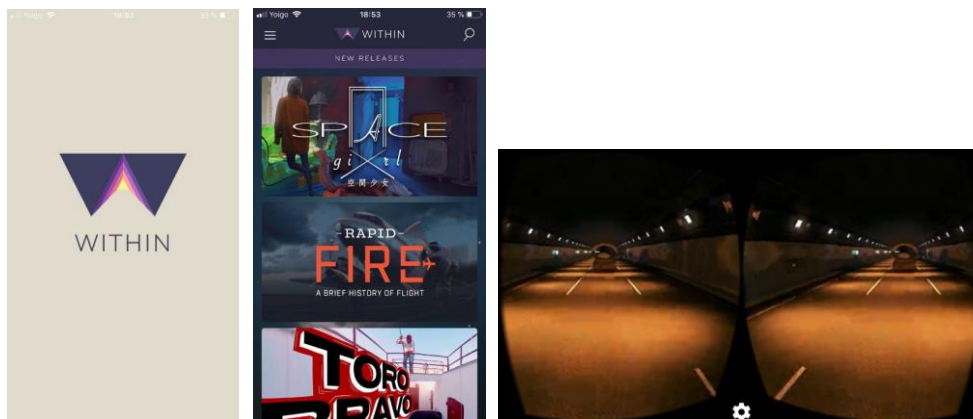


Figura 19. Capturas de la aplicación Within

### ***VR Cave***

En el sector del entretenimiento no solo hay aplicaciones de videos inmersivos, también existen muchos juegos como son los de escapismo, en los que el protagonista se encuentra encerrado en algún lugar y debe conseguir salir o los de exploración, en los cuales el protagonista se

encuentra en algún entorno fantástico y desconocido y debe explorarlo. Un ejemplo de este último tipo es el juego del que vamos a hablar a continuación, *VR Cave* [21].

En esta aplicación el **usuario** tendrá la oportunidad de **explorar** una **cueva** con todo lujo de detalles, aunque las interacciones son muy limitadas ya que el usuario solo puede mirar y caminar por la cueva, el entorno está muy logrado y la experiencia de inmersión es bastante buena (figura 20).

Por todo ello la aplicación ha conseguido más de un millón de descargas y una puntuación de 4 sobre 5 en la tienda de aplicaciones.

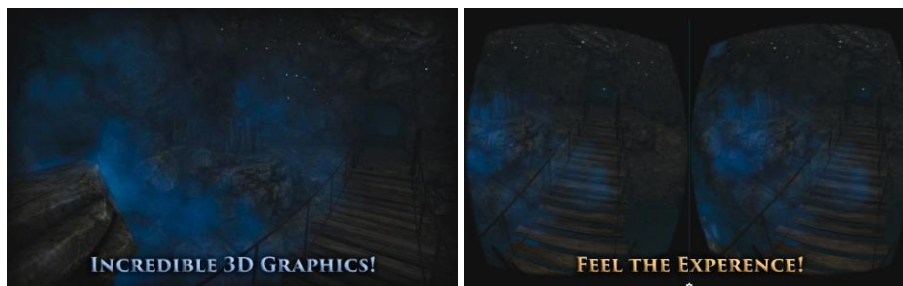


Figura 20. Imágenes de VR Cave

### ***VR Thrills: Roller Coaster 360 (Google Cardboard)***

Otro de los estilos de juego que más ha triunfado en el mundo de la realidad virtual para móvil han sido los **juegos de montañas rusas**, en los cuales el protagonista se encuentra montado en el vagón de una montaña rusa y puede ver todo el recorrido en 360º (figura 21).

En concreto *VR Thrills Roller Coaster 360* [22] cuenta con la simulación de un gran número de montañas rusas diferentes, no solo por su recorrido sino también por todo el entorno que las rodea.

Da la oportunidad de jugar tanto en modo táctil como en VR, incluso sin giroscopio. Esto junto con su gran sensación de inmersión con la cual crees realmente estar cayendo en picado a gran velocidad o girar hacia un lado u otro han hecho que esta sea una aplicación muy demandada con más de 10 millones de descargas y mantenga una buena puntuación de 3,9 sobre 5.

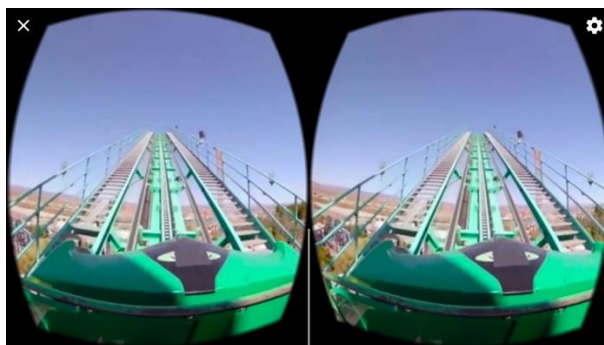


Figura 21. Imágenes de VR Thrills: Roller Coaster 360

### **VR Roller Coaster Temple Rider**

Otro ejemplo de lo que comentábamos en el punto anterior es *VR Roller Coaster Temple Rider* [23], una **montaña rusa** que te llevará por **templos**, **cuevas** y mundos submarinos (figura 22).

Ofrece un entorno virtual en 360 grados muy detallado además de una gran sensación de inmersión que te hará creer que realmente estas allí.

Cuenta con 5 millones de descargas y una puntuación de 4,1 sobre 5, además de unos comentarios realmente favorables en la tienda de aplicaciones.

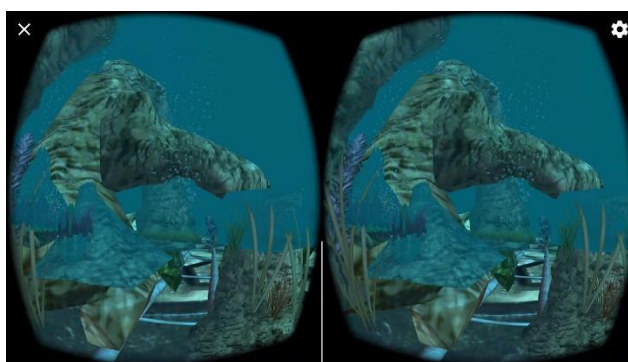


Figura 22. Imágenes de VR Roller Coaster Temple Rider

### 3.2.2. Competencia y referentes

Aunque ya se han comentado algunas de las aplicaciones que se han probado y han servido de inspiración para decidir el estilo del juego y el tipo de interacciones que son posibles en realidad virtual para móvil. A continuación, se comentan algunas aplicaciones que han servido como ejemplo y que podrían considerarse **competencia**, ya que siguen la **misma temática** y objetivo

que el juego desarrollado en este proyecto; y pueden verse como referentes debido a que han ayudado a tomar algunas decisiones de interacciones, estética y funcionalidades del proyecto.

Si bien, aunque como ya hemos comentado, existen otros juegos para móvil de **laberintos en realidad virtual**, **ninguno** de ellos ha tenido una **gran acogida** entre el público, no presentan un gran número de descargas, de los encontrados pocos tienen más de cien mil descargas y sus valoraciones no son demasiado buenas.

Algunos ejemplos de estos juegos son *Laberinto VR Maze* [24], *Maze VR Forest* [25] o *Maze VR – Cardboard* [26]. Todos ellos poseen una estética simple y muy poco realista que, aunque no afecta a la inmersión si causa cierto rechazo por parte de los usuarios (figura 23).



Figura 23. Imágenes de GooglePlay de “Laberinto VR Maze” y “Maze VR – Cardboard” respectivamente

Por otro lado, una de las **críticas** que más se repite en todos ellos es la sensación de **mareo** que producen al andar por el laberinto. No solo por la inmersión en sí, sino también porque los laberintos eran demasiado grandes y complicados.

Hablando del tipo de interacciones presentes en ellos, todos se basaban en mirar hacia los lados para elegir la dirección hacia la que avanza el personaje y por tanto para **todos** ellos es **necesario** disponer de un **móvil con giroscopio**. Aparte de esto, también se hace uso de los imanes de la Cardboard para seleccionar opciones en los menús o, por ejemplo, en segundo juego mencionado para caminar y detener al personaje.

No obstante, aunque los juegos mencionados no han tenido demasiado **éxito**, sí que existen muchos **juegos de laberintos en 3D**, pero **sin realidad virtual**, que han conseguido ser muy descargados y conseguir buenas valoraciones.

Algunos **ejemplos** son *3D Maze (The Labyrinth)* [27] con más de 5 millones de descargas y una puntuación de 4 sobre 5 en la tienda de GooglePlay. Este juego incluye monedas y llaves que el jugador debe recoger para poder salir del laberinto, además de un cronómetro que te va avisando del tiempo que tardas en finalizar cada nivel. Por último, dispone de un modo

multijugador para lograr enganchar y entretener más al usuario haciéndole competir contra otros jugadores.

*Maze World 3D* [28] que cuenta con un número de descargas y puntuación similar al juego anterior. Al igual que el anterior cuenta con llaves, monedas y tiempos en cada nivel, con estos calcula una puntuación basada en estrellas que es comunicada al usuario al final de cada nivel y representada en el menú inicial (figura 24).

Y el último que vamos a comentar, *Laberinto 3D II* [29] que cuenta con más de 10 millones de descargas y una puntuación de 3,7 sobre 5 de media de sus más de 88 mil valoraciones. Este juego cuenta con características similares a los anteriores, pero añade personajes seleccionables, monstruos que te impiden el paso y atacan; y *powers ups* que ayudan al jugador a sobrevivir y completar en el nivel correctamente.

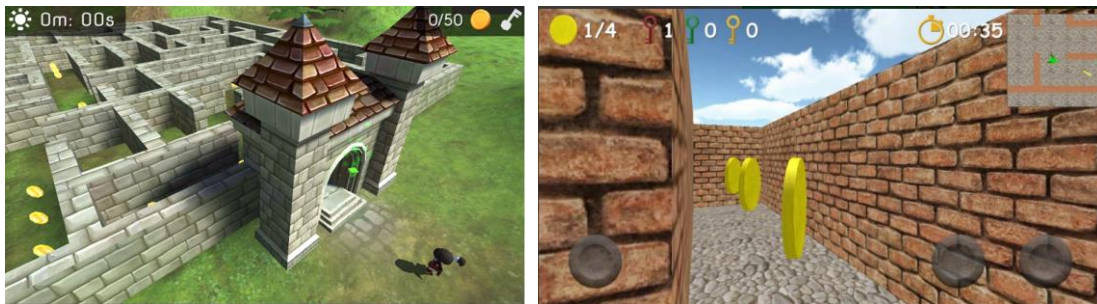


Figura 24. Imágenes de GooglePlay de “3D Maze (The Labyrinth)” y “Maze World 3D” respectivamente

El éxito de estos juegos demuestra que la temática sí es demandada por los jugadores, pero sin embargo los juegos publicados hasta el momento que la unen con la realidad virtual no han conseguido captar la atención de estos usuarios debido a su estética, complejidad o mala inmersión.

## 4. Herramientas utilizadas

### 4.1. Motores de videojuegos

Para la realización de este proyecto se ha hecho uso de un motor de videojuegos, en nuestro caso *Unity 3D*. A continuación, explicaremos el porqué del uso de un motor gráfico, en que consiste un motor gráfico, que ventajas nos aporta y por qué hemos elegido *Unity* para nuestro desarrollo.

#### 4.1.1. ¿Qué es un motor de videojuegos?

Un motor de juegos o *game engine* es un **entorno de desarrollo software** que brinda a los creadores de juegos todas características necesarias para crear juegos de una manera rápida y eficiente para todo tipo de plataformas.

Un motor de juegos es un entorno para los desarrolladores que reúne varias funcionalidades básicas, puedes **importar** arte y **assets**, en *2D* y *3D*, de **otros** tipos de **software**, tales como *Maya*, *Blender* o *3ds Max* o *Photoshop*; ensamblar estos *assets* para formar escenas y entornos; agregar iluminación, audio, efectos especiales, físicas, colisiones, animaciones, interacción y lógica de mecánica del juego, que programando sin un motor sería muchísimo más complicado y requeriría mucho tiempo.

Además, todos los motores de juegos incluyen una representación de motor o “*render*” para poder mostrar los gráficos al usuario y funciones para editar, depurar y optimizar el contenido para las plataformas de destino de los juegos.

Algunas de las características que deben tenerse en cuenta a la hora de elegir un motor gráfico son sus capacidades gráficas para el estilo de gráficos que queremos (*2D*, *3D*), aspectos como los polígonos, la iluminación, la facilidad para exportar juegos a diferentes plataformas o la facilidad de aprendizaje para el uso del motor. [30, 31, 32]



## 4.1.2. Principales motores en la actualidad

### ***Unreal Engine 4***



Figura 25. Logo de Unreal Engine y vista de su interfaz

*Unreal Engine* [33] (figura 25) fue creado por *Epic Games* en 1998 y en 2012 se presentó la versión actual *Unreal Engine 4*. Su principal lenguaje de programación es C++.

*Epic Games* permite el uso de su plataforma de forma inicialmente gratuita, aunque si tu empresa gana más de 3.000 dólares por producto al trimestre, pide un 5% de las ganancias generadas.

Un ejemplo de grandes compañías que hacen uso de este motor son *Electronic Arts* o *Ubisoft*. Y ofrece la posibilidad de hacer cosas tan impresionantes como *Batman Arkham Knight*

Su mayor inconveniente es que hace falta dedicarle muchas horas para conseguir familiarizarse a su entorno y editor.

Por otro lado, ofrece el control completo de su *engine* ya que ofrece su código de forma abierta y gratuita para que los usuarios puedan hacer las mejoras que quieran. Otro aspecto muy interesante y muy a tener en cuenta de *Unreal Engine* es la opción de programación basada en nodos y componentes, que facilita el trabajo a los no programadores y hace el prototipado muy rápido. Este sistema de *blueprints* muy parecido a crear diagramas de flujo hace de este motor gráfico una herramienta muy versátil y potente.

## Unity 3D

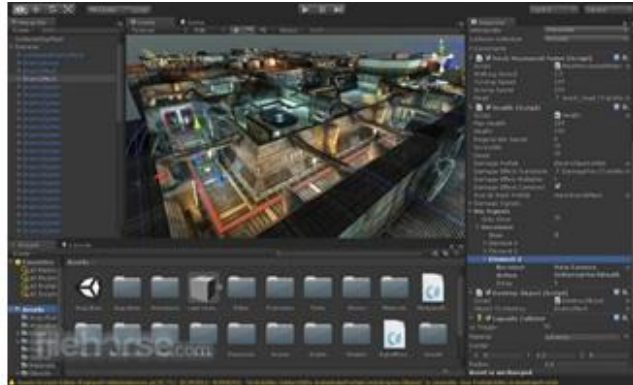


Figura 26. Logo de Unity y vista de su interfaz

Unity [34] (figura 26) fue lanzado en 2005 por *Unity Technologies*, su principal lenguaje de programación es C#. Este motor de juego permite a los usuarios usar una versión gratuita para desarrollar sus juegos, siempre y cuando se programe para *Windows*.

*Unity 3D* es una de las innovaciones más importantes creadas por la comunidad científica y de videojuegos ya que permite jugar los videojuegos creados con el motor en un navegador con el reproductor *Unity Web Player*, eliminando la necesidad de instalar el videojuego.

Hasta hace poco la versión gratuita tenía ciertas limitaciones que se podían eliminar si se compra la versión *Premium*, pero con *Unity 5* también han querido apuntarse un tanto y eliminar todas las restricciones. Esto quiere decir que la versión gratuita de este motor vendrá igual de completa que la versión de pago. La única diferencia es que por 75 dólares al mes podremos disfrutar de algunos servicios en la nube.

Muy sencillo e intuitivo, lo que permite crear grandes escenarios, mecánicas de juego, y animaciones en muy poco tiempo. Su curva de aprendizaje también es bastante pequeña, comparada, por ejemplo, con Unreal. Esto sumado a la gran cantidad de contenido, tutoriales, modelos y material que *Unity Technologies* pone a disposición de todo el mundo, conforma gran punto a favor para comenzar a usar este motor. Además, cuenta con una gran comunidad activa a la que puedes plantear tus dudas en cualquier momento.

## Cryengine

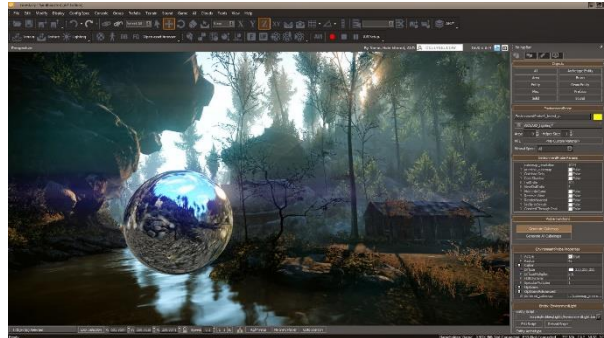


Figura 27. Logo de CryEngine y vista de su interfaz

Cryengine [35] (figura 27) es un motor desarrollado por la empresa *Crytek* en 2002. Su salto a la fama surgió a partir de los efectos de iluminación y el realismo conseguido por los juegos “*Far Cry*” y “*Crysis*”. Este motor está diseñado principalmente para juegos de *PC* y consolas como *PS4*, *Xbox One* o para juegos de realidad virtual.

Este motor permite crear juegos 3D estereoscópicos y grandes escenarios como junglas y bosques. Además, es capaz de crear efectos realmente sorprendentes tanto de fluidos en tiempo real como de sonidos.

Este motor es inicialmente gratuito, pero si la facturación llega a superar los 5.000 dólares, exigen el 5% de los ingresos, de esta forma buscan convencer a grandes compañías para que den una oportunidad a su motor y puedan descubrir todo su potencial.

Sus principales **puntos negativos** se centran en su **complejidad**, ya que, aunque cuenta con una comunidad relativamente amplia y tiene una gran cantidad de assets en el mercado, su curva de aprendizaje es bastante grande. Además está centrado en el **desarrollo de grandes juegos 3D** para pc y consolas.

### 4.1.3. Conclusión

Para el **desarrollo** de este trabajo se ha decidido hacer uso del motor de **Unity**, debido a que los proyectos que se realizan con este motor se pueden portar fácilmente a plataformas móviles y conseguir el rendimiento necesario para funcionar correctamente en estas plataformas. [36]

Por otro lado, la **curva de aprendizaje** necesaria para este motor es **menor** que en los otros que hemos valorado. Además, su gran comunidad nos proporciona una gran cantidad de tutoriales y *assets* que serán de gran ayuda durante el desarrollo de todo el proyecto.

Por último, Unity **cuenta** con el **SDK de Google VR** para poder **desarrollar** de forma sencilla para las gafas **Cardboard**, las cuales hemos elegido para el desarrollo de este trabajo.

## 4.2. Software de modelado

Los principales softwares de modelado que hemos valorado en este proyecto han sido dos, *Blender* y *3ds Max*. A continuación, hablaremos sobre estos dos y de sus principales puntos a favor y en contra. [37]

### 4.2.1. Blender

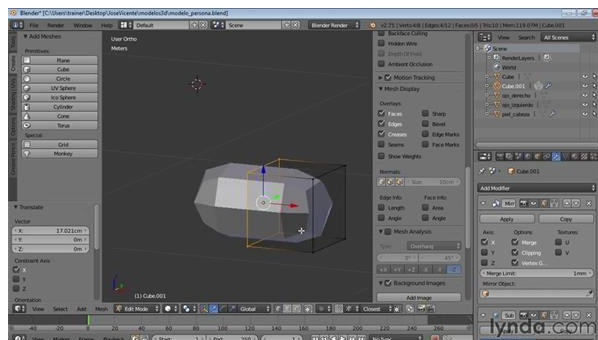


Figura 28. Logo de Blender y vista de su interfaz

*Blender* [38] (figura 28) es una suite multiplataforma de creación 3D gratuita y de código abierto. Este *software* permite modelar, animar, simular y renderizar todo tipo de modelados 3D e incluso realizar ediciones de video y diseño 2D.

Además, dispone de un sistema de escultura digital, con topografía dinámica que, aunque es menor que *Zbrush* y otros programas dedicados en exclusiva a esta tarea, consigue unos resultados realmente buenos.

Por otro lado, otro gran punto a favor de este programa es la posibilidad de crear texturas y materiales para tus modelos y poder visualizarlas directamente en el modelo 3D mientras los creas y editas. [39, 40]

### 4.2.2. Autodesk 3ds Max

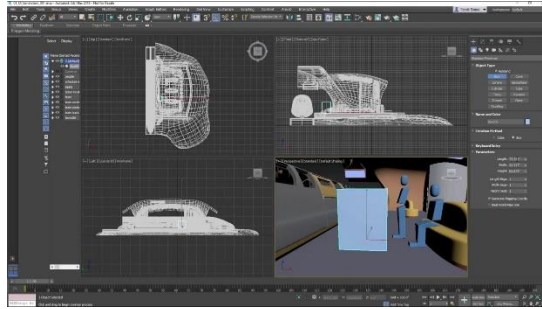


Figura 29. Logo de 3DS Max y vista de su interfaz

*3ds Max* [41] (figura 29) es un programa de creación de gráficos y animación 3D desarrollado por *Autodesk Media and Entertainment*. Aunque se trata de un programa de pago, ha sido valorado para la realización de este trabajo ya que ofrece licencias de uso gratuitas para estudiantes.

*3ds Max* incluye *shaders* que permiten crear sistema de partículas, simulaciones dinámicas, radiación y muchos otros efectos para dar un acabado muy profesional. Además, cuenta con un sistema para crear y editar animaciones muy robusto que permite incluso añadir efectos especiales a dichas animaciones.

Por otro lado, cuenta con soporte en una gran cantidad de idiomas para poder ayudar a todos sus usuarios, entre ellos español.

### 4.2.3. Conclusión

Aunque existen más programas de modelado que aportan muy buenos resultados como puede ser *Maya*, los dos anteriores son los únicos que se han valorado para la realización del proyecto. Esto se debe a que son los que hemos visto durante el grado y por tanto el tiempo necesario para conocer y utilizar el programa correctamente es menor.

Finalmente, he **decidido** utilizar ***Blender***, ya que es en el que tengo mayor experiencia y creo que para la complejidad de este proyecto aporta todo lo que necesito. Además, da la posibilidad de crear texturas y visualizar el resultado en todo momento sobre los modelos, lo que ahorra el tiempo de crear las texturas en otro programa y luego ver si coinciden correctamente con el modelo con el que se corresponden.

## 4.3. Herramientas de diseño y texturizado

### 4.3.1. Adobe Photoshop

La herramienta de texturizado que más se ha utilizado durante este proyecto ha sido *Adobe Photoshop* (figura 30), ya que nos permite crear nuevos mapas de texturas a partir de los mapas de UV exportados de *Blender* y editar aquellas texturas que ya hemos comenzado a pintar para lograr que tengan un acabado mucho más profesional.

Photoshop dispone de un gran número de pinceles que permiten añadir muchos detalles a las texturas y mejorar el resultado obtenido en *Blender* que, aunque dispone de pinceles para pintar la textura de los modelos, ofrece muchas menos opciones y es mucho menos preciso. Aun así, poder pintar las texturas en *Blender* ha sido un punto clave del desarrollo ya que permite obtener una primera aproximación de la textura que una vez mejorada con *Photoshop*, dará un gran resultado.

Además, con la ayuda de sus múltiples herramientas, *Photoshop*, permite crear mapas de oclusiones, normales y brillos a partir de las texturas que ya hemos obtenido, de forma que nuestros materiales serán mucho más realistas.



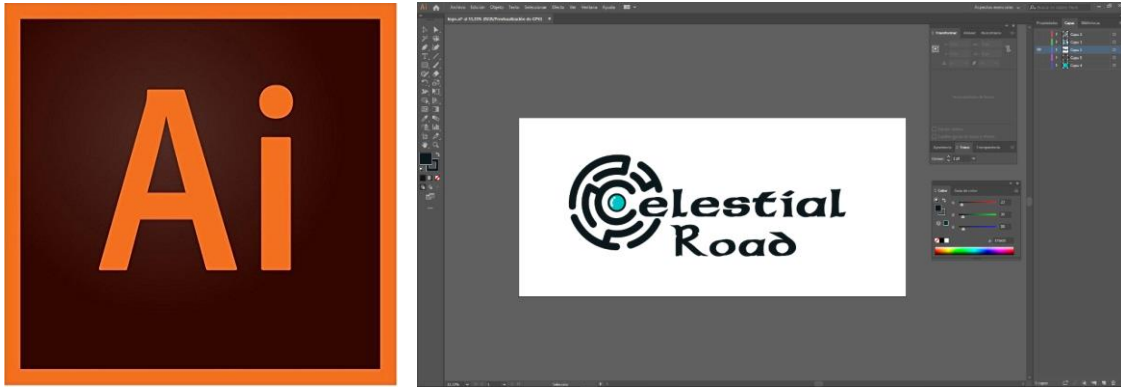
Figura 30. Logo Photoshop y vista de su interfaz

### 4.3.2. Adobe Illustrator

*Adobe Illustrator* (figura 31) ha sido utilizado para crear todos los elementos de las interfaces de juegos, como son los botones, adornos y lo más importante, el logo e icono de nuestro juego.

Este programa ha sido elegido ya que nos aporta la posibilidad de crear gráficos de manera vectorial, que puedan ser escalados a cualquier tamaño sin perder calidad.

Además, aunque su manejo es más complicado, muchas de sus herramientas y su interfaz en general son similar a la de Adobe Photoshop y por tanto su curva de aprendizaje disminuye ligeramente.



*Figura 31.* Logo Illustrator y vista de su interfaz

## 5. Objetivos

El objetivo principal de este proyecto es desarrollar un juego en realidad virtual para smartphone que entretenga al usuario, le haga pensar y le aporte nuevas formas de interactuar. Este juego se compondrá de distintos niveles que irán aumentando la dificultad según se avance por ellos. En cada nivel el usuario tendrá que intentar salir de un laberinto y solucionar los retos que se le planteen para lograr llegar a la salida.

Dentro de este trabajo también nos encontraremos otros objetivos específicos como son:

- Estudiar y comparar la propuesta con otros videojuegos para móvil que hagan uso de la realidad virtual y que actualmente estén en el mercado.
- Aprender a desarrollar juegos y aplicaciones en 3D en el entorno de Unity.
- Implementar interfaces en 2D para los menús necesarios en Unity teniendo en cuenta nociones básicas de UX.
- Plantear y diseñar los distintos niveles del juego, sus laberintos, para lograr que el usuario lo considere un reto y quiera seguir jugando.
- Aprender nuevas técnicas de modelado para poder crear todos los elementos presentes en los diferentes niveles del juego.
- Aprender a crear texturas y materiales necesarios para los modelos del juego que sigan la estética elegida para el juego.
- Integrar la realidad virtual el juego.
- Implementar un sistema de bloqueo y guardado de niveles para asegurarnos de que el usuario pueda avanzar de forma ordenada por los niveles.



## 6. Metodología

En este apartado se explica la metodología seguida en el proyecto junto con el porqué de su elección y las herramientas utilizadas para conseguir llevarla a cabo correctamente.

### 6.1. Metodología de desarrollo

Tras analizar distintas metodologías como las de cascada, desarrollo rápido y desarrollo ágil, decidí hacer uso de una **metodología ágil** como es la metodología **SCRUM**. En **SCRUM** se realizan entregas parciales de forma regular al cliente para lograr conseguir resultados de una forma rápida y continua, aumentando la satisfacción del cliente. Este punto para un proyecto con un tiempo limitado como es este, es muy importante ya que nos obliga a seguir unos tiempos muy ajustados para conseguir los objetivos propuestos en cada plazo.

Por otro lado, esta metodología permite **aceptar** cualquier **cambio** en los requisitos y funcionalidades del proyecto incluso en **etapas tardías** del desarrollo, evitando que un cambio resulte fatal en el desarrollo y mejorando las características de nuestro producto. [42]

Otro gran punto a favor del uso de esta metodología ha sido el uso de ella durante la carrera y, por tanto, aparte de conocer su funcionamiento y estar acostumbrada a su metodología de trabajo mediante iteraciones, ya he comprobado su funcionamiento y resultados. Por ello, sé que se trata una buena elección para proyectos con características similares a este y por ello creo que es la mejor elección para este proyecto.

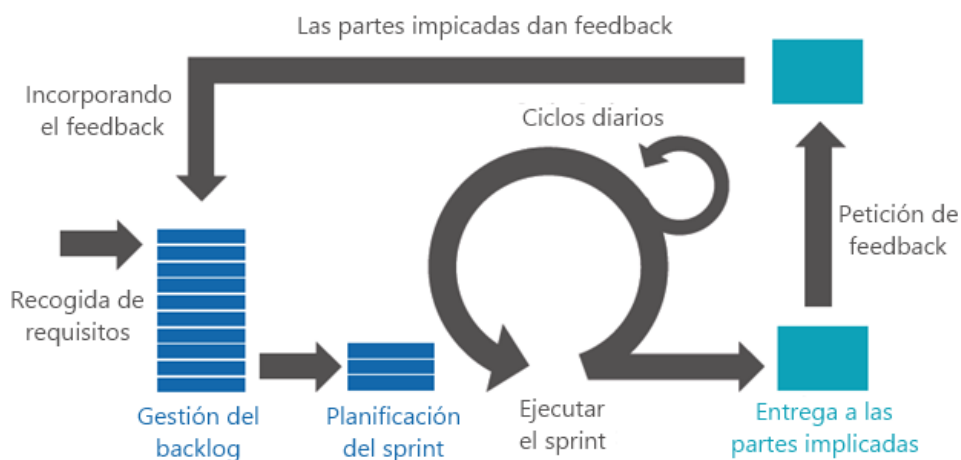


Figura 32. Esquema metodología SCRUM. Fuente [42]

Para seguir esta metodología se han seguido las fases presentes en la figura 32:

1. Se recogieron los requisitos del proyecto tanto de la aplicación en sí, como los puntos necesarios para completar la memoria y con todos ellos se creó una lista de tareas.
2. Se planifica el *sprint*, en mi caso planifiqué sprints semanales que, aunque aportaban un menor avance en cada sprint aseguraban el que el trabajo continuara en todas las semanas de este desarrollo evitando dejarlo por estar en verano.  
Para su planificación se decidían que tareas eran prioritarias y necesarias para no retrasar el proyecto y se realizaban en primer lugar. Si alguna tarea era demasiado extensa la subdividía en tareas más pequeñas y sencillas de finalizar.
3. Una vez decididas las tareas a realizar en el *sprint*, se realizaba cada día un ciclo diario de trabajo con alguna de las tareas del *sprint* y esta tarea no cambiaba hasta completarla.
4. Al finalizar el *sprint* revisaba el trabajo realizado y me aseguraba de guardar la nueva versión tanto de la aplicación como del documento para evitar pérdidas de información y asegurar un punto de referencia al que retroceder en caso de que apareciera algún error imprevisto.
5. Tras revisar el *sprint* y ver que es necesario cambiar y corregir se añaden las nuevas tareas y requisitos a la lista inicial de tareas y se vuelve a planificar el *sprint*, vuelta al punto 2, hasta que la lista de tareas este totalmente terminada.

## 6.2. Gestión del proyecto

Para llevar el control del proyecto y saber en todo momento las tareas que estaban pendientes, finalizadas, planificadas por el sprint o en las que se estaba trabajando se ha hecho uso de la herramienta web *online Trello [43]*.

Esta herramienta permite crear distintas listas de tareas y tareas en sí mismas, que se pueden mover fácilmente entre las listas simplemente arrastrándolas. De esta forma es muy sencillo y visual saber el estado de cada tarea en todo momento.

Además, *Trello* permite asignar fechas de finalización a cada tarea y te avisa en caso de que alguna tarea no finalizada supere dicha fecha para que te des cuenta del retraso y evitar que este se acumule.

Por otro lado, permite hacer uso de un código de colores para diferenciar las tareas. En mi tablero (Figura 33) use estos códigos de color para diferenciar el tipo de tarea, según fuera de implementación, diseño, modelado o redacción de la memoria.

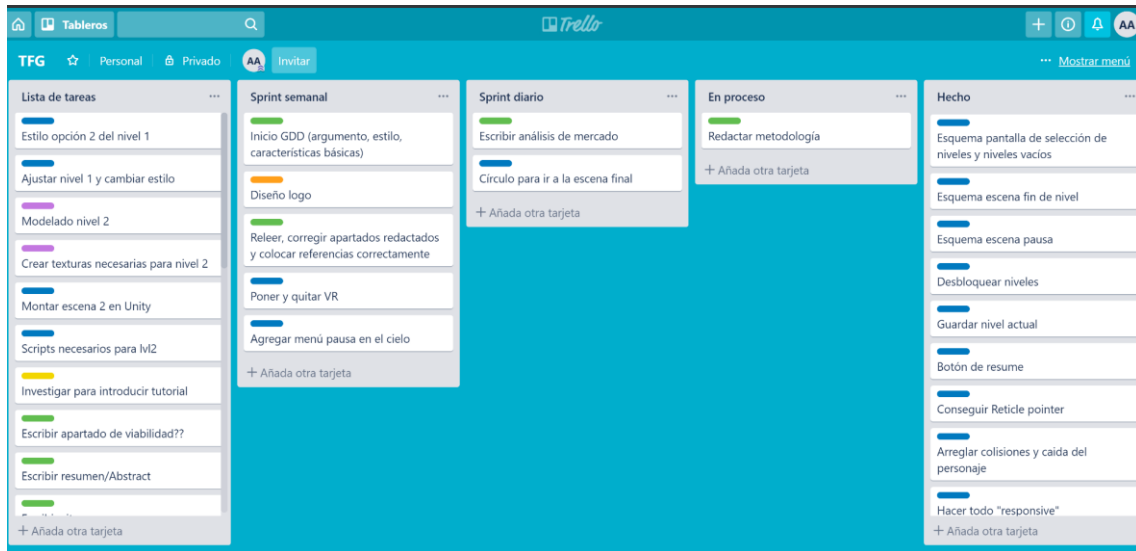


Figura 33. Tablero de Trello

### 6.3. Control de versiones

Para realizar un control de versiones y evitar la pérdida de datos del proyecto, se ha decidido hacer uso de dos herramientas. Por un lado, se ha hecho uso de **Google Drive [45]** para mantener las versiones de la memoria y otros documentos del proyecto como pueden ser las texturas e imágenes utilizadas entre otros.

Por otro lado, para asegurarnos de mantener correctamente el código generado para nuestra aplicación y poder revertir los cambios en caso de la aparición de algún error, hemos hecho uso de **GitHub [44]**.

Ambas herramientas han sido escogidas por su gran potencial y el conocimiento de su uso y funcionamiento ya que han sido utilizadas en proyectos anteriores.

## 7. Game Design Document (GDD)

En este apartado se describen todas las especificaciones que debe tener nuestro juego, así como todas las decisiones de diseño que se han tomado para su desarrollo.

De esta forma, se aporta una clara visión de cómo debe ser el juego una vez finalizado, tanto estéticamente como funcionalmente. Este apartado es de mucha ayuda no solo para conocer el videojuego una vez finalizado, sino también durante todo el desarrollo del proyecto ya que en él se registran todas las decisiones tomadas en el proyecto, evitando dudas e incongruencias durante el desarrollo. [46]

### 7.1. Concepto y características básicas

**Título del juego:** *Celestial Road*

**Plataforma:** Desarrollado para dispositivos móviles *Android* con giroscopio. Para jugarlo también es necesario disponer de un visor de realidad virtual como las *Cardboard*.

**Género:** Puzzles y laberintos en realidad virtual.

**Idioma:** Castellano

**Público objetivo:**

*Celestial Road* está disponible para todos los públicos, aunque se dirige especialmente a personas a las que les gusten los juegos RPG, los puzzles y laberintos y quieran experimentar esta experiencia en realidad virtual. Por esto último, no se recomienda el juego para menores de 6 años ya que, aunque el juego no contiene ningún elemento que les pueda asustar, no es conveniente el uso de realidad virtual a edades tan tempranas.

**Sinopsis:**

¿Alguna vez has pensado que te gustaría vivir una experiencia como la de los grandes aventureros? ¿Encontrarte en lugares misteriosos e inexplorados llenos de tesoros y acertijos que resolver? ¡Pues esta es tú oportunidad!

*Celestial Road* es un juego de laberintos que busca ofrecer nuevos retos y experiencias al usuario. El jugador tendrá que recorrer los distintos laberintos del juego para poder descubrir todos sus secretos y conseguir encontrar la salida.

Además, tendrá la oportunidad de vivir todo ello en primera persona en un entorno 3D y en *¡Realidad Virtual!*

### **Resumen del ciclo de juego:**

El ciclo o flujo del juego sigue siempre el mismo patrón (contando desde que el jugador ya ha entrado en el juego y se encuentra en el menú de selección de nivel):

1. El usuario debe elegir el nivel que quiere jugar, este ha de estar desbloqueado para poder acceder a él.
2. Se carga el nivel seleccionado y el jugador debe recorrerlo y encontrar el camino correcto hacia la salida.
3. Dependiendo de las acciones del jugador puede variar el final del nivel:
  - 3.1. Si el jugador encuentra la salida correctamente se cargará la escena de final de nivel donde se felicitará al usuario y se le indicará cuanto ha tardado en recorrer el laberinto y se le asignará una puntuación. Además, se le dará la oportunidad de repetir el nivel, avanzar al siguiente o volver al menú para seleccionar él mismo el nivel que quiere jugar, si es la primera vez que gana el nivel se desbloqueará del menú inicial el siguiente nivel.
  - 3.2. Si el jugador pone en pausa el juego, podrá decidir si continuar jugando, reiniciar el nivel o salir. En este último caso, se le dirigirá al menú principal para seleccionar otro nivel y se considerará que no ha superado el nivel en el que estaba.

## **7.2. Jugabilidad y mecánicas**

En este apartado se describen las acciones que puede realizar el usuario para interactuar dentro del juego, así como los controles necesarios para realizarlas.

### **Cámara**

El juego se desarrolla en un entorno 3D y en realidad virtual, por lo que la cámara está colocada para obtener una vista en primera persona, simulando la visión desde los ojos del jugador. Para poder jugar correctamente es necesario el uso de un visor de realidad virtual como las *Cardboard*.

## **Acciones y controles**

**Mirar o girar la cámara:** para girar la cámara y mirar hacia otro lado, el usuario solo tendrá que mover la cabeza hacia un lado u otro, suponiendo que este jugando con el móvil situado en el visor de realidad virtual y este correctamente colocado sobre los ojos. En cualquier otro caso con mover el *smartphone* ya cambiará la vista de la cámara.

**Caminar:** para poder avanzar por el laberinto y conseguir llegar a la salida, el usuario tendrá que caminar por el laberinto, para ello solo tendrá que inclinar ligeramente la cabeza y mirar hacia el suelo en la dirección en la que quiera avanzar y el personaje se moverá automáticamente.

Igualmente, para **parar** o detener al personaje solo tendrá que mirar hacia arriba, simplemente sobrepasando la línea de horizonte, y el personaje dejará de caminar, permitiendo al jugador observar el entorno y decidir su nuevo rumbo.

**Abrir menú pausa:** para acceder al menú de pausa el jugador tendrá que mirar al cielo donde se encuentra el botón para abrir el menú de pausa, situar su vista sobre él y hacer clic con los imanes disponibles en las *Cardboard* o simplemente quedarse mirando al botón hasta que el círculo de carga se complete (Figura 36).

**Interacción con botones u objetos dentro de un nivel:** todos los botones y objetos con los que se puede interactuar se diferencian de resto ya que, al fijar la vista sobre ellos, el punto de mira que se encuentra en el centro de cámara reaccionará y cambiará su tamaño (Figura 34 y 35) y comenzará a rellenarse un círculo de carga (Figura 36), para seleccionar o interactuar con cualquiera de ellos el usuario solo tendrá que esperar a que se complete el círculo o hacer clic con los imanes de las *Cardboard*.

De la misma forma, si se quiere seleccionar algún botón u objeto en realidad virtual directamente desde el *smartphone*, en caso de estar jugando sin un visor, bastará con fijar la mira sobre dicho objeto hasta que se complete el círculo de carga o tocar la pantalla del *smartphone* para hacer el clic.

**Interacción con botones fuera del nivel:** para interactuar con cualquier menú que no esté en realidad virtual valdrá con tocar el botón deseado en la pantalla del *smartphone*.

**Recoger objetos:** el usuario solo tendrá que pasar sobre los objetos o tocarlos para recogerlos.



Figura 34. Punto de mira reticular cerrado



Figura 35. Punto de mira reticular abierto

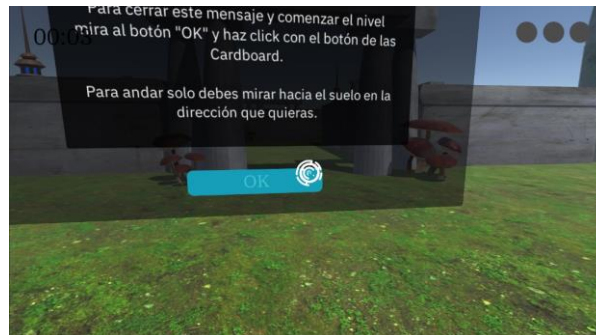


Figura 36. Círculo de carga casi completo

## Puntuación y avance en el juego

Para poder desbloquear los niveles y avanzar por el mundo de Celestial Road el jugador tendrá que ir superando todos los laberintos en orden.

Cuando el jugador entra por primera vez al juego solo puede entrar en el nivel 1, debido a que todos los demás se encuentran bloqueados. Cuando el jugador logre superar el laberinto y llegar a la salida (Figura 37), se desbloqueará el siguiente nivel, permitiéndole acceder a él desde la pantalla de fin de nivel (Figura 38) o desde el menú de selección de nivel (Figura 48). De esta forma, podrá ir avanzando por los laberintos de Celestial Road.

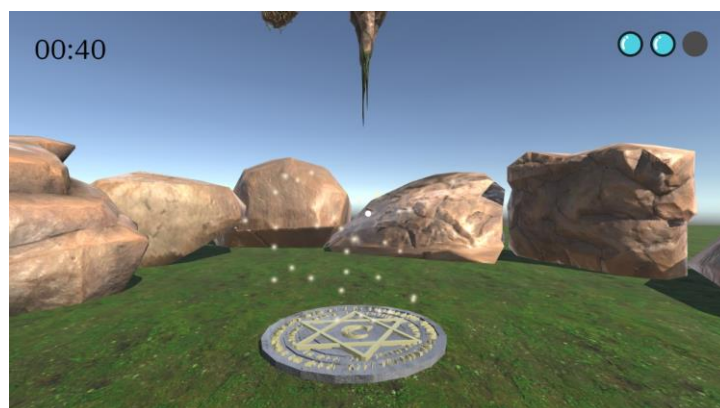


Figura 37. Círculo para salir del laberinto y superar el nivel

Por otro lado, para darle mayor emoción se ha incluido un sistema de puntuación por estrellas, mediante el cual se valorará el desempeño del jugador durante en nivel, dotándole de una puntuación de entre 1 y 3 estrellas (Figura 38), que quedará presente en el menú de selección de nivel.

Tras cada nivel se valorará el tiempo invertido desde que el usuario entra en el laberinto hasta que consigue llegar al final de este; y si ha recogido todos los objetos escondidos en el laberinto. Las puntuaciones son las siguientes:

- 3 estrellas: el jugador ha conseguido superar el nivel en menos tiempo del límite estimado<sup>1</sup> y ha recogido todos los objetos del laberinto.
- 2 estrellas:
  - El jugador ha completado el laberinto en menos tiempo del límite estimado, pero no ha recogido todos los objetos.
  - El jugador ha recogido todos los objetos del laberinto, pero ha superado el límite de tiempo estimado.
- 1 estrella: El jugador ha conseguido terminar el laberinto, pero ha superado el tiempo límite estimado y no ha recogido todos los objetos.

En caso de que el jugador salga del nivel antes de conseguir llegar al final, se le dotará de 0 estrellas y se considerará que no ha terminado el nivel, por lo que el siguiente nivel no se desbloqueará.



Figura 38. Interfaz de fin de nivel con distintas puntuaciones

<sup>1</sup> El tiempo límite estimado varía en cada nivel. En el apartado 7.5. Niveles se concretará cual es el correspondiente para cada uno.



## Guardar y cargar partida

El juego se guardará automáticamente cada vez que el jugador complete un nivel, evitando que pierda cualquier avance si se olvida de guardar la partida o si surge algún problema inesperado antes de que pueda hacerlo.

De la misma forma, cuando el usuario entre en la aplicación se cargará de forma automática el archivo con los datos guardados y así podrá continuar desde el mismo nivel en el que lo dejó la última vez.

### 7.3. Estilo y ambientación

Los niveles de Celestial Road cuentan con un estilo semi-realista, los laberintos están compuestos de muros, columnas de piedra y faroles, buscando aparentar ser un laberinto abandonado de una civilización antigua. Aunque no se puede olvidar el motivo que da nombre al juego, todos los laberintos de Celestial Road se encuentran en islas flotantes, por lo cual el usuario no puede salir del laberinto de forma normal y debe salir mediante un círculo mágico que se encuentra al fondo del laberinto y que podemos ver en la figura 37. Por este motivo el jugador puede ver sobre todos los laberintos más islas flotantes, ya que el mismo se encuentra en una.

Todos los laberintos se encuentran al aire libre para evitar mareos, sensaciones de agobio y claustrofobia para el usuario. Cada uno está ambientado en una hora distinta del día, siendo el primero por la mañana con mucha luz y el último al anochecer, iluminado por luces y cristales, como se puede ver en la figura 39. Además, los objetos que forman el laberinto también van cambiando poco a poco simulando un avance en la civilización que los construyó.



*Figura 39. Comparación del primer y último nivel respectivamente*

Por otro lado, los menús siguen un estilo muy sencillo e intuitivo. Todos los botones son cuadrados, pero con los bordes redondeados, con un único color, el color principal del juego, el *Celestial Blue* #4dd0e1.

### 7.3.1. Guía de estilo

En este apartado se detallan todas las características relativas al diseño del logo y los elementos presentes en las interfaces del juego.

#### 7.3.1.1. Misión

**Celestial Road** busca entretener y aportar nuevas experiencias al usuario dándole la oportunidad de vivir una aventura única y fantástica, en la que se adentrará en un laberinto de una civilización antigua y deberá conseguir escapar de él en el menor tiempo posible, habiendo recogido todas las joyas ocultas.

Para lograr este objetivo y poder llegar a los usuarios, se ha tenido en cuenta el gran auge actual de las tecnologías, como son los avances en los *smartphones*, cada día más potentes y con más funcionalidades; y las novedades en el campo de la realidad virtual y aumentada, que cada vez se encuentran más presentes en nuestro día a día y han demostrado captar la atención de muchísimos usuarios.

Esto, unido a la gran disponibilidad de dichas tecnologías en la actualidad, ayudará a que un mayor número de usuarios disfrute de esta nueva aventura y valore **Celestial Road** para poder seguir mejorando y creciendo.

#### 7.3.1.2. Filosofía de diseño

Como ya se ha comentado anteriormente, Celestial Road busca introducir al usuario en un mundo semi-realista para intentar hacer creer al usuario que realmente se encuentra dentro de un laberinto y lograr que la sensación de inmersión sea lo más correcta posible.

Por otro lado, el resto de las interfaces presentes en el juego buscan ser muy sencillas e intuitivas, intentando que el usuario sea consciente de todas las opciones e interacciones

disponibles en todo momento. Para ello se apuesta por interfaces limpias y sencillas con pequeñas animaciones que hagan la experiencia del usuario más atractiva y entretenida.

### 7.3.1.3. Logo y colores

#### Logo

El logo de Celestial Road, figura 40, está compuesto de su símbolo (Figura 41), que representa un laberinto redondo y una C, la C de Celestial Road en el logo completo, junto con una joya o gema en el centro, que representa las gemas que el usuario encontrará dentro del laberinto y con las que conseguirá la puntuación de 3 estrellas en cada nivel.

Por otro lado, se encuentran las letras que forman el nombre del juego, la tipografía usada busca un estilo antiguo que dé la sensación de pertenecer a alguna antigua civilización celta, intentando seguir el estilo y la ambientación decidida para el juego.



Figura 40. Logo completo sobre fondo claro



Figura 41. Símbolo o icono del juego

También se ha creado la versión en blanco y negro del logo, así como el logo sobre fondos oscuros, por si en algún momento fuera necesario hacer uso de alguno de ellos, figuras 42 y 43.



Figura 42. Logo completo en blanco y negro



Figura 43. Logo completo sobre fondo oscuro

## Colores

Los colores utilizados para el logo serán también los colores principales de nuestro juego, y por tanto son los usados para el diseño de los menús e interfaces del juego, como se puede ver en el apartado 7.5 de este documento.



## Tipografía

La tipografía utilizada para el logo es **Celtic MD**, que como ya hemos comentado busca un estilo antiguo que dé la sensación de pertenecer a alguna antigua civilización celta, intentando seguir con la ambientación del juego.

Por otro lado, la tipografía utilizada en el resto del juego son de la familia **IBM Plex** en dos de sus variantes, **Sans** y **Serif**.

La primera de ellas, *IBM Plex Sans* ha sido utilizada en todos los carteles y textos del juego. La segunda sin embargo solo ha sido utilizada en los números de los niveles del menú principal y para el título de pantallas como la de fin de nivel.

### 7.3.2. Música

Toda la música utilizada en el juego ha sido descargada de la página *Jamendo* [47], asegurando que disponían de licencia para su reutilización. Todas las canciones utilizadas poseen una licencia *Creative Commons CC BY-NC*, que permiten su reutilización de forma no comercial siempre que se reconozca la autoría de las obras, solo la canción utilizada en el menú principal, al inicio del juego tiene una licencia más restrictiva, *CC BY-NC-ND* que, además de lo anterior, prohíbe la modificación de la obra.

En este caso ninguna de las obras ha sido modificada, el juego no va a ser monetizado, por lo que no se va a conseguir una compensación monetaria de él y todas las obras han sido referenciadas tanto en esta memoria como en los créditos del juego.

Por otro lado, dejando el tema legal, la música elegida es toda música instrumental que sirvan de fondo y acompañamiento para el usuario. Se ha intentado que todas las canciones de los niveles tengan un carácter parecido, que incite al usuario a explorar el laberinto de una forma animada y rápida para evitar el aburrimiento del jugador.

Solo la obra elegida para el sexto nivel tiene un matiz más enigmático ya que, este nivel es distinto a los demás, no solo es el más complicado, la estética en general cambia, ofreciendo un escenario con la luz tenue de un atardecer y una gran cantidad de elementos brillantes durante el recorrido del laberinto, por lo que incluir una música con un carácter más de misterioso se ha considerado la mejor opción para este entorno.

Todas las canciones que podemos encontrar en los niveles, en la pantalla de fin de nivel y en los créditos del juego han sido descargadas de *Jamendo* [47] y pertenecen a *FiluAndDina* [48] con licencia *CC BY-NC 4.0* [49].

La música elegida para presentar el juego al usuario, así como para el menú inicial ha sido *Celtic Knight de Ebunny* [50], también descargada de *Jamendo* [47] y con una licencia *CC BY-NC-ND 4.0* [51]. Esta obra aporta un carácter enérgico y aventurero al juego, además de mantener la temática celta que también se intenta mostrar con el logo y su tipografía.

## 7.4. Estados del juego

En este apartado se puede observar el esquema de flujo de estados (Figura 44), en él se indica como se relaciona cada estado del juego con el resto, desde cual se puede llegar y hacia cual se puede avanzar.

En el siguiente apartado se pueden ver las interfaces relacionadas con cada estado, son aquellas que tienen el mismo nombre que el estado con el que se corresponden. Solo el estado nivel está compuesto de varias interfaces, el H.U.D. y el menú de opciones.

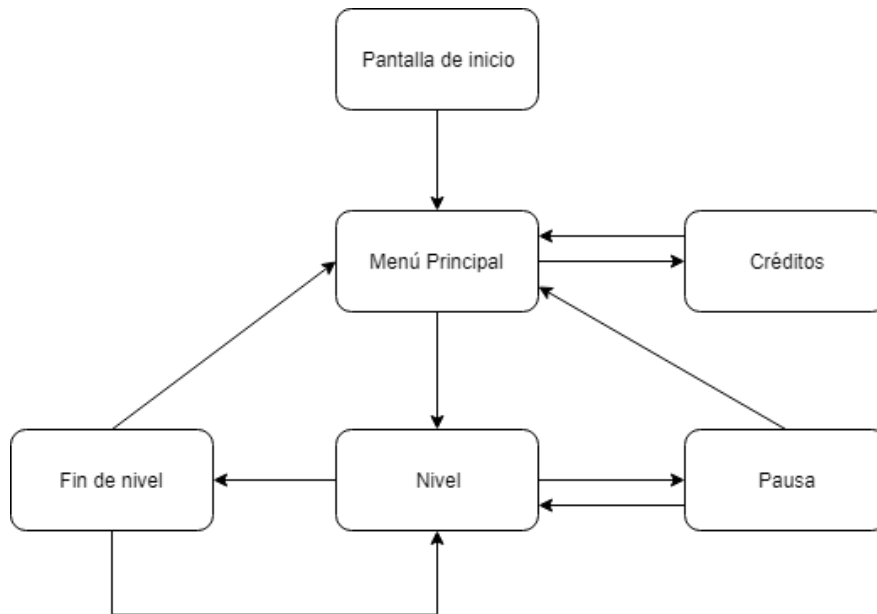


Figura 44. Esquema de flujo de estados

## 7.5. Interfaces

En este apartado se especifican las interfaces de Celestial Road junto con su nombre y descripción. Todas ellas siguen el estilo artístico descrito en el apartado 7.3. Estilo y ambientación de este documento.

### Pantalla de inicio

Es la primera pantalla que se encuentra el usuario al abrir el juego, muestra el logotipo del juego. El usuario solo debe hacer clic en ella para avanzar a la siguiente pantalla, el menú principal.



Figura 45. Boceto pantalla de inicio



Figura 46. Interfaz pantalla de inicio

## Menú principal o de selección de nivel

En esta pantalla se visualizan los niveles del juego, diferenciando los desbloqueados y ya superados que tendrán su valoración con estrellas; el último nivel desbloqueado, pero no superado al que el usuario podrá entrar, pero todavía no tiene ninguna estrella; y los bloqueados de los cuales no pueden verse los números ya que tienen un candado encima y por lo que, el usuario no puede acceder a ellos.

De esta forma, el usuario podrá elegir el nivel que quiere jugar, siempre que esté desbloqueado, y se le redirigirá al laberinto del nivel seleccionado.

Desde esta pantalla también se puede silenciar el juego, haciendo clic en el icono situado en la esquina inferior derecha, o acceder a la pantalla de créditos, cuando se da al icono de información de la esquina izquierda.

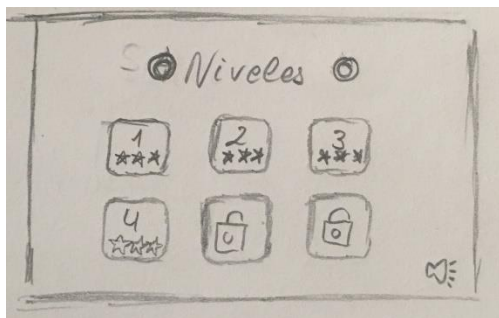


Figura 47. Boceto menú principal



Figura 48. Interfaz menú principal

## Nivel y H.U.D. (Heads-up Display)

En todos los niveles podremos observar el mismo H.U.D., compuesto de dos partes diferentes, a la derecha podemos ver los objetos que vamos encontrando durante el recorrido y a la izquierda se puede visualizar el cronómetro que indica cuanto tiempo llevamos en el laberinto. El usuario no podrá interactuar con el H.U.D. de ninguna manera.

Además, el H.U.D. solo será visible cuando el jugador juegue con la realidad virtual desactivada, en otro caso, la información de esta interfaz podrá visualizarse en el menú de opciones.

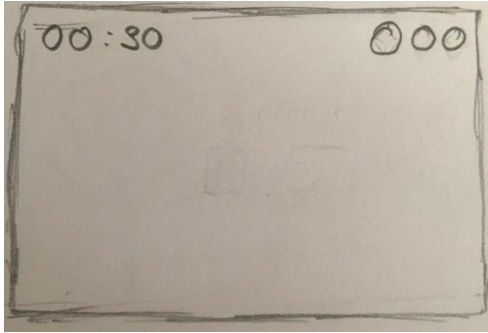


Figura 49. Boceto H.U.D.



Figura 50. Interfaz del H.U.D. de los niveles sin VR

### Menú de opciones (Estado nivel)

Otro menú que nos encontramos dentro de niveles se encuentra sobre el jugador en todo momento, por lo que no se ve a simple vista al entrar. Si el usuario mira hacia el cielo encontrará el menú de opciones, compuesto de dos botones desde los que podrá poner en pausa el juego o desactiva la realidad virtual para jugar simplemente desde su smartphone, sin visor. Además, si se está jugando con el modo de Realidad Virtual encendido, se encontrará la información del tiempo que ha transcurrido desde que entró al nivel y el número de objetos que ha recogido hasta el momento, si no tiene la Realidad Virtual encendida esta información la encontrará en el H.U.D. del juego.

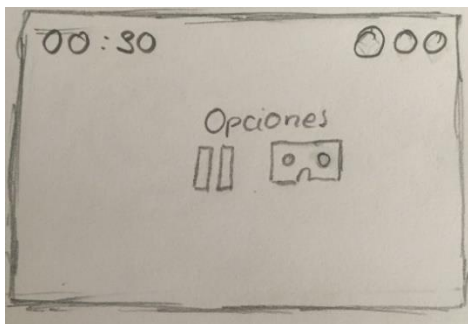


Figura 51. Boceto menú de opciones

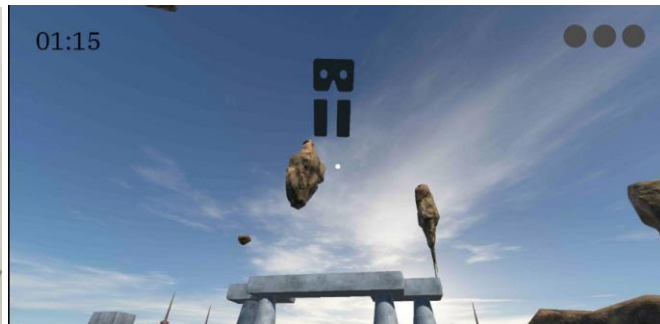


Figura 52. Interfaz menú de opciones en el modo no VR

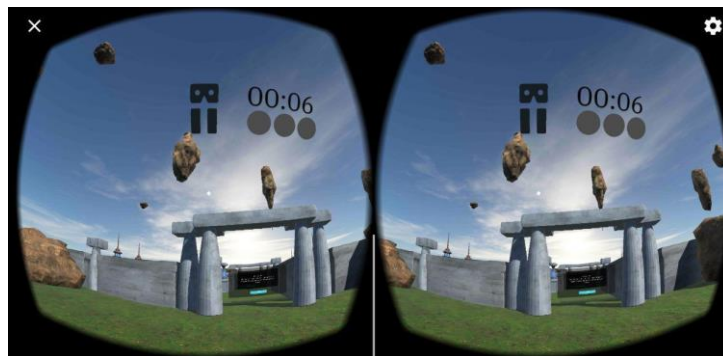


Figura 53. Interfaz menú de opciones en el modo VR



## Menú de pausa

Si el jugador pone en pausa el juego desde el menú anterior accederá a esta interfaz que se encuentra compuesta por tres botones. El primero de ellos le sacará de nivel y le llevará al menú principal y se considerará que el nivel no ha sido superado, el segundo reiniciará el nivel y el tercero le devolverá al laberinto para continuar desde el mismo punto en el que estaba. Además, le dará la opción de silenciar o activar el sonido del juego.

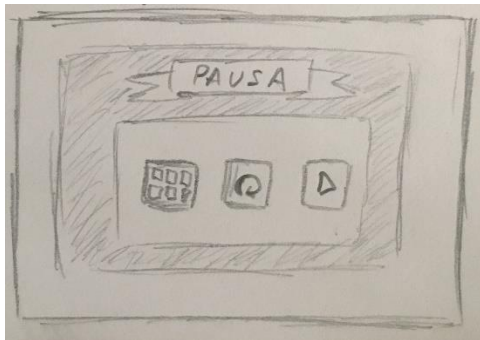


Figura 54. Boceto menú de pausa



Figura 55. Interfaz menú de pausa

## Fin de nivel

Esta interfaz aparece cuando el usuario ha conseguido llegar al final del laberinto y, por tanto, el nivel se considera superado. En esta pantalla se felicitará al usuario, se le indicará cuanto ha tardado en recorrer el laberinto y se le dará una valoración de entre 1 y 3 estrellas.

En la parte inferior se encuentran tres botones, el primero de ellos para ir al menú principal, el segundo para repetir el nivel actual y el tercero para avanzar al siguiente nivel.

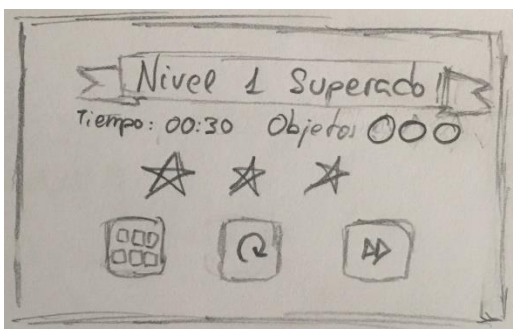


Figura 56. Boceto fin de nivel



Figura 57. Interfaz fin de nivel

## Pantalla de carga

Esta interfaz está constituida únicamente del icono de la aplicación en el centro de la pantalla girando. Aparece cuando el usuario selecciona un nivel, de esta forma el usuario recibe un feedback inmediato de su acción mientras se carga el nivel seleccionado.

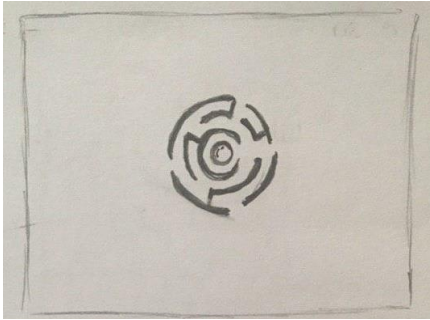


Figura 58. Boceto pantalla de carga



Figura 59. Interfaz pantalla de carga

## Créditos

En esta pantalla el jugador podrá ver los créditos del juego, en ellos se menciona el objetivo del juego, nombre del autor, título de las canciones utilizadas y otros datos importantes del juego. La pantalla irá haciendo scroll hacia arriba para mostrar todos los datos.



Figura 60. Interfaz escena créditos

## 7.6. Niveles

En este apartado se definen las características básicas de cada nivel, se indicará cual es el mapa del nivel, el lugar en el que se encuentran colocadas las gemas de cada nivel y el tiempo límite disponible para superar el nivel y conseguir la máxima puntuación.

### Nivel 1

#### Descripción:

Se trata del primer nivel al que accede el usuario, se trata de un nivel tutorial ya que en él se encuentran diversos paneles que explican el funcionamiento y las interacciones del juego; además se trata de un nivel muy corto y sencillo.

En este nivel se encuentran distintos carteles explicativos donde se le especifica en qué consiste el juego, cómo debe moverse el usuario y cómo debe interactuar con los distintos menús y objetos del laberinto.

Al tratarse del primer nivel, está iluminado para simular que es primera hora de la mañana, conforme pasen los niveles también irá cambiando la hora del día que intentan simular.

#### Mapa:

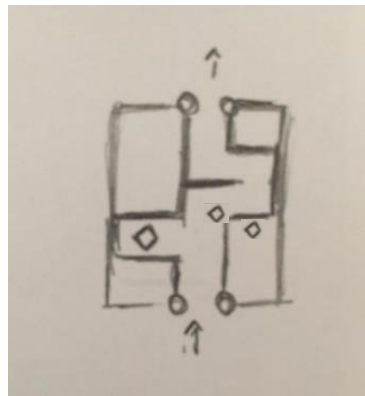


Figura 61. Boceto laberinto nivel 1



Figura 62. Vista aérea nivel 1

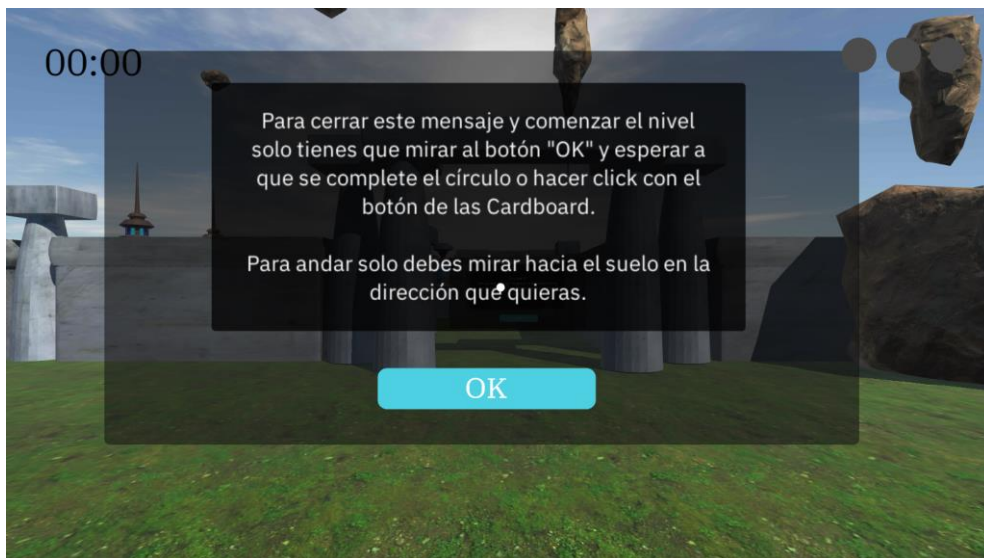
#### Tiempo límite:

El tiempo límite establecido para superar el nivel es de **50 segundos** porque, aunque es muy sencillo, se ha considerado conveniente dar más tiempo al usuario para que observe el entorno

en sí. El tiempo para leer los paneles no se tiene en cuenta ya que durante este tiempo se pausa el cronómetro.

Además, para el tiempo límite de todos los niveles se ha tenido en cuenta que se recogen todas las gemas del laberinto.

**Vista inicial:**



*Figura 63. Vista inicial del nivel 1 con el panel de inicio*



*Figura 64. Vista inicial del nivel 1 sin panel*

## Nivel 2

### Descripción:

El segundo nivel es un poco más complicado ya que tiene varios posibles caminos para el usuario, aunque todos los caminos erróneos son muy cortos para evitar perder al usuario. Además, las gemas colocadas en el nivel para conseguir las 3 estrellas de puntuación, están localizadas de forma que sean muy sencillas de encontrar, e incluso ayudan al usuario a ver el camino correcto, las gemas están colocadas de igual forma que los círculos del boceto, figura 65.

Este nivel está ambientado por la mañana, a una hora aproximada de las once de la mañana como se puede ver por el color claro de la luz y las sombras oblicuas.

Además, incluye algunos objetos nuevos como son varios tipos de setas decorativas con las que el usuario no puede chocarse y nuevas columnas en las esquinas más altas y delgadas que el resto de las columnas presentes en el nivel.

### Mapa:

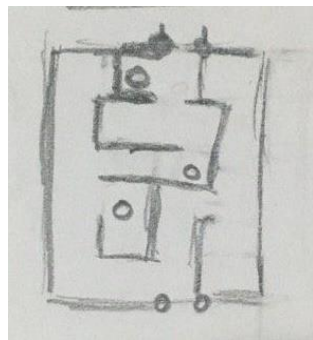


Figura 65. Boceto laberinto nivel 2

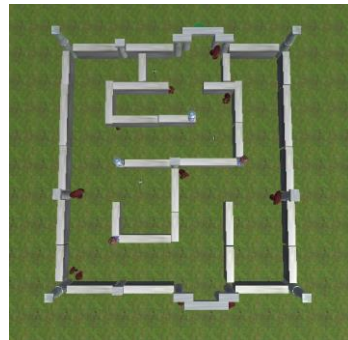


Figura 66. Vista aérea nivel 2

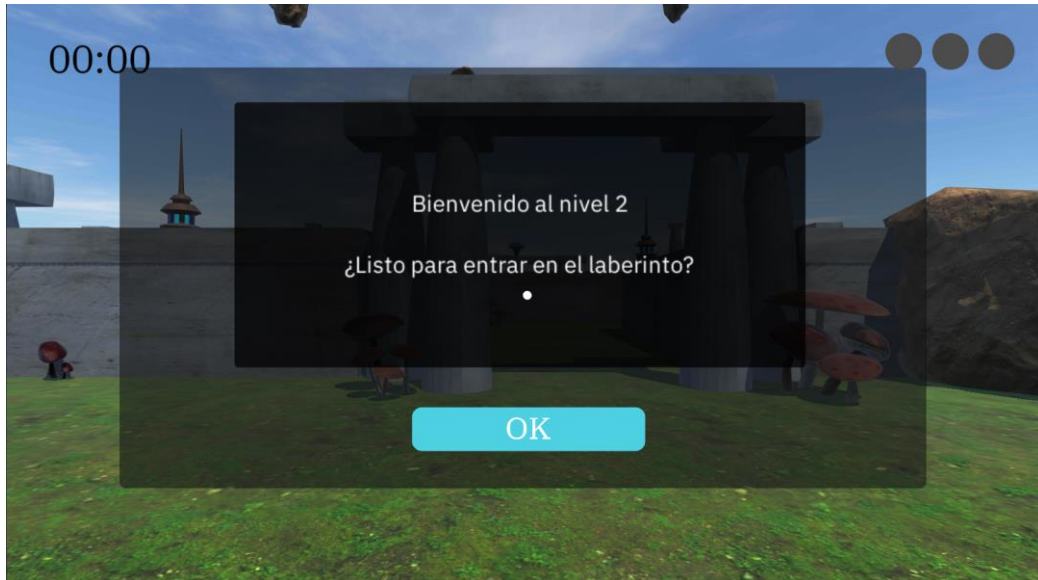
### Tiempo límite:

El tiempo límite estimado para completar el nivel y recoger las 3 gemas del nivel es de **70 segundo**, aunque el nivel se puede completar en menos tiempo, se ha considerado que al ser el segundo nivel era importante dejar al usuario más tiempo para que se vaya habituando al entorno y los controles.

### Vista inicial:

Como se puede ver en la figura 67, la vista inicial tiene un panel en el que se le avisa al usuario del nivel en el que se encuentra y se le pregunta si está listo para comenzar. Este panel se ha implementado en todos los niveles para dar al usuario tiempo suficiente para colocarse el visor

correctamente antes de comenzar el nivel y por tanto hasta que el usuario no accede a comenzar el cronómetro no cuenta el tiempo transcurrido.



*Figura 67. Vista inicial del nivel 2. Panel de inicio*



*Figura 68. Vista inicial del nivel 2 sin panel de inicio*

### **Nivel 3**

#### **Descripción:**

Este nivel es un poco más amplio que el anterior, aunque sigue siendo bastante sencillo ya que una vez llegado a la mitad casi se puede ver la salida. Aun así, las gemas del nivel se encuentran

más escondidas que en el caso anterior, lo que dificulta un poco conseguir las 3 estrellas. Este nivel está ambientado a mediodía y con una luz más amarillenta que los anteriores.

**Mapa:**

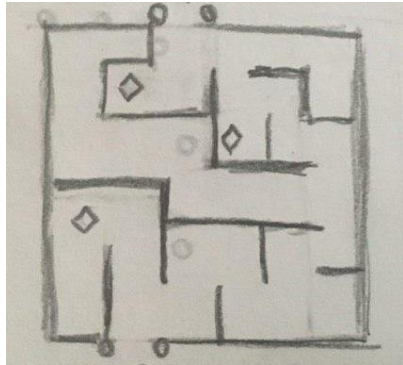


Figura 69. Boceto laberinto nivel 3

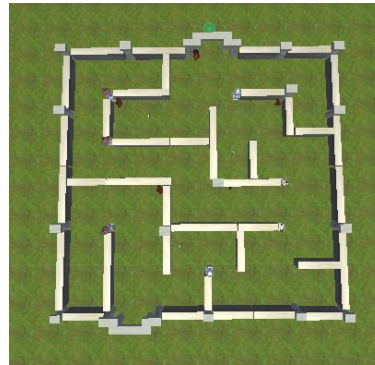


Figura 70. Vista aérea nivel 3

**Tiempo límite:**

El tiempo límite estimado es de **70 segundos**, tiempo suficiente para recoger los objetos y observar un poco el laberinto.

**Vista inicial:**



Figura 71. Vista inicial del nivel 3 sin panel de inicio

## Nivel 4

### Descripción:

El nivel 4 está ambientado justo pasado el mediodía, la luz es un poco más amarilla y las sombras empiezan a alargarse en sentido contrario al de los niveles iniciales.

Además, en este nivel se añaden nuevos muros que muestran un avance en la civilización que creó los laberintos, estos muros son de una piedra más clara y refinada, además ajustan mejor entre ellos evitando que se pueda ver entre un muro y el siguiente.

### Mapa:

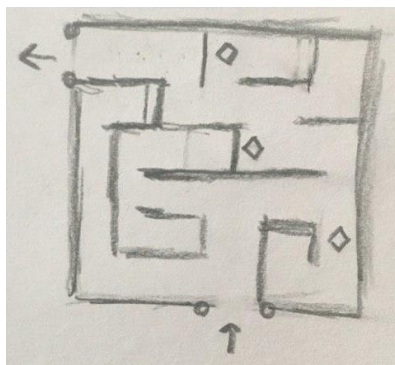


Figura 72. Boceto laberinto nivel 4

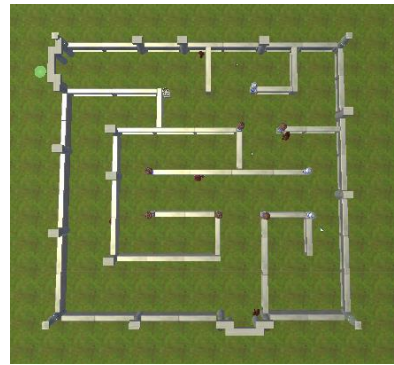


Figura 73. Vista aérea nivel 4

### Tiempo límite:

Como es un nivel algo más complicado el tiempo límite establecido es mayor, **105 segundos**.

### Vista inicial:



Figura 74. Vista inicial del nivel 4 sin panel de inicio



## Nivel 5

### Descripción:

El quinto nivel ya es bastante complicado, permite que el usuario se equivoque varias veces de camino y los caminos erróneos son relativamente largos. Además, las gemas se encuentran más escondidas que en los casos anteriores, haciendo que el usuario de más vueltas para poder encontrarlas.

Este nivel está basado en altas horas de la tarde, más o menos sería sobre las 7 de la tarde, su luz es ya anaranjada y las sombras son bastante más alargadas.

Además, incluye algunos objetos nuevos, como un tipo de muros y unos faroles, ya que la civilización sigue avanzando.

### Mapa:

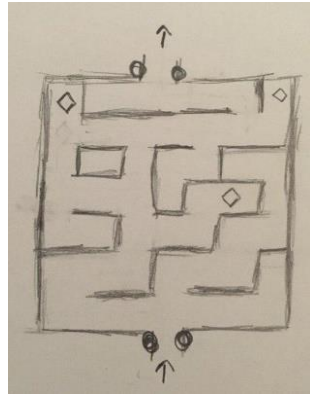


Figura 75. Boceto laberinto nivel 5

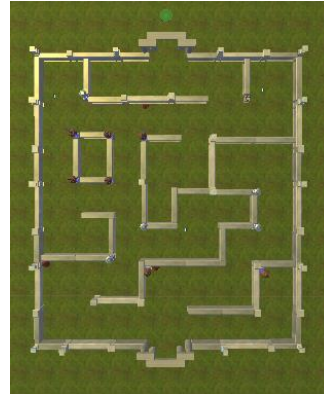
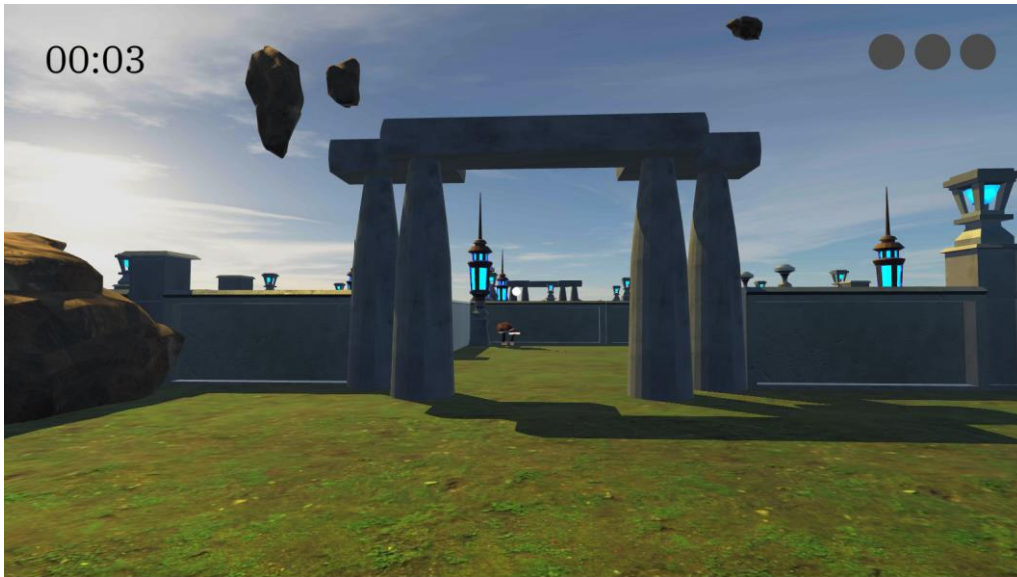


Figura 76. Vista aérea nivel 5

### Tiempo límite:

Aunque el nivel es bastante más complicado que los anteriores por lo menos para lograr la máxima puntuación, el tiempo límite estimado es de **100 segundos**, un poco ajustado pero suficiente para superarlo, de esta forma va subiendo la dificultad.

### Vista inicial:



*Figura 77. Vista inicial del nivel 5 sin panel de inicio*

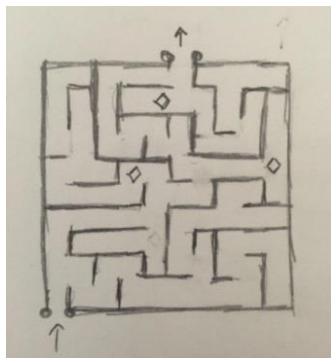
## **Nivel 6**

### **Descripción:**

Se trata del último nivel del juego, plantea un auténtico reto para el jugador. Se trata de un nivel mucho mayor que todos los anteriores, las opciones para que el jugador se equivoque son muchas más y más largas.

Este nivel está ambientado al atardecer, muchos de los elementos presentes en él brillan y desprender luz propia. Para ello, se han implementado nuevos objetos como adornos brillante sen los muros o cristales, además de haber cambiado los materiales de objetos como las setas para que brillen y destaquen más.

### **Mapa:**



*Figura 78. Boceto laberinto nivel 6*

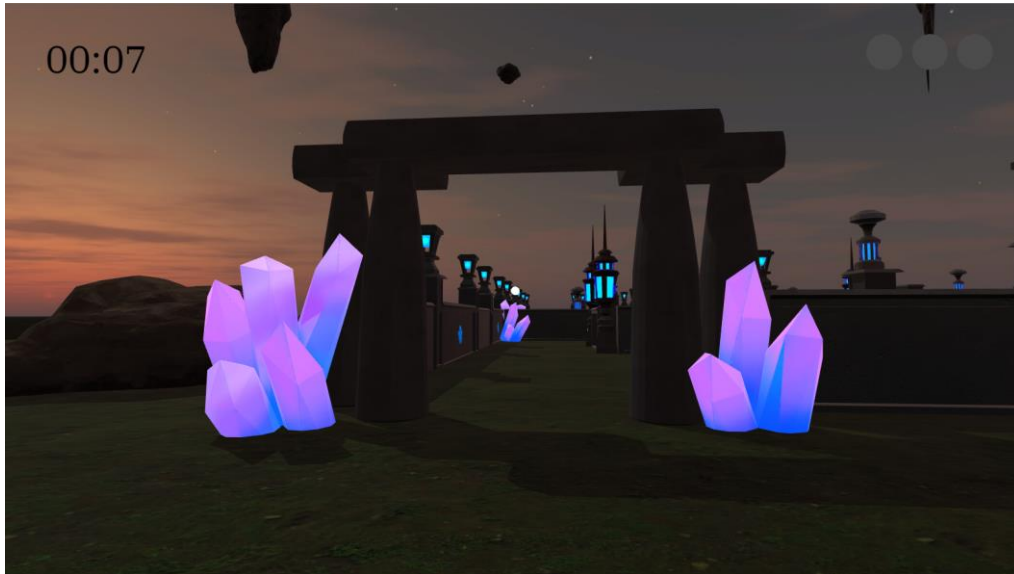


*Figura 79. Vista aérea del nivel6*

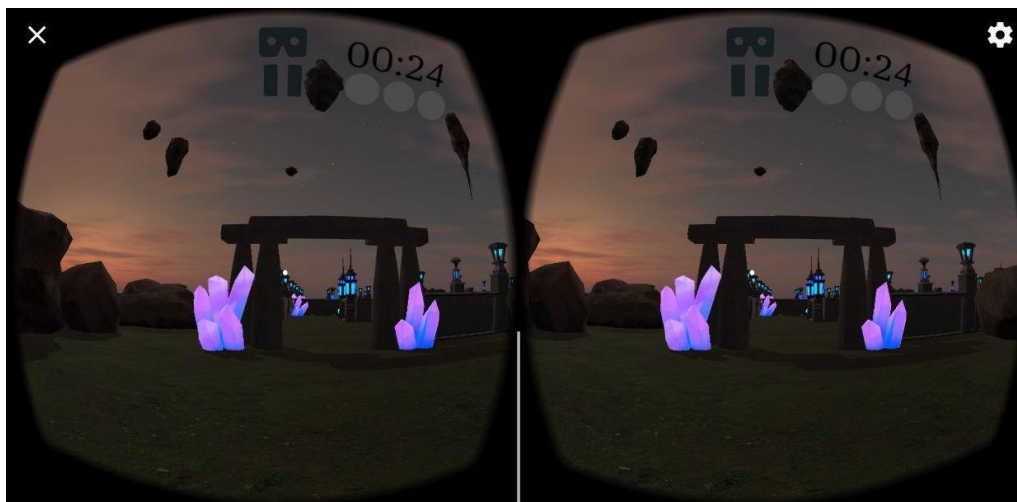
### Tiempo límite:

El tiempo límite estimado para este nivel es bastante ajustado, **100 segundos**, aunque se trate del nivel más bastante más largo y el porcentaje de caminos erróneos en el laberinto es mucho mayor, se ha considerado ajustar más el tiempo para que sea todo un reto conseguir las 3 estrellas de este último nivel.

### Vista inicial:



*Figura 80. Vista inicial del nivel 6 sin panel de inicio*



*Figura 81. Captura del nivel 6 desde móvil con el modo VR activado*

## 8. Requisitos funcionales y no funcionales

En esta sección se detallan todos los requisitos funcionales y no funcionales del sistema. Todos los requisitos están compuestos por un número o identificador único, de un título y de una descripción donde se explica de forma clara y concisa en qué consiste dicho requisito.

### 8.1. Requisitos funcionales

*Tabla 2. RF – 1. Seleccionar nivel*

Identificador	RF - 1
Título	Seleccionar nivel
Descripción	El usuario debe poder elegir qué nivel quiere jugar, para ello solo debe hacer clic en el menú principal sobre el botón que corresponde con el número del nivel deseado. De esta forma accederá al nivel siempre y cuando este esté desbloqueado.

*Tabla 3. RF – 2. Mirar o girar la cámara*

Identificador	RF - 2
Título	Mirar o girar la cámara
Descripción	El jugador debe poder mirar hacia todas las direcciones dentro del laberinto únicamente girando la cabeza, si lleva el visor con el smartphone sobre la cabeza o moviendo el smartphone hacia los lados en cualquier otro caso.

*Tabla 4. RF – 3. Caminar*

Identificador	RF - 3
Título	Caminar
Descripción	Para caminar por el laberinto el usuario solo tendrá que inclinar ligeramente la cabeza hacia el suelo en la dirección en la que quiera avanzar y el personaje se moverá automáticamente.

*Tabla 5. RF – 4. Parar de caminar*

Identificador	RF - 4
Título	Parar de caminar

Descripción	Para detener al personaje el jugador solo tendrá que dejar de mirar hacia el suelo, simplemente sobrepasando la línea de horizonte, y el personaje dejará de caminar, permitiendo al jugador observar el entorno y decidir su nuevo rumbo.
-------------	--

*Tabla 6. RF – 5. Pausar juego*

Identificador	RF - 5
Título	Pausar juego
Descripción	El jugador debe poder pausar un nivel en cualquier momento, para ello solo debe mirar hacia arriba, donde se encuentra el menú de pausa y hacer clic con los imanes de las Cardboard en el botón de pausa o esperar a que se complete el círculo de carga.

*Tabla 7. RF – 6. Reiniciar el nivel*

Identificador	RF - 6
Título	Reiniciar nivel
Descripción	Cuando el usuario se encuentre en el menú de pausa debe poder reiniciar el nivel en todo momento haciendo clic sobre el segundo botón del menú (Figura 55).

*Tabla 8. RF – 7. Volver al nivel*

Identificador	RF - 7
Título	Volver al nivel
Descripción	Cuando el jugador se encuentre en el menú de pausa debe poder volver al nivel y continuar jugando desde el mismo punto en el que lo pausó, haciendo clic en el tercer botón de dicho menú.

*Tabla 9. RF – 8. Salir de un nivel*

Identificador	RF - 8
Título	Salir de un nivel
Descripción	El jugador debe poder salir de un nivel e ir al menú principal entrando en el menú de pausa (RF - 5) y haciendo clic en el primer botón de este menú (Figura 55). En este caso el nivel se considerará no superado y por tanto no se le dará una valoración ni se desbloqueará el siguiente nivel.

*Tabla 10. RF – 9. Avanzar al siguiente nivel*

Identificador	RF - 9
Título	Avanzar al siguiente nivel
Descripción	En la pantalla de fin de nivel (Figura 57) el usuario debe poder avanzar al siguiente nivel del juego dando en el tercer botón de dicha pantalla.

*Tabla 11. RF – 10. Repetir nivel*

Identificador	RF - 10
Título	Repetir nivel
Descripción	En la pantalla de fin de nivel el usuario debe poder repetir el nivel que acaba de jugar haciendo clic en el segundo botón de dicha interfaz.

*Tabla 12. RF – 11. Ir al menú principal desde la pantalla de fin de nivel*

Identificador	RF - 11
Título	Ir al menú principal desde la pantalla de fin de nivel
Descripción	En la pantalla de fin de nivel el usuario debe ir al menú principal haciendo clic en el primer botón de dicha interfaz.

*Tabla 13. RF – 12. Desbloquear niveles*

Identificador	RF - 12
Título	Desbloquear niveles
Descripción	Cada vez que el jugador supere por primera vez un nivel se desbloqueará el siguiente y desaparecerá del menú principal el candado del botón de dicho nivel para que el jugador pueda acceder a él cuando desee. El nuevo nivel inicialmente tendrá una puntuación de 0 estrellas.

*Tabla 14. RF – 13. Guardar partida*

Identificador	RF - 13
Título	Guardar partida
Descripción	Cada vez que el jugador termine un nivel se guardará la partida para evitar perder el avance del usuario.

*Tabla 15. RF – 14. Cargar partida*

Identificador	RF - 14
Título	Cargar partida
Descripción	Cuando se habrá el juego se cargará el archivo guardado y se cargarán los datos de los niveles desbloqueados y las puntuaciones obtenidas para cada uno de ellos.

*Tabla 16. RF – 15. Valorar nivel*

Identificador	RF - 15
Título	Valorar nivel
Descripción	Cuando el usuario llegue a la salida del laberinto y pise el círculo para salir del nivel, se valorará su desempeño dentro de este y se le dará una valoración de entre una y tres estrellas, tal y como se especifica en el apartado de puntuación y avance en el juego dentro del punto 7.2 de este documento.

*Tabla 17. RF – 16. Cronometrar nivel*

Identificador	RF - 16
Título	Cronometrar nivel
Descripción	Cuando el jugador entre en un nivel se comenzará a cronometrar el tiempo que tarda hasta llegar al final del laberinto. Este cronómetro estará representado en la parte superior izquierda del H.U.D. para que el usuario pueda visualizarlo en todo momento.

*Tabla 18. RF – 17. Coger objetos*

Identificador	RF - 17
Título	Coger objetos
Descripción	El jugador debe poder coger los objetos recolectables del laberinto simplemente tocándolos o pasando sobre ellos.

*Tabla 19. RF – 18. Activar y desactivar realidad virtual*

Identificador	RF - 18
Título	Activar y desactivar realidad virtual

Descripción	En el menú de opciones que se encuentra sobre el usuario dentro de todos los niveles, se le dará al usuario la opción de desactivar y volver activar la realidad virtual del nivel.
-------------	---

*Tabla 20. RF – 19. Activar y desactivar sonido*

Identificador	RF - 19
Título	Activar y desactivar sonido
Descripción	En el menú principal se le dará al usuario la opción de desactivar y el sonido del juego, haciendo clic en el botón de la esquina inferior derecha.  Igualmente pasará con el botón que se encuentra en la esquina inferior derecha del menú de pausa.

## 8.2. Requisitos no funcionales

*Tabla 21. RNF – 1. Lenguaje de programación*

Identificador	RNF - 1
Título	Lenguaje de programación
Descripción	El juego se programará en C# ya que es el lenguaje utilizado para programar en Unity.

*Tabla 22. RNF – 2. Idioma del juego*

Identificador	RNF - 2
Título	Idioma del juego
Descripción	El juego se desarrollará en castellano y por tanto los textos del juego y los comentarios del código se encontrarán en este idioma.

*Tabla 23. RNF – 3. Controles*

Identificador	RNF - 3
Título	Controles
Descripción	Para que el juego funcione correctamente y sea posible moverse por el laberinto es necesario que el smartphone en el que se use posea giroscopio.



*Tabla 24. RNF – 4. Versión de Android*

Identificador	RNF - 4
Título	Versión de Android
Descripción	Para que funcione correctamente la aplicación en realidad virtual es necesario que el dispositivo en el que se ejecute tenga una versión de Android igual o superior a la Android 4.4 “Kitkat”.

*Tabla 25. RNF – 5. Tiempo real*

Identificador	RNF - 5
Título	Tiempo real
Descripción	El juego debe reaccionar en tiempo real a todas las interacciones del usuario, tanto en los menús como en los niveles de realidad virtual para evitar sensaciones de mareo y que se pierda la sensación de inmersión.

*Tabla 26. RNF – 6. Sencillo e intuitivo*

Identificador	RNF - 6
Título	Sencillo e intuitivo
Descripción	Todas las interfaces del juego serán muy sencillas e intuitivas para que todo el mundo pueda usar este juego.  Dentro del primer nivel se encuentra un tutorial para explicar los controles del juego, de esta forma el usuario siempre sabrá cómo debe interactuar.

## 9. Fases de la implementación

A continuación, se comentan a grandes rasgos las fases más importantes que se han llevado a cabo durante el desarrollo de *Celestial Road*.

### 9.1. Diseño de los mapas e interfaces

Para comenzar el desarrollo del proyecto, el primer paso fue diseñar como iban a ser los laberintos de cada nivel, intentando que la dificultad de los recorridos aumentara poco a poco tras cada nivel e intentando que ninguna de las soluciones de estos recorridos superase el minuto y medio. De esta forma, se evita que resulten demasiado complicados y tediosos para el usuario. Debido a que si el usuario se equivoca de camino y tiene que dar la vuelta se puede llegar a hacer muy largo el nivel, aburriendo al jugador e incluso llegando a marearle.

Se tomó la decisión de realizar los niveles más cortos y sencillos tras la implementación del actual nivel 6, inicialmente planteado como nivel 3, porque después de jugarlo en primera persona y en realidad virtual se vio que era demasiado largo y complicado.

Igualmente, algunos de los usuarios a los que se le dio a probar este nivel afirmaron que era demasiado complicado para ser un tan temprano en el juego y que el salto de dificultad era demasiado elevado para los primeros niveles del juego, aunque como reto de un nivel más avanzado podría ser una buena opción, por lo que se decidió que fuera el último nivel de *Celestial Road*.

Todos los bocetos de los niveles diseñados se pueden ver en el apartado 7.6. Niveles. La siguiente decisión que hubo que abordar fue elegir la temática o decoración de los laberintos del mundo de *Celestial Road*. Como ya se ha comentado en el apartado de diseño y ambientación, decidimos que se trataran de laberintos de alguna civilización antigua y para lo que se comenzaron a diseñar objetos con esta temática, que más tarde se modelarían con la herramienta *Blender*, figura 82.

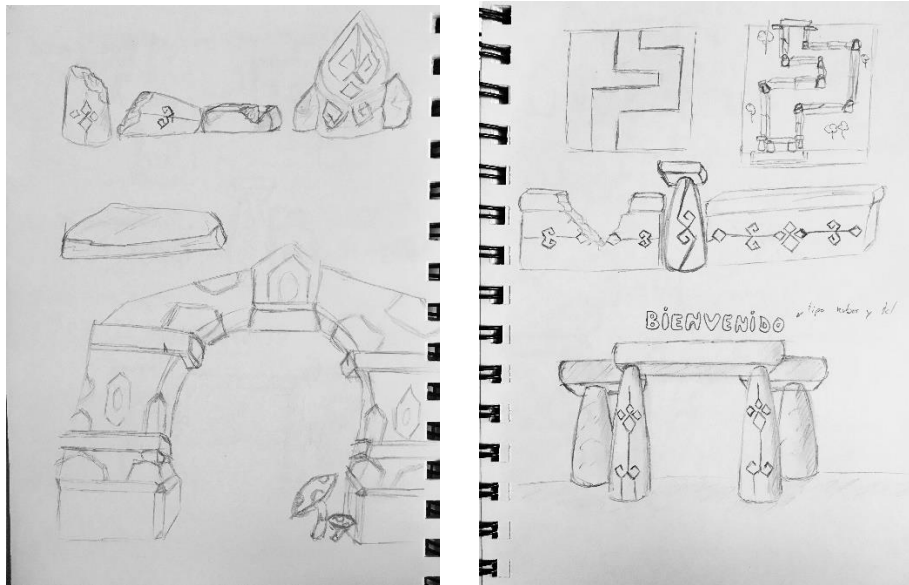


Figura 82. Bocetos de objetos que forman el laberinto

Una vez diseñados los mapas del juego y algunos de los objetos que los forman, se diseñaron las interfaces de los menús, los bocetos se pueden ver en el apartado de interfaces de este documento, junto con todos los botones e iconos necesarios para dichas interfaces, además del logo de la aplicación. Y se digitalizaron haciendo uso del programa *Adobe Illustrator*, figura 83.

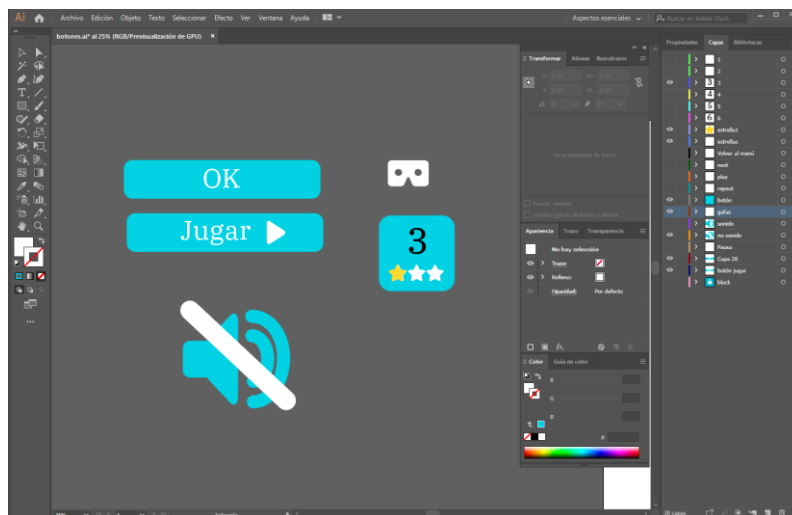


Figura 83. Diseño de botones en Adobe Illustrator

## 9.2. Modelado

La siguiente fase del desarrollo consistió en modelar los muros, columnas, farolillos y demás adornos que conforman los laberintos en *Blender* (figura 84).

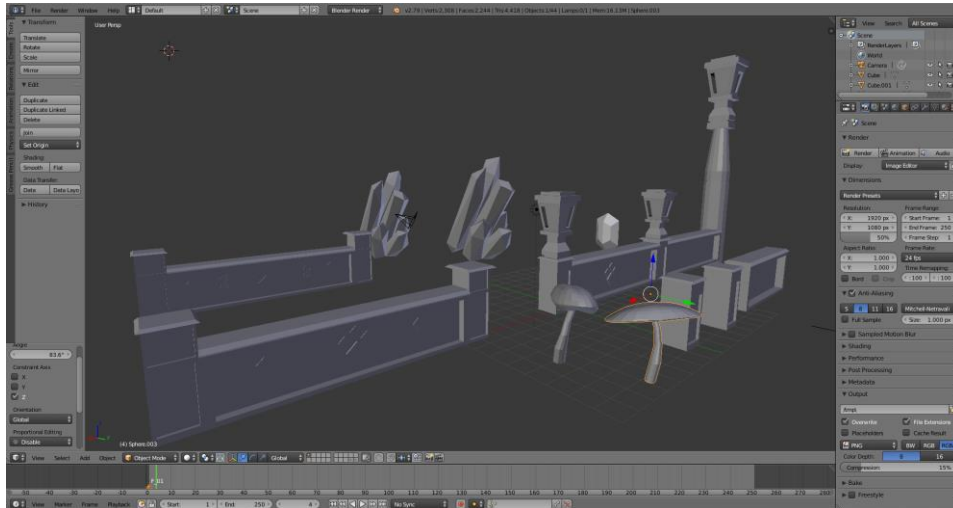


Figura 84. Modelado de objetos en Blender

En el anexo 0.1 se pueden ver todos los objetos modelados que conforman los laberintos de los distintos niveles.

Todos los objetos modelados se han intentado hacer lo más sencillos y con el menor número de vértices posible para evitar incrementar mucho el peso del juego. Para ello, no solo se han intentado hacer modelos sencillos, sino que además en aquellos más complicados se ha hecho uso del operador *Decimate* que reduce el número de caras de los objetos. Un ejemplo de esto se puede ver en la figura 85, en la que la primera seta consta de 386 caras y tras usar el modificador baja a 226 y luego a 138 caras.

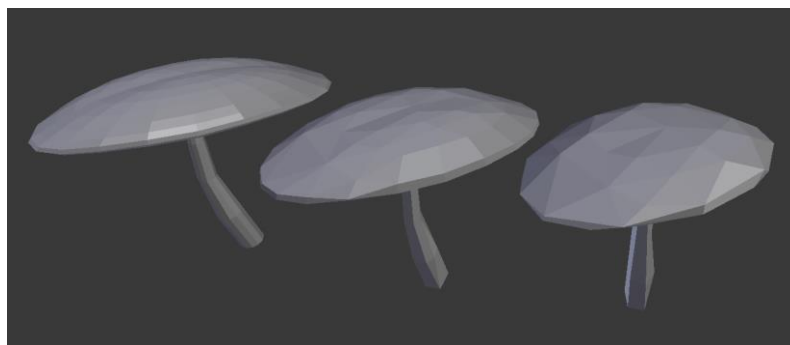


Figura 85. Reducción de caras de un objeto haciendo uso del operador Decimate.

Una vez que se ha terminado de modelar cada uno de los objetos, el siguiente paso consistía en mapear las coordenadas UV del objeto, necesarias para poder asignarle texturas.

Para comenzar, se le asigna un material al objeto, desde el menú que se encuentra a la derecha de la interfaz de *Blender*. Una vez hecho esto, hay que cambiar el modo de la interfaz de *Blender* a *UV Editing* y crear un nuevo archivo donde se guardará el mapa de texturas.

A continuación, hay que marcar las costuras del objeto, para lo que hay que seleccionar las aristas por las que se quiere que se desmonte el objeto y en el menú que aparece al *presionar Ctrl + E* seleccionar la opción de marcar texturas, *Mark Seam*. Las líneas de las aristas seleccionadas tomarán un tono rojizo, de forma que ya si seleccionamos todo el objeto dando a la letra A y en el menú que parece cuando se da a la U, seleccionamos la opción de *Unwrap* ya podemos ver como se dibuja el mapa de la textura del objeto en la parte izquierda de la pantalla (Figura 86).

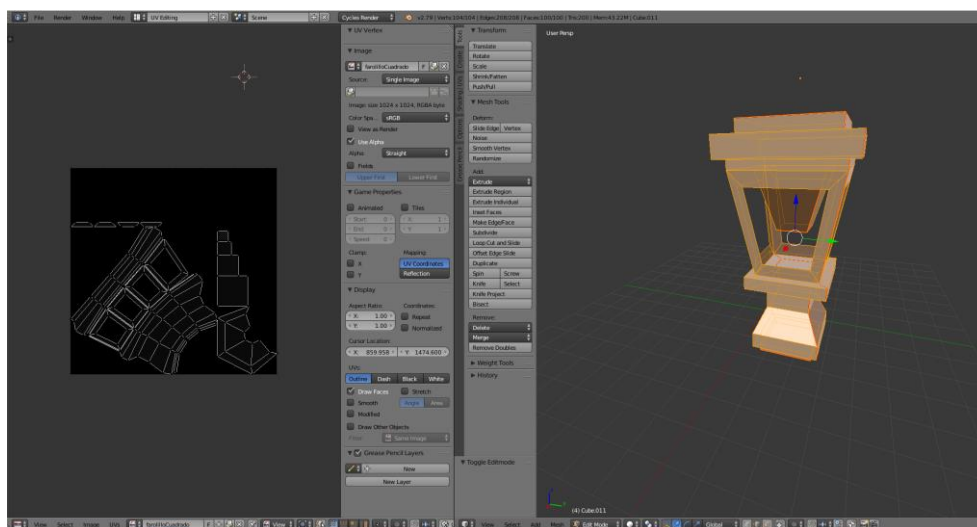


Figura 86. Vista de Blender en el modo UV Editing.

Llegados a este punto podemos pintar la textura, como veremos en el siguiente apartado, y exportar el objeto que ya posee coordenadas de texturas.

Para exportar el objeto y que luego se pueda introducir en Unity correctamente basta con dar a *File > Export > FBX* y en la nueva pantalla que aparece elegir donde guardar el modelo y cómo quieres nombrarlo.

En caso de tener más de un objeto en el mismo archivo se debe seleccionar solo el que se quiere exportar y marcar la opción de “Selected Objects” en el menú inferior izquierdo de la pantalla de guardado.

### 9.3. Creación de texturas

Una vez que los modelos ya tienen asignadas sus coordenadas UV, se puede comenzar a pintar la textura del objeto a partir de la imagen creada en el apartado anterior, parte izquierda de la figura 87.

Para ello, hay que asignar dicha imagen, previamente guardada, al color del material asignado al objeto. Para ver el resultado en el objeto se cambia el *Viewport Shading* a *Texture* en la parte inferior de la interfaz.

Una vez hecho esto, ya se puede comenzar a pintar la textura cambiando el modo a *Texture Paint*. De esta forma, pues ir eligiendo distintos pinceles y colores para ir pintando tanto sobre la imagen de la textura como sobre el objeto en sí.

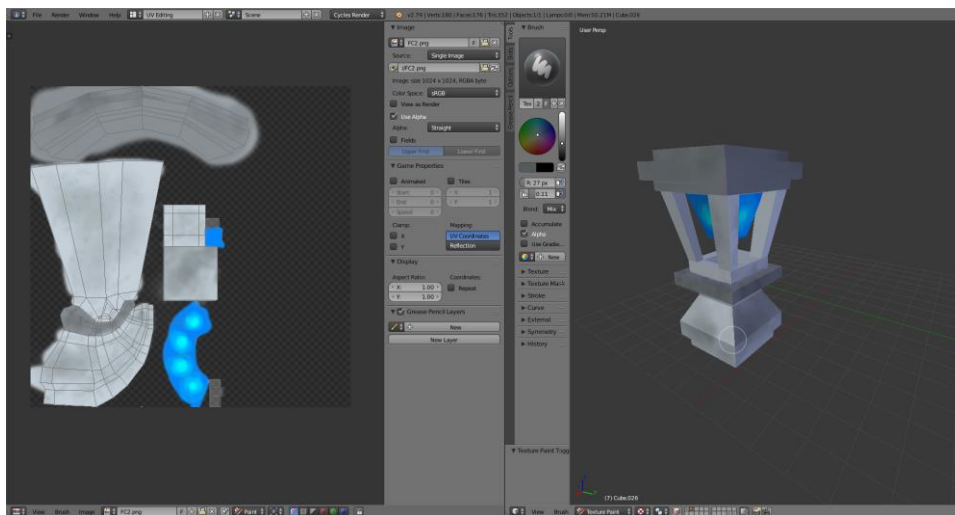


Figura 87. Vista de Blender en modo Texture Paint

Una vez que se ha terminado de pintar la textura se guarda, la imagen generada ya puede ser utilizada en nuestro motor de juego asignándosela a cualquier material o mejorarla y editarla en otros programas como Photoshop, hasta que esté como se quiera.

Como ya hemos obtenido el mapa *Albedo* del material, ahora importamos la imagen a Photoshop y vamos creando el resto de mapas, el normal para dar más profundidad o el de *emisión* si queremos que algo brille como es en este caso, figura 88.

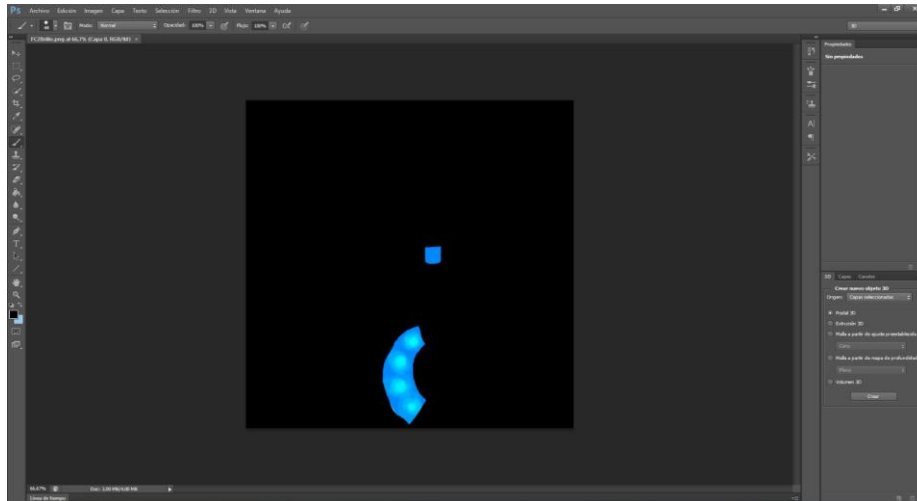


Figura 88. Creación del mapa de emisión en Photoshop

Por otro lado, comentar que la textura visible en el suelo de los niveles y los mapas de oclusión y normales de los muros del primer y segundo nivel son descargados de la web de Poliigon [52] para intentar lograr un mayor realismo en los niveles. La licencia de estas texturas permite su reutilización siempre que te registres en la plataforma y las texturas no puedan ser directamente descargadas desde el juego, en caso de que pudieran ser descargadas debe añadirse un licencia de texto que contenga las palabras *“Contains assets from Poliigon.com - Assets may not be redistributed”*.

Igualmente, las dos imágenes de texturas utilizadas como referencia para el mapa *Albedo* en las piedras que se encuentran al inicio y final de los niveles, fueron descargadas de Google imágenes, asegurando que su licencia permitía su reutilización y modificación, tanto de forma comercial como no comercial.

Aparte de estas, el resto de las texturas del juego, correspondientes a los distintos faroles, columnas, muros, antorchas o plataformas, fueron creadas siguiendo el proceso descrito anteriormente.

## 9.4. Importación a Unity

Para importar los modelos y las texturas creadas a *Unity* solo hay que arrastrarlas desde las carpetas en la que están ubicadas hasta la carpeta *Assets* que encontramos en la parte inferior de la interfaz de *Unity*, de esta forma los archivos ya pasan a formar parte del proyecto y pueden ser usados siempre que se quiera.

Además, dentro de *Unity* se puede crear un sistema de carpetas para poder ordenar y clasificar todos los archivos usados, por ejemplo, separando los modelos en una carpeta y las texturas en otra. De esta forma será mucho más rápido y sencillo encontrar los archivos necesarios en cada momento.

Una vez que ya están los archivos en *Unity*, es igual de sencillo incluirlos en una escena. Para incluir un modelo en una escena solo es necesario arrastrar el modelo que queremos de la carpeta en la que se encuentre a la parte de escena, tanto en el lateral izquierdo donde se ven los objetos presentes en la escena como en el propio panel de escena, situado en la parte central, donde podemos ver esta de forma gráfica (Figura 89). De esta forma el objeto se incluirá en la escena y ya se podrá colocar en el lugar que se desee y darle los atributos necesarios, visibles en la parte derecha.

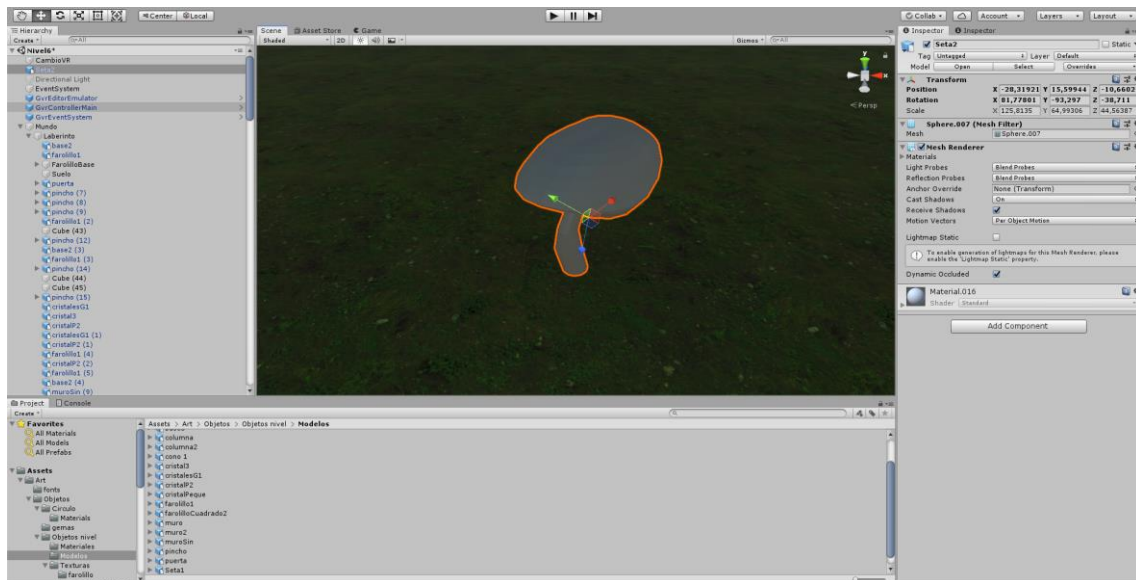


Figura 89. Modelo incluido en una escena en Unity

Al exportar un modelo, este no tiene ninguna textura ni material asociado por lo que se vuelve de color gris, como podemos ver en la figura 89. Para crear un material y poder asociarle las



texturas que se han importado anteriormente, solo es necesario hacer clic derecho > *Create > Material*, de esta forma se creará un nuevo material al que debemos asignarle la textura creada anteriormente y los mapas de normales, de emisión, etc.

Si hacemos clic en el nuevo material se abrirá la vista del inspector del material, donde se especifican todos los detalles de este, como se puede ver en la figura 90.

En el primer campo, *Albedo*, se debe colocar la textura que creada en *Blender* y *Photoshop*, correspondiente con los colores del objeto.

En caso de ser metálico se podría crear un mapa con las partes metálicas y asignarlo en el campo *metallic*, donde además se pueden elegir los valores de cuanto se quiere que brille y como de nítido tiene que ser el brillo.

En los siguientes campos se pueden asignar los mapas de normales, para dar profundidad; los de oclusión o los de emisión, en caso de que queramos que alguna parte del objeto brille y desprenda luz; entre otros.

Una vez creado el material solo hay que arrastrarlo hasta el modelo y ya se le asignará correctamente.

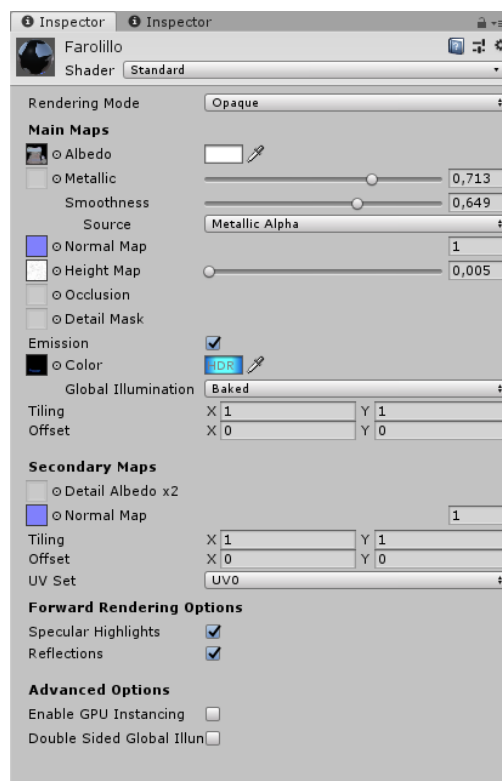


Figura 90. Detalles de un material en Unity

## 9.5. Programación y scripts necesarios

Para lograr que el juego funcione correctamente ha sido necesario poner nuevos atributos a muchos objetos como por ejemplo box colliders a todos los modelos que conforman el laberinto, para que el usuario no pueda atravesarlos o a algunos botones para que puedan ser pulsados y los reconozca el punto reticular de la mira.

Otro atributo muy importante que ha habido que incluir en muchos objetos como los botones o el círculo de teletransporte para finalizar el nivel han sido los scripts. Durante todo el desarrollo se han creado un gran número de scripts para controlar tanto las acciones del usuario como era que caminara o hiciera clic en algún botón como para acciones propias del juego como cambiar de escena, cambiar el modo de realidad virtual a normal para visualizar correctamente los menús, para pausar el juego y con ello el cronómetro o simplemente para calcular la valoración obtenida por el usuario al finalizar cada nivel.

Uno de los scripts más sencillos en cuanto a líneas, pero más difíciles en cuanto al tiempo necesario para saber cómo hacerlo funcionar correctamente, fue el script encargado del movimiento del menú de opciones que se encuentra siempre sobre el usuario, conseguir que le siguiera e incluso girara con el usuario llevo mucho más tiempo del esperado.

Aunque tras mucho investigar se vio que eran simplemente dos líneas que debían actualizarse en todos los frames del juego, figura 91; y que miraba la posición del usuario y giraban el menú hacia el mismo lado al que girara el usuario, el mayor problema fue el uso de quaternions, hasta el momento desconocidos y tener que acceder a la rotación de un solo eje de la cámara del jugador y conseguir este valor de forma que funcionara correctamente al pasarlo a quaternion.

```
void Update()
{
    miTransform.rotation = Quaternion.Euler(0, cam.transform.rotation.eulerAngles.y, 0);
    miTransform.position = cam.transform.position;
}
```

*Figura 91. Método que actualiza la posición del menú de opciones*

Otra parte importante en los scripts son las funciones que guardan el avance del usuario y la valoración que este ha obtenido en cada nivel. Sin estas funciones cada vez que el jugador abriera el juego habría perdido todas sus valoraciones y tendría que volver a superar los laberintos para desbloquear nuevamente los niveles.

En la figura 92 se puede ver una de las funciones que se ejecutan cuando el usuario termina un nivel, en ella se aumenta el número del nivel en el que se encuentra el jugador y se mira si ese nivel es mayor que el número de niveles desbloqueados, caso que sucede siempre la primera vez que se juega cada nivel, y en caso de ser mayor se guarda en el dispositivo el nuevo número de niveles desbloqueados, el cual se mirará en la página de inicio para ver cuántos niveles deben estar desbloqueados.

```
public void DesbloquearNivel()
{
    GlobalVariables.nivelActual++;
    GlobalVariables.nivelesDesbloqueados = PlayerPrefs.GetInt("levelsUnlock");

    if (GlobalVariables.nivelesDesbloqueados <= GlobalVariables.nivelActual)
    {
        GlobalVariables.nivelesDesbloqueados = GlobalVariables.nivelActual;
        PlayerPrefs.SetInt("levelsUnlock", GlobalVariables.nivelesDesbloqueados);
    }
}
```

*Figura 92. Función para guardar y desbloquear niveles*

## 10. Pruebas y validación

Todas las funcionalidades y requisitos descritos en el apartado 8 de este proyecto han sido probados y validados para asegurar el correcto funcionamiento del juego. Para llevar a cabo esta validación se ha **instalado** el juego varios **móviles diferentes**, tanto por su marca como por su **versión** de **Android**, de esta forma no solo se asegura el funcionamiento del juego en distintos dispositivos y en distintas resoluciones, también se valida su funcionamiento en móviles más antiguos o desactualizados.

Un ejemplo de los móviles en los que se ha probado han sido un *bq Aquaris E4.5* con una versión de *Android 4.4.2, "Kitkat"*, una de las versiones más antiguas con las que funciona *Celestial Road* debido a que la *API* de *Google* para el uso de *Cardboard* no soporta versiones anteriores. O, por otro lado, un móvil *Xiaomi Mi 8 Lite*, ya con una versión actual de *Android, Android 9 "Pie"*. En ambos el juego ha funcionado correctamente, permitiendo probar su validez para una amplia cantidad de versiones de *Android*.

Por otro lado, no ha sido posible probar el juego en dispositivos que funcionen con *IOS*, debido a que no disponía de ningún ordenador que me permitiera compilar el juego para dicho sistema.

En todos los móviles se ha comprobado el funcionamiento de los requisitos mencionados, asegurando que todos ellos funcionan correctamente en los distintos niveles del juego. Además, se ha comprobado que el juego permita avanzar correctamente por los niveles y guarde todos los avances y las preferencias del usuario.

Probar el juego en distintos dispositivos ha sido muy importante, ya que, al desarrollar *Celestial Road* para realidad virtual en el ordenador, no se podía comprobar el funcionamiento en realidad virtual y, por tanto, algunas de las funcionalidades que inicialmente parecían ir correctamente, una vez probadas en el smartphone no funcionaban como se esperaba. Esto conllevó un gran número de pruebas y mucho tiempo para poder compilar e instalar cada versión, aunque gracias a ello, todos los requisitos han sido probados y muchas de las funcionalidades han ido mejorado constantemente, hasta lograr un resultado pulido y completamente funcional.

Además, no solo se comprobó el funcionamiento de los requisitos básicos, también se tuvo en cuenta comprobar pequeños detalles como eran que todas las interfaces se vieran correctamente en distintas resoluciones, que el usuario no pudiera traspasar los muros del laberinto o, por ejemplo, que no pudiera ver a través de ellos al encontrarse muy cerca.

## 10.1. Validación

Dejando de lado las pruebas realizadas para comprobar el funcionamiento del juego, pasamos a hablar de la validación del juego. Durante el desarrollo del juego varios usuarios han tenido la oportunidad de probar las distintas versiones de Celestial Road y opinar sobre ellas.

### **Primera prueba**

En la **primera fase de pruebas**, posiblemente la más importante, los usuarios probaron tres laberintos del juego, los actualmente correspondientes con los niveles 1, 5 y 6 del juego. Tras esto, comprobamos que el incremento de la dificultad de los laberintos debía ser menor entre cada nivel para evitar que el usuario se aburriera y deseperara con niveles demasiado complicados, lo que supuso replantear nuevamente los laberintos desarrollados para lograr que fueran mucho más cortos y sencillos.

De igual forma, se decidió que el recorrido correcto de cada nivel no podía superar el minuto y medio de recorrido, ya que, si el usuario se perdía, el tiempo para superar el nivel podía ser muy elevado; igualmente, se vio que los caminos erróneos no debían ser demasiado largos, para evitar perder al usuario y que terminara abandonando el nivel.

Gracias a realizar esta prueba desde el inicio se evitó implementar laberintos demasiado largos y complicados, evitando una gran cantidad de tiempo y esfuerzo desaprovechado.

### **Segunda prueba**

La **segunda prueba** realizada ayudó a validar las interfaces de los menús del juego, para ello se les enseñaron las interfaces propuestas a los usuarios y se les preguntó que pensaban que hacía cada botón y si pensaban que debían colocarse en algún lugar diferente. En esta fase apenas hubo cambios en el diseño, pero ayudó a asegurar que el juego fuera claro e intuitivo.

Además, en esta fase se decidió que para conseguir que todo el funcionamiento estuviera claro se añadiera un pequeño tutorial en el primer nivel del juego, explicando el objetivo del nivel y los controles.

### **Tercera prueba**

En la **tercera fase** de pruebas, ya se presentó a aproximadamente doce usuarios una versión beta del juego con todas las interfaces y niveles implementados. Los distintos jugadores fueron probando el juego y anotando los errores que fueron encontrando junto con algunas mejoras o ideas que se podían implementar. Por otro lado, todos ellos se mostraron muy interesados por poder obtener el juego para seguir jugándolo y opinaron que la estética del juego y sobre todo los niveles era muy bonita y estaba muy trabajada, mucho más que en otros juegos que habían probado con anterioridad.

Además, se comprobó que el juego no provocaba sensación de mareo en la mayoría de personas, solo dos de ellas tras probar varios niveles seguidos sentían una pequeña sensación de mareo.

## 11. Conclusiones

En este apartado se comentan los resultados obtenidos, todo lo aprendido y las conclusiones extraídas después de realizar todo el proyecto.

### 11.1. Estado actual del juego

Tras meses de trabajo, Celestial Road se puede considerar finalizado, se han logrado incluir de forma exitosa todos los requisitos y funcionalidades definidas inicialmente. Además, ha pasado varias fases de pruebas que han ayudado a corregir muchos pequeños errores que inicialmente pasaron desapercibidos, de esta forma el juego está totalmente funcional y listo para ser probado por muchos más usuarios.

Igualmente, todos los objetivos definidos para el trabajo se han logrado cumplir, desde la investigación de la tecnología y de la competencia hasta el propio desarrollo del juego en realidad virtual.

Con todo ello, aunque el juego haya cumplido con los objetivos iniciales del proyecto, no quiere decir que este no pueda crecer y mejorar mucho más, durante el desarrollo han surgido nuevas ideas que podrían hacer aún más divertido y entretenido el juego de Celestial Road, aunque por falta de tiempo no se han podido implementar en esta primera versión del juego. En el siguiente apartado se detallan las mejoras y ampliaciones que se podrían hacer y en las que se seguirá trabajando para lograr que Celestial Road llegue a ser un juego imprescindible que todo el mundo quiera probar.

### 11.2. Mejoras y ampliaciones

Como ya hemos comentado, Celestial Road tiene muchas posibilidades de mejorar y crecer todavía. A continuación, se comentan algunas de las mejoras y ampliaciones que se irán implementando en las próximas versiones del juego.

- **Nuevos niveles:** cómo se trata de un juego con niveles independientes una de las mejoras más claras que se puede incluir es expandir el número de niveles disponibles

en el juego. Subiendo poco a poco la dificultad de los laberintos se podrían ir incluyendo todos los niveles deseados.

- **Añadir animaciones:** para lograr hacer más vistoso el juego habría que incluir animaciones tanto en los menús como dentro de los niveles, como puede ser que haya pájaros o intentar simular aire o cualquier otra animación que ayude a aparentar un entorno vivo y real.
- **Añadir puzzles y nuevos desafíos:** para hacer el juego más entretenido se podrían incluir puzzles y enigmas que el usuario deba resolver para poder desbloquear el camino correcto, un ejemplo podría ser esconder números por el laberinto con los que deba abrir una cerradura para lograr salir del nivel.
- **Incluir nuevos objetos ocultos:** otros de los elementos que se podrían incluir son objetos ocultos en algunos niveles para intentar que el usuario quiera volver a jugarlos. Cuando el usuario consiga todos los objetos de un nivel se le avisaría y señalaría poniendo una medallita en la esquina del botón de ese nivel del menú principal.
- **Poner todo el juego en VR:** una importante mejora podría ser dar al usuario la posibilidad de elegir si quiere que todo el juego este en realidad virtual o no, de esta forma cuando el usuario finalice un nivel no tendrá que quitarse el visor para seleccionar la opción de avanzar al siguiente nivel.
- **Incluir sistema de ayuda:** se podría incluir un sistema de ayuda que dibujara el mapa del laberinto en la esquina superior conforme el jugador vaya avanzando. De esta forma, aunque se pierda siempre sabrá por donde puede volver y que caminos no ha explorado.

Otro posible rumbo que podría tomarse en un futuro sería orientar Celestial Road al campo de la educación, creando laberintos con distintos colores y con nuevos caminos podría ser utilizado para ayudar a mejorar la orientación tanto física como temporal de niños con problemas de este ámbito. Este nuevo punto de vista fue aportado por uno de los usuarios que probó la versión beta del juego en la tercera fase de pruebas, este usuario se trata de una psicóloga, que me aportó nuevas ideas y consejos para lograr orientar el juego a esta nueva funcionalidad.

### 11.3. Nociones aprendidas

Desarrollar este juego desde cero me ha permitido aprender mucho no solo de programación y de tecnologías hasta ahora casi desconocidas, como era Unity, sino que también me ha dado la



oportunidad de conocer un poco más de la historia de la Realidad Virtual, un campo que desde siempre me ha resultado muy llamativo e interesante.

Comenzar a usar nuevas herramientas, como es el caso de Unity, siempre es complicado, pero gracias a ello he tenido la oportunidad de conocer un nuevo modo de hacer aplicaciones, para mi gusto bastante sencillo e intuitivo, que creo que me será muy útil en el futuro.

Gracias al diseño actual del juego, con interfaces 2D y niveles 3D, he podido aprender a desarrollar ambos tipos de entornos, viendo cómo funcionan las interfaces en los dos modos de visualización. Además, debido a la inclusión de la Realidad Virtual me he visto forzada a pensar nuevos tipos de interacciones que permitieran al usuario interactuar con los menús en 3D de la forma más cómoda posible y de manera que se pudiera jugar Celestial Road haciendo uso de cualquier visor, e incluso sin hacer uso de un visor. Por tanto, para conseguir que todo funcionara e interactuara como quería he tenido que investigar y aprender mucho más sobre los atributos disponibles en Unity y sobre las funcionalidades que aporta la API de Google para desarrollar para Cardboard en este motor.

Por otro lado, este proyecto me ha dado la oportunidad de ampliar mis conocimientos de modelado y sobre todo de texturizado. Crear tantos materiales distintos, uno distinto para cada objeto, con unas características diferentes como era que fuera metálico, que una parte del objeto brillara, etc.; me ha obligado a conocer cómo hacer todos los mapas de textura a partir del mapa inicial, tarea que hasta el momento nunca había tenido que realizar.

Por último, comentar dos de los aprendizajes más importantes de este proyecto. El primero de ellos es la importancia de la planificación, llevar a cabo sola un trabajo tan amplio ha sido todo un reto contra el tiempo y más en estos meses de vacaciones de verano tras todo un año de trabajo, donde las distracciones y contratiempos han sido constantes. Aun así, gracias a una buena planificación desde el inicio del proyecto ha sido posible sacarlo adelante y finalizarlo para la fecha prevista. En segundo lugar, destacar la importancia de ir validando los avances del proyecto antes de avanzar tanto que no sea posible corregirlos. Ir enseñando las distintas versiones del juego a posibles jugadores ha resultado un punto clave, ya que ha permitido corregir algunas de las decisiones tomadas y ahorrar mucho tiempo de desarrollo corrigiendo partes ya implementadas. Igualmente ha permitido conseguir un resultado más amigable para el usuario y validado por varios jugadores.

#### 11.4. Resultado y conclusión final

Tras meses de trabajo por fin se puede decir que Celestial Road es un juego terminado, con un resultado muy satisfactorio personalmente. Se ha logrado completar una primera versión del juego que cumple todos los objetivos previstos en este trabajo y que, además, ya se encuentra disponible en la tienda de aplicaciones de GooglePlay para que muchos otros usuarios puedan probarla y dejar sus comentarios. Celestial Road se encuentra disponible en:

<https://play.google.com/store/apps/details?id=com.AlbaGA.CelestialRoad>.

El tiempo dedicado a este proyecto ha cumplido con el estimado para un trabajo de fin de grado, aunque me hubiera gustado disponer de un poco más de tiempo para conseguir incluir algunas de las mejoras descritas anteriormente. Igualmente estoy muy contenta con el resultado y más tras todo el trabajo realizado en fechas tan complicadas como han sido las vacaciones de verano.

Finalmente comentar que haber logrado completar este trabajo de forma individual tras todo el esfuerzo dedicado me aporta una gran sensación de orgullo y satisfacción, dándome una nueva motivación para afrontar el próximo proyecto que me depare el futuro.

## Bibliografía

1. Muñoz, R. *El número de líneas móviles supera por primera vez a la población mundial*. Periódico “El País”. 2018. Disponible en: [https://elpais.com/tecnologia/2018/02/27/actualidad/1519725291\\_071783.html](https://elpais.com/tecnologia/2018/02/27/actualidad/1519725291_071783.html)
2. Informe de GSMA. 2019. Disponible en: <https://www.gsma.com/r/mobileeconomy/>
3. Sempere, M. Apuntes asignatura de realidad virtual. 2019.
4. Trinick, C. *Sensorama*. 1962. Disponible en: <https://proyectoidis.org/sensorama/>
5. 10 hitos de la historia de la Realidad Virtual. Disponible en: <http://www.deusens.com/hitos-historia-realidad-virtual/>
6. Breve historia de la realidad virtual. Disponible en: <https://evemuseografia.com/2018/03/30/breve-historia-de-la-realidad-virtual/>
7. Mazuryk T. y Gervautz M. (2015): *Virtual Reality: History, Applications, Technology and Future*. Institute of Computer Graphics, Vienna University of Technology, Austria.
8. Oculus. Dispositivo de VR. Disponible en: <https://www.oculus.com/>
9. Información sobre las gafas Oculus. Dispositivo de VR. Disponible en: <https://www.gafasoculus.com/rift/>
10. HTC Vive. Dispositivo de VR. Disponible en: <https://www.vive.com/us/product/vive-virtual-reality-system/>
11. PlayStation VR. Dispositivo de VR. Disponible en: <https://www.playstation.com/es-es/explore/playstation-vr/>
12. Samsung Gear VR. Dispositivo de VR. Disponible en: <https://www.samsung.com/es/wearables/gear-vr-sm-r325nzvaphe/>
13. Cardboard. Dispositivo de VR. Disponible en: <https://vr.google.com/cardboard/>
14. Sites in VR. Aplicación para smartphone. Disponible en: <https://play.google.com/store/apps/details?id=air.com.ercangigi.sitesin3d>
15. InMind VR. Aplicación para smartphone. Disponible en: <https://play.google.com/store/apps/details?id=com.nivalvr.inmind>
16. El cuerpo humano en 3D. Aplicación para smartphone. Disponible en: <https://play.google.com/store/apps/details?id=com.rendernet.humanmale>
17. Jurassic VR. Aplicación para smartphone. Disponible en: <https://play.google.com/store/apps/details?id=com.lunagames.jurassicvr>

18. Google Spotlight Stories. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.google.android.spotlightstories&hl=es>
19. Página oficial de Google Spotlight Stories. Disponible en:  
<https://atap.google.com/intl/es/spotlight-stories/>
20. Within. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.shakingearthdigital.vrsecardboard&hl=es>
21. VR Cave. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.vw.vrcave>
22. VR Thrills: Roller Coaster 360. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.rabbitmountain.rollercoaster>
23. VR Roller Coaster Temple Rider. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.rabbitmountain.templecoaster>
24. Laberinto VR Maze. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=vrlabyrinth.maze.vr&hl=es>
25. Maze VR Forest. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.arloopa.mazeforestvr>
26. Maze VR - Cardboards. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.xtech.labirint>
27. 3D Maze (The Labyrinth). Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.RAM.AMAZINGMAZE&hl=es>
28. Maze World 3D. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=ru.empiregames.maze3d&hl=es>
29. Laberinto 3D II. Aplicación para smartphone. Disponible en:  
<https://play.google.com/store/apps/details?id=com.mobadu.Maze2&hl=es>
30. Motores de juegos - ¿cómo funcionan? Disponible en: <https://unity3d.com/es/what-is-a-game-engine>
31. Carrasco, A. *¿Qué es un motor de videojuegos?* 2018. Disponible en :  
<https://blogs.upm.es/observatoriogate/2018/07/04/que-es-un-motor-de-videojuegos/>
32. Ward, J. *What is a Game Engine?* 2008. Disponible en:  
[https://www.gamecareerguide.com/features/529/what\\_is\\_a\\_game\\_.php](https://www.gamecareerguide.com/features/529/what_is_a_game_.php)
33. Unreal Engine. Motor de videojuegos. Disponible en:  
<https://www.unrealengine.com/en-US/>
34. Unity. Motor de videojuegos. Disponible en: <https://unity.com/>

35. Cry Engine. Motor de videojuegos. Disponible en: <https://www.cryengine.com/>
36. Mula, J. *Pros y contras de programar en Unity vs en Unreal Engine*. 2017. Disponible en: <https://www.deustoformacion.com/blog/disenio-produccion-audiovisual/pros-contras-programar-unity-vs-unreal-engine>
37. 8 Ventajas de Blender frente a 3DSMax, Maya o XSI. Disponible en: <http://www.digital-rebel.com/8-ventajas-de-blender-frente-3dsmax-maya-o-xsi/>
38. Blender. Herramienta de modelado 3D. Disponible en: <https://www.blender.org/>
39. Educba. *Maya vs 3ds Max vs Blender*. Disponible en: <https://www.educba.com/maya-vs-3ds-max-vs-blender/>
40. Educba. Differences Between 3DS Max vs Blender. Disponible en: <https://www.educba.com/3ds-max-vs-blender/>
41. 3DS MAX. Herramienta de modelado 3D. Disponible en: <https://www.autodesk.es/products/3ds-max/overview>
42. Leal, B. Metodología Scrum en proyectos digitales. 2017. Disponible en: <https://blog.ida.cl/estrategia-digital/metodologia-scrum-en-proyectos-digitales/>
43. Trello. Herramienta para administrar proyectos. Disponible en: <https://trello.com/>
44. GitHub. Herramienta para gestión de versiones de código. Disponible en: <https://github.com>
45. Google Drive. Herramienta de almacenamiento para copias de seguridad. Disponible en: [https://www.google.com/intl/es\\_ALL/drive/](https://www.google.com/intl/es_ALL/drive/)
46. Documentación en Videojuegos: Documento de diseño (GDD). Disponible en: <https://eldocumentalistaudiovisual.com/2015/02/06/documentacion-en-videojuegos-documento-de-diseno-gdd/>
47. Jamendo. Música gratuita con licencia para su reutilización. Disponible en: <https://www.jamendo.com>
48. Obras musicales presentes en los niveles del juego y en la pantalla de fin de nivel. Disponibles en: <https://www.jamendo.com/artist/490314/filuanddina>
49. Licencia Creative Commons, CC BY-NC 4.0. Disponible en: <https://creativecommons.org/licenses/by-nc/4.0/>
50. Obra musical presente en el menú inicial del juego. Disponibles en: <https://www.jamendo.com/track/1604168/celtic-knight>
51. Licencia Creative Commons, CC BY-NC-ND 4.0. Disponible en: <https://creativecommons.org/licenses/by-nc-nd/4.0/>
52. Texturas gratuitas con licencia para reutilización. Disponible en: <https://www.poliigon.com>



# Anexo

## A. Objetos modelados

En este anexo se pueden ver todos los objetos que se han modelado para crear los distintos niveles del juego. Algunos se han modificado para conseguir el tamaño deseado para los muros y estas modificaciones no se han incluido en este anexo ya que son muy similares a los ya presentes en las imágenes.

En la figura 93 podemos ver los objetos presentes en los primeros niveles del juego como son el muro redondeado y las columnas, junto con los distintos faroles, antorchas y la puerta de entrada a los laberintos, presentes en casi todos los niveles.

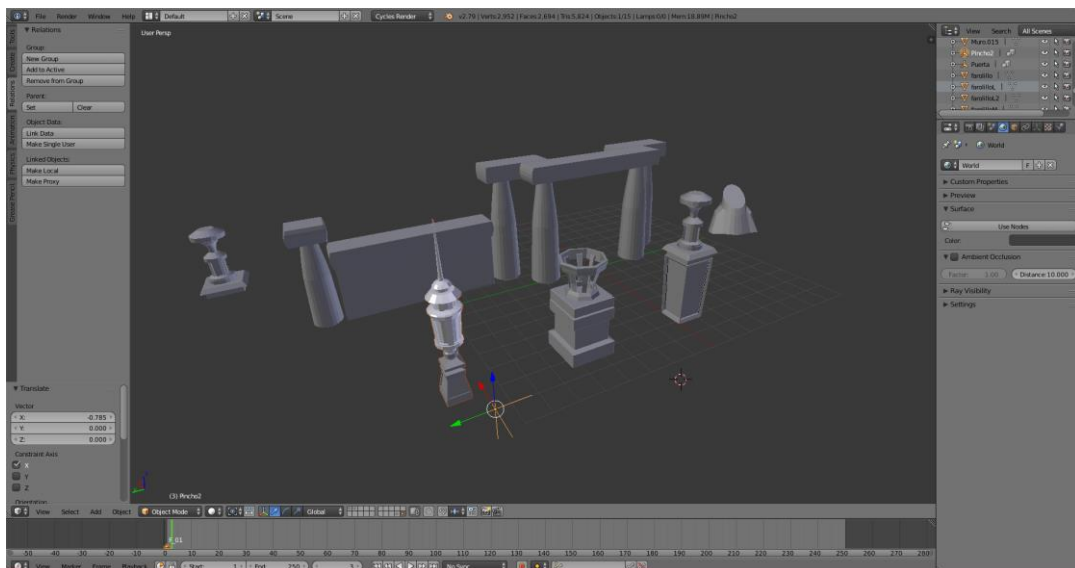


Figura 93. Objetos iniciales y comunes entre niveles

En la figura 94, podemos ver los nuevos muros, columnas y adornos, como son los cristales o las setas, que se van incluyendo en los niveles 4, 5 y 6 del juego, cuando se supone que la civilización del juego ha avanzado, estos muros son mucho más rectos y refinados que los anteriores para lograr que cierren mejor entre ellos y dar una sensación de avance tecnológico en la civilización que creó el laberinto.



Figura 94. Objetos y adornos de los últimos niveles

En la siguiente imagen podemos ver el modelo del círculo de fin de nivel, el cual debe pisar el jugador para conseguir salir del laberinto y finalizar el nivel.

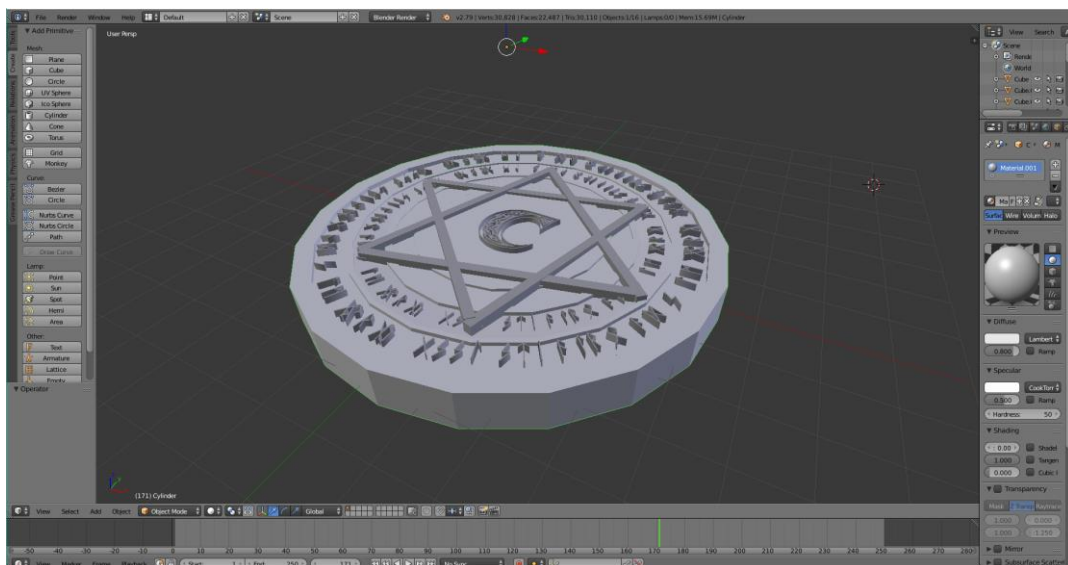


Figura 95. Círculo de fin de nivel

A continuación, se pueden ver las piedras modeladas para que el usuario no pueda salir del nivel, se encuentran tanto al inicio como a la salida de todos los niveles, de esta forma el único camino para superar el nivel es el diseñado como solución del laberinto.



Estos modelos son muy simples pero una vez aplicada la textura el resultado es bastante bueno. Para conseguir que pareciera que hay un mayor número de piedras diferentes, estos modelos fueron escalados en los distintos ejes dentro de Unity.

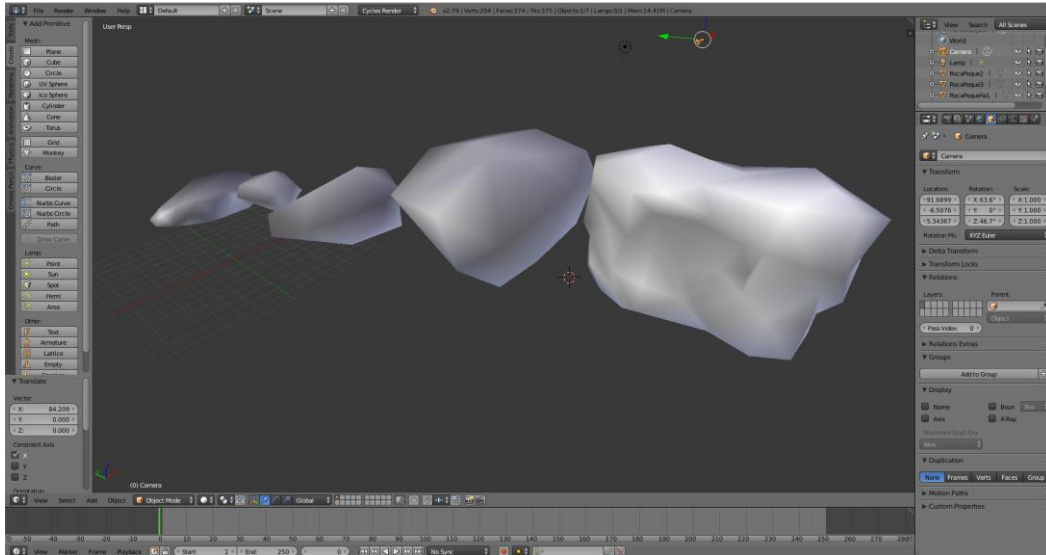


Figura 96. Piedras del suelo del laberinto

Por último, se pueden ver los modelos de las piedras que flotan sobre los laberintos, igual que en el caso anterior dentro de Unity se modificaron sus dimensiones, se deformaron en distintos ejes y se les asignaron distintos materiales para intentar aparentar que cada isla flotante era distinta.

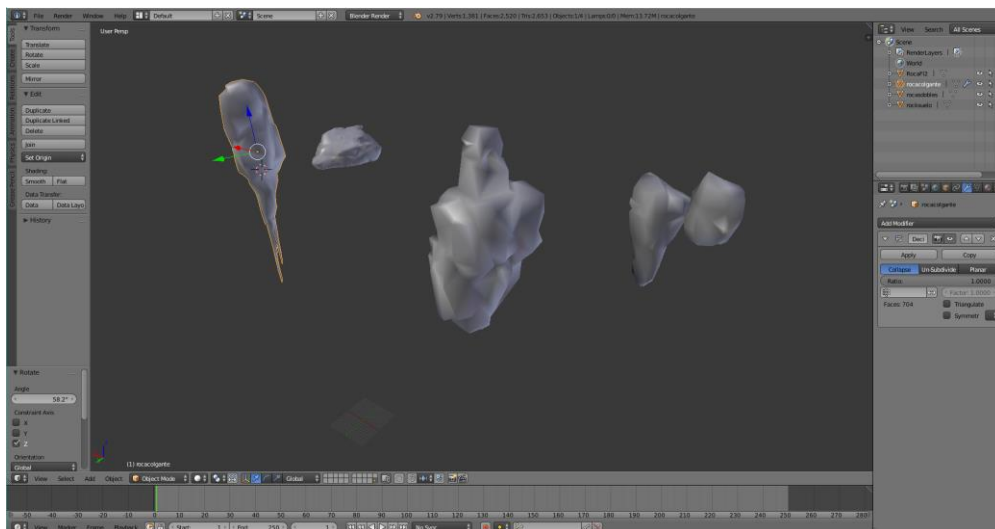


Figura 97. Piedras que flotan sobre el laberinto

