

TFG

DISEÑO Y MODELADO DE ESCENOGRAFÍA Y ENTORNO PARA EL PROTOTIPO DE UN NIVEL DE VIDEOJUEGO 3D.

Presentado por Carlos Juan Navarro

Tutor: Francisco Martí Ferrer

Facultat de Belles Arts de Sant Carles

Grado en Diseño y Tecnologías Creativas

Curso 2021-2022



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA
FACULTAT DE BELLES ARTS DE SANT CARLES

RESUMEN

Este trabajo de fin de grado consiste en la creación de un escenario para un nivel de videojuego 3D. Para llevarlo a cabo, se traslada al formato de videojuego 3D una preproducción de animación 2D realizada anteriormente en un proyecto grupal. Debido al cambio de formato y medio, han sido necesarios un rediseño y una adaptación de los distintos elementos. Finalmente, tras pasar por todas las fases de desarrollo como modelado y texturizado, se ha implementado el escenario en un motor de videojuegos.

PALABRAS CLAVE

3D, *Environment Artist*, Escenografía, Modelado, Videojuegos.

ABSTRACT

The present bachelor project consists on creating an environment for a 3D video game. In order to perform it, a 2D animation pre-production, which was previously conducted as a group project is adapted to 3D video game format. Due to the change of format and medium, it was necessary to redesign and adapt different elements. Finally, after following all the development stages such as modelling and texturing, the environment has been tested in a game engine.

KEYWORDS

3D, *Environment Artist*, Scenography, Modelling, Video games.

CONTRATO DE ORIGINALIDAD

El presente Trabajo de Fin de Grado ha sido realizado íntegramente por el alumno Carlos Juan Navarro. Este es el último trámite para la obtención del título de la promoción 2017/2021 del grado en Diseño y Tecnologías Creativas de la Universidad Politécnica de Valencia.

Este documento es original y no ha sido entregado como otro trabajo académico previo, y todo el material tomado de otras fuentes ha sido citado correctamente.

FIRMA:

FECHA: 24/11/2021

A handwritten signature in black ink, consisting of a stylized initial 'C' followed by 'J.' and a horizontal line extending to the right.

AGRADECIMIENTOS

A Paco, sin cuya inestimable ayuda este trabajo no habría sido posible.

A KB, por ser lo que siempre he buscado y por haber estado ahí siempre que he necesitado apoyarme en amigos de verdad.

A Alberto, si he crecido tanto en esta etapa, ha sido gracias a ti.

A Carlos y a Sandra, sois lo más importante que me llevo de estos años y sin vosotros nada hubiera sido lo mismo.

Y por supuesto, a mi familia, gracias por darme la oportunidad de haber llegado hasta aquí.

Gracias.

1. INTRODUCCIÓN	6
1.1. Motivación, justificación y estructura de la memoria	6
1.2. Objetivos	7
1.3. Metodología	8
<i>1.3.1 Planificación del trabajo</i>	8
<i>1.3.2. Software y flujo de trabajo</i>	10
2. CONTEXTUALIZACIÓN	12
2.1. El entorno en el videojuego 3D	12
2.2. Fuentes documentales	14
2.3. Referentes profesionales	16
3. DESARROLLO	18
3.1. Preproducción	19
<i>3.1.1. Briefing</i>	19
<i>3.1.2. Conceptualización</i>	21
<i>3.1.2.1. Composición</i>	22
3.2. Producción	24
<i>3.2.1. Modelado</i>	24
<i>3.2.1.1. Blockout y Low Poly</i>	24
<i>3.2.1.2. Highpoly</i>	26
<i>3.2.2. UVs, Baking y Texturizado</i>	28
<i>3.2.2.1. UVs y Baking</i>	28
<i>3.2.2.2. Texturizado de props</i>	30
<i>3.2.2.3. Materiales de arquitectura y terreno</i>	32
<i>3.2.3. Implementación en el motor</i>	33
<i>3.2.3.1. Terreno</i>	33
<i>3.2.3.2. Props</i>	38
<i>3.2.3.3. Animaciones</i>	39
<i>3.2.3.4. Iluminación y atmósfera</i>	42
4. CONCLUSIONES	44
5. BIBLIOGRAFÍA	45
6. ÍNDICE DE FIGURAS	47
7. ANEXOS	49

1. INTRODUCCIÓN

El desarrollo de este proyecto se realiza durante el curso académico 2020-2021. Se trata de un trabajo realizado de forma individual y orientado al desarrollo de un escenario en 3D para un videojuego desde el concepto hasta la implementación en el motor de videojuego.

1.1. MOTIVACIÓN, JUSTIFICACIÓN Y ESTRUCTURA DE LA MEMORIA

El proyecto nace a partir de una motivación personal sobre el mundo de los videojuegos y se desarrolla en torno al objetivo principal de diseñar, elaborar e implementar un escenario para el prototipo de un nivel de videojuego 3D. De este modo, el desarrollo de este trabajo de fin de grado supone un proceso continuo de aprendizaje tanto creativo como técnico, y permite la incorporación de los resultados finales a un portfolio profesional en el ámbito de *3D environment artist*.

El desarrollo de este entorno en 3D ha sido realizado íntegramente durante el proceso de este trabajo de fin de grado, a partir de un proyecto para la asignatura “Preproducción de animación”¹, cursada en el 3º curso de nuestro grado, en el que mis tareas principales fueron diseñar -junto a Sandra Guilabert- todos los escenarios de la serie, y realizar el dibujo final de cada uno de ellos. También me encargué de varios *testframes*, así como del diseño y dibujo de varios *props*.



Fig.1. Portada del *Sale Folder* de “Pachuki Forest”.

¹ Consistente en la preproducción de “Pachuki Forest” (fig. 1), una serie de animación 2D infantil en cuya etapa de preproducción se desarrollaron diferentes personajes y entornos, así como la línea narrativa general que seguiría. El proyecto fue realizado en conjunto con otros cuatro alumnos del grado: Carlos Manuel Bea Masero, Sandra Guilabert Berlamas, Borja Martín Más y Jorge Irles Lara, bajo la supervisión del profesor Raúl González Monaj.

Desde la realización de este proyecto colectivo y durante los meses previos al comienzo del presente trabajo, concreté la idea general: adaptar uno de los escenarios ideados durante la preproducción de la serie de animación en un entorno 3D para su implementación en un videojuego, operación compleja que me permitiría incorporar los conocimientos adquiridos durante el grado, desarrollar un estilo gráfico para 3D y aprender técnicas y flujos de trabajo utilizadas en la industria del videojuego. Estas cuestiones se abordan en el apartado 1.3 *Metodología*, y en el capítulo 3 *Desarrollo*, se especifican en relación con las distintas fases del proyecto.

Previamente al desarrollo, se ha introducido un capítulo (cap. 2) que contextualiza el trabajo y reflejan los principales referentes que se han utilizado para investigar cuestiones técnicas y estilísticas.

1.2. OBJETIVOS

El objetivo general de este trabajo consiste en elaborar el escenario para el prototipo de un nivel de videojuego 3D, rediseñando la preproducción para una serie de animación 2D, e implementarlo en un motor de juegos añadiendo iluminación, entorno y animación de algunos elementos.

Podemos desglosar este objetivo general en una serie de objetivos específicos:

- Concretar el estilo artístico a partir del material elaborado para la serie 2D y los referentes técnicos y artísticos, teniendo en cuenta factores como el nivel de estilización, la composición y el color.
- Determinar el software adecuado para el propósito de este proyecto a partir de documentación, evaluando las capacidades actuales de los motores de juegos en función del estilo artístico del proyecto.
- Durante todo el proceso, incorporar los conocimientos adquiridos durante el grado y aprender técnicas de modelado y texturizado 3D utilizadas en la industria del videojuego, como el flujo de trabajo *high poly-low poly* y la creación de texturas *seamless*.

- Diseñar y modelar los elementos arquitectónicos y *props* de la escenografía.
- Implementar estos elementos en un motor de juegos y añadir iluminación, y fondos.
- Incorporar los resultados a mi portfolio con el fin de mostrar mis capacidades en el ámbito profesional de *environment artist*.

1.3. METODOLOGÍA

Para la realización de este trabajo de fin de grado ha sido necesario realizar un trabajo de documentación y análisis sobre diferentes aspectos estilísticos y técnicos relacionadas con el perfil profesional de *environment artist*. A partir de este análisis se han sintetizado flujos de trabajo teniendo en cuenta los estándares en la industria del videojuego, desde la fase de concepto de cualquier elemento de la escena hasta su implementación, contemplando modelados *lowpoly* y *highpoly*, topología, despliegue de UVs, transferencia de detalle mediante mapas, materiales, mapas de color e iluminación y características del motor de juego y de las plataformas de destino.

En cuanto a la temporalización, el proyecto se ha desarrollado durante la etapa comprendida entre marzo y noviembre del 2021. Es importante mencionar que la realización de este trabajo de fin de grado se ha dividido en tres etapas, debido a mi participación durante un periodo intermedio en el programa de becas Meridies del Centro de Cooperación al Desarrollo de la UPV.

En los siguientes apartados, se detallan el cronograma del proyecto, el proceso que se ha seguido para la creación de la escena, el software utilizado y las técnicas de 3D concretas empleadas.

1.3.1. Planificación del trabajo

La duración total del proyecto ha sido de ocho meses, comenzando en el mes de marzo y finalizando en noviembre del 2021. Durante el mes de julio, participé en el programa de becas Meridies referido anteriormente, organizado por el Centro de Cooperación al Desarrollo de la UPV, y realicé una estancia en el

Centro Cultural de España en Guatemala, llevando a cabo diferentes proyectos de carácter audiovisual.

Para una correcta planificación del trabajo, se realizó un diagrama de Gantt (Fig.2) contemplando las diferentes etapas y fases del proyecto.

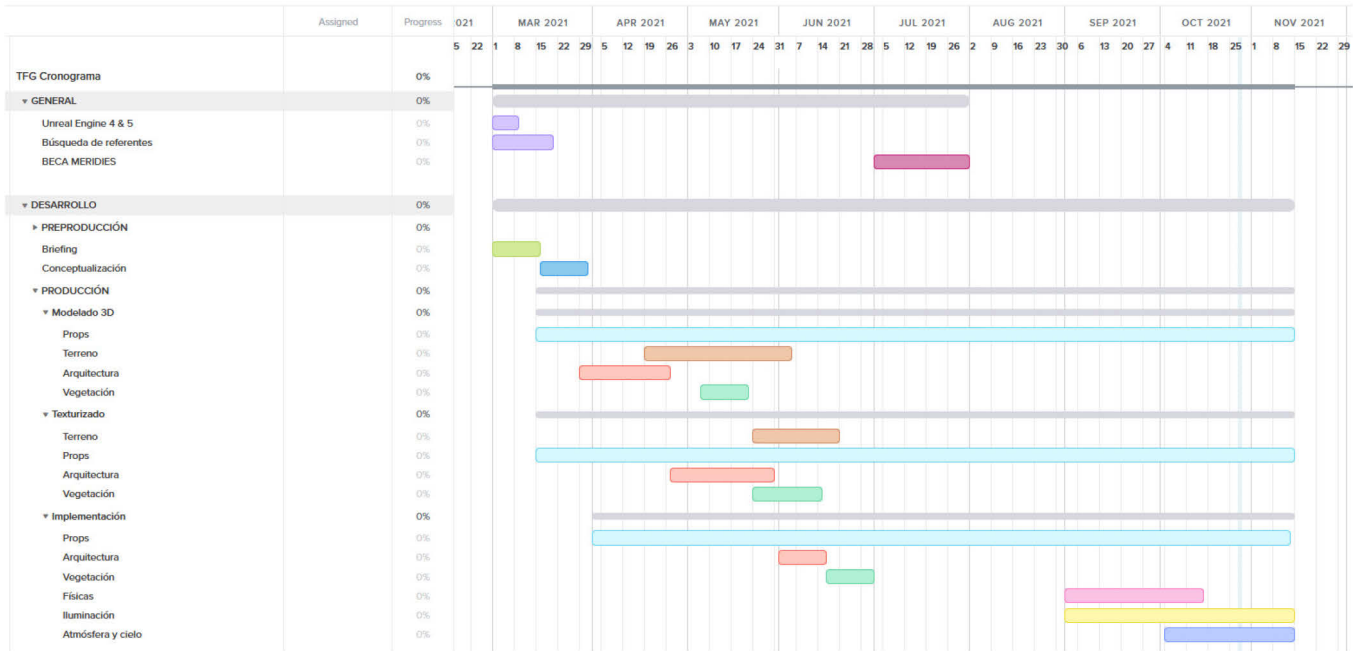


Fig. 2. Cronograma del proyecto en formato de diagrama de Gantt.

En lo que respecta al desarrollo efectivo del trabajo, esta previsión inicial se ha cumplido de forma general, si bien (como es habitual en este tipo de proyectos), se han intercalado tareas necesarias para ajustar aspectos de fases precedentes, en función de la consecución de objetivos concretos.

De este modo, la distribución final del desarrollo del proyecto ha sido la siguiente:

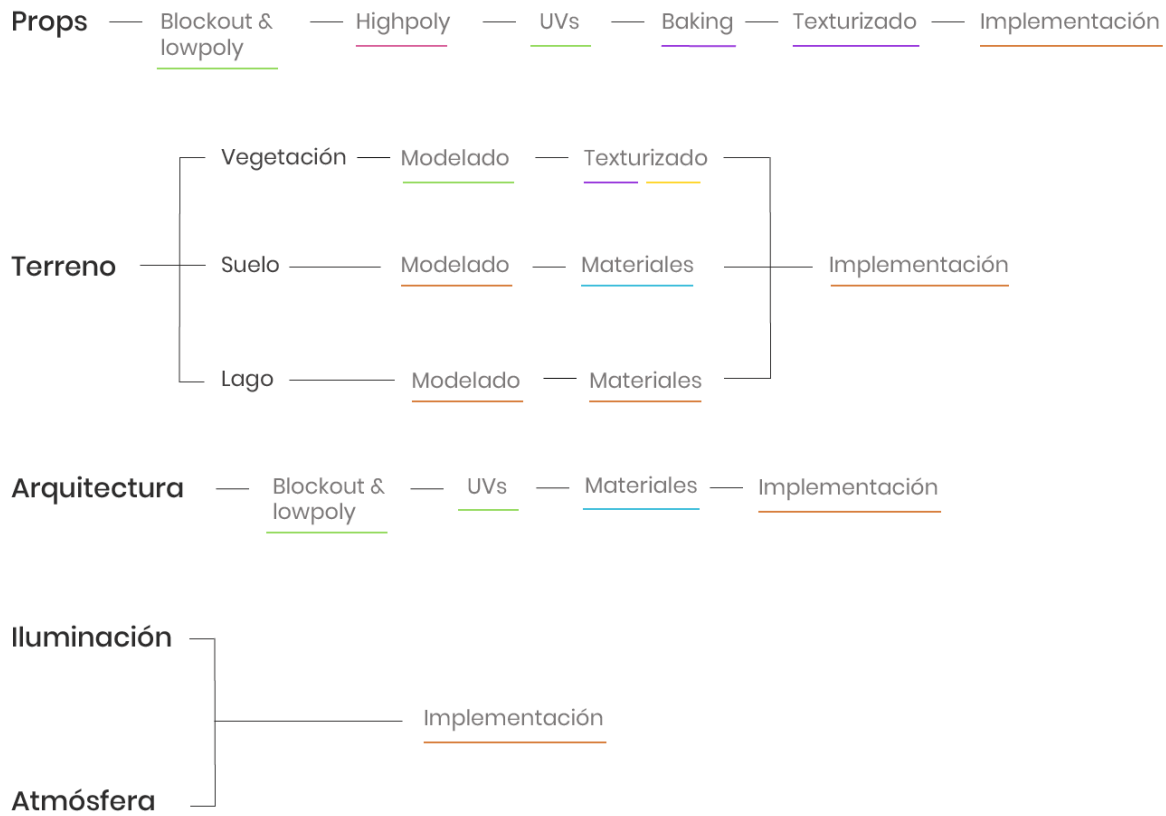
- **Marzo a junio de 2021:** se lleva a cabo una gran parte del desarrollo de la escenografía en 3D. El terreno queda definido, así como los elementos principales de la composición, y se comienza a trabajar en algunos elementos detalle. Tanto la arquitectura como la vegetación quedan en un estado avanzado que permite trabajar más tarde en la incorporación de nuevos elementos, así como de otros factores como la iluminación. Además, durante estos meses, se comienza la planificación de la memoria del proyecto, estableciendo los diferentes apartados y comenzando a escribir las líneas generales del contenido.

- **Julio 2021:** Durante este mes, se avanza en la memoria del proyecto, simultaneando esta tarea con la participación en el programa de becas Meridies, en este caso en el Centro Cultural de España en Guatemala.
- **Agosto a noviembre de 2021:** se finaliza el desarrollo del entorno en 3D, prestando atención a los diferentes *assets* restantes, y a otros elementos como la iluminación y el entorno. Habiendo implementado todos los elementos, se realizan los renders pertinentes y se da por acabada la escenografía. Se acaban de redactar los apartados técnicos de la memoria, se reflejan las conclusiones y se elaboran los diferentes anexos que se consideran necesarios.

1.3.2. Software y flujo de trabajo

En atención a las necesidades del proyecto, se ha definido un flujo de trabajo para cada tipología de objeto (*props*, vegetación, etc...) y se ha ido adaptando, atendiendo a las necesidades particulares de cada elemento. Como es habitual en el entorno profesional, se han utilizado diferentes *softwares* especializados.

Así pues, lo primero que se hizo fue una labor de documentación sobre la industria profesional y sobre técnicas que se utilizan actualmente, para posteriormente hacer pruebas de *workflow*. Tras determinar cuál era el flujo de trabajo óptimo para cada tipo de elemento, se procedió a su aplicación, contemplando algunas variaciones, dependiendo de las particularidades de determinados elementos. El siguiente esquema (Fig.3) muestra el flujo de trabajo para la elaboración de cada tipología de objeto.



Blender ZBrush Substance Painter Substance Designer Adobe Photoshop Unreal Engine 5

Fig. 3. Esquema del flujo de trabajo

Las aplicaciones de software utilizadas han sido: Blender y ZBrush para modelado y UVs; Substance Painter para *baking* y texturizado general; Adobe Photoshop para el texturizado de objetos específicos; Substance Designer para materiales PBR y Unreal Engine 5 para la implementación de la escena, iluminación y entorno.

El proyecto comenzó siendo pensado para ser implementado en el motor de videojuego Unreal Engine 4, por su fácil accesibilidad de cara al usuario, así como por todas las herramientas que ofrece y la gran cantidad de documentación que se puede encontrar *online*. No obstante, durante el desarrollo de la primera fase del proyecto, en mayo, Epic Games dio acceso a una versión *beta* de Unreal Engine 5. Tras examinar la documentación oficial y distintos foros profesionales,

se consultó a una *environment artist* en activo, valorando las nuevas capacidades del motor y su aplicación a un proyecto de estas características, concluyéndose que era viable y recomendable migrar el proyecto a esta nueva versión.

2. CONTEXTUALIZACIÓN

2.1. EL ENTORNO EN EL VIDEOJUEGO 3D

Si hablamos de los videojuegos más conocidos, o si nos paramos a preguntar a una persona aleatoria sobre el mundo del videojuego, es muy posible que nos hable de las sagas más relevantes de la historia y las que más trascendencia han tenido a nivel mediático. Mario, de la saga de videojuegos homónima, Link, de la saga *The Legend of Zelda*, o incluso Pikachu, mascota principal de la saga *Pokémon*, son algunos de los personajes de videojuegos más conocidos mundialmente. Pero, ¿qué hay del mundo donde viven estos personajes?

El entorno que habitan los protagonistas de las historias que vivimos en cualquier videojuego es tan importante como los personajes en sí y, a menudo, nos ofrecen información muy valiosa sobre el universo en el que nos estamos sumergiendo. Si prestamos atención, es muy posible que nos percatemos de una gran cantidad de elementos y de detalles que nos cuenten la historia del lugar que estamos visitando, así como información sobre los personajes que viven (o no) allí, y una infinidad de cosas más que nos ayudarán en uno de los objetivos principales del videojuego: crearnos el mundo que se nos presenta. Como dice Ara Shirinian (2010): “Si se observa a la mayor escala posible, prácticamente cualquier componente visual de un videojuego puede llevar consigo una narrativa que nos cuente algo sobre el contexto o la historia” (parr. 20).

En los últimos años hemos visto muchos ejemplos de videojuegos que dan mucho valor a la transmisión información sobre el universo mediante el entorno, y no mediante cinemáticas, textos o conversaciones con personajes, dotando así de gran importancia al escenario y al apartado artístico como herramienta narrativa. Podemos ver ejemplos de esto en *Portal* (Valve

Corporation, 2007), cuando en tu intento de escapar de las instalaciones subterráneas de *Aperture Laboratories*, encuentras dibujos, símbolos y mensajes que dejaron en las paredes todos los que intentaron huir antes que tú (Fig.4), o en *Bioshock* (Irrational Games, 2007), cuando llegas a la ciudad sumergida de Rapture y encontramos pancartas, sangre y otros elementos que indican que en ese mismo lugar hubo protestas violentas contra el gobierno y sus medidas (Fig.5).

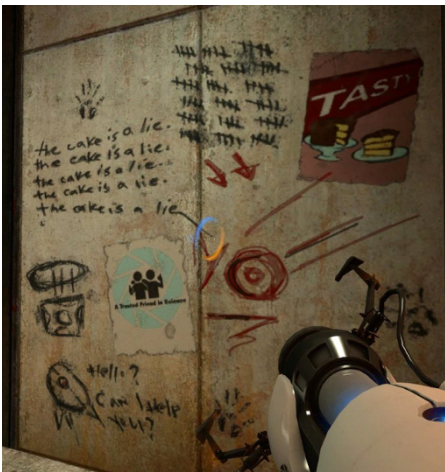


Fig. 4 y 5. Captura de pantalla del videojuego *Portal*. Captura de pantalla del videojuego *Bioshock*.

Más recientemente tenemos el caso de *The Last of Us Part II* (Naughty Dog, 2020), en el que podemos ver los efectos y el rastro físico de la catástrofe que ocurrió en el mundo post-apocalíptico que se nos plantea (Fig.6).

Fig. 6. Captura de pantalla del videojuego *The Last of Us Parte II*.



2.2. FUENTES DOCUMENTALES

Para la realización de este proyecto, el aprendizaje continuo y la búsqueda de información ha sido una parte fundamental. Como se comentaba en la introducción, la adaptación de un proyecto pensado para el 2D a un motor de videojuegos 3D supone no solo un reto a nivel artístico si no también técnico.

A continuación, se hace referencia a todas las fuentes documentales que se han seguido durante la elaboración del proyecto:

- **Documentación oficial de Unreal Engine**

La principal fuente de información ha sido la que se puede encontrar en la página web oficial del motor de videojuegos. Dentro de ella, existe un listado con todos los apartados dentro del motor, desde los actores que se pueden utilizar hasta los materiales, terreno, etc. Es importante mencionar que la versión de Unreal Engine 5 que se ha utilizado en el proyecto, no cuenta con documentación final y actualizada ya que se encuentra aún en acceso anticipado. Es por ello que se ha consultado la documentación oficial de la última versión disponible, la 4.27, y no ha existido ningún problema por cambio de versiones.

- **Tutoriales**

La otra fuente principal de información han sido los tutoriales, ya sean en formato de vídeo, de artículo de revista, de diferentes páginas webs o tutoriales oficiales del software correspondiente. Entre todos ellos, han resultado especialmente importantes y he visitado en más de una ocasión: *Making a Wardrobe Prop in 3ds Max, Substance Painter & UE5*², de Diego Torres, como guía general del flujo de trabajo *low poly-high poly*, texturizado e implementación (Fig.7), y *Beginners Guide to Creating Overwatch-Style Game Art [3D*

². TORRES ORTIZ, DIEGO. *Making a Wardrobe Prop in 3ds Max, Substance Painter & UE5* (18 de octubre del 2021). Recuperado de <https://80.lv/articles/making-a-wardrobe-prop-in-3ds-max-substance-painter-ue5/>

MODELING/TEXTURING]³, de Eric García como guía a la hora de utilizar Substance Painter y entender el texturizado estilizado (Fig.8).



Fig. 7 y 8. Render final del tutorial de Diego Torres. Render final del tutorial de Eric García.

- **The 3D Artist's Coloring Book⁴**

Es un curso online que adquirí durante los primeros meses del proyecto y que se puede completar en cualquier momento. Está formado por gran cantidad de ejercicios de todo tipo y ofrece los archivos para poder practicar dentro de Substance Painter. Gracias a este curso aprendí las bases y diferentes técnicas para texturizar distintos *props* y materiales con un acabado relativamente estilizado.

- **Consultas a profesionales**

Fue también de gran ayuda la posibilidad de consultar a Yolanda Cruz Girona, Environment/Level Artist de la empresa INGAME Studios, tanto por su *feedback* a nivel creativo como por los problemas técnicos de Unreal Engine que me ayudó a resolver.

³ GARCÍA, ERIC. Stylized Station. *Beginners Guide to Creating Overwatch-Style Game Art [3D MODELING/TEXTURING]* (23 de junio del 2020). [Archivo de video]. Recuperado de <https://www.youtube.com/watch?v=PVUw5Mheypl>

⁴ SMITH, THOMAS. Stylized Station. *The 3D Artist's Coloring Book* (2020). Recuperado de <https://courses.stylizedstation.com/p/3dcoloringbook>

2.3. REFERENTES PROFESIONALES

- ***Sunflower House***⁵ – Yee Dong Park

Este proyecto (Fig.9) se ha tomado como referencia principalmente para diferentes elementos como las rocas o la vegetación, tanto en la hierba como en los árboles. Se trata de un proyecto modelado en Blender y Zbrush e implementado en Unreal Engine, cuyos efectos del viento, el agua o el cambio de color por la perspectiva atmosférica son elementos que se busca incorporar en mi trabajo de fin de grado.

Fig. 9. Render final del proyecto “*Sunflower House*”.



- ***Bird House***⁶ – Jasmin Habezai-Fekri

Este proyecto (Fig.10), realizado en los mismos softwares que se han empleado para el desarrollo de este TFG, ha servido como referente por varios elementos. Por un lado, la utilización de una *Runtime Virtual Texture* (RVT), explicada con mayor detalle en el apartado de desarrollo, que permite que la hierba adopte el color de la textura sobre la que se encuentra, consiguiendo como resultado una variación cromática de la vegetación que enriquece visualmente la escena.

⁵. JEE, DONG PARK. *Sunflower House* (Mayo del 2021). Recuperado de <https://www.artstation.com/artwork/zOLDR6>

⁶. HABEZAI-FEKRI, JASMIN. *Bird House* (Noviembre del 2020). Recuperado de <https://www.artstation.com/artwork/mDW6Yd>

También ha servido como referente por otros elementos como el cielo relativamente estilizado o la animación con físicas de la bandera. La animación de la ropa que hay tendida, algo que también he incorporado en mi proyecto, se consigue dentro del propio material con el nodo *Rotate About Axis*, también explicado con mayor detalle en el apartado de desarrollo.

Fig. 10. Render final del proyecto "Bird House".



Fig. 11. Render final del proyecto "Pirate Radio Station".

- **Pirate Radio Station**⁷ – Zhang Xuan

Se trata de otro proyecto (Fig.11) desarrollado con los mismos softwares, que constituye un buen ejemplo de hacia donde se pretende avanzar técnicamente con el desarrollo de este proyecto. Este *environment* se ha tomado como referente por motivos similares a los expuestos en los anteriores trabajos: utilización de RVT para la hierba, nivel de estilización, perspectiva atmosférica, color y diseño del cielo, y también por cómo utiliza en su composición elementos en primer plano que no llaman demasiado la atención y que logran transmitir sensación de profundidad.

⁷. XUAN, ZHANG. *Pirate Radio Station* (Octubre del 2021). Recuperado de <https://www.artstation.com/artwork/zOgmYQ>

- **Breeze, mapa del videojuego Valorant**⁸ – Riot Games

Este mapa pertenece al *shooter* táctico multijugador de la compañía Riot Games, lanzado en el año 2020. El mapa Breeze, lanzado antes de verano del 2021, ha sido elegido como referente por contener zonas y elementos con una estética muy similar a la que se pretende conseguir en este proyecto. Así, se han realizado capturas de pantalla *ingame* y se han tomado como referencia diferentes elementos como texturas para la pared de la casa, rocas, pintura en la pared, o planchas de metal.



Fig. 12. Captura de pantalla del mapa Breeze del videojuego Valorant.



Fig. 13. Captura de pantalla del mapa Breeze del videojuego Valorant.



Fig. 14. Captura de pantalla del mapa Breeze del videojuego Valorant.

3. DESARROLLO

Las diferentes etapas por las que pasa un escenario en 3D desde su conceptualización hasta que es implementado implican un desarrollo complejo que ha de ser ejecutado correctamente, tanto en el apartado artístico como en el técnico. Para ejecutar un proyecto, se realizan procesos concretos y se aplican diferentes técnicas para que el flujo de trabajo sea lo más eficiente posible. La

⁸. RIOT GAMES. *Breeze, mapa del videojuego Valorant*. (Mayo del 2021). Recuperado de <https://playvalorant.com/es-es/maps/>

especificación de estos procesos resulta especialmente importante en una empresa AAA, en la que hay grandes equipos implicados, y donde cada *asset* pasa por diferentes etapas y es trabajado por diferentes personas antes de implementarse en el motor de videojuego.

El desarrollo de este proyecto engloba prácticamente todas las etapas de realización de un escenario en un videojuego, desde la preproducción hasta la implementación. Así, se tiene que conceptualizar, modelar, optimizar, texturizar e implementar cada *asset*, tratando siempre de mantener las buenas prácticas necesarias para este correcto funcionamiento, como seguir un orden preestablecido y lógico en la elaboración de cada *asset* o la correcta nomenclatura de los archivos. Asumir individualmente la elaboración de todos los elementos de un escenario es común en empresas *indie* o pequeñas empresas.

A continuación, se explica cómo se ha llevado a cabo el desarrollo de la escena, mostrando las técnicas y las diferentes aplicaciones de software empleadas.

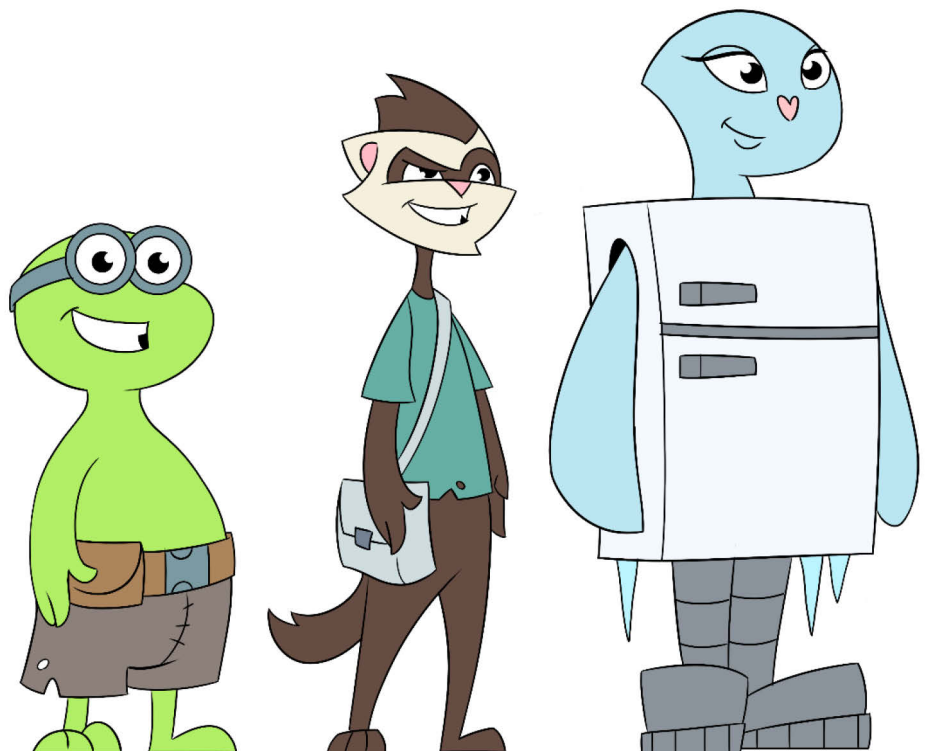
3.1. PREPRODUCCIÓN

3.1.1. Briefing

Cuando hablamos de *briefing* creativo nos referimos al documento, normalmente elaborado junto al cliente, en el que se concretan el problema y los objetivos, así como las características necesarias para llevar a cabo el proyecto satisfactoriamente. En ese, se detallan las diferentes directrices, tanto técnicas como creativas, que marcarán el desarrollo del proyecto en sí.

En este caso, no obstante, se está llevando a cabo un trabajo propio (lo que se denomina autoencargo), con lo que dichas pautas pasan a formar un *autobriefing*. Así, queda definido que el objetivo principal del proyecto es el desarrollo e implementación de un prototipo de escenografía para un nivel de videojuego 3D basado en la preproducción de una serie de animación infantil 2D. Esta serie de animación trata sobre un bosque en el que viven dos animales humanizados: un sapo y un hurón (Fig.15). Ellos recogen basura que hay desperdigada por el lugar y en las inmediaciones del lago en el que habitan, y la utilizan para construir su casa y diferentes artilugios. Un día, encuentran un tercer animal: una foca (Fig.15) perdida en el lago, a la que ayudan confeccionándole un traje para mantener su temperatura corporal. Así, se hacen amigos y viven las aventuras que marcan la línea narrativa de cada capítulo de la serie, diseñando nuevos aparatos e investigando el bosque y a sus habitantes.

Fig. 15. Personajes principales de “Pachuki Forest”.



Para la adaptación en 3D del escenario, se ha elegido la casa de los animales, así como el entorno del lago. Se busca encontrar una armonía entre un paraje natural con vegetación, agua, etc., con la casa que construyen y la basura que van acumulando. Así, el estilo escogido se aleja un poco de lo infantil que es la

preproducción original, pero sin llegar a ser completamente realista, buscando un acabado estilizado que concuerde con la idea original de la serie.

3.1.2. Conceptualización

Esta primera fase se ha basado en todo el trabajo realizado durante el curso anterior en la asignatura Preproducción de Animación. De este modo, se ha revisado el diseño del escenario original en 2D (Fig. 16 y 17), y se han valorado distintas opciones a la hora de adaptarlo a 3D, añadiendo, eliminando y modificando diferentes elementos, así como el terreno y el espacio natural que sirve de ambientación.

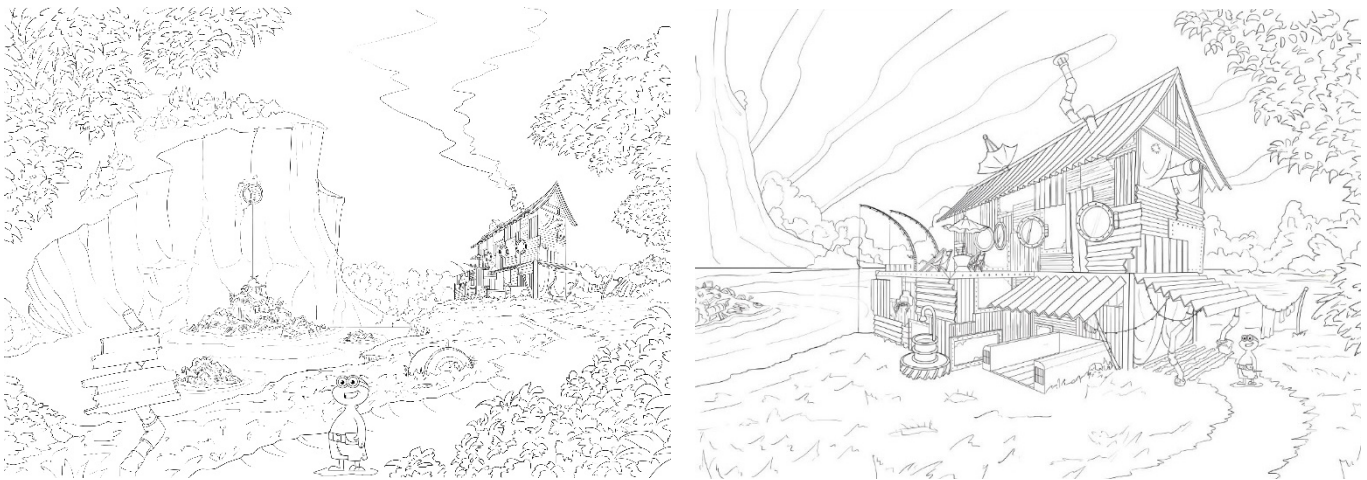


Fig. 16 y 17. Escenario final a línea de "Pachuki Forest": vista general. Escenario final a línea de "Pachuki Forest": casa.

Entre todos los cambios que se han realizado, son especialmente destacables el terreno, que ha pasado a ser menos escarpado para situar la casa más cerca del lago. Este, a su vez, ha cambiado de forma para extenderse más hacia la parte izquierda de la composición. Por otro lado, se ha simplificado la textura de las paredes del edificio en sí para mejorar su visibilidad tanto a corta como a larga distancia (Fig. 18).

Fig. 18. Composición final de la escena en 3D.

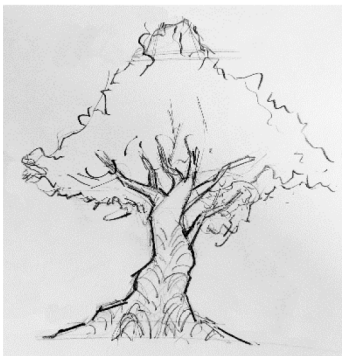


Fig. 19. Boceto del árbol a lápiz.

Algunos de los elementos como los neumáticos o las tuberías no han precisado de más preproducción antes de pasar al modelado en 3D, pero para otros como el sofá (elemento nuevo que no estaba en la preproducción original), o los árboles (Fig. 19), sí ha sido necesario realizar diseños y dibujos previos. La cantidad de basura y chatarra del escenario también se ha reducido para mejorar su limpieza visual.

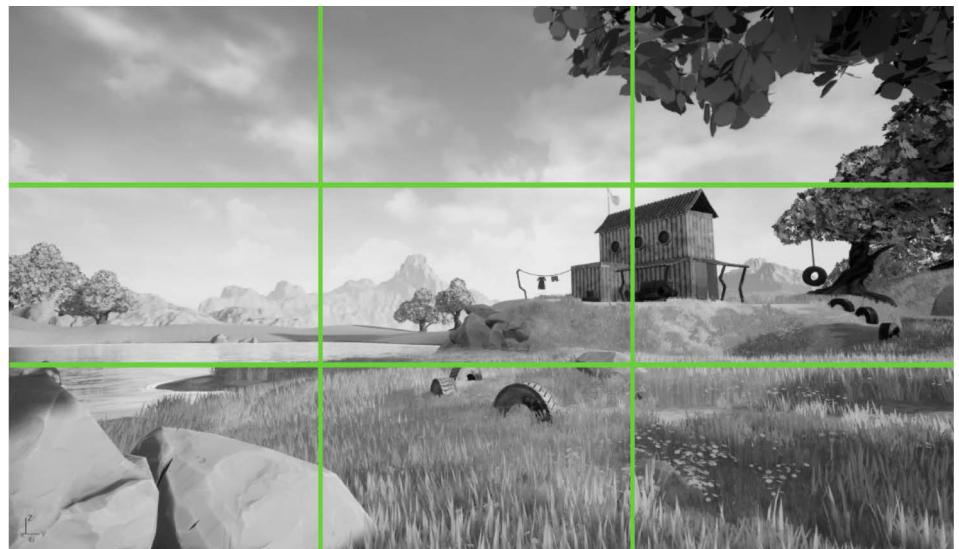
3.1.2.1. Composición

Otro de los aspectos importantes a tener en cuenta debido a la adaptación del 2D al 3D es la composición de la escena. En el proyecto original se hicieron gran cantidad de bocetos y pruebas, y finalmente se dibujó el escenario en un plano más general y otro más cercano (Fig. 16 y 17). De este modo, y al tratarse de un proyecto en tres dimensiones, se han de tener en cuenta ciertos factores para definir la composición de cara a los *renders* finales, como puede ser la posición de la cámara, su apertura, si realiza algún movimiento, etc.

Así, y desde el principio, resulta especialmente importante mantener una escala correcta de los diferentes elementos con respecto al original, así como una organización espacial coherente. Para lograr mantener cierto grado de fidelidad con los dibujos en 2D, se coloca una cámara en una posición que nos da un encuadre similar al original y se trabaja en base a dicha composición (Fig.18).

Finalmente, se opta por la misma distribución de los elementos que en el proyecto original, aunque contemplando una serie de cambios. Con el fin de lo tener que alejar la casa demasiado y no perder todos los detalles que se pretendían mostrar, se elimina el acantilado sobre el lado y se sustituye por una elevación del terreno. El lago se sitúa a la izquierda, y queda abierto en esa dirección permitiendo una correcta lectura de la imagen en el sentido izquierda-derecha tradicional. El camino, por otro lado, realiza dos curvas en lugar de una para dar mayor riqueza visual al terreno y se eleva para llegar a la casa. Se utilizan los neumáticos para marcar dichas curvas y guiar el ojo del espectador, ya que son elementos más oscuros que destacan entre el brillo de la hierba y las flores. Por último, se utilizan los árboles tanto en primer como en segundo y tercer plano, colocados a la derecha de la composición, para encuadrar esa parte y establecer una especie de “marco” en conjunto con las rocas que aparecen en primer plano en la esquina inferior izquierda (Fig. 20).

Fig. 20. Composición con regla de los tercios.



Por último, es importante hablar sobre el nivel y la cantidad de detalle y de riqueza visual que se ha tratado de aportar a las diferentes partes de la composición. Basándome en el videojuego *Outer Wilds*, (2017, Mobius Digital), este ha sido un aspecto que se ha considerado durante todo el proceso de composición y posterior implementación. Al igual que en los diferentes planetas de la galaxia de dicho videojuego, en los que hay grandes zonas con poco detalle, concentrándose este en los lugares con importancia narrativa o jugable, se ha

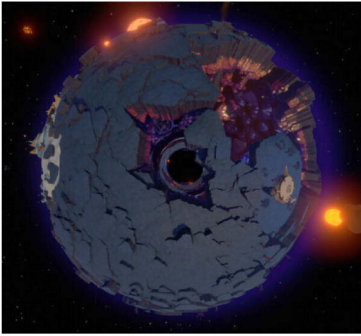


Fig. 21. Planeta Hondonada Frágil, del videojuego *Outer Wilds*.

tratado que toda la zona de la casa contenga esa importancia a nivel visual, igual que sería un videojuego real en el que una de las maneras de guiar al jugador es colocar más elementos en los lugares donde hay algo que hacer y dejar extensiones grandes y menos importantes con menos detalle. Un claro ejemplo de esto puede verse en Hondonada Frágil, uno de los planetas del videojuego que hemos mencionado, en el que hay una gran superficie de placas terrestres sin mucho detalle, y lugares importantes con mayor cantidad de elementos y riqueza visual (Fig. 21).

3.2. PRODUCCIÓN

A continuación, se explica con más detalle este flujo de trabajo dividido en diferentes etapas, así como los softwares que han sido necesarios para el desarrollo del proyecto y la explicación de cómo se ha construido cada elemento.

3.2.1 Modelado

Constituye la siguiente fase después de tener bien definida toda la preproducción. Para modelar todos los elementos necesarios de la escena (salvo el terreno propiamente dicho) se han utilizado dos programas distintos: Blender, debido a su versatilidad y accesibilidad, y Zbrush, por su potencia y su capacidad para manejar mallas con muchos polígonos correctamente.

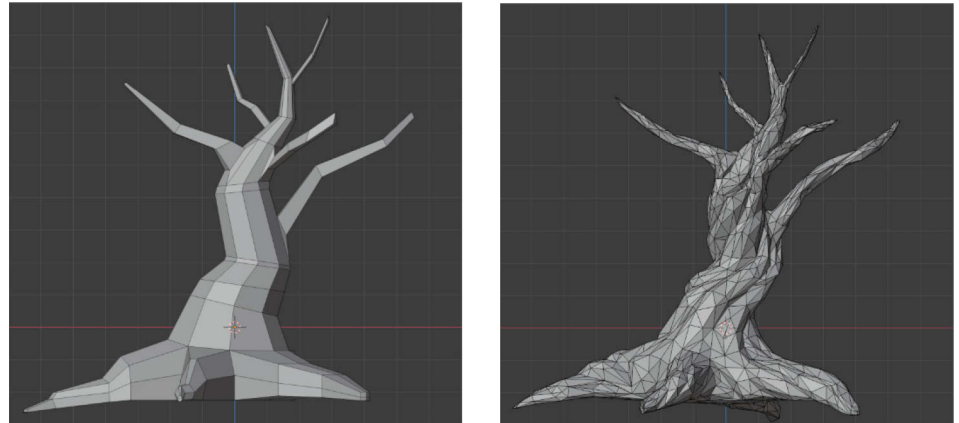
El proceso de modelado de cada *prop* ha seguido un flujo de trabajo más o menos similar, aunque se han contemplado variaciones según las necesidades del elemento en sí. Además, podemos dividir esta fase en tres etapas que se explican a continuación:

3.2.1.1 Blockout y Low Poly

Se comienza modelando la “base” o *blockout* del *prop*, prestando especial atención a la forma general, así como a la silueta del objeto. En esa fase, se busca una malla con una topología limpia y sencilla, cuyo número de polígonos es bastante bajo, sobre la que trabajar en todas las fases posteriores (Fig.22).

Gracias a las herramientas propias del modo edición de Blender, y partiendo del *blockout* mencionado, se realiza el modelado *lowpoly* (Fig.23). Algunas de estas herramientas utilizadas son *extrude*, *inset faces*, *loop cut*, o *knife*, así como diferentes modificadores, como *mirror* para objetos simétricos.

Fig. 22 y 23. *Blockout* del árbol en Blender. Modelado *lowpoly* con triangulación del árbol en Blender.



En esta parte ha sido importante revisar cada malla poligonal antes de pasar al siguiente paso, evitando caras de más de cuatro lados y realizando las comprobaciones habituales, como la de que la escala de cada malla no sea negativa o que todas las normales apunten hacia el exterior de la malla.

Además, se puede diferenciar entre *props* “simples” y “complejos”. En los primeros, como por ejemplo las diferentes tuberías, solo existe una malla poligonal al tratarse de un elemento sencillo con relativamente poco detalle. En los segundos, como por ejemplo el sofá que hay en el exterior, ha sido necesario separar el objeto en varias mallas poligonales, con lo que resulta especialmente importante optimizar el número de polígonos y limpiar cada malla. En este tipo de *props*, se ha modelado cada elemento por separado y posteriormente se han juntado todos con su material correspondiente.

También se puede diferenciar la vegetación según su grado de complejidad. La hierba (Fig. 24) y las flores (Fig. 25) están formadas por una malla simple ligeramente subdivididas para luego poder otorgarles movimiento en motor de juegos, y los árboles están formados por una malla más compleja que forma el tronco, y por un sistema de partículas (Fig. 27) que distribuye un plano subdividido (Fig. 26) para formar las hojas de la copa con la forma correspondiente.

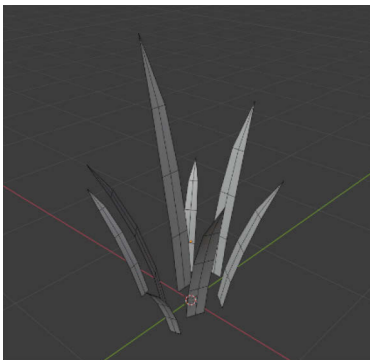


Fig. 24. Modelado *lowpoly* de la hierba en Blender.

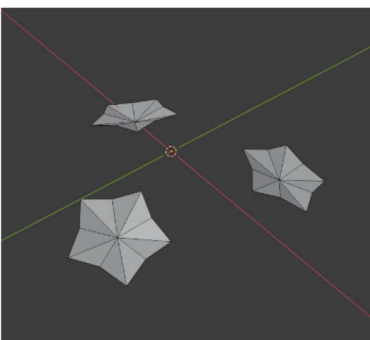


Fig. 25. Modelado *lowpoly* de las flores en Blender.

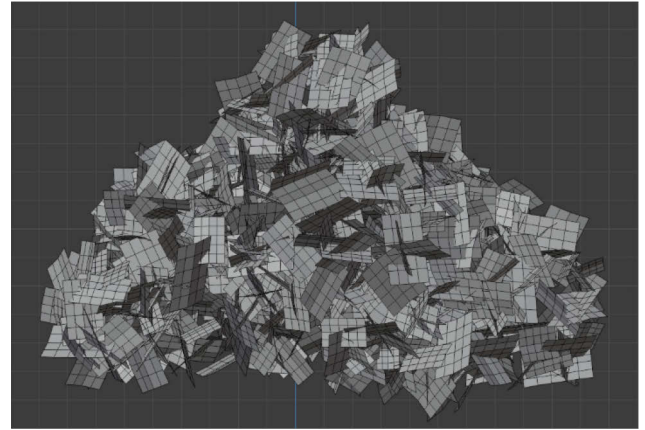
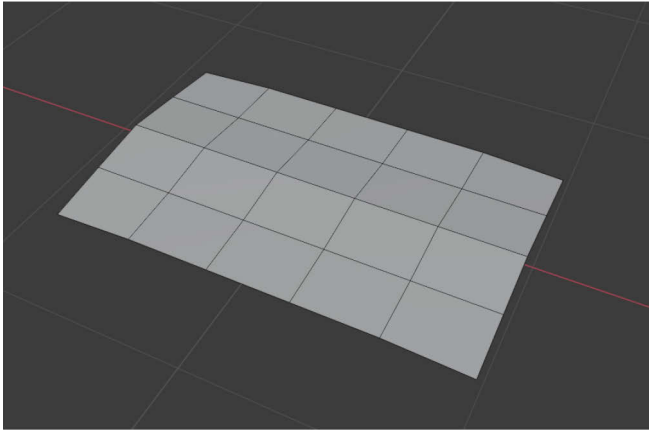


Fig. 26 Y 27. Plano subdividido de las hojas en Blender. Copa del árbol mediante sistema de partículas en Blender.

El último paso antes de pasar al siguiente software es exportar cada elemento. Así, exportamos un archivo *fbx* (*Filmbox*), lo que nos permite almacenar toda la información necesaria y, en caso de *props* con varios elementos, contener varias mallas dentro de un solo archivo.

3.2.1.2 Highpoly

Esta fase se desarrolla en ZBrush y consiste en dar detalle a cada *prop* partiendo de la malla poligonal base. Se importa el archivo *fbx* exportado desde Blender, quedando así dividido en varias *SubTools*, en caso de ser un *prop* con varios elementos (Fig. 31 y 32), o en una sola, si es un *prop* sencillo. De toda la vegetación de la escena, solo el tronco de los árboles (Fig. 28) pasa por la fase *highpoly*. Las hojas están compuestas por planos con tres o cuatro subdivisiones y se texturizan dentro de Unreal Engine (explicado en el apartado 3.2.3.1).

Para cada elemento, lo primero que hacemos es utilizar la herramienta *ZRemesher*, para obtener una topología organizada en *quads* (Fig. 29). De este modo, las posteriores subdivisiones que se realizan a la malla no causan problema a la hora de esculpir y el resultado tiene más detalle y uniformidad.

Una vez la malla está subdividida y tenemos la cantidad de polígonos que necesitamos, podemos comenzar a esculpir cada elemento. Los pinceles que he utilizado han sido los básicos como el *TrimDynamic*, para abolladuras y el daño en los bordes de los objetos, el *Move*, para realizar ajustes de posición, o el *ClayBuildup*, para añadir relieve y detalles. Además, se ha utilizado el pack



Fig. 28. Modelado *highpoly* del árbol importado en Blender.

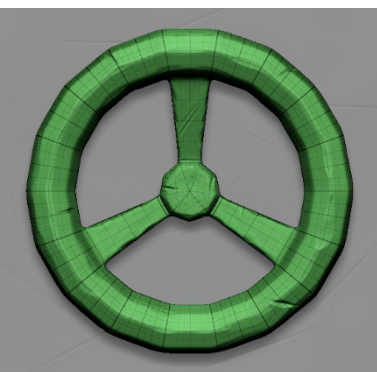


Fig. 29. Manivela de la puerta subdividida con *ZRemesher* en ZBrush.



Fig. 30. Modelado *highpoly* de un sector de tubería en ZBrush.

“Orb”⁹ de Michael Vicente, artista Senior en Blizzard Entertainment. Este pack incluye algunos pinceles que han resultado muy útiles a la hora de esculpir diferentes detalles, como el *Rock_Detail*, para darle detalles en la superficie a las rocas o al ladrillo del sofá, o el *Orb_Cracks*, para las diferentes fracturas y rayaduras de los elementos tanto metálicos como de otros materiales. También ha sido útil el uso de las máscaras para el esculpido, así como el de los diferentes pinceles de *Smooth* para lograr una superficie suave en las partes que lo necesitan.

En esta fase hay que decidir qué detalles o elementos añadir mediante esculpido como tal, y cuáles dejar para más tarde en el apartado de texturizado. Por ejemplo, en las diferentes partes de la tubería, los arañazos o las abolladuras en los bordes del metal se esculpen en ZBrush, pero las manchas y oquedades que tiene el material se añaden posteriormente en Substance Painter. De esta manera, resulta mucho más sencillo de modificar la posición o forma de dichos detalles y se hace viendo una versión prácticamente final del *prop*, con lo que el control es mayor y es más fácil asegurarse de que el resultado es el que se busca.



Fig. 31 y 32. Modelado *highpoly* de la puerta en ZBrush. Modelado *highpoly* del sofá en ZBrush.

⁹. VICENTE, MICHAEL. *Orb Brush Pack* (2015). Recuperado de <https://www.artstation.com/artwork/9kwVo>

Una vez acabado el esculpido de cada malla poligonal, se utiliza la herramienta *Decimation Master*, que reduce significativamente la cantidad de polígonos de cada objeto mediante teselación triangular, permitiendo distribuir la densidad de polígonos a las áreas donde es necesario representar detalle.

El último paso es exportar las mallas decimadas, utilizando de nuevo el formato *fbx*, teniendo así dos versiones del mismo objeto: una *lowpoly* que nombramos siguiendo el código “*nombreProp_lp*” y otra *highpoly* con todos los detalles que nombramos “*nombreProp_hp*”.

3.2.2 UVs, baking y texturizado

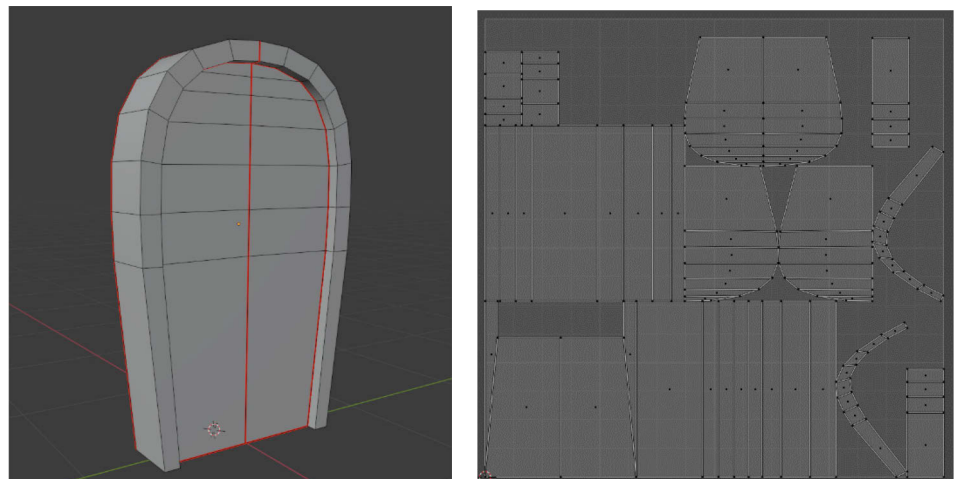
Antes del texturizado, es necesario generar el mapa de coordenadas UV de cada *prop*, desplegando las mallas sobre el espacio bidimensional de las texturas. Dichas texturas son de dos tipos: las generadas por proyección de la malla *highpoly* sobre la *lowpoly* (*baking*) y las elaboradas separadamente. Ambas se han realizado en *Substance Painter*.

En el caso de las paredes de la casa y el terreno, se han creado materiales específicos para albergar las texturas utilizando *Substance Designer*.

3.2.2.1 UVs y baking

El primer paso para generar las UVs consiste en determinar las costuras (Fig. 33) por las que queremos que se realice el despliegue correspondiente, e ir modificándolas hasta conseguir el resultado deseado (Fig. 34). Esta fase se hace sobre el modelo *lowpoly* de cada *prop* dentro del editor de UVs de Blender.

Fig. 33 y 34. Costuras del modelo *lowpoly* de la puerta en Blender. UVs del modelo *lowpoly* de la puerta desplegadas en Blender.



Una vez realizado el despliegue, y habiendo comprobado que el punto de anclaje, escala, rotación y posición del objeto es la correcta, se puede exportar de nuevo el modelo *lowpoly*.

El siguiente paso se denomina *baking*, y consiste en la proyección de los detalles de la malla *highpoly* sobre la *lowpoly* de cada objeto, en el software Substance Painter. Gracias a esta técnica, se generan una serie de mapas bidimensionales que guardan todos los detalles de la malla detallada, lo que permite texturizar la malla con pocos polígonos y mantener todo lo que se ha esculpido anteriormente.

Fig. 35. *Bake* del árbol en Substance Painter.



Para algunos de los *props* del proyecto, como por ejemplo el neumático, se ha utilizado una funcionalidad interna de Substance Painter llamada *Auto UV-Unwrap*, que realiza el despliegue de las UVs de manera automática y que ha dado buenos resultados debido a la poca complejidad del elemento. No obstante, para los *props* más elaborados como el sofá se ha realizado el despliegue manual para tener más control sobre el proceso.

3.2.2.2. Texturizado de *props*

El texturizado de los *props* se ha realizado en el mismo software en el que se ha realizado el *baking*. Dentro de Substance Painter, y teniendo ya la malla *lowpoly* con todos los detalles de la malla *highpoly*, se pasa a darle color, textura y otros detalles que se habían dejado para esta parte. El proceso que se sigue para cada elemento de la escena varía dependiendo del material y la naturaleza del objeto en sí. No obstante, y para lograr una estilización similar y un resultado visual coherente, se han seguido, de manera general, una serie de directrices a la hora de texturizar cada objeto.

Se comienza aplicando los colores planos de cada material (Fig. 36), como el metal, madera, cristal, caucho, etc. Una vez se tiene la base, en la que se aplica únicamente el color, se selecciona la opción que permite modificar el *roughness*, (rugosidad, valor que indica el detalle de la difusión de la luz sobre una superficie). Al estar desarrollando un entorno con cierto grado de estilización, se ha mantenido esta característica con valores relativamente altos (poca difusión), para poder añadir posteriormente detalles de brillos pintados directamente sobre el color. Para añadir riqueza visual al color base, se aplica un *3D Linear Gradient* (Fig. 37) lo que permite variar un poco los colores desde una parte de la malla hasta otra, evitando que todo el objeto tenga exactamente el mismo tono. Es importante que las luces y sombras del *prop* estén correctamente aplicadas, por lo que se aplica una capa de oclusión ambiental (Fig. 36), gracias al mapa 2D que se ha obtenido haciendo el *baking*, y otra capa por encima con el brillo de los bordes (Fig. 40).

Fig. 36 y 37. Árbol: *base color* y *ambient occlusion*. Árbol: *3D linear gradient*.



Por último, se añade variación de las texturas en los materiales (Fig. 38 y 39), dependiendo de si es madera, metal... y se ajustan los niveles de *roughness* para los diferentes elementos. Las capas de relleno y las máscaras inteligentes resultan especialmente útiles para añadir todos los detalles necesarios y, si es necesario añadir más variaciones, se pueden hacer a mano (Fig. 39) sobre el modelo final.

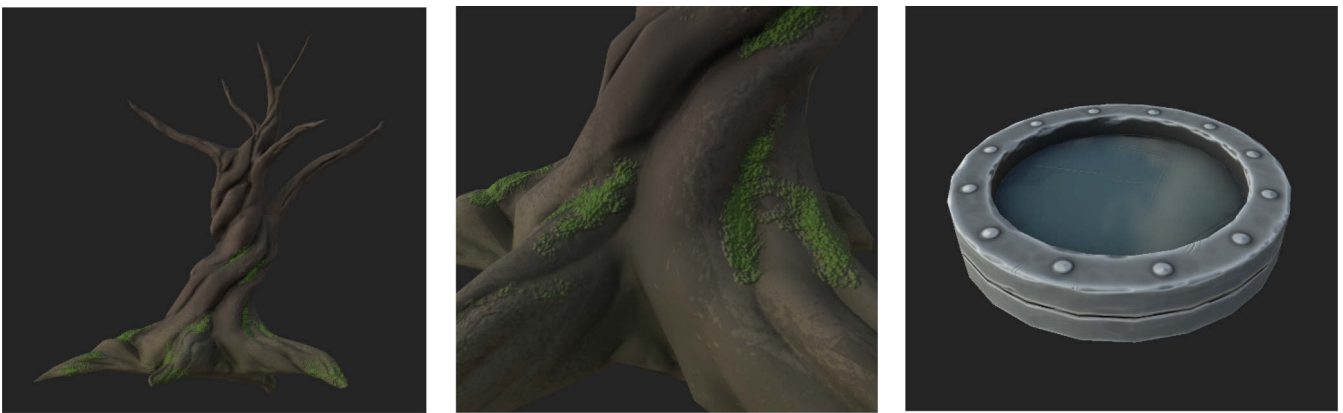


Fig. 38, 39 y 40. Árbol: final.
Árbol: detalles finales.
Ventana: detalles y final.

Una vez correctamente texturizado, se exportan los tres mapas de textura bidimensionales que importaremos más tarde en el motor de videojuegos: *base color* (Fig. 41), *normal* (Fig. 42) y *ambient occlusion, roughness y metallic* (Fig. 43).

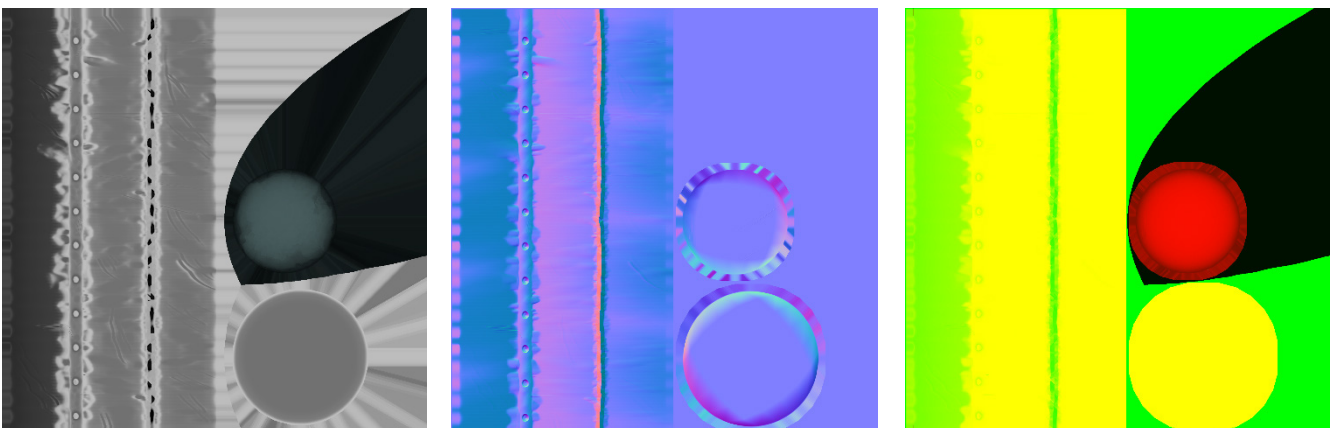


Fig. 41, 42 y 43. Mapa de textura de la ventana (*base color*). Mapa de textura de la ventana (*normal*). Mapa de textura de la ventana (*ambient occlusion, roughness y metallic*).

3.2.2.3. Materiales de arquitectura y terreno

El otro tipo de texturizado ha sido realizado en el software Substance Designer, que permite la creación de texturas *seamless* (materiales), es decir, que se pueden utilizar para repetirlas en una matriz sin cortes. Este tipo de texturas se diseñan principalmente para superficies relativamente grandes, como podría ser el suelo o la pared, o incluso el terreno natural con diferentes tipos de material. En este proyecto se han utilizado dichos materiales para el terreno (tierra del camino y del lago, y base para la zona de la hierba) y para la pared exterior de la casa, con sus diferentes variaciones de color.

Substance Designer permite crear texturas de una alta calidad con mucho control sobre los diferentes elementos y detalles, todo mediante un sistema de nodos interconectados (Fig. 44). De este modo, el proceso que se ha seguido para la creación de las texturas ha comenzado siempre por la creación de los *heightmaps*, trabajando así desde el principio en blanco y negro y prestando solamente atención a la profundidad de los elementos y al relieve de la textura.

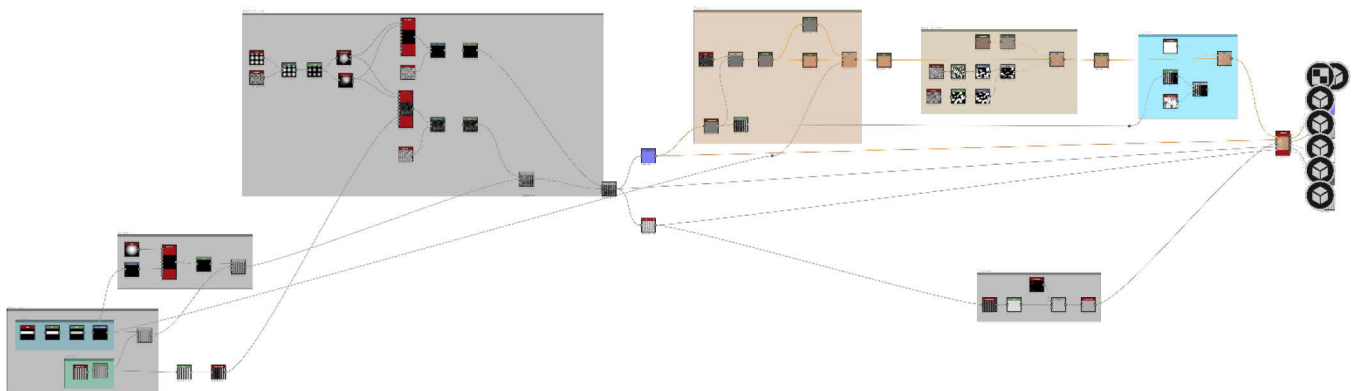


Fig. 44. Nodos correspondientes al material de la pared de la casa.

Algunos de los nodos que se han utilizado frecuentemente han sido los de *Blend* (para mezclar o eliminar partes, así como variaciones en la textura), *Levels* (para ajustar los niveles de todos los nodos), *Tile Sampler* (para repartir diferentes formas, abolladuras, arañazos...), o *Warp* (para deformar siluetas y añadir variación).

Una vez finalizada esa parte, se pasa a ajustar los valores de *roughness* (al igual que cuando se texturizan objetos individualmente), buscando valores altos que ofrezcan un acabado más estilizado. Por último, se aplica al color gracias al

nodo *Gradient Map*, y se ajusta con el de *HSL (Hue, Saturation and Lightness)*. Con el fin de obtener un resultado cohesionado con el resto de objetos, también se aplican el nodo de oclusión ambiental, y se añaden detalles de brillos en los bordes gracias a otro *Gradient Map* aplicado solo a ciertos lugares.

Por último, y habiendo finalizado ya las texturas, se exportan en diferentes mapas bidimensionales RGB, de 8 bits por canal (lo cual resulta importante para poder importarlos correctamente desde el motor de videojuegos), obteniendo así cinco imágenes distintas: *base color* (Fig. 45), *roughness* (Fig. 46), *normal* (Fig. 47), *ambient occlusion* (Fig. 48) y *height* (Fig. 49).

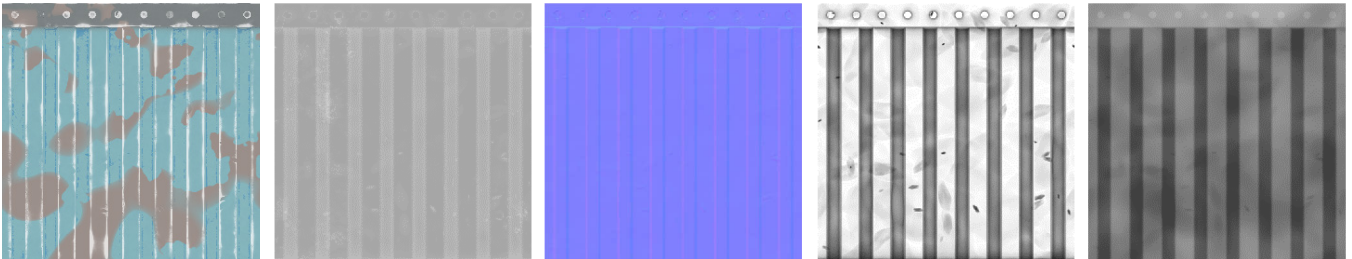


Fig. 45, 46, 47, 48 y 49.
Material de la pared de la casa: *base color*, *roughness*, *normal*, *ambient occlusion* y *height*.

3.2.3 Implementación en el motor

3.2.3.1 Terreno

- **Suelo**

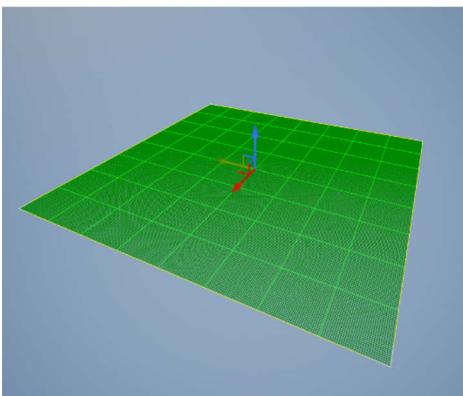


Fig. 50. Herramienta “terreno” de Unreal Engine 5.

La superficie sobre la que se construye el resto del escenario ha sido modelada íntegramente dentro del motor de videojuegos Unreal Engine 5. Para ello, se ha utilizado la herramienta “terreno” del motor (Fig. 50), que permite la creación de un mapa de altura, así como su modificación. Al tratarse de una zona relativamente pequeña y delimitada por elementos como el lago o los árboles del bosque, se ha optado por la creación de un terreno de 63x63 *Quads*, al que se han añadido y eliminado secciones posteriormente para ampliar o reducir el tamaño según lo necesario para la composición.

Una vez creado, se ha utilizado la herramienta de esculpido para modificar el terreno y definir los diferentes accidentes geográficos

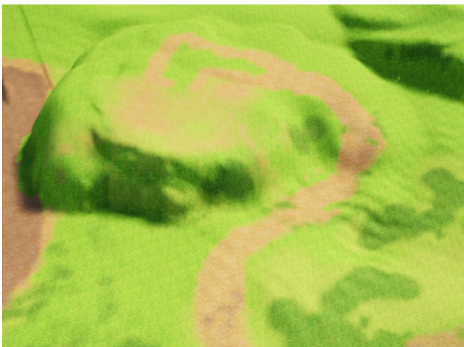
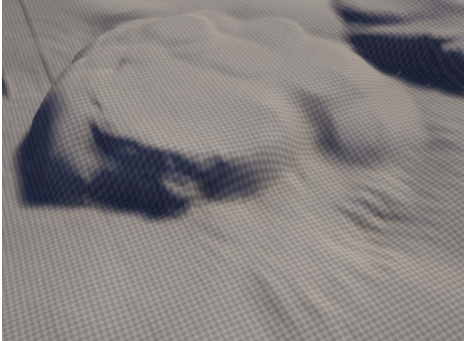


Fig. 51 y 52. Superficie en Unreal Engine 5.
Superficie pintada en Unreal Engine 5.



Fig. 53. Vegetación de la escena en Unreal Engine 5.

que existen, como la pequeña elevación sobre la que se asienta la casa, o toda la zona donde se encuentra el lago (Fig. 51).

Por último, utilizando la herramienta de pintura, se ha procedido a aplicar las texturas *seamless* creadas en Substance Designer (apartado 3.2.3.1.), definiendo así en qué lugar se sitúa el camino que lleva a la casa, la hierba, y la tierra que forma el fondo y la orilla del lago. Para lograr el efecto deseado a la hora de pintar el terreno, se ha creado un *Master Material* que contenía las tres texturas principales y sus variaciones. Así, gracias a las funciones de los materiales de Unreal Engine, y mediante un nodo de “multiplicar”, se consigue crear variaciones más claras y más oscuras de las texturas dentro del propio motor. Por último, utilizando el nodo *Layer Blend*, se mezclan todas las variaciones para poder aplicarlas al terreno libremente (Fig. 52).

- **Vegetación**

Ha sido implementada en la superficie referida anteriormente utilizando la herramienta “follaje” del motor, que permite “pintar” los distintos tipos de vegetación sobre el terreno. Para este proyecto, se han definido dos tipos distintos de hierba, unas flores con tres variaciones de color, y los árboles.

Para lograr pequeños cambios en el tono de la hierba se ha utilizado una “*Runtime Virtual Texture*”, que obtiene la información del color de la superficie de todo el escenario y, posteriormente, se puede implementar dentro del material de la hierba. Así, se combina con el color original de la hierba mediante un nodo “*Linear Interpolate*”, logrando que el color verde de la vegetación cambie y se adapte al del suelo que tiene debajo (Fig. 53). Gracias a los diferentes tonos del suelo, como la hierba clara u oscura y la tierra del camino, se obtiene vegetación entonada a verde, marrón, etc, logrando el efecto buscado.



Fig. 54. Imagen con información alfa de la hoja.

La textura de las hojas de los árboles consiste en una imagen RGBA con fondo transparente con la forma de las hojas en color blanco. Así, se utiliza un valor alfa (Fig. 54) y se aplica dicha imagen 2D a los planos subdivididos de la copa. Mediante un material dentro de Unreal Engine, se le da color (Fig. 55) y el movimiento del viento (este último explicado en el apartado 3.2.3.3). Para finalizar, se aplican los materiales del tronco y las hojas al *prop* del árbol (Fig. 56). Las flores tienen aplicado un material con el mismo sistema para la animación del viento, y un color plano conectado a *emissive color*, que les dota de una sutil iluminación propia.

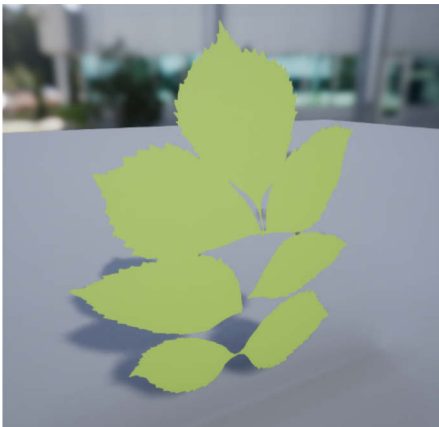


Fig. 55 y 56. Plano de la hoja con material de Unreal Engine 5.
Árbol final en Unreal Engine 5.



- **Lago**

Una parte importante de la composición es el lago en cuya orilla se asienta la casa del hurón, el sapo y la foca. Para su desarrollo se ha creado un material dentro de Unreal Engine que simula la apariencia del agua en movimiento sin necesidad de un sistema de

físicas. Así, se aplica el material a un único plano, obteniendo tanto la superficie del lago como el efecto visual para las profundidades.

Dentro del material del agua se han creado diferentes grupos de nodos atendiendo a los diferentes aspectos que se buscaban conseguir en el lago:

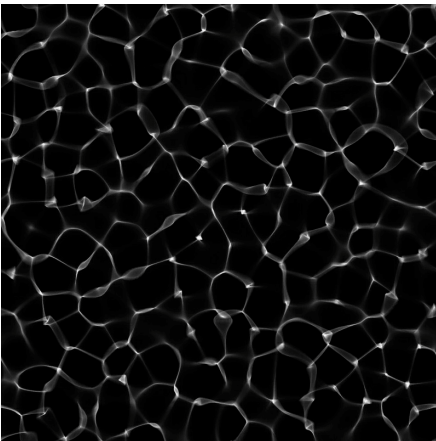


Fig. 57. Textura con valor alfa para los reflejos del agua.

Para los reflejos del sol sobre la superficie del agua se ha utilizado una textura con valor alfa (Fig. 57) duplicada que se mueve horizontal y verticalmente gracias al nodo *"panner"*. Exponiendo parámetros como la intensidad o el contraste, se puede controlar el valor de estos reflejos fácilmente desde el propio inspector de elementos.

La profundidad del agua también se puede controlar desde fuera del material, y es posible modificar la opacidad del fondo del lago, el color, etc. Para este apartado se han utilizado diferentes nodos como *"Scene Depth"* o *"Pixel Depth"*, *"Fresnel"*, o *"Depth Fade"*, y combinados gracias a una serie de *"Subtract"*, *"Add"* y *"Multiply"*, nos permiten el control buscado sobre cómo se ve el material.

Para la parte de la orilla del lago, o de cualquier material que esté dentro del agua y que sobresalga, se ha buscado crear un efecto de espuma que se mueve hacia el interior. Algunos de los nodos utilizados han sido *"Absolut Global Position"*, *"Distance Field Gradient"*, y *"Vector to Radial Value"*. Al igual que en otros apartados, se han expuesto parámetros como la velocidad de la espuma o el contraste para poder modificarse fácilmente viendo el resultado en tiempo real.

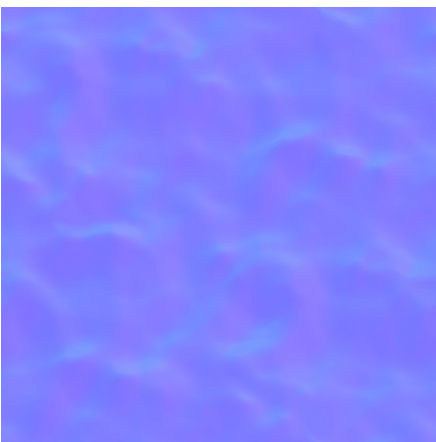


Fig. 58. Textura de normales para el agua.

Por último, ha sido importante añadir un mapa de normales (Fig. 58) que gracias a un nodo *"Panner"* y a otro *"Time"* le da la forma deseada a la superficie del agua. Los diferentes parámetros que se pueden modificar en este apartado son la velocidad tanto en el eje X como en el Y, o la intensidad de las normales.

El resultado final, pues, incorpora todos estos detalles siendo en realidad un único plano dentro de Unreal Engine 5 (Fig. 59).



Fig. 59. Material del agua del lago en Unreal Engine 5.

- **Montañas**

En el fondo de la composición se han colocado una serie de montañas que sirven para compensar la composición y para completar un poco más el entorno en el que se sitúa el bosque y el lago (Fig. 60). Estas montañas han sido modeladas en Blender e importadas como objeto posteriormente en el motor. Para las texturas, se han utilizado las mismas que se desarrollaron para las rocas pequeñas de la imagen, tratando así de buscar una cierta coherencia visual.

Fig. 60. Escena final con las montañas en el fondo.



3.2.3.2 Props

Antes de colocar en la escena cada *prop*, hay que importar correctamente tanto el modelado *lowpoly* del elemento en sí (apartado 3.2.2.1), como las texturas creadas en Substance Painter (apartado 3.2.3.1) y aplicarlas mediante un material de Unreal Engine. Para ello, se importa el modelado con pocos polígonos que se ha creado en Blender anteriormente, y se transfieren todos los detalles que se han esculpido en Zbrush gracias al material.

A la hora de importar las texturas, se tiene en cuenta que se han exportado en tres imágenes diferentes: “*base color*”, “*occlusion, roughness and metallic*” y “*normal*”. La primera se importa con el tipo de muestreado “*color*”, y se conecta en el material con su salida RGB. La segunda, que contiene información sobre tres parámetros distintos, se importa como “*color lineal*” y se conecta su nodo R con el de “*ambient occlusion*”, su nodo G con el de “*roughness*” y su nodo B con el de “*metallic*” (Fig. 61).

Una vez conectados los tres mapas de texturas al material, se puede aplicar este sobre el *prop*, manteniendo el detalle y sin aumentar el número de polígonos. En el caso de algún *prop* más complejo como puede ser el sofá o la puerta, el flujo de trabajo es el mismo, pero aumentando el número de materiales, ya que al tratarse de un *prop* formado por varios elementos, se crea un material para cada uno de ellos.

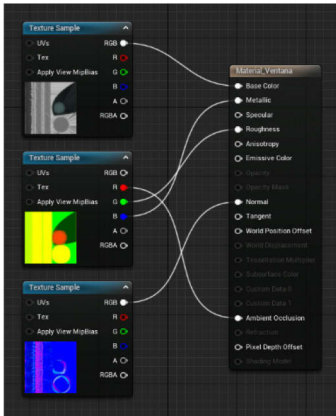


Fig. 61. Material de la ventana con todos los nodos conectados.

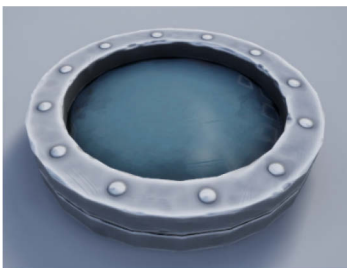


Fig. 62. Ventana con material final aplicado.



Fig. 63 y 64. Puerta con materiales finales aplicados.
Sofá con materiales finales aplicados.

En el caso de las paredes de chapa, se decidió probar la capacidad de la pintura por vértice (*Vertex Painting*) del motor, creando un material con variaciones cromáticas de la textura que se interpolan al aplicarlo al objeto. (Fig. 65). Si bien el procedimiento obliga a subdividir la malla optimizada, la cuenta de polígonos resultante no es excesiva para la configuración de la escena y las capacidades de optimización del motor, especialmente *Nanite*¹⁰, que a partir de un formato propio de malla, permite manejar mallas densas y en el *rendering* trabaja únicamente sobre el detalle que puede ser percibido a escala de píxel.

Fig. 65. “Vertex painting” aplicado a las paredes de la casa.



3.2.3.3 Animaciones

- *Físicas*

Algunas de las animaciones que podemos ver en la escena han sido implementadas mediante el sistema de físicas de Unreal Engine. Este sistema nos permite la creación de animaciones a partir de la simulación en el motor de fuerzas naturales como la gravedad o el viento.

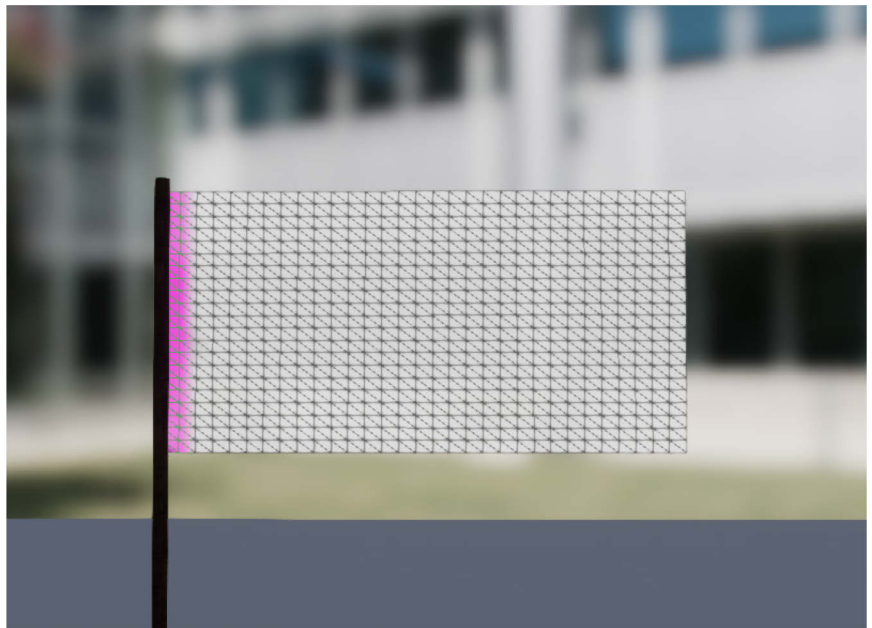
El columpio que hay al lado de la casa ha sido animado utilizando dicho sistema de físicas, y está compuesto por tres elementos. Por un lado, el neumático y por otro la cuerda, unidos mediante un “*Physics*

¹⁰ Epic Games. (2021). Recuperado de <https://docs.unrealengine.com/5.0/en-US/RenderingFeatures/Nanite/>

Constraint Actor". La parte de arriba de la cuerda está anclada en su posición inicial, y el neumático se encuentra ligeramente desplazado hacia atrás y hacia arriba, de modo que la gravedad hace funcionar la animación de balanceo cuando se simula el juego dentro del motor.

También utiliza el sistema de físicas la bandera que vemos en la casa. Esta tiene asociado un *"Skeletal Mesh Component"* que nos permite asignar valores a las distintas partes de la malla, indicando así cuánto se verá afectada por los fenómenos físicos, mediante la técnica *"Cloth Painting"* (Fig. 66). Así, el efecto de la bandera al moverse es una animación física producida por el viento de la escena.

Fig. 66. "Cloth painting" de la bandera.



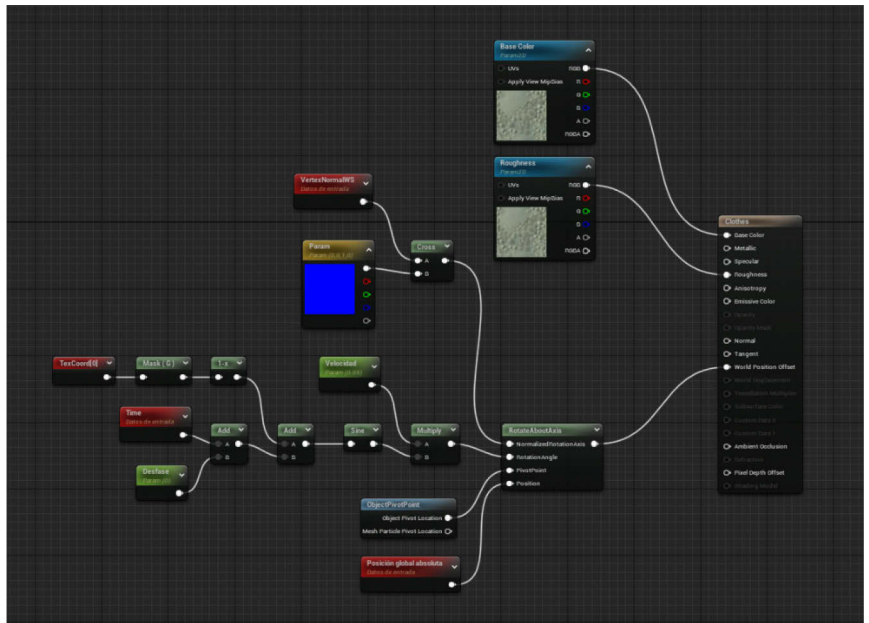
- **Animaciones de material**

Las demás animaciones que vemos en la escena final han sido llevadas a cabo mediante diferentes nodos dentro de los materiales asociados a los distintos elementos.

La animación de la ropa que hay en el tendedero está basada en el nodo *"Rotate About Axis"* que permite rotar al objeto al que está asociado el material siguiendo una serie de parámetros. Este nodo (Fig. 67) permite fijar un punto de anclaje, una posición inicial, un ángulo de rotación, y cómo rota. Para la ropa, se ha elegido una onda sinusoidal

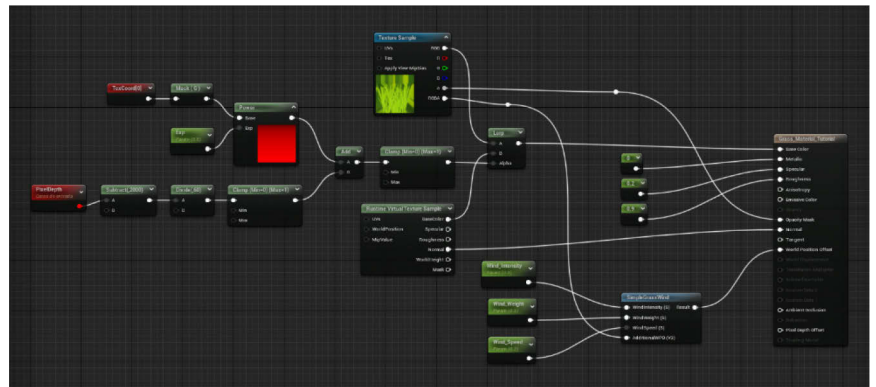
ligeramente pronunciada que imita el efecto del viento sobre la ropa. Al realizar esta animación de este modo, y no mediante las físicas del motor, logramos mantener el movimiento que queremos sin que la ropa se deforme y se pierda la silueta que resulta crucial para una correcta visibilidad desde larga distancia.

Fig. 67. Nodos del material de la ropa.



La otra animación realizada dentro del propio material ha sido la de las hojas de los árboles, la de la hierba y la de las flores. Ambas están basadas en el mismo nodo, “Simple Grass Wind”, (Fig. 68) que permite modificar diferentes parámetros como la intensidad, fuerza o velocidad del viento.

Fig. 68. Nodos del material de la hierba.



- **Sistemas de partículas**

Para la animación del humo de la chimenea se ha utilizado un sistema de partículas sencillo, compuesto por un único emisor. Este genera una serie de puntos en el espacio y los dirige en una dirección concreta, pero contemplando una ligera variación para simular el efecto que tendría el viento sobre dicho humo. El material que se ha asociado a dicho sistema de partículas cuenta con un nodo *"Particle Color"*, para lograr el color blanco del humo, y otro nodo *"RadialGradientExponential"* que aplicado al parámetro *opacity*, permite difuminar un poco los bordes de las partículas, obteniendo así un mejor resultado.

3.2.3.4 Iluminación y atmósfera

- **Iluminación**

La iluminación del proyecto consiste por un lado en la *"sky light"* que aporta el hecho de tener una *"sky sphere"*, como se explica posteriormente, y por otro lado una *"directional light"*. Esta última simula la luz proveniente del sol y permite hacer cambios rápidamente sobre la intensidad o color de la luz, así como realizar rotaciones en todos los ángulos para simular diferentes momentos del día. La escena principal tiene una iluminación con un ángulo de 45º, evitando así el sol de mediodía y logrando algunas sombras para dar profundidad y riqueza visual al escenario (Fig. 69 y 70).



Fig. 69 y 70. Composición final sin *"directional light"*.
Composición final con *"directional light"*.

- **Atmósfera**

Todo el proyecto se encuentra dentro de una “*sky sphere*” que permite modificar diferentes parámetros como la altura del sol, el color del cielo en los diferentes momentos del día, etc. Además, las nubes se proyectan como una imagen de 360º sobre esta esfera que forma el cielo.

Por otro lado, y para el fondo de la imagen, se ha buscado que las montañas y otros elementos como vegetación añadida no tengan mucho peso visual en la composición. Para ello, se ha utilizado un actor de Unreal Engine llamado “*Exponential Height Fog*” que permite añadir una niebla a los elementos que se encuentran más alejados, simulando así la perspectiva atmosférica que nos interesa (Fig. 71 y 72) y que se ha visto como ejemplo en varios referentes visuales del punto 2.3.



Fig. 71 y 72. Composición final sin
“*Exponential Height Fog*”.
Composición final con
“*Exponential Height Fog*”.

4. CONCLUSIONES

Teniendo en cuenta todos los puntos desarrollados y reflejados en esta memoria, considero que se ha seguido en líneas generales la planificación marcada desde el inicio, lo que ha permitido alcanzar los objetivos iniciales con la calidad prevista. Comenzar a desarrollar un estilo propio se ha convertido en un aspecto muy positivo del desarrollo de este trabajo, y se han incorporado algunos elementos cuya conveniencia ha surgido durante el desarrollo, como diferentes sistemas de partículas o de físicas, permitiendo así dotar a la escena de mayor dinamismo y riqueza visual.

Siendo la preproducción una fase necesaria y fundamental para el desarrollo de un proyecto, el hecho de que este trabajo se base en una preproducción elaborada con anterioridad me ha permitido centrarme en aspectos técnicos y visuales del prototipo. La adaptación de un proyecto bidimensional a uno tridimensional y el cambio de medio han supuesto una oportunidad perfecta para consolidar los conocimientos adquiridos durante el grado y ponerlos en práctica sobre un proyecto hasta conseguir su implementación.

Por tanto, considero que la elaboración de este trabajo de fin de grado conlleva un gran aprendizaje tanto académico como profesional. Ver implementada en un motor de videojuegos una idea que tuvimos hace ya dos años, ha supuesto para mí el punto álgido de esta etapa universitaria, abriéndose ahora la posibilidad de continuar en esta línea y de incorporar este trabajo a mi portfolio, aspecto que me facilitará el proceso para integrarme en un contexto profesional dentro de la industria del videojuego.

5. BIBLIOGRAFÍA

ARTÍCULOS

- Habezai-Fekri, Jasmin. (29 de octubre del 2020). Bird House: Working on a Stylized Landscape in UE4. 80LV. <https://80.lv/articles/bird-house-working-on-a-stylized-landscape-in-ue4/>
- Shirinian, Ara. (26 de enero del 2010). The Uneasy Merging of Narrative and Gameplay. *Game Developer*. <https://www.gamedeveloper.com/design/the-uneasy-merging-of-narrative-and-gameplay>
- Torres Ortiz, Diego. (18 de octubre del 2021). Making a Wardrobe Prop in 3ds Max, Substance Painter & UE5. 80LV. <https://80.lv/articles/making-a-wardrobe-prop-in-3ds-max-substance-painter-ue5/>

PÁGINAS WEB

- Epic Games. (2015). Unreal Engine 4 Documentation. <https://docs.unrealengine.com/4.27/en-US/>
- Epic Games. (2021). Unreal Engine 5 Early Access Documentation <https://docs.unrealengine.com/5.0/en-US/>
- Irrational Games (2007). Bioshock. <https://store.steampowered.com/app/7670/BioShock/>
- Jee, Dong Park (2021). Sunflower House. <https://www.artstation.com/artwork/zOLDR6>
- Mobius Digital Games (2020). Outer Wilds. <https://www.mobiusdigitalgames.com/outer-wilds.html>
- Naughty Dog (2020). The Last Of Us Part II. <https://www.playstation.com/es-es/games/the-last-of-us-part-ii/>
- Riot Games. (2021). Valorant Maps: Breeze. <https://playvalorant.com/en-us/maps/>
- Smith, Thomas. Stylized Station. (2019). The 3D Artist's Coloring Book. <https://courses.stylizedstation.com/p/3dcoloringbook>
- Valve (2007). Portal. <https://store.steampowered.com/app/400/Portal/>
- Vicente, Michael (2015). Orb Brush Pack. <https://www.artstation.com/artwork/9kwVo>
- Zhang, Xuan (2021). Pirate Radio Station. <https://www.artstation.com/artwork/zOgmYQ>

VÍDEOS

- CodeLikeMe (3 de noviembre de 2020). Unreal Stylized Landscape and Grass Materials. [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=IOS0Aufq5Ko&t=1823s>
- Dean Ashford (4 de julio de 2020). UE4 - Tutorial - Stylised Wind Particles (Cascade). [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=6HHHAcaiQkl>
- Dean Ashford (9 de marzo de 2018). UE4 - Tutorial - Physics Constraint Actors. [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=erqu8mGJAYE>
- Markom 3D (5 de enero de 2019). Unreal Engine 4 - Vertex Painting – Tutorial. [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=eeMLA9pwk4g>
- Noclip – Video Game Documentaries (1 de enero de 2020). The Making of Outer Wilds - Documentary. [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=LbY0mBXKKT0>
- Ryan Huber (22 de agosto de 2017). UE4: the basics of Rotate About Axis [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=VwAM1xyn0Bg&t=582s>
- Snaplaunch (26 de octubre de 2020). UE4.26 Simulate a Cloth Flag. [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=fIRIW00WgVI>
- Stylized Station (23 de junio de 2020). Beginners Guide to Creating Overwatch-Style Game Art [3D MODELING/TEXTURING] [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=PVUw5MheypI>
- Upside Down (13 de julio de 2020). Unreal Engine 4 Tutorial - How to Create a Rope with Physics. [Archivo de Vídeo]. Youtube.
<https://www.youtube.com/watch?v=FF0mkxx5Mns>

VIDEOJUEGOS

- 2K Games. (2007). Bioshock (PC) [Videojuego]. Boston, EEUU: Irrational Games.
- Annapurna Interactive. (2019). Outer Wilds (PC). [Videojuego]. Los Ángeles, EEUU: Mobius Digital.
- Sony Interactive Entertainment. (2020). The Last of Us Part II (PS4). [Videojuego]. Santa Mónica, EEUU: Naughty Dog.
- Valve. (2007). Portal (PC) [Videojuego]. EEUU: Valve.

6. ÍNDICE DE FIGURAS

- Fig. 1. Portada del *Sale Folder* de “Pachuki Forest”.
- Fig. 2. Cronograma del proyecto en formato de diagrama de Gantt.
- Fig. 3. Esquema del flujo de trabajo.
- Fig. 4. Captura de pantalla del videojuego *Portal*.
- Fig. 5. Captura de pantalla del videojuego *Bioshock*.
- Fig. 6. Captura de pantalla del videojuego *The Last of Us Parte II*.
- Fig. 7. Render final del tutorial de Diego Torres.
- Fig. 8. Render final del tutorial de Eric García.
- Fig. 9. Render final del proyecto “*Sunflower House*”.
- Fig. 10. Render final del proyecto “*Bird House*”.
- Fig. 11. Render final del proyecto “*Pirate Radio Station*”.
- Fig. 12. Captura de pantalla del mapa *Breeze* del videojuego *Valorant*.
- Fig. 13. Captura de pantalla del mapa *Breeze* del videojuego *Valorant*.
- Fig. 14. Captura de pantalla del mapa *Breeze* del videojuego *Valorant*.
- Fig. 15. Personajes principales de “*Pachuki Forest*”.
- Fig. 16. Escenario final a línea de “*Pachuki Forest*”: vista general.
- Fig. 17. Escenario final a línea de “*Pachuki Forest*”: casa.
- Fig. 18. Composición final de la escena en 3D.
- Fig. 19. Boceto del árbol a lápiz.
- Fig. 20. Composición con regla de los tercios.
- Fig. 21. Planeta Hondonada Frágil, del videojuego *Outer Wilds*.
- Fig. 22. *Blockout* del árbol en Blender.
- Fig. 23. Modelado *lowpoly* con triangulación del árbol en Blender.
- Fig. 24. Modelado *lowpoly* de la hierba en Blender.
- Fig. 25. Modelado *lowpoly* de las flores en Blender.
- Fig. 26. Plano subdividido de las hojas en Blender.
- Fig. 27. Copa del árbol mediante sistema de partículas en Blender.
- Fig. 28. Modelado *highpoly* del árbol importado en Blender.
- Fig. 29. Manivela de la puerta subdividida con *ZRemesher* en ZBrush.
- Fig. 30. Modelado *highpoly* de un sector de tubería en ZBrush.
- Fig. 31. Modelado *highpoly* de la puerta en ZBrush.
- Fig. 32. Modelado *highpoly* del sofá en ZBrush.
- Fig. 33. Costuras del modelo *lowpoly* de la puerta en Blender.
- Fig. 34. UVs del modelo *lowpoly* de la puerta desplegadas en Blender.
- Fig. 35. *Bake* del árbol en Substance Painter.
- Fig. 36. Árbol: *base color* y *ambient occlusion*.
- Fig. 37. Árbol: *3D linear gradient*.
- Fig. 38. Árbol: final.
- Fig. 39. Árbol: detalles finales.
- Fig. 40. Ventana: detalles y final.

Fig. 41. Mapa de textura de la ventana (*base color*)

Fig. 42. Mapa de textura de la ventana (*normal*)

Fig. 43. Mapa de textura de la ventana (*ambient occlusion, roughness y metallic*).

Fig. 44. Nodos correspondientes al material de la casa.

Fig. 45. Material de la pared de la casa: *base color*.

Fig. 46. Material de la pared de la casa: *roughness*.

Fig. 47. Material de la pared de la casa: *normal*.

Fig. 48. Material de la pared de la casa: *ambient occlusion*.

Fig. 49. Material de la pared de la casa: *height*.

Fig. 50. Herramienta "terreno" de Unreal Engine 5.

Fig. 51. Superficie en Unreal Engine 5.

Fig. 52. Superficie pintada en Unreal Engine 5.

Fig. 53. Vegetación de la escena en Unreal Engine 5.

Fig. 54. Imagen con información alfa de la hoja.

Fig. 55. Plano de la hoja con material de Unreal Engine 5.

Fig. 56. Árbol final en Unreal Engine 5.

Fig. 57. Textura con valor alfa para los reflejos del agua.

Fig. 58. Textura de normales para el agua.

Fig. 59. Material del agua del lago en Unreal Engine 5.

Fig. 60. Escena final con las montañas en el fondo.

Fig. 61. Material de la ventana con todos los nodos conectados.

Fig. 62. Ventana con material final aplicado.

Fig. 63. Puerta con materiales finales aplicados.

Fig. 64. Sofá con materiales finales aplicados.

Fig. 65. "Vertex painting" aplicado a las paredes de la casa.

Fig. 66. "Cloth painting" de la bandera.

Fig. 67. Nodos del material de la ropa.

Fig. 68. Nodos del material de la hierba.

Fig. 69. Composición final sin "*directional light*".

Fig. 70. Composición final con "*directional light*".

Fig. 71. Composición final sin "*Exponential Height Fog*".

Fig. 72. Composición final con "*Exponential Height Fog*".

En este anexo se pueden encontrar dos enlaces a los dos vídeos que se han hecho a modo de render de la escena. El primero se centra en una vista general de la escena mientras que el segundo está enfocado en mostrar la casa y sus elementos un poco más de cerca:

<https://www.youtube.com/watch?v=-5xjpXmM0E>

https://www.youtube.com/watch?v=n_xNnmdKIkE

ANEXO II

RENDERS

PRESENTADO POR CARLOS JUAN NAVARRO
TUTOR: FRANCISCO MARTÍ FERRER

FACULTAT DE BELLES ARTS DE SANT CARLES
GRADO EN DISEÑO Y TECNOLOGÍAS CREATIVAS
CURSO 2021-2022



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA
FACULTAT DE BELLES ARTS DE SANT CARLES

RENDERS

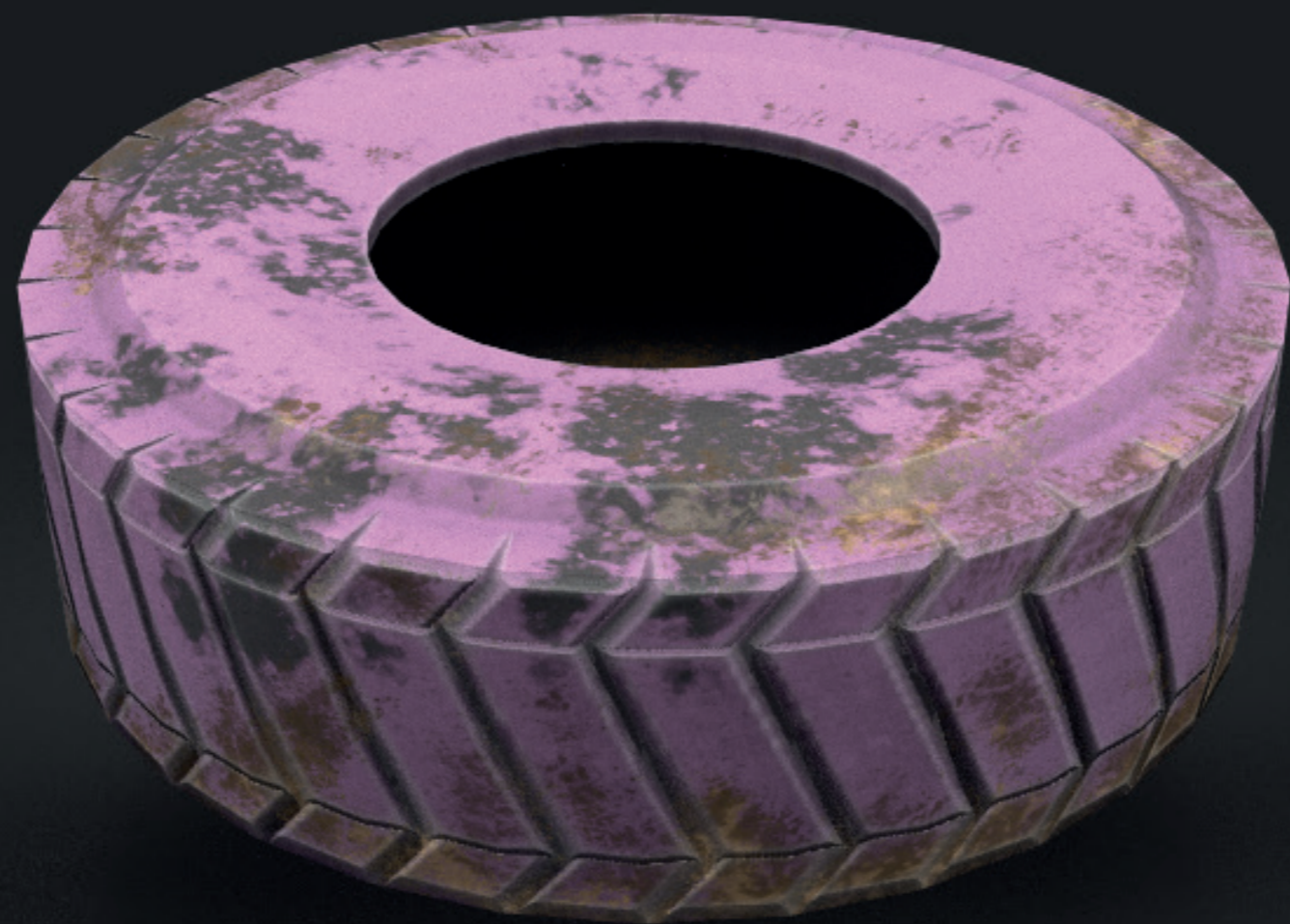
ENTORNO VIDEOJUEGO 3D

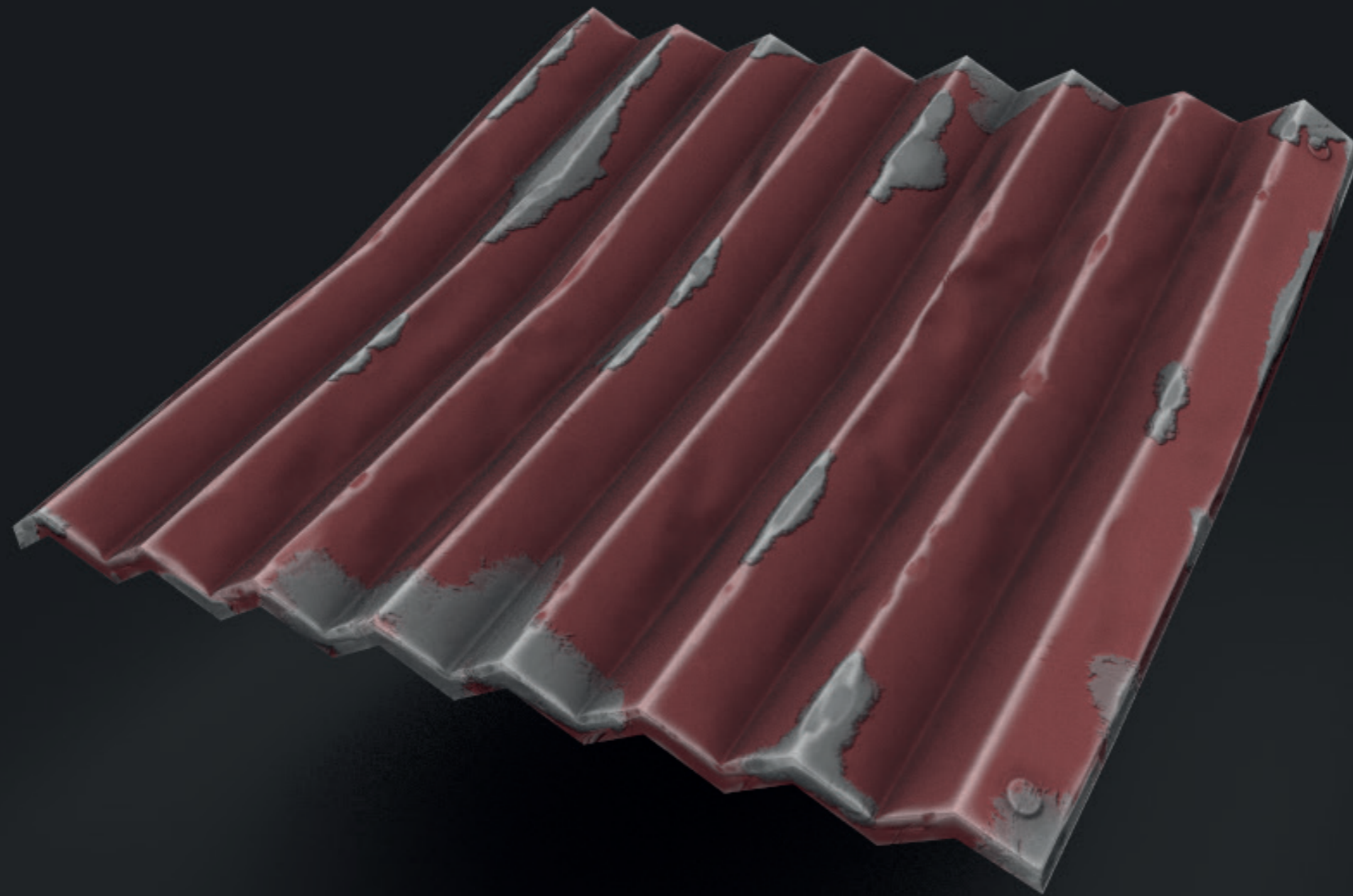
PROPS











PROPS Árbol Barril Neumáticos **Plancha** Puerta Rocas Sofá Tuberías Ventana Viga



PROPS Árbol Barril Neumáticos Plancha **Puerta** Rocas Sofá Tuberías Ventana Viga









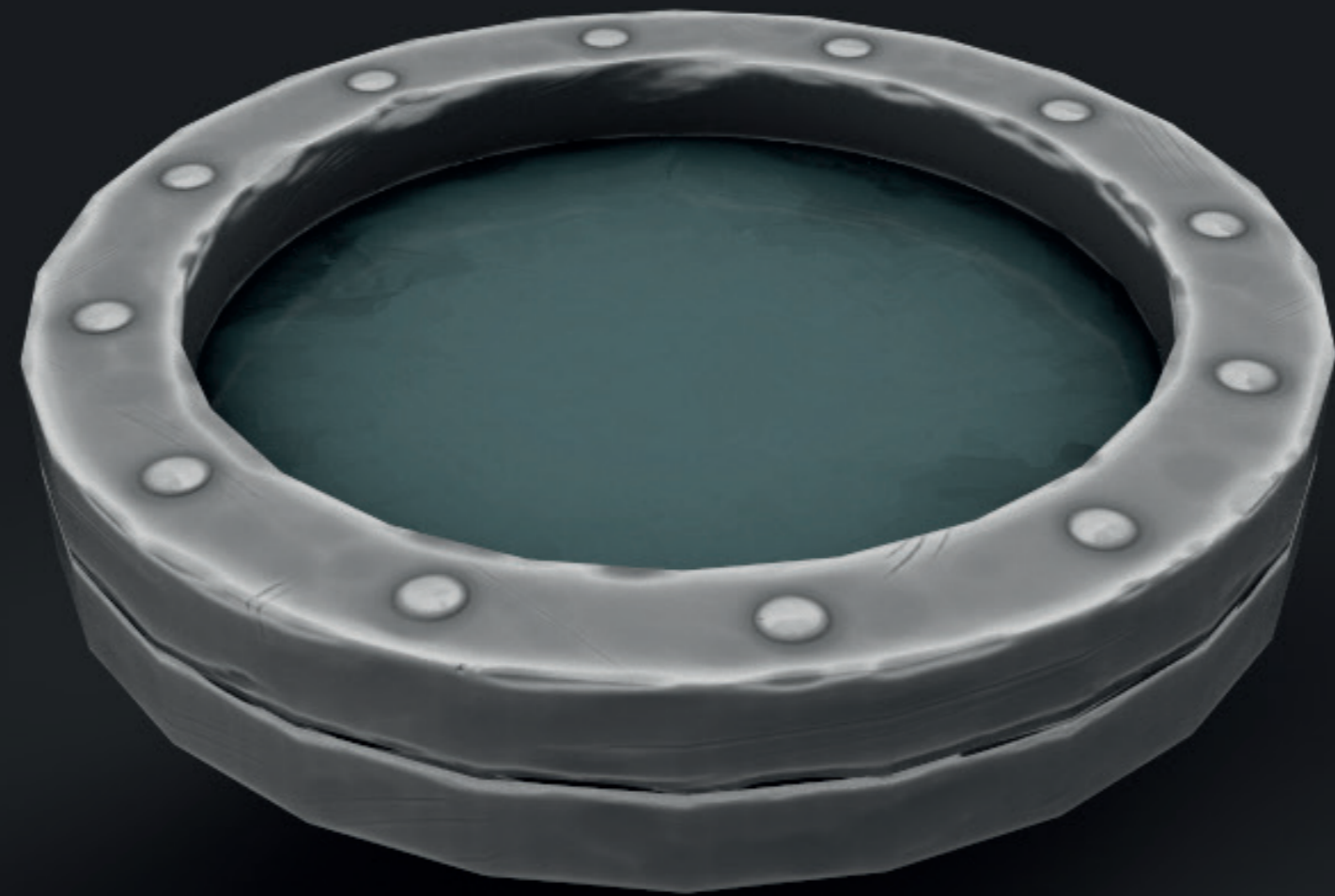








PROPS Árbol Barril Neumáticos Plancha Puerta Rocas Sofá **Tuberías** Ventana Viga





MATERIALES

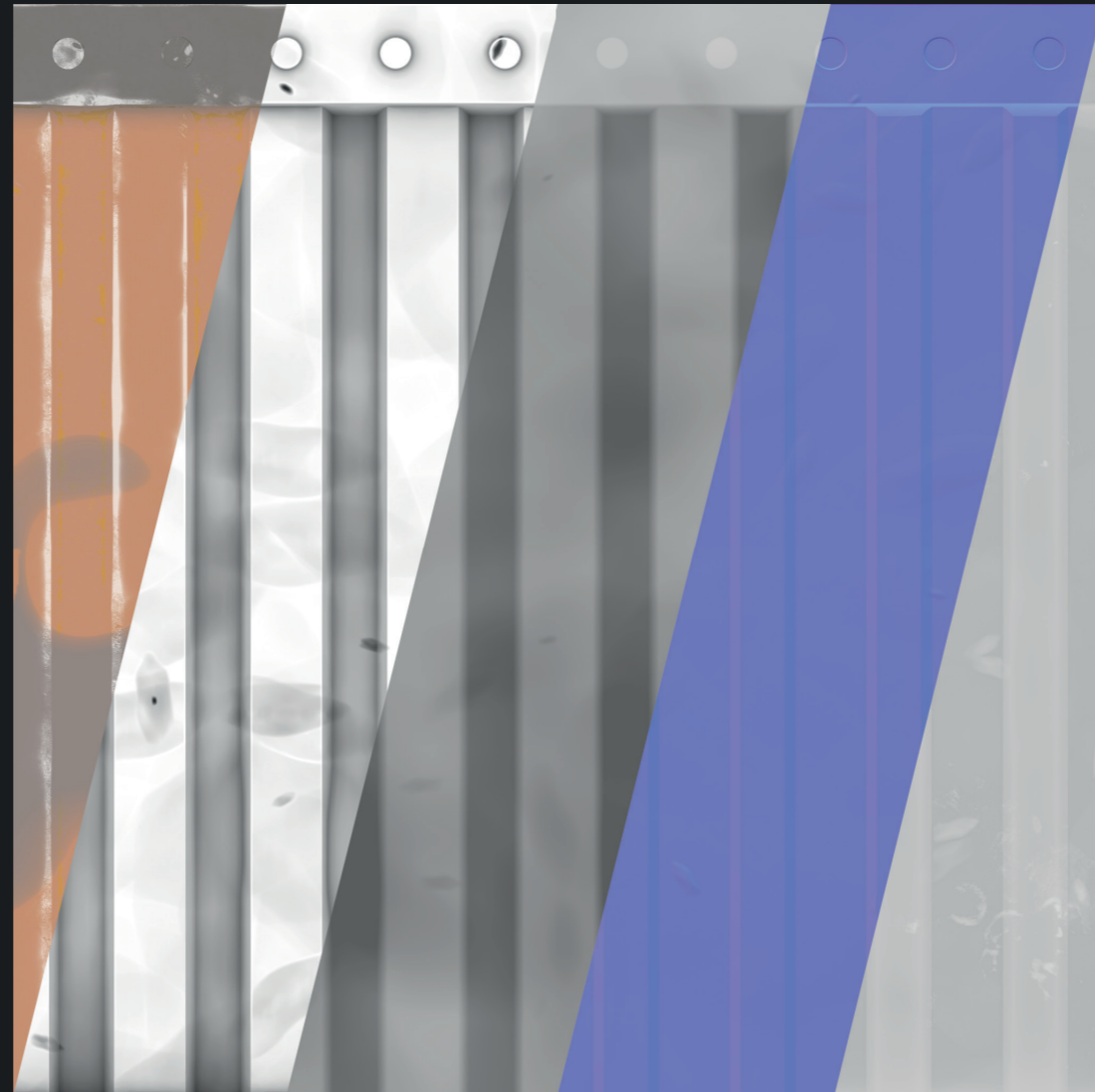


MATERIALES

Material pared

Base Color, Height, Ambient Occlusion, Normal y Roughness

Variaciones del base color

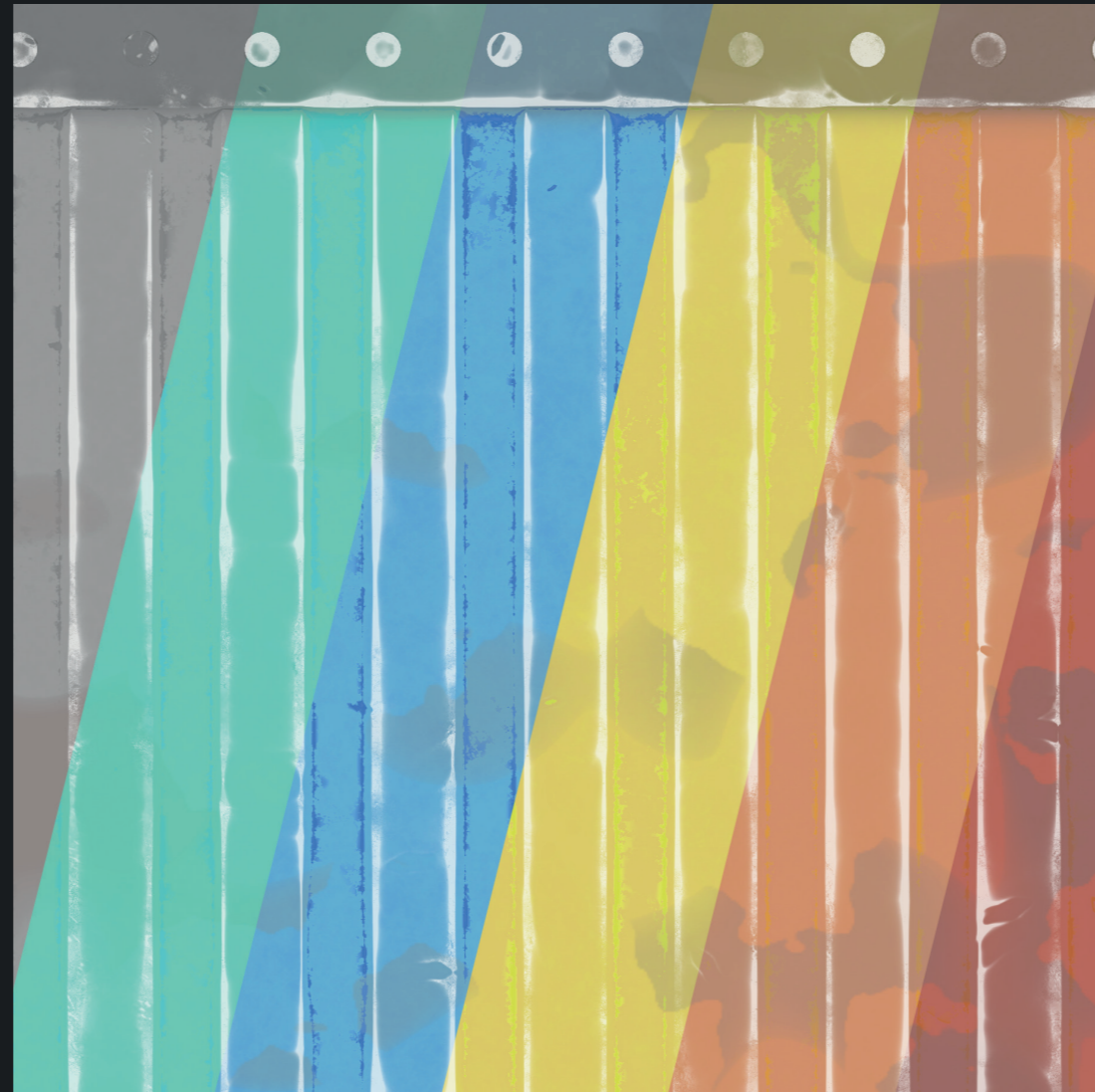


MATERIALES

Material pared

Base Color, Height, Ambient Occlusion, Normal y Roughness

Variaciones del *base color*



MATERIALES

Material pared

Base Color, Height, Ambient Occlusion, Normal y Roughness

Variaciones del *base color*

ESCENA





