



Universidad de León



Ingeniería Técnica Industrial

Modelado de la Escuela de Ingenierías, Edificio Tecnológico y su entorno en Unity3d

Modeling School of Engineering, Technological Building and environment in Unity3d

ESCUELA DE INGENIERÍAS INDUSTRIAL, INFORMÁTICA Y AEORNÁUTICA

UNIVERSIDAD DE LEÓN

TUTOR: D. Fernando Jorge Fraile Fernández
AUTOR: Carlos Miguel Fernández Fernández

Este documento cuyo título es **Modelado de la Escuela de Ingenierías, Edificio Tecnológico y su entorno en Unity3d** constituye el Trabajo Fin de Carrera de **Carlos Miguel Fernández Fernández**, alumno de la Escuela de Ingenierías Industrial, Informática y Aeronáutica de León, con el objetivo de obtener el título de Ingeniero Técnico Industrial en la especialidad de Estructuras.

La tutoría de este trabajo ha sido llevada a cabo por D. Fernando Jorge Fraile Fernández, profesor del Departamento de Química, Física y Expresión Gráfica de la Universidad de León.

Vº Bº Oficina Técnica:

Tutor:

Fdo.: D. Manuel Castejón Limas

Fdo.: D. Fernando Jorge Fraile Fernández

Autor:

Fdo.: Carlos Miguel Fernández Fernández

En el proyecto que se va a desarrollar a continuación se va a explicar la metodología a seguir para realizar un paseo virtual por la Escuela de Ingenierías y por el Edificio Tecnológico.

Para ello se detallará el proceso de recopilación y actualización de datos utilizados, la creación y modelado de los recintos necesarios y la construcción a partir del programa Unity3d de la aplicación que nos permita este paseo virtual.

Para finalizar se comprobará si los objetivos planteados al principio se han llevado a cabo con éxito.

El trabajo incluirá adjunta una base de datos con los archivos de *AutoCAD* utilizados para su realización, incluyendo la descripción de cada capa y la textura utilizada en el paseo virtual.

ÍNDICE

INDICE

0 INTRODUCCIÓN	1
1 OBJETIVOS	9
2 METODOLOGÍA	12
2.1 Estudio de posibilidades	13
2.1.1 <i>SecondLife</i>	14
2.1.2 <i>Open Simulator</i>	15
2.1.3 <i>RealXtend</i>	15
2.1.4 <i>Taiga/Naali</i>	16
2.1.5 <i>Unity3d</i>	17
2.2 Método de trabajo	19
2.3 Trabajo con <i>AutoCAD</i>	20
2.3.1 Revisión de modelos anteriores	20
2.3.2 Modelado de nuevas zonas	25
2.3.2.1 Modelado del Salón de Actos	25
2.3.2.2 Modelado de Biblioteca	30
2.4 Trabajo con <i>3ds Max</i>	35
2.5 Creación de texturas y materiales	38
2.5.1 Texturas y materiales	38
2.5.2 Creación de texturas	40
2.6 Trabajo en <i>Unity3d</i>	45
2.6.1 Introducción a <i>Unity3d</i>	45
2.6.2 Interfaz gráfica del programa	46
2.6.3 Trabajo en una sola escena	47
2.6.4 División del proyecto en escenas	51
2.6.5 Funcionamiento general de Unity	53
2.6.6 Proceso de creación de escenas	63
2.6.6.1 Escena 01 Exterior	64
2.6.6.2 Escena 02a Escuela P. Baja	73
2.6.6.3 Escena 02b Escuela P. Primera ...	80
2.6.6.4 Escena 03 Edificio Tecnológico ...	81
2.6.6.5 Escena 04 Salón de Actos	85
2.6.6.6 Escena 05 Biblioteca	90
2.6.6.7 Escenas Restantes	95
2.6.7 Programación	103
2.6.8 Creación de la aplicación final	105
3.- CONCLUSIONES	108
4.- BIBLIOGRAFÍA	110
4.1 Libros	111
4.2 Proyectos	111
4.3 Páginas web	112
ANEXO A Base de datos. Archivos <i>AutoCAD</i>	113
A.1 Exterior	114
A.2 Escuela de Ingenierías	116
A.3 Edificio tecnológico	127
ANEXO B Instrumentos utilizados.....	130
B.1 Hardware.....	131

B.2 Software	131
B.3 Otros instrumentos	131
ANEXO C Scripts	132
C.1 Cambio de escenas	133
C.2 Apertura de puertas	134
ANEXO D Comparativa Realidad/Unity3d/RealXtend	136

0 INTRODUCCIÓN

0 INTRODUCCIÓN

Un **MUNDO VIRTUAL** es una simulación de un entorno físico real o imaginario creado a partir de modelos en tres dimensiones, en donde los usuarios pueden interactuar entre sí y con los objetos y bienes pertenecientes a este mundo en tiempo real.

Para que una representación de un entorno físico sea considerada como mundo virtual ha de estar activo y disponible para los usuarios continuamente.

Los mundos virtuales actualmente están diseñados para el ocio, ya que económicamente es la mejor forma de obtener un rendimiento, aunque hay también mundos virtuales con funciones educativas y profesionales. Los MMOLE (Massively Multilearner Online Learning Environments) son mundos virtuales diseñados para la enseñanza online. Entre otros podemos citar *Sloodle* (www.sloodle.org) que es un proyecto gratuito y de código abierto que integra varios mundos virtuales dedicados a la enseñanza o *ArsMetaverse* (<http://artsmetaverse.arts.ubc.ca/>) creado por la universidad de Columbia, que simula varios entornos reales como una réplica del Machu Pichu o la misma Universidad de Columbia.



Figura 0-01: *Sloodle*

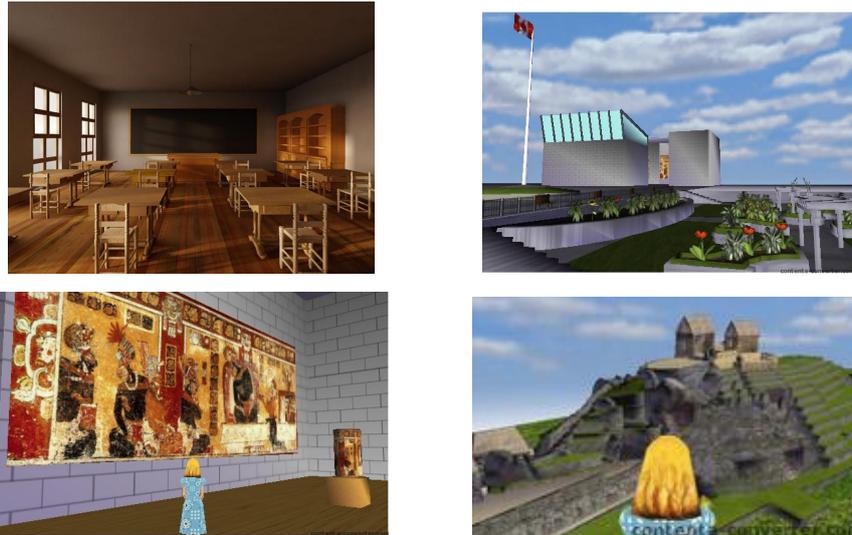


Figura 0-02: Imágenes de los recintos virtuales en ArsMetaverse

Entre los mundos virtuales dedicados al ocio podemos distinguir los comerciales y los libres. Entre los comerciales, el más conocido es *Second Life* y entre los libres, el paradigma es *Open Simulator*.

En mundos virtuales comerciales como *Second Life*, además de una cuota mensual para poder ser usuario, se han de pagar tasas por la obtención de terrenos y la construcción de edificaciones en ellos. Esto es una consecuencia lógica de la necesidad de alojar una cantidad enorme de objetos, datos y usuarios en los servidores de *Second Life*.

Siguiendo la filosofía de los programas de código abierto aparece *OpenSim*, que es una plataforma gratuita para la creación de mundos virtuales, es muy parecido a *Second Life* pero con la diferencia fundamental de que el usuario es el que aloja en su servidor el mundo que construye. Por un lado esto lo convierte en gratuito, pero por otro limita la visita del mundo virtual a los usuarios que tengan permiso específico del creador del mundo.

Hablaremos con más detalle de estas plataformas más adelante.



Figura 0-03: OpenSim

En la Escuela de Ingenierías Industrial, Informática y Aeronáutica se llevan realizando desde hace unos años varios proyectos dirigidos por Fernando Jorge Fraile, profesor del Departamento de Química, Física y Expresión gráfica con características similares. Estos trabajos están enfocados a la creación de un mundo virtual que simule los edificios pertenecientes a esta escuela.

En septiembre de 2000 se presentó un Modelado del campus en 3d realizado con *AutoCAD* y *3ds Max*. Dentro de este proyecto se incluía la fachada de la Escuela y la del Edificio Tecnológico en detalle. El objetivo de este proyecto consistía en la realización de una serie de videos de paseos aéreos sobre el campus y más detenidamente sobre la Escuela de Ingenierías y el Edificio Tecnológico. En la figura 0-04 podemos apreciar los resultados obtenidos.

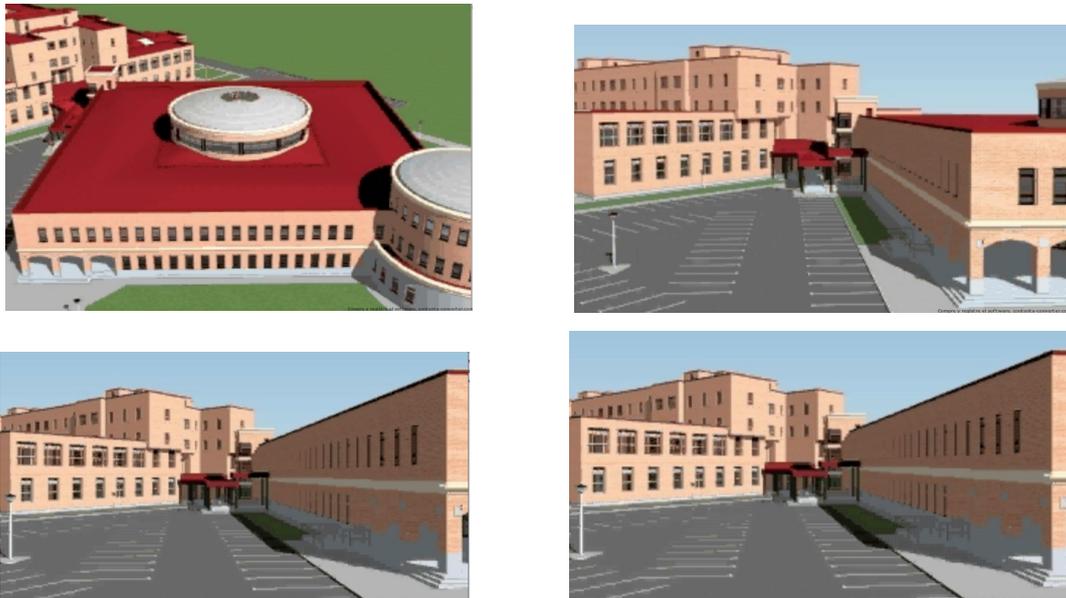


Figura 0-04: Imágenes de los videos realizados el 2000

Debido a que en aquella época los ordenadores domésticos tenían rendimientos muy bajos en comparación con los de ahora, los modelados no podían ir mucho más allá. Hasta la aparición de procesadores de doble núcleo y tarjetas gráficas más potentes a disposición de los alumnos, el embarcarse en proyectos de modelado gráfico era casi impensable.

Entre 2007 y 2008, ya disponiendo de mejores recursos de hardware y software, se realizó el modelado de la Escuela de Ingenierías en detalle incluyendo varios recintos del interior: el hall, la tabiquería de la planta baja, las escaleras y los pasillos de la primera planta. La presentación de estos modelos se realizó mediante videos de paseos en escenas del interior de

la Escuela, de forma similar al anterior trabajo. En este caso, a los objetos se les había aplicado materiales de forma bastante realista mediante *3ds Max*.

Vemos imágenes capturadas de los videos realizados en la Figura 0-05



Figura 0-05: Imágenes capturadas de los videos de 2008

En febrero de 2008 se realizó el modelado del Aula 1 y se ampliaron los accesorios del hall. Hasta este momento el objetivo de los proyectos se limitaba al modelado y aplicación de texturas y materiales realistas.

En Junio de 2009 se comienza con la idea de un mundo virtual que simule la escuela. Además de la corrección y actualización de lo hecho hasta ahora, se inicia la construcción del mundo virtual en *RealXtend*. *RealXtend* es una plataforma para la creación de mundos virtuales de código abierto. Esta basado en *OpenSim*, esto es, el creador del mundo virtual lo aloja en su

propio servidor y da permiso a determinados usuarios para que entren en su mundo. Se construyó el hall de la escuela, el aula 2 y el aula 216. El objetivo del proyecto era la realización de un paseo virtual en estos tres recintos a través del visor de *RealXtend*, incluyendo pequeñas animaciones de apertura de puertas y ventanas.



Figura 0-06: Imágenes del paseo virtual en *RealXtend*

Conjuntamente con este proyecto se realizan otros dos trabajos que lo completan: por un lado la creación de un aula virtual donde exista interactividad entre profesor y alumnos y por otro la virtualización administrativa del centro.

Para realizar el aula virtual se modeló con detalle el aula 204 (aula de dibujo) que serviría de base para las actividades de enseñanza/aprendizaje en el mundo virtual. Para la virtualización de la parte administrativa del centro se modelaron la secretaría y la conserjería. Todos los recintos se realizaron con gran detalle. A la hora de incluirlos en RealXtend y aplicar texturas y materiales estuvieron limitados por las posibilidades gráficas de esta plataforma.

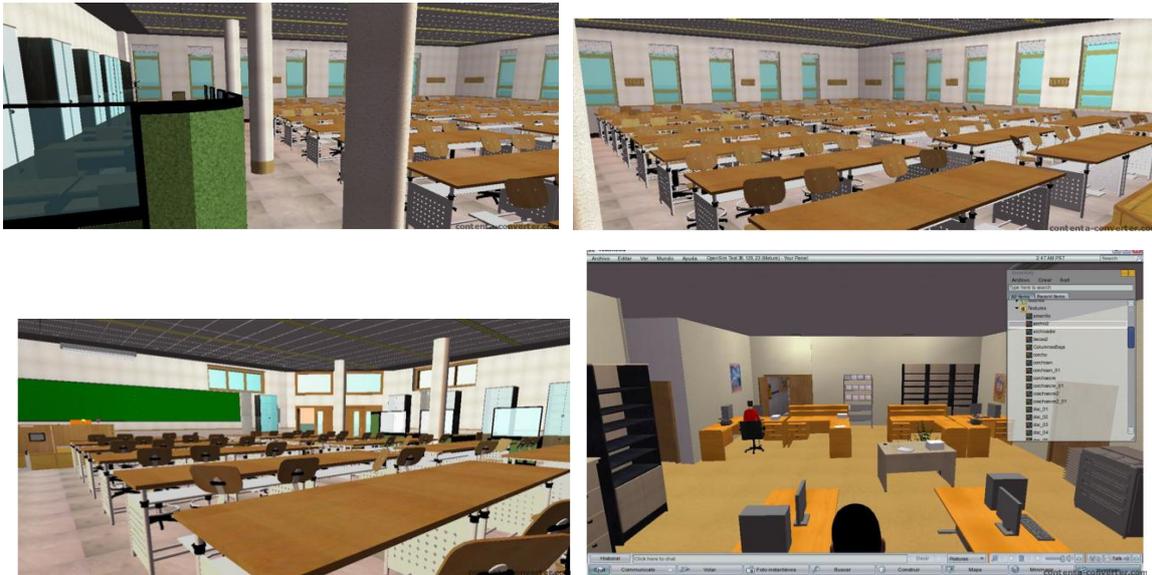


Figura 0-07: Imágenes del aula 204 y la secretaría en RealXtend

Hasta este momento era posible el paseo por hall, aulas y algún recinto en la escuela, la interacción entre varias personas mediante chat hablado y escrito, y la interacción con objetos de la secretaría como impresos de matricula y diversos tipos de solicitudes. La única limitación para una puesta en marcha definitiva vendría implícita en la misma naturaleza de RealXtend: al servidor solo se puede acceder con el permiso del administrador de este mundo, y esto es difícil de unificar ya que cada trabajo se aloja en un servidor distinto. Eso se complica aun más si se quiere poner en funcionamiento en red, y más aun si se quiere abrir para el uso público en la Universidad.

En un trabajo paralelo en 2009 se realizó el modelado del Edificio Tecnológico, incluyendo además de hall, pasillos y zonas comunes, el laboratorio H3 y un baño de la planta inferior. También se construyó otro mundo virtual con *RealXtend* para la posibilidad de realizar paseos virtuales. Dada la complejidad del diseño del edificio los resultados del modelado fueron sorprendentes.

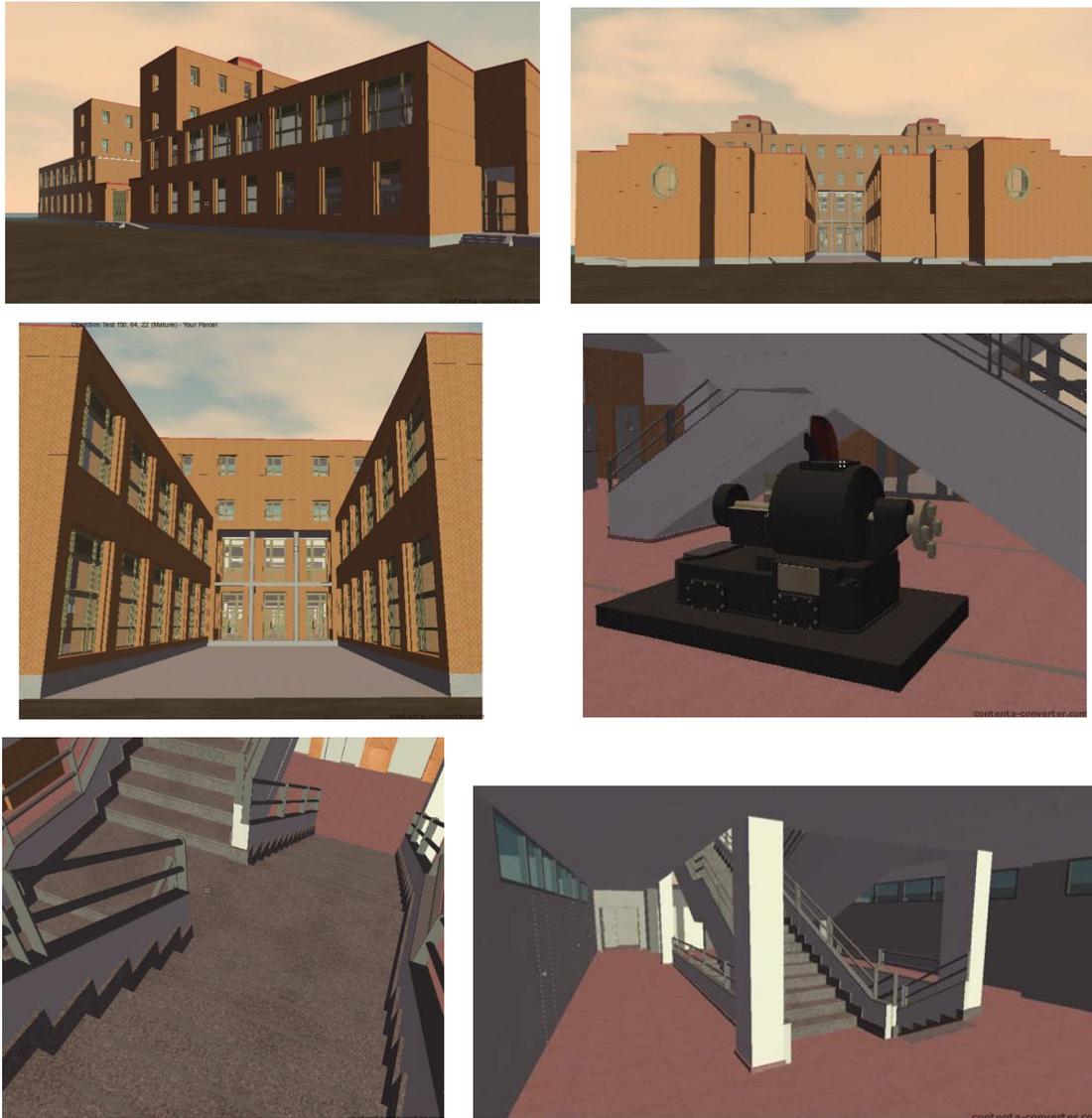


Figura 0-08: Imágenes del Edificio Tecnológico en RealXtend

Poco a poco se va viendo más cerca la posibilidad de crear un mundo virtual en la Escuela de Ingenierías de León, para que tanto alumnos como profesores puedan interactuar entre ellos y con los objetos del mundo y facilitar así labores educativas y administrativas en el centro. El trabajo que se presenta a continuación es un paso más para lograr el objetivo final de la creación de una Escuela de Ingenierías Industrial, Informática y Aeronáutica totalmente virtualizada.

1 OBJETIVOS

1 OBJETIVOS

El objetivo final y principal del trabajo será crear un mundo virtual con los dos edificios de la Escuela de Ingenierías Industrial, Informática y Aeronáutica y el terreno exterior colindante incluyendo los recintos comunes representativos como son la secretaría, el salón de actos, la biblioteca y distintas aulas. Éste ha de ser lo más realista posible, para ello se buscará la mayor exactitud en el dimensionado, la elección de texturas y materiales, la iluminación y sombreado y la representación de mínimos detalles. La pretensión es que si el trabajo se aloja en un servidor de la red, cualquier persona autorizada pueda acceder a este mundo desde cualquier lugar a través de internet.



Figura 1-01: Escuela de Ingenierías y Edificio Tecnológico



Figura 1-02: Unión entre los dos edificios

Para lograr este objetivo se va a partir de modelados realizados en 3d algunos de los cuales ya están hechos en trabajos anteriores y se van a incluir en una plataforma que permita la creación de un mundo virtual interactivo para los visitantes exteriores.

Para ello se seguirán los siguientes pasos:

- 1) Se completarán los trabajos realizados hasta el momento modelando las zonas comunes de los edificios que no han sido realizados aun, esto es: el salón de actos y la biblioteca.
- 2) Se actualizarán los modelados existentes con las modificaciones que se han llevado a cabo en los edificios desde que se realizaron hasta la actualidad.
- 3) Una vez completados los dibujos se pretende englobar todos ellos igualando escalas, unidades y otras propiedades para poder crear una base de datos con los archivos utilizados, la descripción del contenido, las texturas y materiales usados en cada objeto y otros datos de interés con el fin de ayudar en lo posible a posteriores trabajos.
- 4) El paso definitivo será la realización del mundo virtual en una plataforma que reúna las mejores condiciones para su visualización.

2 METODOLOGÍA

2 METODOLOGÍA

Para llevar a cabo los objetivos con éxito el proceso a seguir será el siguiente:

1. Estudio de posibilidades
2. Creación de un método de trabajo
3. Trabajo con *AutoCAD*
4. Trabajo con *3dStudio*
5. Creación de texturas, uso de *PixPlant*
6. Trabajo con *Unity3d*

2.1 Estudio de posibilidades

Antes de empezar a modelar hemos de decidir que herramientas vamos a utilizar teniendo en cuenta las necesidades que tenemos y las restricciones fundamentalmente económicas que un trabajo de este tipo supone. Dependiendo de los programas usados se afrontará el trabajo de una forma u otra. Principalmente vamos a realizar dos trabajos diferenciados: el modelado de los objetos 3d que usaremos para la creación del mundo y la creación de este mundo por medio de una plataforma a partir de los objetos modelados.

Para completar los trabajos anteriores y modelar los recintos nuevos se va a usar AutoCad ya que es el programa con el que se ha trabajado anteriormente, y es el disponible en la escuela. Este programa nos permite realizar cualquier objeto en 3d con exactitud dimensional mediante la creación de sólidos básicos, extrusión de planos 2d, barrido de objetos 2d siguiendo trayectorias y creación de sólidos de revolución. Estas operaciones se detallarán en capítulos posteriores.

Para la elección de un motor gráfico o una plataforma que reúna las cualidades necesarias para la obtención del objetivo vamos a estudiar las siguientes opciones:

1. *Second Life* (SL) – Plataforma más extendida de mundos virtuales.
2. *OpenSim* (OS) – Plataforma de código abierto similar a *Second Life*
3. *RealXtend* (RX) – Plataforma basada en OS, con la posibilidad de importación de meshes y scripts.
4. *Taiga/Naali* – Nueva versión dentro de RX con mayores posibilidades de introducción de mallas o meshes, texturas e iluminación. Plataforma más realista.
5. *Unity* – Motor gráfico para creación de aplicaciones interactivas, visualizaciones y animaciones en 3D y tiempo real.

Los criterios para la elección serán los siguientes:

- Posibilidades económicas.
- Nivel de calidad suficiente para una simulación lo más realista posible.
- Buen funcionamiento de programas de código abierto.
- Rendimiento de equipos y de redes.

2.1.1 *Second Life*

Second Life es un mundo virtual donde se pueden adquirir parcelas y dentro de ellas se pueden construir objetos con ciertas limitaciones.

Para nuestro trabajo esta plataforma se descarta ya que:

- Solo se pueden usar prims o primitivas para la construcción y a pesar de que existen plug-ins para transformar los archivos .dwg de *AutoCAD* en objetos reconocibles por SL esto requeriría un proceso largo y laborioso en el que tendríamos como resultado una cantidad de objetos que excederían ampliamente el límite de prims por parcela impuesto en SL
- Cada parcela tiene unos costes tanto de adquisición como de mantenimiento mensual con limitación de primitivas a usar que no son asumibles para la realización de este trabajo.



Figura 2.1-01: Imagen de *SecondLife*

2.1.2 Open Simulator

OpenSim es una plataforma de código abierto para la realización de mundos virtuales de forma privada. Se asemeja a *SecondLife* pero en este caso el servidor donde se alojan las parcelas es privado, cada usuario tiene su servidor y solo tiene acceso a las parcelas que se alojen en éste las personas autorizadas.

Está limitado al uso de primitivas básicas con lo que no es una opción para nuestro proyecto.



Figura 2.1-02: *OpenSim*

2.1.3 RealXtend

RealXtend es una plataforma basada en *OpenSim* con varias mejoras. Los objetos para construir siguen siendo prims como en OS pero en este caso es relativamente sencillo pasar archivos .dwg de *AutoCAD* a .mesh que son objetos reconocibles por *RealXtend*, mediante un paso por el formato .3ds. Los archivos .mesh se pueden agregar a una prim tomando ésta la forma de el dibujo original realizado en *AutoCad*. En cada servidor se pueden incluir un numero de prims solo limitado por la capacidad física del equipo usado. Para mostrar los resultados obtenidos no es necesaria una conexión continua a una red externa.

También permite el uso de scripts o pequeñas programaciones en lenguaje *python* que nos facilitan en gran medida la construcción de nuestro mundo

En un principio esta sería la mejor elección ya que reúne las principales condiciones: es gratuito, nos da la posibilidad de incluir todo lo construido por *AutoCad*, podemos usar texturas a partir de fotografías de los objetos dibujados que nos da una sensación de realismo muy aceptable.

Por todo ello ésta ha sido la elección realizada en los trabajos anteriores a éste. Los creadores de RX a mediados del 2010, cuando se comienza a plantear éste trabajo diseñaron

una nueva plataforma de mundos virtuales con un servidor llamado *Taiga* y un visor llamado *Naali* que nos ofrecía unas opciones mucho mas favorables que la plataforma anterior.



Figura 2.1-03: Visor de RealXtend

2.1.4 *Taiga/Naali*

Taiga es el nuevo servidor para mundos virtuales creado por *RealXtend*, muy similar a éste pero con mayores detalles en cuanto al realismo de escenas y mayor facilidad de importación de objetos 3d creados con otros programas.

Naali es el visor creado para *Taiga* con el que se puede editar gráficamente lo alojado en el servidor.

Como novedades con respecto a la anterior plataforma tenemos la posibilidad de incluir texturas con mayor calidad, iluminación con creación de sombras mucho más realista y la posibilidad de importar escenas completas desde programas como *3ds Max* con sus texturas correspondientes, lo que facilitaría mucho el trabajo. Gracias a esta herramienta el objetivo principal de crear en un mismo mundo los dos edificios con los recintos comunes incluidos empezó a verse factible dada la cantidad y complejidad de objetos 3d existentes y los que quedaban por dibujar.

El problema de estas herramientas de código abierto es que por lo general les falta una continuidad en la elaboración. En un periodo relativamente corto de tiempo se fueron actualizando versiones de *Naali* que iban corrigiendo problemas y adaptando la compatibilidad entre éste visor y los modeladores mas usados como *Blender*, *Maya* o *3ds Max*, pero a partir de enero de 2011 el equipo de *RealXtend* comenzó a trabajar en *Tundra*, que es otra nueva plataforma de código abierto que reúne el servidor y el visor en un mismo programa. *Tundra*

salió en su primera versión en Mayo de 2011 con lo que no se ha estudiado su viabilidad ya que no había tiempo material para poder presentar este trabajo en el plazo requerido.

El principal problema que nos dio *Naali* es que no daba posibilidad de crear permisos en el mundo virtual que creamos, esto es, que cualquier usuario que entrara en nuestro mundo lo podría modificar con el visor *Naali*. En RX esto lo solucionábamos dando permisos en la edición.

Otro fallo importante es que la importación de escenas completas con sus texturas desde 3d Studio se realizaba con un plug-in de código abierto llamado *ogremax* que exportaba desde 3d Studio la escena directamente con todos sus objetos en formato .mesh. Estos meshes (o mallas) se exportaban en una versión actualizada del formato y *Naali* solo fue capaz de reconocer una versión antigua del formato .mesh, nos llevó mucho esfuerzo y contactos conseguir una versión antigua del *ogremax*.

También nos dieron problemas las teleportaciones entre un mundo y otro. La idea original era la de realizar varios mundos cada uno con una parte del trabajo con idea de no cargar demasiado el servidor y que el rendimiento fuera óptimo para poder tener movimientos fluidos en el mundo virtual. Si *Naali* no nos permite viajar fácilmente de un mundo a otro el proyecto no podría realizarse con fluidez.



Figura 2.1-04: *Naali*

2.1.5 Unity3d

Unity3d es un motor gráfico que nos permite crear aplicaciones 3d, animaciones y visualizaciones. No es en sí una plataforma de mundos virtuales, lo que nos permite Unity3d es crear aplicaciones para diversos sistemas operativos que contienen estas visualizaciones en 3d. También nos permite publicar en una página web el trabajo realizado de forma que es posible verlo en cualquier navegador con un sencillo plug-in gratuito, el *Unity web player*. Básicamente

este motor gráfico es usado para la creación de videojuegos interactivos en 3d en primera persona pero dadas sus características lo podemos usar para nuestros fines ya que cumple casi todos los requisitos que exigíamos a la plataforma buscada:

- *Unity3d* es gratuito, los creadores de la plataforma distribuyen una versión de pago, el *Unity3d pro*, que incluye herramientas más avanzadas para creación de sombras y texturas, filtros de sonido y posibilidad de creación de aplicaciones para sistemas como *Android* o *iOS* para smartphones y tablets más avanzados. La versión gratuita cumple ampliamente los requisitos buscados para el trabajo a realizar.

- Podemos importar a *Unity* cualquier objeto 3d realizado con los modeladores más usados. La calidad de gráficos es muy superior a cualquier plataforma estudiada hasta el momento, la sensación de realismo va a ser por lo tanto muy superior. Con esta herramienta podremos modificar texturas e iluminación de una forma impensable en plataformas del tipo *RealXtend*.

- Con este programa no vamos a tener los problemas de una creación de código abierto experimental, El motor gráfico se está convirtiendo en uno de los más usados por creadores de videojuegos a niveles básicos con lo que se está continuamente actualizando y corrigiendo los posibles fallos.

- La aplicación que vamos a crear exigirá lógicamente unos requisitos mínimos de equipo para ser reproducida. Estas exigencias dependerán de la resolución y del nivel de calidad en el renderizado, Unity nos da la posibilidad elegir un nivel de detalles menor en la visualización para un mejor funcionamiento de equipos con poca memoria gráfica o niveles de visualización excelentes para equipos más potentes.

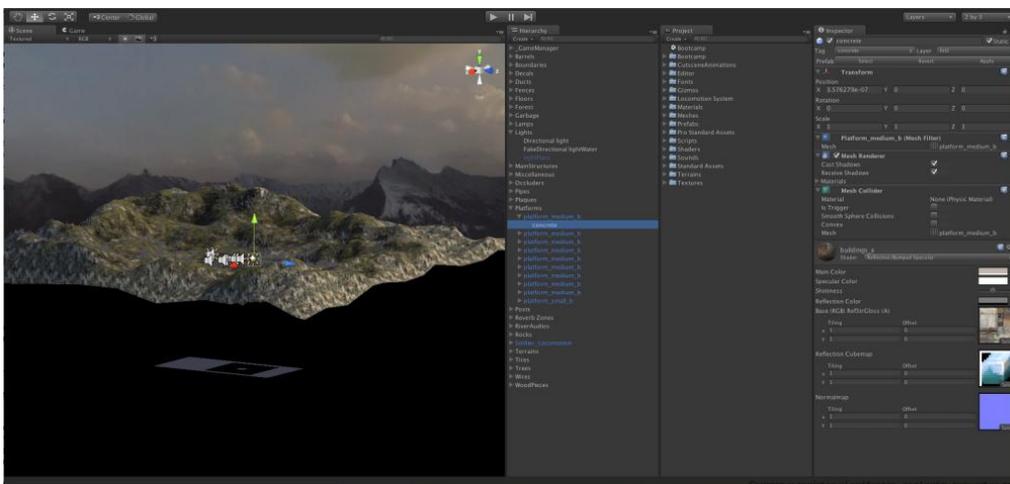


Figura 2.1-05: Interfaz gráfica de Unity

En todos estos aspectos el uso del motor *Unity3d* es muy superior a lo visto anteriormente pero por otro lado el uso de *Unity* nos va a limitar sobre todo en lo referente a la interactividad. Las plataformas de mundos virtuales se basan en la interactividad de los usuarios y ya traen por defecto herramientas que posibilitan la comunicación entre usuarios y la interacción de éstos en el mundo, en nuestro caso la creación de aulas virtuales con múltiples usuarios conectados a la vez.

En *Unity3d* es posible la creación de escenas multi-jugador que asemejarían esta interacción de usuarios entre si, pero el nivel de conocimientos necesarios para esta creación requiere un tiempo de estudio del programa que excedería el plazo fijado para este trabajo. Dado que en otros aspectos el motor es muy superior a las plataformas de creación de mundos virtuales de código abierto hemos decidido centrarnos en la creación del mundo virtual para posibles visitas o paseos virtuales y dejar abierta la creación de este espacio interactivo para posteriores trabajos.

2.2 Método de trabajo

Una vez decididas las herramientas a usar hemos de crear una línea de trabajo. Para la creación del mundo virtual en Unity necesitamos todos los objetos que queramos importar en un formato .fbx, éste es un formato estándar de modelos 3d reconocido por programas como *Maya*, *3ds Max* o *Blender* pero *AutoCAD* no es capaz de exportar directamente en este formato. Eso si, mediante 3d Studio vamos a poder importar las escenas .dwg de *AutoCad* y exportar en .fbx. Como conclusión, necesitamos todos los modelos 3d en archivos .dwg de *AutoCAD*. Aunque Unity nos da la posibilidad de cambios de escala, movimientos y giros para el montaje de las escenas, para una mayor exactitud y dado que los objetos que voy a usar han sido modelados por distintas personas que no seguían un método de trabajo uniforme, he decidido hacer previamente un trabajo de organización e igualación de criterios en todos los dibujos anteriores. De esta forma se conseguirán varios objetivos:

- Objetos con las mismas proporciones y unidades.
- Normalización de nombres de capas y texturas a usar.
- Puntos de referencia comunes a todos los objetos colindantes para facilitar el montaje.
- Facilidad de actualización de estructuras modificadas recientemente
- Creación de un punto base para la realización de los recintos que quedan por modelar
- Visualización global del trabajo.

Teniendo en cuenta todo esto podemos seguir la siguiente línea de trabajo:

1- Trabajo en *AutoCAD*:

- Revisión y homogenización de los archivos de otros trabajos realizados anteriormente.
- Modelado de los recintos que faltan y actualización de lo dibujado anteriormente.

2- Trabajo con *3ds Max*:

- Conversión de todas las escenas a formato .fbx mediante el paso por *3ds Max*.

3- Creación de texturas y materiales.

4- Trabajo con *Unity3d*:

- Montaje en *Unity3d* de las escenas con sus texturas correspondientes
- Creación de los scripts necesarios en las escenas creadas
- Creación de la aplicación final ya sea para su ejecución en modo standalone para *Windows* o *MacOSX* o bien para su uso desde una página web como página html.

Aunque visto así parece un trabajo ordenado cronológicamente una vez que se empieza a trabajar es fácil entender que es un proceso en continua retroalimentación: Una vez montada una escena en *Unity3d* te puedes dar cuenta de errores que no se ven claros en *AutoCAD* y hay que repetir el proceso desde el inicio al menos con los objetos erróneos.

2.3 Trabajo con *AutoCAD*

AutoCAD es un programa de diseño asistido por ordenador (CAD) en dos y tres dimensiones. Es quizá el programa más extendido para creación y diseño de planos en estudios de ingeniería y arquitectura.

Con este programa vamos a realizar dos tareas fundamentalmente:

- 1- Revisión, normalización, actualización y corrección de errores en los modelados existentes
- 2- Modelado de Salón de Actos, Biblioteca, zonas exteriores y todos los elementos necesarios para la actualización de los modelos anteriores.

2.3.1 Revisión de modelos anteriores

Desde el año 2000 diversos alumnos de esta escuela han presentado varios proyectos para los cuales han necesitado modelar partes de los edificios. Cada uno ha hecho los diseños

pensando en los objetivos de cada proyecto, no se ha seguido una directriz de trabajo ni una normativa determinada. Lo que se pretende en esta parte del trabajo es modificar todos los archivos de forma que se puedan utilizaren este proyecto y en proyectos posteriores relacionados con éste.

Lo primero es marcarse ciertas directrices:

- a. Escala – Vamos a trabajar en metros a escala natural (1:1), con unidad de medida en metros, dadas las dimensiones del proyecto.
- b. Los objetos de texturas y materiales iguales en cada archivo los englobaremos en capas diferenciadas
- c. Se usarán puntos de referencia iguales en la mayoría de los archivos principales. En caso de que sea más sencillo mantener las referencias originales, estas quedarán debidamente detalladas. Para casos de objetos repetidos como puertas, sillas y mesas, bancos o papeleras se usará como punto base el más característico dentro del plano donde esté apoyado el objeto.
- d. Solo vamos a usar solidos 3d. Se eliminarán de los archivos todos los objetos 2d innecesarios (líneas, arcos, polilíneas, planos...) y se llevarán a capas auxiliares los objetos 2d necesarios para la comprensión del dibujo como los planos originales. Al nombrar estas capas auxiliares siempre comenzarán con “00” para distinguirlas perfectamente de las capas restantes.
- e. Los archivos de *AutoCAD* .dwg no superarán un tamaño de 15 MB excepto en casos muy concretos. Los archivos más pesados hacen que se ralentice la apertura y trabajo con *AutoCAD*. Estos archivos quedarán claramente denominados con el nombre del edificio al que pertenecen y el nombre de su contenido.
- f. Todos los archivos se guardarán en formato .dwg 2007 para asegurar la compatibilidad con otros programas.

Para realizar esta tarea en primer lugar se han seleccionado los archivos a utilizar usando como criterios:

- Los archivos que completen al máximo los edificios.
- Los archivos más actualizados
- Los archivos con mayor nivel de detalle

En cada archivo se han creado las capas necesarias o renombrado las existentes incluyendo en cada una los objetos con el mismo material.

Una vez organizadas las capas se procede a la eliminación de objetos innecesarios como líneas auxiliares que puedan provocar errores a la hora de cambiar de formato en otros programas. Esto lo hacemos con la herramienta de selección rápida eliminando o trasladando a capas auxiliares todo lo que no sean sólidos 3d o en casos muy determinados sólidos de revolución. Los planos y las mallas indispensables para la comprensión del dibujo las transformaremos en sólidos 3d ya que en muchos programas de modelado a la hora de poner texturas éstas solo se representan en una de las caras de la malla. Para esta transformación usaremos la herramienta de extrusión de planos usando trayectorias en casos de planos inclinados.

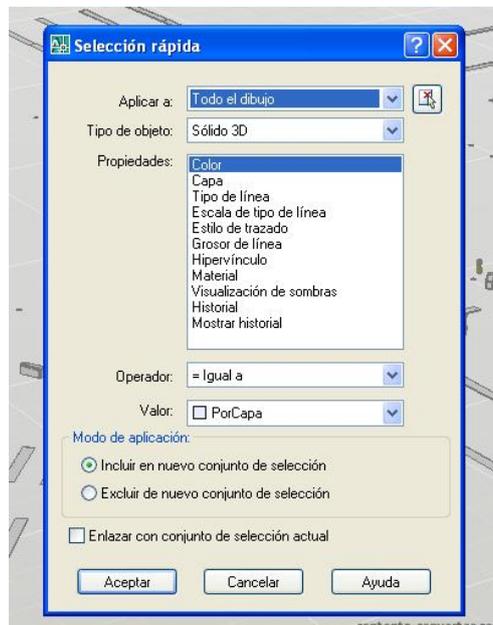


Figura 2.3-01: Herramienta de selección rápida

Dado que en muchos programas puede haber incompatibilidades a la hora de trabajar con bloques se van a eliminar todos los bloques usados transformando estos en sólidos 3d. En caso de múltiples repeticiones se creará un archivo con el elemento repetido para que el archivo inicial no supere el tamaño buscado.

Mediante la orden *limpia* se eliminarán los bloques, capas y otros objetos no utilizados en cada archivo.

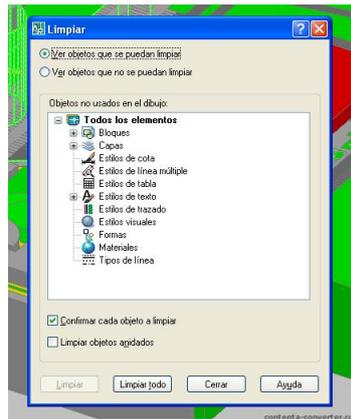


Figura 2.3-02: Orden Limpia

Una vez limpios los archivos y organizados se procederá al montaje dividiendo el trabajo en tres grandes bloques:

Exterior – Zona exterior de los dos edificios incluyendo las estructuras de los edificios colindantes y las estructuras visibles de la escuela de ingenierías y el edificio tecnológico. El punto de referencia (0,0,0) será la esquina de la base inferior izquierda del edificio tecnológico (capa “Muro gris”) más próxima a la escuela de ingenierías.

Edificio Tecnológico – Estructura e interior de las 5 plantas del edificio tecnológico. El punto de referencia en este caso será el mismo que en el exterior.

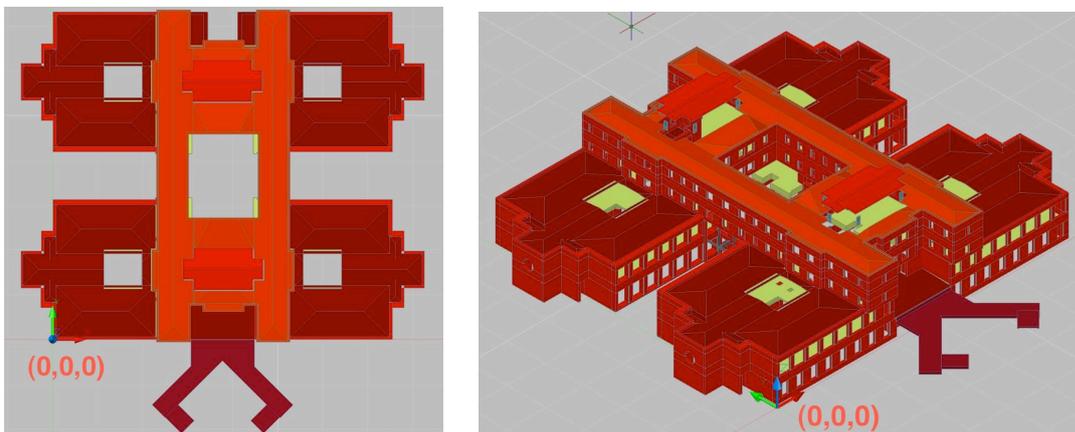


Figura 2.3-03: Referencia en Exterior y Edificio tecnológico

Escuela – Estructura exterior e interior de la Escuela de Ingenierías Industrial, Informática y Aeronáutica. Se incluyen el Aula 4, el Aula de dibujo 204, la conserjería y la secretaría. Como referencia el punto (0,0,0) será el punto inferior de la base del edificio (capa “Base”) más próximo al edificio tecnológico. Éste corresponde a las coordenadas (-37’7840, -0’0900,

7'2524) del punto base de la zona exterior, con un ángulo de rotación de 225° con respecto al resto de estructuras.

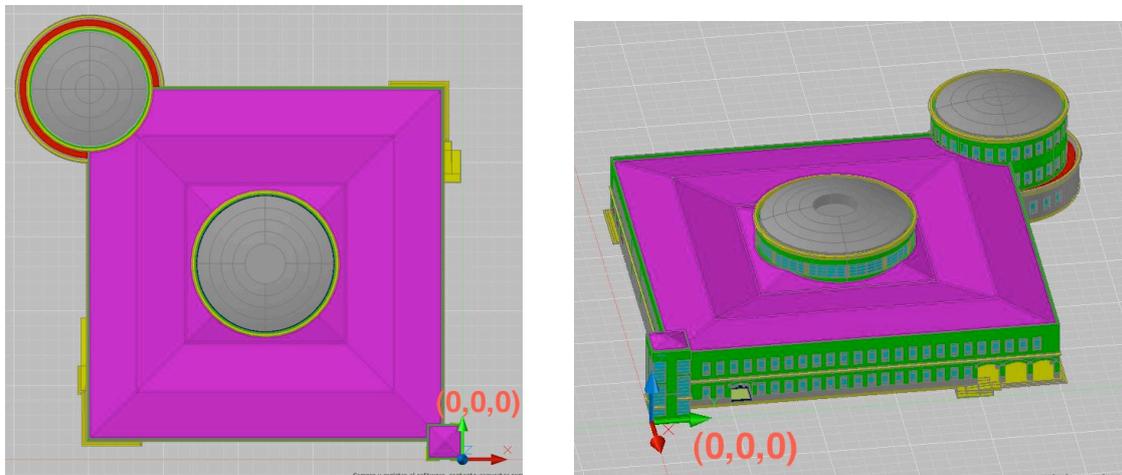


Figura 2.3-04: Punto de referencia en Escuela

El paso siguiente será montar cada edificio y dividirlo en archivos de forma que no sean excesivos sin superar un tamaño que permita su manejo en una sesión normal de AutoCAD.

Como último paso se guardarán todos los archivos en formato .dwg versión 2007 dejando activas todas las capas utilizables y desactivando las capas 0, defpoints y las auxiliares creadas a posteriori.

En el transcurso de este trabajo inicial se han observado multitud de inexactitudes debidas a la modificación de las estructuras reales con el paso del tiempo (cambios en tabiquería) y a cambios en el mobiliario. También se han descubierto muchos dibujos incompletos que en su día no eran necesarios para la realización del proyecto para el que fueron diseñados. Se ha procurado completar estos objetos modificando los originales y en muchos casos remodelando desde cero el objeto afectado.

En las **figuras 2.3-05** y **2.3-06** se observa un mismo archivo antes y después del trabajo. Aunque a simple vista no se notan muchas diferencias, internamente el archivo está libre de elementos auxiliares, capas, bloques y otros objetos que ralentizaban el procesador y podían crear incompatibilidades.

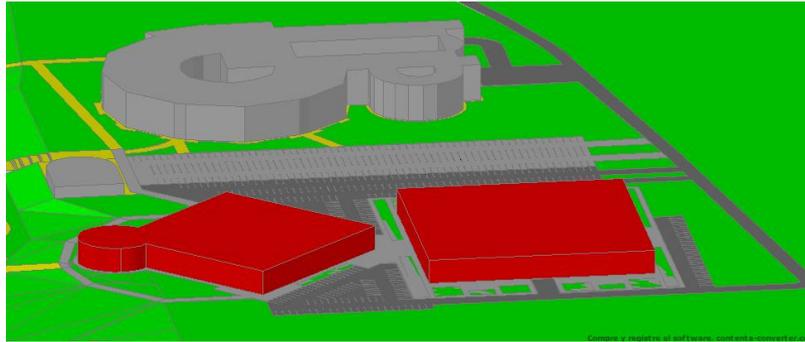


Figura 2.3-05: Archivo original, creado con mallas y sin limpiar

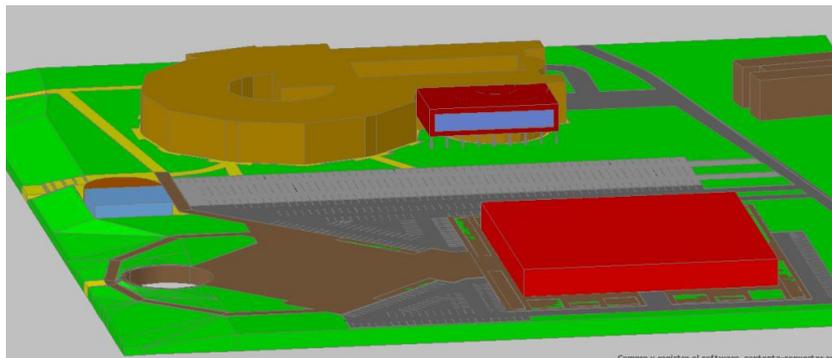


Figura 2.3-06: Archivo final actualizado, limpio de elementos no utilizados y formado por solidos3d

2.3.2 Modelado de nuevas zonas

Una vez revisados y actualizados todos los archivos se observa que faltan por modelar dos zonas comunes muy importantes en la Escuela: El salón de actos y la Biblioteca. A continuación se describe el proceso de modelado de dichas zonas.

2.3.2.1 Modelado del Salón de Actos

El salón de actos está situado en la parte cilíndrica del edificio en la planta baja con un desnivel profundizándose por debajo de la cota 0 para poder colocar las butacas escalonadas y facilitar tanto la visión de la pantalla como la acústica del recinto. Para su modelado se parte de los muros ya construidos en otros archivos y de los planos facilitados por la universidad de León. Estos planos resultan ser inconcluyentes ya que por un lado solo existen vistas en planta y por otro lado las medidas planificadas difieren bastante de lo construido realmente. Por ello se procede a una toma de medidas in-situ.

Para esta toma de medidas se ha usado un metro convencional para medidas pequeñas y un medidor láser proporcionado por Fernando J. Fraile, tutor del proyecto, para longitudes

grandes y sitios inaccesibles. El medidor usado es de marca *LEICA* modelo *DISTO D3*. Éste es un instrumento sobradamente capacitado para la toma de medidas necesarias para este trabajo.

La toma de medidas se realiza a partir de zonas ya dibujadas como el muro y las puertas, partiendo de la cota del suelo que es la misma que en el resto de la planta baja del edificio. Desde aquí calculamos la inclinación de la rampa central y de los escalones donde se sitúan las butacas con el propio medidor laser, que incluye la función de medición de desniveles y midiendo alturas al techo y profundidad para mayor exactitud. Todas las medidas tomadas se repiten varias veces y se usa la media de las mediciones. Se presta especial atención a detalles como las distintas cotas del techo, la diferenciación entre los dos lados de la tarima, el izquierdo con escalones y el derecho con rampas, las cotas de los huecos de los radiadores y las ventanas. También se toman medidas, esta vez minimizando más aun los posibles errores, de todo el mobiliario y accesorios importantes característicos del Salón.

Obtenidas ya las medidas necesarias se procede al modelado con *AutoCAD*. Para dicho modelado se mantienen los criterios adoptados en el resto de archivos. Partiendo del muro ya hecho y del suelo de la planta baja se modela primero la base del salón, esta es especialmente complicada debido a la conjunción de planos horizontales, inclinados con diversas pendientes y a la pared cilíndrica. Para resolver este modelo se comienza dibujando los escalones y la tarima por medio de extrusiones.

Una vez hecha la base se realizan varias operaciones booleanas de unión, intersección y sobre todo de diferencia con objetos auxiliares en forma de tubo cortados por planos con la inclinación del suelo del Salón. Como punto final se realizan los escalones y las pequeñas rampas de acceso a la tarima también mediante complicadas operaciones booleanas con sólidos auxiliares.

El proceso se puede ver en la **Figura 2.3-07 y 2.3-08**:

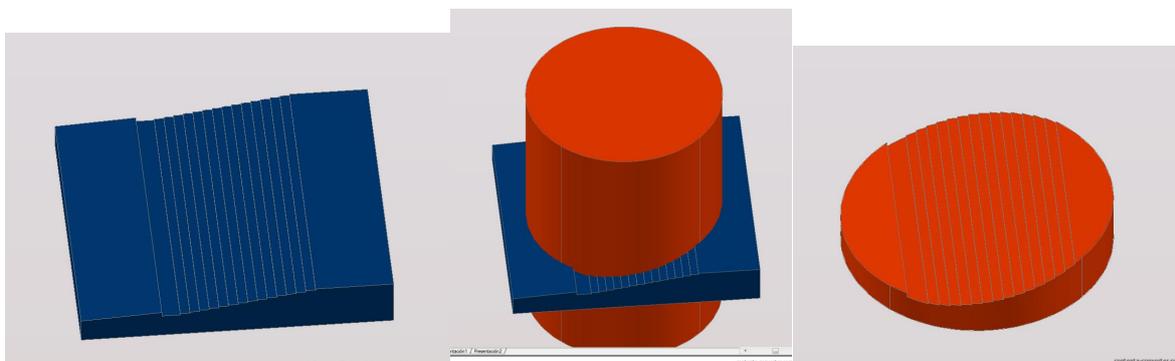


Figura 2.3-07: Modelado de suelo base de Salón de actos

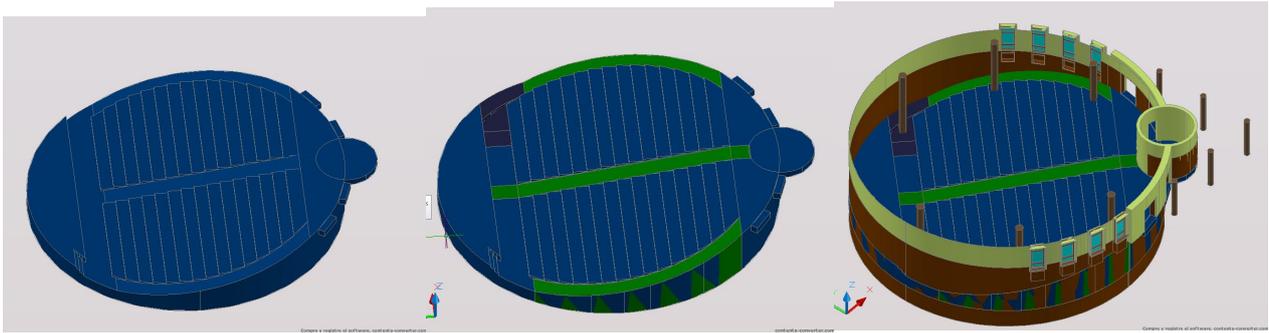


Figura 2.3-08: Modelado de estructura de Salón de Actos

Una vez colocada la base el resto es relativamente sencillo, se dibujan las diferentes alturas del techo, los huecos para los radiadores y ventanas, el recubrimiento de madera y el recinto para el proyector.

El techo se dibujó partiendo de la altura de las columnas modelando por orden la zona más alta, el doble techo de los laterales y el recubrimiento de las vigas como se muestra en la **Figura 2.3-09**.

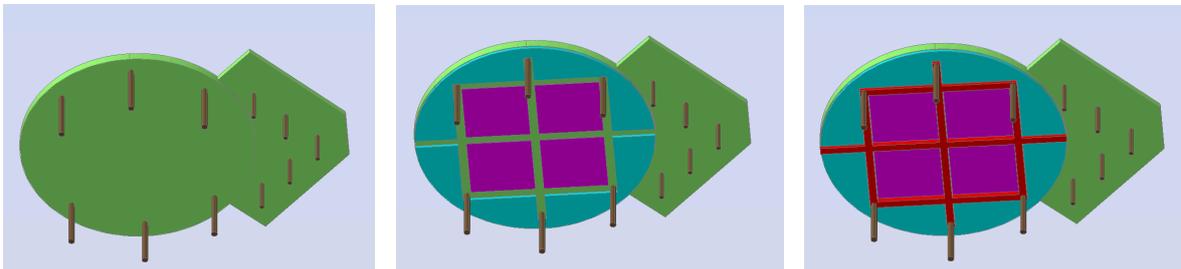


Figura 2.3-09: Modelado de techo de Salón de Actos

El siguiente paso consistió en el modelado del hall del Salón de Actos: Se aprovechó el suelo y las divisiones interiores del modelado de la escuela y se les añadieron los tabiques circulares que dividen hall y salón con sus huecos correspondientes para puertas y ventanas. Los tabiques se hicieron mediante extrusión de regiones planas y los huecos mediante diferencias con prismas. En este archivo también se añadieron las ventanas y puertas.

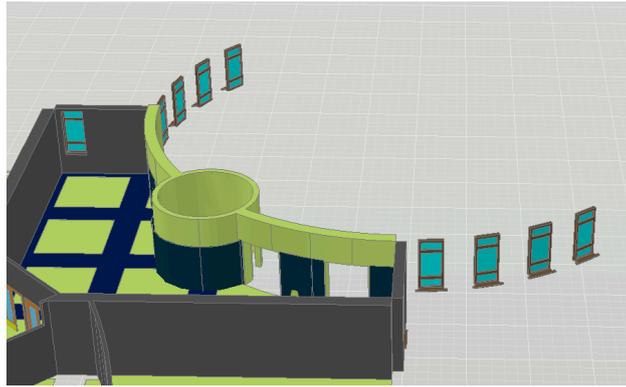


Figura 2.3-10: Hall de Salón de Actos

A continuación añadimos el mobiliario y los accesorios característicos del Salón de actos: pantallas de ordenador, micrófonos, Reflectantes laterales, pantalla, altavoces laterales y empotrados en el techo, alumbrado y adornos metálicos en el suelo y la tarima... Todos estos objetos se realizan primero en archivos independientes y luego se van situando en el salón.

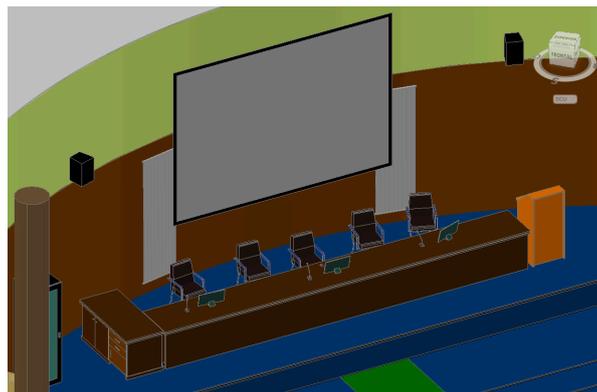


Figura 2.3-11: Detalle de mobiliario, pantalla y accesorios frontales

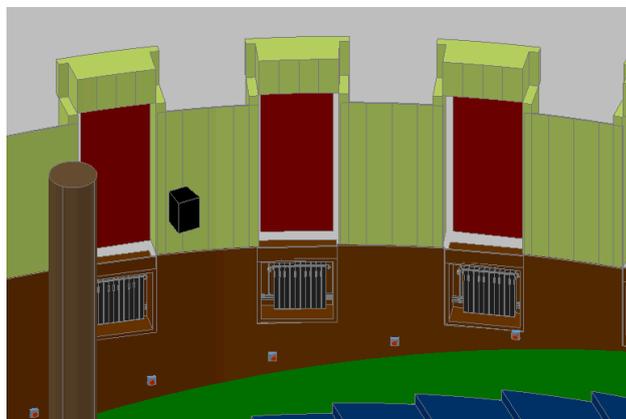


Figura 2.3-12: Detalle de radiadores, altavoces, reflectantes y cortinas

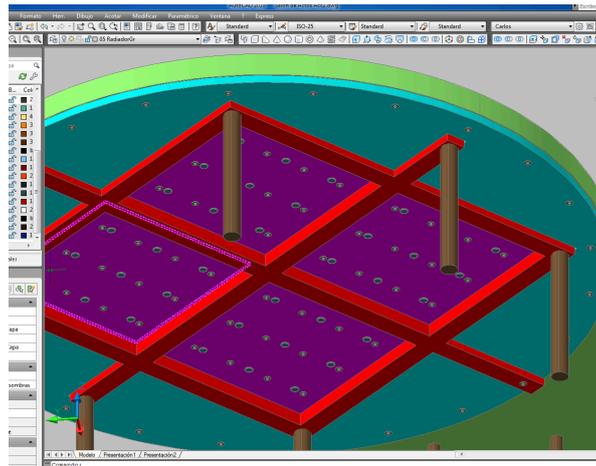


Figura 2.3-13: Detalle de iluminación y altavoces en el techo

Como último paso añadimos las butacas. Se realizó una de ellas a partir de prismas realizados por extrusiones, se recortaron los laterales para crear una leve inclinación y por fin se suavizaron las aristas con la orden empalme.

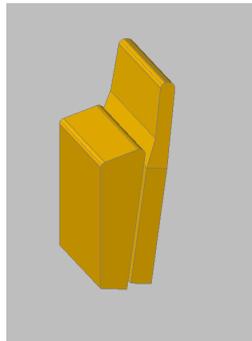


Figura 2.3-14: Modelado de tapicería de butaca

A esto se le añadieron las patas y el respaldo de plástico.



Figura 2.3-15: Modelado de butaca

Esta butaca se repitió las veces que hizo falta según la distribución real del salón de actos y se remató el modelado con los adornos laterales de tela de los inicios de cada fila.

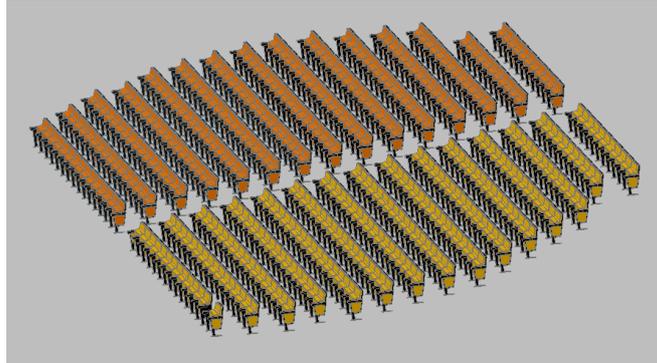


Figura 2.3-16: Modelado de butacas

Para cumplir con el criterio adoptado inicialmente se divide el archivo resultante en tres archivos principales: Estructura, Accesorios y Butacas. De esta forma ningún archivo tiene más de 15 MB de tamaño y el trabajo en *AutoCAD* es más fluido.

2.3.2.2 Modelado de Biblioteca

La biblioteca se sitúa justo encima del salón de actos, dentro del recinto cilíndrico de la estructura, con acceso desde la planta primera de la Escuela. Está constituida por dos plantas. El acceso al segundo nivel se hace a través de una escalera que parte de la primera planta de la biblioteca rodeando la columna central.

Para el modelado de la biblioteca los pasos son parecidos. En primer lugar se toman medidas que complementen los planos. En este caso los planos son de poca ayuda ya que la escalera no se corresponde en absoluto con la diseñada originalmente y las columnas tampoco mantienen las dimensiones de estos planos. Las medidas se van tomando a partir de la escalera central hacia el exterior. Para situar las columnas se toman varias medidas con respecto a sitios conocidos y como en el caso anterior todas las medidas son resultado de la media de varias mediciones. Una vez medida la estructura se mide individualmente el mobiliario y se sitúan en un croquis las mesas grandes y pequeñas, las estanterías y el resto de armarios.

Para el modelado en *AutoCAD* se comienza tomando como base el muro y el suelo de la primera planta del edificio ya dibujado. A partir de aquí se parte del centro del cilindro para modelar la columna central y la escalera. Esta se realiza a base de extrusiones en distintos

planos de referencia, es fundamental en este trabajo utilizar siempre el sistema de coordenadas (SPC) más apropiado para cada tramo de escalones.

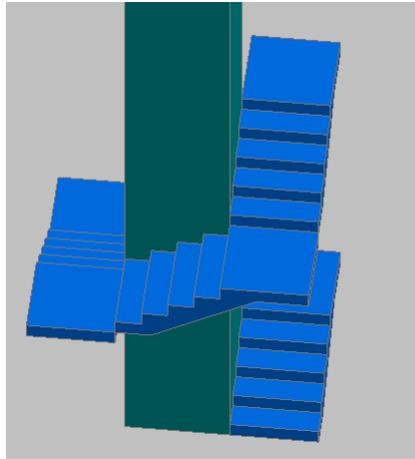


Figura 2.3-17: Modelado de escalones

Se usa la herramientas de chaflán para recortar el semimuro donde va apoyada la barandilla y la herramienta de barrido para realizar la barandilla de forma que las intersecciones de las barras queden lo más realistas posibles. La forma de la base de la escalera se consigue a base de operaciones de intersección y diferencia entre diversos prismas y cuñas.

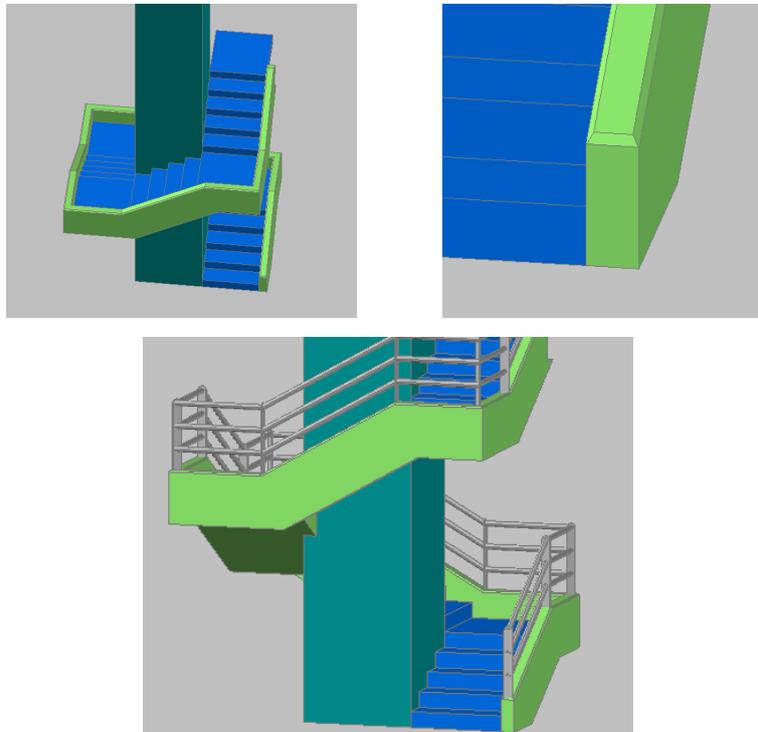


Figura 2.3-18: Detalles de escalera y barandilla colocada

Otra zona complicada de modelar fue la división entre las dos plantas, Ésta se consiguió mediante la diferencia de un sólido de revolución semejando el inverso de la cornisa y el techo propiamente dicho. A esto no hubo más que unirle el empalme con la escalera y recortar el lateral cilíndrico con un tubo de las mismas medidas que el muro exterior.

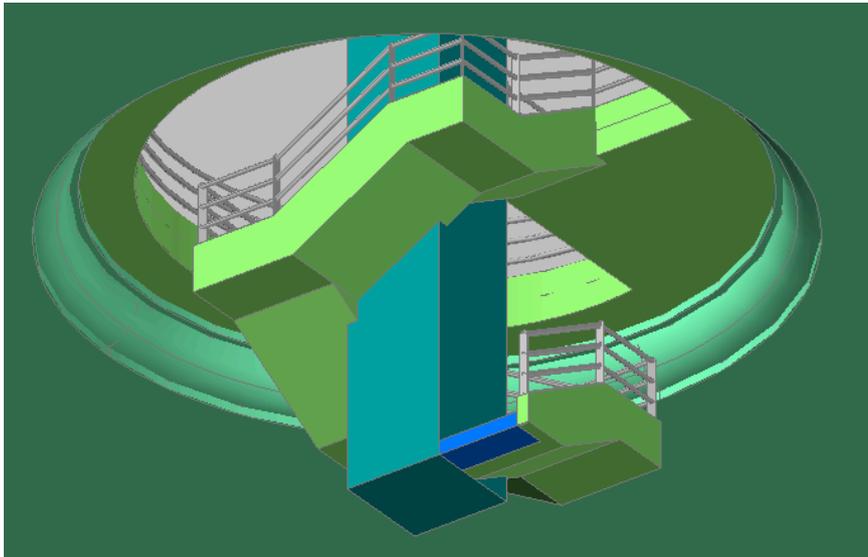


Figura 2.3-19: Modelado de Techo P1

La colocación de las columnas se realizó pensando más en las medidas in-situ que en el plano original.

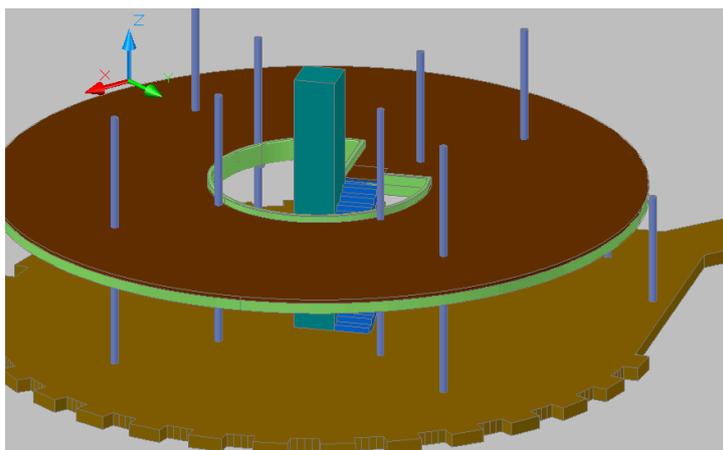


Figura 2.3-20: Colocación de columnas

El recubrimiento de madera y los huecos de radiadores y ventanas se realizaron de forma similar al Salón de Actos. Para estas construcciones sirvió de mucha ayuda la herramienta de matriz polar de *AutoCAD*.

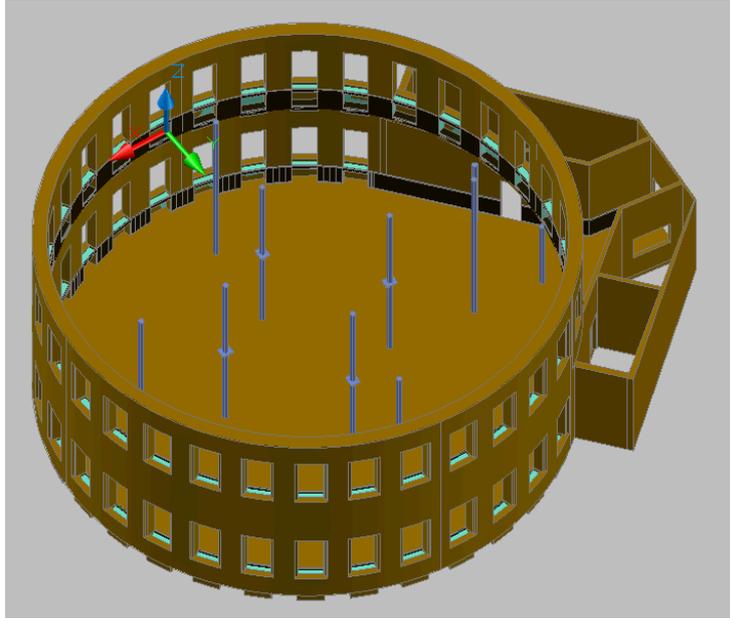


Figura 2.3-21: Construcción de rec de madera y huecos.

Finalmente a la estructura se le añadieron fragmentos de suelo y techo, ventanas y puertas quedando este resultado:

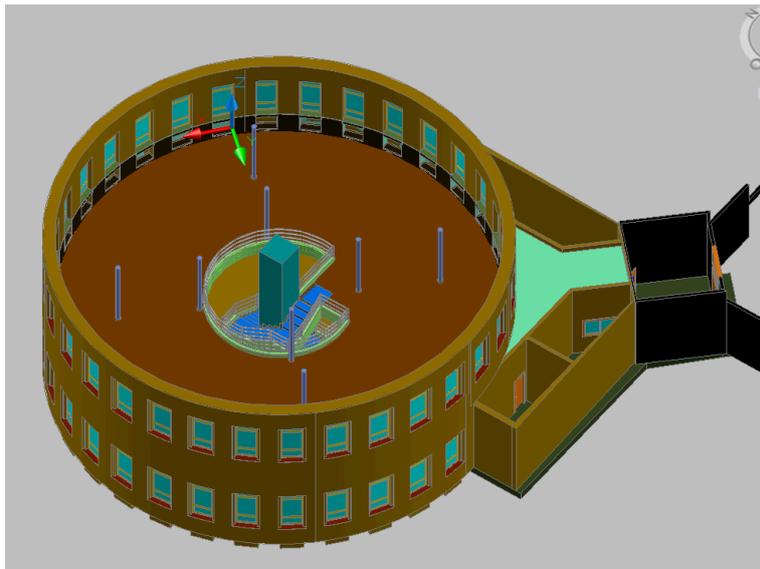


Figura 2.3-22: Estructura de Biblioteca

Para incluir los accesorios se usaron dos formas de trabajo:

- Para objetos únicos o repetidos brevemente como la mesa de la entrada, el clasificador o el detector de la entrada se siguió el mismo proceso que en el Salón de Actos, dibujando los objetos en archivos individuales y colocándolos luego en su posición en el archivo principal. En este archivo se incluyeron fluorescentes y focos en los techos siguiendo la distribución real de la biblioteca.

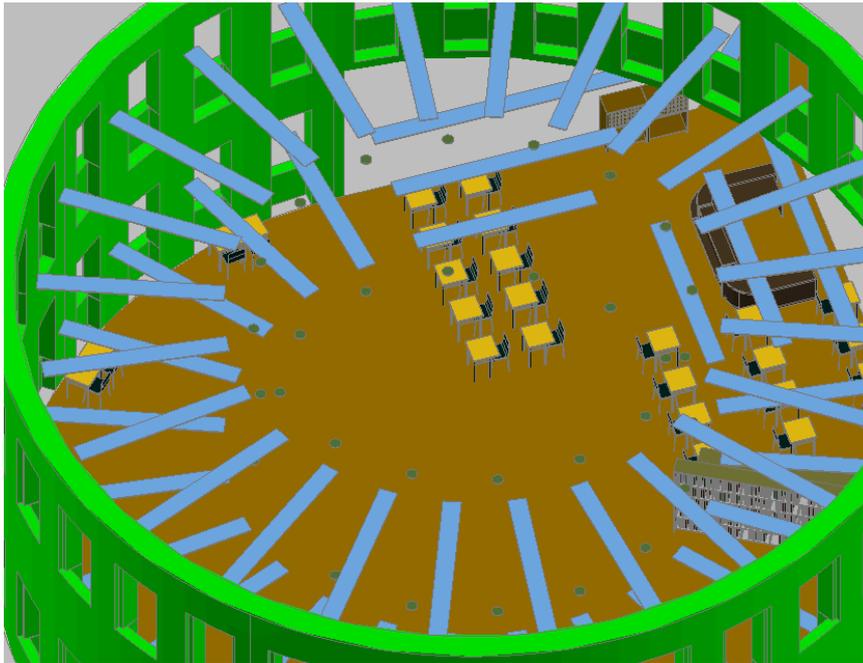


Figura 2.3-23: Añadido de accesorios.

- Para objetos muy repetidos se optó por el ahorro de recursos. Teniendo en cuenta que en Unity3d y en general en cualquier plataforma de mundos virtuales ocupa mucho menos un objeto repitiéndolo en el mismo programa que la importación de múltiples objetos iguales en un archivo. Se modeló solamente una pieza de cada objeto repetido colocándola en su sitio para mantener las referencias y se guardaron cada archivo o grupo de archivos individualmente. Podemos ver algún ejemplo de estos objetos:

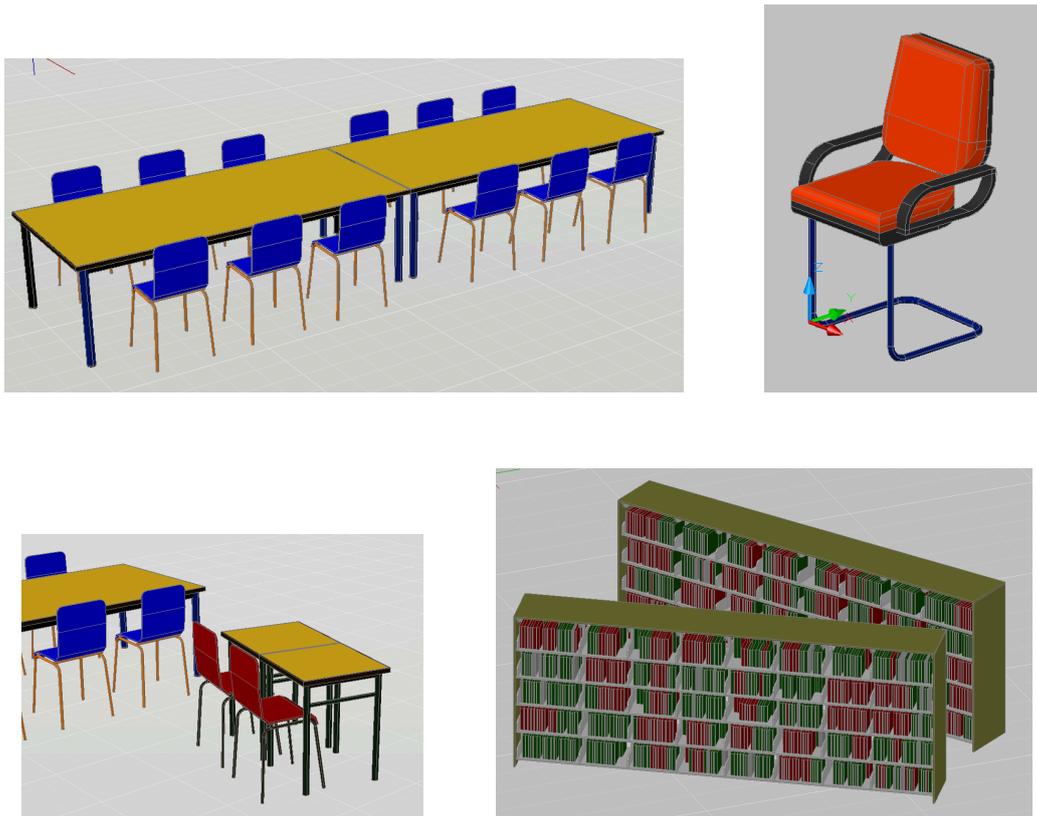


Figura 2.3-24: Mesas, sillas y estanterías de biblioteca.

El archivo general resultante se dividió en dos archivos por razones de tamaño, por un lado la estructura y por otro los accesorios no repetidos.

2.4 Trabajo con 3ds Max

El programa *Unity3d* es capaz de reconocer múltiples formatos de objetos 3d, pero *AutoCAD* no es capaz de exportar en ninguno de estos formatos; para llevar nuestros diseños al programa hemos de hacer un cambio de formatos mediante *3ds Max*.

3ds Max de *Autodesk* es uno de los programas más usados para modelado, diseño y animación en 3d. Este programa nos permite importar archivos .dwg desde *AutoCAD* y exportarlos a un formato reconocible por *Unity3d*. Después de informarnos en foros especializados elegimos el formato .fbx que es un formato estándar que reconocen múltiples programas relacionados con el diseño 3d y además nos permite exportar de una vez escenas con múltiples capas incluyendo texturas y materiales.

En esta conversión se tuvo en cuenta especialmente el que los objetos mantuvieran todas sus propiedades originales. Al importar de *AutoCAD* a *3ds Max* tuvimos ciertos problemas cuando

los objetos tenían formas cilíndricas o esféricas. La importación nos los convierte en mallas triangulares que dependiendo del nº de vértices que elijamos serán apreciables o no en el nuevo formato. Un cilindro por ejemplo se nos convertiría en un prisma con base un polígono con cierto número de lados: si elegimos un nº de lados alto (64 o más) el prisma se verá como un cilindro pero tendremos muchos más vértices, lo que conlleva más gasto de rendimiento a la hora de procesarlos y renderizar y archivos mucho más pesados. Eligiendo una baja resolución, polígonos de pocos lados nos disminuyen mucho el tamaño de los archivos y su coste de rendimiento pero en objetos grandes como las paredes interiores de la escuela, o incluso en columnas vemos una clara falta de realidad.

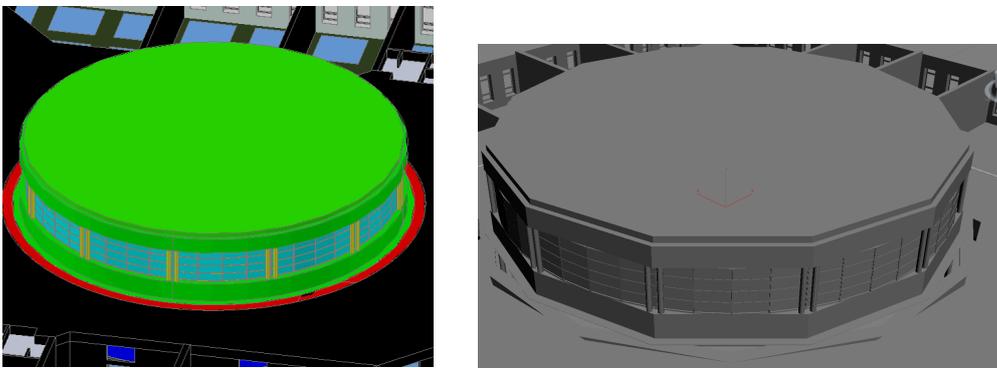


Figura 2.4-01: Comparación entre original de AutoCAD y importación en 3ds Max a baja resolución

La decisión que se toma para este problema dependerá de los archivos a tratar: en objetos con superficies curvas en los que se prevea una sensación de irrealidad se optará por disminuir el margen lo que conlleva al aumento de tamaño. Para objetos pequeños como pomos de puertas o sillas con partes curvas no se tendrá tanto en cuenta estos detalles ya que a la hora de incluir texturas las diferencias son inapreciables.

Una mención especial para esta disyuntiva es el caso de las columnas, que si bien son elementos característicos de los edificios y han de ser modelados lo más fielmente posible, el disminuir los márgenes en la importación nos creaba archivos exageradamente grandes. El programa Unity3d permite realizar formas básicas de modelado y una de ellas es el cilindro. Se decidió omitir las columnas de las importaciones y usar cilindros de Unity3d, de esta forma el rendimiento es máximo por la forma de tratar a los objetos 3d de Unity3d que se explicará en siguientes capítulos.

Se ha de comentar también que en los modelos más recientemente creados por otros alumnos como son el Edificio Tecnológico y El Aula 204 este problema ya había sido solventado convirtiendo en *AutoCAD* los cilindros en prismas de 64 lados.

En esta importación se tiene también en cuenta las capas que nos interesan y las que eran auxiliares para el modelado en *AutoCAD*, solo vamos a importar las que nos interesan. Recordemos que se renombraron las capas auxiliares con un 00 delante para facilitar este paso.

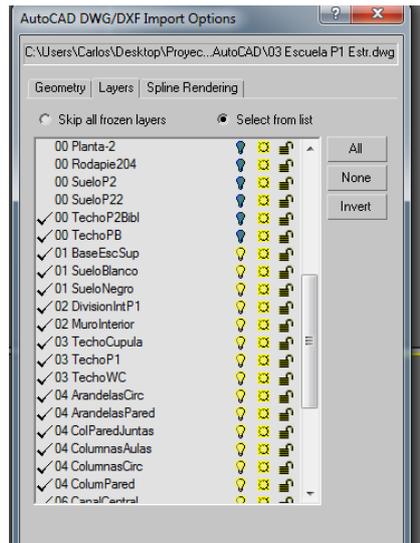


Figura 2.4-02: Selección de capas

Para exportar desde *3ds Max* a formato *.dxf* solo hemos de tener en cuenta el que las unidades se mantengan, en nuestro caso hemos trabajado siempre en metros y lo seguiremos haciendo.

Hemos comentado que el formato *.fbx* nos da la posibilidad de llevar consigo las texturas y materiales de cada objeto y se nos planteó la posibilidad de aplicar estas texturas en *3ds Max*, ya que este es un programa con el que se puede llegar a un realismo difícilmente distinguible con escenas reales al usar tratamientos de texturas especializados. Después de varias pruebas nos dimos cuenta de que el programa *Unity3d* reconoce los objetos, las texturas básicas (difuse) pero no los tratamientos especiales realizados con ellas en *3ds Max* con lo que decidimos aplicar los materiales desde *Unity3d* que sin ser tan especializado en este tipo de modelados, también permite usar una serie de tratamientos de texturas o shaders con las que se llega a un realismo muy aceptable. Estos tratamientos los explicaremos más detenidamente en el siguiente capítulo.

2.5 Creación de texturas y materiales

2.5.1 Texturas y materiales

Las texturas y materiales que vamos a aplicar a cada objeto son la base para que la sensación de realidad que buscamos sea la óptima.

En primer lugar hemos de distinguir entre textura y material. La textura va a ser la imagen con la que vamos a cubrir a cada objeto, estas texturas pueden ser continuas, con un solo color, cíclicas, con un patrón que se repite horizontalmente, verticalmente o en ambas direcciones, o con colores y formas diversos.

Los materiales implican muchas más características. Un material además de contener una o varias texturas podemos configurarlo de forma que realice ciertas funciones ópticas o físicas para acercarse más a la realidad, en el renderizado en el caso de las cualidades ópticas y en las animaciones con las opciones físicas. En nuestro caso salvo en contadas excepciones vamos a crear objetos estáticos y vamos a fijarnos en las posibilidades de renderizado que nos da Unity.

En Unity tenemos múltiples opciones de tratamiento de texturas, que el programa llama sombreados o shaders. Explico brevemente las más utilizadas en este trabajo:

- ***Difuse shader*** – Sería la textura sin más modificación, solo lo usaremos en casos de objetos sin relevancia ya que aunque es la que más ahorra en espacio y rendimiento, es también la menos real.

- ***Bumped shader*** – Mediante esta herramienta obtenemos una sensación de profundidad en imágenes que representen algún tipo de relieve. Esto lo conseguimos a partir de la textura normal de la imagen original, esto es una imagen que representa las zonas más y menos alejadas de la textura en forma de imagen monocroma más o menos intensa en esas zonas. Se ha usado esta opción para materiales con relieves del tipo ladrillos, alfombras, asfaltos o telas.

- ***Specular shader*** – El tratamiento specular nos da una sensación de brillos o reflejos más o menos intensos según la iluminación aplicada. Este tipo de sombreado se ha usado en metales y plásticos y en muchas ocasiones combinado con el bumped para maderas barnizadas con relieves, cubiertas y metales rugosos.

- ***Transparent*** – Como su nombre indica se usa para objetos transparentes u opacos, usa texturas alfa en escala de grises de forma que cuanto más cercano al negro sea la textura, más transparente se verá. Este shader se usó para cristales de ventanas más y menos opacos.

- **Transparent Cutout** – Dentro de los sombreados transparentes el cutout lo que hace es volver totalmente transparente una parte de la textura y totalmente opaco el resto. Es muy útil a la hora de hacer elementos con mallas o rejillas. En nuestro caso se usó para los bancos metálicos y las rejillas de los fluorescentes en algunas aulas. También es el usado en las texturas de las hojas en el editor de arboles del que hablaré más adelante.

- **Reflective Shader** – Simula superficies de tipo cromado. No es literalmente un reflejo tipo espejo, el efecto lo logra con cubemaps que son texturas adaptadas para este tipo de shader. Se ha usado en metales especialmente pulidos.

- **Self-illuminated shader** – Simula desprender luz del interior del objeto. Se ha usado en bombillas, fluorescentes y reflectantes.

Hay muchos más tipos de shaders más especializados pero en nuestro caso dada la magnitud del proyecto nos vamos a basar en estos ya que de otra forma el proceso se alargaría exageradamente y el número de texturas diversas a utilizar sería exageradamente abultado.

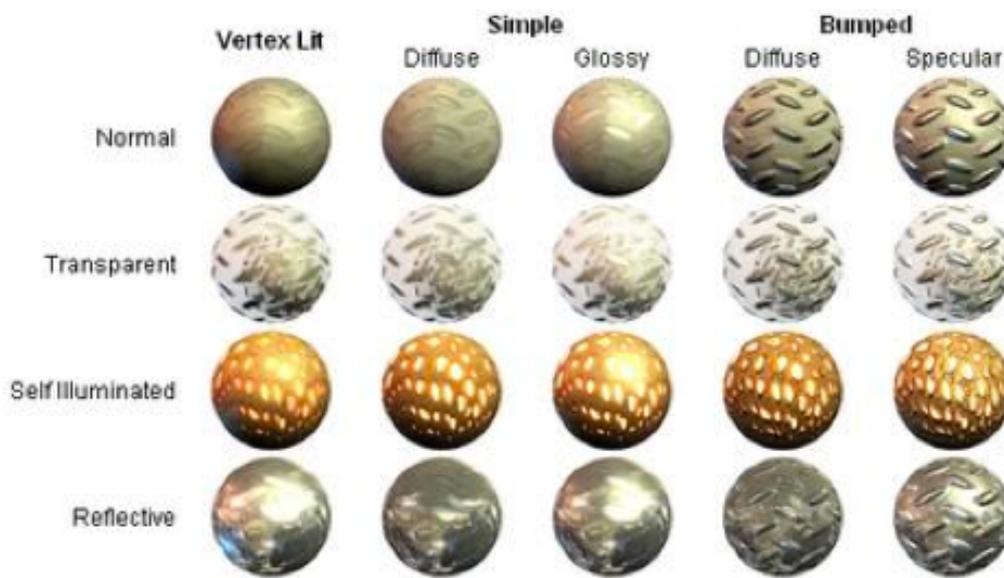


Figura 2.5-01: Tipos de shaders en Unity

En *Unity3d* vamos a aplicar un material a cada capa importada de *AutoCAD*. Para que la sensación de realismo sea la adecuada vamos a partir siempre de fotografías realizadas a los objetos reales. A cada fotografía se le va a dar un tratamiento distinto dependiendo del tipo de material que se va a generar a partir de ella.

2.5.2 Creación de texturas

Vamos a describir el proceso de creación de las texturas a partir de las cuales generaremos los distintos materiales. Como he dicho anteriormente, hay fundamentalmente tres tipos de texturas: las podríamos denominar monocromas, cíclicas y abstractas:

- Texturas monocromas o continuas:

Son materiales con un solo color base como metales, plásticos y pinturas lisas. Vamos a incluir aquí también asfaltos, hormigones en masa y pinturas con gotelé, ya que el tratamiento es muy parecido.

Unity nos da la opción de usar un color base para la construcción de cada material, y para elegir este color además de las típicas ruedas de colores, tenemos una herramienta de captura de un color en la pantalla, la pipeta tan usada en los programas de edición fotográfica.



Figura 2.5-02: Herramienta de selección de color

Con el fin de ahorrar espacio en el proyecto final, para muchos materiales monocromos hemos usado esta herramienta en las fotografías realizadas a los materiales reales y a partir de aquí se le ha dado el tratamiento adecuado a cada material.

- Texturas repetitivas o cíclicas:

Estos son materiales que siguen un mismo patrón. En la mayoría de los casos este patrón se repite en sentido vertical y horizontal. Lo que vamos a hacer es tomar una porción representativa de la fotografía del objeto y mediante un programa apropiado realizar la textura de forma que en los límites de esta porción de fotografía la textura siga de forma continua.

Para este proceso nos hemos ayudado de un programa de creación de texturas llamado **PixPlant** en su versión 2.0.50. Este programa se comercializa tanto como plug-in para photoshop como en su versión standalone. Es una herramienta muy sencilla de usar y rotalmente indicada para nuestros propósitos.

Trabajo con **PixPlant**

1) En primer lugar PixPlant nos da una herramienta de enderezado de la imagen o straighten seed. Una vez importada la fotografía al programa, creamos un rectángulo sobre la zona que nos interesa. Este rectángulo tiene los vértices móviles, de forma que con el ratón podemos acercar cada vértice a las supuestas esquinas del patrón que queremos repetir. Una vez hecho esto el programa nos convierte en paralelas las caras opuestas del paralelogramo creado sobre la imagen original. En la figura 2.5-03 observamos el resultado obtenido al enderezar la fotografía de una textura de ladrillos.

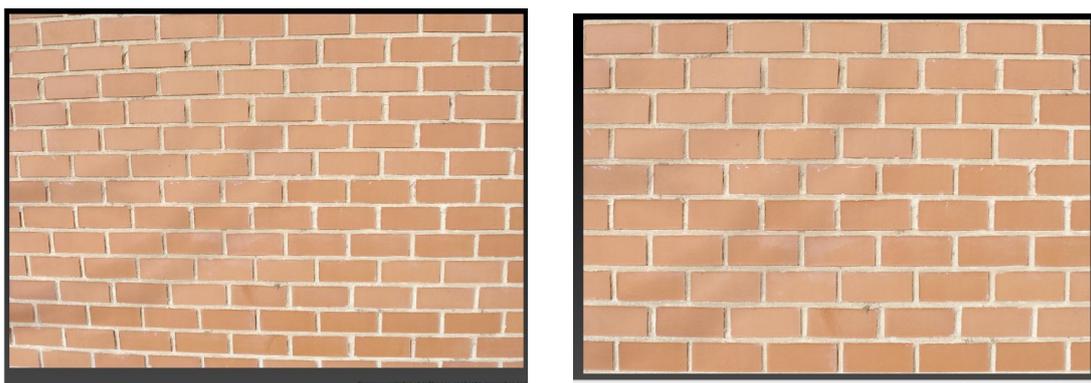


Figura 2.5-03: Textura antes y despues del enderezado

2) Dentro de cada fotografía siempre se ha procurado coger varias veces el patrón simple, para una textura de ladrillos vamos a coger al menos 3 o 4 filas y columnas de ladrillos, así al no repetir siempre el mismo ladrillo el realismo es mayor. La siguiente herramienta es la creación del patrón a repetir, en caso de que éste exista. Mediante esta herramienta también por medio del ratón y de forma sencilla creamos una malla que delimite los patrones simples que se van a repetir en la textura.

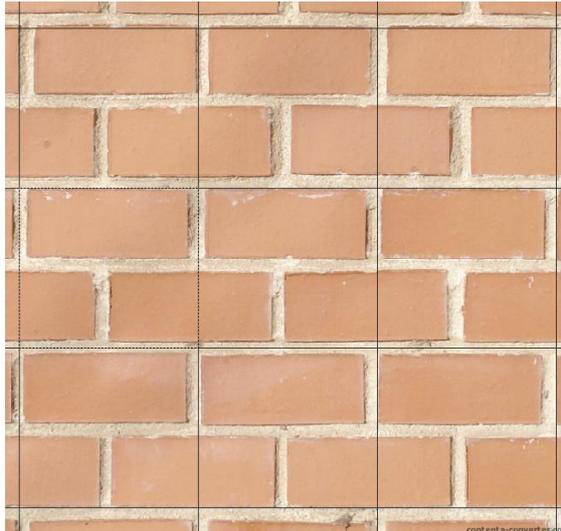


Figura 2.5-04: Patrón de textura de ladrillos

3) Una vez hecho esto, simplemente con la opción de generar o *generate* el programa nos da la textura buscada. *PixPlant* repite de forma aleatoria el patrón que hemos definido anteriormente y suaviza los límites verticales y/o horizontales de forma que cree la continuidad buscada para nuestra textura. Vemos la textura obtenida en la **Figura 2.5-05:**

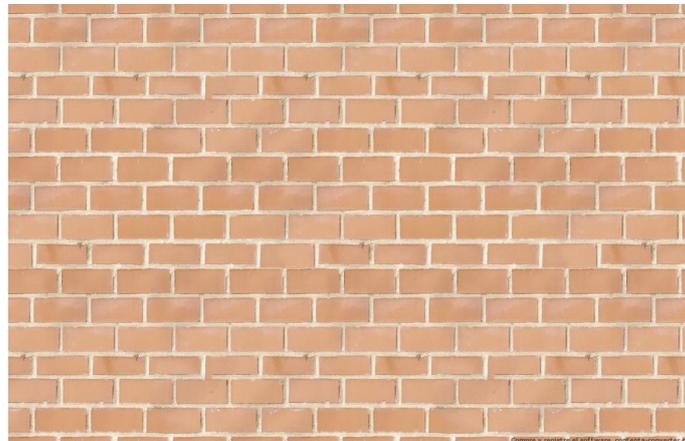


Figura 2.5-05: Textura creada con *PixPlant*

4) *PixPlant* además de crearnos la textura bidimensional tiene la opción de crear las texturas specular, diffuse y normal que buscamos para el tratamiento de materiales en cualquier programa de modelado 3d. En nuestro caso buscamos la textura normal para los sombreados bumped. En la pestaña “3d Material” tenemos múltiples herramientas para crear estas texturas. Podemos ajustar la intensidad del relieve, la proporción de detalles finos, medios o gruesos y la proporción de relieve entre estos detalles y la base real de la textura. En

la **Figura 2.5-06** comprobamos los distintos resultados regulando la proporción de detalles de distinta forma.

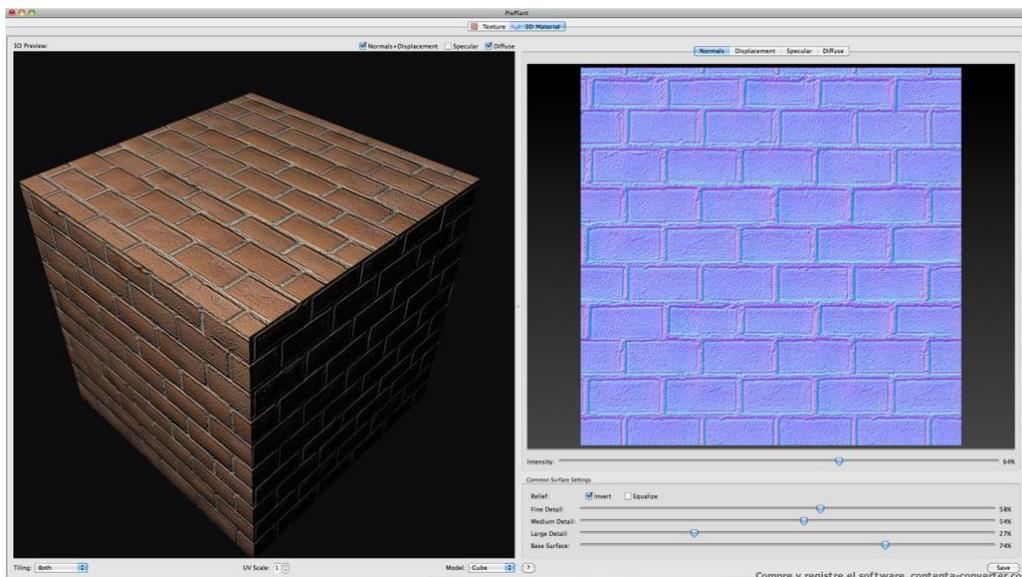
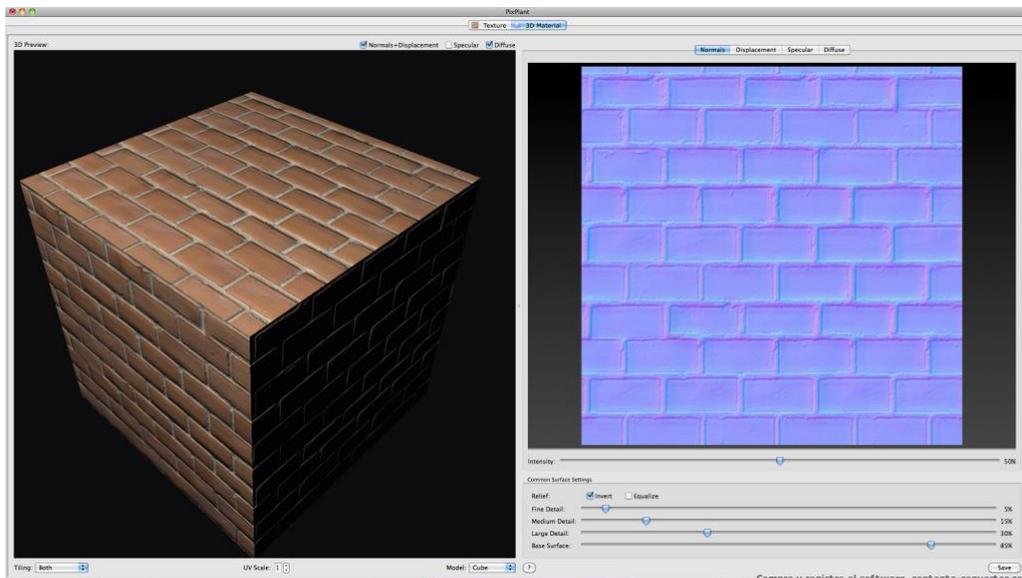


Figura 2.5-06: Ladrillo liso y poroso

5) Una vez creadas las texturas diffuse y normal que necesitamos para la construcción de materiales en *Unity*, *Pixplant* nos puede guardar el trabajo en diversos formatos estándar de imagen. Vamos a elegir el formato comprimido .jpeg ya que nos da la calidad suficiente para nuestro trabajo ocupando bastante menos tamaño que el resto de formatos.

Para texturas de elementos con relieve pero sin un patrón fijo como pinturas con gotelé, telas y asfaltos el proceso es el mismo sin marcar un patrón fijo. De aquí nos interesan fundamentalmente las texturas normales, ya que la base en muchos casos la vamos a obtener con un color base en *Unity*.

- **Objetos con colores abstractos o sin repetición**

Va a haber determinados objetos que o bien por su complejidad de modelado o por sus características no se pueden dividir en materiales. Estos serán los carteles de las paredes y los paneles informativos, las pantallas de publicidad, las revistas, las maquinas de alimentación, las mangueras del sistema contra incendios o las fachadas de los edificios colindantes.

Para representar estos elementos se ha decidido partir de la forma exterior básica y sobreponer la textura del elemento real en las superficies de estas formas.

Para esta labor la textura que necesitamos serán normalmente imágenes rectangulares que incluirán lo más detalladamente posible el elemento a imitar.

Para la obtención de estas texturas hemos usado un sencillo programa de tratamiento fotográfico llamado **DxO Optics**. Éste es un programa de retoque fotográfico y mejora de calidad de la imagen. Mediante varios pasos podemos recortar, enderezar y corregir pequeños errores en las imágenes buscadas. También tenemos la opción de actuar sobre balances de iluminación, color, geometría y otros detalles para mejorar la imagen. Vamos a explicar el proceso que se ha seguido mediante un ejemplo de una de las texturas:

- Importamos la imagen original
- Recortamos la parte que necesitamos
- Enderezamos y cuadramos la porción de imagen que nos interesa
- Retocamos si es necesario contraste, color, exposición etcétera
- Procesamos la fotografía convirtiéndola en el formato que más nos interese, en este caso .jpeg.

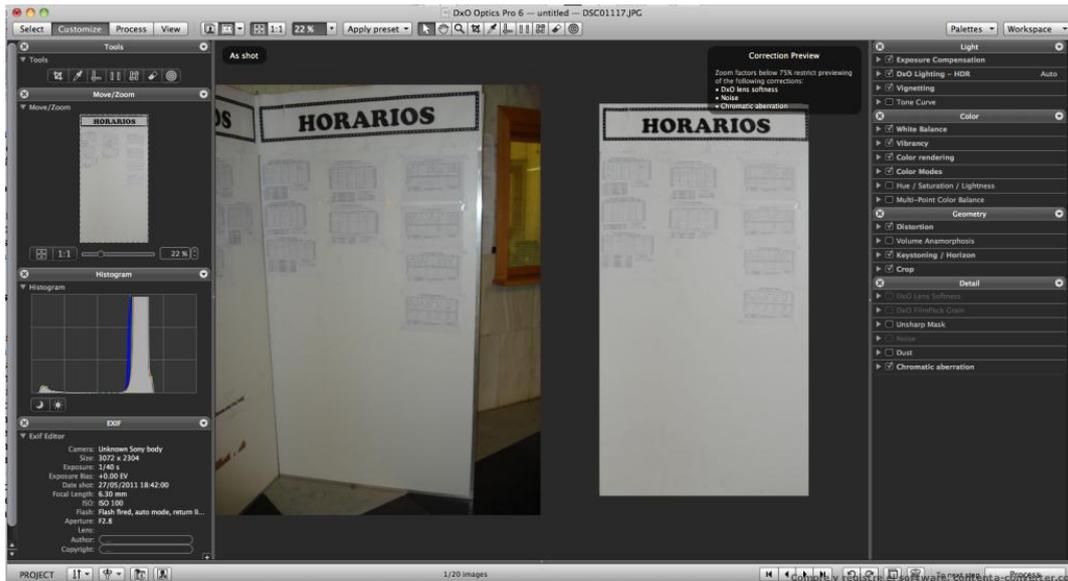


Figura 2.5-07: DxO Optics

Una vez que conseguimos la textura, la exportaremos a *Unity*. El proceso de creación de materiales en *Unity* lo explicaré más detalladamente en el siguiente capítulo.

2.6 Trabajo en *Unity3d*

2.6.1 Introducción a *Unity3d*

Como ya he explicado, *Unity3d* es un programa de creación de aplicaciones con gráficos y animaciones 3d. Además de formar un pequeño mundo virtual, *Unity3d* es capaz de además de formar un pequeño mundo virtual, crear animaciones y personajes con inteligencia artificial.

La forma de trabajar más usual de *Unity* sería partir de modelados con sus texturas propias y animaciones hechos con *3ds Max*, *Maya* o *Blender* y una vez importados crear las acciones y movimientos de la aplicación mediante pequeñas programaciones, denominadas scripts, asignadas a cada objeto.

En nuestro trabajo partimos de unos objetos realizados con *AutoCAD* que se han dividido en capas según el material con el que están hechos. Hemos preferido realizar el trabajo de texturizado en el mismo programa *Unity* para aprovechar al máximo la creación de materiales de este motor gráfico.

Una vez que tenemos los objetos con sus materiales se han añadido luces y pequeñas animaciones para dar más realismo al trabajo.

El último paso será englobarlo todo en una sola aplicación.

2.6.2- Interfaz gráfica del programa

La interfaz gráfica de usuario o GUI de Unity consta de 5 regiones principales de trabajo:

1.- **Scene View** o Vista de Escena – Es el área de construcción gráfica de Unity. Nos permite además de visualizar nuestro trabajo, interactuar sobre los objetos que hay en él con movimientos, giros, cambios de tamaño y dotación de propiedades a cada objeto.

2.- **Game View** o Vista de Juego – En esta vista tenemos una previsualización de nuestra aplicación.

3.- **Project View** o Vista de Proyecto – es donde vamos a tener un listado de todos los recursos o assets de los que disponemos para crear la aplicación. Cada asset se puede usar en distintas escenas.

4.- **Hierarchy View** o Vista de Jerarquía - listado de todos los elementos que se están utilizando en la escena que se está creando.

5.- **Inspector View** o Vista de Inspector – Muestra todas las propiedades del objeto seleccionado.

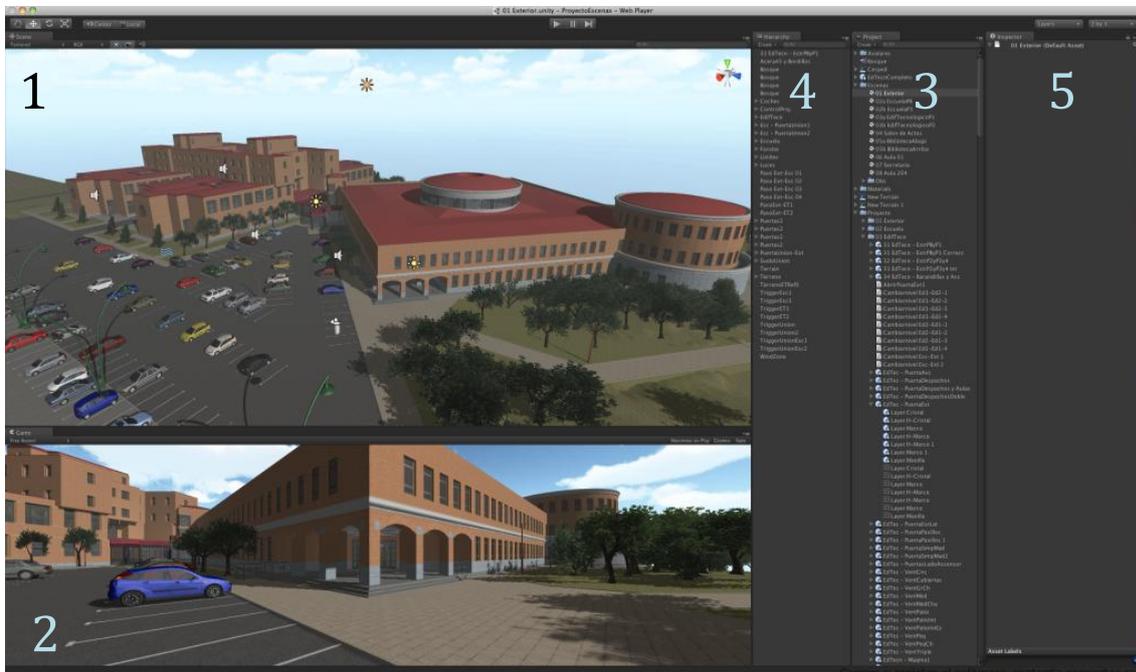


Figura 2.6-01: Interfaz gráfica de Unity3d

Estas cinco zonas importantes se pueden colocar dentro de la pantalla de la forma más cómoda para el usuario modificando la posición por medio de pestañas.

La forma de trabajo es muy sencilla: En primer lugar se importan los assets a utilizar. Hay que decir que Unity viene con una serie de assets predeterminados que en muchos casos nos han ahorrado bastante trabajo. Para importar los objetos 3d, texturas y scripts no hay más que arrastrarlos a la Vista de proyecto. Al importar los objetos, *Unity* hace una copia de éstos en la carpeta seleccionada como carpeta de proyecto y los deja preparados para su uso en cualquier escena del proyecto. Para usar estos objetos en la escena solo hay que arrastrar el objeto en cuestión desde la Vista de Proyecto a la Vista de Jerarquía o directamente a la Vista de Escena.

Además de importar objetos 3d externos, *Unity* también nos permite crear elementos básicos como cubos, cilindros y esferas que nos van a servir para hacer pequeñas correcciones sin tener que volver a pasar por el proceso de modelado en *AutoCAD* y cambio de formato en *3ds Max*.

Unity viene preparado para crear un proyecto en varias escenas y así ahorrar recursos. Para pasar de una escena a otra se hace mediante scripts básicos.

Para dar realismo a nuestra escena, *Unity* pone a nuestra disposición unas cuantas herramientas como editores de terreno, creadores de árboles, cámaras, luces, skybox, nieblas... que explicaremos según las vayamos utilizando a la hora de montar la escena.

2.6.3 Trabajo en una sola escena

Como he mencionado antes, *Unity* nos permite realizar nuestro mundo virtual en una sola escena o bien dividirlo en varias escenas para ahorrar recursos. Si dividimos el trabajo en varias escenas cada una de ellas se cargará y gastará recursos del ordenador solamente cuando esté activa. Por otro lado, ya que tenemos todos los objetos de los edificios referenciados a un sistema de coordenadas común, es mucho más sencillo montarlo todo en la misma escena.

Decidimos hacer una prueba con el terreno exterior y los edificios con sus respectivas aulas en una misma escena.

Para llevar a cabo este trabajo importamos todos los objetos al programa siguiendo unos pasos que serán comunes a todas las importaciones que se van a realizar en el futuro y que vamos a explicar con detalle:

- **Importación a la Vista de Proyecto** – Esto consiste simplemente en arrastrar los archivos en .fbx a la Vista de Proyecto. Dada la cantidad de objetos con los que vamos a trabajar es necesario un poco de orden para este proceso. *Unity* nos permite crear carpetas y subcarpetas para organizar nuestro trabajo de forma similar al de cualquier sistema operativo.

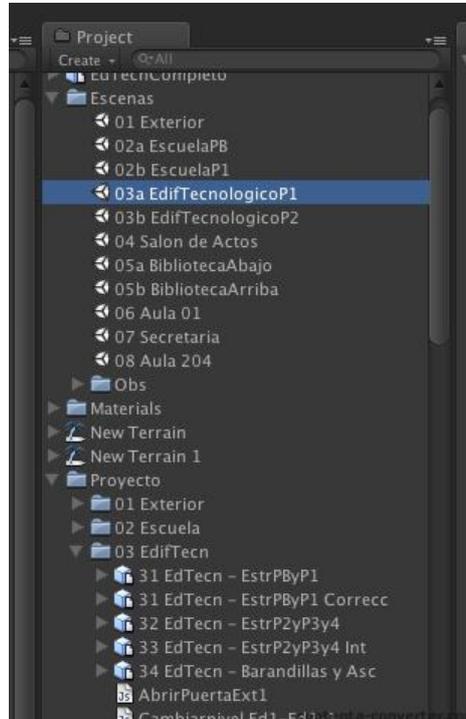


Figura 2.6-02: Detalle de Carpetas en la vista de Proyecto

- **Adaptación de escala** – En la Vista de inspector, al seleccionar cualquier objeto añadido, nos aparecen las propiedades de importación de fbx. Por defecto nos aparece como factor de escala 0.01. Como en los programas usados anteriormente hemos usado como unidades los metros, hemos de aplicar un factor de escala de 1.

- **Compresión de mallas** – Unity tiene una herramienta de compresión de mallas que nos va a venir bien para reducir tamaño de vértices en objetos no relevantes pero con un número exagerado de vértices. Recordemos que el gasto de espacio y recursos de este tipo de programas depende fundamentalmente del número de vértices de cada modelo. Unity no es capaz de trabajar con modelos de más de 65000 vértices. Este número puede parecer inalcanzable, pero en algún caso de objetos con formas curvas y muy repetidos nos ha traído problemas, como en las barandillas del Edificio Tecnológico o en pies de sillas de las aulas.

Como norma no hemos comprimido las mallas excepto en casos muy claros con compresiones suaves.

- **Creación de colisiones** – En nuestra escena va a haber objetos con los que el controlador principal tenga que chocar, o no atravesar. Esto aunque en la realidad sea una cosa lógica, para la creación de mundos virtuales es un tema a estudiar detenidamente. En un principio podemos suponer que todos los objetos son sólidos y no se pueden atravesar con lo que aplicaríamos colisiones o *colliders* a todas las partes. Esto implicaría un gasto de recursos enorme, cada objeto va a tener los vértices de su renderizado y los vértices de su colisión. Para ahorrar recursos vamos a generar colisiones solo en los sitios necesarios. En las propiedades de importación de fbx nos aparece una casilla con la orden *generate colliders* que nos genera colisiones en todas las partes de nuestro objeto 3d. Solo la vamos a activar en los objetos donde todos los distintos materiales o la mayoría de ellos necesiten colisión. En el resto de casos generaremos las colisiones a posteriori colocando el tipo de colisión que mejor se adapte a cada caso.

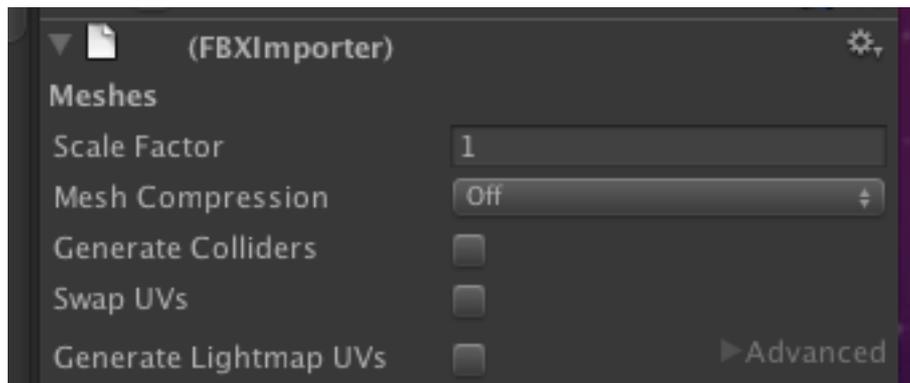


Figura 2.6-03: Configuración de importación de archivos .FBX

- **Importación a la escena** - Una vez realizadas estas correcciones llevamos nuestros objetos a la Vista de Jerarquía o a la Vista de Escena arrastrándolos con el ratón. Si arrastramos a la Vista de Jerarquía, el objeto se colocará con su punto de referencia o pivot en las coordenadas 0,0,0. Si arrastramos a la Vista de Escena el objeto se colocará en el punto donde lo soltemos en la escena. El pivot será el punto 0,0,0 que dejáramos en el archivo de AutoCAD o bien el centro del objeto para objetos creados directamente con Unity. En esta fase como en la de importación a la Vista de Proyecto hemos de prestar especial atención a la organización en carpetas y objetos múltiples ya que van a ser muchos los objetos a tratar. Como resultado

vamos a visualizar el objeto importado en la vista de Escena aunque con una única textura grisácea generada en la creación del archivo .fbx, ya que aun no hemos definido las texturas de cada objeto.

- **Asignación de materiales** – La forma más sencilla de crear materiales es arrastrar simplemente la textura base del material que queramos crear a algún objeto que vaya a llevar ese material. De esta forma el programa nos crea una carpeta en donde estuviera la textura llamada “materials” y dentro de ella tendremos un nuevo material con shader diffuse que podremos modificar según lo que busquemos. Al modificar el material se modificarán en escena todos los objetos a los que se les haya asignado.

- **Creación de luces** – Para visualizar mejor lo que vamos importando es bueno que tengamos alguna luz exterior, estas pueden ser puntuales, direccionales o focos. Explicaremos detalladamente el uso de la iluminación más adelante.

- **Creación de un controlador principal** – *Unity* por defecto en cada nueva escena nos crea una cámara principal, que es la que nos da la imagen que sale en la Vista de Juego. Para poder realizar el paseo virtual necesitamos un controlador que podamos manejar con algún aparato externo, bien sea con teclas del teclado, con el ratón o con cualquier interface externo. Estos controladores se pueden crear a base de aplicar scripts de movimiento a cualquier objeto. *Unity* por defecto trae Controladores prefabricados tanto en primera persona como en tercera persona utilizando avatares animados. Para no malgastar tiempo y recursos hemos optado por usar un controlador en primera persona prefabricado por *Unity* realizando en él unas modificaciones sencillas que detallaremos más adelante. Con este controlador nos podemos mover por nuestro mundo usando las teclas de cursor del teclado o bien las teclas *AWS* y *D*, y orientando nuestra visión con el ratón.

Enseguida nos dimos cuenta de que incluso sin llegar a un tratamiento especializado de texturas, y con solo un par de luces generales, el renderizado del ordenador iba muy lento, descartamos la opción de incluir todo el proyecto en una sola escena ya que el objetivo del trabajo es que la visita sea fluida para cualquier tipo de ordenador con unas prestaciones mínimas.



Figura 2.6-04: Los dos edificios con todas las escenas incluidas

2.6.4- División del proyecto en escenas

Una vez decidido crear el proyecto en escenas se ve bastante lógico hacer al menos 3 escenas diferenciadas: Exterior por un lado y cada edificio por separado. El problema que se nos plantea ahora es que en el exterior han de estar los dos edificios al completo, y no solo eso, las ventanas son transparentes y se necesita ver algo del interior para crear realismo. Si incluimos todo el edificio estaremos en el caso anterior y no ahorraremos ningún recurso. Hemos de volver al principio, a los archivos de *AutoCAD* para realizar las partes visibles desde el exterior de cada edificio por separado. Lo mismo nos pasará con las escenas de interior, desde dentro no vemos objetos como cubiertas, muros de ladrillo exteriores y cosas así. El tomar la decisión de hacer el trabajo por escenas nos llevará a replantear desde el comienzo todo el proceso de trabajo.

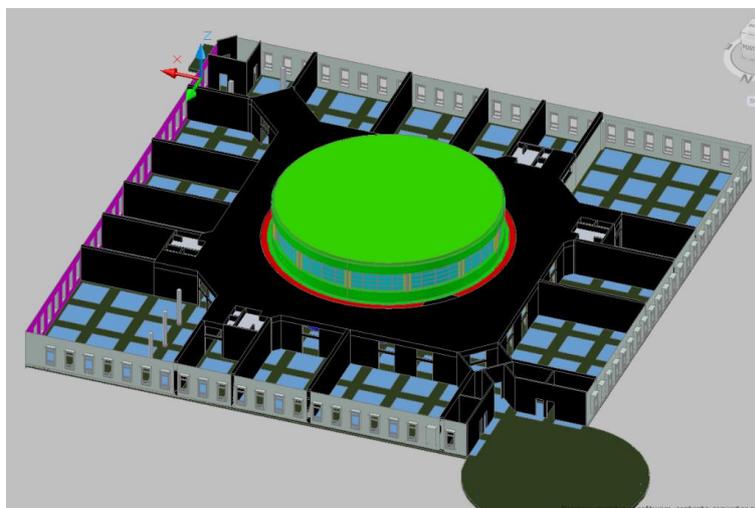


Figura 2.6-05: Segunda planta de la Escuela con los elementos que se ven desde el hall.

Se pensó que al introducir cada edificio con sus aulas en una sola escena, el rendimiento de la tarjeta gráfica del ordenador podría llegar a su límite para ver las escenas con fluidez así que para no estar continuamente rehaciendo escenas se pensó en dividir los edificios en hall y zonas comunes por una parte y aulas y recintos cerrados por otra. A la hora de aplicar iluminación con sombras se descubrió que los resultados son mucho más realistas si en cada escena hay solo un nivel, con lo que la división de escenas quedó de la siguiente manera:

- 01 – Exterior
- 02 a – Escuela Planta Baja y Conserjería
- 02 b – Escuela Planta 1ª
- 03 a – Edificio Tecnológico Plantas baja y Primera
- 03 b – Edificio Tecnológico Plantas 2ª, 3ª y 4ª
- 04 – Salón de Actos
- 05 a – Biblioteca Planta 1ª
- 05 b – Biblioteca Planta 2ª
- 06 a – Aula 1
- 06 b – Aula 2
- 06 c – Aula 3
- 06 d – Aula 4
- 07 – Secretaría
- 08 – Aula 204
- 09 – Aula 216
- 10 – Laboratorio H3
- 11 – Baño ET
- 12 – Despacho 277

2.6.5- Funcionamiento general de Unity3d

Antes de explicar en detalle cada escena por separado vamos a detallar varios procesos comunes a todas las escenas, de esta forma no será necesario hacer hincapié en estos procesos escena por escena:

Creación de materiales

Ya hemos dicho que *Unity* crea los materiales con diversos tipos de sombreado a partir de una textura base. Estudiaremos en detalle cada tipo de shader utilizado:

- Diffuse

Prácticamente no se ha usado este sombreado ya que es el más sencillo y menos realista. Como ya hemos dicho, al arrastrar una textura a un objeto Unity nos crea este material por defecto

- Bumped diffuse

Para materiales con relieves usaremos la textura base y la textura normal obtenida con PixPlant.

Al seleccionar el material en la Vista de Inspector nos aparece al lado del nombre una lista para elegir el tipo de shader, elegimos Bumped diffuse.

Tenemos la opción de usar un color principal “main color” o una textura base. Para materiales monocromos usaremos el color principal sin textura base, para el resto usaremos la textura utilizando blanco como color principal o algún color suave si queremos dar un efecto teñido.

Para usar la textura normal antes hemos de definirla como normalmap, esto lo haremos en la vista de inspector seleccionando la textura y cambiando el tipo de textura a Normal Map. Si no hacemos esto, al arrastrar la textura al material nos aparecerá un aviso diciendo que no está definida como Normal Map, dándonos la opción de transformarla haciendo clic en “fix”.

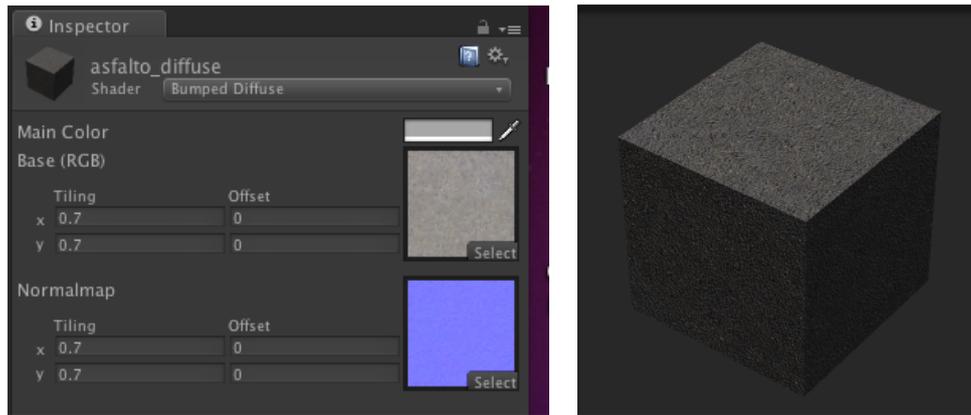


Figura 2.6-06: Configuración de textura bumped diffuse

Solo nos queda adaptar la textura al objeto con las herramientas Tiling y Offset:

Tiling es el numero de veces que se va a repetir la textura por unidad de medida (en nuestro caso por metro lineal) tanto horizontal como verticalmente.

Offset se refiere al punto donde va a comenzar la textura tomando valores de 0 a 1 también en las dos direcciones de la superficie. Lógicamente los cambios realizados de tiling y offset en la textura base tendrán que repetirse en la textura normal para crear el efecto de relieve correcto

- **Specular**

Este sombreado lo usaremos en materiales con cierto tipo de brillo. Simplemente elegiremos la textura o color base y la cantidad de brillo la controlamos con la variable shininess. Tenemos la opción de elegir el color del reflejo, esto nos resulta útil para distinguir metales pulidos con un color mas blanco y plásticos mas toscos con un color de brillo más grisáceo y con menos shininess.

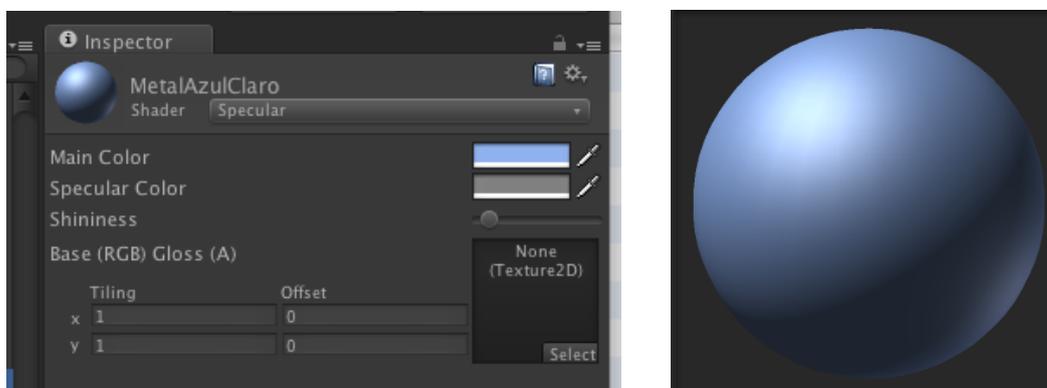


Figura 2.6-07: Configuración de textura specular

- Bumped specular

En este tipo de sombreado tendremos las opciones de los dos anteriores para lograr materiales brillantes con relieve.

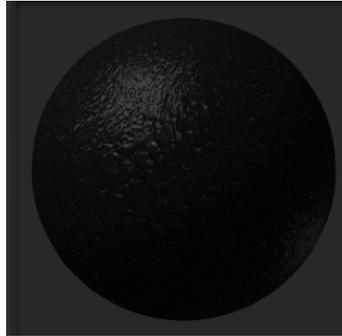


Figura 2.6-08: Textura bumped specular

- Transparent

Para materiales transparentes y translúcidos necesitaremos una textura base la opción *generate Alfa from greyscale* en la vista de Inspector activada. Las texturas que hemos usado son colores planos en varias tonalidades de grises, así cuanto más se acerca la textura al negro, más transparente será el objeto. En nuestro caso hemos usado el shader transparent/specular , de esta forma tenemos las opciones del shader specular con la base transparente para crear el vidrio de las ventanas más o menos opacas dependiendo del uso en la escena.



Figura 2.6-09: Vista Inspector en texturas

- Transparent Cutout

Para crear la malla del banco metálico de el hall de la escuela usamos este tipo de sombreado. En este caso la textura base nos va a definir la parte transparente (negro) y la opaca (blanco). Elegimos una textura blanca con pequeños orificios redondeados negros

activando el Alfa. Eligiendo un color principal cercano al negro y estirando los orificios redondeados obtenemos la textura deseada. Con la variable “Alfa cutoff” seleccionamos el umbral de corte en la escala de grises, esto es, a más cutoff, más fina quedaría la rejilla.

- Self-iluminated

En este caso además del color principal y/o la textura base vamos a añadir una textura “Illumin” que va a ser lo que brille en nuestro material. Hemos elegido una textura blanca para dar una iluminación uniforme y tener la posibilidad de oscurecerla variando el color principal.

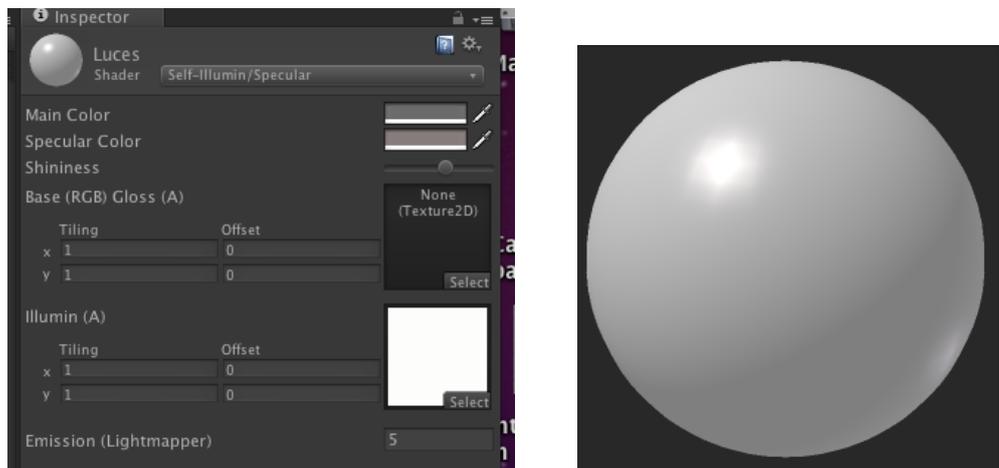


Figura 2.6-10: Configuración de textura Self-Illuminated

- Reflective

Hay varios objetos especialmente pulidos o cromados a los que viene bien este sombreado. Aquí hemos de construir un reflection cubemap que va a ser el objeto que se refleje en el material. Este mapa cúbico es como su nombre indica una cubo con 6 texturas. Para nuestra textura hemos creado un cubemap con una cara negra, otra blanca y un degradado en escala de grises en los laterales. Los resultados obtenidos asemejan un dorado brillante para los metales dorados en puertas y bases de columnas y un cromado plateado para diversos elementos metálicos.



Figura 2.6-11: Configuración de textura reflectiva

Creación de colisiones

Ya se ha explicado la importancia de las colisiones en la construcción de mundos virtuales. Unity nos permite hacer varios tipos de colisiones en cada objeto. Los más utilizados son el *mesh collider* o colisión de malla, que crea colisión en el objeto respetando todas sus formas y vértices y el *box collider*, que crea colisión sobre un cubo que aunque en principio recubre completamente el objeto al que se le aplica, puede ser modificado en tamaño y posición. Como es fácilmente comprensible el uso de mesh colliders es más exacto pero ocupa muchos más recursos y en muchos casos es innecesario.

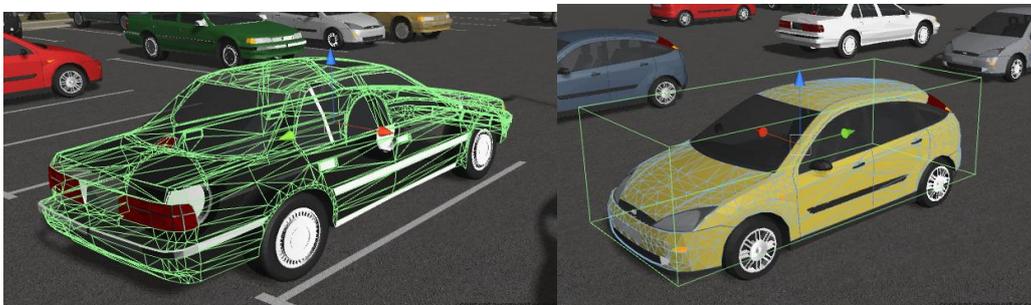


Figura 2.6-12: En las dos imágenes comprobamos como obtenemos funcionalmente los mismos resultados con una *box collider* disminuyendo considerablemente el nº de vértices usado.

Para seleccionar qué elementos llevarán colisiones y de que tipo se estudió cada uno por separado: El suelo o las paredes queda claro que han de llevar colisión, normalmente de tipo mesh; los techos y objetos inalcanzables está claro que no tienen porqué llevar colisiones. El dilema viene en objetos compuestos u objetos muy pequeños. Por ejemplo en las mesas solo es necesario colocar un box collider en la madera, las patas que llevan muchos más vértices no tienen porqué llevar colisión. En pomos de puerta que suelen tener un nº elevado de vértices

tampoco es necesario crear colisiones ya que el objeto principal que no se puede atravesar es la misma puerta. Tampoco es necesario generar colisiones en objetos que recubran paredes como la pared de mármol, los radiadores grandes o los paneles informativos. Recordemos que la cantidad de recursos que gasta un objeto va a depender principalmente del nº de vértices que tenga, y no de su tamaño real: de esta forma ocupará más un pomo de una puerta con caras curvas que la representación de un edificio exterior mediante tres o cuatro prismas rectangulares.

Creación del controlador principal

Como hemos dicho anteriormente vamos a usar como controlador principal un objeto prefabricado (prefab) que nos da por defecto Unity. A este controlador le vamos a hacer ciertas modificaciones en su comportamiento para adaptarlo a nuestro entorno.

El controlador es simplemente un objeto con forma de cápsula al que se le han añadido una cámara y varios scripts con funciones de movimiento al entrar ordenes por medio bien del teclado, el ratón u otro interface externo.

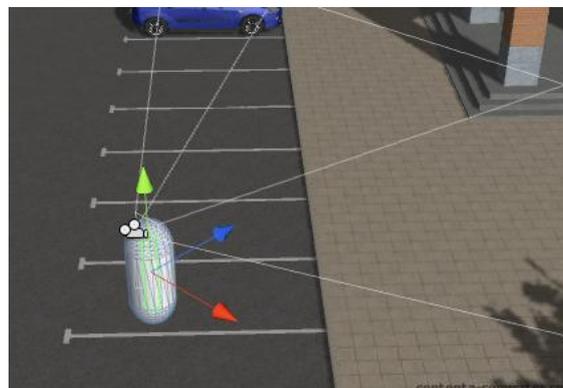


Figura 2.6-13: Controlador

Para configurar estos elementos de entrada hemos de ir al submenú Project settings en el menú edit y elegir la opción input. En la vista de Inspector nos aparecen todos los movimientos o acciones que podemos realizar con sus respectivas teclas de entrada. Esto lo podemos modificar añadiendo otras personalizadas o quitando las que no se usen. A la hora de realizar el script nos referiremos a las teclas de entrada por el nombre de la acción que realicen (jump será salto y se referirá por defecto a la tecla space del teclado convencional).

En nuestro controlador en 1ª persona vamos a modificar en primer lugar la altura del personaje, por defecto viene de 2m. y al atravesar puertas casi roza el listón superior del marco. La altura para asegurarnos de que no tiene problemas la dejaremos en 1.6. Lo mismo le pasa al radio, por defecto viene de 0.4m y de esta forma no podría atravesar puertas de hoja inferior a .8m. Lo colocamos un radio de 0.15 metros ya que aunque pueda parecer exageradamente estrecho nos facilitará el paso en filas de asientos.

Hay una variable llamada slope limit que limita el ángulo de inclinación de la pendiente que el controlador es capaz de subir. Vamos a situar en 70 grados este límite para asegurarnos de que el controlador suba perfectamente escaleras. También limitaremos la altura de salto que por defecto viene de 1m. Y nos parece un poco exagerada, lo reducimos a 0.4m. También vamos a reducir la sensibilidad del giro de la cámara con el ratón ya que en nuestro caso parece muy sensible.

El resto de variables se refieren a acciones físicas como gravedad y velocidades en los distintos ejes que iremos variando según la escena. Por defecto viene el valor 6, que es de suponer una velocidad de 6 m/s. En casi todas las escenas bajaremos esta velocidad bastante, una persona andando en un aula es bastante más lenta, seguramente menos de 1 m/s, en el hall de la escuela podría llegar a 2m/s andando rápido. En el exterior en cambio aunque la velocidad es exagerada nos viene bien aumentar algo la velocidad para perder menos tiempo al ser un espacio tan grande.

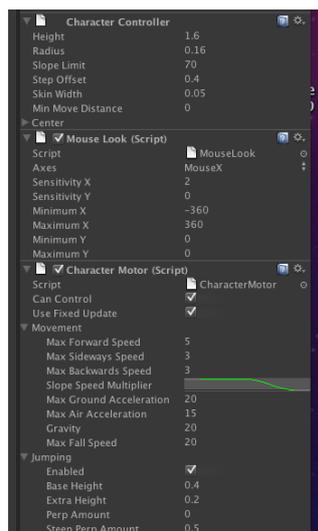


Figura 2.6-14: Configuración de controlador

Duplicación y cambio de eje de coordenadas en los objetos: trabajo con objetos

Una de las herramientas más potentes de el motor gráfico de Unity es la capacidad de duplicar objetos consumiendo un mínimo de recursos. No va a ser lo mismo el importar tres armarios iguales que el importar uno solo y duplicarlo con Unity. Aunque el resultado sea el mismo, el ahorro de recursos es considerable. Por otro lado Unity maneja los objetos de una forma peculiar, además de poder crear formas simples como cubos, cilindros y esferas, Unity puede crear objetos vacíos. Cualquier objeto importado o creado puede depender de otro simplemente arrastrándolo a este segundo objeto en la Vista de Jerarquía. De esta forma es sencillo manejar un grupo de objetos como por ejemplo una fila de sillas creando un objeto vacío, llenando éste con la fila de sillas y a partir de ahí cambiar de posición, girar o cambiar de tamaño a todas a la vez y duplicarlo las veces que sean necesarias. Para cambiar el punto de giro o pivot en un objeto basta con crear uno vacío, colocar el vacío en la posición que queramos y llenarlo con el objeto al que queramos cambiar el pivot.

Componentes de los objetos

Los componentes de los objetos van a definir el comportamiento de cada objeto dentro de Unity. Estos componentes son muy variados. En el menú Components podemos elegir de muchos tipos; los fundamentales serían los de malla o *mesh*, que definen que malla corresponde a cada objeto, los físicos o *physics* donde encontramos la generación de colisiones y la creación de controladores y los de renderizado o *rendering* que definen como se va a ver el objeto y los scripts, que permiten asignar programaciones de muchos tipos a cada objeto o grupo de objetos.

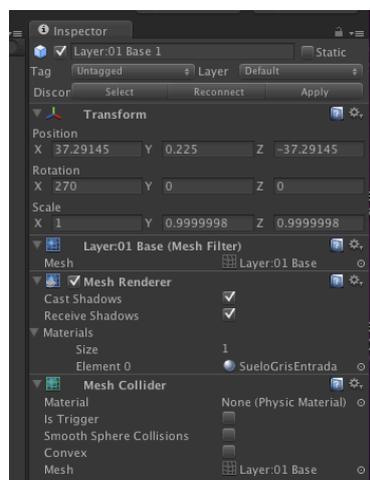


Figura 2.6-15: Componentes en Vista de Inspector

Creación de luces

Además de la acertada elección de materiales para los objetos, la principal cualidad de *Unity* para asemejar la realidad, y uno de los motivos para elegir este programa para la realización del proyecto es la capacidad de creación de luces y la generación de sombras a partir de ellas.

Unity utiliza tres tipos de luces:

- **Point lights** o luces puntuales – Son como su nombre indica puntos de luz en los que podemos variar, además de posición como en cualquier objeto de *Unity*, el alcance, el color de la luz, la intensidad e incluso podemos dibujar un halo o pequeña neblina alrededor de ellas para distinguir su posición. Estas luces no nos van a generar sombras, solo afectarán iluminando más las caras de los sólidos a las que ataquen más perpendicularmente oscureciendo aquellas en las que el ángulo de incidencia sea mayor y más aun las caras no afectadas por ellas.

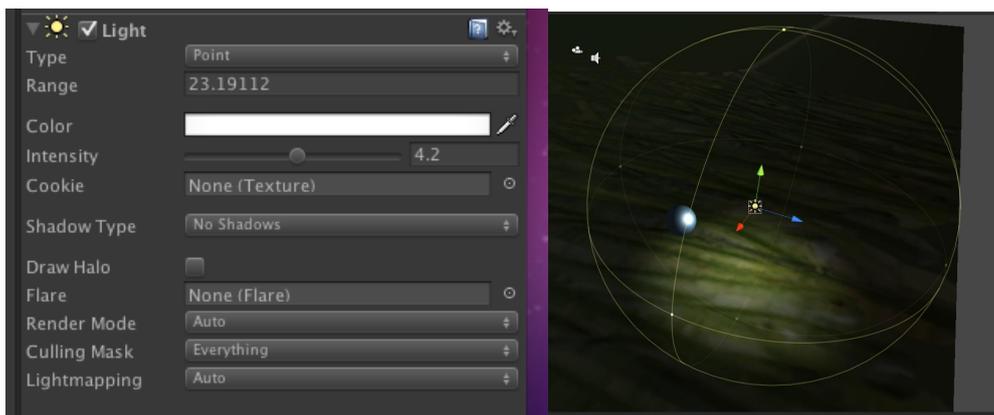


Figura 2.6-16: Luz puntual

- **Directional lights** o luces direccionales – Además de los controles de los puntos de luz nos van a dar la posibilidad de crear sombras. Estas pueden ser suaves o duras (soft o hard) Sobre estas sombras hay ciertos parámetros modificables como la fuerza o Strength, la resolución el bias o la suavidad de los límites de la sombra que ajustándolos de forma óptima nos dan una sensación a la vista muy cercana a la real. Para escenas interiores nos dieron un pequeño problema, la posición de la luz es independiente de la creación de la sombra, la sombra se crea siempre aun cuando el objeto está por detrás de la luz. Para crear sombras en espacios interiores la opción pasa por desactivar la captación de sombras en los objetos que tapen el interior como paredes y techos y activar solo las correspondientes a los objetos interiores, este proceso se explicará más adelante.



Figura 2.6-17: Luz direccional

- **Spot lights** o haces de luz – Representan focos en los que se pueden regular además de lo mismo que la luz puntual, el ángulo de emisión del foco. No generan sombras pero si crean una sensación parecida en los límites de la luz generada.

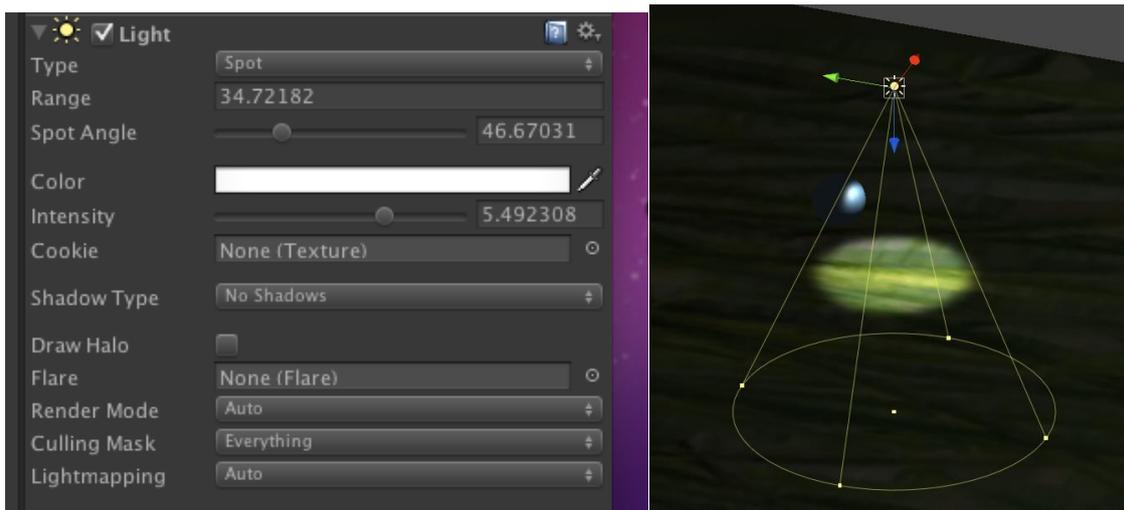


Figura 2.6-18: Spotlight

Generación de cielos y brumas

Otra gran herramienta de *Unity* es la simulación de cielos a partir de skybox. Los skybox son algo similar a los cubemaps que usábamos en las texturas reflectantes. Son cubos con seis texturas de simulación de cielos o fondos del terreno que forman en el renderizado una especie de esfera continua alrededor de nuestra escena. Unity viene con varios skybox preinstalados, en este caso no fue necesaria la creación del cubo. Para seleccionarlo, simplemente en la

configuración del renderizado o *render settings* en el menú de edición nos aparece un apartado para seleccionar el tipo de cielo que queremos.

En este mismo menú tenemos la posibilidad de crear niebla o fog eligiendo el color, la densidad, el punto de inicio y la fuerza de la niebla. Mediante esta herramienta podemos crear una leve bruma que nos da una sensación similar a la de la mañana de un soleado día de verano.

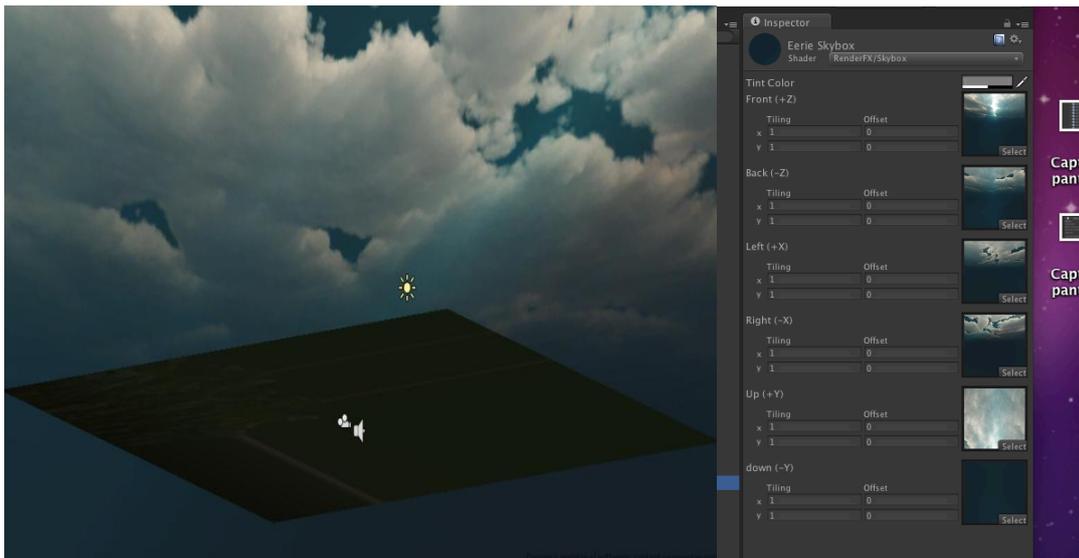


Figura 2.6-19: Ejemplo de skybox nublado

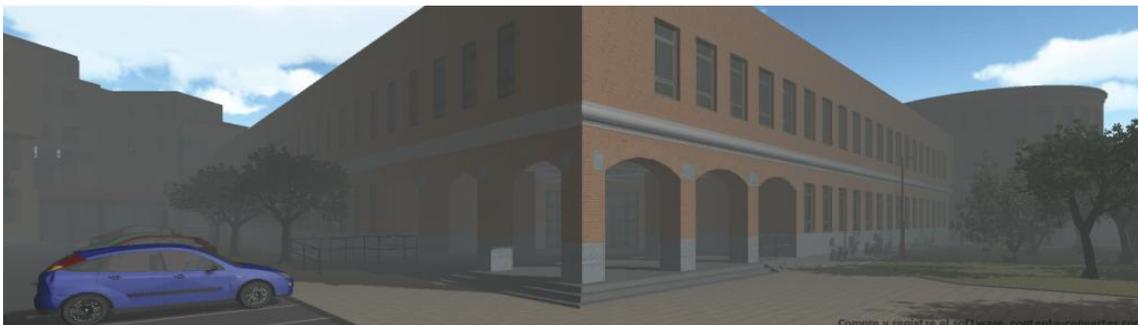


Figura 2.6-20: Vista de la Escuela envuelta en niebla

2.6.6 Proceso de creación de escenas

Una vez expuesto brevemente el funcionamiento de *Unity* vamos a explicar en detalle las escenas más representativas obviando los procesos que se hayan explicado en escenas anteriores para no extendernos en la redacción de este proyecto.

2.6.6.1 Escena 01 Exterior

Para plantearnos esta escena hemos partido de un archivo de *AutoCAD* realizado hace tiempo que representaba el campus tal como estaba en el año 2000, que fue cuando se hizo. Este archivo ya había sido modificado en el proceso de recopilación y actualización de datos con *AutoCAD*, se había recortado para tener solo los objetos que nos interesan y además se había actualizado con los edificios que faltan: la biblioteca de Ciencias Económicas y Empresariales y el edificio de la Facultad de Educación.

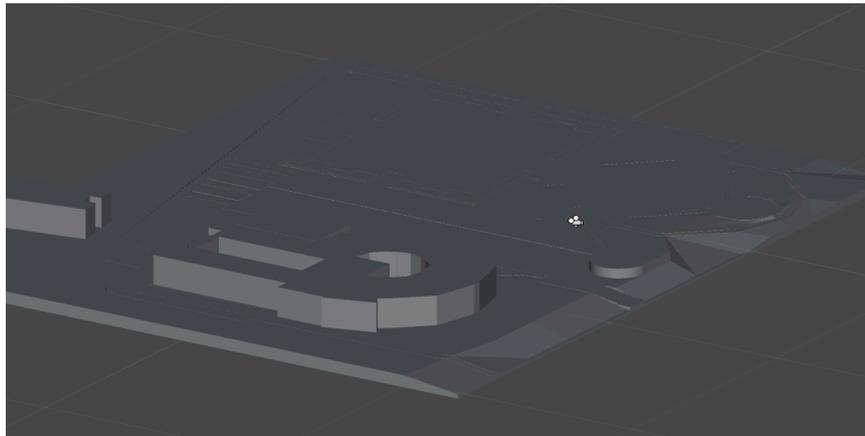


Figura 2.6-21: Terreno base sin materiales definidos

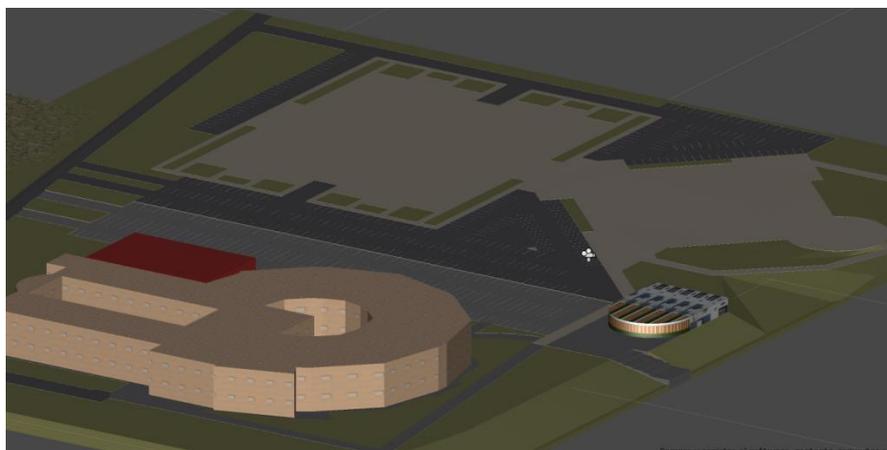


Figura 2.6-22: Terreno base con materiales

A pesar de estos retoques, el exterior seguía pareciendo irreal dada la cantidad de pequeños detalles que faltan en el dibujo, esencialmente los arboles, el terreno no siempre plano, el cielo...

Creación del terreno

Unity nos provee de una herramienta excepcional para creación de terrenos exteriores, recordemos que *Unity* está pensado para la creación de videojuegos y éstos en muchos casos transcurren en exteriores. Hemos usado esta herramienta fundamentalmente para la sustitución de la capa “césped” del dibujo original hecha a base de mallas con caras planas, y para la colocación de arbustos y árboles.

Mediante esta herramienta creamos un terreno con la orden *create terrain* en el menú *terrain* y nos aparece una base rectangular y en la vista de Inspector la herramienta en sí. Aquí vamos a tener 7 pestañas, las 3 primeras para crear elevaciones en el terreno, las 3 siguientes para diseñar el aspecto del terreno con texturas, arboles y hierba, y la última de configuración del terreno.

En casi todos los casos las operaciones son parecidas, se elige un tipo y un tamaño de pincel (brush) y se actúa directamente con el ratón sobre el terreno dibujado.

Ya que en nuestro caso el exterior es prácticamente plano, solo lo usamos para crear el desnivel de terreno cercano a la cafetería y breves pinceladas en las zonas verdes para que el efecto de la hierba no fuera totalmente plano.

Sí nos fue de gran utilidad la herramienta de creación de árboles. En la pestaña de emplazamiento de arboles nos da la opción de editar árboles. *Unity* en un principio no trae árboles hechos, solamente una palmera que no nos iba a ser de mucha utilidad, pero buscando en la misma página web de *Unity* se pueden bajar gratuitamente varios ejemplos de arboles que aunque no se adapten exactamente a nuestras necesidades, nos pueden servir realizando pequeñas modificaciones. Mención especial se merecen los arbustos alrededor de el edificio de Ciencias Económicas y Empresariales, que tuvieron que ser realizados desde cero ya que no había nada que nos sirviera. Esto nos dio oportunidad de conocer el manejo del creador de arboles de *Unity*:

En el menú *gameobject*, en la pestaña *create other* encontramos *tree* y esto nos crea un árbol sin hojas con una sola rama y sin texturas. En el menú Inspector vemos gráficamente una ventana donde se representa nuestro árbol con la rama que tiene. A partir de aquí podemos crear más ramas, otras ramas sobre estas primeras ramas, y por fin grupos de hojas en cada rama. Sobre cada objeto que creamos podemos actuar sobre el tamaño, las texturas, las inclinaciones y la forma de repartirse sobre el árbol creando poco a poco lo que buscamos. El funcionamiento es bastante intuitivo, incluso podemos actuar directamente sobre la Vista de

escena. A pesar de que el funcionamiento no es complicado, si lo es el llegar a un resultado satisfactorio, y vale más la experiencia en el manejo de la herramienta que el conocimiento intenso de ésta, fue laborioso el llegar a un arbusto que sirviera a nuestros propósitos.

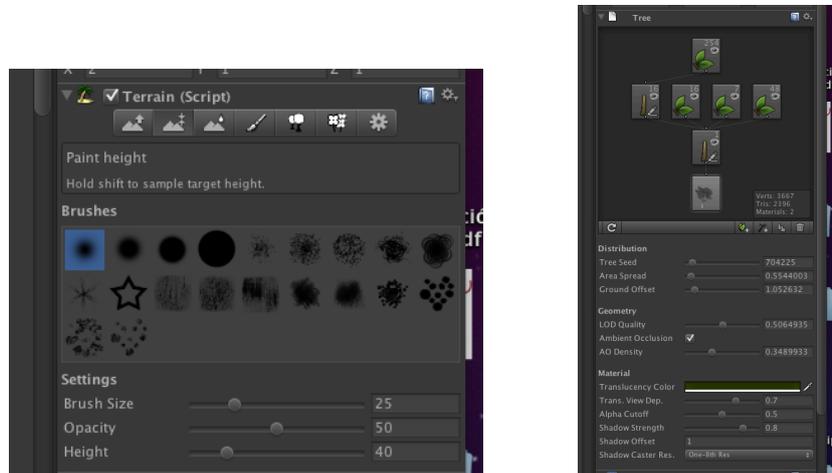


Figura 2.6-23: Herramienta de creación de terrenos y editor de árboles.



Figura 2.6-24: Arbusto creado con Unity

Una vez que mediante creación o modificación de árboles existentes conseguimos tener varias opciones se colocaron los árboles siguiendo como guías fotografías realizadas sobre el terreno real y la ayuda inestimable del *google earth*.

Se ha añadido también en el menú de *render settings* un *skybox* de día soleado un mínimo de bruma que le da al ambiente más credibilidad.

Como broche final para dar realismo a los árboles se creó una zona de viento o *wind zone*. Para ello en la opción de *create* de la pestaña de la vista de jerarquía se seleccionó la opción

wind zone. Esta opción, eligiendo una configuración suave en la vista de inspector, crea un pequeño movimiento en las hojas que da un realismo excelente a nuestros recién creados árboles.

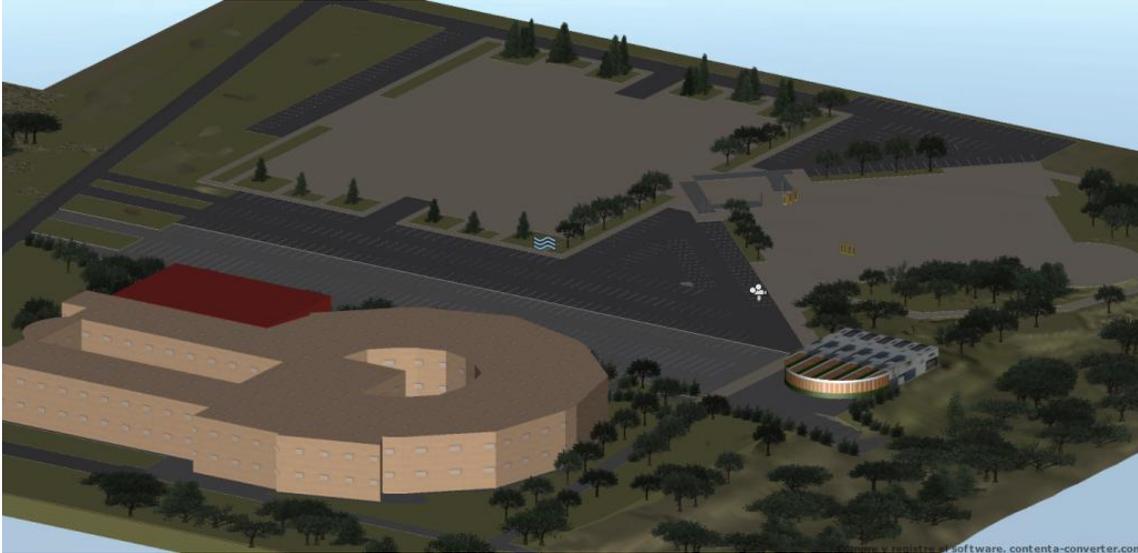


Figura 2.6-25: Terreno

Posicionamiento de los edificios.

En el archivo de AutoCAD teníamos colocados ya los edificios colindantes y una referencia de ayuda de dónde iban a ir los dos edificios de nuestro proyecto, esto nos sirvió de gran ayuda. El problema fue, como ya dije anteriormente, que no podíamos incluir todos los archivos que tenemos de cada edificio ya que el consumo de recursos sería exagerado. Se decidió volver a AutoCAD y seleccionando solo las partes vistas desde el exterior crear dos nuevos archivos de los dos edificios. Se incluyeron también algún objeto del interior cercano a las puertas y ventanas translúcidas, para no crear la imagen de un edificio vacío al acercarse a una ventana. Para que estos objetos consumieran un mínimo de recursos se decidió utilizar texturas con sombreado difuso y desde luego no usar colisiones.

Dado que la zona principal de nuestro proyecto es la que engloba los dos edificios y poco más se decidió colocar muros invisibles en una zona determinada para que el visitante se centrara en el entorno de los dos edificios y no se saliera de los límites del terreno. Esto se logró con simples cubos eliminando el componente mesh renderer de la vista de Inspector para volverlos invisibles y manteniendo la colisión.

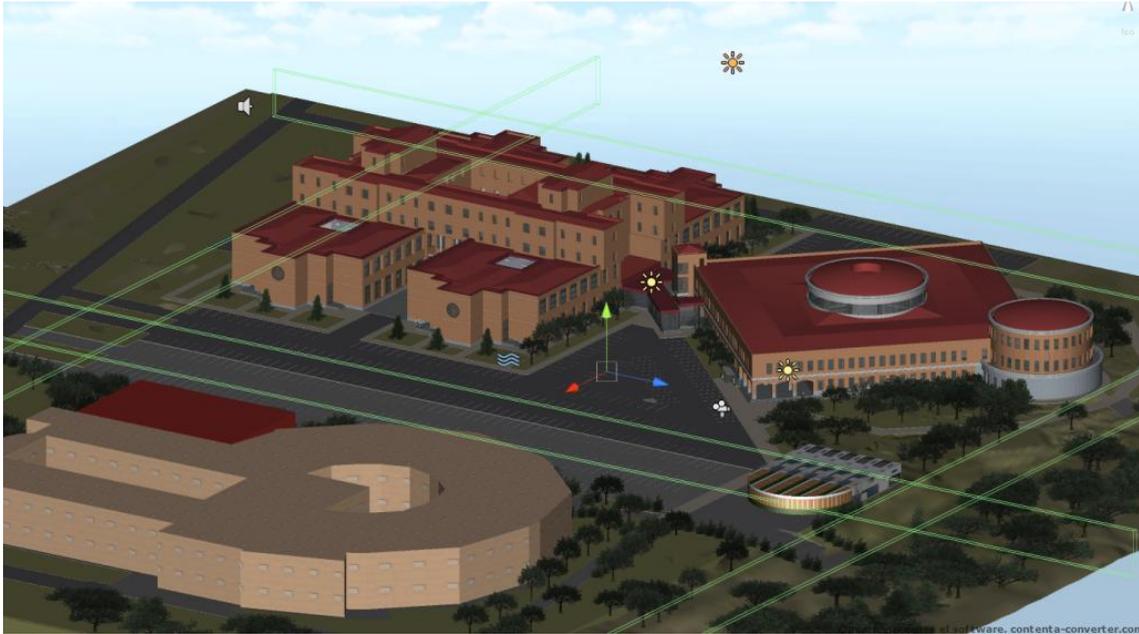


Figura 2.6-26: Terreno, edificios y muro limitador

Añadido de accesorios.

Ahora que ya tenemos los elementos principales de nuestro terreno, ahora vamos a incluir una serie de objetos para conseguir la sensación de realidad buscada. Para ello se pensó en incluir coches en el aparcamiento y farolas alrededor de los edificios. Está claro que hay muchos más objetos posibles para enriquecer la escena pero era necesario poner algún límite ya que en caso contrario el trabajo sería interminable. Esto va a suceder con todos los accesorios de el interior de los edificios. Siempre será posible añadir algo más y siempre nos podremos quedar cortos.

La inclusión de los coches se hizo a partir de modelados basados en mallas que se encontraron en páginas gratuitas de la web. A estos modelos se les tuvo que aplicar ciertas correcciones en AutoCAD similares a las aplicadas en el resto de modelados del proyecto aunque se decidió mantener las mallas en lugar de convertirlas en sólidos ya que no pareció haber incompatibilidades con Unity. Se eligieron tres modelos de coche y se procedió a su duplicado en Unity. Una vez duplicados se usaron distintos colores de materiales en las carrocerías y se colocaron de forma aleatoria por los aparcamientos.

Las farolas se crearon en AutoCAD y se colocaron sobre el terreno de forma similar al resto de objetos del proyecto.



Figura 2.6-27: Terreno final

En el exterior la iluminación ha sido relativamente sencilla, se ha colocado una luz direccional que simula el sol con la inclinación aproximada que habría en un día de primavera-verano a media mañana. El tipo de sombras es soft, este consume más recursos pero es mucho más real.

En un principio parecía que con eso bastaba pero hay zonas donde al estar totalmente en sombra no se apreciaba bien los relieves de los objetos y se decidió colocar luces puntuales muy leves para solucionar este problema semejando la luz que llegaría a estos sitios sombreados reflejada del suelo. Se colocó una luz puntual en la entrada de la Escuela y otra luz puntual dentro del pasillo de la unión de edificios, en un lugar cercano a las puertas del Edificio Tecnológico.



Figura 2.6-28: Escena final iluminada

Animaciones

Las animaciones se introducen en la escena en forma de scripts. Como nuestro objetivo es realizar un paseo virtual no son necesarias muchas animaciones. Únicamente se han realizado animaciones en las puertas de forma que el controlador al acercarse a 5 metros de la puerta, ésta se comience a abrir, y se cierre al alejarse de la puerta. Esto sucederá en todas las puertas que sea posible atravesar, muchas de ellas han quedado bloqueadas ya que su uso real no es muy común o bien por dar acceso a partes del edificio que aun no han sido modeladas.

Esto va a ser común para el resto de escenas. El usuario será capaz de acceder a todas las zonas en las que haya movimiento en las puertas de entrada. Los scripts necesarios para estas funciones se detallarán en un capítulo aparte.

Controlador Principal

El controlador principal lo dejaremos con una velocidad de 5m/s que aunque se corresponda a una carrera difícil de mantener durante mucho tiempo, nos da la posibilidad de recorrer los grandes espacios sin perder mucho tiempo.

Archivos de audio

Como remate final, dado que quedaba un poco extraño ver el movimiento de hojas sin oírlas, se decidió incluir un archivo de audio simulando el viento en las zonas de los árboles. *Unity* reconoce archivos de audio sin comprimir en formatos .wav y .aiff y formatos comprimidos en .mp3 y .ogg

En nuestro caso conseguimos un archivo de viento entre hojas del banco de imágenes y sonidos del Instituto de Tecnologías Educativas del Ministerio de Educación en formato .wav. Lo editamos usando el editor de audio **Bias Peak** formando un bucle de 30 segundos con la parte más apropiada de la grabación y la comprimimos a .mp3 con un bitrate de 192 Kbps y una frecuencia de muestreo de 44.100 Hz. Este archivo lo arrastramos directamente a la Vista de Proyecto y desde aquí a la Vista de Escena colocándolo en la zona más arbolada. *Unity* maneja el audio de forma que varía el volumen según la distancia del controlador al foco de sonido según una gráfica que podemos variar. Dibujamos esta gráfica de forma que fuera aumentando poco a poco el sonido al acercarnos a la zona arbolada entre la Escuela y el edificio de la Radio Universitaria, sin llegar nunca a ser exagerado. Para simular el resto de árboles copiamos el

archivo de audio disminuyendo un poco el volumen y colocamos varios focos en zonas arboladas.

Se introdujeron también archivos de audio simulando la apertura y cierre de puertas disparados por el mismo script que realiza la animación, se verá más detenidamente en el capítulo dedicado a programaciones. Estos archivos de audio se grabaron directamente en las puertas de la Escuela con un grabador de audio digital **TASCAM** modelo **DR-07**

Aquí vemos alguna imagen con los resultados







Figura 2.6-29: Imágenes del exterior en Unity3d

2.6.6.2 Escena 02a Escuela P. Baja

La escena 2ª correspondiente al hall de la Escuela se dividió en dos subescenas. Dada la cantidad de objetos que se incluyeron se prefirió minimizar gastos de rendimiento para que la aplicación fuera fluida en cualquier ordenador.

Aun así la forma de construirla fue similar:

Colocación de la estructura

La introducción de datos en esta escena es similar a la anterior. En este caso no fue necesario la creación de un terreno ya que estamos en el interior aunque si se dibujó un plano exterior

con textura de césped y se aplicó el mismo skybox que en el exterior para que al mirar por cristales no se viera el vacío.



Figura 2.6-30: Estructura de planta baja

Añadido de Accesorios y elementos repetidos

Como se ha dicho anteriormente vamos a incluir los elementos muy repetidos en cada escena importando una sola malla y duplicándola con *Unity*. En este caso el elemento más repetido son las puertas. Vamos a importar una sola puerta de cada tipo y la posicionaremos en la interfaz gráfica de *Unity*.

En esta escena hemos de recordar que a la hora de transformar el formato de *AutoCAD* .dwg a .fbx nos surgía un problema con elementos curvos y esto era especialmente visible en las columnas con revestimiento blanco. Aunque hemos incluido las columnas originales éstas solo nos van a servir para señalarmos con más exactitud la posición donde vamos a crear las nuevas: Crearemos un cilindro de radio igual a la columna original y de altura la suficiente para que nos llegue del suelo al techo, lo duplicaremos las veces que haga falta y los colocaremos sobre las columnas que van a sustituir. Vamos a hacer lo mismo con la base de las columnas, esta vez igualando también la altura de la base original. Una vez sustituidos todos los elementos cilíndricos eliminaremos las capas correspondientes a las columnas originales. Este proceso lo repetiremos en las escenas que sean necesarias.

Hay una serie de accesorios que no se incluyeron originalmente en el archivo de *AutoCAD* general. Se dibujaron de forma independiente por distintas razones:

Extintores y mangueras – Se realizaron dibujos en archivos independientes porque son elementos que se repiten en los dos edificios y en prácticamente todas las escenas, de esta forma es más sencilla su colocación



Figura 2.6-31: Extintor y manguera colocado

Maquinas dispensadoras, mesas, revisteros y vitrina– Son elementos que han cambiado de posición en los últimos años y pueden volver a cambiar en el futuro, es más fácil manejarlos individualmente y no tener que modificar el archivo de *AutoCAD* original.



Figura 2.6-32: Maquinas dispensadoras y mobiliario del hall

Carteles y paneles informativos – Esos son elementos con mayor movilidad aun, ya que ni siquiera nos valdría la misma textura de un año para otro.



Figura 2.6-33: Paneles informativos

Papeleras y bancos – En este caso se juntan los dos factores, son objetos que podrían moverse en un futuro y además se repiten en los dos edificios.

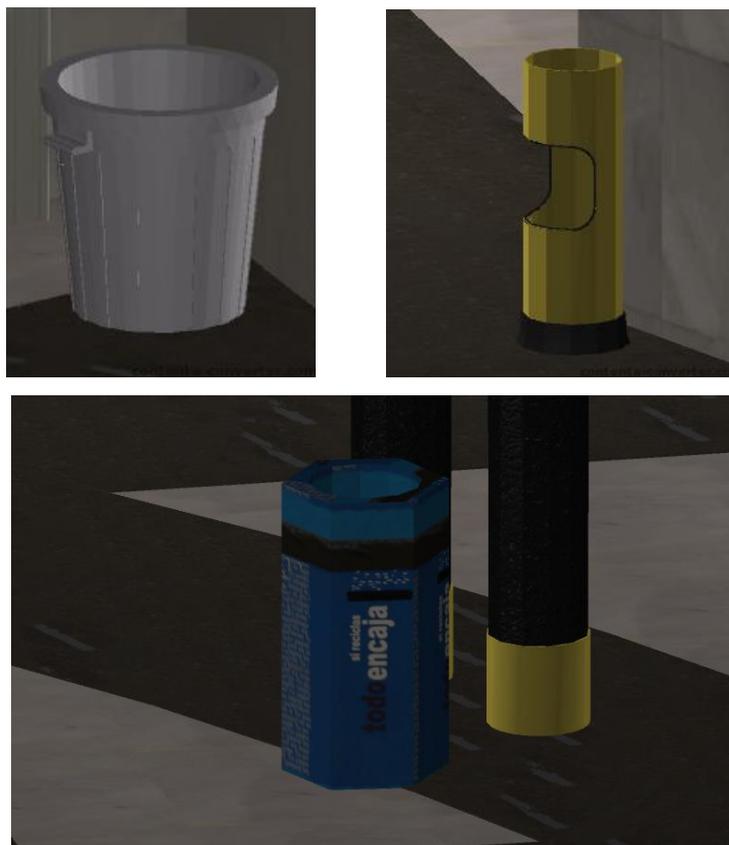


Figura 2.6-34: Papeleras

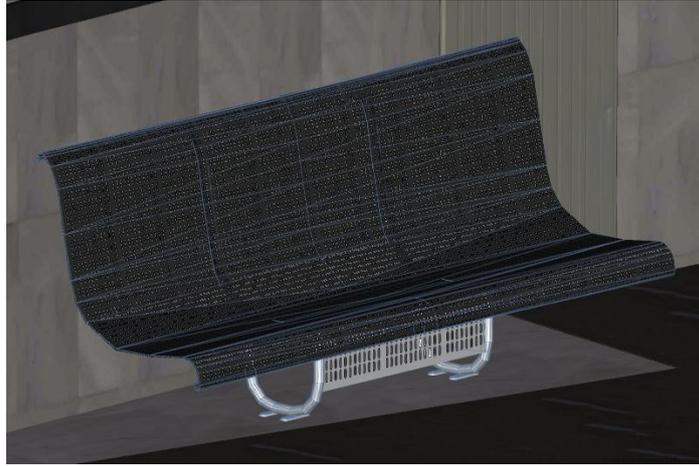


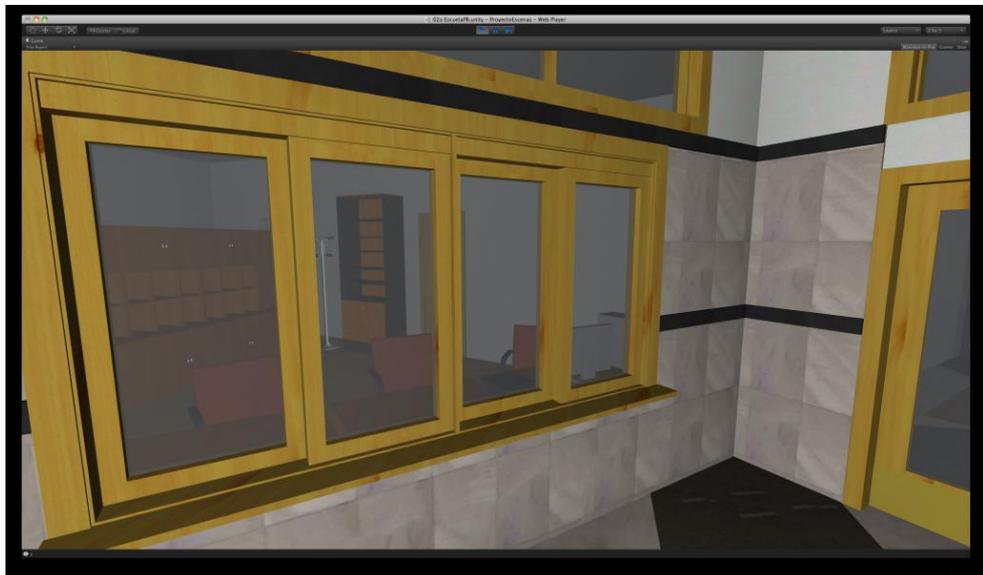
Figura 2.6-35: Banco

Iluminación

En este caso estamos en el interior del edificio, no vamos a tener la luz solar directa pero sí va a crear sombras de gran realismo correspondientes a la luz que entra por las ventanas superiores. Tuvimos atención en colocar la luz direccional con la misma inclinación que en la escena del exterior para que la sensación al pasar de una escena a otra fuera similar.

Además de la luz direccional, se han colocado varias luces puntuales alrededor del hall normalmente coincidiendo con fluorescentes o focos para que el tratamiento de relieves y brillos de las texturas fuera apropiado. Aquí vemos unos ejemplos del resultado obtenido.





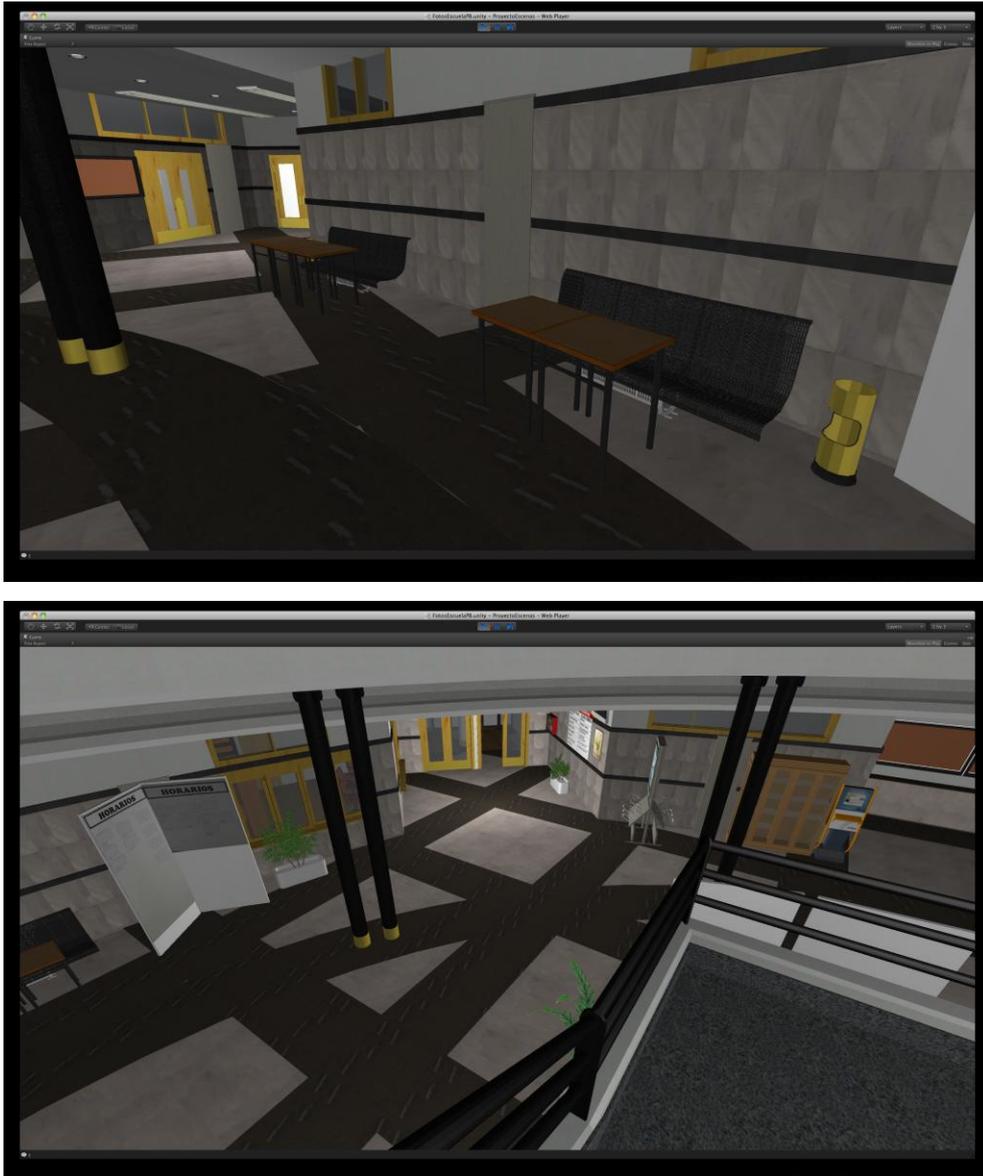


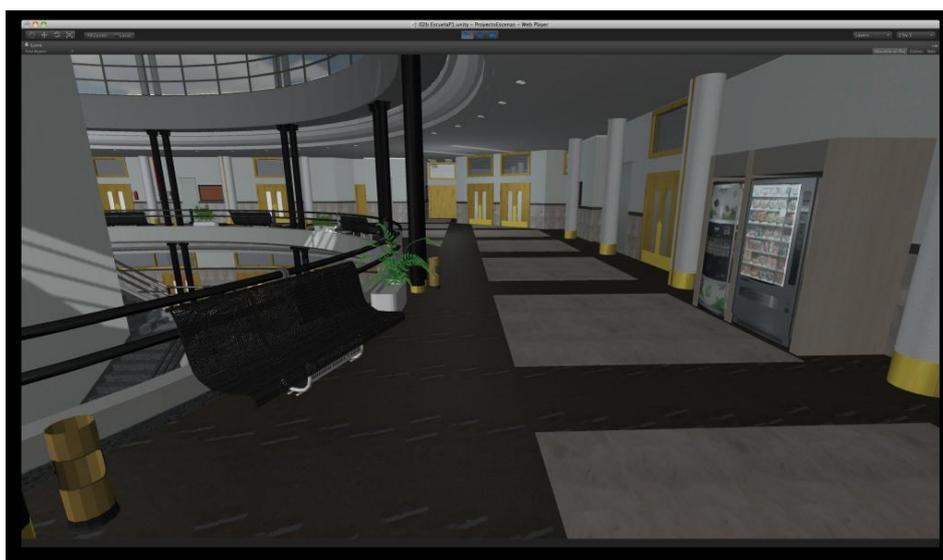
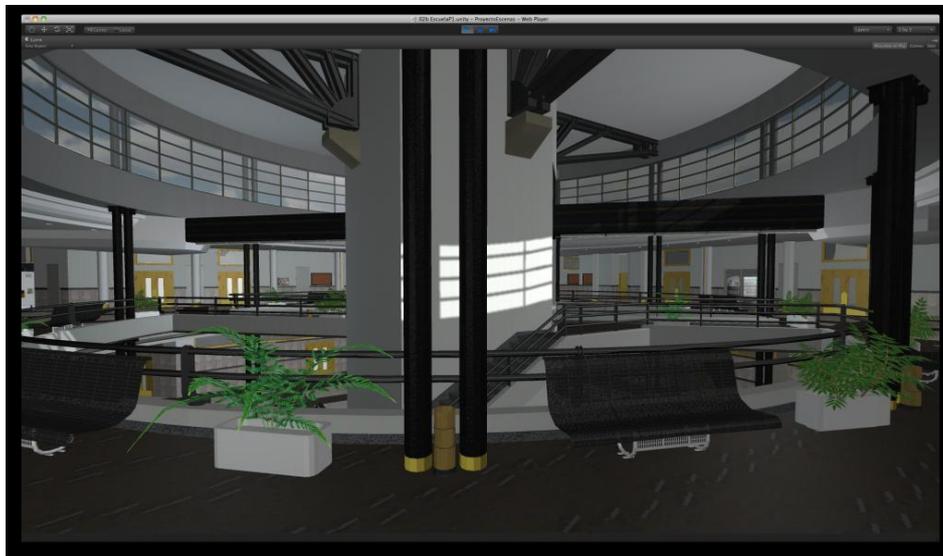
Figura 2.6-36: Imágenes de la P. Baja de la Escuela en Unity3d

El resto de procesos aplicados son similares a los realizados en la escena anterior. En este caso no hay animaciones ya que todos los elementos son estáticos, no se incluyó ningún archivo de audio pues la sala está vacía y sería extraño oír ruidos de ambiente.

Se redujo la velocidad del controlador a 3m/s que nos parece una velocidad lo suficientemente baja como para simular una persona andando rápido y lo suficientemente alta como para que no resulte pesado el paseo virtual.

2.6.6.3 Escena 02b Escuela P. Primera

El proceso de creación de la primera planta de la Escuela es similar al descrito para la primera planta. Aquí vemos unas escenas que muestran el resultado:



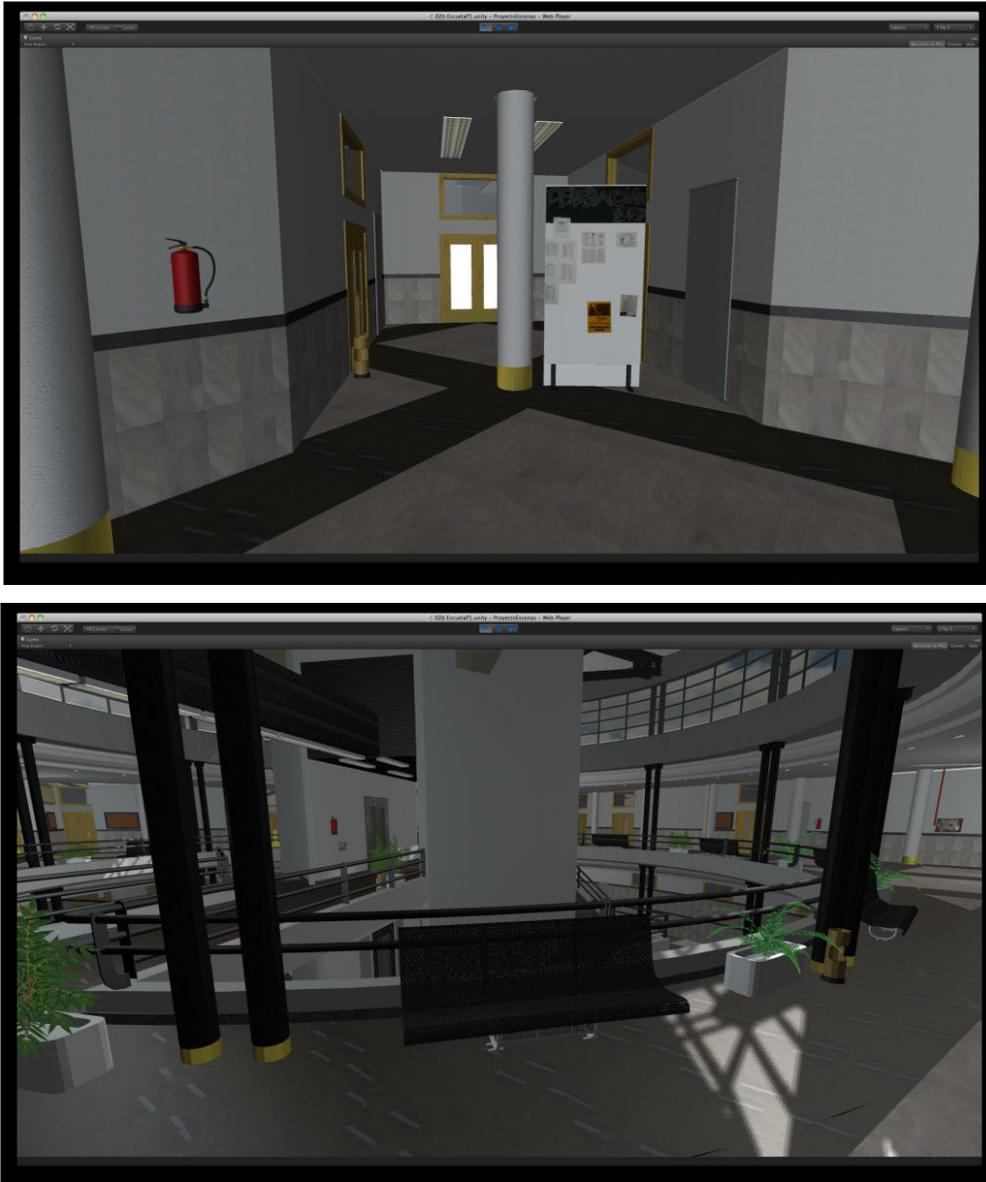


Figura 2.6-37: Imágenes de Primera Planta en Unity3d

2.6.6.4 Escena 03 Edificio Tecnológico

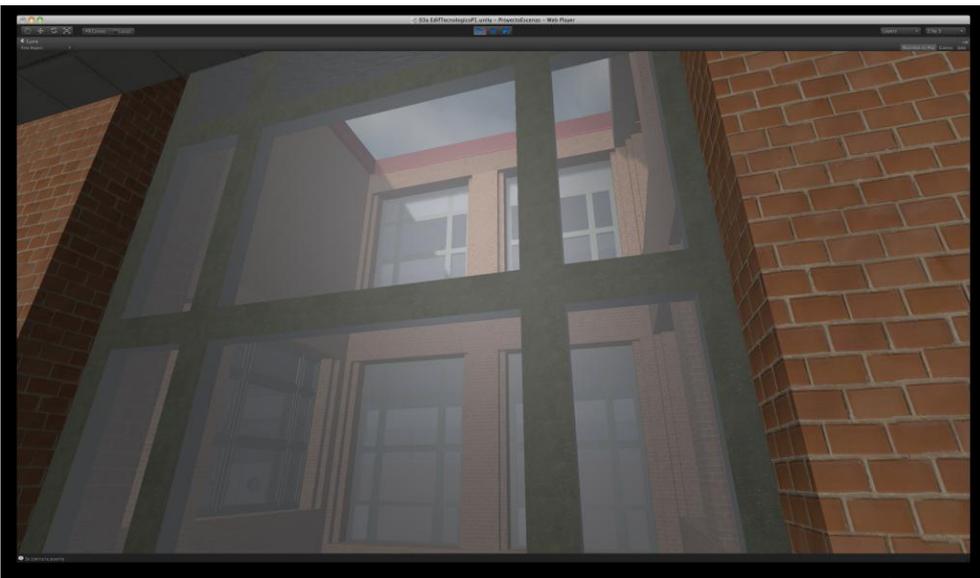
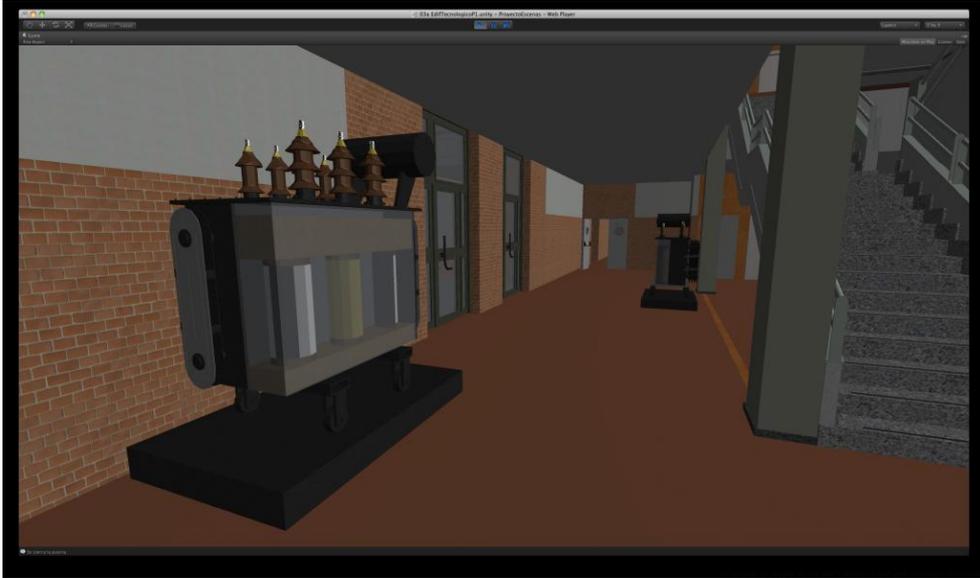
Las únicas variaciones a la hora de introducir datos en este edificio han sido a nivel de iluminación, se ha mantenido la luz direccional exterior para que cree sombras en las ventanas y se han incluido diversos puntos de luz en los distintos pasillos para mejorar el renderizado de los materiales. Caben destacar las luces puntuales en los patios interiores para simular la claridad que entra por éstos.

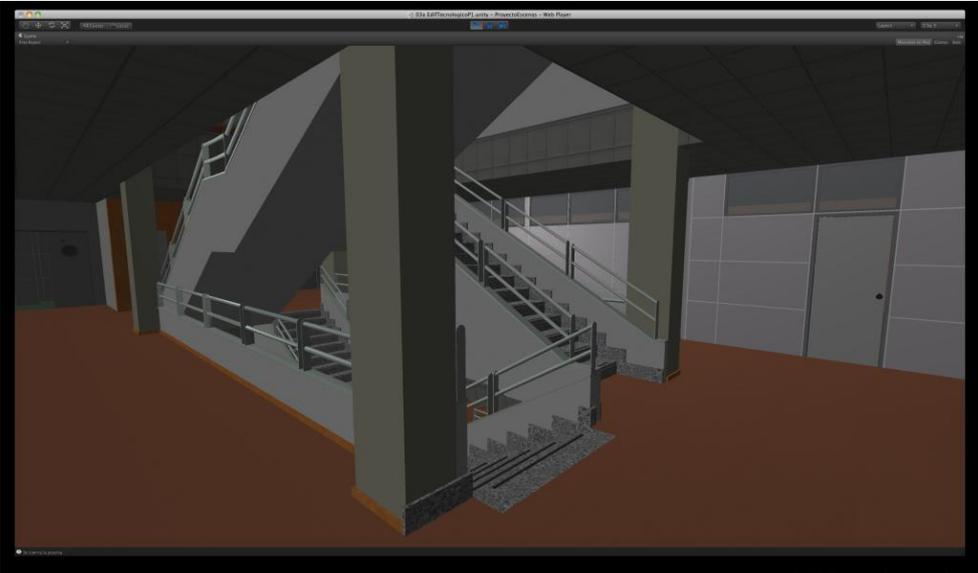
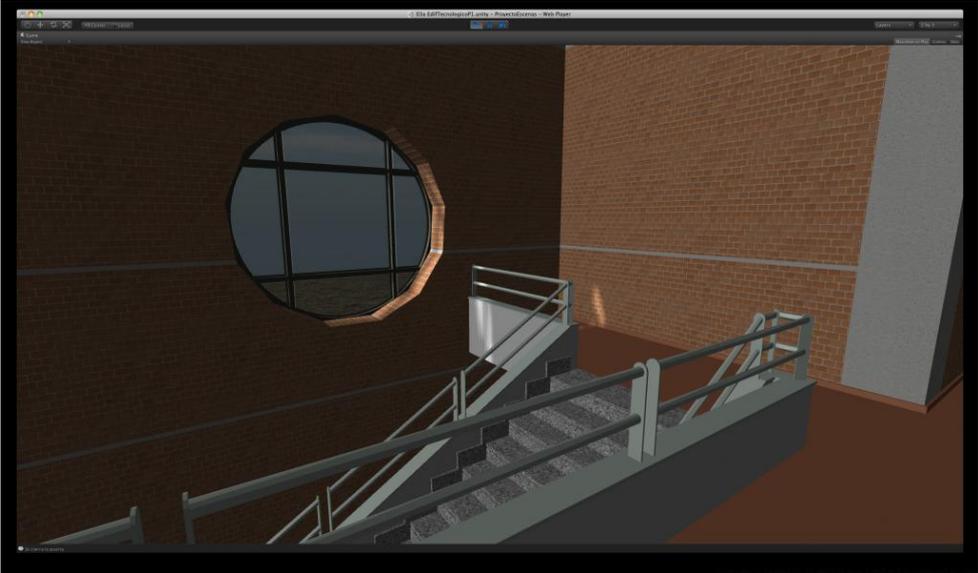
El edificio se dividió en dos partes, por un lado la planta baja y el primer piso y por otro lado el resto de pasillos

La velocidad del controlador se mantuvo en 3 m/s como en el hall de la Escuela.

Vemos unos ejemplos del resultado:







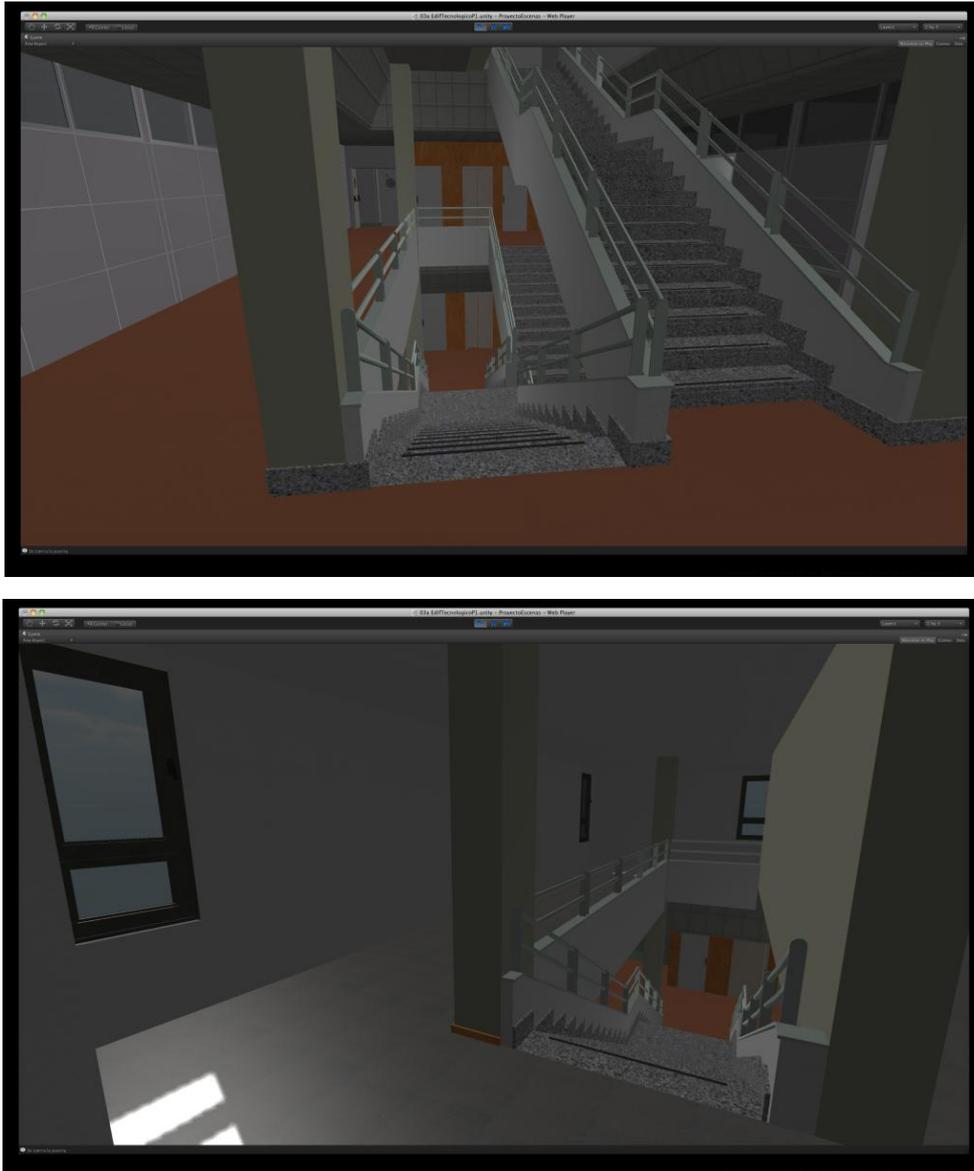


Figura 2.6-38: Imágenes del Edificio Tecnológico en Unity3d

2.6.6.5 Escena 04 Salón de Actos

En esta escena ha habido variaciones importantes en cuanto a la iluminación con respecto a las anteriores:

El Salón de Actos por su funcionalidad es un recinto que va a estar normalmente con las cortinas bajadas en las pocas ventanas que tiene. La iluminación principal va a estar centrada en la tarima principal.

En este caso no nos es necesaria una luz del sol para crear sombras, vamos a aprovechar las luces direccionales para crear sombras artificiales en los objetos del interior del recinto.

Esto lo conseguiremos desactivando la captación de sombras en techos y paredes de forma que la luz direccional nos incida directamente sobre los objetos del interior. Recordemos que las luces direccionales funcionan según la dirección que les demos, no según su posición.

Para eliminar la captación de sombras, en la Vista de Inspector de cada objeto, en el componente mesh renderer desactivamos las casilla de Cast Shadow. Debajo de ésta vamos a tener otra para recibir sombras que se desactivará también en objetos que no van a recibir sombras como bombillas o techos, para ahorrar rendimiento.



Figura 2.6-39: Componente Mesh renderer

Una vez hecho esto vamos a colocar spotlights sobre los focos que están por encima de la tarima, aumentando el ángulo de incidencia hasta 150 de forma que simulen los halógenos reales sobre la pared como se aprecia en la **Figura 2.6-40**. Hemos de asegurarnos de dar el valor important a la casilla de render mode para que todos los focos actúen por igual.

Después colocaremos una luz direccional apuntando hacia las butacas de forma que simule la sombra generada por los focos delanteros. En esta luz variaremos la fuerza y la suavidad de contornos para crear sombras más difusas y simular la acción de varios focos a la vez.

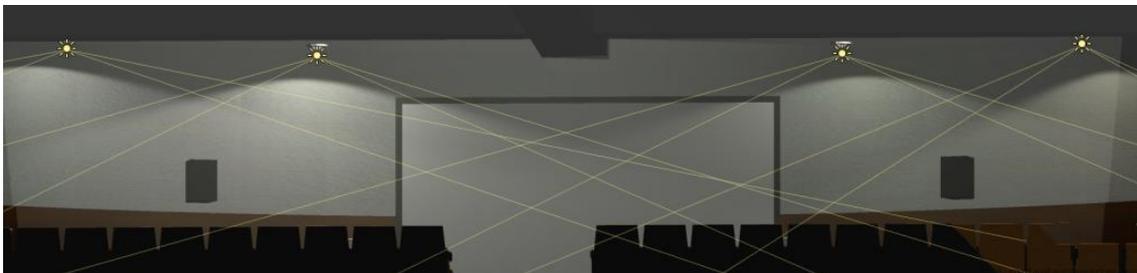


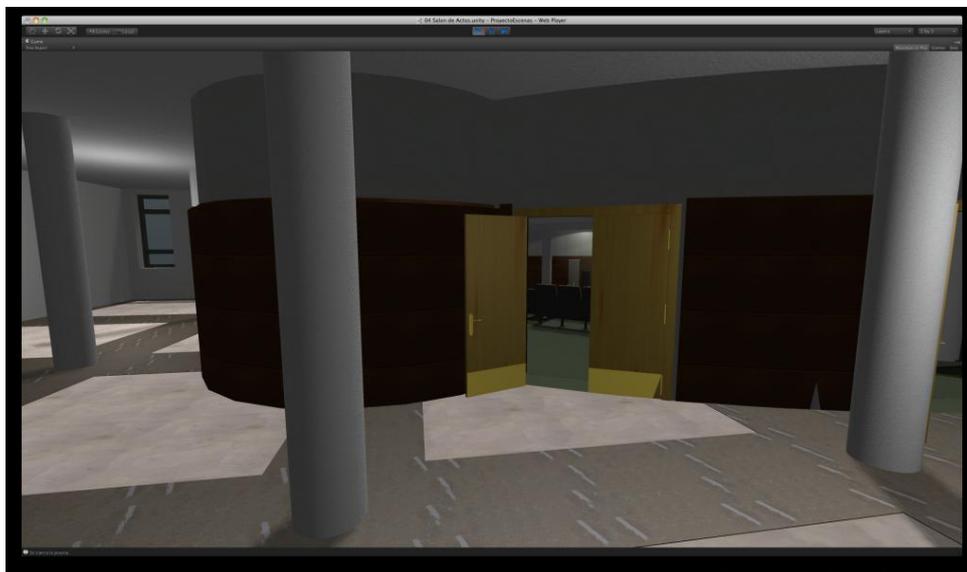
Figura 2.6-40: Colocación de focos en Salón de Actos

Para terminar vamos a colocar una luz puntual por encima de las butacas para que éstas no queden demasiado oscuras, esta luz la colocaremos fuera del recinto para que no cree brillos exagerados en paredes y techo.

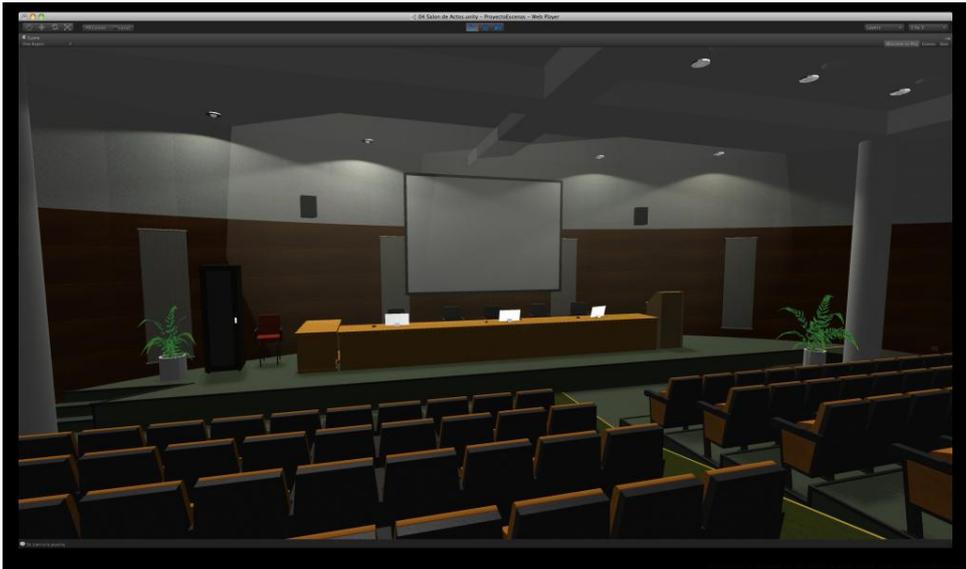
El resultado obtenido es bastante satisfactorio y vale la pena el pesado trabajo de análisis de cada elemento para que capte o no las sombras de la luz direccional.

El resto de procesos para la creación de la escena son similares a los anteriores: También se han sustituido las columnas por cilindros y se han duplicado en Unity elementos muy repetidos para ahorrar rendimientos. Al controlador principal se le dio una velocidad de 2m/s como en el resto de aulas y recintos pequeños.

Vemos unos ejemplos del resultado:







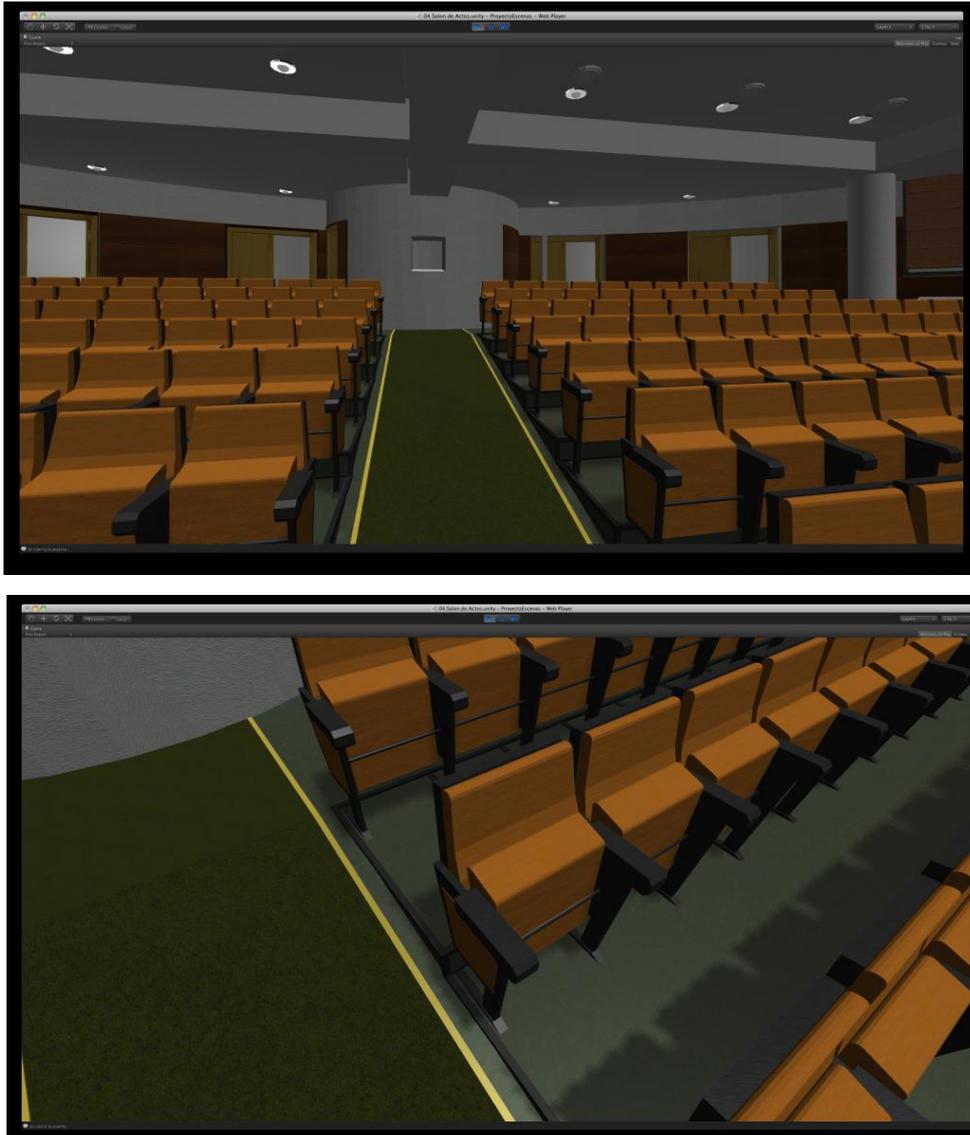


Figura 2.6-41: Imágenes del Salón de Actos en Unity3d

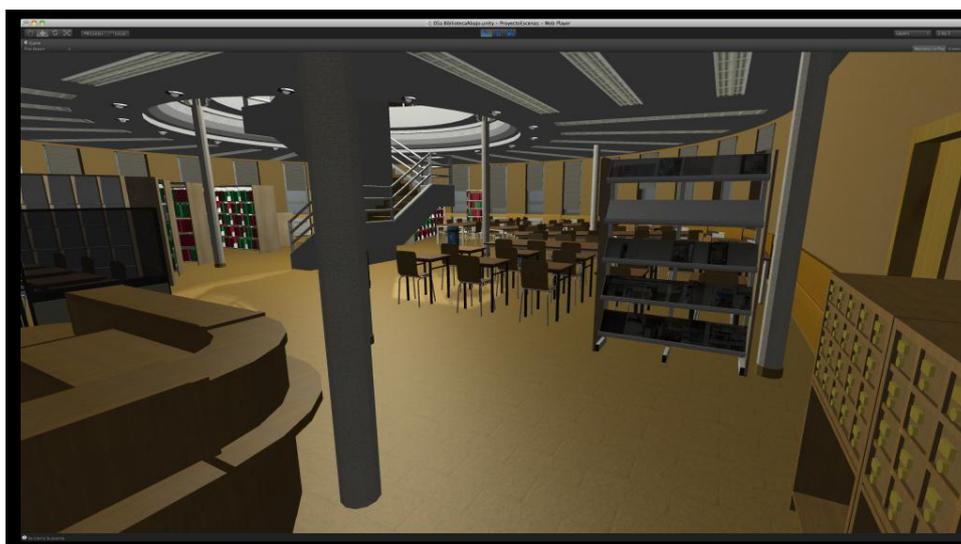
2.6.6.6 Escena 05 Biblioteca

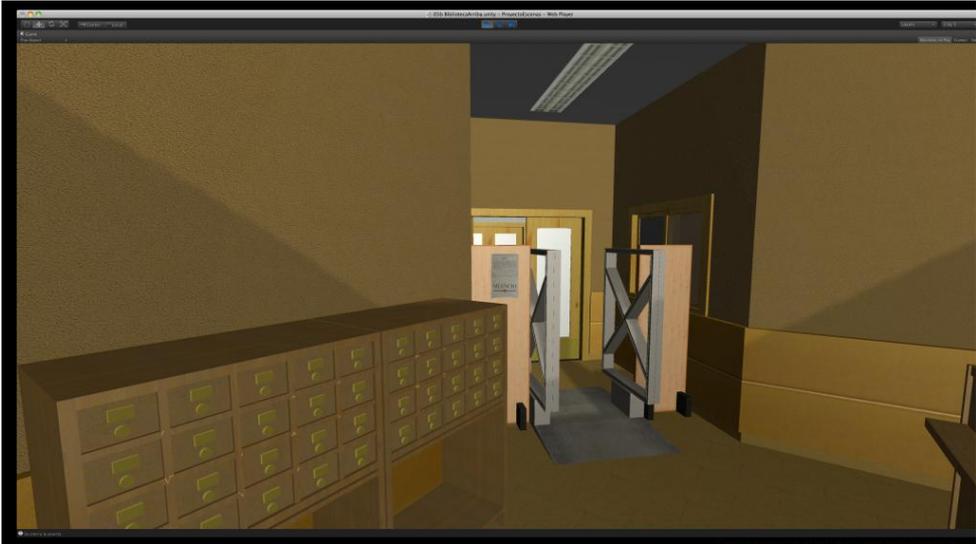
En esta escena se nos plantea la duda de trata la iluminación como en el resto de la escuela, usando la luz direccional como la luz solar creando sombras en las ventanas o repetir el tratamiento del salón de actos. Tras comparar las dos opciones se vio más realista el tratamiento seguido en el resto de aulas, se usó una luz direccional para crear sombras en el interior de la biblioteca.

Se dividió la escena en dos partes fundamentalmente por el tratamiento de luces elegido. Si hubiéramos creado una sola escena las sombras de los objetos del segundo piso nos crearían efectos muy extraños en el primer piso. De esta forma se colocaron las luces de forma

independiente en los dos recintos. Un procedimiento más laborioso pero con mucho mejor resultado.

Aquí vemos alguna imagen:









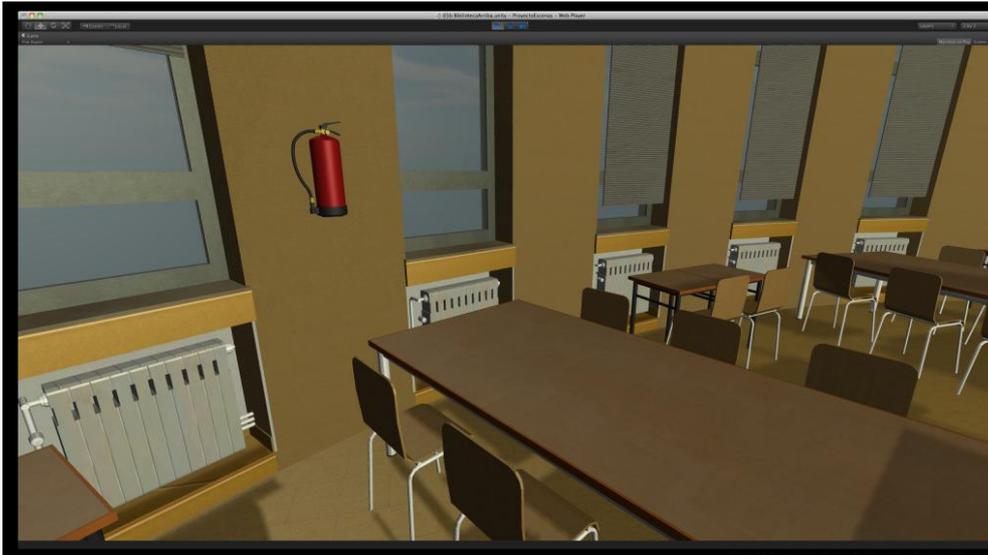


Figura 2.6-42: Imágenes de la Biblioteca en Unity3d

2.6.6.7 Escenas restantes

El resto de aulas y recintos modelados llevan un tratamiento similar al Salón de actos. En todos los casos se sitúan en la parte oeste del edificio de forma que la luz solar prácticamente no afecta a las sombras con lo que se usaron las luces direccionales para crear sombras internas.

Aquí podemos ver algunos ejemplos del resultado obtenido en la secretaría y las distintas aulas de la Escuela:



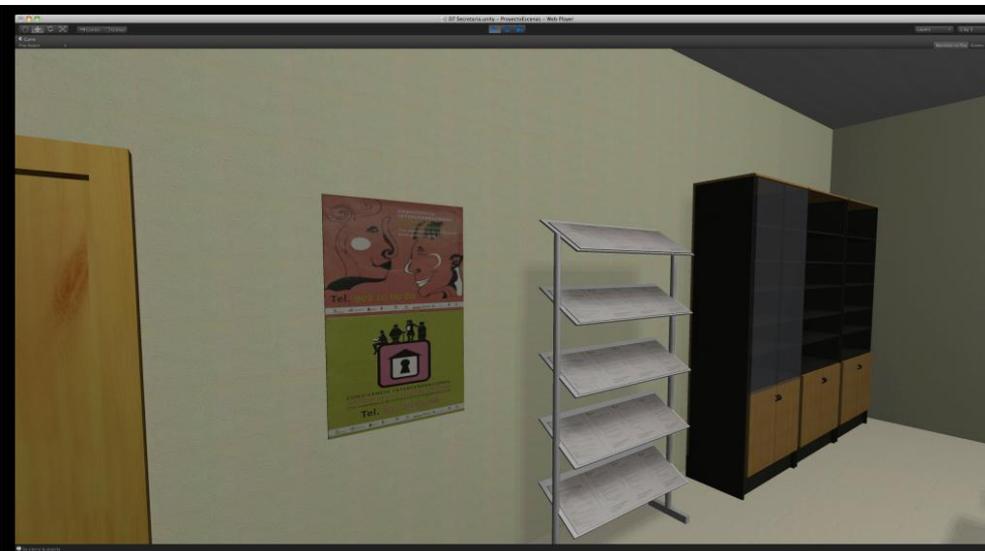
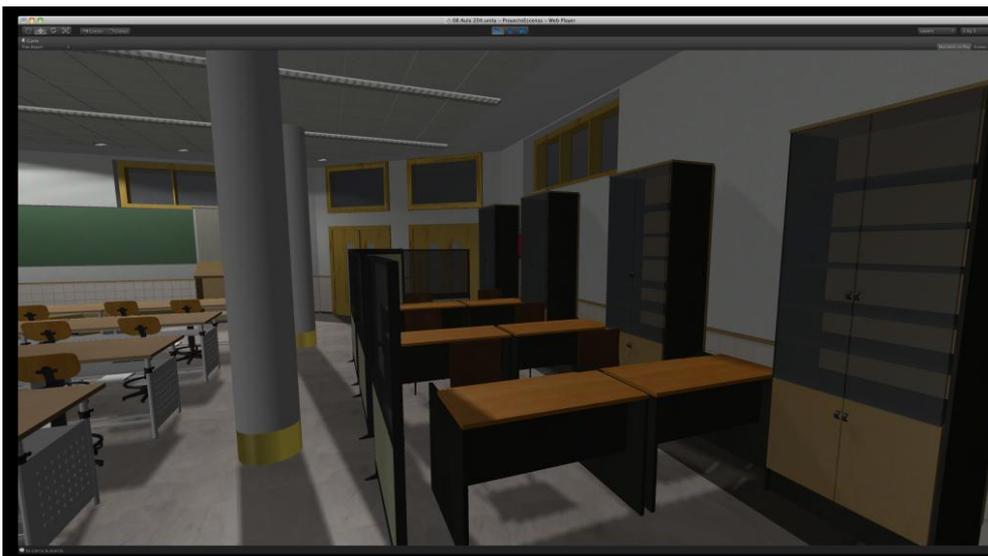
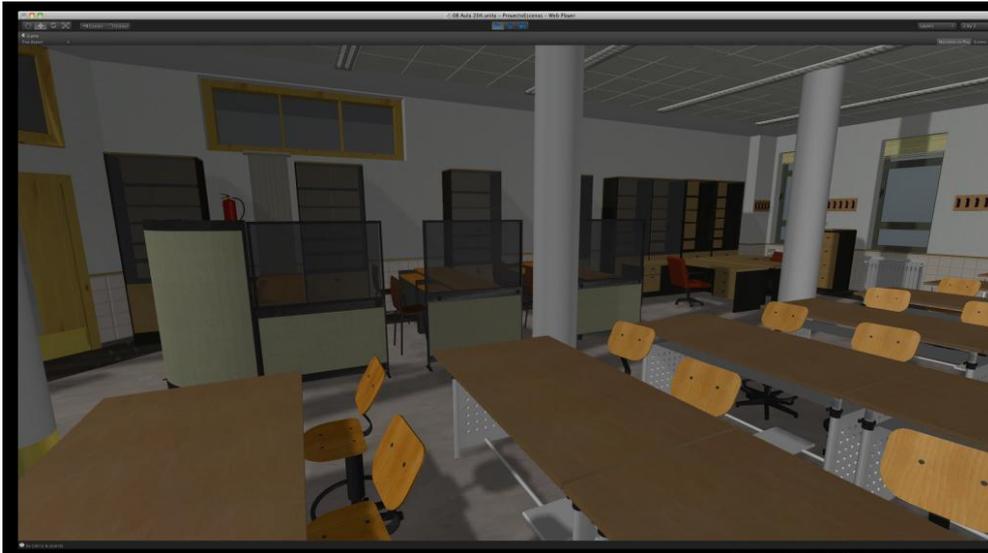




Figura 2.6-43: Imágenes de la Secretaría en Unity3d







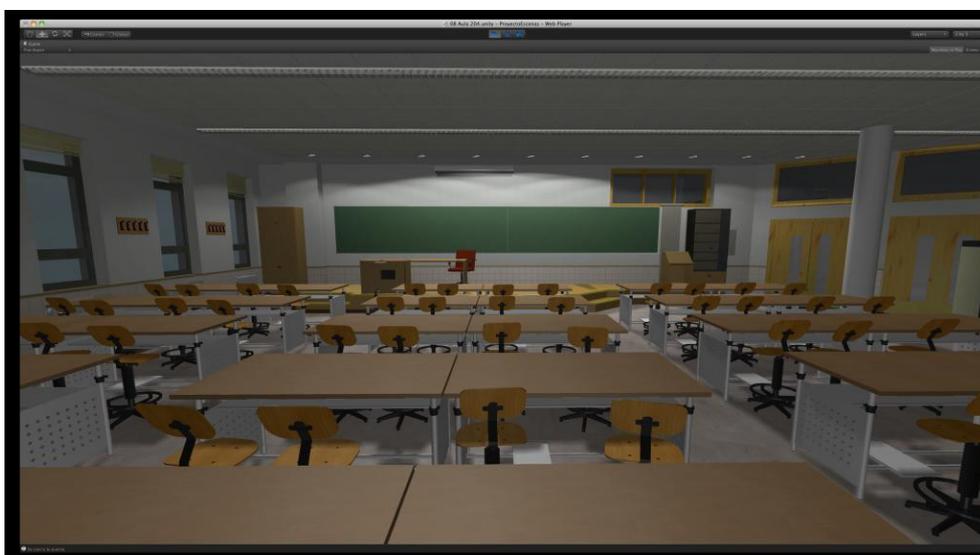


Figura 2.6-44: Imágenes Aula 204 en Unity3d





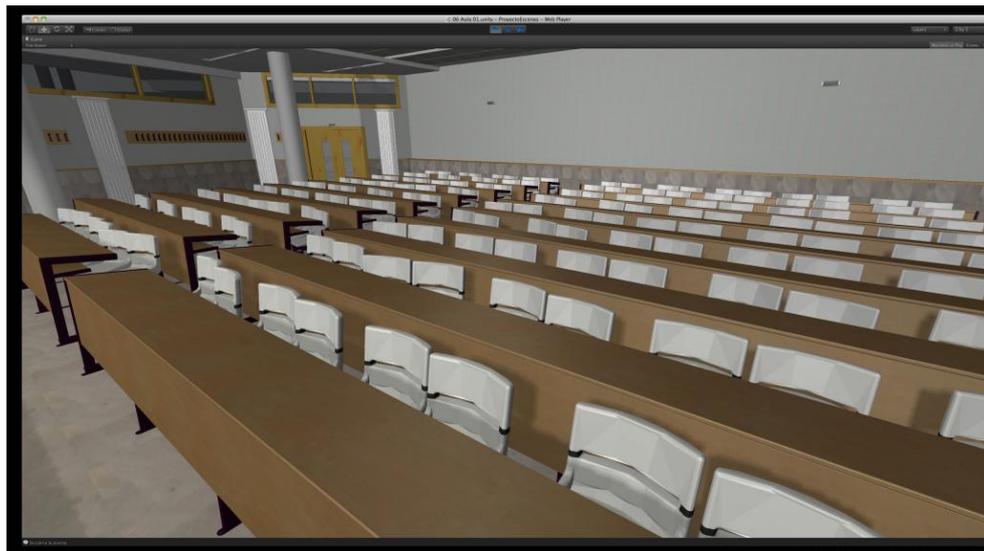




Figura 2.6-45: Imágenes de las Aulas de la Planta Baja en Unity3d

2.6.7 Programación

Unity es un programa capaz de crear animaciones en objetos y personajes con inteligencia artificial avanzada usando complicados scripts. El programa reconoce tres lenguajes de programación: *C#*, *Boo* (un derivado de *Python*) y una versión propia de *JavaScript*. El funcionamiento con scripts es mas sencillo y directo; lo poco que vamos a programar lo haremos de esta forma. La versión que usa *Unity* de *JavaScript* es una compilación con funciones específicas para el programa.

El trabajo con scripts se basa fundamentalmente en definir variables, ejecutar funciones sobre estas variables, aplicar condicionantes a las funciones y crear bucles en que determinen el tiempo de aplicación de la función sobre la variable.

El objetivo de nuestro trabajo se limita a un paseo virtual lo más realista posible así que no nos es necesario el uso de programaciones complejas. Únicamente vamos a utilizar scripts para funciones básicas como el manejo del controlador, el cambio de una escena a otra, la apertura de puertas y la activación de elementos multimedia:

- Manejo del controlador principal

Ya hemos explicado que estamos usando un controlador prefabricado por los creadores de Unity, solamente hemos modificado algún detalle sin tener que cambiar el diseño del script.

- Cambios de escenas

Se ha utilizado una sencilla programación que nos permite además de cambiar de escena, colocar el controlador principal en la posición y orientación buscada dependiendo de la puerta que atraviere. Para ello se han usado 3 scripts:

1- Se crea un pequeño script definiendo la variable *puerta* y dándole el valor 0

2- Se crearon cubos invisibles pero con colisión en las zonas que dividen una escena y otra.

En ellos se introdujo un script para enviar a otra escena al objeto que colisione con el cubo incluyendo un cambio de valor de la variable *puerta* con un valor diferente en cada puerta

3- En el controlador de la escena a la que queremos llegar se introdujo un script de translación de forma que según el valor de la variable *puerta* el controlador se coloca en unas coordenadas y con una dirección apropiada.

Este script fue realizado por el tutor del proyecto Fernando Jorge Fraile. En general todo el trabajo de scripts ha sido asesorado por él.

- Apertura de puertas

En este script se define la variable *AngleY* y varias variables privadas de posición inicial y final de la puerta. También se basa en cubos invisibles que al ser atravesados disparan la ejecución de la función. Para atravesar los cubos y que el script note la colisión activaremos la casilla *on trigger* del componente *collider* de los cubos y usaremos la función *on trigger enter* del script. En este caso vamos a girar un objeto *target* cuando el controlador atraviere el cubo. Para que el objeto girado se corresponda con la hoja de la puerta hemos de crear un objeto vacío con pivote en el eje de la bisagra de la puerta. Este objeto lo hemos llamado *bisagra* y lo hemos colocado en cada puerta para mantener un orden. Dentro de este objeto hemos colocado los objetos de la hoja de la puerta: madera, metal dorado, cristal y pomo, de forma que al girar el objeto *bisagra* sobre su *pivot* giran todos los objetos contenidos en él.

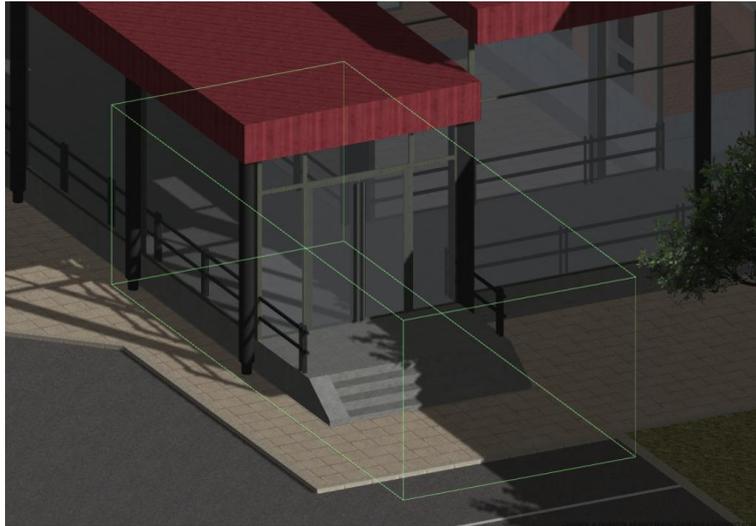


Figura 2.6-46: Cubo invisible

En la **Figura 2.6-47** Se observa como al acercar el controlador a la puerta, ésta se abre.

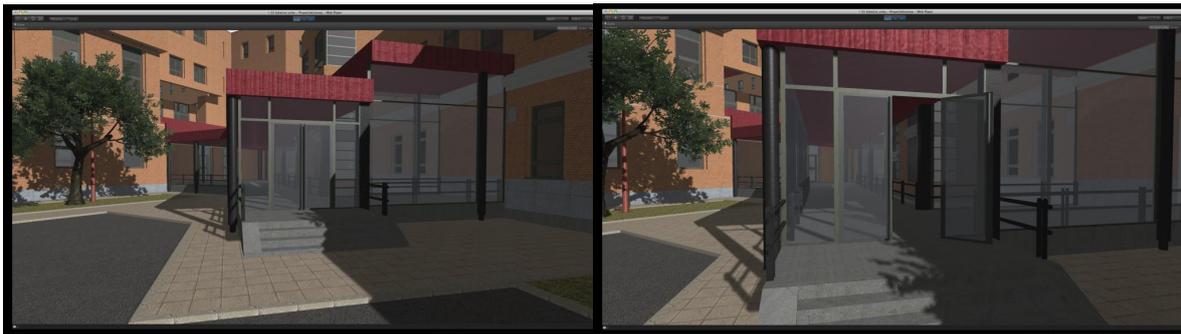


Figura 2.6-47: Apertura de puerta

Además de el giro de puertas se ha introducido un apartado en el script que reproduce sonidos al entrar en el cubo y al salir, o sea, al abrir y cerrar la puerta. Asignamos sonidos que se grabaron directamente en las puertas afectadas, de esta forma se logra el realismo buscado.

Los scripts utilizados están transcritos en el **Anexo C** adjunto a este proyecto dedicado a esta función.

2.6.8 Creación de la aplicación final

Ya tenemos todas las escenas definidas y la posibilidad de cambiar de escena al atravesar puertas. Solo nos falta juntar todo esto en un archivo ejecutable en el sistema operativo que

deseemos. Esto es lo que Unity llama la **build** de la aplicación. En el menú *File* tenemos el apartado de *build settings* y el de *build and run*. Para configurar y construir nuestra aplicación.

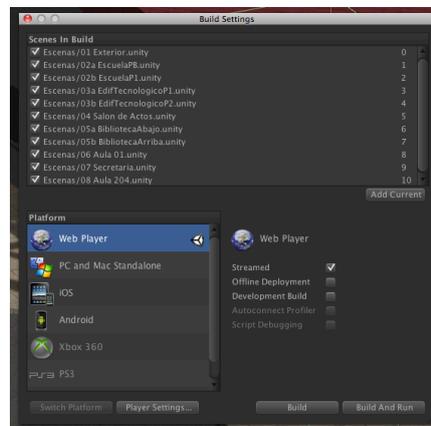


Figura 2.6-48: Build settings

En la ventana de *build settings* la parte principal llamada *scenes in build*, es la que nos muestra las escenas que vamos a incluir en la aplicación. En nuestro caso incluiremos las escenas en el orden lógico de forma que la primera sea la escena 01 Exterior. De esta forma al arrancar el ejecutable será la primera en cargarse.

A continuación tenemos la configuración de salida del ejecutable. Aquí en primer lugar elegimos la plataforma o sistema operativo en la que queremos que corra nuestra aplicación y en cada una de las opciones tendremos distintas posibilidades de configuración.

Unity nos permite realizar ejecutables para *Windows*, *MacOSX*, *iOS* de *iphone*, *Android*, *xBox*, *Wii*, *PSP* y *Unity web player*. Esto tiene ciertas restricciones legales: no se pueden usar los ejecutables de consolas y de *iPhone* si no tienes licencias específicas cedidas por los fabricantes de estas plataformas. De todas formas lo que nos interesa en este trabajo es crear ejecutables para ordenadores convencionales y crear esto mismo para su funcionamiento desde una página web.

Para la creación en modo *Standalone*, esto es, para que funcionen directamente en *Windows* o *MacOSX* simplemente hay que seleccionar el sistema operativo que queramos y construir pulsando *build*. Para la construcción de un ejecutable que se reproduzca desde una página web usaremos la opción *web player*. Aquí tenemos la posibilidad de activar la opción *streamed* y de esta forma en lugar de cargarse todas las escenas al inicio se irán cargando una a una reduciendo el tiempo de espera en la escena inicial. Para reproducir la aplicación necesitamos instalar un pequeño plug-in en nuestro navegador llamado *Unity web player*, este plugin es

gratuito y si no se encuentra instalado en el ordenador que quiera acceder al paseo desde la web, la misma página de *Unity3d* va indicando paso a paso el proceso de instalación de forma sencilla. Es similar al procedimiento para reproducir archivos flash.

En la creación del ejecutable para web se han reducido las resoluciones y la calidad de renderizado para que el paseo sea lo más fluido posible.

3 CONCLUSIONES

3 CONCLUSIONES

Una vez finalizado nuestro trabajo y realizado el test de la aplicación, nos sentimos muy satisfechos del resultado obtenido. En nuestra opinión se ha cumplido la totalidad de los objetivos principales del proyecto.

Por un lado se ha reorganizado todo el material existente referente a modelados de la Escuela y el Edificio tecnológico, cosa que espero que facilite la realización de nuevos trabajos o ampliación de los existentes en un futuro.

Por otra parte se ha conseguido englobar los dos edificios en un mismo paseo virtual consiguiendo un realismo muy aceptable y desde luego mucho mejor logrado que si se hubiera utilizado una plataforma específica para mundos virtuales como *Second Life* u *OpenSim*. En el **Anexo D** se han incluido una serie de imágenes comparativas mediante las que se pueden observar los resultados obtenidos en *Unity3d* y en la plataforma *RealXtend*.

Ha quedado abierta la posibilidad de crear este mundo de forma más interactiva añadiendo enlaces a documentos alojados en la web en la secretaría, la conserjería o la biblioteca, también sería posible la proyección de vídeos o pases de diapositivas en la pantalla del salón de actos y en las aulas. Existiría también la opción de crear una sesión multijugador en la que varios visitantes accedieran al mundo a la vez de forma que se pudieran comunicar entre ellos. Esto facilitaría la creación no solo de un aula virtual sino de la Escuela completa interactiva.

Al reproducir el resultado en la web se ha notado que la velocidad de transmisión de datos en Internet y las tarjetas gráficas de muchos ordenadores comunes aún obstaculizan la fluidez del paseo. Esto es algo que no nos preocupa demasiado ya que tal como se presenta el futuro, muy pronto las velocidades de los proveedores de internet permitirán este tipo de aplicaciones y los ordenadores básicos ya traerán incluidos recursos suficientes como para hacer rentable el trabajo realizado hasta ahora y crear la Escuela de Ingenierías totalmente virtualizada.

4 BIBLIOGRAFÍA

4 BIBLIOGRAFÍA

4.1 Libros:

ALESSI, Jeremy. iPhone 3D Game Programming All in One. Boston, Course Technology, a part of Cengage Learning, 2011.

CREIGHTON, Ryan. Unity 3D Game Development by Example. Birmingham, Packt Publishing Ltd, 2010.

CROS, J. y MOLERO, J). AutoCAD 2007 Práctico. Barcelona, Inforbook's S.L. 2007.

GOLDSTONE, Will. Unity Game Development Essentials. Birmingham, Packt Publishing Ltd, 2009.

MENARD, Michelle. Game Development with Unity. Boston, Course Technology CENGAGE Learning. ISBN-13: 978-1-4354-5658-7HENSON, 2011

4.2 Proyectos

ALLER RODRÍGUEZ, Beatriz. Entorno Virtual de enseñanza con RealXtend. León, Escuela Universitaria de Ingenierías Industrial, Informática y Aeronáutica, 2009.

DÍAZ PRIETO, Antonio. ENTORNO VIRTUAL DE GESTIÓN EN REALXTEND. León, Escuela Universitaria de Ingenierías Industrial, Informática y Aeronáutica, 2009.

FRESNADILLO CANTORAL, Francisco. CAD en Mundos Virtuales: Construcción de la EIII en RealXtend. León, Escuela Universitaria de Ingenierías Industrial, Informática y Aeronáutica, 2009.

MONTERO BENAVIDES, José Luis y MORALES GONZÁLEZ, Jorge. CAD en Mundos Virtuales: Construcción del Edificio Tecnológico de la EIII en RealXtend. León, Escuela Universitaria de Ingenierías Industrial, Informática y Aeronáutica, 2009.

4.3 Páginas web

<http://unity3d.com/unity>

<http://exploradorvirtual.blogspot.com>

<http://www.hispagrid.com>

<http://www.realxtend.com>

<http://www.secondlife.com>

<http://www.unityspain.com>

ANEXO A

Base de Datos.

Archivos *AutoCAD*

BASE DE DATOS. Archivos AutoCAD

A.1 Exterior

<u>OBJETO</u>	<u>ARCHIVO</u>	<u>CAPAS</u>	<u>DESCRIPCIÓN</u>	<u>MATERIAL</u>
Terreno	Campus-3DRefET	AceraCrema	Acera de baldosines	AceraCrema
		Aceras 45	Acera alrededor de escuela	Acera 45
		Aceras	Acera alrededor de edificio tecnológico	Suelo ext.
		Asfalto	Carretera y parking Industriales	Asfalto
		BibliotEmpres	Añadido a Empresariales	PlásticoRojoClaro
		00 Ayuda	Sombra de edificios	_____
		Bordillos	Bordillos	Bordillo
		Cemento	Parquing Empresariales	Cemento
		Cesped	Cesped	Cesped
		Cafetería	Cafetería	Cafetería
		ColumnasEmp	Column bajo Biblioteca Empresariales	MuroGrisLiso
		CristalBibl	Cristal Biblioteca Empresariales	ExtVentEmpres
		Empresariales	Empresariales	Empresariales
		Magisterio	Facultad de Educación	Magisterio
		RadioUniv	Radio Universitaria	RadioUniv
		ParedInv	Pared invisible alrededor del terreno	_____
		Parking	Lineas blancas en parking	Parking
Coches	Audi	Carroceria	Carroceria	Metal aleatorio
		CarrocNeg	Carroceria negra	Metal negro
		CarrocSuperior	Capota	Metal aleatorio
		CristalFaros	Focos blancos	CristalOpac
		Faros	Focos delanteros	Luces

		Vidrios	Ventanas	CristalCoches
		FocosTras	Focos traseros	Luz roja
		Gomas	Plástico	PlasticoGrClaro
		InteriorAsientos	Interior y asientos	PlasticoGrClaro
		InteriorPINegro	Plástico negro interior	PlasticoNegro
		Llantas	Llantas	MetalGrisClaro
		Ruedas	Ruedas	PlasticoNegro
Coche3		Bajos	Bajos	MetalNegro
		Carroceria	Carrocería	MetalAleatorio
		Cristales	Cristales	CristalCoche
		FocoAmarillo	Intermitentes	PlasticoNaranja
		FocoBlanco	Focos blancos	Luces
		FocoRojo	Focos rojos	LuzRoja
		MetalPlat	Metal plateado	MetalAleatorio
		PlasticoNegro	Interior del coche	PlasticoNegro
		Ruedas	Ruedas	PlasticoNegro
Focus		Carrocería	Carrocería	MetalAleatorio
		Cristal	Ventanas	CristalCoches
		CristalFocos	Cristal de focos delanteros	CristalOpac
		FocoRojo	Focos rojos	LuzRoja
		FocosDel	Focos delanteros	Luces
		Ford	Logotipo	MetalAzul
		Interior	Interior del coche	PlasticoNegro
		Intermit	Intermitentes	PlasticoNaranja
		Llantas	Llantas	MetalAleatorio
		Matricula	Matricula	MetalGrisClaro
		PINegro	Plástico negro	PlasticoNegro
		RetrovisyAdornos	Retrovisores y adornos	PlasticoNegro
		Ruedas	Ruedas	PlasticoNegro
Farolas	FarolaGR	Bombilla	Bombilla	MetalBlanco
		Metal	Cuerpo de farola	MetalVerdeFar
	Farola PQ	Bombilla	Bombilla	MetalBlanco
		Metal	Cuerpo de farola	MetalRojoFar

A.2 Escuela de Ingenierías

<u>OBJETO</u>	<u>ARCHIVO</u>	<u>CAPAS</u>	<u>DESCRIPCIÓN</u>	<u>MATERIAL</u>
ESC PB	Escuela EstrPB			
		01 BaldosinBlanco	Bald Blanco en Hall	SueloBlancoEsc
		01 BaldosinNegro	Baldosin Negro en Hall	SueloNegroEsc
		01 Base	Suelo entradas y punto de referencia	SueloGrisEntrada
		01 Escaleras	Marmol Escaleras	MarmolEscalera
		02		
		ColumnaCentral	Columna central	ParedBlancaOsc
		02 DivisionesInt	Divisiones Interiores - Pintura	ParedBlanca
		02 MarmolPared	Placa de marmol en paredes de hall	MarmolParedEsc
		02		
		AdornoMarmolPared	Adorno en marmol de paredes en Hall	MarmolNegro
		03 TechoEntradas	Techo sobre entradas en el exterior	Escayola
		03 TechoPB	Techo sobre Hall	Escayola
		05 Cubrescaleras	Base de escalera	ParedBlanca
		05 RodapieEscal	Rodapié sobre escalera	MarmolEscalera
		06 BarandillasExt	Barandilla en la entrada	BarandillaExt
		07 Arcos	Arcos de ladrillo en muros	LadrilloEscVert
		07 CornisaExt	Cornisa en muro exterior	MuroGrisLiso
		07 CornisaInt	Cornisa alrededor del techo de 1ª pl	MuroGrisLiso
		07 MuroExtGris	Muro de cemento Inf en exterior	MuroGris2
		07 MuroExtLadri	Muro de ladrillo de plata baja	LadrilloEscuela
		08 Ventanas carp	Marcos de ventanas exteriores	MetalMarcos
		08 VentanasCrist	Cristal de ventanas Planta baja	CristalOpac
		09 Saliente secr	Barra al lado de Secretaría	SueloBlancoEsc
		09 SalienteTapa	Encimera de barra de secretaría	SueloNegroEsc
		10 Radiadores	Radiadores Planta baja y escaleras	RadiadorEsc
		11 PuertasMetal	Puertas metalicas en entrada	MetalNegroR
	Esc Acces PB	BotonAsc	Botón ascensor	MetalPlata
		BotonAsc2	Borde de boton	MetalGrisClaro
		Corcho	Corcho	Corcho
		FluorescTubos	Fluorescentes	MetalGrisClaro
		FocosBombilla	Bombillas de halogeno	Luces

	FocosMetal	Aro halógeno	MetalPlata
	Kiosco	Plastico azul en kiosco	MetalAzulGris
	KioscoGris	Plastico gris en kiosco	PlasticoGris
	KioscoNaranja	Metal naranja en kiosco	PlasticoNaranja
	KioscoNegro	Metal negro en kiosco	PlasticoNegro
	MarcoCorPINegro	Marco de plastico negro en corchos	PlasticoNegro
	MarcoCorMetal	Marco metalico en corchos	MetalPlata
	MetalAscensor	Puertas metalicas de ascensor	MetalAscEsc
	Moqueta	Felpudo de entrada	Felpudo
	Maceta	Maceta	PlasticoBlanco
BancoEsc	BancoBajo	Metal blanco debajo del banco	MallaBancoBlanca
	BancoMetBlanco	Patas blancas del baco	MetalBlanco
	BancoMetNegro	Metal negro del banco	MetalNegro
	BancoSillon	Rejilla negra metalica	MallaBancoNegra
ScreyCons	CopyPrintFax	Bandeja copiadora	PlasticoGris
	MadArmSecret	Madera de armario 1	MaderaRoja
	MadSecret	Madera de armario 2	MaderaOsc
	MetalSecret	Metal en secretaría	MetalGrisClaro
	NegArm1	Madera negra de armario 1	MaderaNegra
	PerchBola	Bolas de perchero	MetalPlata
	PerchPalo	Cuerpo de perchero	MetalNegro
	PlasticoNegro	Plástico negro en conserjería	PlasticoNegro
	Negro	Mad negra armarios de secretaría	MaderaNegra
	Persianas	Persianas de secretaría	PlasticoPersianas
	Plastico	Plastico en secretaría	PlasticoNegro
	PlasticoBlanco	Plastico blanco en secretaría	PlasticoBlanco
	PomArm1	Pomos de armario	MetalNegro
	SueloSecret	Suelo de conserjería	SueloSecret
	Tela	Tela de sillas	TelaRoja
	PcCpu	CPUs	PlasticoGris
	PcPantalla	Pantalla de PCs	PlasticoNegro
	PcTeclado	Teclado	PlasticoNegro
Maquinas	CristalMaquina	Cristal máquina 1	CristalMaquina
	MaderaGris	Madera gris	MaderaGrisMaq
	MaderaMarr	Madera marrón	MaderaMaq

	MaquinaA1	Máquina 1	Maq1EscPB
	MaquinaB1	Maquina 2	Maq2Esc
	Maquina3	Maquina 3	Maquina3
MesaMaq	Madera	Madera	MaderaAtril
	Metalnegro	Base de mesa	MetalNegroR
PapMaq	Escuela-PapMaq	Papelera al lado de máquinas	PlasticoGrisClaro
PuertaEntr	CristalPEnt	Cristal puertas	CristalPuertas
	PuertasMetal	Marcos de metal	MetalMarcos
Vitrina	Cristal	Cristal	Cristal
	Madera	Madera	MaderaVitrina
	Pomos	Pomos	MetalDorado
VentanasInt	CristalVentInt	Cristal de ventanas interiores	CristalOpac
	Madera	Madera de ventanas interiores	MaderaPuertas
	VentConsCristal	Cristal de ventana de conserjería	CristalOpac
	VentConsMadera	Madera de ventana de conserjería	MaderaPuertas
Extintor	Base	Base de extintor	PlasticoNegro
	Cuerpo	Cuerpo de extintor	MetalRojoExt
	Mango	Mango	MetalNegroR
	Oro	Metal dorado de extintor	MetalDorado2
	Tubo	Tubo de extintor	PlasticoNegro
Manguera	Frente	Frontal de manguera	Manguera
	Metal	Caja metálica	MetalRojoExt
	Tubo	Tubo metálico exterior	MetalRojo
PanelBlanco	MarcoCorchoPINE		
	PlasticoBlanco	Pies de plástico negro	PlasticoNegro
		Panel blanco	PlasticoBlanco
Pantalla1	MarcoPantalla	Marco de la pantalla	MetalGrisClaro
	Pantalla	Pantalla	PlasticoNegro

Pantalla2	MarcoPantalla	Marco de la pantalla	MetalGrisClaro
	Pantalla	Pantalla	PlasticoNegro
Papelera	Oro	Metal dorado	MetalDorado2
	PapelerasPINegro	Plástico negro	PlasticoNegro
Puerta Aulas	Madera	Madera estática	MaderaPuertas
	MaderaHoja	Madera de la hoja movil	MaderaPuertas
	OroBisagras	Bisagras	MetalDorado
	OroHoja	Metal dorado en hoja movil	MetalDorado2
	OroPuerta	Metal dorado estática	MetalDorado2
	VentanaPuerta	Cristal estático	CristalPuertas
	VentanaPHoja	Ventana en la puerta movil	CristalPuertas
Puertas2	Madera	Madera estática	MaderaPuertas
	MaderaHoja	Madera de la hoja movil	MaderaPuertas
	HojaOro	Metal dorado en hoja movil	MetalDorado2
	Oro	Metal dorado estática	MetalDorado2
	CristalPuerta	Cristal estático	CristalPuertas
	HojaCristal	Ventana en la puerta movil	CristalPuertas
PuertSecret	Madera	Madera estática	MaderaPuertas
	MaderaHoja	Madera de la hoja movil	MaderaPuertas
	OroBisagras	Bisagras	MetalDorado
	OroHoja	Metal dorado en hoja movil	MetalDorado2
	OroPuerta	Metal dorado estática	MetalDorado2
PuertaPeq	Madera	Madera estática	MaderaPuertas
	MaderaHoja	Madera de la hoja movil	MaderaPuertas
	OroPomo	Metal dorado en hoja movil	MetalDorado2
	OroPuerta	Metal dorado estática	MetalDorado2
PuertExt-U	CristalPEntrada	Cristal puerta de entrada	CristalPuertas
	PuertasMetal	Marco de metal	MetalNegroRI
Revistero	Cuerpo	Cuerpo revistero	PlasticoPias
	Metal	Barras metálicas	MetalPlata

Aulas				
	Aula01	PlasticoBlanco	Plástico blanco en aulas	PlasticoBlanco
		Cristal	Cristal de luces de emergencia	CristalOpac
		EnceradoBlanco	Encerado blanco	PlasticoBlanco
		EnceradoMetGris	Marco metálico de encerado	MetalPlata
		Fluorescente	Fluorescente luz de emergencia	Luces
		Led	Led naranja en luces de emergencia	FluorescNaranja
		Marmol	Baldosín de ventanas	BaldosinVentana
		MesaPr	Madera de mesa del profesor	MaderaClara
		Oro	Base dorada de columnas	MetalDorado2
		PlasticoGris	Caja de interruptores	PlasticoPias
		PlasticoNegro	Plástico negro en aula	PlasticoNegro
		ProtectAguj	Plástico en mesa del profesor	PlasticoNegro
		RadiadorGr	Radiador grande	MetalBlanco
		RadiadorPeq	Radiador pequeño	MetalGrisClaro
		RecubCables	Canaleta de cables	PlasticoMarron
		RodapieGranito	Rodapie	SueloNegroEsc
		TarimaOro	Bordes dorados en tarima	MetalDorado2
		Techo	Techo	Escayola
		TechoFluores	Fluorescentes	Luces
		Tela	Tapizado silla	TelaRoja
		Tizas	Tizas	Blanco
		Trapo	Borrador	TelaGris
		EnceradoVerde	Encerado verde	MetalVerdeFar
		MetalNegro	Visagras y manillas ventanas	MetalNegroR
		ParedAzulejos	Azulejos en pared	BaldosinAulas
		ParedMadera	Listón de madera en la pared	MaderaClara
		Persianas	Persianas	PlasticoPersianas
		SueloBlanco	Suelo blanco	SueloBlancoEsc
		SueloGris	Suelo negro	SueloNegroEsc
		Tarima	Tarima	SueloBibl
		TechoCuadrados	Placas techo	TechoCuad
		Columnas	Columnas	ParedBIColumn
	Mesa	Madera	Madera	MaderaMesaBibl
		MetMorado	Metal morado	MetalMorado
	SillaAula	MetMoradoSilla	Base metalica	MetalMorado

	MetMoradoSilla1	Base metálica en sillas	MetalMorado
	PlasticoNegro	Plástico negro	PlasticoNegro
	Sillas	Plástico blanco	PlasticoBlanco
Percha	Percha	Percha metálica	MetalMorado
Ventana	CristalVentana	Cristal de ventana	CristalOpac
	MarcoVentana	Marco de ventana	MetalMarcos
SalActs	Estructura		
	01 Escaleras	Escalones grandes base de butacas	SueloVerdeSA
	01 Rampa	Rampas y base de tarima	AlfombraVerdeSA
	01 RampaAzul	Rampa azul	AlfombraAzulSA
	02 DivisInt	Division con el resto de escuela	ParedBlanca
	ParedMaderaHall	Pared madera en hall	MaderaParSAExt
	ParedMadera	Recubr de madera en paredes	MaderaParedSA
	Techo	Techo en SalActs y hall de salón	Escayola
	Columnas	Columnas en salón y entrada	ParedBIColumn
	RadiadorPeq	Radiadores pequeños bajo ventanas	PlasticoGrisClaro
	PlasticoBlanco	Grifo de radiadores	PlasticoBlanco
	PlasticoNegro	Bordes de escalera en plástico negro	PlasticoNegro
	Oro	Bordes de alfombra metálicos	MetalDorado2
	BaldosinVentana	Baldosín de ventanas	BaldosinVentana
	08 VentanasCarp	Marco de ventanas	MetalMarcos
	08 VentanasCrist	Cristal de ventanas	CristalOpac
Accesorios	Bombilla	Bombilla halógenos	Luces
	Foco	Aro halógenos	MetalPlata
	RadiadorGr	Radiador grande	RadiadorGrSA
	Cristal	Cristal de armario negro	CristalOpac
	Maceta	Maceta	PlasticoBlanco
	MaderaCj	Madera de cajón	MaderaOsc
	MaderaMesas	Madera de mesas de tarima	MaderaMesaSA
	MaderaNegra	Madera armario negro	MaderaNegra
	PlastBlReflect	Plástico blanco de reflectantes	PlasticoGris
	Pomos	Pomos de mesa	MetalPlata

	Reflectantes	Reflectantes	ReflectRojo
	BasePC	Base pantalla	PlasticoGris
	MetalNegro	Micrófonos	MetalNegro
	MetalPlata	Pomo de armario	MetalPlata
	PantallaPC	Pantalla	MetalPlata
	PlasticoNegro	Altavoces, ratón y teclado	PlasticoNegro
	CueroSilla	Asientos de cuero	PlasticoNegro
	MetalSilla	Estructura metálica sillas	MetalNegroR
	Persianas	Cortinas	CortinasSA
	CueroSilla2	Tapizado silla roja	TelaRoja
	Butacas		
	TelaNaranja	Tapizado butacas	TelaNaranja
	MetalNegro	Estructura metálica butacas	MetalNegroR
Esc P1	Escuela EstrP1		
	01 BaseEscInt	Base circular alrededor de escalera	ParedBlanca
	01 BaseEscSup	Suelo alrededor de barandilla	SueloNegroEsc
	01 SueloBlanco	Suelo P1, cuadrados	SueloBlancoEsc
	01 SueloNegro	Suelo el P1 además de los cuadrados	SueloNegroEsc
	02 DivisionIntP1	Divisiones interiores entre aulas	ParedBlanca
	02 MuroInterior	Yeso en muros que dan al exterior	ParedBlanca
	03 TechoCupula	Techo bajo cubierta circular	Escayola
	03 TechoP1	Techo en zonas comunes en P1	Escayola
	03 TechoWC	Techo en baños P1	Escayola
	04 ArandelasCirc	Base de columnas metálicas	MetalDorado2
	04 ArandelasPar	Base de columnas	MetalDorado2
	04 ColParJuntas	Junta de madera en columnas	MaderaPuertas
	04 ColumnAulas	Columnas en aulas 1ª plata	ParedBlancaFino
	04 ColumnasCirc	Columnas metálicas negras pB y 1ª	ColumnaNegra
	04 ColumPared	Columnas cercanas a pared, PB y P1	ParedBlancaFino
	05 CubiertaP1	Cubiertasobre planta 1ª	Cubierta
	05 CubiertaP2	Cubierta circular central	Cubierta
	06 CanalCentral	Canal metálico central	RejillaCanal
	06 Cerchas	Cerchas	MetalNegroR
	06 CierreCanal	Parte superior metálica de canal	RejillaCanal
	07 Cornisa1	Cornisa circular en techo planta 1ª	MuroGrisLiso

07 Cornisa2	Cornisa circular alred de cubierta	MuroGrisLiso
07 CornisaBibl	Cornisa sobre muro gris en bibl	MuroGrisLiso
07 CubiertillaBibl	Cubierta sobre muro gris PB en bibl.	MuroGrisLiso
07 MuroExtLadr	Muro exterior en ladrillo	LadrilloEscuela
07 MuroP2Centro	Muro circular hasta cubierta circular	MuroGris
08 CarpP2	Carpintería ventanas del circulo central	MetalMarcos
08 CarpventBibl	Marcos de ventanas en biblioteca	MetalMarcos
08 Cristal	Cristal de ventanas exteriore	CristalOpac
08 CristalVentBibl	Cristales en ventanas de bibloteca	CristalOpac
08 MarcoVentP1	Marcos de ventanas exteriores	MetalMarcos
08 MarmolVent	Marmol en ventanas	BaldosinVentana
08 MarmVentBibl	Marmol en ventanas biblioteca	BaldosinVentana
09 Rodapie	Rodapie en escaleras,P1 y PB	MarmolEscalera
10 RadiadoresP1	Radiadores en planta 1ª	MetalGrisRad
11 Barandilla	Barandilla circular en planta 1ª y esc	MetalNegro

AccesoriosP1

BotonAsc2	Boton del ascensor	MetalGrisClaro
Fluorescentes	Fluorescentes	MetalGrisClaro
HalogBombilla	Bombilla de Halógenos	Luces
HalogFoco	Aro de Halógenos	MetalPlata
Macetas	Macetas	PlasticoBlanco
Corcho	Corcho	Corcho
MarcoCorPINegro	Marco negro en corchos	Plasticonegro
MarmolPared	Recubrimiento de marmol en paredes	MarmolParedEsc
TiraMarmolPared	Tira negra en marmol de pared	MarmolNegro

Biblioteca

Estructura

Barandilla	Barandilla	MetalBlanco
BaseBarandillaP2B	Base de barandilla en planta2ª	ParedBlancaFino
ColumnaCentralB	Columna central de biblioteca	ParedBlanca
Columns	Columns	ParedBlColumn
CubreEscalera	Recubrimiento de escalones	ParedBlanca
DivisIntBiblMarr	Pared marrón en biblioteca	ParedMarronBibl
Escalones	Escalones	SueloBibl
Rodapie	Rodapie	SueloLisoBibl
SueloBibl	Suelo de biblioteca planta 1	SueloBibl
SueloP2	Suelo de biblioteca planta2	SueloBibl

	TechoP1B	Techo de biblioteca planta 1	Escayola
	ParedMadera	Pared de madera	MaderaParedBibl
Accesorios	BuzonN	Metal negro en buzón	Plasticonegro
	BuzonR	Metal rojo en buzón	PlasticoRojo
	Clasificador	Madera de clasificador	MaderaClara
	ClasificPuertas	Puertas de cajones del clasificador	MaderaAtril
	ClasifOro	Metal dorado en clasificador	MetalDorado2
	EstantBlanco	Metal blanco de estanterías	MetalBlanco
			MaderaEstantgran
	EstantGrande	Madera de estanterías	leB
	HalBombilla	Bombilla de halógenos	Luces
	HalFoco	Aros de halógenos	MetalPlata
	LibroBlanco	Libros blancos	Blanco
	LibroVerde	Libros verdes	LibroVerde
	LibroRojo	Libros rojos	LibroRojo
	MesaEntr	Mesa de la entrada	MaderaClara
RadiadorGr	Radiador grande	RadiadorGrBibl	
Detector	Alfombra	Alfombrilla	AlfombraBiblEntr
	MadClara	Madera clara	MaderaDetector
	MadNegra	Tacos de madera negra	MaderaNegra
	MadOsc	Madera oscura en bordes	MaderaOsc
			PlasticoGrisOscBib
	PlasticoGris	Plastico gris detector	
	PlasticoGrisClaro	Plastico gris claro detector	PlasticoGrisClBibl
SillaTela	Cuero	Tapizado de tela	TelaRoja
	Metal	Estructura metálica	MetalNegroR
	PlasticoNegro	Respaldo y base de plástico	PlasticoNegro
Sillamesa	Mesa	Madera mesa	MaderaMesaBibl
	MesaBorde	borde de mesa	MaderaOsc
	MesaPatatas	patas	MetalNegroR
	SillaCuerpo	cuerpo	MaderaClara
	SillaPatatas	patas	MetalBlanco
Mesas	Mesa	Madera mesa	MaderaMesaBibl

		MesaBorde	borde de mesa	MaderaOsc
		MesaPatatas	patas	MetalNegroR
		MesasGoma	Pata de goma en mesas	PlasticoNegro
		SillaCuerpo	cuerpo	MaderaClara
A. 204	Aula 204	Arm1Cristal	Cristal de armario1	Cristalopac
		Arm1Mad	Madra d armario 1	MaderaAtril
		Arm1Pomo	Pomos de armario 1	MetalNegroR
		Arm1Negro	Madera negra en armario 1	MaderaNegra
		Arm2Mad	Madera de armario2	MaderaClara
		Arm2negro	Madera negra de armario2	MaderaNegra
		Arm3Negro	Madera negra de armario 3	MaderaNegra
		Atril	Atril	MaderaAtril
		BiomboMetal	Estructura metálica de biombo	MetalNegroR
		BiomboCristal	Cristal de biombo	CristalCoches
		BiomboMoq	Tela de biombo	TelaBiombo
		BiomboPlaNeg	Plastico negro en biombos	PlasticoNegro
		Cablesrecub	Canaleta de cables en tarima	PlasticoMarron
		ColumnaOro	Base dorada de columna	MetalDorado2
		ColumnaYeso	Columna	ParedBIColumn
		CuadroLuces	Cuadro de luces	PlasticoPias
		EnceradoMetal	Marco de encerado	MetalPlata
		Enceradoverde	Encerado verde	MetalVerdeFar
		FluorescCub	Cubre fluorescentes	FluoresCub
		FocosBombilla	Bombilla de focos	Luces
		FocosMetal	Aro de focos	MetalPlata
		Marco	Marco de ventanas	Metalmarcos
		MeasDibMad	Madera de mesa de dibujo	MaderaMesaBibl
		MesaDibPat	Patatas de mesa de dibujo	MetalBlanco
		MesaDibPla	Plastico en mesas de dibujo	PlasticoNegro
		MesaPeqMadera	Mesas pequeñas madera	MadMesaBiblArr
		MesaPqMadnegra	Madera negra en mesas pequeñas	MaderaNegra
		MesaOrd2Mad	Mesa de ordenador grande	MaderaAtril
		MesaOrd2Pomo	Pomos de mesa de ordenador grande	PlasticoNegro
		MesaOrd2negro	Mad negra en mesa ordenador grande	MaderaNegra
		MesaProfMadera	madera de mesa de profesor	MaderaAtril
		ParedBaja	Pared alicatada	BaldosinAulas

ParedDibujo	Revestimiento de pared	ParedBlanca
ParedMad	Liston de madera en pared	MaderaAtril
ParedSuperior	Pared exterior de hall piso1	_____
PatasSilla	Patas de silla de ordenador pequeño	MetalNegroR
PerchBola	Bola de perchero pie	MetalPlata
PerchMetalNegro	Cuerpo de perchero pie	MetalNegroR
PerchMadera	Madera de percheros de pared	MaderaClara
PerchMetMorado	Percheros metálicos	MetalMorado
Persianas	Persianas venecianas	PlasticoPersianas
Planta 1 A	Pared exterior con huecos de radiador	ParedBlanca
PlasticoNegro	Plastico negro	PlasticoNegro
Proyec	Pantalla de Proyector	PlasticoBlanco
Proyecbarra	Barra de proyector	PlasticoGris
ProyecNegro	Plastico negro en proyector	PlasticoNegro
Radiador	Radiador grande	RadiadorEsc
Rodapie	Rodapie	Suelonegroesc
SillaDibMad	Madera en silla de dibujo	MaderaSillaDibujo
SillaDidMetal	Patas de sills de dibujo	MetalnegroR
SillaDibMetCirc	Reposapies en silla d deibujo	MetalNegroR
SillaDibPla	Plastico de silla de dibujo	PlasticoNegro
SillaOrdCuerpo	Cuerpo de silla de ordenador pequeño	PlasticoMarron
SillaProfPlaNeg	Plastico negro en silla de profesor	PlasticoNegro
SillaProfTela	Tapizado de silla de profesor	TelaRoja
SueloBla	Suelo blanco	SueloBlancoEsc
Sueloneg	Suelo negro	SueloNegroEsc
Tarima	tarima	SueloBibl
TarimaOro	Borde dorado de tarim	MetalDorado2
Techo	Techo	Escayola
TechoCuadrados	Placas cuadradas en techo	TechoCuad
TechoFluoresc	Fluorescentes	Luces
TripArMarc	Marco de madera en ventana triple	MaderaPuert
TripArVent	Cristal de ventana triple	CristalOpac
VentBaldosin	Baldosín de ventanas	BaldosinVent
VentCristal	Cristal de ventanas	CristalOpac

A.3 Edificio Tecnológico

<u>ARCHIVO</u>	<u>CAPAS</u>	<u>DESCRIPCIÓN</u>	<u>MATERIAL</u>
PbyP1	BarandillaEscLat	Barandilla en escalera lateral	barandillaET
	BaseColP12	Base de columnas en piso 1º	barandillaExt
	BaseColPB	Base columnas en Planta baja	barandillaExt
	BaseColPBMarmol	Base de columnas cuadradas	MarmolRojo
	bordechapa	Chapa alrededor de cubierta	PlasticoRojo
	ColumnaExt	Columnas exteriores en laterales	SueloGrisEntrada
	Columnas	Columnas centrales	ParedBlanca
	ColumnasEsc	Columnas en escaleras	ColumnEscET
	Cornisa	Cornisa entre muros de ladrillo	MuroGrisLiso
	Cristal	Cristal en ventanas de cubierta central	CristalOpac
	Cubierta	Cubierta	Cubierta
	CubreEscET	Estructura que sostiene los escalones	ParedBlancaFino
	EscaleraExt	Escaleras y rampas exteriores	SueloGrisEntrada
	Escalones	escalones	MarmolEscaleraET
	Felpudo	Felpudo	Felpudo
	InterseccionEsq	columnas en esquinas	MuroGrisLiso
	Ladrillo	Ladrillo	LadrilloEscuela
	LadrilloIntP1	Ladrillo interior en planta 1	LadrilloEscuela
	LadrilloIntPB	Ladrillo interior en planta baja	LadrilloEscuela
	Ladrojo	Ladrillo Planta 2 (enlace)	ladrilloEscuela
	LadTechoExt	Ladrillo horizontal en bordes de p2	LadrilloEscuela
	Madera	Listones de madera en columnas	MaderaClara
	Murogris	Muro gris exterior	MuroGris2
	Pa-Cerradura	Componentes puerta ascensor	MetalDorado2
	Pa-Cristal	Componentes puerta Tipo A	CristalOpac
	Pa-Pomo	Componentes puerta Tipo A	PlasticoNegro
	Pa-Puerta	Componentes puerta Tipo A	MetalGrisClaro
	Pa-Vent	Componentes puerta Tipo A	CristalOpac
	Paredblanca	Revestimiento blanco en paredes	ParedBlanca
	Pb-Marco	Componentes puerta Tipo B	MaderaPuertas
	Pb-Pomo	Componentes puerta Tipo B	MetalDorado2
	Pb-Puerta	Componentes puerta Tipo B	MaderaPuertas
	RodapieEsc	Rodapie de escalera	MarmolEscalera
	SueloBlancoBy1	Suelo Blanco en planta Baja	SueloBlanco
	SueloBy1	Suelo Rojo en plantas B y 1	SueloRojo

TechoP1	Techo P1	EscayolaET
TechoP2	Techo P2	EscayolaET
TechoPB	Techo PB	EscayolaET
TirasEscalones	Antideslizantes en escalones	TelaNegra
UnionBarandilla	Barandilla en Union de edificios	MarcosUnion
UnionColumnas	Columnas en UE	MetalNegroRallado
UnionCristal	Cristal en UE	CristalOpac
UnionCubierta	Cubierta en UE	CubiertaUnion
UnionMarco	Marcos metálicos en UE	MarcosUnion
UnionSuelo	Suelo de UE	SueloGrisEntrada
VentIntP1Cristal	Cristal de ventanas interiores en p1	CristalOpac
VentIntP1Marco	Marco de ventanas interiores en P1	MetalGris
VentIntPBCristal	Cristal de ventanas interiores en pB	CristalOpac
VentIntPBMarco	Marco de ventanas interiores en PB	MetalGris

PI 2, 3 y 4

AdornoMarmolEsc	Adorno de marmol en escalera	MarmolRojo
bordechapa	Chapa alrededor de cubierta	PlasticoRojo
Cornisa	Cornisa exterior entre ladrillos	MuroGrisLiso
Cubierta	Cubierta	Cubierta
Ladrillo	Muro de ladrillo	LadrilloEscuela
LadrilloIntP23	Muro del patio interior	LadrilloEscuela
Suelo	Suelo blanco alrededor de escalera P4	SueloBlanco
SueloDespachos2	Suelo de despachos y pasillos P2	SueloAzulDespachos
SueloDespachos3	Suelo de despachos y pasillos P3	SueloAzulDespachos
SueloRojo2	Suelo rojo alrededor de escalera P2	SueloRojo
Techo 2y3	Techo P2 y P3	EscayolaET
SueloRojo3	Suelo rojo alrededor de escalera P3	SueloRojo

PI 2 y 3 Int.

Cristal	Cristal de ventanas interiores	CristalOpac
Ladrojo	Ladrillo P2 y P3	LadrilloEscuela
Marco	Marco de ventanas interiores	MetalGris
Paredblanca	Pared Blanca	ParedBlanca
ParedGris	Pared de despachos	PlasticoGrisClaro
ParedMarmolAsc	Pared de marmol de ascensores	MarmolRojo
Tiras	Tiras en pared despachos	MetalBlanco

Barand y Asc

Ascensor	Puerta ascensores	MetasAscET
----------	-------------------	------------

Ascensorb1	Borde del boton	PlasticoNegro
Ascensorb2	Boton ascensor	PlasticoGrisClaro
BarandillaCentroP1	Barandilla centro en P1	BarandillaET
BarandillaEsc	Barandilla escaleras	BarandillaET
CubreEscET	Estructura de escaleras (enlace)	ParedBlancaFino

ANEXO B

Instrumentos utilizados

Instrumentos Utilizados

B.1 Hardware

- *iMac 20" 7.1*
- *iMac 27" 12.2*

B.2 Software

- *AutoCAD 2007*
- *AutoCAD 2010*
- *3ds Max 2009*
- *iPhoto '11*
- *PixPlant 2.0.5*
- *DxO Optics Pro6*
- *Unity3d 3.30f4*
- *Unitron 2.1.5*
- *Bias Peak 6*

B.3 Otros instrumentos

- Medidor Laser *LEICA DisctoD3*
- Cámara de fotos *SONY DSC-W35*
- Grabador de audio digital *TASCAM DR-07*

ANEXO C

Scripts

Scripts

C.1 Cambio de escenas

Script 1 – En controlador de la escena 0

```
static var Puerta=0;
function Update () {
}
```

Script 2 – En Zonas de paso de una escena a otra en la escena de donde se parte

```
function OnTriggerEnter() {
    CambioPuerta.Puerta=4;
    Application.LoadLevel("02a EscuelaPB");
}
```

Script 3 – En Controlador de la escena donde se quiere llegar (en este caso 01 Exterior)

```
print(CambioPuerta.Puerta);
if(CambioPuerta.Puerta==1){
    transform.position=Vector3(-10.5,2,53);
    transform.Rotate(0,180,0);
}
if(CambioPuerta.Puerta==2){
    transform.position=Vector3(-10.5,2,55.4);
    transform.Rotate(0,180,0);
}
if(CambioPuerta.Puerta==3){
    transform.position=Vector3(-32,2,22.5);
```

```

        transform.Rotate(0,225,0);
    }
    if(CambioPuerta.Puerta==4){
        transform.position=Vector3(-42.5,2,22);
        transform.Rotate(0,315,0);
    }
    if(CambioPuerta.Puerta==5){
        transform.position=Vector3(-36,2,-1.95);
        transform.Rotate(0,90,0);
    }
    if(CambioPuerta.Puerta==6){
        transform.position=Vector3(-39.5,2,-1.95);
        transform.Rotate(0,90,0);
    }
}

```

C.2 Apertura de puertas

Colocado en un cubo sin renderizar con “on trigger” activado en el componente collision

```

var AngleY : float = 90.0;

private var targetValue : float = 0.0;
private var currentValue : float = 0.0;
private var easing : float = 0.05;

var puertaAbrir: AudioClip;
var puertaAbrir2: AudioClip;
var Target : GameObject;

function Update(){
    currentValue = currentValue + (targetValue - currentValue) * easing;
    Target.transform.rotation = Quaternion.identity; // set rotation to zero
    Target.transform.Rotate(0, currentValue, 0); // apply full Rotation
}

function OnTriggerEnter (other : Collider) {
    targetValue = AngleY;
    currentValue = 0;
    AudioSource.PlayClipAtPoint(puertaAbrir, transform.position);
}

function OnTriggerExit (other : Collider) {
    currentValue = AngleY;
}

```

```
targetValue = 0.0;  
AudioSource.PlayClipAtPoint (puertaAbrir2, transform.position);  
}
```

ANEXO D

Comparativa

Realidad/Unity/RealXtend

Escuela exterior: Realidad – Unity – RealXtend





Escuela y Edificio Tecnológico: Realidad – Unity



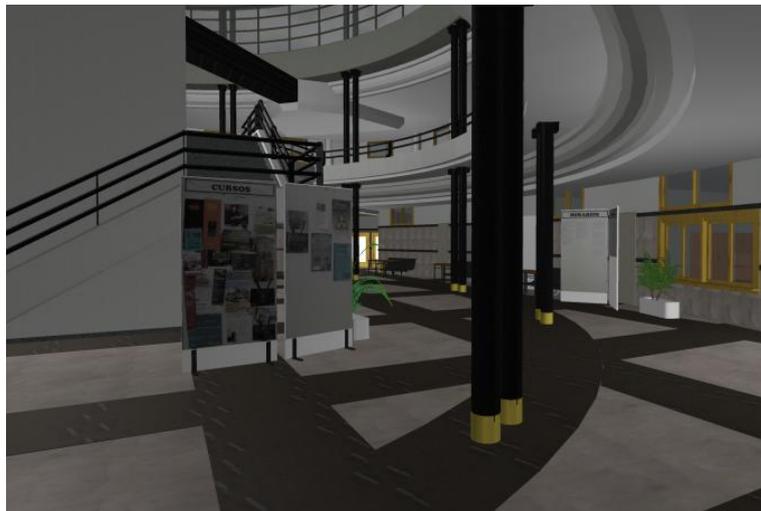
Edificio Tecnológico: Realidad – Unity – RealXtend



Interior de Edificio tecnológico: Realidad – Unity



Interior Escuela de Ingenierías: Realidad - Unity - RealXtend



Aula 4: Realidad - Unity - RealXtend



Aula 204: Realidad - Unity - RealXtend





Biblioteca: Realidad - Unity



Salón de Actos: Realidad – Unity





