



**UNIVERSIDAD
DE BURGOS**

Informática Gráfica

Set Blender para mobiliario de jardín

Profesor: Bruno Rodríguez García

Alumno: David Álvarez Castro

(dac1005@alu.ubu.es)

Índice

1 Introducción.....	3
2 Consideraciones.....	4
3 Desarrollo del proyecto.....	5
Texturas.....	5
Modelos.....	8
UVs.....	9
Mapas.....	10
Bounding volumes.....	12
4 Problemas e inconvenientes.....	18
Caídas repentina de Blender.....	18
Texturas erróneas.....	18
Múltiples texturas.....	19
Caras repetidas en los modelos <i>high-poly</i>	20
5 Mejoras.....	21
6 Conclusiones.....	22

1 Introducción

El objetivo del proyecto final es el diseño y desarrollo de un **set de assets en Blender** con una temática determinada. Se creará un set con una temática concreta a nivel de detalle alto y bajo y se aplicarán sobre ellos una textura desarrollada por el alumno, también en dos niveles de detalle distintos.

Adicionalmente será necesario analizar y decidir los *bounding volumes* de cada uno de los objetos justificando la decisión tomada y crear los mapas de UV utilizados para aplicar las texturas a los objetos y los *lightmaps* de visualización. Todos los elementos diseñados tienen que contar con un origen y una nominación adecuada siguiendo las pautas explicadas en los vídeos de ejemplo.

El objetivo de este documento es explicar el **proceso seguido** y citar las **decisiones** que se han ido tomando para obtener el resultado final que puede ser visualizado en el archivo *ProyectoFinal_DavidAlvarezCastro.blend* adjunto en la entrega.

Para llevar a cabo el proyecto se han seguido los siguientes pasos, tal y como se han explicado en los vídeos de ejemplo:

- Modelado de los elementos básicos: tablas, vigas y cadena de hierro.
- Selección de materiales e imágenes de la [página comentada](#) en las sesiones prácticas para crear una textura compuesta utilizando GIMP.
- Generar textura en dos niveles de detalle.
- Generar UVs de los elementos básicos en la vista *UV Editing* y aplicar textura en la vista de *Shading*. (textura *low quality*)
- Diseñar y modelar los objetos que formarán el *set*: mesa, maceta, columpio y valla.
- Copiar los elementos y generar los modelo *high quality* aplicando las texturas oportunas.
- Utilizar los elementos HQ para generar los objetos HQ.
- Generar *bounding volumes* de los modelos HQ (principalmente debido a que son los que más detalle tienen y han sufrido modificaciones).
- Generar *lightmaps* y correcta nominación de todos los elementos.

En principio se había considerado un *set* de mobiliario para la casa y el jardín, pero esto supone un gran abanico de elementos con múltiples texturas y materiales. Después de hablarlo con el profesor se decidió enfocarlo más a **mobiliario de jardín basado en madera** (estilo **rústico**), lo que podría resultar en aprovechar mucho más los ejemplos mostrados y centrar todos los elementos en madera, por ejemplo.

2 Consideraciones

A la hora de afrontar este proyecto he tenido en cuenta una serie de consideraciones:

- He considerado diseñar un número aproximado de elementos similar al ejemplo mostrado en vídeo.
- Me he basado en el *set* mostrado en los vídeos de ejemplo para simplificar el proyecto y porque realmente me parecía un *set* muy interesante y versátil.
- He intentado realizar exclusivamente una textura compuesta para intentar maximizar lo máximo posible los materiales que la forman.
- Para los modelos a diseñar me he basado en imágenes de Google y elementos conocidos que forman parte de mi jardín.

3 Desarrollo del proyecto

Este apartado describirá con cierto detalle todo el proceso que se ha seguido así como las decisiones y el porqué de éstas para obtener el resultado final.

Texturas

Aunque realmente lo primero que se ha hecho ha sido elegir qué elementos básicos formarían parte del *set* a modelar, y a continuación se le aplicaron los materiales decididos para, a partir de aquí, modelar el resto de objetos; se encuentra muy importante describir este apartado al inicio del todo. El principal motivo es que, antes incluso de modelar los elementos básicos (es decir, generar los elementos en Blender), realmente ya había preparado el conjunto de materiales que formarían la textura que quería aplicar y había preparado el fichero *png* en GIMP con una primera versión (véase apartado de Problemas e inconvenientes).

Lo primero de todo, tal y como se ha indicado, era pensar en el conjunto de elementos básicos que formarían el proyecto final: tablas, vigas, cadena y clavo de hierro (véase apartado de Modelos a continuación). Se pensaron en los materiales necesarios para ello, como son madera, tablones, hierro, material oxidado, perfiles de tablones. Se decide utilizar materiales basados en imágenes porque dan un aspecto más realista en comparación a los materiales procedurales, además, son más más compatibles con diferentes softwares.

Para obtener las imágenes de los materiales he utilizado la página textures.com. El criterio de selección ha sido un poco prueba y error y teniendo en mente ejemplos basados en mi jardín (color y tipo del tablón, principalmente).

Como resultado final, se ha escogido la siguiente imagen para los **tablones**:



En cuanto a la **madera utilizada para las vigas**, he escogido la siguiente imagen, que es bastante parecida en saturación y tono:



En cuanto a los **perfiles de las tablas**, se ha escogido la misma imagen que en el vídeo (por simplicidad y porque era bastante compatible a los tonos escogidos):



Por último, destacar las imágenes para los **elementos metálicos**. Se ha escogido una imagen de tinte metálico normal y otra con tonos oxidados para poder jugar con elementos metálicos no oxidados y oxidados, tal y como se muestran a continuación:



La textura de alta calidad generada es de tamaño 2048x2048, mientras que la de baja calidad es de 1024x1024. Como se puede apreciar, las texturas utilizadas son **potencia de 2** para mantener la relación de aspecto con diferentes niveles de detalle (las texturas también tienen diferentes niveles de detalle). Se genera una imagen de tamaño potencia de 2 siguiendo un poco las reglas de *OpenGL*, eso permite simplificar la gestión de texturas en diferentes niveles de detalle (hay que recordar que *OpenGL* seleccionaba la textura acorde al nivel de detalle).

A la hora de organizar una textura es muy importante tener claro qué elementos son los más importantes (cosas más importantes primero y con mayor resolución respecto al resto de materiales), en mi caso he priorizado las tablas y las vigas porque son los elementos que considero necesarios en todos los objetos a desarrollar. El tipo de textura utilizado es una **textura compuesta** porque es la más óptima para permitir adaptarse a distintos tipos de objetos; dentro de esta textura compuesta se pueden encontrar las famosas texturas *tileables* (textura de madera y de metal por ejemplo) que permiten repetirse infinitamente en nuestro modelo. Dentro de esta textura compuesta formado por los materiales de tablón y madera, por último, coloco con un mismo nivel de resolución tanto los perfiles de las tablas como los materiales para el hierro en un mismo nivel.

Se muestra a continuación el resultado final obtenido para las texturas de alta calidad:



En cuanto a la colocación de las tablas y el material de madera, creo muy necesario e importante destacar que se colocan en dirección horizontal para permitir que sean *tileables* a lo largo del plano horizontal (si se analizan algunas tablas utilizadas, la proyección es mucho más grande que la textura, pero eso no provoca errores en el objeto).

Por último comentar el motivo de haber elegido texturas compuestas sobre otras alternativas. En este caso, las **texturas personalizables** no tienen cabida porque nuestro objetivo es diseñar una textura que funcione en múltiples modelos, no en un modelo exclusivo; las texturas personalizables funcionan muy bien en personajes de videojuegos por ejemplo. Las **texturas simples** tampoco tienen mucho sentido en este proyecto porque el objetivo es generar una textura que valga para múltiples objetos y sea lo más reutilizable posible.

Modelos

El motivo principal del *set* es conseguir un conjunto de elementos basados en madera para poder modelar diferentes objetos de carácter rústico para un videojuego de miedo tipo *Silent Hills* o un videojuego de la edad media, así como para el diseño de mobiliario de jardín, por ejemplo. Por lo tanto, los elementos básicos más importantes serán tablas y vigas para poder modelar otros elementos más complejos a partir de éstos.

La ventaja de querer basar el *set* en elementos básicos como las tablas y vigas es que su diseño se basan en elementos poligonales muy sencillos, lo que nos va a permitir generar un conjunto significativo de tablas y vigas que posibilitará generar elementos más complejos a partir de ellos.

Como no se pretende modelar muchos objetos, el *set* de elementos básicos estará formado por:

- una serie de **tablas**,
- una serie de **vigas** de diferentes tamaños y formas,
- **cadena** (en este caso se hizo uso del modificador *array*, lo que nos puede permitir a partir de una hebilla generar una cadena con el número de eslabones necesarios, así como aplicar modificadores a la textura), y
- **clavo de hierro** para simular que las tablas están clavadas.

Con este conjunto de elementos básicos, la idea será generar objetos un poco más complejos que hagan uso de éstos. A continuación se muestra el listado:

- **mesa:** se ha intentado seguir un modelo muy parecido a una [imagen obtenida de internet](#), la idea será utilizar las tablas como la base donde apoyar cosas en la mesa y las vigas como el elemento central de la estructura que mantenga fija la mesa.
- **maceta:** se ha seguido un poco el modelo presentado en la [siguiente imagen](#) para conseguir un modelo muy parecido a una serie de macetas que tengo en el jardín de mis padres. La base de la maceta serán las tablas, utilizando vigas simplemente para formar el ángulo entre las caras y sujetar las diferentes tablas que la forman.
- **columpio:** se ha intentado obtener una [imagen de internet](#) de un modelo parecido al columpio que construyó mi padre de pequeño. La idea será utilizar tablas para la estructura de sujeción y una viga en la parte superior para fijar las cadenas del columpio con el asiento.
- **vallas:** también se quieren hacer unas vallas típicas, un poco siguiendo el mismo modelo de valla con el que cuento en mi jardín para completar este conjunto de objetos. En este caso se utilizarán exclusivamente tablas, tanto para la parte frontal como para la sujeción trasera.

Como se puede apreciar, son modelos muy básicos que se pueden encontrar en el mundo real, aunque también se podrían utilizar perfectamente para un videojuego con esta temática. La ventaja de todo esto es que son objetos que se pueden formar a partir de los elementos básicos descritos al inicio del apartado, un poco siguiendo el objetivo del proyecto de **diseñar un set lo más reutilizable posible con el menor número de materiales**.

Tanto para las tablas como para las vigas se ha partido de un cubo y se le ha ido dando forma hasta conseguir una silueta con forma de prisma rectangular; a partir de una de modelo se fueron haciendo las siguientes (tras aplicar la textura, claramente).

En el caso de la cadena, se ha partido de un *toro* de menos de 12 segmentos para conseguir esa forma peculiar de eslabón (no se ha considerado una resolución mayor debido a la utilidad del elemento). A partir de este elemento, aplicando el modificador *array* se puede conseguir duplicar las veces que se quiera el eslabón consiguiendo una cadena bastante real.

El clavo fue modelado a mano a partir de un cilindro con menos de 12 caras también. A partir del cilindro, se dio forma a la cabeza del clavo (escalando) y a partir de ahí se fue obstruyendo un cilindro interior hasta conseguir la forma requerida. Para conseguir la forma de punta se obstruyó por segunda vez y se hizo un *Merge At Center* conseguir la forma de pirámide.

UVs

Las proyecciones llevadas a cabo sobre los elementos básicos modelados son bastante simples debido a que todo el conjunto de elementos sigue un modelado basado en elementos poligonales. Las excepciones son el clavo y la cadena de hierro.

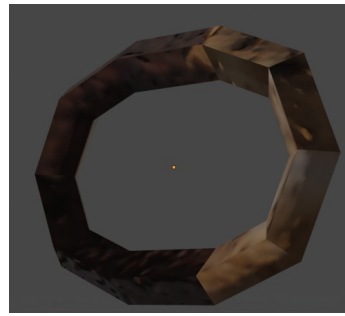
Los elementos básicos están basados en prismas rectangulares, por lo que la proyección que mejor se adapta es la de **project from view** para cada una de las caras, debido a que permite mantener la escala y las formas utilizadas a partir del ángulo desde el cual has llevado a cabo la proyección.

El procedimiento seguido para crear el UV map es el siguiente:

1. *Project from view* desde el alzado
2. *Project from view* desde el perfil
3. *Project from view* desde la planta

De esta forma conseguimos proyectar cada una de las caras de las vigas y tablas desde diferentes ángulos, y al tratarse de elementos poligonales muy básicos, el resultado es un unwrap muy sencillo.

Tanto en el caso del clavo como de la cadena, se ha optado por probar la proyección **Smart UV Projection** para analizar el resultado. Como el resultado obtenido era bastante preciso y permitía adaptar el *unwrap* a las texturas seleccionadas sin apreciar ningún efecto raro (también esto es debido a que el tamaño de los objetos era pequeño), se decidió escoger esta proyección y no probar ninguna más. Tal y como se ha comentado, en este caso esta proyección se adaptaba correctamente al propósito porque se trataba de objetos pequeños en donde no se apreciaban efectos extraños provocados por los tamaños de las caras. En caso de objetos más grandes, quizás fuese necesario modificar algunos de los vértices de *UV map*.



En el caso de la cadena, lo mejor sería seguir el procedimiento seguido en el vídeo de prácticas aplicado el elemento serpenteante, mientras que en el caso del clavo quizá una proyección cilíndrica hubiese sido la más correcta. Pero reitero, que el resultado obtenido tras aplicar *Smart UV Projection* es más que suficiente para lo que se quiere conseguir con estos modelos (son pequeños, casi inapreciables, y no forman parte de los elementos principales).

Dentro de este apartado, destacar la necesidad de **crear los *lightmaps*** de cada uno de los elementos (permite configurar la iluminación de cada una de las caras de los modelos). Para ello se ha seguido el mismo procedimiento que el explicado en clase, se ha generado un nuevo *UV map*, seleccionado todo el objeto, pulsado “u” y seleccionar *Lightmap pack*.

Mapas

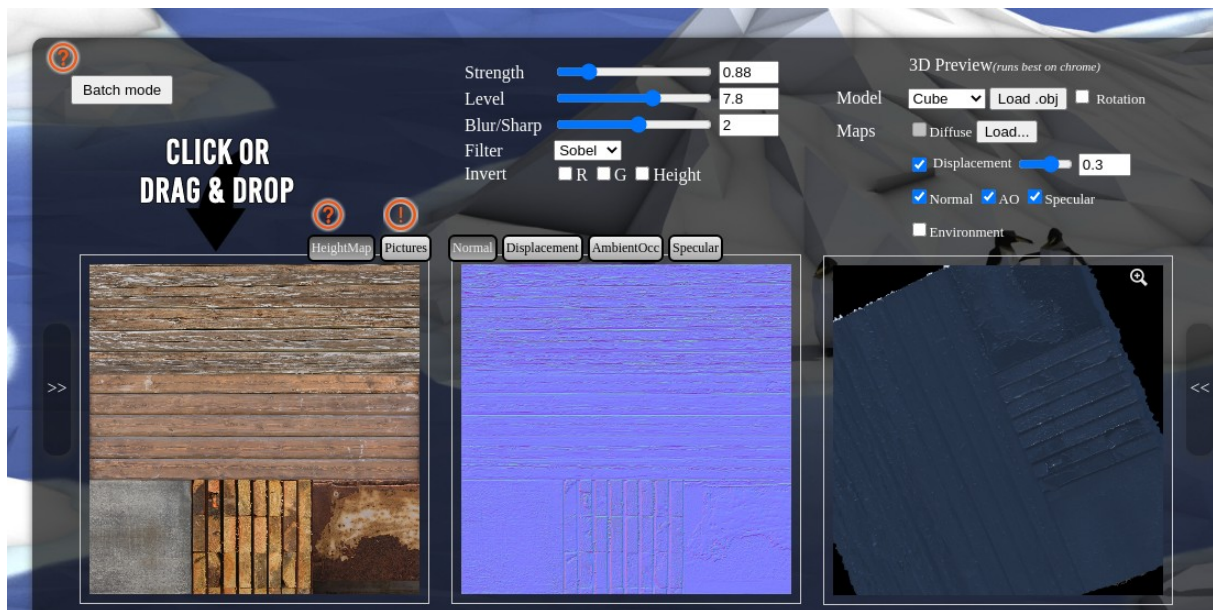
Tras aplicar las texturas correctamente siguiendo las proyecciones explicadas en el apartado anterior, se ha procedido a generar los mapas para el resto de materiales que forman las texturas y los elementos (tanto para la textura de alta como baja calidad). Los materiales se componen de diferentes mapas, tal y como hemos visto en clase:

- albedo con la información de color,
- height con la información de altura,
- normal con la información de los normales,
- roughness con la información de reflectividad, y
- ambient occlusion

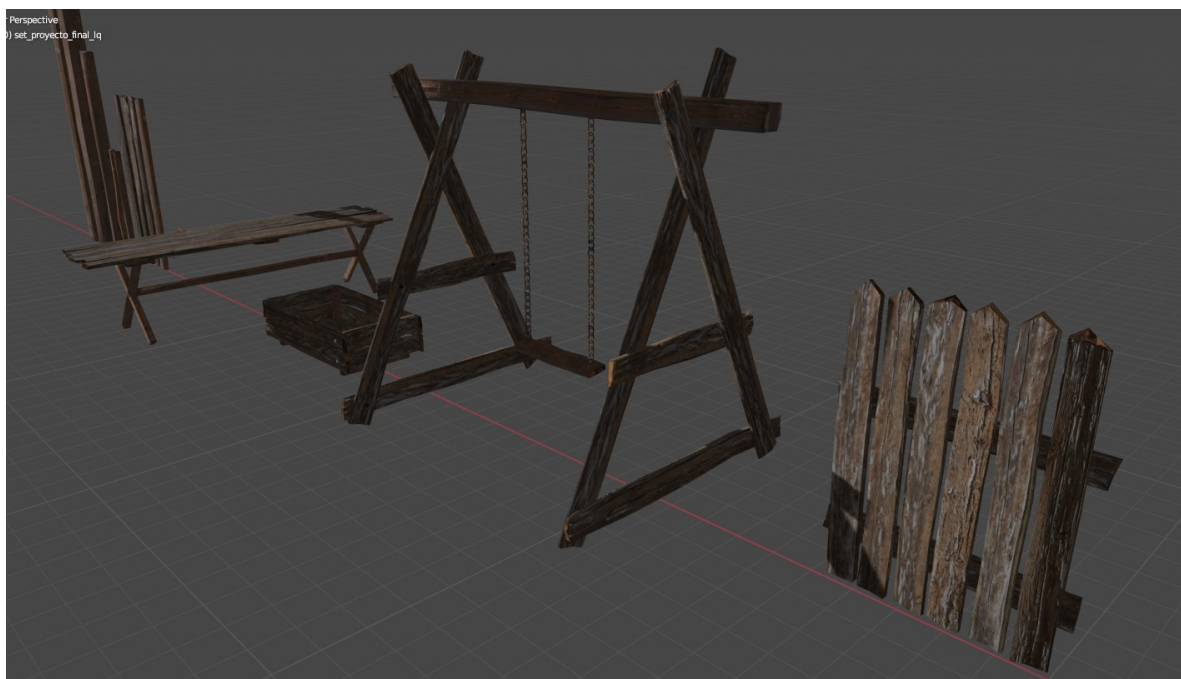
En este caso se van a trabajar con 3 mapas simplemente, que son los que se van a añadir en la vista de *Shading*:

- **mapa de *normal*** que permite guardar la información de dirección de cada una de las caras que forman el objeto (representado con los colores RGB),
- **mapa de *displacement*** que permite generar profundidad en el material añadiendo diferentes geometría, y
- **mapa de *specular*** que permite definir la intensidad con la que la superficie va a brillar en cada uno de los puntos de la geometría.

Para generar estos mapas se ha utilizado la [herramienta online](#) comentada en las sesiones práctica; en cuanto a los valores utilizados en cada uno de los mapas, se ha tenido en cuenta las explicaciones de los videos de ejemplo así como el criterio propio para intentar conseguir que la textura no quedase muy marcada.



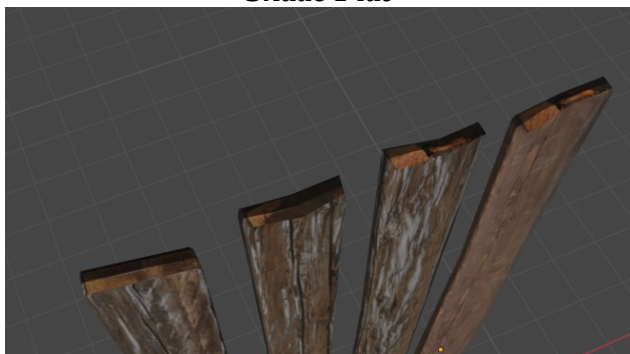
El **resultado final obtenido** para las texturas de alta calidad se muestra en la siguiente imagen:



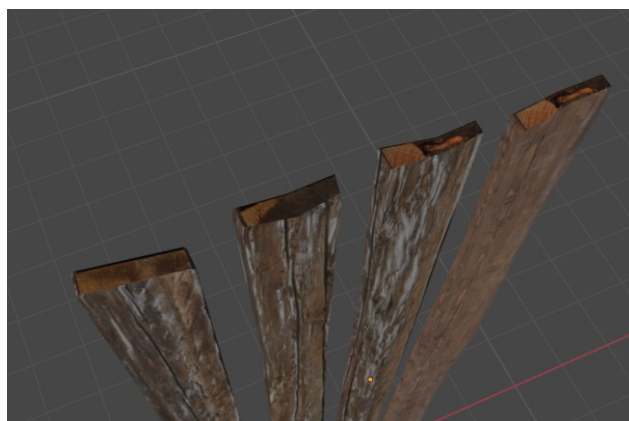
Al aplicar todos los mapas de texturas, se detectaron problemas en las sombras que fueron solucionadas aplicando *Shade Smooth* a todos los elementos, tanto los de alta como baja calidad.

Se muestra a continuación la diferencia entre *Shade Flat* y *Shade Smooth* en los modelos (en las tablas, el cambio abrupto de textura es debido al material elegido, no tiene nada que ver con las sombras):

Shade Flat



Shade Smooth

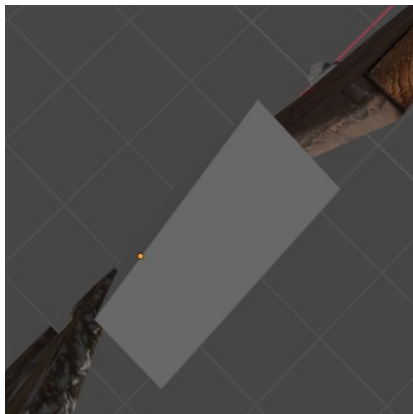


Bounding volumes

Los bounding volumes son las estructuras que envuelven a los modelos para gestionar su interacción, por ejemplo las hit boxes de los modelos de CSGO para calcular los impactos de las balas o el choque con las paredes. Teniendo todo esto en cuenta, se explicarán los volúmenes escogidos para cada uno de los modelos y los motivos de dicha selección.

Para simplificar y contar con un proyecto menos cargado, se han diseñado los volúmenes exclusivamente para los **modelos de alta calidad**. Aunque tanto los modelos de alta como de baja calidad deberían de contar con un volumen igual, en este caso se han llevado a cabo modificaciones que pueden afectar como puede hacer puntiagudas las vallas o cambiar de tamaño las tablas de soporte del columpio.

- **Tablas, vigas:** al tratarse de elementos poligonales muy básicos, al fin y al cabo siguen una forma de prismas cuadrangulares en ambos casos pero con proporciones diferentes, la mejor opción que se ajusta a los volúmenes que generan estos elementos son los volúmenes mínimos (MBR). Al ser polígonos muy simples y solo querer detectar colisiones con usuarios (normalmente con forma de cápsula) esta opción es la que mejor se adapta a su forma. Se había considerado aplicar cortes al prisma para adaptar mejor el volumen a las rotaciones y desplazamientos en el eje Y aplicados, tal y como se puede apreciar en la siguiente imagen:

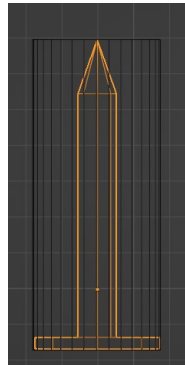
Tabla diseñada**MBR obtenido****Idea de corte**

Realmente no lo veo del todo necesario debido a que simplemente se quiere detectar el choque con ellas, y, aunque se quisiese gestionar para cogerlas por un NPC o usuario, creo que incluso el MBR sería suficiente, porque aunque la colisión no sea completamente real, la diferencia entre el *bounding volumen* diseñado y el propio objeto será de 1-10 cm, inapreciable para el jugador. Todo esto es gracias a que se tratan de elementos que se adaptan muy fácilmente a formas poligonales muy simples.

- **Cadena:** Al tratarse de un objeto complejo, posiblemente volver a aplicar MBR no sea lo más correcto y sea mejor optar por un volumen con forma de elipsoide o cilíndrica. El problema es que el objetivo de la cadena es formar parte de las tiras metálicas que sujetan el asiento del columpio, por lo que solo nos va a interesar saber si un usuario choca con ella, no va a ser necesario cogerla ni nada por el estilo. Teniendo esto en cuenta y el volumen optado por el columpio (véase siguientes puntos explicativos), creo que la mejor opción es un simple volumen mínimo debido a que permite cumplir el cometido de saber si un usuario ha impactado con la cadena.
- **Clavo:** El objetivo de este elemento es simplemente decorativo, forma parte de todos los objetos modelados. Por lo que no va a ser necesario cogerlo ni nada por el estilo; de todas

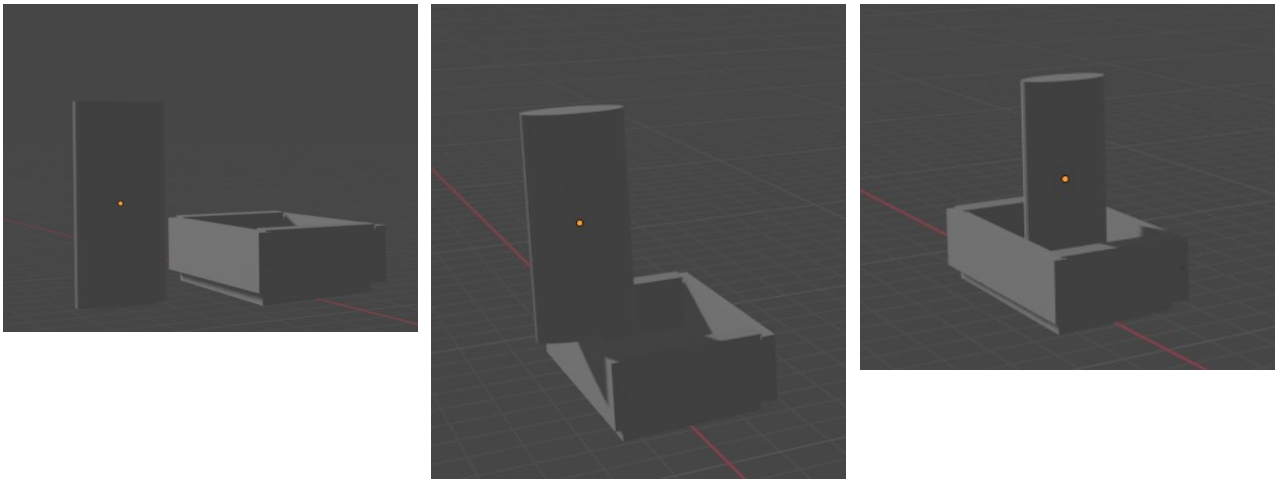
formas, se ha optado por aplicar un volumen cilíndrico debido a que la movilidad, en principio, es muy parecida a, por ejemplo, el hacha del vídeo explicativo. El usuario puede cogerlo, pero no es necesario el mayor de los detalles. Con esta frase me refiero a que el cilindro no es la forma correcta del clavo, la forma es más parecida a un cono, pero la diferencia de distancia entre el cilindro diseñado como volumen y la punta del clavo es inapreciable a la hora de calcular si una mano de un usuario coge o no coge la punta.

En la siguiente imagen se puede apreciar esta distancia, se muestra el clavo en naranja y en negro la malla que forma el volumen cilíndrico:

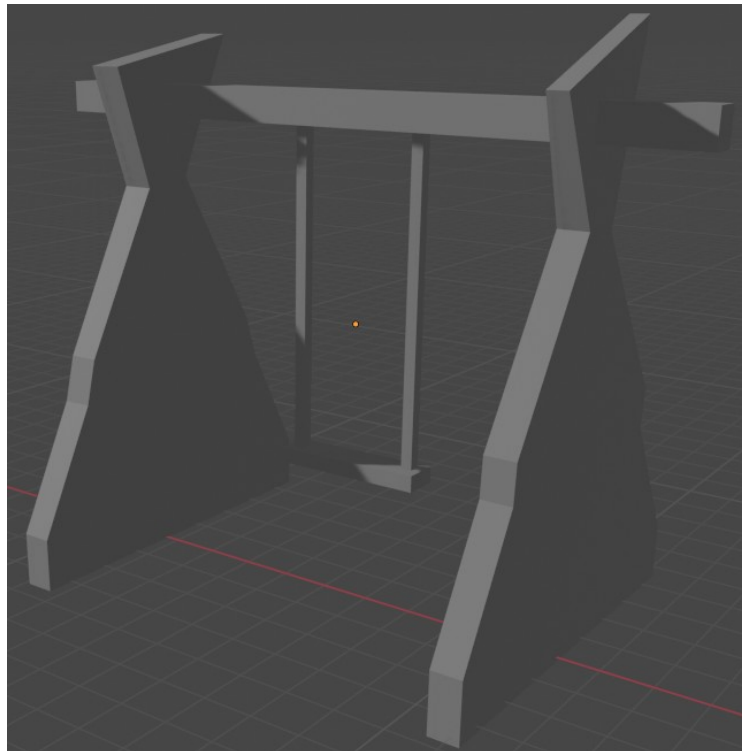


- **Mesa:** El volumen escogido para la mesa también es un volumen mínimo MBR. El motivo es que solo nos interesa saber si el usuario choca o no con la mesa, y como normalmente los personajes tienen un volumen con forma de cápsula, este es el mejor volumen que se adapta a estas situaciones. En este caso no se piensa que el usuario se agache, en dicho caso lo mejor sería optar por un volumen más personalizado teniendo en cuenta la parte superior de la mesa y los laterales, es decir, las patas.
- **Maceta:** La maceta también sigue una forma poligonal muy sencilla y simple, por lo que el volumen mínimo podría ser más que suficiente para saber si el usuario choca con la maceta o pasa por encima (por ejemplo, para “eliminar” las plantas que están en su interior). Como creo que ya he abusado mucho del MBR, opté por un volumen más personalizado que se adaptara a la forma de la maceta haciendo uso de prismas rectangulares por cada cara lateral y base. Con esto consigo hacer el efecto de que el usuario cambie de altura si se encuentra por dentro de la maceta, y puede cambiar de altura si se encuentra por encima de los bordes, lo que resulta en una interacción más natural del personaje con los elementos externos.

Las siguientes imágenes muestran este efecto (el cilindro deformado representa a un personaje):



- Columpio:** El columpio es un elemento bastante más grande que el resto, la interacción que sufrirá con el personaje es simplemente saber si choca con los elementos. Una opción habría sido elegir un volumen MBR, pero considerando la gran cantidad de espacio entre el volumen resultante y algunos elementos como las cadenas y la silla, así como el hecho de que el personaje estaría parado delante del columpio (y esto quedaría bastante raro sin haber “chocado” con nada), se ha considerado hacer un volumen personalizado basado en rectángulos.

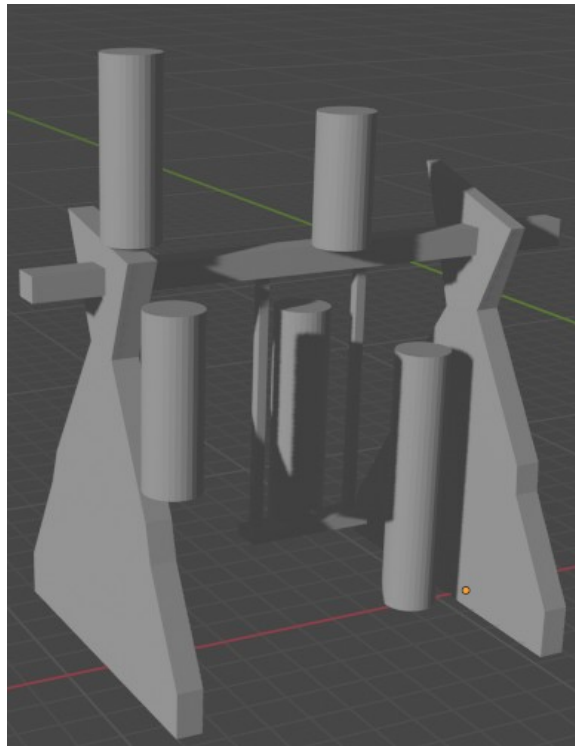


El resultado permite al personaje acercarse al columpio y pasar entre la estructura y el propio columpio, así como saltar y subirse encima de la silla y gestionar los choques del

personaje con las cadenas. Este volumen más personalizado, también permite detectar choques a diferentes niveles sobre la estructura:

- permite saber si el personaje se encuentra por encima de la estructura y simular diferentes niveles de altura dependiendo de si se encuentra en la viga superior o en las tablas de soporte, y
- permite una interacción más natural con los laterales debido a que el volumen se adapta a la forma, gestionando los salientes de las tablas de apoyo, o los ángulos que forman.

A continuación se muestra una imagen siguiendo el modelo de las macetas con las diferentes opciones descritas:



- **Valla:** El motivo es muy parecido a lo comentado en la mesa. Los personajes cuentan con un volumen en forma de cápsula, por lo que solo nos interesa saber el choque de dicha cápsula con el resto de objetos. Adicionalmente, en este caso, una valla es un elemento poligonal muy simple, y el volumen escogido es el que mejor se adapta.

El único problema es que no se podría detectar si el usuario está encima de las tablas que forman el soporte trasero, pero sería muy complicado con un volumen en cápsula; en caso de tratarse de personajes de CSGO (por ejemplo), se optaría por un volumen más personalizado que tuviera en cuenta las tablas salientes. Aunque no se tenga en cuenta este escenario, si que se podría detectar y permitir que el usuario suba por encima de la valla, lo que podría desembocar la acción de quitar vida al personaje por hacerse daño con las formas puntiagudas.

4 Problemas e inconvenientes

En esta sección se citarán y describirán brevemente los problemas que han ido surgiendo durante el desarrollo del modelado en Blender y las soluciones aplicadas. Se intentará ser lo más breve posible describiendo el problema y la solución aplicada.

Caídas repentina de Blender

Este problema lo he tenido desde el inicio de las prácticas de la asignatura. Por algún motivo que todavía no he podido entender, el controlador de la gráfica de Nvidia que tengo instalado no le acaba de gustar a mi versión de Ubuntu y de vez en cuando, por algún acceso a memoria no permitido, provoca que se **cierre Blender de forma no controlada**. Todo ello conlleva la pérdida total de los cambios llevados a cabo y no guardados.

La solución, por costumbre, ha sido estar guardando continuamente el proyecto en Blender; aunque no sin algún que otro susto.

Texturas erróneas

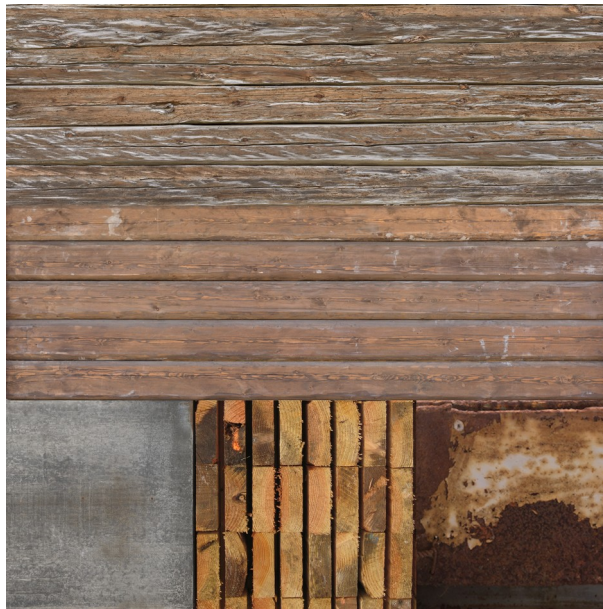
Al principio se había optado por reutilizar el mismo material de madera de la práctica anterior, tal y como se indicaba en el vídeo de ejemplo. Se muestra a continuación el material:



El principal problema era que el resultado final en los elementos no era el esperado y no me gustaba el resultado, por lo que fue necesario elegir un material diferente. En el caso del material de *plank*, pasó algo muy parecido. La primera selección resultaba en un modelo un tanto extraño a la vista y no me acabó de convencer del todo. Se adjunta imagen:

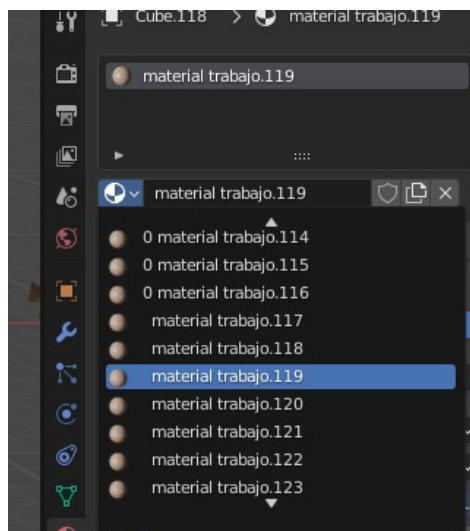


Al final, la **textura compuesta elegida** es la que se puede apreciar en la siguiente imagen, tal y como se indica en apartados anteriores.



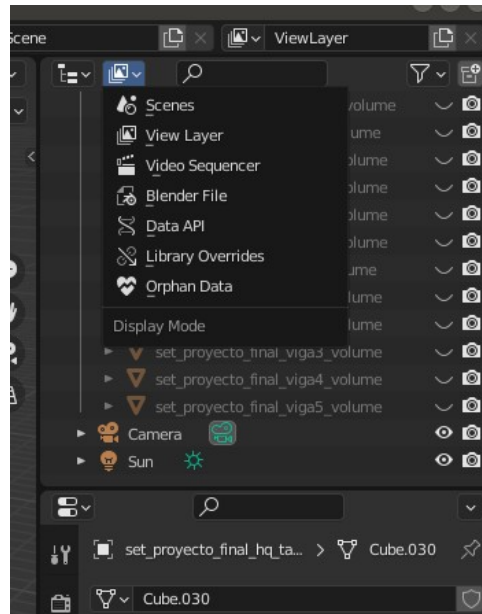
Múltiples texturas

Uno de los problemas a los que me he enfrentado al finalizar el proyecto en Blender era el hecho de contar con una **lista con múltiples materiales con nombres duplicados**, tal y como se puede apreciar en la siguiente imagen:



Al principio se envió un email al profesor explicando lo sucedido, pero no se llegó a encontrar el origen del problema y se había optado por dejarlo así marcando los nombres finales para las texturas correctas.

Después de investigar un poco por la interfaz de Blender, conseguí averiguar como **eliminar los elementos duplicados**. El proceso seguido fue, dirigirnos al menú de la derecha parte superior, en donde aparecen el listado de elementos modelados y cambiar la vista a *Blender Files* en el menú seleccionado en la siguiente imagen, y proceder a su eliminación:



Caras repetidas en los modelos *high-poly*

Una vez finalizados los modelos de alta resolución procedí a comprobar si había caras duplicadas; para ello entraba en modo edición y seleccionaba todo y me dirigía a la parte superior de la vista y pulsaba *Mesh → Clean up → Merge by Distance*. Tanto en el columpio como en la mesa parecía haber **caras duplicadas**, lo que me resultaba bastante raro porque los elementos básicos utilizados (tablas y vigas) ni el resto de objetos creados a partir de ellos contaban con caras duplicadas.

Después de analizar ambos objetos y no encontrar el motivo del error, en lugar de eliminar las caras duplicadas, opté por **rehacer ambos modelos**. El resultado es un **conjunto de modelos sin caras duplicadas**.

5 Mejoras

Una vez finalizado el proyecto se han encontrado una serie de *elementos* o factores a mejorar en caso de querer conseguir un *set* mucho mejor y más reutilizable, por decirlo de alguna manera.

1. Una de las mejoras obvias al analizar los objetos modelados es la **poca importancia de las vigas**, que quedaron relegadas a elementos secundarios en la mayor parte de objetos. Para mejorar este aspecto se podría haber optado por modelar un quinto elemento en donde las vigas fueran protagonistas, como puede ser el esqueleto de una pérgola de jardín estilo rustica o la estructura de un armario para almacenar material de jardín.
2. El material utilizado para las tablas, desde mi punto de vista, ha quedado un poco sobreexplotado debido al gran uso de las tablas en los objetos diseñados. Creo que una opción habría sido haber utilizado **dos materiales distintos de madera** para las tablas en detrimento de la textura de vigas y las texturas metálicas y de óxido (reduciendo sus respectivos tamaño).
3. Modelar **elementos más complejos** que hubiesen requerido de escultura, por ejemplo, o simplemente que tuviesen una complejidad mayor. Algunos elementos podrían haber sido un perchero rústico o incluso una palloza típica de una zona de mi tierra.



6 Conclusiones

La realización de este trabajo final me ha permitido poner en práctica todos los conocimientos adquiridos a lo largo de la asignatura:

- **Modelado básico** de formas poligonales como pueden ser primas rectangulares (utilizados, por ejemplo, para el modelado de tablas y vigas) hasta poder ser capaz de modelar figuras más complejas, teniendo en cuenta mi nula base de modelado, como cadenas de hierro o columpios.
- Diseño y desarrollo de **texturas y materiales compuestos**.
- Diseñar y analizar la importancia de los **UVs** y de los **lightmaps**.
- Diseñar y entender la importancia de los **bounding volumes** en el diseño de videojuegos y cómo influyen en la interacción entre objetos.

En definitiva, he encontrado muy **interesante y necesario este proyecto**, al fin y al cabo, en mi caso, mi base se basa en ingeniería del software, gestión de proyectos y desarrollo de aplicaciones y el poder haber estudiado de una manera muy superficial el mundo de la informática gráfica me ha resultado de mucho interés y he podido comprender el gran trabajo y horas de desarrollo que hay detrás de los videojuegos y otros desarrollos similares.

Haciendo un análisis muy breve de todo lo que ha sido la asignatura para mi, el principal problema al que me he enfrentado en la asignatura es el **componente artístico** que, desde mi punto de vista, es muy necesario a la hora de modelar y diseñar este tipo de elementos. Principalmente es debido a que mi perfil es muy técnico y metódico, el componente artístico brilla por su ausencia, y todo esto me ha supuesto esfuerzo adicional a la hora de hacer las prácticas y este mismo proyecto. Este ha sido uno de los motivos por los que he elegido este *set*, de esta forma he podido basarme en los ejemplos explicados y no partir de una base de cero.

Con todo ello, estoy muy contento con los resultados obtenidos en este proyecto; si alguien me hubiese dicho que sería capaz de hacer un columpio o una mesa que pudiese ser utilizada en un juego o un desarrollo gráfico realmente no me lo creería.