

# TFG

---

## VIDEOJUEGO EN FASE ALFA. DISEÑO Y MODELADO DE NIVELES

Presentado por Rubén Calap Quintana  
Tutor: Francisco Giner Martínez

Facultat de Belles Arts de Sant Carles  
Grado en Bellas Artes  
Curso 2015-2016



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA  
FACULTAT DE BELLES ARTS DE SANT CARLES

## RESUMEN

En este documento expondré mi trabajo final de grado, de carácter práctico, el cual consiste en el diseño y elaboración de los distintos niveles que aparecerán en un videojuego, actualmente en desarrollo, utilizando para ello el motor gráfico para videojuegos UNITY.

El videojuego en cuestión, está siendo elaborado actualmente por una única persona. Debido a este hecho, las características de producción -recursos y personal- son limitadas, incluso para un juego de carácter independiente. He asumido la realización en solitario tanto de personajes, diseño, escenografía, modelado 3D, materiales, texturas, animación, sonido y programación. Este juego, de título por determinar, será una experiencia para un solo jugador.

Ambientado en diferentes entornos, basados tanto en entornos reales como de propia creación imaginativa, el personaje ira avanzando por los distintos niveles utilizando su inteligencia, percepción y habilidad para llegar hasta el final del recorrido. Los niveles cuentan tanto con escenas de plataformas, como con laberintos, puzles y zonas de exploración. En cuanto al apartado artístico y gráfico, se ha optado por modelos de alto nivel de polígonos y un apartado visual realista (no fotorrealista), por este motivo estará destinado a equipos de gama media-alta, los requisitos aparecerán más adelante en el apartado optimización, para poder mover con suavidad y con la máxima calidad gráfica la aplicación.

Como resultado práctico presentaré todos los niveles jugables realizados hasta la fecha dentro de un archivo ejecutable.

El desarrollo de la historia no irá vinculado con la presentación.

Palabras Clave: Unity, Highpoly, Lowpoly, videojuego, fase" alpha", diseño, niveles, jugabilidad, Walking Simulator, edición 3D.

In this document I will show my practical final degree work, which consists in the use of the graphic engine UNITY to design and create the different levels of a videogame, currently in development.

The game is currently being developed by a single person. Due to this fact, the characteristics of production -resources and staff- are limited, even for an independent game. I have assumed character creation, design, set design, 3D modeling, materials, textures, animation, sound and programming. This game, now without title, will be a single-player experience.

Set in different environments, based both in real and imaginary environments, the main character will move through the different levels using their intelligence, perception and ability to reach the end of the journey. The game levels have platforms, mazes, puzzles and exploration areas. Regarding the artistic and graphic component, high-polygon models and realistic visuals (non-photorealistic) have been chosen. For this reason, the game will be intended for mid-end computer. The requirements to move smoothly and with maximum quality graphics this application will appear below in the optimization section.

As a practical result I will present all levels, which contains the objects, characters and functions created up to date, within an executable file. The development of the story will not be part of the presentation.

Keywords: UNITY, Highpoly, lowpoly, video game, "alpha" state, design, levels, gameplay, Walking Simulator, 3D editing.

## Índice

1. Introducción	5
2. Motivación	7-8
3. Objetivos	8-9
4. Metodología	9
5. Desarrollo de videojuegos	10
5.1. Compañía multinacional	10
5.2. Estudio independiente	11-12
5.3. Lobos solitarios	12-13
6. Referentes	13
6.1 Referentes Gráficos	13-16
6.2 Referentes de jugabilidad	16-19
7. Producción	19
7.1. Contexto	19-22
7.1.1.Workflow	22-25
7.1.2.Programas utilizados	25-28
7.2. Elementos utilizados	29
7.2.1.Props	29-31
7.2.2.Texturas	32-35
7.3. Definición de los niveles	35
7.3.1.Niveles exteriores	35-39
7.3.2.Niveles ficticios	39-40
7.3.3.Interiores	40-41
8. Testeo	41
9. Optimización	42
10. Conclusiones	43
11. Bibliografía	43-44
11.1. Material online	44
12. Anexos	44

# 1.INTRODUCCIÓN

En esta memoria expondré mi trabajo de final de grado de carácter práctico, el cual consiste en el diseño y elaboración de los distintos niveles gráficos de un videojuego. Prescindiendo de la parte narrativa, aparecerán en el videojuego todos los niveles realizados, que están actualmente en desarrollo, utilizando para ello el motor gráfico para videojuegos UNITY<sup>1</sup>, motor gráfico adaptado para ser multiplataforma. En este caso, el desarrollo solo se realizará para ordenadores compatibles.

El videojuego en desarrollo, está siendo elaborado actualmente en solitario, debido a experiencias infructuosas con equipos de proyectos anteriores. Por dicho motivo, las características de producción-recursos y personal- son realmente limitadas incluso para un juego de carácter independiente. He asumido la realización en solitario tanto de personajes, como de diseño, escenografía, modelado 3D, materiales, texturas, animación, sonido y programación.

Este juego, de título por determinar, será una experiencia para un solo jugador, que se embarcará en un viaje a través del cual se visitarán entornos con diferentes ambientaciones.

En estos entornos tendremos que realizar una exploración de los mismos y superar los retos que se nos presenten para desbloquear el siguiente nivel, ya sea mediante llaves ocultas por los amplios territorios que exploraremos, basándonos en pistas tanto visuales como auditivas, o haciendo uso de nuestras habilidades cognitivas. Pasearemos por el terreno creado siguiendo las pistas que se nos den hasta encontrar la forma de abrir la puerta que nos lleve a la siguiente zona. Al tratarse de un juego en el que básicamente exploramos el entorno podemos determinar que las referencias más directas

---

<sup>1</sup> Es un motor de videojuegos multiplataforma, creado por Unity Technologies.  
<https://unity3d.com/es>

con los videojuegos sería “DEAR ESTHER”<sup>2</sup>, un juego en el que paseas con una voz en off que nos narra sucesos del mapeado, o como el videojuego independiente “THE WITNESS”<sup>3</sup> basado en puzles y laberintos; por estas características el videojuego se enmarcaría en el género “Walking simulators”.

Nuestro personaje irá avanzando por los distintos niveles utilizando su inteligencia, percepción y habilidad para llegar hasta el final del recorrido. Los niveles contarán tanto con escenas de plataformas al estilo de “MARIO BROS”<sup>4</sup>, en las que nuestra habilidad en el manejo del personaje sea la que necesariamente deba predominar, ya sea saltando, corriendo o manteniendo el equilibrio... como puzles en los que contará más la capacidad de memorización, la percepción perspectiva, del color o el sonido. Todo ello en las ya mencionadas zonas de exploración en las que buscaremos las llaves, pistas o caminos en los amplios terrenos concebidos para llegar al siguiente nivel. Nuestro personaje se encontrará oscilando entre un entorno conocido, su habitación, y otros entornos, basados en la realidad y la imaginación. Podremos estar paseando por un bosque y en la siguiente escena estar en las cumbres de unas montañas nevadas o en el mismísimo desierto. Por otra parte, también visitaremos lugares que nada tienen que ver con la realidad. Un universo de laberintos, escaleras flotantes o habitaciones sin ninguna lógica.

*“No nos arriesguemos a definir lo fantástico (...) Intentemos más bien delimitar el territorio de lo fantástico precisando sus relaciones con los dominios vecinos. (...) La narración fantástica, por el contrario, se deleita en presentarnos a hombres como nosotros, situados súbitamente en presencia de lo inexplicable, pero dentro de nuestro mundo real.”<sup>5</sup>*

En cuanto al apartado artístico y gráfico, se ha optado por modelos de alto nivel de polígonos y una componente visual realista, por este motivo, estará destinado a equipos con considerable potencia, para poder mover con

---

<sup>2</sup> Ver Referentes de jugabilidad pag.12

<sup>3</sup> Ver Referentes de jugabilidad pag.12

<sup>4</sup> Ver Referentes de jugabilidad pag.12

<sup>5</sup> VAX, LOUIS. Arte y Literatura fantásticas. Bs.As., Eudeba, 2da. ed., 1971.

suavidad y con la máxima calidad gráfica el título. A pesar de ello, se procurará que esté lo mejor optimizado posible para alcanzar una tasa de imágenes por segundo aceptable en cada una de sus opciones visuales.

Como resultado práctico aparecen presentes los niveles jugables, en su “fase alpha”<sup>6</sup>, así como todos los modelos de escenario, objetos, personajes, texturas, materiales y funciones dentro del juego. La cuantía de niveles jugables son tantos como el tiempo disponible y mi grado de autoexigencia me ha permitido realizar, en torno a los 6 o 7, con la posibilidad abierta de aumentarlos más adelante. A pesar de que los diferentes niveles están justificados por una trama y un guion argumental a seguir, el desarrollo de la historia no irá vinculado con la presentación, ya que será lo último en lo que trabajaré una vez el proyecto entre en una fase “beta”<sup>7</sup> y como es obvio, por el tiempo que requiere, tendrá que ser pospuesto a un periodo del desarrollo del videojuego más tardío.

## 2.MOTIVACIÓN

El desarrollo de este proyecto fue iniciado por un grupo de tres personas y la propuesta del videojuego inicialmente iba a ser de un FPS<sup>8</sup>, con similitudes jugables a las vistas en juegos como “DOOM”<sup>9</sup>. Por problemas con las fechas establecidas y con el mismo desarrollo, se descartó la idea inicial y se empezó prácticamente de cero, con un RPG<sup>10</sup> de perspectiva Isométrica. Al ser un equipo independiente, nuestro referente principal iba a ser el juego independiente “STASIS”<sup>11</sup>. Por dificultades sobrevenidas para el trabajo en equipo, fueron rechazadas ambas ideas, cancelados los proyectos y desmontado el grupo de trabajo, con la pérdida de 4 meses de desarrollo. A pesar de ello, todo ese tiempo sirvió para ampliar conocimientos y descubrir

---

<sup>6</sup> Fase donde un producto todavía es inestable, y está abierto a cambios. A pesar de sus errores a corregir, su funcionalidad tiene que ser completa.

<sup>7</sup> Primera versión completa del producto donde no se le añadirá nuevo contenido. Solamente se harán pequeñas ediciones o se corregirán errores.

<sup>8</sup> First Person Shooter, juego de disparos en primera persona.

<sup>9</sup> Videojuego de disparos en primera persona. Compañía: id Software. 1993

<sup>10</sup> Rol Playing Gaming, literalmente «juego de rol»

<sup>11</sup> Aventura gráfica de terror y ciencia ficción en perspectiva isométrica creada por Christopher Bischoff en 2015

mejores formas de trabajar. Así pues, tras concebir los primeros proyectos como fracasos a nivel productivo, decidí embarcarme en un nuevo proyecto, ahora ya en solitario. La motivación por ver algo propio en movimiento y con la capacidad de ser jugable me animaba a trabajar en algo que fuera asumible, que no estuviera fuera de mis capacidades. Una vez al margen de las metodologías que se imponían al estar en un grupo de trabajo, se posibilitó trabajar más rápido y hacer algo más personal, evitando siempre plantearme metas inalcanzables.

### 3.OBJETIVOS

Como objetivos principales de este trabajo aparecen:

- Diseñar cada uno de los niveles con todos los modelos 3D requeridos por la escena, asumiendo las diferentes formas de modelado para el posterior renderizado.
- A pesar de estar en una fase temprana del desarrollo, realizar niveles jugables y que nos permitan avanzar con nuestro personaje, sin incidencias ni “bugs”.
- La creación de cada una de las texturas y sus diferentes mascarar para la creación de materiales.
- La inclusión de los scripts<sup>12</sup> que nos permitan avanzar a lo largo del juego. Ya sean para determinar el control del personaje, la posibilidad de coger objetos, interactuar con puertas, cambiar de niveles, muros invisibles o zonas de *respawn*<sup>13</sup>.
- Ambientar los niveles con el audio que les sea apropiado, ya sea mediante sonido ambiente, efectos o música.
- Realizar la Rúbrica necesaria que contemple las cuestiones que surgen en torno al proyecto y a su investigación de forma correcta, dando a entender que los conocimientos han sido aplicados de forma correcta

---

<sup>12</sup> En informática, un script es un conjunto de instrucciones. Permiten la automatización de tareas, creando pequeñas utilidades. Son ejecutados por un intérprete de línea de órdenes y usualmente son archivos de texto.

<sup>13</sup> Palabra inglesa que se refiere a la aparición de un jugador muerto en una localización concreta.

en dicho proyecto, sacando el máximo partido y rendimiento con el tiempo dado.

- El resultado de dicho proyecto determinaría la obtención del título, lo cual me permitiría trabajar de forma más pausada y buscar el acabado del videojuego de la mejor forma posible sin la problemática de cumplir una fecha de entrega. Dándole tiempo a desarrollarse fuera de un marco temporal y con la posibilidad de encontrar un equipo.
- Poder realizar o al menos plantear un juego completo de carácter RPG con apartado visual más personal e imaginativo.

## 4.METODOLOGÍA

Para realizar este proyecto he comenzado con una fase previa de investigación. Esta investigación abarcara diferentes aspectos para poder trabajar con el máximo de información posible. Primero sobre las empresas de videojuegos y más tarde sobre la creación de los mismos. Es importante comprender y aprender cada uno de los aspectos que rodean la producción: El modelado, la Texturización, la Animación, la Programación y el Sonido.

Después de la fase de investigación comienza una fase de pruebas donde se demuestra, en un pequeño nivel, la funcionalidad de lo aprendido y de cada uno de los elementos que crearemos para llevarlos a nuestro trabajo final. Una vez acabada esta fase se comprueba que la naturaleza del trabajo funciona correctamente haciendo uso de personas que probarán la demo y darán sus resultados.

Por último, además de la parte práctica del proyecto, se realiza su propia memoria. Analizaremos que los objetivos planteados se corresponden con lo conseguido en la parte práctica. Su redacción se realiza a mitad de la producción para tener una base práctica sobre la cual se pueda trabajar. Haciendo un análisis y organizando la información de todo el desarrollo, se procede a completar la memoria con todo el proceso, fuentes y las conclusiones obtenidas.

## 5. DESARROLLO DE VIDEOJUEGOS

Comprendemos como **desarrollo de videojuegos** el proceso de creación de un videojuego. En él se dan las fases por las cuales pasará hasta convertirse en un proyecto finalizado en su estado final. Su desarrollo abarca diferentes disciplinas, todas necesarias, para su elaboración completa, involucrando de este modo todo tipo de aspectos profesionales de los campos de la programación, la animación, el diseño gráfico, el modelado, el sonido, el doblaje, etc...

El desarrollo de un videojuego puede llevarse a cabo de distintas formas dependiendo del contexto en el que se plantee. De este modo podemos ver desarrolladores trabajando en solitario o con un gran negocio a sus espaldas. Realizamos ahora un recorrido por los diferentes contextos en lo que realizar este tipo de productos.

### 5.1. Compañías multinacionales

Los proyectos más costosos y de mayor magnitud suelen estar financiados por una distribuidora y normalmente abarcan todas las plataformas del mercado a la hora de sacar un producto. Estas grandes compañías cuentan con desarrolladoras con un gran número de trabajadores, llamadas *Third-party*, y se encargan generalmente de sacar los títulos más conocidos y los catalogados como "triple A"<sup>14</sup>.

Las compañías más conocidas serían:

- Microsoft Studios. (2002)
- Nintendo. (1889)
- Sony Computer Entertainment. (1993)
- Valve Corporation. (1996)

Por su nivel de reconocimiento y de ingresos a nivel mundial estas serían las *Third-party* más reconocidas actualmente: Electronic Arts, Bioware, Ubisoft,

---

<sup>14</sup> Los videojuegos triple A son aquellos desarrollados por grandes empresas de videojuegos y que cuentan con grandes presupuestos.

Blizzard, Rockstar Games, Bethesda Softworks, Activision, Konami, Capcom, Square Enix, Eidos Interactive, Take-Two Interactive, 2K Games, Namco Bandai, THQ, Sega.

## 5.2. Estudios independientes

Los estudios independientes comúnmente conocidos como estudios “Indie”, son estudios compuestos por un grupo reducido de personas. Su apoyo financiero no parte de grandes distribuidoras y actualmente muchos reciben su apoyo económico gracias a las donaciones recibidas en páginas de financiación colectiva como pueden ser *Kickstarter*<sup>15</sup> o *Greenlight*<sup>16</sup>. A pesar de ello, algunos estudios cuentan puntualmente con el apoyo económico de alguna empresa. Estos estudios al tener un presupuesto reducido en la mayor parte de los casos procuran distanciarse de los títulos “triple A” haciendo videojuegos más modestos con propuestas distintas a lo que el mercado actual mueve en el sector más comercial. Por lo tanto, los videojuegos de estas empresas son conocidos tanto por su innovación, como por su creatividad y la experimentación artística. Otro de los factores condicionantes del presupuesto es su duración. Por lo general estos títulos tienen una duración más corta a la habitual en los videojuegos, pero al no contar con la financiación de una gran empresa a sus espaldas suelen tener menos limitaciones creativas.

A pesar de no estar financiados por grandes compañías que respalden sus proyectos, los estudios independientes no tienen por qué estar formados por novatos dentro del ámbito de los videojuegos. Muchas de estas compañías han sido creadas por ex empleados de empresas más grandes, con el objetivo de crear sus propias obras de forma más libre.

Al tratarse de juegos producidos por grupos de personas independientes y en la actualidad, existir mayores facilidades a la hora de producir dichos juegos, el mercado ha sufrido un crecimiento exponencial. La gran mayoría son títulos que no obtienen ningún tipo de reconocimiento por la gran cantidad de juegos

---

<sup>15</sup> Sitio web de micromecenazgo para proyectos creativos. <https://www.kickstarter.com/>.

<sup>16</sup> Aprobación oficial de financiación para la producción de un videojuego en la plataforma Steam.



Fig. 1 Jonathan Blow: *Braid*, 2008.



Fig. 2 Scott Cawthon: *Five Nights at Freddy's*, 2014.

“basura” que se producen y la falta de una distribución comercial a gran escala y un marketing adecuado. A pesar de ello, algunos juegos independientes se han convertido en un gran éxito financiero, *World of Goo*<sup>17</sup>, *Super Meat Boy*<sup>18</sup>, *Minecraft*<sup>19</sup>, *Outlast*<sup>20</sup>.

### 5.3. Lobos solitarios

Como en todos los sectores profesionales, existen ciertas personas que, bien por comodidad, o por su situación personal, deciden trabajar en solitario. En el ámbito de los videojuegos también existe este tipo de posicionamientos y podemos encontrar desarrolladores que hacen sus creaciones completamente solos, sin ningún tipo de apoyo por parte de equipos de trabajo. Por lo general, estas personas, tienen ciertas nociones en las áreas de programación y diseño de videojuegos. No son muchos estos casos, pero siempre se encuentran algunos ejemplos que nos sorprenden con sus creaciones y tienen cierta repercusión. Entre ellos podemos encontrar a los creadores de videojuegos como *Braid*<sup>21</sup> (fig.1) o *Five Nights at Freddy's*<sup>22</sup> (fig.2). Estos videojuegos, por lo general, tienden a dar más importancia al concepto que da origen al videojuego que, a su aspecto técnico-visual, algo comprensible por la magnitud del trabajo a desarrollar.

A lo largo de los años que hemos dedicado a este campo, desde una y otra perspectiva, hemos podido ver como estos desarrolladores han adquirido importancia y muchos de sus primeros trabajos se han convertido en una puerta de una carrera fructífera. Gracias al beneficio económico producido por las ganancias de sus primeros juegos han podido producir nuevos videojuegos con mayor calidad.

<sup>17</sup> Videojuego de lógica enfocado a la física. Desarrollado por 2Dboy, 2008

<sup>18</sup> Videojuego de plataformas desarrollado por Team meet. Lanzamiento 2010

<sup>19</sup> Videojuego independiente de construcción y recolección. Desarrollado por Markus Persson. Lanzamiento en 2009.

<sup>20</sup> Videojuego de terror psicológico desarrollado y publicado por Red Barrels Games. 2014

<sup>21</sup> Juego que mezcla plataformas 2D con elementos puzle dentro de un sistema de juego que permite manipular el tiempo de la acción. Creación de Jonathan Blow, 2008.

<sup>22</sup> Videojuego de terror indie desarrollado por Scott Cawthon, 2014.

A su vez, la innovación y las ideas expuestas en estos títulos, han hecho que estudios de envergadura superior, se fijen en la capacidad de estos programadores, ofreciendo puestos en empresas mayores, financiándoles, o ayudándoles en la gestación de esos proyectos. Podemos ver casos como *Undertale*<sup>23</sup>, *Unravel*<sup>24</sup> o *The Witness*<sup>25</sup> en los que las distribuidoras han visto futuro a estos títulos y les han dado soporte para su lanzamiento.

## 6. REFERENTES

Los referentes en los cuales he basado el trabajo que presento, tanto en su aspecto visual como en lo relacionado con su jugabilidad, son varios. Al tratarse de una demo con niveles de diferentes estilos y ambientaciones, encontramos una amplia lista de referentes en función de su utilidad. En ellos aparecerán los diferentes videojuegos en los que me baso explicando sus características básicas y que partes han aportado algo a mi proyecto.

Primero hablaré sobre los referentes gráficos, y explicaré su influencia sobre el apartado visual, y como se ha realizado la propia edición de los niveles y los distintos elementos de detalle gracias a ellos.

Y en segundo lugar, hablaré de los que han influenciado el estilo de juego en cada uno de los niveles, junto al tipo de jugabilidad que tendrá nuestro *Player*.<sup>26</sup>

### 6.1. Referentes gráficos

Para poder hablar de los referentes gráficos que he utilizado hay que tener en cuenta varios factores.

El primero y más importante es la elección del motor gráfico sobre el cual se está trabajando, el software que nos proporciona un espacio de trabajo con capacidades de renderizado, detección de colisiones, sonidos, animación,

---

<sup>23</sup> *Aventura de rol con especial interés en la toma de decisiones. Desarrollador Tobyfox. 2015.*

<sup>24</sup> *Juego de aventuras perteneciente al género de las plataformas. Desarrollado por Coldwood. 2016*

<sup>25</sup> *Videojuego basado en rompecabezas diseñado por Thekla Inc. 2016*

<sup>26</sup> *Palabra inglesa para referirse al jugador*

inteligencia artificial, comunicación con la red para juegos multijugador, etc..., es básicamente la plataforma donde se sustenta el videojuego. Uno de los motivos para seleccionar un motor gráfico es su capacidad gráfica, la cual nos proporcionará una calidad visual determinada.

En la actualidad la cantidad de motores gráficos es amplia y a pesar, de que sus diferencias graficas están presentes, la evolución de estos motores y su afán por representar la realidad de forma más fidedigna hacen que estas diferencias entre ellos se acorten cada vez más. Hoy en día los motores más utilizados y que mejores resultados ofrecen son *Frostbite*<sup>27</sup>, *Unreal Engine*<sup>28</sup>, *Cry engine*<sup>29</sup>, *REDengine*<sup>30</sup>, *Anvil*<sup>31</sup> y *Unity5*<sup>32</sup>.

Para mi proyecto he escogido *Unity 5*. Este motor de juego me proporciona las herramientas adecuadas para desarrollar el proyecto con facilidad y obtener la imagen visual final deseada.

Los motivos por los cuales hago uso de este motor son: la sencilla interfaz, que te permite adaptarte rápidamente al espacio de trabajo, la posibilidad de exportar e importar archivos en varios formatos de forma sencilla y la cantidad de títulos desarrollados tanto a nivel profesional como independiente. La capacidad gráfica del mismo, que a pesar de que no es de los motores con mejor renderizado final, sí que proporciona una calidad gráfica más que aceptable. Por ese motivo muchos títulos en los que me he basado comparten motor gráfico. Entre los referentes más representativos que utilizan *Unity* aparecerían:

---

<sup>27</sup> *Frostbite* es el motor gráfico desarrollado y empleado por EA Digital Illusions CE. Con el paso de los años, DICE ha ido evolucionando el motor desde el *Frostbite 1.0* hasta el *Frostbite 4*.

<sup>28</sup> *Unreal Engine* es un motor de juego de PC y consolas creados por la compañía Epic Games.

<sup>29</sup> *CryEngine* es un motor de juego creado por la empresa alemana desarrolladora de software Crytek, originalmente un motor de demostración para la empresa Nvidia.

<sup>30</sup> *REDengine* es un motor de juego desarrollado de forma independiente por CD Projekt RED

<sup>31</sup> Motor de videojuegos creado en 2007 por la desarrolladora de videojuegos Ubisoft Montreal

<sup>32</sup> Motor de videojuego multiplataforma creado por Unity Technologies



Fig.3 Krillbite Studios: Among the Sleep, 2014



Fig.4 Campo Santo: Firewatch, 2016



Fig.5 Endnight Games: The Forest, Estado Alpha

- *Among the Sleep*: desarrollado por Krillbite Studios (29 de mayo de 2014) videojuego de perspectiva en primera persona. Nos muestra una aventura de terror por diversos entornos y criaturas surrealistas, donde se presentan obstáculos tanto físicos como mentales. Las facetas graficas de Unity en este videojuego me han aportado nociones de cómo inclinar ciertos efectos a mi favor a la hora de hacer el videojuego. El uso de efectos de cámara como: *Bloom*<sup>33</sup>, *ambient occlusion*<sup>34</sup>, *antialiasing*<sup>35</sup>, para que la visualización final de la demo sea de mayor calidad. WEB: <http://www.krillbite.com/ats/>.
- *Firewatch*: Desarrollado por Campo Santo estudio independiente (9 de febrero de 2016) videojuego de aventura y exploración. Muchos lo catalogan dentro del género "walking simulator"<sup>36</sup>. Este juego trabaja la imagen como si de un dibujo animado se tratase. La sencillez a la hora de construir espacios me ha servido para plantear la creación de los propios procurando una mejor optimización en las mallas geométricas de los objetos. WEB: <http://www.firewatchgame.com/>.
- *The forest*: desarrollado por Endnight Games estudio independiente (lanzamiento por determinar, previsiblemente en 2016) Survival-Horror de mundo abierto. La creación y edición de vegetación, como el uso de shaders en el terreno han servido como referencia para hacer los propios. Las texturas del terreno y los modelos como piedras, arbustos y otros elementos naturales han servido de ejemplo para realizar los propios, ya sea creando mallas y trabajándolas con materiales ricos en efectos o haciendo uso de efectos gráficos como el

<sup>33</sup> Efecto gráfico que afecta a las zonas iluminadas creando la ilusión de que el foco es más luminoso de lo que es en realidad.

<sup>34</sup> Es un método de sombreado o shading 3D que se utiliza en gráficos computarizados que ayuda a añadir realismo a la reflexión de los modelos teniendo en cuenta la atenuación de la luz, debido a la oclusión ambiental.

<sup>35</sup> Efecto gráfico para el suavizado de contornos.

<sup>36</sup> Genero de videojuego en el que predomina la exploración.



Fig.6 Nvyve: P.A.M.E.L.A., Fecha de lanzamiento por determinar

teselado<sup>37</sup>. Al tratarse de un videojuego ambientado casi exclusivamente en un entorno natural ha servido también para fijar las bases de cómo deben ser utilizadas las texturas en el suelo y como tienen que ser empleados ciertos modelos para darle cierta naturalidad al nivel. WEB: <http://survivetheforest.com/>.

- *P.A.M.E.L.A.*: Desarrollador por Nvyve estudio independiente (lanzamiento por determinar, previsiblemente en 2016) Se trata de un FPS de supervivencia ambientado en un mundo futuro. Este videojuego me ha servido para ver las capacidades técnicas a la hora de trabajar en las texturas de interiores, como su utilización de iluminación dinámica<sup>38</sup> en las mismas. WEB: <http://www.pamelagame.com/>.

A pesar de que hay más ejemplos, estos serían los principales referentes que harían uso del mismo motor gráfico Unity. Fuera de este marco nos encontramos con juegos que harían uso de otros motores gráficos ya nombrados. Muchos de ellos han servido para conocer de qué forma se han realizado ciertos detalles dentro de estudios con una larga trayectoria en el tiempo y en proyectos. Hacer mención a videojuegos que han servido de referente a la hora de crear modelos 3D, utilizándolos como ejemplo a la hora de crear “props”<sup>39</sup> como la vegetación, las rocas, el mobiliario, etc.... Entre estos entraría *Witcher 3*<sup>40</sup>, *Crysis 2*<sup>41</sup>, y por último *Saga Mass Effect*<sup>42</sup>.

## 6.2. Referentes de jugabilidad

Para poder hablar de los referentes de jugabilidad primero hay que comprender este término en el ámbito de los videojuegos. La jugabilidad es un término que nace a raíz de los videojuegos y es un elemento primordial para su

<sup>37</sup> Efecto gráfico que crea texturas planas en texturas tridimensionales.

<sup>38</sup> Renderizado en tiempo real de la iluminación.

<sup>39</sup> Utilería empleada en cualquier producción audiovisual.

<sup>40</sup> 2015. Videojuego de rol de mundo abierto.

<sup>41</sup> 2011. Secuela “FPS” con cortes de mundo abierto y exploración, con un toque de rol para la personalización de nuestro personaje

<sup>42</sup> 2008-2012. juego de rol de ciencia-ficción futurista. Controlado en tercera persona con navegación limitada por los distintos niveles.

desarrollo y posterior valoración. Es empleado tanto en el análisis como en el diseño de un juego y nos describe la calidad del mismo con su funcionamiento. Hace referencia a las experiencias basadas en la interacción del jugador con el sistema. En pocas palabras, sería aquello que hace el jugador. El análisis sobre lo que es la jugabilidad hace que entendamos su complejidad y para comprender su relación con el videojuego, es necesario dividir el término en diversos atributos.

- Jugabilidad Intrínseca: Es la propia capacidad del jugador de relacionarse con el entorno virtual, está ligada con el Gameplay<sup>43</sup>.
- Jugabilidad Mecánica: Es la jugabilidad directamente ligada con la capacidad del software. En ella podemos ver la fluidez entre escenas, iluminación, comportamientos de los personajes y entorno, sonido, etc...
- Jugabilidad Interactiva: Como su propio nombre indica, es la capacidad de interacción del jugador con los sistemas de control como podrían ser, diálogos, interfaz, manipulación de objetos, etc...
- Jugabilidad Artística: En ella vemos elementos como la calidad artística tanto visual, sonora, literaria y narrativa.
- Jugabilidad Perspectiva: Es la jugabilidad relacionada directamente con los sentimientos que le producen el juego al jugador. Es de un alto valor subjetivo ya que nos habla de la inmersión con el mundo virtual.
- Jugabilidad Interpersonal: Esta solo hace acto de presencia cuando nos encontramos con un videojuego multijugador. Muestra las sensaciones relacionadas con el juego en grupo ya sea cooperativo, competitivo, o colaborativo.<sup>44</sup>

Una vez claros estos conceptos mencionaré los videojuegos que han influido en la jugabilidad de la demo. En cuanto a los referentes, hay que tener en cuenta el género en el cual está basado este proyecto o demo "*Walking Simulator*". Hablamos de los videojuegos en los que la experiencia de juego

---

<sup>43</sup> Conjunto de acciones que puede realizar un jugador para interactuar con el juego.

<sup>44</sup> Stanford Ontology Library Video game's Elements Ontology: A video game's elements ontology by González Sánchez, J. L. and Gutiérrez Vela, F. L. University of Granada, Spain.

está marcada por la exploración del entorno. Este género ha ido creciendo a lo largo de los años gracias a la aparición de un mayor número de desarrolladores independientes, con lo cual la mayoría de estos títulos no son considerados “triple A”, ya que son juegos en los que la jugabilidad no se basa en sus mecánicas. Lo que predominan son las experiencias del viaje por encima de las de mecánicas de juego. Si hacemos un listado de los videojuegos más influyentes para realizar mi proyecto aparecerían:

- *Super Mario*. Desarrollado por Nintendo (1985) videojuego de plataformas en los que nuestra habilidad nos permitirá llegar al final del recorrido antes de que se nos acabe nuestro tiempo o nuestra vida.

Este último aparece por dos motivos diferentes. El primero, sería su faceta jugable. Este videojuego se basa en sobrepasar unos obstáculos mediante la habilidad dentro del juego (plataformas). A pesar de que la demo no se centra en este aspecto, también aparecen niveles en los que será necesario tener cierto control del personaje para pasar a la siguiente fase. Y el segundo motivo es el tipo de juego. No pretende ser una experiencia que nos abrume con sus mecánicas de juego, tiene unos controles sencillos y a pesar de tener una historia de fondo, el objetivo está claro. Llegar al último nivel del juego.

- *Journey* desarrollado por Thatgamecompany (2012) experiencia online donde los jugadores se sitúan en un mundo desconocido, en el que tendrán que explorar, relacionarse e interactuar.

Este juego se convierte en una referencia por su utilización de los recursos gráficos, la música y el paisaje.

Web: <http://thatgamecompany.com/games/journey/>.

- *Dear Esther* desarrollado por Dan Pinchbeck y Robert Briscoe (2012) nos hace investigar una isla abandonada, en donde seremos nosotros los que dirigiremos la historia de forma libre por el mapa, con una voz en off, que narrará los hechos como si de una novela se tratase. A pesar de que nuestra demo es una pequeña parte de lo que sería el juego final que concebimos, está orientada a conseguir lo que



Fig.7 Thatgamecompany: Journey, 2012



Fig.8 Dan Pinchbeck and Robert Briscoe: Dear Esther, 2012



Fig.9 Jonathan Blow: *The Witness*, 2016



Fig.10 The Chinese Room: *Everybody's gone to the rapture*, 2016

consigue este juego mediante elementos narrativos. WEB:

<http://dear-esther.com/>.

- *The Witness* (2016) Un juego de rompecabezas en el cual, atrapados en una isla misteriosa, deberemos resolver los acertijos para seguir avanzando.

Este juego lo descubrí a mitad del desarrollo de la parte práctica. A pesar de que la idea de ambos es muy similar, ambos basados en obstáculos mentales, me sirvió para perfeccionar lo hecho y cambiar el enfoque de algunos niveles.

WEB: <http://the-witness.net/news/>.

- *Everybody's gone to the rapture* (2016) de los creadores de *Dear Esther*, han creado una aventura narrativa, ambientada en un pueblo aparentemente abandonado tras una catástrofe. Predomina el dominio de la imagen con bellas estampas y fuerza en la banda sonora. Todos estos factores incitan al jugador a querer explorar todos los recovecos del mapeado. Al igual que *Dear Esther* este juego me interesa por su capacidad de mantener el hilo narrativo a lo largo del juego. WEB: <http://www.thechineseroom.co.uk/games/everybodys-gone-to-the-rapture/>.

## 7.PRODUCCIÓN

### 7.1. CONTEXTO

Para empezar a trabajar en un videojuego hay que tener en cuenta en cuantos ámbitos y en cuantos sectores se divide una empresa de videojuegos, y en qué orden hay que empezar a desarrollarlo. Al tratarse de un trabajo independiente en el que no se va a mostrar un solo nivel con una única ambientación, si no varios niveles dentro de un futuro videojuego, conviene tener claros los conceptos de producción y como se estructura el trabajo en una empresa de videojuegos.

Desde el principio de la historia de los videojuegos, las cosas han ido cambiando. Antes se encargaba una sola persona de la historia, las imágenes,

la programación, la música, la animación... todo lo que el juego era en sí mismo. Podríamos decir, que el desarrollo de este proyecto, es parecido a lo que en un principio era esta industria, pero con un matiz diferente. El grado de complejidad ha cambiado en función de lo que queramos. Si nos mantuviéramos en los videojuegos de 8 bits, podríamos realizar de forma más cómoda y parecida lo que antes se hacía.

Pero si salimos de ese marco de los 8 bits, entramos generalmente en lo que viene a ser el universo 3D y, por consiguiente, el mundo de los videojuegos actuales. Las cosas cambian un poco, aparecen los denominados desarrolladores, que forman parte de una empresa de videojuegos y se estructuran en diferentes sectores dependiendo de sus cualidades, modelador líder, director de arte, modelador, texturizador y otros.

Actualmente, se comenzaría por una idea de lo que se quiere crear para nuestro videojuego y se llevaría al director de arte, quien dirige los diseños del artista líder que a su vez se lo pasaría al artista conceptual. Este lo pasaría a modeladores, texturizadores, diseñadores de interfaces, animadores y guionistas. De esta forma se crearía una estructura de trabajo ramificada partiendo del jefe de grupo. Esto no significa que solo trabajen los de abajo, sino que el artista conceptual o el director de arte también pueden tener su trabajo como modelador, texturizador o cualquier otra faceta relacionada con su especialidad. Otra de las funciones del director de arte sería el dirigir el trabajo de los creadores de niveles, los cuales utilizarían los elementos creados por estos últimos, modelador y texturizador. Esto no quiere decir que todos estén bajo la batuta del director de arte, al final todo se retroalimenta del trabajo de todos. De esta forma existe un flujo de trabajo en el que todos tienen material de todos para ejercer su función dentro de la empresa como es mostrado en la imagen siguiente (fig.11).

Por la parte de la programación, estaría de responsable el programador líder que dirigiría al resto de programadores, los cuales, a su vez, estarían en conexión con los artistas, para hacer funcionar lo que estos crean necesario, y llevarlo al juego. Ya fuera el movimiento del personaje, la animación de

objetos, la destrucción de escenario, los efectos de partículas, la interacción con los modelos, etc... de esta forma se convierten en una pieza esencial para conseguir que todo funcione.

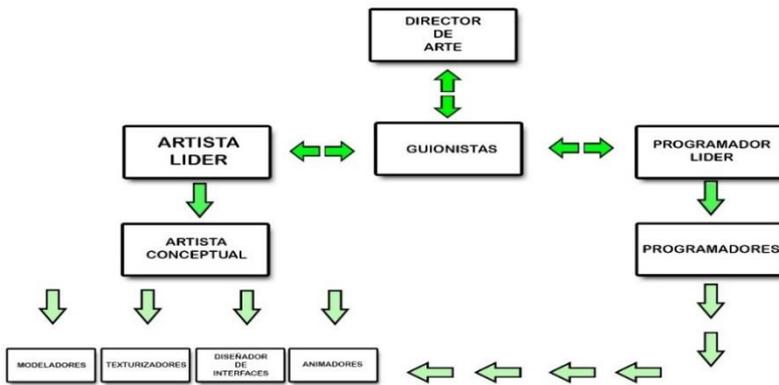


Fig.11 Organigrama de empresa

Una vez que cada miembro tiene asignada su función comenzaría el desarrollo del videojuego que queremos hacer. Hay que transformarlo desde la idea hasta llevarlo a la pantalla.

Lo primero es crear un espacio donde desarrollarse, con lo cual necesita de un diseño y la creación de un entorno virtual ya

que es una de las partes esenciales del juego y es lo que vamos a estar observando en todo momento, a no ser que nuestro juego presente un menú de inventario o cualquier otra interfaz que se solape con nuestra visión del entorno.

Dentro de éste, será donde se lleven a cabo la mayor parte de nuestras acciones, por lo tanto la ambientación y el espacio tienen que estar bien relacionados entre sí ya que será donde se realicen las pruebas y los desafíos. Esto presenta dos trabajos para el desarrollador, crear esa ambientación y ese mundo 3d por el que nos movemos y gestionar el desafío que en él aparecerá. Una vez construidos los elementos de juego básicos que componen la escena, comienza el momento del detalle para nuestro escenario. Dotarlo de más riqueza y realismo. De esta forma nos aseguramos de que la escena funcionará y posteriormente será dotada de una atmósfera única.

Todo esto a lo largo del tiempo se ha ido perfeccionando. Y nuestro desarrollador de niveles empieza a tener un trabajo cada vez más costoso y detallado. Gracias a la evolución de la tecnología, con la aparición de procesadores más potentes, mejores tarjetas gráficas y mayor capacidad de almacenamiento, los propios escenarios han evolucionado, ya que tienen mayor margen de edición. Los escenarios ahora tienden a ser más abiertos y

con mayor grado de detalle. Los efectos de shaders<sup>45</sup>, partículas, texturas, iluminación, etc... han sido mejorados de forma que la verosimilitud de los escenarios es mayor, otorgándoles personalidad única o simplemente acercándose a la representación de la propia realidad de forma más creíble. De esta forma, el aprovechamiento de estos recursos se convierte no solo en simple ambientación, sino en parte del juego.

A lo largo de la creación de estos escenarios, que para el jugador son comprendidos como niveles, las cosas han cambiado con el paso de los años, ya que la industria del videojuego ha evolucionado y han aparecido nuevos géneros creando lugares de juego con grandes extensiones para la exploración. Formando parte de un nivel mayor a gran escala y diferenciando las zonas de juego de otra forma, sin la utilización de niveles independientes como lo solían hacer a mediados de 1970. Muchos de los títulos actuales han diluido las diferencias entre niveles, de forma que todo es parte de un mismo conjunto, haciendo que las transiciones entre ellos sean casi inexistentes. A pesar de ello seguimos viendo videojuegos actuales que mantienen la esencia de los primeros juegos, en los que cada nivel era diferente, y seguían una línea de progresión marcada por la dificultad. De estos últimos, he tomado ejemplo para realizar mis propios niveles, en los que cada uno se transforma en una nueva misión con dificultad propia, cuya superación nos permite avanzar hasta la siguiente.

Otro factor a tener en cuenta es la evolución que han tenido los niveles con la entrada del 3D, permitiendo al jugador la navegación en cualquiera de sus ejes. Gracias a ello el marco de posibilidades se amplía, pero a su vez provoca que el diseñador tenga que ejercer mejor su papel ya que el jugador es libre de andar por este territorio, y en su función como diseñador tiene que controlar todos y cada uno de los espacios de este mundo tridimensional.

### **7.1.1 WORKFLOW**

El workflow (flujo de trabajo) corresponde con los aspectos operacionales de una tarea, trabajo o actividad. En él se estructurarán las tareas, de forma que

---

<sup>45</sup> Un procedimiento de sombreado e iluminación que permite al artista/programador especificar el renderizado de un vertex o de un pixel. Los shaders son utilizados para realizar transformaciones y crear efectos especiales, como por ejemplo iluminación, fuego o niebla.

se automaticen las acciones, tareas o actividades relacionadas con el proceso de creación teniendo en cuenta las dificultades y etapas por las que se pasará.

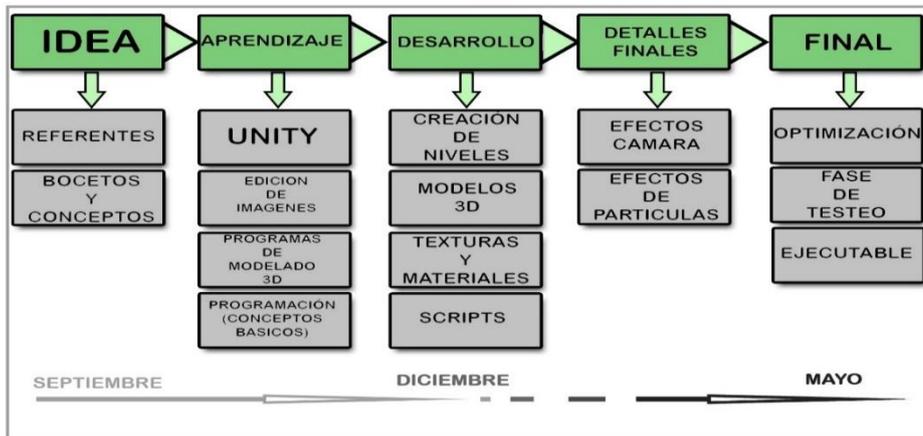
Las primeras dificultades a la hora de trabajar aparecen cuando las labores de trabajo recaen sobre una única persona. El aprendizaje, presente en cada uno de los aspectos, requiere un consumo de tiempo a la hora de realizar las tareas. Por este motivo es creado un esquema de fases (fig.12), en el cual aparecerán las diferentes fases de trabajo comenzando por una idea y continuando por el aprendizaje de que será necesario.

Teniendo claro todo esto, hay que gestionar el tiempo y las partes que son esenciales. Al ser una fase Alfa, la más temprana en el desarrollo, nos es necesario llevar el trabajo a un estado final, sino crear una demo que muestre la idea y el concepto de lo que más tarde será el videojuego desarrollado. En ella veremos los diferentes niveles por los que estará compuesta la obra.

De acuerdo con las necesidades que aparecen en este largo desarrollo, se hace necesaria una búsqueda de información para profundizar en los métodos más adecuados en la creación de videojuegos, ya sea creación 3D, optimización, conocimiento de shaders que aumenten la calidad gráfica, gestión, tanto de las herramientas del motor gráfico, como de los diferentes programas de diseño 3D, creación de texturas, aprendizaje en los ámbitos de la animación y la programación, etc...

Mediante un curso intensivo de captura de información de diseño 3D, animación, creación de videojuegos (Niveles, LOD, Shaders, texturas, materiales, animación, optimización, etc...) conocimientos de programación tanto en tutoriales oficiales como no oficiales, se conforma una base teórico-práctica sobre cómo desarrollar el videojuego. Una vez claros estos conocimientos, podremos volver a consultarlos y confrontarlos ante futuras dudas. Se desarrolla un esquema de trabajo en el cual se divide el tiempo

desde el inicio, hasta la fecha final y se determina el tiempo necesario para la realización de cada uno de los niveles.



Una vez determinados los procedimientos de desarrollo fijados en el esquema de fases, se inicia la búsqueda de referentes relacionados con la propuesta e información sobre la creación de videojuegos que sea afín a los gustos propios y que convengan al diseño de los

Fig.12 Esquema de fases

niveles concebidos.

Se crea un listado de los posibles niveles futuros. Se desechan los peores o los que por gestión de tiempo y grado de dificultad no podrían ser realizados, junto con los que dificultarían la realización de otros o ralentizarían el trabajo general.

A la hora de seleccionar los finales, se tiene en cuenta la variedad entre ellos para que el jugador viaje por diferentes lugares y no aparezca una sensación de monotonía. De ese modo se escogen los mapas en función de su jugabilidad, su ambientación y su dificultad.

Una parte importante, antes de comenzar con el trabajo sobre el motor gráfico, es el orden que deberán tener estos niveles. Como previamente se ha mencionado, basándome en su ambientación y en su jugabilidad intercalé mapas de carácter natural con los niveles más imaginativos, para no sobrecargar la sensación de juego y buscar puntos de descanso en los mapas naturales. A su vez, estos mapas van ordenados por el tipo de jugabilidad y dificultad. Primero iría un nivel interior que marca el inicio adentrándonos en los mapas de exploración, continuarían los mapas de laberintos, seguirían los de plataformas y para cerrar la demo, finalizaría en otro interior (Fig. 13). Otro de los motivos que marca este orden es la dificultad. Han sido construidos de forma que la dificultad tenga una curva ascendente, de este modo los retos

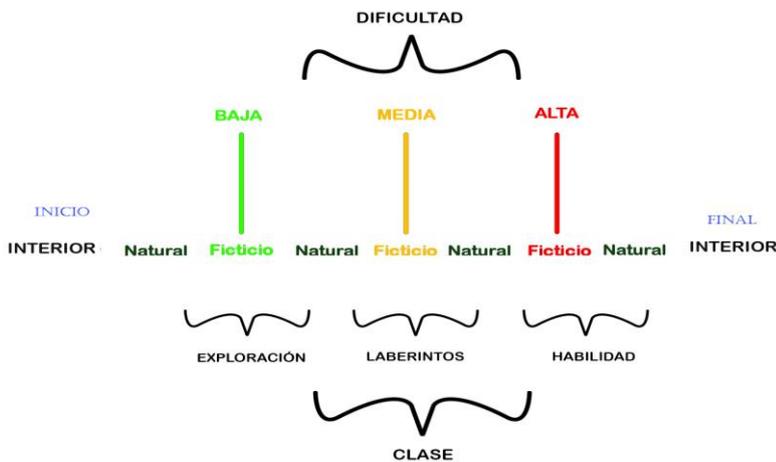


Fig.13 Fases de juego

que se nos plantean harán que nos habituemos al control y nos centremos cada vez más para llegar hasta el final.

Teniendo claros todos los conceptos, y sabiendo cual es el trabajo a realizar, se comienza a trabajar cada uno de los niveles respetando su orden, pero

preocupándonos únicamente por la funcionalidad del mismo. Primero se trabaja en un nivel de bloques con modelos simples para comprobar la funcionalidad del nivel. Una vez acabada esa fase se corrigen los posibles errores y se comienza a trabajar el detalle sustituyendo los modelos simples con los diferentes elementos creados para la escena. Crearemos los diferentes *assets*<sup>46</sup> que utilizaremos para llenar los entornos, y darles algo más de riqueza, pero sin olvidarnos de que es un juego en estado alfa, no es necesario un grado de detalle elevado. El desarrollo de los niveles será explicado más adelante y con mayor detalle de forma individual.

Por último, se realiza un testeo a nivel jugable sin prestar atención a los pequeños errores que no entorpecen la experiencia jugable y se genera el ejecutable para su posterior disfrute.

### 7.1.2. Programas utilizados

Durante la fase de trabajo todos los programas informáticos se utilizan casi al mismo tiempo. A pesar de ello, sí que hay un cierto orden a la hora de su empleo, que viene marcado por la propia necesidad del nivel. Primero se utilizaría el propio motor gráfico, en él crearemos la escena disponiendo el espacio requerido, después se abordaría la segunda fase de trabajo con los programas de modelado, que crean las piezas geométricas necesarias para conformar la escena, luego, los programas de edición de texturas, para

<sup>46</sup> Cada uno de los elementos que componen el juego (animaciones, modelos, IA, sonidos, etc.). Reutilizables en diferentes niveles o escenas.

posteriormente editar los materiales en el motor gráfico. Una vez creado todo el nivel, utilizaríamos el motor gráfico para la edición de efectos y de sonido.

A la hora de realizar un videojuego es necesario apoyarse en programas que permitan el desarrollo de forma fiable y que posean las características adecuadas para la exportación o importación de lo que hacemos entre ellos, sin que nos causen problemas o nos consuman innecesariamente el tiempo que necesitamos para dicho desarrollo. Hoy en día, hay ya gran número de programas que están adaptados para la creación de videojuegos, con muchas herramientas a nuestra disposición, y están perfectamente preparados para ser compatibles con los motores gráficos que existen destinados a la producción de un videojuego. A la hora de elegir con cuales trabajar, es importante conocerlos bien, y saber qué características tienen, así como también que coste económico nos van a suponer. A pesar de que hay muchos programas que son propietarios, con un coste, bien mensual o bien por el pack completo, podemos encontrar programas como *Blender* que tienen gran cantidad de herramientas y son gratuitos.

Los programas que aparecen a continuación son los que yo he empleado para la realización de este proyecto y será explicada su función específica.

-*Blender* (versión 2.77.a): Programa dedicado a la edición 3D con posibilidades en el modelado, renderizado y animación. Posee herramientas de modelado escultórico, incluyendo topología dinámica y la técnica procesal de nodos. También se ha de indicar que Blender posee su propio motor para videojuegos.

Este programa ha sido utilizado para el modelado de los elementos de juego principalmente para los más geométricos o menos orgánicos. Se han creado tanto modelos *highpoly*<sup>47</sup> como *lowpoly*<sup>48</sup>, al igual que los modelos necesarios para el proceso de *LOD*<sup>49</sup>. También se ha realizado

---

<sup>47</sup> Modelo de alta poligonización.

<sup>48</sup> Modelo de baja poligonización.

<sup>49</sup> (*Load of distance*) Proceso de carga de objetos por distancia. Sustituye modelos de alta poligonización por otro de baja o lo elimina en función de la distancia con el jugador. Se utiliza para aligerar la carga del sistema.

en la extracción de *UV's* para el mapeado, como la creación de *distintos tipos de máscaras*. Y por último se ha hecho uso de la retopología<sup>50</sup> de forma manual en algunos elementos. Actualmente es un programa gratuito y esta conducido por un equipo de voluntarios procedentes de diversas partes del mundo unidos por un interés de promover un sistema completamente libre y abierto de creación 3D. WEB: <https://www.blender.org/>.

-*ZBrush* (versión 4r6): Software de modelado 3D, escultura y pintura digital. Este programa ha sido utilizado en paralelo con Blender para los modelos de mayor detalle en su versión *highpoly*, también se han utilizado sus herramientas de *Unwrap*<sup>51</sup> para la extracción de *UV's*<sup>52</sup> y *Normals*. Principalmente se ha utilizado en los modelos más orgánicos y que requieren de mayor detalle al estar mejor optimizado y ofrecer la posibilidad de crear modelos más complejos. También se ha utilizado para la retopología de algunos elementos su herramienta *Zremesh*<sup>53</sup>. Zbrush pertenece a Pixologic, empresa Californiana, con oficinas en Los Ángeles y el área de la bahía de San Francisco. ZBrush tiene un coste de \$ 795. WEB: <https://pixologic.com/>.

-*Unity 5*: Plataforma para desarrollar videojuegos y experiencias interactivas 2D/3D que permite el soporte multiplataforma. Ha sido el motor gráfico en el que se ha trabajado y compilado toda la Demo. En él se han importado todos los modelos realizados por los programas anteriores. También se han creado los Scripts (lenguaje Javascript) necesarios para el desarrollo de la demo y la creación de todos los materiales de los modelos junto con sus "*Colliders*"<sup>54</sup>.

---

<sup>50</sup> Proceso de disminución de polígonos sobre un modelo 3D.

<sup>51</sup> Despliegue de la malla del modelo, sobre un plano 2D.

<sup>52</sup> El mapeado resultante de la herramienta Unwrap

<sup>53</sup> Herramienta que permite la reorganización de polígonos en un modelo 3D

<sup>54</sup> Elemento de juego que permite colisión mediante cálculo de físicas.

Está siendo desarrollado por Unity Technologies y actualmente es un programa con versiones de licencia gratuita. WEB:  
<https://unity3d.com/es/5>.

-*Speedtree modeler for Unity 5*: Paquete de programación y modelado de vegetación, que genera follaje virtual para animaciones en tiempo real. Dedicado a la creación de videojuegos y simulaciones. Como el propio programa específica, se ha hecho uso de él para todo el apartado relacionado con la vegetación. Producto desarrollado y comercializado por Interactive Data Visualization, Inc. Las licencias son distintas y a distintos precios en función de lo que busquemos. WEB:  
<http://www.speedtree.com/>.

-*NDO y DDO de Quixel*: Herramienta de *normal mapping* que convierte Photoshop en un conjunto de herramientas para editar detalles de superficie. Cada herramienta puede ser utilizada para esculpir normales en tiempo real o extraerlos de fotografías al instante. Programa de pago desarrollado por Quixel y su comunidad de usuarios. WEB:  
<http://quixel.se/>.

-*Adobe photoshop CS6*: Aplicación de edición de imágenes y diseño tanto para fotografías, diseños 3D, video, ilustraciones, etc...  
Se ha hecho uso de este programa para la creación de texturas, ya sean a mano o utilizando fotografías como referencia. Desarrollado por Adobe Systems Incorporated. Programa de pago. WEB:  
<http://www.adobe.com/es/products/photoshop.html>.

-*Adobe audition CS6*: Programa que ofrece herramientas para conseguir un alto nivel de rendimiento en edición, mezcla, restauración y efectos de sonido. Como el propio programa indica, se ha utilizado para todo el apartado sonoro, ya sea para efectos de sonido, ambientales o música. Desarrollado por Adobe Systems Incorporated. WEB:  
<http://www.adobe.com/es/products/audition.html>.

### 7.3. ELEMENTOS UTILIZADOS

Dentro del desarrollo de niveles se realizan múltiples tareas. Entre ellas está el confeccionar una biblioteca de recursos para el videojuego, allí encontraremos todos los *Assets* o *Props*. Son los elementos que componen el juego, desde una silla a una piedra. Cada uno de estos elementos contara con sus propias texturas y materiales.

A lo largo de los siguientes apartados explicare como se ha realizado cada uno de los procesos para la creación de estos elementos.

#### 7.3.1. *Props*

Como ya se ha explicado más arriba los *Props* son los elementos que conforman el videojuego, objetos de utilería o attrezzo. Para su creación se han utilizado diferentes programas según su naturaleza.

*Blender* y *Zbrush* han sido los principales programas de creación de objetos y ambos comparten una forma de trabajar similar, diferenciándose por algunos detalles, como puede ser la optimización en el uso de modelos con gran cantidad de polígonos, técnicas de aplicar texturas y otros elementos a los que no les daremos más importancia. A pesar de estas diferencias, la forma de trabajar con los modelos ha sido muy similar.

Para la mayor parte de los *Props* se ha seguido la misma metodología, por ello se hará una explicación general sobre la creación y edición, sin explicar cada uno de los elementos que conforman el juego. De esta forma explicare detalladamente los procesos por los cuales pasa un modelo durante la creación de su forma física hasta su modificación a través de los distintos mapas que le otorgan detalle.

El primer paso a la hora de crear un modelo es analizarlo y tener en cuenta que clase de objeto es. Si son objetos secundarios y que pasan desapercibidos, su número de polígonos será inferior. Por el contrario, si contamos con que es un objeto importante en la escena contara con mayor número de los mismos.



Fig.14 Proceso de retopología

Una regla que he establecido a la hora de marcar esta pauta, es tener en cuenta cual es el objeto más importante en un videojuego, el personaje principal. Actualmente estamos viendo personajes que alcanzan los 45.000 polígonos en videojuegos comerciales. Por ese motivo la creación de los objetos de attrezzo no debería de superar los 1.000. En el caso de los árboles, al contar con la vegetación, es distinto. A pesar de esta pauta, el número de polígonos puede oscilar por encima o por debajo de esos números dependiendo del objeto y la cantidad de los mismos en la misma escena.

Ahora que está claro este principio, se crea un modelo de alta polinización con todo el detalle que se requiera. Mediante un proceso de retopología<sup>55</sup> (Fig.14) comenzaría la creación del modelo, en una versión de baja poligonización (*lowpoly*).

El siguiente paso lo constituye realizar un despliegue de la malla del modelo, sobre un plano 2D, es lo que conocemos como generar un mapeado UV, estas siglas se utilizan para definir el ancho y alto de las coordenadas de una imagen y no confundirlas con XYZ de las coordenadas de la geometría 3D. Este desplegado de la superficie, nos servirá como guía para posteriormente colocar nuestras texturas sobre el modelo. El proceso se realiza con la herramienta *Unwrap* que realizará esta extracción y relacionará nuestro mapeado con la posición espacial de nuestro modelo. Es importante tener claro cómo se va a realizar, se puede hacer de forma automática o manual. De la forma automática, dependiendo del modelo, puede ocasionarnos problemas. Si *Blender* no consigue el mapeado deseado, se puede optar por importar el modelo a *Zbrush* y realizarlo allí, ya que este programa hace muy bien esa función. De todas maneras, yo me decanto por *Blender*, ya que me permite de forma manual una fácil edición del mapeado y crearlo de la manera que más me convenga para la posterior colocación de texturas (Fig.15).

En el siguiente paso vamos a realizar los detalles del objeto. Esto se puede realizar en *Blender* y *Zbrush* gracias a su herramienta de modelado *Sculpt*. En

<sup>55</sup> Proceso que pasa un modelo 3D. Reestructura su maya con el fin de obtener una copia con menor número de polígonos pero con un aspecto visual similar al original.



Fig.15 UV's, Textura aplica sobre coordenadas Uv, modelo 3D.

los casos de mayor detalle es preferible pasar a *Zbrush* por su mejor optimización a la hora de trabajar con un mayor número de polígonos. En la mayoría de casos yo he escogido *Blender* ya que el modelo suele encontrarse dentro de este programa. Podemos realizar una copia y dejar en invisible el original, de esa forma será más rápido trabajar a la hora de extraer los detalles. Otro de los motivos por los que elijo este programa es su herramienta de retopología dinámica (*Dyntopo*) que facilita la labor en el esculpido pudiendo alterar el número de polígonos en la malla de forma contenida y localizada y en función de las necesidades del modelador. (Fig.16)

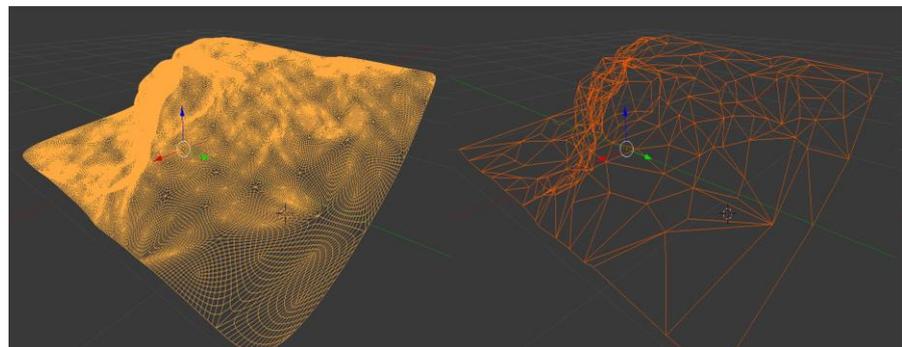


Fig.16. Vistas Wireframe de Highpoly y Lowpoly

Para realizar estos detalles, es necesario aumentar considerablemente el número de polígonos, haciendo subdivisiones de la malla. En mi caso, el número de subdivisiones varía según el software utilizado, y depende de la fluidez en el modelado que necesite. Si tomamos el mismo modelo y pretendiendo una fluidez similar, en *Blender* rondarían los 800.000 polígonos y en *Zbrush* 1.200.000. Esta fluidez también está determinada por el ordenador que poseamos y hay que tener en cuenta que, aunque un software de edición lo soporte bien, debemos pensar en su funcionamiento posterior en *Unity*. En estos *highpoly* (modelos de alta poligonización) realizaremos los detalles utilizando las brochas que necesitemos, dentro de la herramienta de esculpir. Una vez resueltos los modelos los remataríamos aplicándoles la textura conveniente.

### 7.3.2. Texturas

Las texturas en sí mismas no son más que un conjunto de píxeles de colores situados en un determinado lugar, en este caso un polígono. Estas otorgan propiedades a los materiales de los objetos. Por ese motivo, no solo otorgan la apariencia externa al material, sino que le otorgan características como pueden ser, rugosidad, brillo, transparencia, suavidad, etc...

A lo largo del desarrollo se han ido creando nuevas texturas en función de los nuevos materiales. No en todos los casos ha sido igual. En los modelos de mayor envergadura, se necesita no solo la textura base que le da un primer aspecto, sino que también necesita texturas para detalle, texturas que se solapan con la principal. Por ejemplo, si necesitaba un material de madera, iba a necesitar su respectiva textura de madera, junto con su *Normal*, su *AmbientOcclusion*, *Especular*, etc... Todas estas imágenes conformarán el resultado final del material.

Para la creación se han seguido varios procesos dependiendo del modelo sobre el cual van a ir colocadas y de qué tipo de textura estamos hablando.

Primero iría la **textura base**, esta textura es la que define el material. Si es una silla de madera, esta textura tendrá que ser una imagen de madera, si es una plancha de hierro, la textura base será una imagen de hierro. El concepto es bastante simple cuando hablamos de texturas que existen en la realidad y que conforman todo el modelo 3D, pero cuando queremos una textura en la que aparecen distintos materiales, porque nuestro modelo lo requiere, entonces tendremos que editar la imagen para que se acople a nuestra UV's.

En función del mapeado creado previamente tendremos que pintar las texturas en la posición correcta. Para hacer esto tenemos varias formas:

- Programas de edición de imagen: extrayendo las coordenadas UV podemos modificar las texturas editando la imagen. De esta forma sabemos que parte de nuestro modelo será coloreada. Una vez puesta se aplicará respetando las coordenadas. En la siguiente imagen, vemos a la izquierda las coordenadas UV del modelo de la derecha.

Fig.19

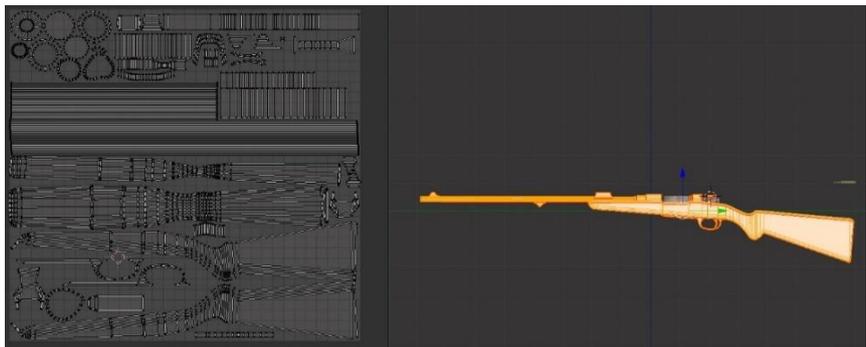


Fig. 19 Despliegue de la malla en coordenadas UV

- Usando el programa 3D: Mediante esta forma tenemos varios medios. Algunos programas como *Blender* y *Zbrush* permiten pintar directamente sobre nuestro modelo 3D mediante una gran variedad de brochas. Lo que pintemos en nuestro modelo será añadido a nuestra textura acoplándose a las coordenadas UV. En la siguiente imagen se muestra la interfaz de brochas de Blender
- El otro modo de editar la textura dentro del programa es haciendo un “*Bake*” (horneado) sobre nuestro modelo. Primero separaremos nuestro modelo por partes en función de los materiales que gaste. Crearemos otro modelo igual que esté formado en una sola pieza. Una vez hecho esto, haremos un “*Bake*” del primer modelo sobre el segundo. De esta forma se aplicarán las diferentes texturas sobre nuestro modelo en una sola.

Todas estas **texturas base** conforman una gran biblioteca y proceden de distintas fuentes. Las texturas en sí mismas son imágenes, y cuanto mejor calidad tengan, mejor funcionará la textura, pero hemos de tener en cuenta que las texturas con mucha resolución, también pueden cargar en exceso la

memoria RAM del sistema y ralentizar el juego. Las fuentes han sido variadas para su obtención. Las imágenes que por diferentes motivos no se pueden obtener de forma propia, han sido recogidas de diferentes páginas de internet especializadas en texturas para videojuegos. En cambio, para otras texturas he hecho uso de propias a partir de fotografías o edición de las que ya poseía.

El otro tipo de texturas que acompaña a las texturas base son las **Texturas de Normales**. Estos mapeados corresponden con un vector tridimensional y determinan la posición de la cara de un objeto. Esta información es recogida por cada pixel. De esta forma se permite simular relieve ya que cada pixel recibe luz o se sombrea según la orientación de su vector normal. Las principales son las *“Normal Bump”*, que se utilizan para simular el relieve en una superficie plana. Este efecto se consigue mediante la combinación de los ángulos de la luz y de la posición de la cámara. Por ese motivo si el ángulo que conforman ambos elementos, no es suficiente en relación a la superficie, el objeto se verá plano y el *Normal Bump*, no funcionará.

Otro de los mapeados de los que he hecho uso son los *“Ambient occlusion map”*. Esta textura crea un sombreado suave sobre el modelo, creando mejores efectos de volumetría, y permitiendo crear zonas oscuras de forma natural en función de la luz ambiental y directa que reciben.

Estos son los principales tipos de texturas que han sido empleados en el desarrollo del trabajo. Para crear los distintos efectos en materiales y conseguir los mapeados, se han utilizado tanto los programas de edición 3D para hacer hornados de modelos *highpoly*, como los programas de edición de imagen. En estos últimos, haciendo uso de las herramientas que proporciona el programa *“nDo2”* especializado en la creación de imágenes apoyado por Photoshop.

Por ultimo hablar de la calidad de imagen. En este apartado, hay que tener en cuenta la optimización de las texturas. Si vamos a trabajar para equipos con una cantidad de memoria RAM dedicada limitada, la resolución de las imágenes tendrá que ser inferior. Yo me he movido en el marco de

resoluciones de esta generación de videojuegos, 2048x2048 como máxima y 512x512 como mínimas, dependiendo del modelo, pero siempre de base 2.

## 7.4. DEFINICIÓN DE LOS NIVELES

El siguiente apartado hablará de los niveles que construyen el conjunto de la demo. En él se explican los distintos niveles según su tipología, separándolos en 3 categorías: exteriores, ficticios y de interior. Aparecerá explicada la forma en la que han sido contruidos y que tipo de jugabilidad posee cada nivel.

### 7.4.1. NIVELES EXTERIORES

La demo posee cuatro niveles exteriores naturales con diferente tipo de ambientación. De esta forma se consigue evitar la repetición y que las imágenes de naturaleza aireen al jugador.

Para la construcción de cada uno de los niveles se ha utilizado el mismo método. Primero se decide la medida del terreno *Fig.26* en función de lo que necesitamos o hayamos establecido para el nivel.



Fig.26 Extensión del terreno

Una vez escogidas las medidas del terreno, comienza su proceso de modelado, mediante las herramientas que nos ofrece *Unity Fig.27*. Con estas herramientas podremos crear volúmenes positivos o negativos para modelar nuestras montañas, dunas o caminos.

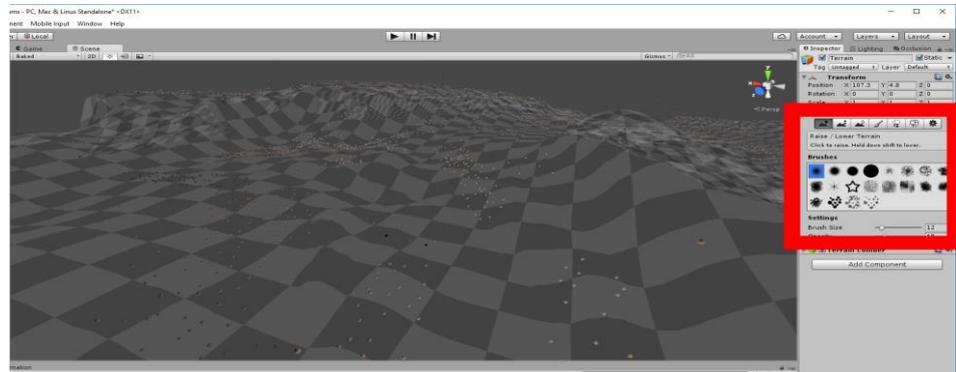


Fig.27 Brochas de modelado para terrenos en Unity

Establecida la topología del mapa, se procede a la colocación de texturas. Para colocarlas, son añadidas a la biblioteca de texturas dentro de nuestro editor de terreno Fig.28, de forma que podamos utilizar diferentes tipos en función de su posición (caminos, laderas, suelo, rocas, etc...).

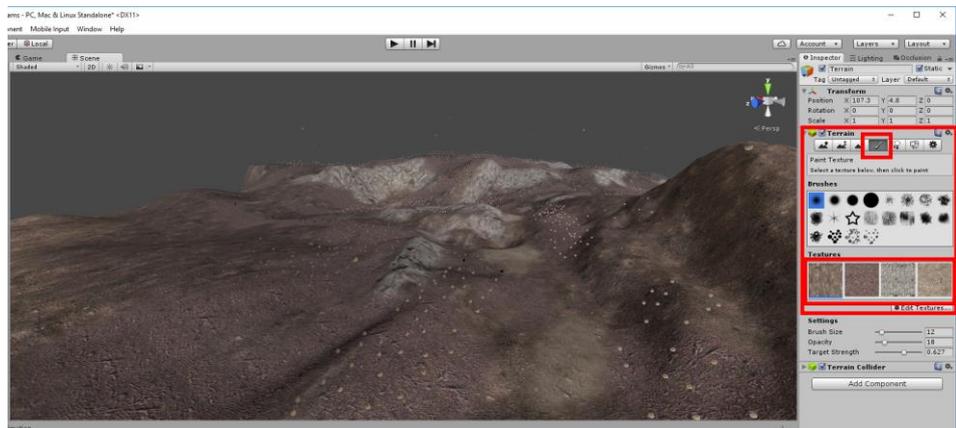


Fig.28 Paleta de textura para terrenos

Acabado este proceso se comienza la colocación de cada uno de los elementos (*Props*) que sean necesarios para completar la ambientación de nuestro nivel, se colocan los efectos de partículas como la calima, la niebla o la nieve y la colocación de muros invisibles que delimiten el área de juego.

Cuando todos los elementos han sido colocados se establece que elementos van a ser estáticos (Fig.29) en escena y cuáles no, para generar la iluminación final. Este proceso es muy importante, los elementos estáticos al tener la iluminación previamente calculada consumen menos recursos que los generados en tiempo real. Para decidir qué elementos serán estáticos, se tiene en cuenta si serán objetos con los que se interactúe o como dice su nombre si

son estáticos o tienen movimiento.



Fig.29 Pestaña para cambiar la iluminación de un objeto a estática

Una vez realizada esta tarea se genera la iluminación dentro de la pestaña *Lighting* (Fig.30) en la que encontraremos parámetros como:

- Ambient Intensity
- Reflexion Intensity
- Reflection Bounces
- Ambient Occlusion
- Fog

Con la ayuda de estos parámetros podemos ajustar sus valores en función de la iluminación que busquemos para nuestro nivel.

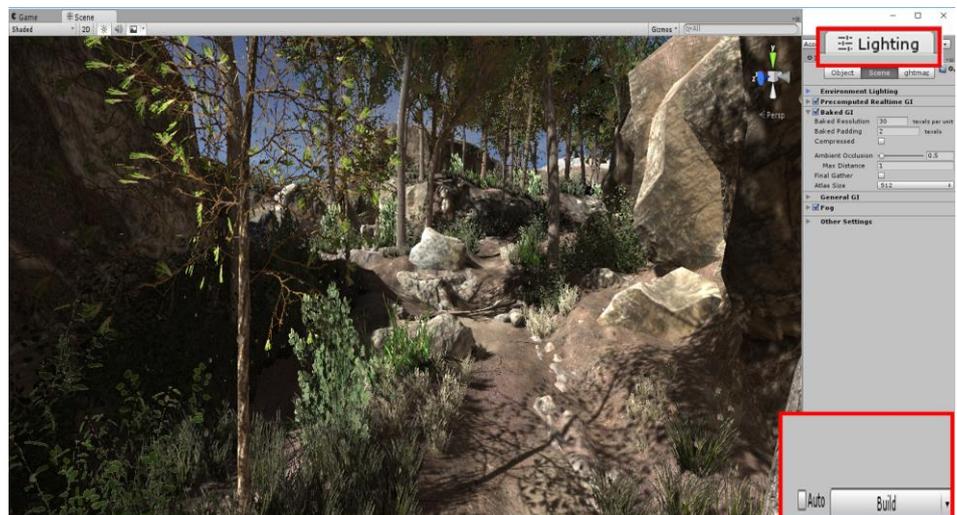


Fig.30 Renderizado final con panel de control de iluminación



Fig.31 Efectos gráficos en la cámara

Por último, se colocan los efectos en la cámara de nuestro jugador (Fig.31) para darle un aspecto visual más profesional añadiendo efectos para el alisado de bordes de sierra, mejoras en la iluminación, rayos solares y niebla.

En estos niveles he procurado conseguir la ambientación deseada haciendo uso de cada uno de los elementos creados junto con la iluminación para dotar a cada uno de estos niveles de personalidad propia.

Como ya se ha mencionado, los distintos niveles naturales que aparecen en la demo, tienen distinto tipo de jugabilidad, a pesar de ello, el objetivo final es conseguir llegar hasta la puerta que nos lleva al siguiente nivel. Fuera de lo que se refiera al apartado gráfico, cada uno de estos niveles, enfrenta al jugador a diferentes retos:

- En el primero, el bosque, (Fig.32) el jugador comenzará en una habitación con una animación de inicio levantándose de la cama. Se encontrará con que la puerta estará cerrada y para ello necesitará buscar la llave que se encuentra en la habitación. Una vez fuera se encontrará con un paisaje boscoso en el cual tendrá que conseguir la llave que le abre la puerta al final del sendero. Para ello tendrá que seguir las pistas dejadas en los diferentes mapas que aparecen a lo largo del mapeado. En ellos aparecerán indicaciones que deberá seguir haciendo uso de su memoria o guiarse por la música que emite cada uno de ellos hasta hallar la llave.
- En el segundo, Montañas nevadas, (Fig.33) tendrá que hacer uso de su vista y orientación. Ayudándose de los jalones y de las huellas dejadas en la nieve tendrá que adentrarse en la tormenta y encontrar el camino correcto para encontrar la siguiente puerta. Conforme el jugador descienda de la montaña la dificultad aumentará ya que la visibilidad será cada vez menor a causa de la densidad de la niebla, con posibilidad de escoger el camino equivocado, volver sobre sus pasos y volver a orientarse.
- En el tercero, El laberinto, (Fig.34) el jugador se encontrará en un interior sin ventanas. La única salida será un camino que lo adentrará



Fig. 32 Nivel 1: Bosque



Fig.33 Nivel 3: Montañas nevadas



Fig.34 Nivel 5: El Laberinto

dentro de un laberinto. Tendrá que guiarse por su orientación. Dentro de este laberinto habrá pasadizos con trampa. Una vez pasado ciertos tramos, equivocados, se convertirán a su espalda en caminos de una sola dirección que se cerrarán y le harán retroceder volviendo al mismo punto.

- El último, El sendero, (Fig.35) situará al jugador en un camino y lo adentrará a una cueva. En ésta, tendrá que hacer uso de su habilidad para superar los obstáculos. Si cae al fondo de la caverna el jugador tendrá que repetir el recorrido desde el principio. Una vez superado el recorrido tendrá que saltar al vacío, para superar el nivel.

#### 7.4.2. NIVELES FICTICIOS

Estos niveles han sido creados fruto de la propia imaginación y de su funcionalidad. De forma que, para realizarlos, primero se pensó en la jugabilidad que iban a poseer. Al ser mapas que no muestran algo real, se prescindió de formas o elementos que podríamos ver en nuestra realidad y luego se simplificó a formas simples. Para crearlos, se ha seguido un proceso similar a los anteriores. Primero se establece un espacio que conforme el terreno de juego y, luego, se colocan los elementos necesarios. Esta vez, los elementos son formas simples y se crean mediante módulos. Estos módulos permiten rápidamente crear un “prefap” que será clonado para ir creando el recorrido deseado. Más tarde, se añaden los elementos de decoración, texturas especiales y animación, que vayan acorde con lo deseado.

Como ya se ha dicho, el objetivo es llegar al final del nivel para pasar al que le sigue. En estos tres niveles se realiza de diferentes formas.

- El primer nivel se ambienta en mitad del espacio (Fig.36). En él, aparece un laberinto de escaleras de un material cristalino que refleja el espacio. Tendremos que hallar el camino subiendo por ellas, haciendo uso de nuestra habilidad y orientación, teniendo en cuenta la falta de gravedad.



Fig.35 Nivel 7: El sendero

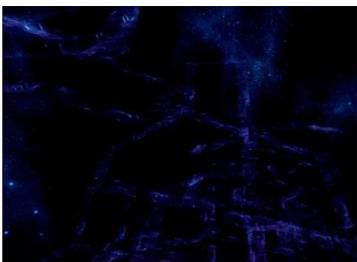


Fig.36 Nivel 2: Escaleras



Fig.37 Nivel 4: Laberinto

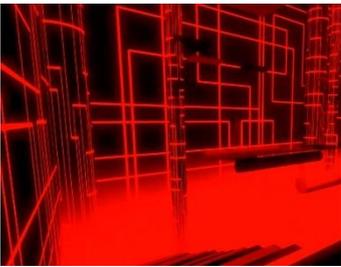


Fig.38 Nivel 6: Nivel de

- El segundo nivel nos sitúa en mitad de una sala al aire libre (*Fig.37*). En ella nos dan la primera llave que nos ayudará a acceder a un laberinto de colores. Conforme superemos los laberintos, nos irán proporcionando llaves, hasta conseguir la que nos permitirá salir. Estos laberintos están contruidos de forma que cada uno tenga un color predominante. Por ejemplo, si te adentras dentro del laberinto amarillo, verás que lo conforman tres colores, pero el camino real siempre tendrá el color del laberinto como color dominante. De esta forma si escoges un camino incorrecto el color del laberinto dejara de estar presente en las paredes que nos rodeen.
- El ultimo nivel está ambientado en una sala de un rojo vibrante (*Fig.38*), con columnas, colores, objetos moviéndose y una música molesta hasta el punto de que no nos permita concentrarnos en nuestra tarea. En él, el jugador tendrá que concentrarse para superar los obstáculos y las plataformas móviles, haciendo uso de su agilidad para no caerse, y tener que volver a comenzar. A pesar de ello, al ser uno de los últimos niveles y, por lo tanto, la dificultad ser mayor, se han preparado puntos de reaparición para ayudar a superarlo y se ha escondido un atajo que nos permitirá saltar al siguiente nivel.

### 7.4.3. INTERIORES

A pesar de que se habla de niveles en plural, realmente sólo un interior es independiente. Esto se debe a que el primero se encuentra dentro de otro por decisión de diseño. Para realizarlos, se han utilizado como referencia espacios reales. Se ha recolectado toda la información en imágenes y medidas de los elementos reales para su posterior creación 3D.

Para realizar la iluminación se ha utilizado un sistema de iluminación por horneado (*Fig.41*) que mejora notablemente el aspecto de la luz y las sombras.

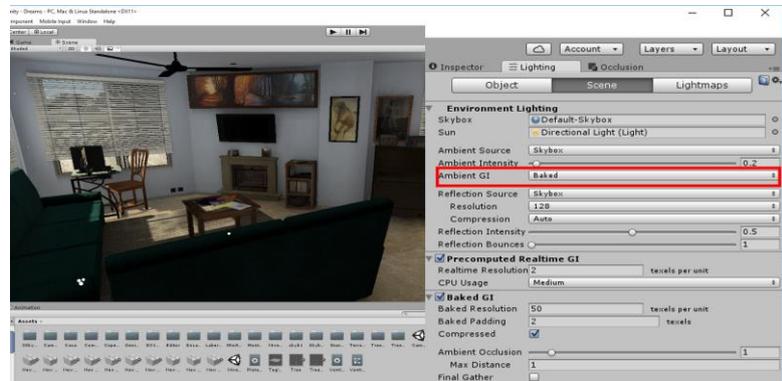


Fig.41 Panel de control de iluminación

Estos niveles definen el principio y final de la historia. A pesar que la historia no está reflejada en la demo, sí aparece la introducción de la misma. El jugador se encuentra en la cama de su habitación. Despierta, se incorpora y se encuentra con esta serie de lugares. Todo finaliza en el último nivel, en el que se despertará en una habitación, en este caso el comedor, y comprobará como los niveles no han sido más que un sueño y un reflejo de su trabajo, el cual aparecerá esparcido en forma de imágenes por la habitación.

## 8. TESTEO

Definimos testeo como la fase que debe pasar un juego en sus diferentes etapas antes del lanzamiento comercial o publicación. En esta fase, se comprueba que la versión que está siendo desarrollada no contenga errores. Si se encuentran, se corrigen para continuar con la siguiente fase del desarrollo.

Cuando han sido encontrados y corregidos estos errores, se siguen realizando comprobaciones a lo largo del desarrollo. Esto se debe a que los juegos pasan por diferentes etapas como la fase Alpha y Beta, y durante su desarrollo se realiza una continua comprobación de errores. En las grandes empresas, en ocasiones estas Betas o Alphas son ofrecidas en abierto al público para que reporten los errores encontrados para su corrección antes de su salida al mercado. Los errores que se pueden encontrar en esta fase pueden ser: Errores de jugabilidad, gráficos, de audio, de red, de memorias, archivos corruptos, de controles, de texto...

En mi caso he contado con conocidos para realizar el testeo y corregir los errores más grandes. Entre ellos cambiar aspectos dentro de la faceta jugable, como fallos en colisiones, errores en la posición de la cámara, scripts que no se ejecutan, falta de respuesta con el controlador y errores gráficos como en la carga de texturas, efectos visuales molestos que causaban dolores de cabeza y estómago, y cambio en algunos elementos, por otros con un mejor acabado.

## 9. OPTIMIZACIÓN

En el mundo de los videojuegos la optimización es una parte esencial. Con ella conseguimos aprovechar mejor los recursos de nuestro sistema y un rendimiento mejor a la hora de mover nuestros videojuegos. Por ello a la hora de optimizar es importante cuidar cada uno de los aspectos del juego, ya sean gráficos, de código, de sonido, etc... todo en general. Dentro de mi proyecto los apartados que más optimización han requerido han sido los modelos 3D y las texturas. Los modelos han tenido que ser construidos con el menor número de polígonos posibles mediante una retopología agresiva y las texturas han tenido que mantenerse en un máximo de resolución de 2048x2048 px. Dentro del juego se ha hecho uso del sistema de carga por distancia, que como su propio nombre indica, carga el modelo completo o un modelo de menos polígonos, dependiendo de la distancia a la que es divisado. De esta forma el juego ha sido optimizado para un equipo de gama media con una tasa de Fps (frames por segundo<sup>56</sup>) en torno a los 30/40. Los requisitos recomendados serían los siguientes:

- Procesador: AMD FX-6300 3,5 Ghz
- Tarjeta gráfica: Gtx 660 2gb o compatible
- Memoria: 6 gb RAM
- Espacio en el disco duro: 3gb
- Sistema Operativo: 64-bit Windows 7, Windows 8.1 y Windows 10

---

<sup>56</sup> Es la velocidad (tasa) a la cual un dispositivo muestra imágenes llamadas cuadros o fotogramas.

## 10. CONCLUSIONES

Una vez acabada la parte práctica del proyecto y atendiendo a los objetivos planteados, puedo concluir que los resultados han sido satisfactorios en cada uno de los apartados planteados.

Teniendo en cuenta que solo he tenido un año de trabajo y la base para trabajar era mínima, realizar 8 niveles y que funcionen de forma correcta, junto con la calidad gráfica deseada, es un resultado más que satisfactorio. Todo ello conlleva una serie de conocimientos que no se tenían al principio del desarrollo y que han servido para nutrirme en este mundo de los videojuegos.

En cuanto a la rúbrica, se ha procurado añadir el máximo contenido posible, para comprender el contexto del proyecto y todo el trabajo que ha conllevado realizarlo. A pesar de ello, el límite de extensión no ha permitido profundizar en otras áreas menores o expandir las que ya aparecen. A pesar de ello, la información descrita en esta rúbrica es la necesaria para abarcar en gran medida los aspectos más importantes del proyecto.

Por último, hablar de las impresiones obtenidas por las personas que han testeado la demo. Gracias a ellas se ha conseguido crear la línea de dificultad deseada para la demo, y ha conseguido mantenerlas en juego un mínimo de 20 minutos, lo cual siendo una alpha y haber sido producido por una sola persona, es un resultado más que satisfactorio.

En este enlace se puede descargar tanto la demo como los archivos que se relacionan en el anexo de la página 44.

<https://mega.nz/#!hhsnUT6B!7ZVxrwHCoNzIU5Cwln8TOUKLD4T5qa0ZF9eu5JrZZtc>.

## 11. BIBLIOGRAFÍA

BIRN, Jeremy. Iluminación y render. Anaya, 2006.

COTO, E. Introducción a los sistemas de partículas. Venezuela: Universidad central de Venezuela, 2004

DAMERS, Owen. Técnicas digitales de textura y pintura. Gustavo Gili, 2002.  
 DÍAZ, J.C. Modelo Metodológico Para El Desarrollo De Videojuegos En 3D.  
 ROLLINGS, A. & MORRIS, D. Game Architecture and Design. New Readers 2004.  
 SALEN, K. & ZIMMERMAN, E. The Game Design Reader. Massachusetts  
 Institute of Technology, 2006.  
 VAX, LOUIS. Arte y Literatura fantásticas. Bs.As., Eudeba, 2da. ed., 1971.

### 11.1 Material Online

Nuevas tecnologías en consolas que impulsan el hiperrealismo en videojuegos.

2013, CNN. Disponible en:

<http://cnnespanol.cnn.com/2013/11/25/nueva-tecnologia-en-consolas-impulsa-el-hiperrealismo-en-videojuegos/>. Consulta el 15/2/15

Industria de los videojuegos, 2016. Disponible en:

[https://es.wikipedia.org/wiki/Industria\\_de\\_los\\_videojuegos](https://es.wikipedia.org/wiki/Industria_de_los_videojuegos). Consulta 8/6/2015

YOUTUBE, Canal Unity: <https://www.youtube.com/user/Unity3D>. Consulta el 12/8/2015

## 12. ANEXOS

Como apoyo al TFG ha sido creado un anexo en el cual aparece un enlace de descarga para la Demo junto con un documento de texto en el cual se especifica la información necesaria para la demo: Requisitos, controles, pistas o atajos para su disfrute. Aparte, se incluyen varios archivos de video. El tráiler de la demo y ocho videos con la demo resuelta.

<https://mega.nz/#!hhsnUT6B!7ZVxrwHCoNzIU5Cwln8TOUKLD4T5qa0ZF9eu5JrZZtc>.

Para acabar he añadido una galería de imágenes de los distintos niveles:

Controles de la demo:

- Movimiento: w,s,a,d
- Salto: Barra espaciadora
- Esprintar: Mayus Izq.
- Abrir puertas/coger llaves: Click izq. Raton



