

Modelado 3D del edificio Sabatini de la Universidad Carlos III de Madrid

*Proyecto Fin de Carrera
Ingeniería Informática Superior*



Universidad Carlos III de Madrid



Autor: Alexis Sánchez López
Coordinadora: Susana Fernández Aguerri
Año: 2008

AGRADECIMIENTOS

El presente proyecto fin de carrera es el culmen a cinco años de estudios que no habrían sido posible sin el apoyo de mi familia. Por ello quiero dedicárselo a mis padres, que han estado ahí todos estos años apoyándome tanto en mi realización personal como profesional.

ÍNDICE

1.-Introducción	8
1.1.- Ámbito del proyecto	8
1.2.- Objetivos	8
1.3.- Estructura del documento	9
2.-Estado de la cuestión	11
2.1.- Diseño 3D.....	11
2.2.- Infoarquitectura 3D	11
2.3.- Modelado.....	12
2.4.- Mapeo y texturizado.....	17
2.4.1 Mapeo	17
2.4.2. Texturizado.....	19
2.5.-Adaptación del modelo para su uso externo	24
2.6.- Exportación	27
2.7.- Motor de renderizado	29
3.- Objetivos.....	32
4.-Memoria	35
4.1.- Fases de desarrollo	35
4.1.1.- Obtención de medidas, planos y fotos del edificio.....	37
4.1.2.- Estudio y comparativa de las técnicas y herramientas de desarrollo	38
4.1.3.- Instalación, configuración y prueba del entorno de desarrollo	40
4.1.4.- Modelado 3D.....	41
4.1.5.- Mapeo y texturizado	46

4.1.6.- Exportación y preparación del modelo para su uso externo.....	48
4.2.- Manual de usuario	51
4.2.1.- Sistema operativo Windows	51
4.2.2.- Otros Sistemas Operativos.....	53
4.3.- Manual de referencia	53
5.- Resultados	57
6.- Conclusiones.....	61
6.1.- Conclusiones sobre el desarrollo.....	61
6.2 Conclusiones personales.....	62
7.- Futuras líneas de trabajo.....	64
7.1.- Modelado.....	64
7.2.- Mapeo y texturizado.....	64
7.3.- Motor de renderizado	64
Bibliografía.....	67
Anexo A. Planos de AutoCAD	69
A.1. Planta Baja	69
A.2. Planta Primera	70
A.3. Planta Segunda	71
A.4. Planta Tercera.....	72

1.-INTRODUCCIÓN

1.1.- ÁMBITO DEL PROYECTO

El diseño en tres dimensiones (3D) es un campo de la informática muy amplio y está muy presente en nuestra vida cotidiana. Desde publicidad en televisión hasta las películas de animación o efectos especiales, pasando por los videojuegos o diseño de espacios y arquitecturas. Todos ellos, con sus similitudes y características propias, se nutren en mayor o menor medida de las técnicas y herramientas del diseño 3D para su realización.

Es precisamente en el diseño 3D de espacios y arquitecturas, donde se enmarca el proyecto fin de carrera “Modelado 3D del edificio Sabatini de la Universidad Carlos III de Madrid” (Proyecto en lo que sigue). El diseño 3D de espacios y arquitecturas se sirve de la tecnología informática aplicada a la arquitectura para realizar o recrear una estructura, un edificio, espacios interiores habitables, etc.

1.2.- OBJETIVOS

El objetivo del proyecto es la realización de un modelo tridimensional del edificio Sabatini de la Universidad Carlos III de Madrid (UC3M) situado en el campus de Leganés. La finalidad es que el modelo pueda ser integrado en terceras soluciones, y en especial, en el desarrollo de un videojuego inspirado en Los Sims [\[1\]](#) llamado “AI-Live” que está siendo desarrollado en paralelo en el departamento PLG [\[2\]](#) de la UC3M. El videojuego utilizará como escenario el edificio Sabatini generado en este proyecto.

Más allá de ofrecer una solución específica para su uso en el videojuego citado anteriormente, se pretende generar un producto genérico y libre de dependencias. Se prestará especial atención a que el modelo generado sea, en la medida de lo posible, lo más flexible y adaptable para su uso en futuros desarrollos externos no contemplados.

1.3.- ESTRUCTURA DEL DOCUMENTO

A continuación se enumeran los capítulos contenidos en el documento junto con una breve descripción de su contenido:

2.- Estado de la cuestión

Analiza y justifica las distintas decisiones tomadas en cuanto a la elección de herramientas y técnicas que conforman el entorno de desarrollo seleccionado para la realización del proyecto.

3.- Objetivos

Expone y establece los objetivos que debe cumplir el desarrollo del modelo así como las características y funcionalidades que pretende cubrir.

4.- Memoria

Especifica y detalla las distintas etapas del proyecto. Dentro de cada etapa se explican en detalle las distintas acciones llevadas a cabo. Así mismo se presentan los diversos problemas que han surgido durante la realización así como las soluciones que se han tomado para afrontarlos.

Por último se incluye tanto el manual de usuario como el manual de referencia que guía y facilita la integración y uso del modelo en futuros desarrollos.

5.- Resultados

Expone los productos obtenidos así como la adecuación de los resultados a los objetivos y propósitos de los mismos.

6.- Conclusiones

Pone de manifiesto las conclusiones de carácter general sobre el ámbito y desarrollo del proyecto para concluir con ciertas reflexiones y conclusiones sobre las aportaciones personales percibidas en el desarrollo.

7.- Líneas futuras

Enumera y detalla las posibles mejoras a introducir así como las líneas futuras de trabajo recomendadas en caso de querer extender y continuar el desarrollo.

2.-ESTADO DE LA CUESTIÓN

2.1.- DISEÑO 3D

El diseño 3D (también referido como infografía 3D) es el campo de la informática dedicado a la creación de imágenes generadas por ordenador que tratan de imitar el mundo tridimensional. Su uso está ampliamente extendido y es un campo en constante evolución; a medida que avanzan las capacidades computacionales se desarrollan nuevos métodos que obtienen resultados cada vez más realistas y detallados que rápidamente suplantando a sus predecesores.

Cuando pensamos en el uso actual que se le da al diseño 3D lo primero que suele venir a la cabeza es su aplicación en la industria del videojuego o en el cine para la creación de efectos especiales. Sin embargo, los campos de aplicación son mucho más amplios. Por ejemplo, en el presente proyecto utilizaremos las herramientas y aplicaciones de diseño 3D para la construcción digital de un edificio en tres dimensiones.

2.2.- INFOARQUITECTURA 3D

Dentro del amplio campo del diseño 3D se encuentra la especialización denominada como infoarquitectura 3D. Esta especialidad comprende la utilización de las herramientas de diseño 3D para construcción de modelos arquitectónicos, tanto espacios interiores como exteriores. En cierto modo se puede considerar como el siguiente paso a los tradicionales planos de construcción realizados en papel.

Con la infoarquitectura 3D se consigue crear un modelo tridimensional de una arquitectura con gran precisión y detalle. La infoarquitectura se usa habitualmente para vislumbrar el resultado de futuros proyectos antes de ser construidos. Sin embargo, este no es el caso particular del proyecto. La arquitectura a modelar ya está construida y lo que se quiere es obtener una réplica digital que represente fielmente los espacios interiores de la misma.

En general, las herramientas y técnicas utilizadas para la infografía 3D aplican también a aquellas usadas en la infoarquitectura 3D. Si bien, es cierto que dada la naturaleza de la misma, y debido a que las arquitecturas siguen un patrón más o menos recurrente existen procedimientos y soluciones a la hora de llevar a cabo un proyecto de este tipo que facilitan y guían las distintas fases de desarrollo.

Si además se tiene en cuenta que la arquitectura a replicar digitalmente es un edificio nos encontramos en un terreno aún más específico. Hablamos del terreno de las arquitecturas habitables; arquitecturas diseñadas con el cometido de albergar y ser utilizados por personas.

Las arquitecturas habitables siguen un patrón recurrente. Además, en el diseño de este tipo de arquitecturas, se usan mayoritariamente polígonos regulares y se caracteriza por la ausencia de curvas o superficies complejas. Siempre, claro está, nos encontramos con diseños vanguardistas que violan estas características, pero por regla general nos encontramos ante un patrón muy regular. Sin ir más lejos, la estructura básica suele ser paredes verticales combinadas con suelos y techos horizontales que se combinan para formar las distintas salas o espacios, los cuales se conectan y delimitan mediante puertas y ventanas.

En un desarrollo enmarcado en el campo del diseño 3D, y por lo tanto, también en el de uno basado en la infoarquitectura 3D, existen varias fases o etapas diferenciadas. Cada una de ellas cuenta con sus propias características y por lo tanto es necesario la disposición de distintos profesionales, técnicas y herramientas para su correcta realización. En las próximas secciones se abordará cada una de estas fases. Mediante un estudio de las técnicas y herramientas se justificará la elección del entorno de desarrollo para la realización del proyecto.

2.3.- MODELADO

El modelado, como su propio nombre indica, es la fase en la que se construye y genera el modelo en sí, representa la fase central del desarrollo y también la etapa que mayores recursos temporales consume.

Existen multitud de herramientas de modelado 3D en el mercado, cada una de las cuales tiene sus puntos fuertes dependiendo del campo de actuación concreto. Sin lugar a dudas, las herramientas más extendidas y usadas, tanto en proyectos de mediana o gran envergadura, son las presentadas por la empresa Autodesk [\[3\]](#).

Específicamente diseñada para la infoarquitectura Autodesk cuenta con AutoCAD; una herramienta profesional con todas las funcionalidades necesarias y de nivel técnico muy alto, la cual es usada incluso para el diseño de piezas en el campo de la ingeniería en las que la exactitud es un factor clave.

La finalidad del modelo que se busca generar no pretende ni necesita de un nivel técnico alto, no se pretenden incluir medidas ni detalles técnicos de la propia arquitectura. Además es relevante considerar la posterior necesidad de texturizado y tratamiento para su exportación y uso en un motor de renderizado en tiempo real. Estas características y requisitos impuestos hacen que las herramientas más indicadas para el desarrollo del proyecto sean aquellas herramientas de modelado 3D de ámbito general. Dicho esto se descarta para su uso herramientas específicas para la infoarquitectura, entre las que se incluyen la anteriormente citada AutoCAD.

A continuación aparece un listado de todas las herramientas de modelado 3D consideradas. Existen infinidad de ellas y posiblemente muchas de las que no están presentadas a continuación sean también totalmente válidas para el cometido que aquí se expone. Los criterios de selección de las herramientas consideradas se han basado principalmente en la popularidad y aceptación general por parte de la comunidad de usuarios, así como el uso de las mismas en importantes proyectos externos con resultados exitosos. La única restricción evidente es que la herramienta debe ser apta para infoarquitectura 3D, un requisito que cumplen todas las enumeradas a continuación.

Producto	Desarrollador	Página web
3D Max Studio	Autodesk	http://www.autodesk.es
AC3D	Inivis	http://www.inivis.es
Blender 3D	Fundación Blender	http://www.blender.org
Cinema 4D	Maxon	http://www.maxon.net
EIAS	EITechonology Group	http://www.eitechnologygroup.com
Houdini	Side Effects Software Inc.	http://www.sidefx.com
LightWave 3D	NewTek	http://www.newtek.com
Maya	Autodesk	http://www.autodesk.com
XSI	Softimage	http://www.softimage.com

Tabla 1. Listado de software apto para infoarquitectura 3D.

La elección de una herramienta particular en esta primera etapa de desarrollo no es una elección crítica. Todas estas herramientas ofrecen, en mayor o menor medida, exportaciones del modelo a formatos estandarizados, de manera que pueden intercambiarse y migrarse desde y hacia la mayoría de las herramientas de modelado disponibles.

Las grandes diferencias a tener en cuenta vienen principalmente en las fases de texturizado y renderizado, así como en el posterior tratamiento del modelo obtenido para adaptarlo a las necesidades particulares de cada situación. La presente fase de modelado no presenta mayores restricciones y problemáticas a tener en cuenta.

La intencionalidad del modelado del edificio Sabatini es doble. Por un lado se busca la integración como escena manejable por el motor de renderizado Ogre 3D [4]. Esta es la finalidad inicial y principal del proyecto pues se está desarrollando paralelamente, en el departamento PLG de la UC3M, un videojuego en el que se desea usar el edificio Sabatini como escenario.

En segundo lugar, y como requisito deseable, se busca que el modelo sea lo más adaptable posible a terceras soluciones. Es decir, se busca maximizar su compatibilidad para que pueda usarse en posibles futuros proyectos no contemplados en el momento de su realización.

Dadas estas restricciones las características buscadas en la herramientas de modelado es que disponga de funcionalidad de exportación a los principales formatos estandarizados presentes en la actual industria del diseño 3D y la posibilidad de exportación del modelo para ser utilizado en el motor de renderizado Ogre 3D.

Una vez analizadas todas y cada una de las herramientas se puede concluir que todas ellas cumplen con el primer requisito impuesto y cuentan, al menos, con la posibilidad de exportación a un formato genérico fuera del formato propietario y exclusivo de la propia herramienta. Existen multitud de herramientas independientes que permiten la transformación automática entre formatos por lo que con que la herramienta disponga de exportación a alguna de ellas, generalmente será posible obtener el modelo en el formato deseado sin grandes dificultades.

El motor de renderizado Ogre 3D está escrito en lenguaje C++ y hace uso de Microsoft Visual Studio. Es posible utilizar lenguajes de programación de alto nivel para programar el manejo de cualquier tipo de formato para el renderizado del edificio en tiempo real. Hay varios formatos estándar y de libre licencia que son aceptados y manejados por Ogre 3D, por el momento, asumiremos que la elección escogida (formato DotScene [\[5\]](#)) es la más conveniente. En la sección "[Adaptación del modelo para su uso externo](#)" será justificada esta elección.

Ogre 3D incorpora, por defecto, un manejador de escenas en formato DotScene (extensión .scene) por lo que para cumplimentar el segundo requisito de exportabilidad a Ogre 3D es necesario que el modelo pueda presentarse en dicho formato. DotScene no es más que un formato genérico por lo que atendiendo a lo expuesto en el párrafo anterior, no supone una tarea difícil en exportar el modelo y realizar la conversión desde cualquiera de las herramientas consideradas.

Teniendo en cuenta lo anteriormente descrito, cada uno puede considerar la herramienta que más se ajuste a sus necesidades. En mi caso particular he reducido la comparación a Blender 3D y 3D Studio Max. A continuación detallo los puntos principales considerados en la elección de estas herramientas sobre el resto, estos puntos se basan tanto en las propias características de las herramientas como en mis preferencias personales:



BLENDER 3D

1. Gratuito y lanzado bajo licencia Open Source.
2. Compatible con Windows, Mac Os y Linux.
3. Gran cantidad de información y soporte en la web oficial.
4. Compatibilidad integrada con un gran número de formatos, incluido el formato DotScene necesario para Ogre 3D.



3D Studio Max

1. Estándar de facto.
2. Experiencia previa en su uso.
3. Disponibilidad de licencia de software.
4. Disponibilidad de libros y documentación.

En primera instancia se seleccionó Blender 3D como herramienta de modelado pues cumple con todos los requisitos impuestos y cuenta con grandes ventajas, destacando la posibilidad de exportación integrada al formato DotScene.

Sin embargo, la elección final fue 3D Max Studio ya que no terminé de adaptarme al uso del interfaz de Blender 3D. El cambio de herramienta se basó principalmente en que trabajé con 3D Studio Max en pasadas ocasiones por lo que estoy habituado a la interfaz y manejo esta herramienta y sus opciones con cierta

soltura. Además, tengo a mi disposición una licencia de este software junto con gran cantidad de documentación y varios libros dedicados al uso de esta herramienta¹.

En el caso de no disponer de experiencia previa con ninguna herramienta de diseño mi recomendación sería el uso de Blender 3D pues, además de ser el único de la comparativa totalmente gratuito, cumple totalmente con los requisitos impuestos para el correcto desarrollo del proyecto.

2.4.- MAPEO Y TEXTURIZADO

Por lo general el modelado y texturizado (junto con el mapeo) representan dos fases totalmente diferenciadas. En proyectos de mediana y grande envergadura existen especialistas diferenciados para cada una de estas fases, una vez el modelador termina su trabajo lo dirige a otra persona que se encarga de mapearlo y texturizarlo.

El mapeo y texturizado (referido generalmente sólo como texturizado) se realiza en paralelo y permiten dotar a un determinado objeto de una apariencia realista. Esta fase sería equivalente a la asignación de un material a un determinado objeto, el cual contiene determinadas propiedades que lo caracterizan, a saber; el color, índice de refracción, rugosidad de superficie, dureza, etc.

A continuación se abordará la fase de mapeo y texturizado de manera independiente para exponer las decisiones tomadas para el desarrollo.

2.4.1 MAPEO

El mapeo (del inglés *mapping*) consiste básicamente en seleccionar y asociar una figura geométrica (la cual contendrá realmente la textura) y que envolverá al objeto original proyectando la textura sobre este último. La clave está en la elección de una figura geométrica que sea lo más simple posible, con ello se busca optimizar al

¹ También dispongo de los 3 tomos del manual de referencia oficial de Discreet [6] que cuenta con cerca de 3.000 páginas sobre el uso de la herramienta.

máximo el uso de recursos computacionales a la hora de su renderizado. Por ejemplo, en la **Figura 1** se puede observar un arco del edificio que contiene 19 caras, el cual ha sido mapeado usando un cubo. La complejidad en el cálculo de la textura e iluminación (entre otros) se reduce enormemente al tener que realizar los cálculos sobre 6 superficies en vez de 19.

La técnica de mapeo más común, la cual será usada para este proyecto, se denomina mapeo UVW (del inglés *UVW Mapping*)². El mapeo UVW es una variante del mapeo genérico anteriormente explicado. El mapeo UVW es básicamente una técnica que permite aplicar una imagen en dos dimensiones (2D), también llamada textura, sobre un objeto 2D o 3D.

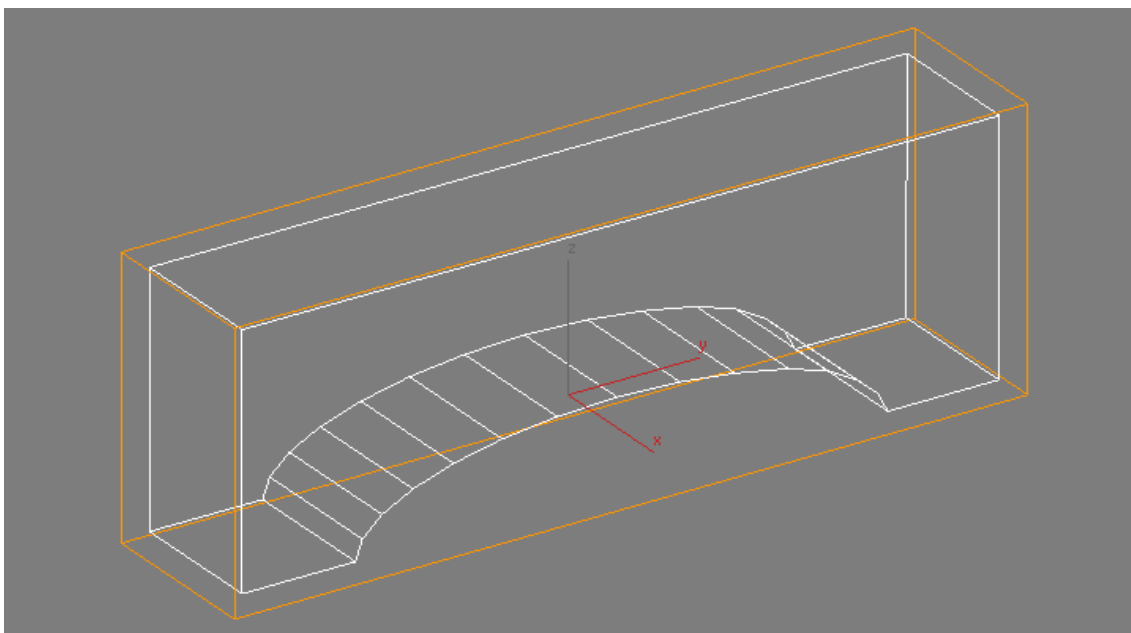


Figura 1. Ejemplo de mapeo de un arco (en blanco) mediante el uso de un cubo (en naranja).

Por lo general todas las herramientas de modelado disponen de las funcionalidades necesarias para realizar el mapeo UVW. En el caso de 3D Studio Max esta opción viene integrada en el software y dispone de multitud de opciones y funcionalidades extra. Para el mapeo del edificio Sabatini bastará con el uso de las opciones básicas. Además, de esta manera se asegura la compatibilidad una vez exportado el modelo. Aunque 3D Studio Max dispone de multitud de opciones complejas, algunas de ellas no son compatibles para su exportación al formato

² Las letras UVW hacen referencia a los ejes de coordenadas del sistema cartesiano (XYZ) y especifican la posición de la textura [\[7\]](#)

DotScene y, por lo tanto, tampoco pueden ser gestionadas y reconocidas por el propio entorno de renderizado Ogre 3D.

2.4.2. TEXTURIZADO

Una vez realizado el mapeo hay que aplicar una textura que irá superpuesta en la figura utilizada como mapeo. El texturizado se puede considerar como el tipo de material y las características con las que se dotan a cada objeto. Se puede especificar su color, rugosidad, índice de reflexión, luminosidad y un sinnúmero de opciones más. El texturizado puede ser una fase muy compleja dependiendo del compromiso y calidad que se quiera obtener, es de ahí que existan profesionales especializados y dedicados en exclusiva a esta fase del diseño 3D.

Para el presente proyecto es de vital importancia tener en cuenta la compatibilidad externa de las técnicas y opciones de texturizado que se usen. Como se había definido en los objetivos del proyecto hay que buscar la mayor compatibilidad externa posible así como cumplir el requisito indispensable de asegurar su compatibilidad con el formato DotScene y por lo tanto con el entorno de renderizado Ogre 3D.

3D Studio Max cuenta con su propia herramienta integrada de texturizado y dentro de la misma presenta hasta 20 de tipos de materiales a los que se pueden aplicar otras tantas técnicas, el resultado es una herramienta muy flexible y potente pero a la vez difícil de manejar. Además, la mayoría de parámetros no son compatibles ni exportables al formato DotScene³.

Para la exportación y uso del modelo en el motor de renderizado Ogre 3D se ha optado por la utilización del complemento (del inglés *plug-in*) OgreMax 3D Studio Max Scene Exporter. En la siguiente sección "[Adaptación del modelo para su uso externo](#)" entraremos en detalle sobre esta herramienta. Hasta entonces sólo cabe mencionar que este complemento añade una nueva técnica de texturizado (gestionado como un material) en 3D Max Studio que permite el texturizado del modelo asegurando la compatibilidad y su correcto renderizado posterior sobre Ogre 3D.

³ El listado completo de parámetros y configuraciones compatibles puede consultarse en el manual de referencia de OgreMax 3D Max Scene Exporter [\[8\]](#).

El uso de este complemento para el texturizado supone un arma de doble filo. Mirándolo por el lado positivo ofrece una solución ágil y eficaz de realizar el texturizado del modelo dentro del mismo 3D Max Studio con la seguridad que proporciona el saber que una vez exportado e integrado en Ogre 3D la escena será totalmente compatible y no producirá fallos a la hora del renderizado .

Sin embargo, añade una nueva dependencia que disminuye notablemente la adaptabilidad y compatibilidad del modelo para su uso en otros contextos. Este complemento está orientado únicamente como vía para compatibilizar el texturizado en el entorno de Ogre 3D, por lo tanto, si se utiliza este material específico sobre un objeto, el texturizado únicamente será compatible y visualizado correctamente en Ogre 3D. Ni si quiera el visor de 3D Studio Max es capaz de mostrar las texturas creadas mediante el complemento de OgreMax.

Por dicho motivo se ha optado por usarlo sólo en el caso de que sea imprescindible. Nos referimos, por ejemplo, en el texturizado de cristales, ya que los materiales estándar de 3D Studio Max no permiten la creación de transparencias que sean directamente exportables y compatibles con formato DotScene y por lo tanto son incompatibles con Ogre 3D. Para mantener el compromiso de ofrecer un modelo adaptable a cualquier otro entorno por cada textura generada mediante el complemento *OgreMax 3D Studio Max Scene Exporter* se acompañara de la correspondiente textura creada con las técnicas estándar para que el modelo pueda ser renderizado en terceras soluciones que no utilizan Ogre 3D.

Se ha comentado que la técnica del mapeo UVW consiste en superponer una textura 2D sobre un objeto 2D o 3D. Sin embargo, no se ha abordado la naturaleza de esta textura 2D aplicada.

Estas texturas pueden usar desde una imagen en formato “jpg” hasta una textura procedural generada mediante fórmulas matemáticas. Pueden modificar la superficie, emitir luz, crear sobras y un sinfín de opciones más. La aplicación de la textura junto con la elección de un motor de renderizado e iluminación condicionan fuertemente el resultado y realismo del resultado. Debido a que la fase de programación y uso del motor de renderizado queda fuera del alcance de este proyecto se ha optado por utilizar una técnica de texturizado simple para potenciar la adaptabilidad. Si se deseara la utilización de técnicas avanzadas de texturizado para el uso de la escena en futuros desarrollos se pondrán a disposición del usuario los archivos fuente del modelo desde los cuales se podrá aplicar efectos específicos para el motor de renderizado e iluminación deseado.

Para la realización del edificio Sabatini se utilizarán imágenes en formato “jpg”. El motivo de esta elección es porque aún siendo una técnica simple consigue unos resultados con gran nivel de detalle. El uso de una imagen permite emular luminosidad, rugosidad, tonalidades. Todos estos detalles y otros tantos podrían ser gestionados mediante técnicas específicas de 3d Studio Max, pero uso supondría una pérdida de flexibilidad además de un aumento de complejidad en el desarrollo.

La toma inicial de fotos del edificio permite obtener referencias exactas de cada uno de los materiales usados en el edificio. Tanto es así, que algunas de las fotos tomadas pueden ser usadas directamente como imagen de texturizado. Para el resto de imágenes se ha optado por su creación o por el uso y adaptación de imágenes en repositorios de internet cuyas imágenes están libres de restricciones legales.

Para la creación y modificación de las imágenes se puede usar un programa de retoque fotográfico. Desde el simple Microsoft Paint hasta herramientas profesionales como Adobe Photoshop podrían usarse para cubrir estas necesidades. En la **Tabla 4** se listan algunas de las herramientas más populares existentes en el mercado.

Producto	Desarrollador	Página web
Gimp	Proyecto colaborativo	http://www.gimp.org
Freehand	Adobe	http://www.adobe.com/es/products/freehand
Paint	Microsoft	http://www.microsoft.com
Paint Shop Pro	Corel	http://www.corel.com
Photoshop	Adobe	http://www.adobe.com/es/products/photoshop

Tabla 4. Listado de software de creación y retoque de imágenes.

Adobe Photoshop es una de las herramientas más populares y está avalada por cientos de profesionales. Debido a la disponibilidad de una licencia de software así como por mi experiencia previa en su uso, Adobe Photoshop es la herramienta elegida para la creación y modificación de las imágenes que servirán como texturas para el modelo del edificio Sabatini.

El siguiente aspecto a tener en cuenta viene condicionado por la aplicación de las imágenes generadas sobre la figura geométrica seleccionada en la fase de mapeo. La idea es que la imagen debe cubrir toda la superficie de la figura de manera que sea proyectada sobre la figura original. Lo más común es que existan dos tipos de texturas. Las primeras y más simples son las texturas cuya aplicación en un objeto cubre toda la superficie y por lo tanto se repiten una única vez. Por otro lado están las texturas tipo mosaico, las cuales no llegan a cubrir toda la superficie del objeto y es necesaria su repetición a través de la superficie del objeto.

En el texturizado del edificio la mayor parte de las texturas usadas han sido texturas de repetición. Aunque las texturas de repetición suelen ofrecer resultados con menos calidad de detalle suele compensar su uso debido a que se optimiza el uso de recursos (ocupan menos y se utilizan en varios objetos). Además, supone un importante ahorro de tiempo en su creación así como un uso menor en los recursos y por lo tanto el espacio usado en memoria es menor.

Para que la visualización de una textura de repetición sea correcta hay que tener en cuenta que el motivo de la textura sea uniforme en los márgenes de la imagen original. La **Figura 2** muestra el modo en que una textura se repite para cubrir por completo la superficie requerida.

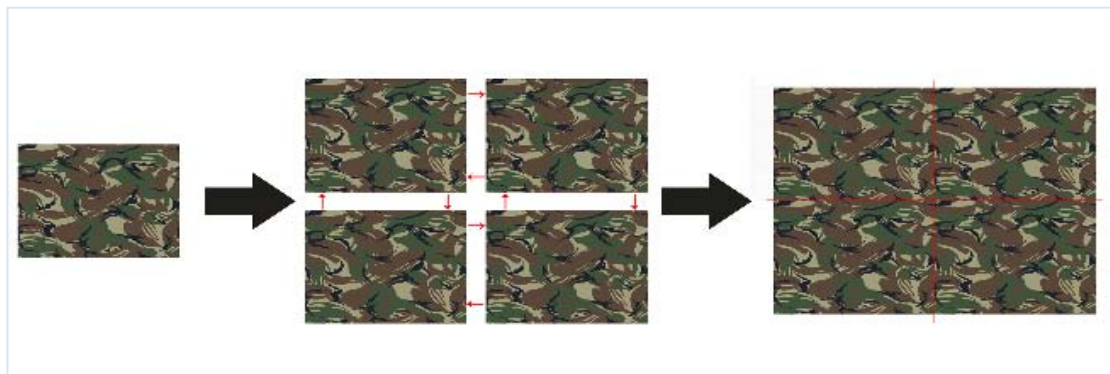


Figura 2. Proceso de ensamblado de imágenes para texturas de repetición.

Dependiendo si la textura se repite únicamente a través de un eje (repetición horizontal o vertical) o si se repite en ambos habrá que considerar la homogeneidad del motivo en los extremos verticales o laterales.

Para las texturas que se han creado desde cero y aquellas que no estaban preparadas para su repetición se han modificado y adaptado mediante el uso de un complemento para Adobe Photoshop llamado Seamless Whorkshop [9] que permite generar texturas de repetición a partir de imágenes que no lo son.

Nuevamente nos encontramos con diversas herramientas que permiten la generación de imágenes preparadas para su repetición y uso en forma de mosaico. En la **Tabla 5** se ofrece un listado con alguna de las herramientas para este propósito y que pueden usarse en sustitución a Seamless Workshop.

Producto	Desarrollador	Página web
Genetica	Spiral Graphics	http://www.spiralgraphics.biz/gen2tour/index.htm
Texture Maker	The Game Creators	http://texturemaker.thegamecreators.com
Seamless Texture Generator	--	http://www.seamlesstexturegenerator.com
MapZone	Allegorithmic	http://www.mapzoneeditor.com

Tabla 5. Listado de software para creación de texturas.

Una vez generada la textura no queda más que aplicarla dentro del software de modelado 3D elegido. Para la aplicación habrá que especificar el tamaño, dirección y orientación de la imagen. En la **figura 3** puede verse un ejemplo del texturizado de un arco del edificio utilizando una imagen de repetición.

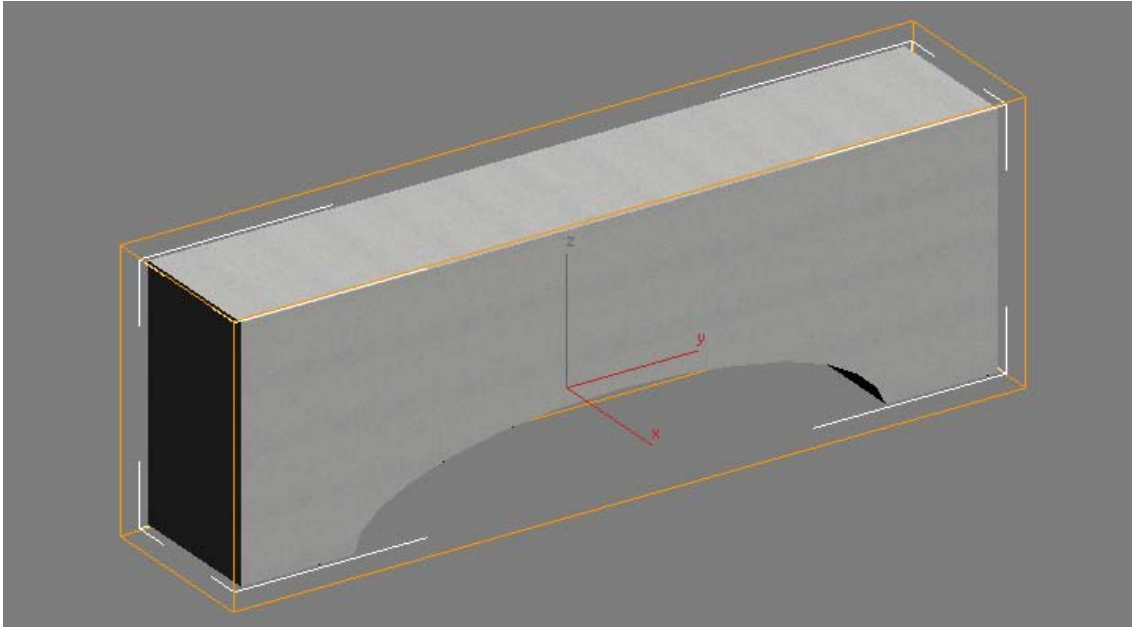


Figura 3. Ejemplo de texturizado con una imagen de repetición.

2.5.-ADAPTACIÓN DEL MODELO PARA SU USO EXTERNO

La realización del modelo, el mapeo y texturizado es sólo la primera fase del desarrollo. Una vez que se dispone de la recreación del edificio en 3D hay que estudiar y realizar su adaptación para su uso posterior. Como se ha expuesto en anteriores ocasiones, el fin principal del edificio es el de servir de escenario para el videojuego “AI-Live”, el cual está siendo programado en Ogre 3D.

El formato propuesto por Ogre3D para cargar y manejar un escenario es el formato DotScene. Este formato no es más que un documento XML estandarizado que contiene información sobre los objetos (en formato “.mesh”), posicionamiento, texturas, iluminación, etc.

Aunque es posible programar y utilizar cualquier otro formato o método para cargar el modelo del edificio en Ogre 3D la mejor opción es la utilización del formato

DotScene. Ogre 3D dispone de una librería especial que ofrece manejadores de escenas en formato DotScene facilitando la tarea al programador así como optimizando la gestión de modelos que utilicen dicho formato⁴. Teniendo en cuenta que el modelo final contiene miles de objetos, si no se decidiese utilizar el formato DotScene, habría que buscar e implementar otra manera de gestionar tal cantidad de objetos, junto con las texturas y propiedades de cada uno de los objetos que componen el modelo.

Al hablar de un escenario se suele diferenciar entre escenarios exteriores y escenarios interiores (comúnmente llamados niveles). El motivo de realizar esta diferenciación es que cada uno de éstos es tratado de manera diferente y existen distintas técnicas que optimizan la construcción y manejo de cada uno. Ogre 3D utiliza el formato DotScene tanto para escenarios interiores como exteriores, sin embargo propone distintas técnicas para la creación de cada uno de las escenas y dispone de distintos manejadores para cada uno de ellos.

El modelo generado para este proyecto encaja dentro de la categoría de un escenario interior. Cuenta también con un espacio exterior, que es el patio central, pero que debido a su pequeña extensión y que el suelo es plano puede ser tratado de manera eficiente utilizando las técnicas para la creación de un escenario interior.

Cuando hablamos de escenarios interiores, existe otro formato bastante popular. Se trata del formato BSP (del inglés *Binary Space Partition*) que se ha venido usando desde el lanzamiento del famoso videojuego Doom y que más tarde popularizó y licenció id Software. Hoy en día, este formato puede usarse sin restricciones, pues Id Software lo liberalizó para su uso sin restricciones. Existen varias versiones y modificaciones de este formato, pero por lo general ofrece la posibilidad de integrar una escena entera, desde el modelo hasta las texturas, pasando por el control de colisiones y diversas técnicas de optimización del renderizado.

Para el presente proyecto se ha considerado su uso e incluso se ha realizado una prueba para la generación de una parte del edificio. Las principales ventajas encontradas en este formato es que implementa un sistema de control de colisiones y técnicas de optimización y mejora del rendimiento.

El modelo diseñado en 3D Max Studio sólo ofrece una visión virtual del edificio. Para su uso en un videojuego es necesario especificar que puede ser o no atravesado (control de colisiones) así como dotarlo de un motor de físicas que añadan la

⁴ Ogre 3D dispone de la clase "SceneManager" cuyas funcionalidades para gestionar escenas en formato DotScene vienen especificadas en la API oficial [\[10\]](#).

simulación de gravedad. El formato BSP y las herramientas de edición de este formato permiten la definición de zonas de colisión, cuya información es integrada dentro del mismo archivo.

El principal problema del escenario es su gran extensión, estamos hablando de un edificio de gran envergadura y cuyo modelo virtual necesita gran cantidad de memoria RAM para el renderizado de todos los objetos en tiempo real. El formato BSP integra la técnica *back-face culling* que determina cuando un polígono es visible o no desde la posición actual de la cámara, de esta manera se optimiza el rendimiento haciendo los cálculos y procesando sólo las superficies que son visibles desde la posición actual de la cámara.

A pesar de las mejoras que integra este formato finalmente ha sido descartado para su uso. Mientras que este formato ha sido muy popular y ampliamente extendido, en la actualidad está siendo sustituido por formatos propietarios o por otras técnicas con las cuales se obtienen mejores resultados de optimización así como de mayor realismo en su renderizado y que están dejando anticuado al formato BSP.

Otro punto importante para la decisión de no usar este formato es que el sistema de colisiones que implementa (uno de sus principales atractivos para su elección) no es directamente compatible con el desarrollo que se está realizando en paralelo dentro del entorno Ogre 3D. El sistema de colisiones propuesto por el formato BSP se basa en un sistema de coordenadas con valores continuos. En el archivo BSP se especifican polígonos que encierran el volumen considerado como pared o suelo. Cuando un objeto en movimiento cambia su posición y sus coordenadas coinciden con las de ese volumen se considera entonces que hay una colisión y prohíbe su avance.

Por el contrario, en el desarrollo del videojuego realizado por el departamento PLG, el espacio es una variable discreta, básicamente el espacio se divide en casillas con una dimensión constante y establecida de antemano. Por lo tanto una casilla podría contener una parte considerada como zona de colisión y otra parte considerada libre de colisión. Por dicho motivo y si se decidiera utilizar el formato BSP habría que realizar un estudio e implementación para compatibilizar ambos. Además, esta es una decisión que debería tomarse y realizarse desde el equipo de desarrollo del videojuego por lo que se escapa a las competencias del presente proyecto.

Para generar un archivo BSP es necesario partir de la escena en formato “.map”, a partir de este archivo en el que sólo se representa el modelo hay que aplicar el texturizado con el uso de una de las herramientas específicas para más tarde compilarlo al formato BSP. No es posible la utilización de un formato genérico en el que se incluye el modelo texturizado. Este implica una dependencia y una pérdida de

flexibilidad. El texturizado realizado en formatos genéricos y estandarizados no es compatible con el formato BSP y viceversa. Por lo tanto, para atender al objetivo impuesto de adaptabilidad habría que realizar dos veces el texturizado, una sobre el formato “.map” y otra sobre 3D Studio Max.

Como punto final para descartar el uso del formato BSP cabe destacar que durante la prueba realizada para crear la escena se han encontrado bastantes problemas y el proceso resultó ser muy dependiente y tedioso. Hay que tener en cuenta que el formato BSP fue creado para su uso exclusivo de Id Software y por lo tanto en su diseño no se contemplaron su escalabilidad y adaptabilidad para un uso generalizado. La mayoría de las herramientas que permiten la creación de los escenarios en BSP han sido discontinuadas o no ofrecen la suficiente estabilidad como para aconsejar su uso.

No obstante si se desea o necesita utilizar este formato es posible exportar el modelo realizado en 3D Studio Max al formato de niveles de Id Software, realizar las modificaciones necesarias y compilarlo en formato BSP mediante el uso de una de las herramientas especificadas en la **Tabla 6**.

Producto	Desarrollador	Página web
GtkRadiant	Comunidad de usuarios	http://www.qeradiant.com
Quake Army Knife(QuArK)	Comunidad de usuarios	http://quark.planetquake.gamespy.com

Tabla 6. Listado de software para compilación de una escena en formato BSP.

2.6.- EXPORTACIÓN

Una vez finalizado el modelo es necesaria una herramienta que permite exportar el edificio Sabatini desde el formato de creación (3D Studio Max) al formato DotScene manejado por Ogre 3D.

Para realizar la exportación se ha optado por buscar un complemento que haga la exportación directamente desde el entorno de 3D Max Studio. Existen múltiples alternativas para la obtención del modelo en formato DotScene. La mayoría de las

alternativas se basarían en la exportación inicial del modelo a un formato genérico (como “.3DS” o “.OBJ”) para posteriormente utilizar una herramienta de conversión independiente.

La elección de un complemento para la exportación frente a una herramienta independiente se basa en la comodidad y agilidad que aporta la exportación directa desde 3D Studio Max.

En la **Tabla 7** se listan las herramientas y complementos que han sido probados y analizados para su posible uso. Todos ellos permiten la exportación a formato DotScene desde 3D Max Studio.

Producto	Desarrollador	Página web
Octopus	Yake	http://www.yake.org
3D Studio Max Scene Exporter	OgreMax	http://ww.ogremax.com
oFusion	ACE Studios	http://www.ofusiontechnologies.com/
LexiExporter	Mark Folkenberg	https://sourceforge.net/projects/lexiexporter

Tabla 7. Listado de complementos para la exportación a formato DotScene.

De los complementos listados en la **Tabla 7** han sido descartados Octopus y LexiExporter debido a que han sido descontinuados y las funcionalidades de exportación que ofrecen no están al nivel de OgreMax 3D Studio Max Scene Exporter y oFusion. Entre estas dos últimas herramientas la más profesional y la que más funcionalidades ofrece es oFusion la cual cuenta con una web con documentación y tutoriales para su uso. Sin embargo este producto es comercial. Ofrecen una versión libre mantenida por la comunidad de usuarios pero que no se actualiza desde 2007 y la cual, además, no es compatible con las últimas versiones de 3D Max. Cabe recordar que la versión utilizada para realizar el modelo es la versión 2009 y por lo tanto no es compatible con oFusion.

La mejor opción, por lo tanto, es el uso de la herramienta ofrecida por OgreMax, es un software libre, desarrollado íntegramente por una única persona. Sin embargo su calidad y funcionalidades se encuentran al nivel de oFusion y resultan más que suficientes para el cometido del presente proyecto. Dispone de página web con documentación y un foro. Es compatible con todas las versiones de 3D Max Studio (incluida la última versión) y es un proyecto en constante desarrollo y evolución. Tanto

es así, que en el momento de la redacción de este documento ha sido anunciado el lanzamiento de una nueva versión que verá la luz próximamente.

La solución que propone OgreMax es una solución de exportación completa que se integra como un complemento en el mismo entorno de 3D Studio Max y que permite ajustar gran cantidad de parámetros para personalizar el resultado de la exportación. Además como se ha comentado en la sección "[Mapeo y texturizado](#)" dispone de materiales manejables directamente desde 3D Studio Max que permiten un texturizado del modelo asegurando la completa compatibilidad en el entorno de Ogre 3D.

2.7.- MOTOR DE RENDERIZADO

Como se ha comentado en anteriores ocasiones la finalidad principal del proyecto es la de servir de escenario para el desarrollo que se está llevando a cabo en paralelo dentro del mismo departamento PLG de la UC3M bajo el motor de renderizado Ogre 3D.

El uso del modelo en Ogre 3D es un campo que se escapa, a priori, de los límites del presente proyecto. Sin embargo, se ha considerado oportuno su configuración y uso para asegurar la perfecta compatibilidad y optimización del edificio dentro de este entorno. Ogre 3D cuenta con librerías y manejadores específicos para gestionar el uso del escenario generado en formato DotScene por lo que el equipo encargado de trabajar en este entorno puede cargar la escena de manera sencilla y por lo tanto adaptarla dentro del proyecto creado en Ogre 3D.

Una vez cargado el edificio en el entorno de Ogre 3D hay varias modificaciones que deberían realizarse para mejorar la calidad del mismo, así como para su correcto funcionamiento. En primer lugar es necesario establecer un sistema de colisiones y una simulación de gravedad, para lo cual el primer paso es escoger un motor de físicas (Ogre cuenta con varios motores de físicas programados por la comunidad de usuarios).

En segundo lugar y si se quiere potenciar el realismo de la escena sería aconsejable la aplicación de una iluminación realista, para lo cual hay que configurar e integrar un motor de iluminación y sombras (de nuevo Ogre 3D propone varios

motores de iluminación utilizables mediante la inclusión de las correspondientes librerías)⁵.

Durante las fases iniciales del proyecto se ha estudiado a fondo la posibilidad de realizar el sistema de colisiones y simulación de gravedad, así como la realización de una iluminación más realista. Había varias posibilidades para realizar dichas funcionalidades pero debido a que es necesario la selección de un motor de físicas e iluminación limitaba enormemente la flexibilidad y posible adaptación futura al mayor rango de entornos en futuros proyectos. Además esta decisión debería tomarse desde el equipo de desarrollo que está usando Ogre 3D pues son los que actualmente están programando el núcleo del videojuego y pueden tomar esta decisión de manera más consciente y sobre unas bases mejor fundadas que las que se podría dar desde la perspectiva de este proyecto.

⁵ Se puede encontrar diversa información y ayuda sobre todas las mejores mencionadas en el libro editado por el creador de Ogre 3D "Pro Ogre 3D Programming" [\[11\]](#).

3.- OBJETIVOS

El objeto del proyecto es la creación de un modelo tridimensional virtual que recree de manera fidedigna el edificio Sabatini de la UC3M situado en el campus de Leganés. El fin principal que se le quiere dar al modelo generado es su uso como escenario para integrarlo en videojuego “AI-Live”.

El videojuego está siendo desarrollado en paralelo por el departamento PLG de la UC3M y por lo tanto aún no está finalizado. El desarrollo es realizado bajo el entorno Ogre 3D y por lo tanto es imprescindible que el modelo se encuentre en un formato compatible con dicho entorno y el cual pueda ser manejado como una escena en la que introducir personajes y objetos en su interior.

En definitiva, el objetivo primordial es la recreación del edificio en un formato integrable y usable en el entorno de Ogre 3D para su uso como escenario en un videojuego.

Como compromiso, más allá del uso inicial previsto, se ha considerado y prestado especial atención a la generación de un modelo flexible y adaptable. Con ello se busca que la recreación virtual del edificio sea lo más genérica posible y por lo tanto pueda ser integrada en futuras soluciones no contempladas durante su realización. El producto obtenido busca, por lo tanto, cumplir los siguientes puntos:

1.- Maximización de la compatibilidad externa.

Utilización de técnicas y herramientas genéricas para la obtención de un modelo en formato estandarizado, flexible y con un alto grado de adaptabilidad. La intención deseada es que el modelo virtual pueda ser usado en terceras soluciones independientemente de las herramientas de desarrollo utilizadas y de la intencionalidad de su uso.

2.- Minimización de la carga y uso de recursos computacionales.

Reducción al máximo posible del número de polígonos en la creación de objetos, aplicación de mapeo para optimizar el texturizado y uso de técnicas de optimización como la instanciación de objetos para evitar la presencia de elementos duplicados en memoria.

3.- Producto libre de restricciones de uso.

Utilización de herramientas que respeten la autoría y permiten el uso del producto obtenido sin restricciones de ningún tipo y permitiendo su uso para cualquier fin futuro deseado. Uso de texturas y objetos libres de derechos de autor. En definitiva que el resultado del proyecto esté libre de cualquier tipo de restricciones de uso.

4.- Equilibrio entre rendimiento y calidad de detalle.

El rendimiento del modelo en un motor de renderizado en tiempo real y la calidad presentada por el mismo son objetivos confrontados entre sí. Se prestará especial atención en buscar un punto de equilibrio entre la obtención de una buena calidad de detalle en la visualización a la vez que se intentará comprometer al mínimo posible el uso de recursos y por lo tanto preservar el rendimiento y fluidez en su uso.

Como último objetivo, se ha querido dotar al proyecto de cierta independencia, de manera que pueda ser visualizado y permitir un recorrido virtual por su interior sin la necesidad de ser integrado y adaptado en desarrollo de terceros.

El producto obtenido como resultado no es directamente visible y utilizable por un usuario final y es necesario su inclusión e integración en terceras soluciones para aprovechar al máximo su potencial. Sin embargo, se ofrecerá un modo de visualización y recorrido virtual del edificio fácilmente ejecutable a disposición del usuario. De esta manera se busca obtener un resultado con entidad y funcionalidad propia, un resultado tangible y evaluable con independencia a desarrollos de terceros.

4.-MEMORIA

El presente capítulo se divide en tres secciones. La primera de ellas corresponde a la descripción de cada una de las fases de desarrollo, en las que se explica y detalla cada una de las acciones realizadas, problemas encontrados así como las soluciones y resultados obtenidos. A continuación se presenta el manual de usuario que servirá de guía para la ejecución y uso por parte del usuario final. Por último, se adjunta un manual de referencia para desarrolladores, este manual pretende facilitar el uso del modelo para su adaptación en futuros desarrollos, así como servir de guía para extender y ampliar el producto obtenido.

4.1.- FASES DE DESARROLLO

El desarrollo del proyecto se divide en seis fases bien diferenciadas. Comienza con la toma de datos y estudios previos para pasar a continuación con la fase de desarrollo del producto, por último el producto obtenido es modificado y preparado tanto para su ejecución por parte del usuario final como para su integración en terceras soluciones.

En el diagrama de Gantt de la **Figura 4** se puede apreciar cada una de las fases junto con la asignación de recursos temporales correspondientes. En total, contando con las fases iniciales de documentación y obtención de datos, la realización de proyecto conlleva 145 días, lo que supone cerca de 5 meses. Dentro de estos el modelado y texturizado requieren la mayor parte del tiempo con unos 85 días para su realización.

4.1.1.- OBTENCIÓN DE MEDIDAS, PLANOS Y FOTOS DEL EDIFICIO

Antes de iniciar la creación del modelo es necesaria la recopilación de la información necesaria sobre el edificio. En primer lugar y para la realización del modelado es necesario contar con las medidas del edificio para hacer una recreación realista a escala. En segundo lugar, para la fase de mapeo y texturizado es necesario saber los acabados y materiales del edificio para crear texturas lo más semejante posible a los materiales usados en el edificio real.

Para la obtención de las medidas del edificio se ha contactado con la unidad de obras y mantenimiento de la universidad que ha facilitado planos de planta del edificio en formato AutoCAD (ver **anexo A**). Sin embargo, estos planos arquitecturales sólo ofrecen medidas de planta y además carecen de medidas exactas de mobiliario y otros elementos como pueden ser escaleras o puertas. Por ello se ha considerado indispensable tomar medidas sobre el propio edificio. Especialmente se ha necesitado tomar todas las medidas de altura para suplir la falta de planos de perfil. Además se han tomado fotos y medidas de puertas, ventanas, escaleras y demás elementos que sólo están representados de forma esquemática en los planos.

Para la obtención de medidas ha sido imprescindible la utilización de un medidor láser (modelo Hilti PD 38) que ha permitido tomar medidas de manera rápida y exacta de alturas interiores y exteriores. Medidas, como la altura exterior del edificio, no hubieran sido posibles capturarlas con el uso de cintas métricas tradicionales.

Junto con las medidas se han realizado fotografías de cada uno de los elementos relevantes del edificio para captar su apariencia y material de construcción. Estas fotografías serán utilizadas posteriormente durante la fase de texturizado del edificio. Algunas de las fotografías se utilizarán directamente como imagen de texturizado de los objetos, mientras que el resto servirá como referencia para buscar y crear nuevas imágenes de texturas que sean lo más similares posible a los materiales usados en el edificio original. Este procedimiento es esencial para dotar al edificio de realismo y detalle, del mismo modo permiten una recreación fidedigna de cada uno de los aspectos del edificio.

Antes de iniciar el desarrollo, y en paralelo a la toma de medidas, se ha realizado un estudio y comparativa de herramientas y técnicas consideradas para el desarrollo del proyecto. La finalidad del estudio es la elección del entorno de desarrollo que mejor se adapte a las necesidades del proyecto y que por lo tanto produzca los mejores resultados posibles, cubriendo así, los objetivos impuestos. El estudio y comparativa permite tomar las decisiones necesarias sobre bases fundamentadas así como facilita y especifica el procedimiento a seguir para la obtención del resultado deseado.

La comparativa y justificación de las herramientas de carácter crítico para el desarrollo del proyecto están detalladas en las correspondientes secciones dentro del [capítulo 2](#) (estado de la cuestión).

Algunas elecciones de software vienen impuestas por las propias restricciones y dependencias de las herramientas base seleccionadas. Otras herramientas, como es el caso de AutoCAD, vienen impuestas por factores externos. En el caso concreto de AutoCAD su elección viene condicionada por el propio formato de recepción de los planos iniciales. El resto de herramientas, como por ejemplo Adobe Photoshop , han sido seleccionadas por motivos personales (experiencia en su uso y disponibilidad) y pueden ser sustituidas por cualquier herramienta similar sin que comprometa la obtención de los resultados buscados.

A continuación se listan, una a una, todas las herramientas así como el objeto de su uso dentro de las distintas fases de desarrollo.

a) 3D Studio Max 2009

Representa la herramienta central de desarrollo. Con ella se realiza el modelado 3D del edificio así como el mapeado y texturizado. Desde esta herramienta se exporta el modelo listo para su uso e integración externa.

b) OgreMax

Mediante la instalación del complemento OgreMax 3D Studio Scene Exporter se permite la exportación del modelo realizado en 3D Max Studio al formato DotScene para que pueda ser manejado en el motor de renderizado Ogre 3D. Dispone, además, de un material especial

integrado en 3D Max Studio que permite el texturizado del modelo asegurando la compatibilidad y correcta visualización posterior en Ogre 3D. Por último, OgreMax Viewer es un proyecto realizado en Ogre 3D que permite cargar el modelo y ofrece una visualización básica en tiempo real del mismo.

c) Ogre 3D / Microsoft Visual Studio 2005 SP1

Ogre 3D es el motor de renderizado usado por equipo de desarrollo encargado de programar el videojuego “AI-Live” que utilizará como escenario el edificio aquí desarrollado. Este es usado para verificar que el edificio sea correctamente visualizado e integrado en Ogre 3D. Además permite ejecutar OgreMax Viewer el cual ofrece la posibilidad de recorrer virtualmente el modelo generado.

d) Adobe Photoshop CS 3

Photoshop es una herramienta de creación y retoque fotográfico. Su utilización en el proyecto se enmarca en la fase de texturizado donde se utiliza para la creación y modificación de las imágenes usadas para el texturizado del modelo.

e) Google SketchUp

Google SketchUp [\[12\]](#) es un software propietario de Google, es una versión simplificada de un software de modelado 3D. Este software maneja un formato propietario (extensión “.skp”) que es utilizado en la extensa galería de modelos 3D [\[13\]](#) de libre acceso y disponibles a través de la propia página web de Google. Este software es usado para la exportación a 3D Max de determinados modelos genéricos específicos que han sido usados en el edificio (bisagras, pomos, etc.).

f) RocaDrawing

Roca, empresa dedicada a la creación de espacios de baño, dispone de un software de acceso libre denominado RocaDrawing [\[14\]](#) donde se tiene acceso a la mayoría de modelos de baño Roca en 3D disponible para su descarga en varios formatos genéricos. Este software se ha usado para la obtención y creación de los espacios de baño del edificio Sabatini.

g) Seamless Workshop plug-in

Seamless Workshop plug-in es un complemento para Adobe Photoshop que permite la creación de imágenes que puedan ser mostradas como una textura de repetición en las que no se aprecian saltos en las zonas de unión.

4.1.3.- INSTALACIÓN, CONFIGURACIÓN Y PRUEBA DEL ENTORNO DE DESARROLLO

Esta fase comienza con la obtención, instalación y configuración de todas las herramientas seleccionadas. Consiste en la preparación del entorno de desarrollo que será usado en las distintas fases del proyecto.

Una vez obtenidas y configuradas las herramientas de uso se ha realizado una prueba del entorno. Esta prueba consiste en la recreación de una pequeña porción del edificio (una habitación pequeña), realizando todas y cada una de las fases hasta la obtención del modelo final texturizado ejecutándose en Ogre 3D. De esta manera se asegura el perfecto funcionamiento de todas las herramientas y certifica que los objetivos marcados son viablemente alcanzables usando el entorno de desarrollo seleccionado.

El entorno final de desarrollo fue obtenido mediante la realización de varios ciclos iterativos de pruebas sobre el entorno. Gracias a las pruebas realizadas sobre cada una de las herramientas se ha conseguido establecer el conjunto global más apropiado para alcanzar los objetivos requeridos.

En la fase de modelado se sustituyó Blender 3D a favor de 3D Studio Max debido a la experiencia personal previa con esta herramienta y la inadaptabilidad inicial de la interfaz gráfica de Blender 3D. En las pruebas de formato de exportación se comprobó las ventajas del uso del formato DotScene frente al formato BSP, las cuales ya han sido detalladas. Del mismo modo permitió comprobar de manera práctica y real la elección de OgreMax como herramienta de exportación y puente entre 3D Studio Max y Ogre 3D. Todas estas elecciones se han justificado en las correspondientes secciones del [capítulo 2](#).

La utilidad de estas pruebas de entorno han resultado ser críticas. Han supuesto un ahorro de tiempo y ha evitado futuras problemáticas, permitiendo comprobar, antes de iniciar el desarrollo, que algunas de las herramientas seleccionadas inicialmente para el entorno de desarrollo no eran las más apropiadas. Si

no se hubiera realizado esta fase los problemas habrían surgido a mitad del desarrollo donde el cambio de herramientas o reconducción del proyecto habría supuesto unos costes mayores.

4.1.4.- MODELADO 3D

La fase de modelado representa, sin lugar a dudas, la etapa que más recursos temporales ha requerido. Como se puede apreciar en el diagrama de Gantt de la **Figura 4** ha consumido un total de 85 días de los 145 días que ha requerido el proyecto en total. Lo que supone cerca del 60% del tiempo total invertido en el desarrollo.

El método inicial utilizado para abordar la ingente cantidad de objetos a crear ha sido la construcción planta a planta, empezando por paredes, puertas y ventanas, para continuar con el resto de detalles.

La cantidad de detalles y objetos a tratar, pronto se hizo inmanejable y agotaba los recursos del equipo informático utilizado. Fue entonces cuando se decidió un acercamiento distinto intentando reducir al máximo posible el número de polígonos y saturación de recursos del equipo informático usado para el desarrollo.

Debido a la naturaleza del edificio existen un gran número de objetos, como pueden ser puertas o ventanas, que aparecen en el edificio multitud de veces. Cada objeto que aparece repetido puede ser instanciado para colocarlo tantas veces como sea necesario en la escena. Instanciar un objeto es similar a la creación de un puntero en un lenguaje de programación. De esta manera sólo es necesario procesar y almacenar en la memoria RAM una copia de este objeto, el resto de referencias apuntarán al original. Mediante el fichero XML del formato DotScene un objeto se puede situar infinidad de veces en distinta posición, escala e incluso con un determinado grado de inclinación en cualquier de los ejes sin necesidad de copiar o multiplicar el objeto en sí. Para alcanzar el compromiso de agilizar la carga computacional se han obviado determinadas indicaciones de los planos originales. Así pues, cada tipo de puerta o ventana cuenta con un único objeto que es instanciado a lo largo del resto del edificio, omitiendo, por ejemplo leves detalles como la dirección de apertura.

El uso de la técnica de instanciación junto con la leve relajación de fidelidad a alcanzar en la recreación del modelo ha supuesto la reducción del tamaño final de la escena y por lo tanto acelerado su carga y fluidez de ejecución.

Una de las limitaciones de 3D Studio Max y que ha condicionado el método de afrontar el modelado es que sólo se permiten instancias dentro de un mismo archivo, si se llevan instancias a distintos archivos y posteriormente se fusionan serán considerados objetos distintos y por lo tanto existiría una duplicidad innecesaria la cual además repercutiría en la duplicación de memoria y recursos usados.

Si bien es posible la posterior modificación sobre el XML de la escena, pero esta opción ha sido descartada debido a la dificultad de posicionar los objetos en su correspondiente posición de manera programática y no mediante una interfaz visual como ofrece el 3D Studio Max.

La imposibilidad de instanciar objetos más allá del propio archivo, junto con la indisponibilidad de un equipo informático lo suficientemente potente como para mover la escena completa ha llevado a cambiar el procedimiento inicial con el que se ha abordado la fase de modelado.

El procedimiento finalmente pasa por crear un archivo en 3D Studio Max por cada tipo de objeto en la escena (independientemente de la planta). De esta manera se consigue tener en el mismo archivo un único objeto junto a todas sus instancias (referenciadas al objeto original), consiguiendo que la carga de un objeto determinado (por ejemplo las puertas de madera de entrada a los despachos) se realice una única vez en memoria. Esta aproximación ha permitido optimizar la velocidad de carga del modelo, disminuir el tamaño de disco consumido y rebajar las capacidades computacionales de memoria RAM y velocidad de CPU para manejar el modelo durante su desarrollo.

El primer paso realizado para la construcción del modelo ha sido la importación de los planos de planta obtenidos a través de la unidad de obras y mantenimiento de la universidad Carlos III en formato AutoCAD. Mediante la utilización de los planos se han levantado las paredes exteriores del edificio, para continuar levantando las paredes interiores de cada planta. El siguiente paso ha sido el de reconocer los elementos que se repiten con mayor frecuencia para establecer su creación en un único archivo mediante la técnica de la instanciación anteriormente explicada. En la **Figura 5** se muestran, como ejemplo, la organización y división de los objetos en distintos archivos necesarios para la recreación de la primera planta.

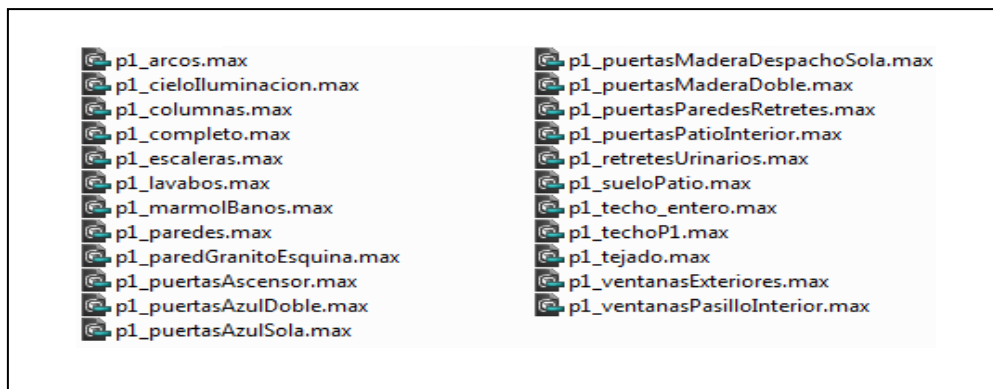


Figura 5. Separación de objetos por archivo de la primera planta.

En general, alrededor del 95% del modelo ha sido generado desde cero. El otro 5% corresponde a objetos genéricos como son bisagras, pomos u otros elementos que han sido obtenidos desde la galería 3D de Google. Esta galería dispone de miles de modelos de todo tipo de objetos que son compartidos por usuarios externos y que ponen a disposición de terceros para su uso sin restricciones. Estos modelos se encuentran en un formato propietario de Google (extensión “.skp”) pero mediante la herramienta Google SketchUp pueden ser exportados a otros formatos genéricos. En el caso del presente proyecto los objetos han sido exportados al formato 3DS para su posterior importación en 3D Studio Max. Una vez que se dispone del objeto en 3D Studio Max se ha modificado y adaptado para que se asimile lo máximo posible al objeto real del edificio Sabatini. En la **Figura 6** se muestra la incorporación de un modelo de árbol obtenido de la galería de Google SketchUp e incorporado en la escena.



Figura 6. Detalle del patio central renderizado en tiempo real por Ogre 3D.

Para la realización de los espacios de baño se ha utilizado la herramienta RocaDrawing que permite obtener modelos 3D de sanitarios de la marca Roca. La

decisión de utilizar ciertos objetos previamente modelados es doble. Por un lado se consigue mayor realismo y calidad en la escena y por otro lado se reduce parcialmente el tiempo de modelado de objetos complejos. Esta reducción de tiempo es un factor clave debido a la cantidad de objetos necesarios para recrear de manera fidedigna el edificio. En la **figura 7** puede apreciarse el resultado de incorporar las griferías y lavabos desde RocaDrawing.



Figura 7. Detalle de lavabo y grifería renderizado en tiempo real por Ogre 3D.

Con el objetivo de maximizar el rendimiento y reducir aún más el uso de recursos, se ha prestado especial atención en eliminar superficies no visibles en los distintos objetos creados. Así, por ejemplo, mientras una pared está representada mediante la creación de un cubo, tanto la parte superior como inferior del mismo no son visibles (quedan tapadas por techo y suelo) por lo que son superficies innecesarias. No tiene sentido dejar superficies no visibles, pues repercuten en el incremento de tamaño de archivo y son procesadas en la generación de texturas, iluminación así como en la visualización en el renderizado, con la correspondiente sobrecarga en los recursos computacionales usados. Es por ello, que todas las superficies no visibles han sido eliminadas de los objetos.

Aunque el objetivo principal del proyecto es el de recrear el interior del edificio Sabatini también se ha realizado una recreación básica de su exterior. Para ello se han creado fachada, puertas, ventanas, tejado, así como otros detalles generales.

La mayor problemática a la hora de realizar el exterior del edificio ha sido la recreación del tejado. No se disponía de planos y por lo tanto tampoco de medidas. Tampoco fue posible la toma de medidas en el edificio real que sirvieran de referencia para desarrollar la versión digital. Las medidas del tejado, son pues, aproximadas y su

modelado ha sido realizado utilizando como base las fotos tomadas sobre el edificio real así como las imágenes satélite ofrecidas por el servicio Google Maps [15]. En la **Figura 8** puede verse el resultado ejecutándose sobre el renderizado en tiempo real en Ogre 3D.

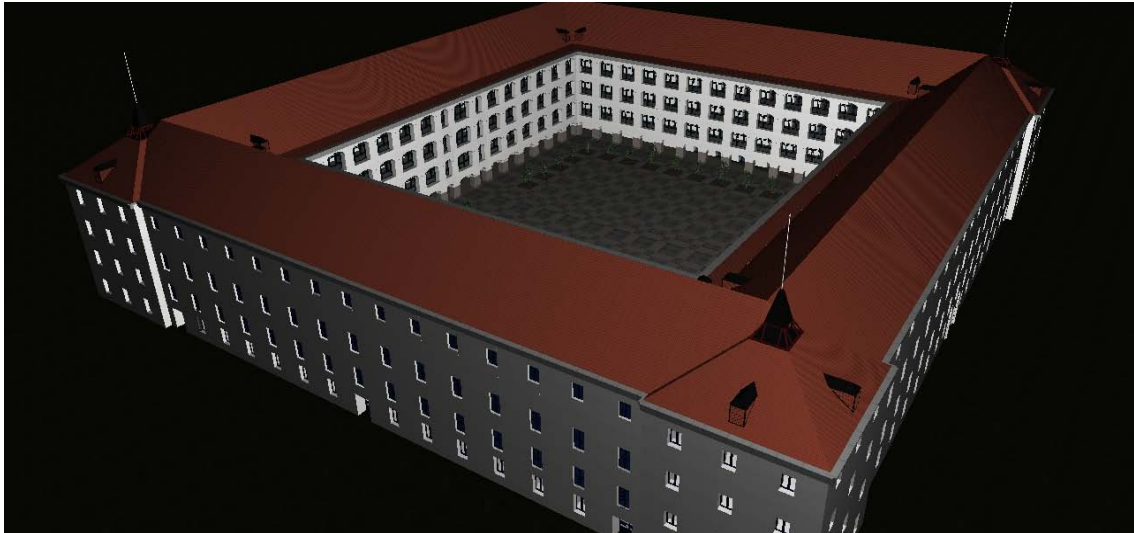


Figura 8. Detalle del tejado del edificio Sabatini renderizado en tiempo real por Ogre 3D.

Tan importante como la recreación del exterior del edificio es dotar a la escena de un entorno. Con entorno nos referimos a proporcionar la sensación de la existencia de cielo y un horizonte, una recreación que será visible tanto en la zona exterior del patio central, como a través de las puertas y ventanas exteriores.

Para este cometido se ha decidido utilizar la técnica conocida como *skydome* que consiste en la utilización de una esfera o semiesfera para la recreación del cielo. La aplicación de una textura sobre la superficie interior de la esfera crea el efecto visual de la existencia de cielo.

Para la realización del entorno del edificio Sabatini se ha combinado el uso de un *skydome* semiesférico cortado por un plano horizontal que representa el suelo. En la **Figura 9** pueden observarse como ambas superficies envuelven el edificio.

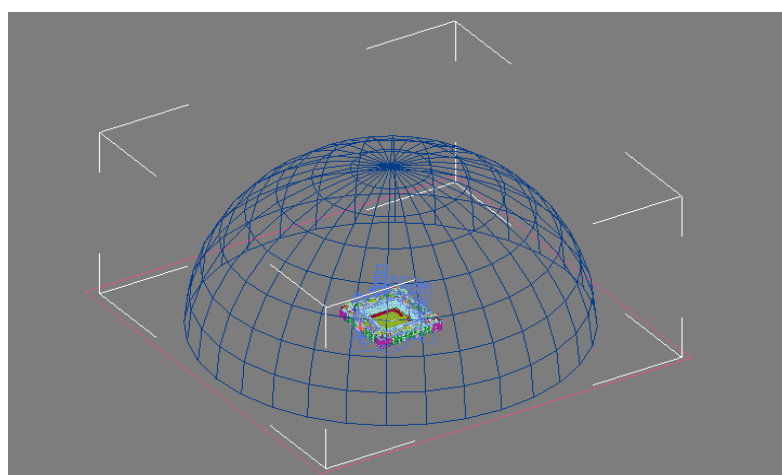


Figura 9. Recreación del entorno mediante la técnica *Skydome*.

4.1.1.5.- MAPEO Y TEXTURIZADO

Se ha comentado previamente que la etapa de mapeo y texturizado suele iniciarse y realizarse íntegramente una vez finalizada la etapa de modelado. De hecho esto es debido principalmente a que existe una persona especializada en el mapeo y texturizado, y que puede representar una etapa compleja (dependiendo del nivel de detalle que se desee obtener).

Debido a que la complejidad de texturizado que se busca alcanzar es de una complejidad media y que todo el desarrollo es realizado por una única persona se ha decidido realizar el mapeo y texturizado de manera paralela al modelado. La razón principal es por comodidad a la hora de su realización.

En el [capítulo 2](#) (estado de la cuestión) se explicó en detalle en qué consiste el mapeado. Básicamente se trata de una técnica que libera y optimiza la carga computacional requerida para el renderizado de una escena. Para cada objeto, por muy complejo que sea, se elige una figura poligonal simple que envuelve el objeto original, de este modo la textura se aplica sobre dicho polígono simplificado y el resultado es proyectado sobre la figura original. Al disminuir el número de caras sobre la que se aplica la textura se reduce también los recursos computacionales necesarios para su cálculo.

Para el mapeo de todos los objetos existentes en la escena se han utilizado tan sólo planos, cubos y cilindros. Y tan sólo una esfera para representar el *skydome* del edificio. No es necesario el uso de polígonos más complejos para dotar de realismo a la escena y de hecho el propio 3D Max Studio ofrece menos de 10 figuras poligonales para realizar el mapeo.

Una vez definida y asignada la figura poligonal usada para envolver el objeto hay que definir el tamaño del mismo, lo cual repercutirá en la visualización de la imagen usada como textura y si esta se repite o no a lo largo de la superficie del objeto. Además hay que especificar la orientación, dirección y grados de inclinación de aplicación de la textura.

Aunque el mapeo y texturizado requieren acciones bien diferenciadas es lógico y aconsejable que se realicen de manera simultánea. Las pautas y orden seguido para realizar el mapeo y texturizado han sido los siguientes.

- a) Obtención o creación de la textura a utilizar.
- b) Elección de la figura poligonal para el mapeo.
- c) Aplicación de la imagen de textura.
- d) Ajuste y modificación de los parámetros de tamaño, orientación e inclinación de la textura para un correcto ajuste y visualización.
- e) Configuración de parámetros de propiedades y características de material.

Para obtener el mayor realismo y similitud al edificio real se han realizado fotos de cada uno de los detalles del edificio para posteriormente buscar o crear texturas que sean lo más parecido posible a los materiales reales del edificio. Algunas texturas han sido obtenidas desde internet en bancos de imágenes libres de restricciones de uso. El resto han sido creadas a partir de las fotos obtenidas del propio edificio y otras tantas se han modificado y adaptado a partir de otras texturas existentes.

El primer paso consiste en el análisis de las características de los materiales del edificio original. Las fotos del edificio real tomadas permiten su utilización directa como imagen de textura o como referencia para la búsqueda de texturas lo más similares posible. En cualquier caso todas las imágenes son importadas a Adobe Photoshop para su tratamiento. Se modifica su tamaño, color y apariencia para asemejarlas lo máximo posible a los materiales originales. También se disminuye su tamaño y calidad para acelerar su carga.

El siguiente paso es estudiar la aplicación de la imagen sobre el polígono usado para el mapeo. En caso de que la imagen de textura se repita varias veces a lo largo de la superficie del polígono será necesario aplicar un tratamiento extra. Mediante el complemento Seamless Workshop se modificará la imagen para eliminar patrones irregulares en los bordes exteriores. De esta manera al repetir la imagen a lo largo de la superficie esta se mostrará de forma regular y sin saltos de patrones.

Por último, y no menos importante, es necesario especificar las propiedades del material. Determinadas características como transparencia, índice de reflexión, iluminación, etc. La especificación de estas propiedades junto con la iluminación terminará de ofrecer mayor realismo y calidad de detalle a la escena final. En la **figura 10** puede apreciarse el resultado del texturizado frente a una foto real de los materiales del edificio.



Figura10. Comparación foto real frente al modelo final texturizado.

4.1.6.- EXPORTACIÓN Y PREPARACIÓN DEL MODELO PARA SU USO EXTERNO

Una vez elaborado el modelo se ha prestado especial atención en la exportación y prueba del mismo en el entorno de Ogre 3D. De esta manera se pretende verificar y asegurar la correcta adaptación y renderizado del edificio. De igual manera, durante las fases de modelado y texturizado, se han llevado a cabo exportaciones periódicas para comprobar el peso del modelo y la carga computacional que supone para el entorno de Ogre 3D. Con estas comprobaciones periódicas se ha conseguido ajustar el compromiso entre nivel de detalle y fluidez de renderizado en el motor de Ogre 3D.

El edificio, junto con todos los objetos y detalles que lo componen, supone miles de objetos a cargar en memoria. La carga y renderizado en tiempo real del edificio completo supone un uso intensivo de los recursos computacionales. El

rendimiento y fluidez de la escena puede verse afectado seriamente en ordenadores con bajas prestaciones. Se ha comentado con anterioridad que existen varias vías para solucionar este problema, con técnicas como la denominada “*back-face culling*”. La implantación de estas técnicas deben ser realizadas desde el propio motor de renderizado Ogre 3D y por lo tanto escapan a los límites del presente proyecto.

Sin embargo, para ofrecer al usuario final la posibilidad de visualizar y recorrer el edificio como si de una visita virtual se tratase se ha dividido el modelo en plantas. De esta manera se han realizado cuatro exportaciones y por lo tanto generado cuatro escenas en formato DotScene que representan cada una de las plantas del edificio Sabatini.

De esta manera se reduce la necesidad de recursos computacionales necesarios para el renderizado en tiempo real sin la necesidad de implementar técnicas de optimización dentro del entorno Ogre 3D. Todas las escenas generadas comparten algunos elementos comunes, como es las paredes exteriores, tejado y ventanas las cuales se visualizan a través del patio interior. Esto crea una duplicidad que provoca la necesidad de mayor espacio en disco duro, sin embargo, es perfectamente asumible dado las ventajas de fluidez en la ejecución que representan.

Para la exportación del modelo al formato DotScene (el formato manejado por Ogre 3D) se ha utilizado la herramienta OgreMax 3D Max Scene Exporter la cual pone a disposición un complemento para 3D Max Studio. Este complemento permite la exportación directa desde el formato nativo de 3D Max Studio así como la posibilidad de ajustar y configurar diversos parámetros para generar la escena.

Para la correcta visualización de la escena es necesario, además de exportar el propio modelo, configurar y establecer las cámaras de visualización así como la iluminación.

En lo que respecta a las cámaras es indispensable, al menos, el establecimiento de una por escena. La cámara especifica la posición inicial y enfoque de la vista. En el caso de Ogre 3D establece el punto inicial desde el que se inicia la visita virtual por el edificio.

Para cada una de las escenas exportadas (una por cada planta) se han establecido seis cámaras esparcidas a lo largo del interior y exterior del edificio. Durante el recorrido podrán cambiarse libremente de cámara de manera que permite visualizar las zonas o detalles más relevantes de cada planta sin necesidad de recorrerla entera.

El establecimiento y configuración de las cámaras se realiza desde el propio entorno de 3D Studio Max y son reconocidas y exportadas a Ogre 3D mediante el complemento OgreMax 3D Max Scene Exporter. En la **Figura 11** se muestran el establecimiento de las cámaras de la primera planta.

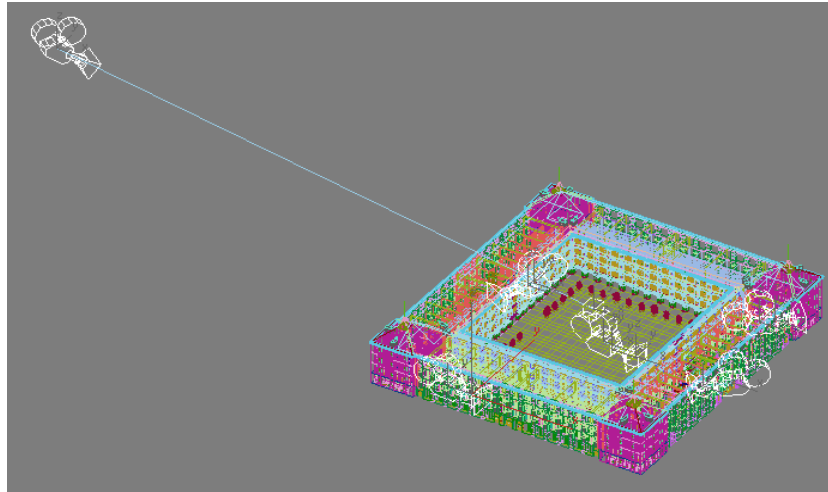


Figura 11. Situación de las cámaras de la primera planta.

En lo que a iluminación respecta se ha abordado anteriormente que es aconsejable la elección e integración de un motor de iluminación en el entorno Ogre 3D. Del mismo modo, es necesaria la preparación específica del modelo para el motor de iluminación escogido y las técnicas que éste permita para su uso. Sin embargo, para ofrecer una visualización básica y gracias a que el visor Ogre Max Viewer permite la configuración básica de iluminación se han incluido un par de puntos de iluminación para alumbrar la escena y producir sombras simples. La iluminación de una escena puede suponer una fase completamente diferenciada y puede consumir bastantes recursos humanos y temporales dependiendo del compromiso de calidad que se quiera obtener. Debido a que el uso del modelo en el entorno de renderizado Ogre 3D queda fuera del ámbito del proyecto se ha considerado sólo la inclusión de una iluminación general para todo el edificio. Esta iluminación básica consta de dos focos de luz cuya situación puede apreciarse en la **Figura 12** en color amarillo y cuyo radio de acción queda reflejado por el perímetro encerrado por las líneas de color marrón.

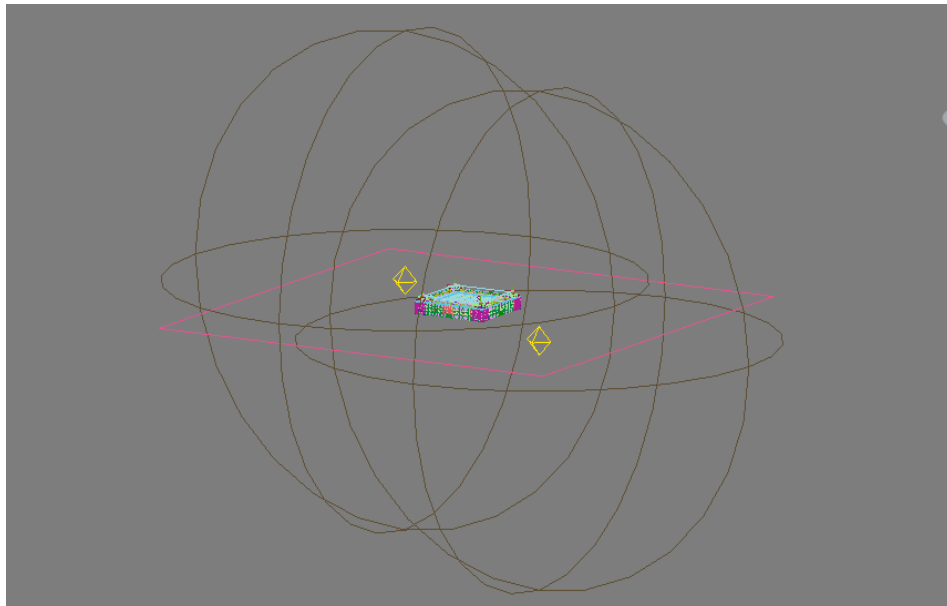


Figura 12. Focos de iluminación de la escena.

4.2.- MANUAL DE USUARIO

4.2.1.- SISTEMA OPERATIVO WINDOWS

La ejecución de la escena desde un entorno Windows es rápida y sencilla. No es necesario instalación de ningún software de terceros. El único requisito es disponer de un ordenador con Windows 2000, XP o Vista que disponga de las librerías OpenGL o Direct 3D.

Si se necesita más información sobre OpenGL o Direct 3D se puede acudir a las siguientes direcciones web donde se puede descargar el software necesario así como obtener diversa documentación.

- www.opengl.org
- <http://www.gamesforwindows.com/en-US/AboutGFW/Pages/DirectX10.aspx>

En el directorio raíz del fichero entregado existen cuatro archivos ejecutables que corresponden a cada una de las plantas del edificio. Para ejecutar una de ellas simplemente habrá que realizar doble clic sobre el archivo requerido y esperar a que se cargue la escena.

La ejecución de la escena se realiza, por defecto, sobre OpenGL, si se desea que la ejecución se realiza sobre Direct3D será necesario editar los archivos ejecutables mediante el uso de cualquier herramienta de edición de texto. Además de elegir entre OpenGL o Direct3D se pueden modificar las siguientes opciones:

OPCIONES

- **CameraMoveSpeed** - Establece la velocidad, en unidades por segundo, de la cámara controlada por el usuario.
- **Width** – Especifica el ancho de la ventana de visualización.
- **Height** – Especifica el alto de la ventana de visualización.
- **FullScreen** - Indica que el visor será ejecutado a pantalla completa.
- **Direct3D o -D3D** – Indica que el sistema de renderizado a utilizar sea Direct 3D.
- **OpenGL o -OGL** – Indica que el sistema de renderizado a utilizar sea OpenGL.
- **VSync** – Indica que la sincronización vertical sea activada.

Una vez ejecutado el archivo deseado y tras finalizar la carga de la escena (el tiempo de carga puede variar dependiendo del ordenador) es posible moverse a través de la escena mediante los siguientes controles:

CONTROLES

- **Teclado**

Desplazamiento - Mediante las flechas de dirección o las teclas W, A, S y D se puede moverse hacia adelante o hacia atrás, así como realizar desplazamientos laterales.

Salir – Mediante la tecla Esc.

Cambiar cámara – Con las teclas 1-6 se puede seleccionar la cámara de visualización.

- **Ratón**

Giro – Con el desplazamiento del ratón se puede cambiar la dirección de desplazamiento y visualización.

Desplazamiento – Con el botón central del ratón se puede avanzar.

Rueda - Si se dispone de rueda de desplazamiento se puede mover la cámara hacia arriba o hacia abajo.

4.2.2.- OTROS SISTEMAS OPERATIVOS

Aunque el proyecto se ha enfocado para su uso en el sistema operativo Windows también es posible su ejecución en otros sistemas operativos. Para ello es necesario obtener la versión de Ogre 3D correspondiente y configurar un nuevo proyecto en el que se han de incluir las librerías de OgreMax y cargar la escena. No es un proceso directo y requiere conocimientos básicos de programación para su configuración y ejecución, por ello esta opción no es recomendada para usuarios con un perfil de conocimientos informáticos bajo.

En el caso de Mac Os existe una versión pre-compilada del SDK (Kit de Desarrollo de Software) lista para su uso. En el caso de Linux será necesario obtener el código fuente y compilarla en la versión de Linux deseada.

Para obtener la correspondiente versión de Ogre 3D para el S.O. (Sistema Operativo) deseado se puede realizar desde la misma página oficial del proyecto (www.ogre3d.org). Para crear un proyecto en Ogre 3D con las librerías de OgreMax y cargar la escena a visualizar es necesario disponer de ciertos conocimientos de programación y realizar ciertas configuraciones del entorno. Desde la página oficial de OgreMax (www.ogremax.com) se puede obtener la documentación e información necesaria para su ejecución sobre Linux o Mac Os.

4.3.- MANUAL DE REFERENCIA

Lo primero que hay que tener en cuenta es que existen dos versiones del modelo, ambas realizadas en 3D Studio Max 2009. Es importante recalcar que será necesaria la versión 2009 para abrir los ficheros pues no es compatible con versiones anteriores de 3D Studio Max.

La versión que se encuentra bajo la carpeta “modelo/3dMax/Generico” está texturizada para permitir su uso genérico en cualquier tipo de desarrollo. Esta versión es apta tanto para su modificación directa en 3D Studio Max 2009, así como su exportación a cualquier otro formato genérico o propietario para ser usado posteriormente en cualquier herramienta de modelado. Para ello se puede descargar

gratuitamente la versión de prueba de 3D Studio Max 2009 y utilizar las propias herramientas de exportación que este software pone a disposición del usuario.

La segunda versión del modelo se encuentra bajo la carpeta “modelo/3dMax/Ogre” y está texturizada y optimizada para su exportación a formato DotScene y posterior uso en el motor de renderizado Ogre 3D. Para abrir los archivos y modificarlos es necesario utilizar la versión 2009 de 3d Studio Max así como tener instalado el complemento 3D Studio Max Scene Exporter de OgreMax (www.ogremax.org). Esta última herramienta permite el uso de texturas en 3D Studio Max compatibles con Ogre 3D así como la exportación al formato DotScene. Para mantener la mayor flexibilidad y sencillez posible únicamente se han utilizado los materiales de OgreMax para recrear las transparencias de los cristales en el modelo, el resto de objetos contienen están texturizados mediante el uso de los propios materiales de 3D Max Studio.

Tanto la versión genérica del modelo como la versión específica para Ogre Max vienen subdivididas en carpetas. La carpeta “comun” contiene el modelo entero dividido según los objetos. En las carpetas que van desde “p1” a “p4” contienen en su interior el modelo específico que representa cada una de las plantas del edificio. Al igual que en la carpeta “comun” cada planta viene subdividida según el tipo de objeto de la escena. En las carpetas de cada una de las plantas también es posible encontrar el archivo “x_completo.max” que contiene el modelo de la planta complet. No es recomendable el uso de este último a no ser que se disponga de un equipo informático potente capaz de mover tal cantidad de objetos.

En el caso de no ser necesaria la modificación del modelo éste se puede utilizar e incorporar en el desarrollo mediante la utilización de la escena en formato DotScene que se encuentra bajo la carpeta “modelo/escena”, esta escena contiene tanto el modelo como el mapeo y texturizado. También es posible obtener el modelo en cualquier otro tipo de formato a partir de los ficheros originales de 3D Studio Max y mediante el uso de herramientas de exportación específicas.

Las texturas pueden encontrarse bajo la carpeta “texturas” y en cada una de las subcarpetas de cada planta que se encuentran bajo el directorio “modelo/escena”. Estas texturas son utilizadas por el visor y cargadas al iniciar el ejecutable correspondiente a cada planta.

En el [capítulo 2](#) (Estado de la cuestión) así como en la [sección 4.2.2](#) del presente capítulo se puede encontrar un análisis y justificación detallada de cada una de las técnicas y herramientas utilizadas en el desarrollo y cuya lectura es recomendable antes de la utilización del modelo para un nuevo desarrollo. La información de

referencia necesaria para la utilización o ampliación del presente proyecto queda completada en el [capítulo 8](#). En ese capítulo se exponen los posibles campos de ampliación y mejora del producto obtenido así como posibles vías de implementación y uso del modelo en futuros desarrollos.

5.- RESULTADOS

Como resultado del desarrollo se ha obtenido todos los productos requeridos, cumpliendo con los objetivos impuestos y prestando especial atención en obtener un resultado funcional y de calidad.

El requisito esencial era la recreación del edificio Sabatini de la UC3M situado en el campus de Leganés. El modelo obtenido representa fielmente el edificio real. Se han realizado toma de medidas para la construcción a escala del edificio preservando las relaciones de tamaño originales. La realización de fotos y estudio del edificio han permitido la reproducción de los espacios, mobiliario y detalles con gran similitud a los originales. La creación de las imágenes de texturas se han basado y creado respetando los materiales originales de construcción.

Para asegurar la compatibilidad y adaptación del modelo para que pueda ser manejado por Ogre 3D se ha realizado un completo estudio de las técnicas y herramientas existentes en el mercado para generar un producto que satisfaga las necesidades requeridas. Como resultado se ha generado el modelo según el formato y peculiaridades requeridas por Ogre 3D. El producto resultante es íntegramente gestionable por Ogre 3D el cual a su vez dispone de librerías para el manejo de la escena completa.

Para asegurar la perfecta visualización y correcto rendimiento de la escena en Ogre 3D, se ha implementado un proyecto que permite el renderizado en tiempo real del edificio mostrando, a su vez, un recorrido virtual del mismo. De esta manera queda totalmente asegurada la compatibilidad del producto obtenido para su uso e incorporación en cualquier solución desarrollada en Ogre 3D y por lo tanto en el videojuego propuesto.

Más allá de los requisitos impuestos se ha buscado en todo momento la obtención de un producto funcional, flexible, optimizado y de calidad. Cada uno de estos objetivos ha sido considerado en cada una de las etapas de desarrollo y los resultados obtenidos en el producto final se traducen en los siguientes puntos:

Funcionalidad

El modelo del edificio Sabatini busca, ante todo, ser funcional. Prueba de ello ha sido la realización de un visor que permite recorrer de manera virtual el edificio. Todos los productos obtenidos están libres de restricciones legales de uso. Desde un principio se ha querido generar un producto que fuese funcional con independencia de terceros desarrollos. El máximo potencial del modelo se obtiene al adaptarlo y optimizarlo para su uso específico en futuros desarrollos, sin embargo, el producto obtenido en este desarrollo dispone de entidad propia.

Flexibilidad

La utilización de técnicas simples y estandarizadas, el uso de formatos genéricos y la utilización de herramientas ampliamente aceptadas repercuten en la obtención de un producto final flexible y adaptable a un amplio rango de desarrollos de distinta naturaleza. Como punto negativo, al desechar el uso de técnicas complejas y específicas de herramientas como 3D Studio Max la calidad de detalle ha quedado acotada en un nivel inferior al potencialmente obtenible.

La preparación del modelo para su uso optimizado en el entorno Ogre 3D choca con el objetivo de ofrecer un producto flexible y adaptable. Para satisfacer ambas necesidades se ha generado dos versiones del mismo modelo. Una versión específica y optimizada para su uso en Ogre 3D y otra versión genérica en la que se potencia su adaptabilidad y flexibilidad.

Optimización

El modelado y diseño 3D es un campo que consume gran cantidad de recursos computacionales. La optimización del modelo ha sido un punto muy tenido en cuenta. Entre la versión inicial y la versión final de producción se ha conseguido reducir el tamaño resultante entre un 50 y un 60%. En esta reducción de tamaño la calidad de detalle en la escena apenas se ha visto influenciada. El uso de técnicas como la instanciación de objetos, utilización de mapeo, texturas de repetición, simplificación del modelo o reducción al máximo posible del número de polígonos en la creación de objetos han sido claves para alcanzar el resultado deseado. El modelo contiene la información y polígonos exclusivamente necesarios abogando por un tamaño contenido. Esto permite el renderizado en tiempo real de la escena con un consumo y necesidad de recursos computacionales lo más contenidos posible.

Calidad

La calidad de detalle del modelo es un objetivo en confrontación directa con el de optimización. Para obtener un buen nivel de detalle en el modelo se ha prestado especial atención en el respeto de las medidas originales. Una recreación fidedigna del mobiliario y detalles del edificio así como el especial cuidado en la realización de texturizado han sido factores clave para aportar un mayor realismo a la escena. La inclusión de iluminación y la generación de sombras junto con la recreación de un entorno exterior (cielo y suelo) terminan por dotar a la escena del nivel de detalle y calidad buscados.

Para finalizar se ha habilitado una página web pública con toda la información sobre el proyecto así como la posibilidad de descargar la documentación y el modelo generado. Esta página web queda abierta al público en general y sin restricciones de acceso en la siguiente dirección:

<http://pfc.alexissanchez.ez>

6.- CONCLUSIONES

6.1.- CONCLUSIONES SOBRE EL DESARROLLO

El diseño 3D es un campo complejo y cambiante. Las técnicas que se utilizaban en el pasado están desfasadas, del mismo modo que las que actualmente se utilizan serán reemplazadas por nuevas técnicas más avanzadas que obtengan mejores resultados. Esto, junto a la gran cantidad de herramientas disponibles en el sector y el hecho de que cada solución de software ponga a disposición técnicas propietarias, hace que la elección del entorno sea crucial. La elección de una herramienta y formato de publicación particular limita los resultados que se pueden obtener. En este sentido la etapa de [“estudio y comparativa de las técnicas y herramientas de desarrollo”](#) ha resultado ser crucial para la obtención de un modelo optimizado para su uso bajo el entorno Ogre 3D y que además fuese adaptable y flexible a futuros desarrollos.

El proceso de modelado ha supuesto un proceso costoso y largo, además ha sido necesario prestar especial atención a su optimización para minimizar el uso de recursos computacionales. El renderizado en tiempo real de una escena del tamaño del edificio Sabatini conlleva importantes problemáticas de optimización. Ha sido necesario acudir a diversas técnicas así como prestar una especial atención a la simplificación de objetos utilizando el menor número de polígonos para su representación. Aún así ha sido necesaria la separación del modelo en plantas y es recomendable el uso de técnicas de optimización específicas al usar el modelo dentro del entorno de programación y motor de renderizado seleccionado.

Las fases de texturizado e iluminación han resultado claves para dotar a la escena de un mayor realismo y similitud con el edificio real. El uso de técnicas como el mapeo o la utilización de texturas de repetición han permitido disminuir el coste computacional necesario para su procesamiento. El uso de una iluminación general basada únicamente en dos focos supone una simplificación importante ya que el cálculo de iluminación y generación de sombras supone un alto coste computacional en un renderizado en tiempo real.

El uso de técnicas genéricas y simples, la utilización de formatos genéricos y el uso de herramientas libres de restricción de uso dotan al modelo de una alta flexibilidad y adaptabilidad. Sin embargo, al no hacer uso de las técnicas específicas y

propietarias del software utilizado la calidad de detalle y optimización de la escena han quedado acotados en un nivel inferior al potencialmente obtenible.

6.2 CONCLUSIONES PERSONALES

En el ámbito personal el presente proyecto me ha permitido obtener una mayor experiencia en el campo del diseño 3D. Había tenido experiencias previas pero nunca antes me había enfrentado a un proyecto de esta envergadura. El estudio previo para la elección de las herramientas y técnicas que más se ajustaban a las necesidades del proyecto me ha aportado un mayor conocimiento. Así mismo ha repercutido en la formación de un criterio de elección asentado sobre bases sólidas y un verdadero conocimiento de las distintas vías disponibles para la consecución de un proyecto de modelado 3D de gran envergadura.

El mayor reto personal al que me he tenido que enfrentar dentro del desarrollo ha sido la creación del modelo tratando de minimizar el uso de recursos así como de lograr la mayor flexibilidad y adaptabilidad posible. A mitad del modelado he tenido que desechar la metodología inicial para abordar el desarrollo mediante un cambio de técnica. Este cambio de técnica se debió a la imposibilidad de continuar por la vía inicial a falta de un equipo con los recursos suficientes como para mover la escena completa en tiempo real. El modelo que estaba generando era muy pesado y poco optimizado por lo que el uso de recursos computacionales era muy superior al esperado.

Como experiencia personal la realización de este proyecto ha supuesto una experiencia muy gratificante y didáctica. He aumentando ampliamente mis conocimientos en la materia y ha supuesto un reto personal en la búsqueda por la obtención de un resultado que reflejase la madurez y experiencia adquirida durante todos estos años de formación académica.

7.- FUTURAS LÍNEAS DE TRABAJO

7.1.- MODELADO

En lo que al modelado se refiere hay varias líneas de trabajo que podrían ser abordadas. En primer lugar no se ha realizado el sótano, la torre de ventilación lateral ni tampoco la zona exterior colindante al edificio (zonas ajardinadas, vallas, etc.).

Mientras que puertas, ventanas, techados, espacios sanitarios y otros detalles interiores han sido recreados no se ha incluido el mobiliario propiamente dicho; sillas, muebles y demás objetos. Esta podría ser, junto con la finalización del edificio el siguiente paso lógico para dar continuación a este trabajo en su fase de modelado.

7.2.- MAPEO Y TEXTURIZADO

Atendiendo al compromiso de obtener un resultado altamente adaptable a cualquier uso futuro se han desechado la utilización de técnicas avanzadas de texturizado. Si se quiere utilizar el edificio en un entorno y motor de renderizado específico tendría lógica aprovechar al máximo las funcionalidades compatibles dentro del marco de trabajo. Por ejemplo se podría aplicar reflexión en espejos, o técnicas como “sombreado en texturas”.

7.3.- MOTOR DE RENDERIZADO

Este es, sin lugar a dudas, el apartado donde más trabajo puede llevarse a cabo. Para aportar cierta independencia al proyecto y la posibilidad de visualizar de manera virtual el edificio se ha usado el visualizador que OgreMax trae por defecto. Este visor que funciona bajo Ogre 3D es bastante básico. Quedaría pendiente la optimización del cargador de escenas, para poder visualizar las distintas plantas y zonas del edificio de manera dinámica, consiguiendo así rebajar el uso de memoria y consumo de recursos.

Para adaptar el modelo como un escenario en un videojuego es necesaria la implementación de un sistema de colisiones bajo un motor de físicas. El sistema de colisiones establecería los volúmenes de colisión. Un volumen de colisión es representado por una pared o el suelo y especifica los volúmenes no pueden ser traspasados por los personajes u objetos incluidos en la escena. Además, del sistema de colisiones también es necesaria la configuración del motor de físicas escogido para la simulación de las fuerzas de gravedad.

Aunque se ha incluido una iluminación en el modelo ésta es muy básica. La elección de un motor de iluminación y la utilización de técnicas avanzadas de iluminación pueden mejorar en gran medida la calidad y realismo de la escena. Además, sólo se han utilizado dos focos generadores de luz, lo idóneo sería la recreación de la iluminación real del edificio estableciendo y configurando una iluminación general recreando la luz solar así como todos y cada uno de los focos de luz artificial existentes en el edificio.

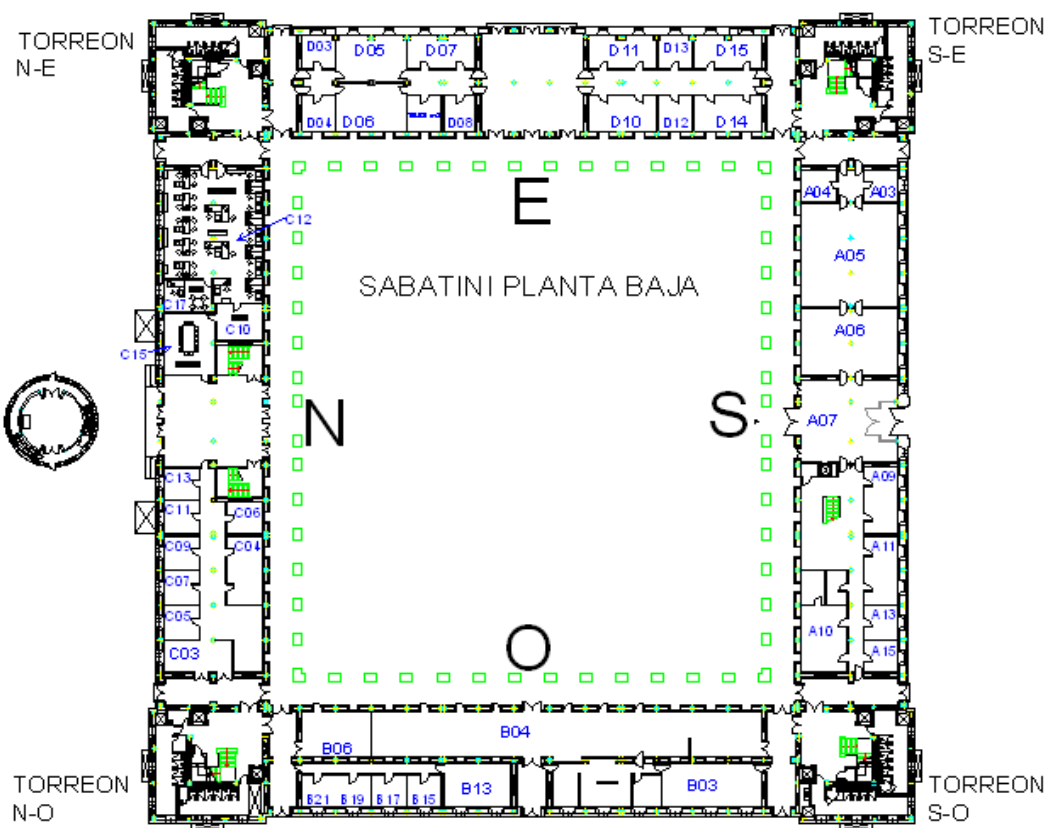
Por último, sería interesante la inclusión de eventos y animaciones en la escena. Así, por ejemplo, se podría implementar las funcionalidades necesarias para actuar con los objetos del edificio y recibir una respuesta animada del mismo. Ejemplos de la implementación de estas funcionalidades podría ser la apertura o cierre de puertas, encendido o apagado de luces, posibilidad de mover el mobiliario, etc.

BIBLIOGRAFÍA

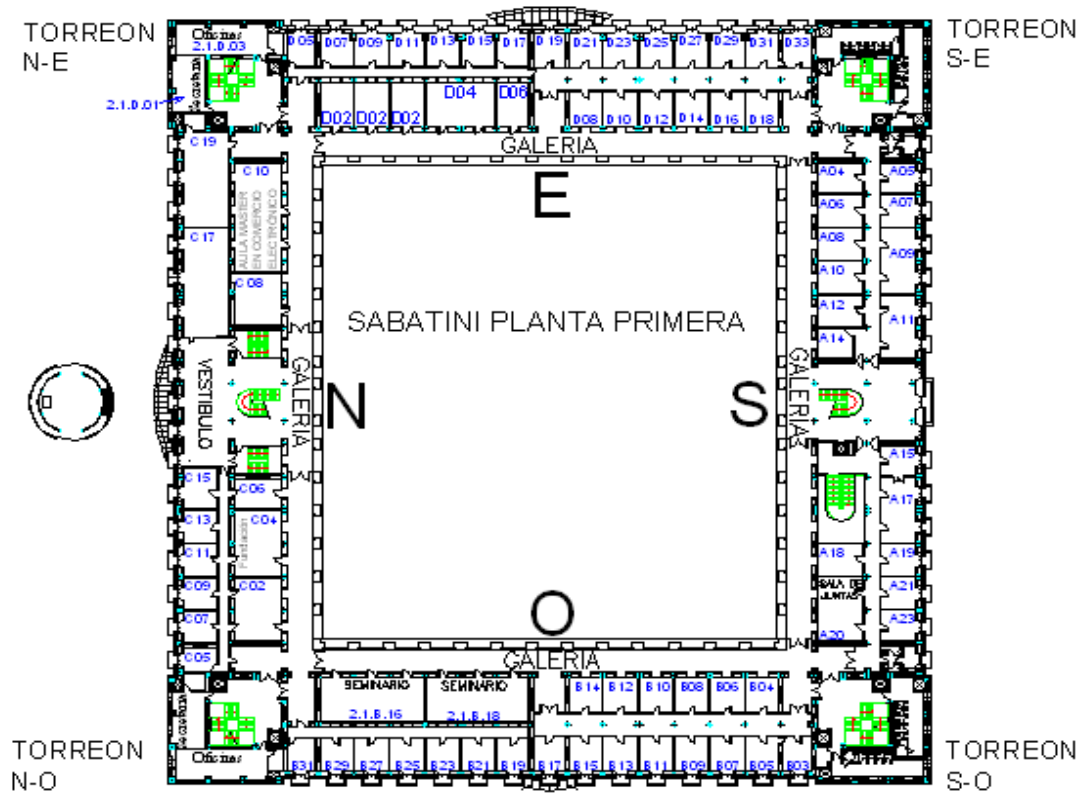
- [1] **Wright, Will.** Los Sims, 2000.
- [2] Grupo de investigación en Planificación y Aprendizaje Automático.
- [3] <http://www.autodesk.es>
- [4] <http://www.ogre3d.org>
- [5] <http://www.ogre3d.org/wiki/index.php/DotScene>
- [6] Referencia del manual de usuario de 3D Max Studio. 2002.
- [7] **Todd, Michael.** 3D Studio Max. Prentice Hall, 1997.
- [8] **Nedelman's, Derek.** Manual de referencia de Ogre Max 3D Scene Exporter.
- [9] <http://www.redfieldplugins.com>
- [10] Referencia de la A.P.I. de Ogre 3D. version 1.6.
- [11] **Junker, Gregory.** Pro Ogre 3D Programming. Apress. 2006
- [12] <http://sketchup.google.com>
- [13] <http://sketchup.google.com/3dwarehouse>
- [14] http://www.roca.es/contents/menus/menu_san_sp/aplicaciones/rocadwg.htm
- [15] <http://maps.google.es>

ANEXO A. PLANOS DE AUTOCAD

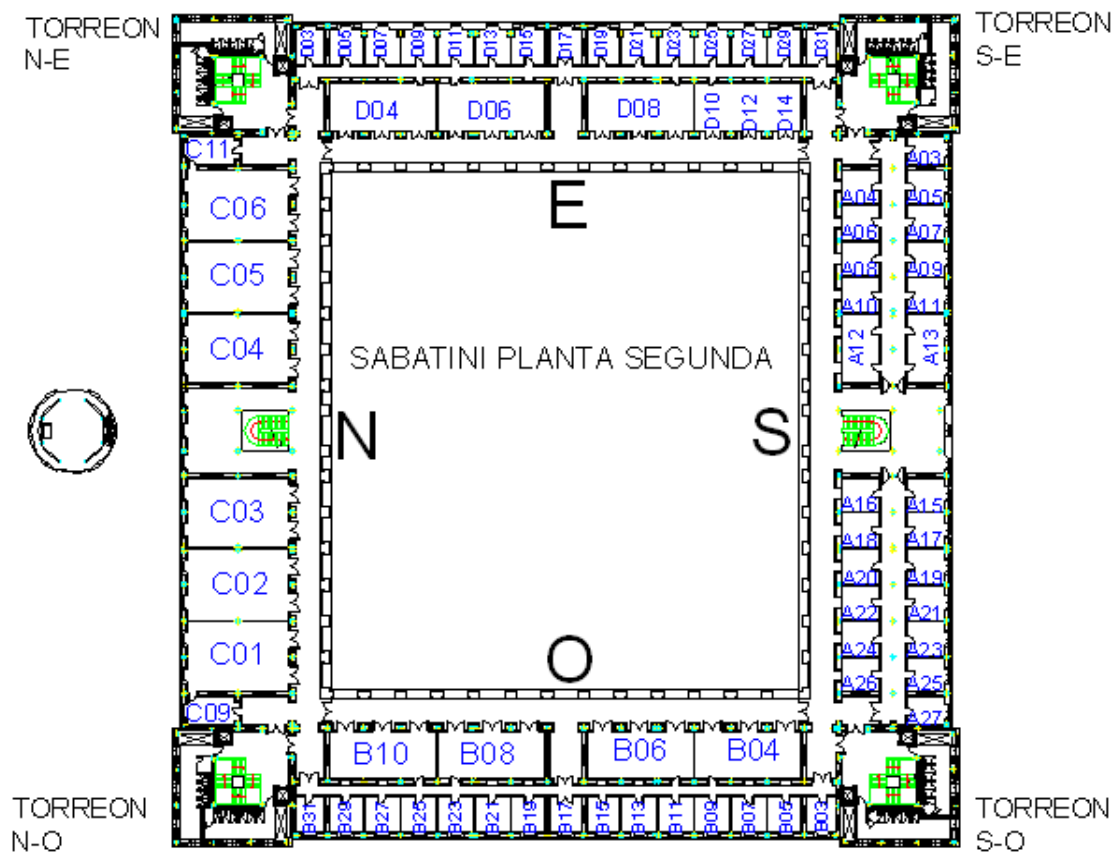
A.1. PLANTA BAJA



A.2. PLANTA PRIMERA



A.3. PLANTA SEGUNDA



A.4. PLANTA TERCERA



El objetivo del proyecto es la realización de un modelo tridimensional del edificio Sabatini de la Universidad Carlos III de Madrid (UC3M) situado en el campus de Leganés. Para su realización se han utilizado técnicas de diseño e infoarquitectura 3D aplicadas al campo de la computación.

El proyecto abarca todas y cada una de las fases de desarrollo; desde la toma de medidas, hasta su integración final en un motor de renderizado en tiempo real, pasando por las fases de modelado, iluminación, texturizado, etc.

La finalidad es que el modelo pueda ser integrado en terceras soluciones. Además, se incluye una implementación que ofrece un recorrido virtual controlado por el usuario y que permite visualizar el resultado obtenido.

Más allá de buscar una solución específica para su uso como escenario virtual, se ha pretendido generar un producto genérico y libre de dependencias. Se ha prestado especial atención en maximizar la flexibilidad y adaptabilidad para su uso en futuros desarrollos externos no contemplados y se ha buscado la mayor fidelidad posible respecto a la arquitectura original.

Alexis Sánchez López

Más información en:

<http://pfc.alexissanchez.es>