



**Benemérita  
Universidad Autónoma de Puebla**

**Facultad de Ciencias de la Computación**

**“La Facultad Virtual”**

*Tesis Profesional*

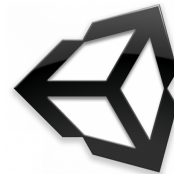
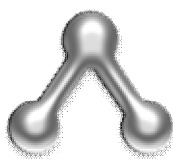
**Para obtener el Título de:  
Ingeniero en Ciencias de la Computación**

Presenta:

**Abraham Sánchez Flores**

Asesor:

**Abraham Sánchez López**



Puebla, Pue.

Julio 2013



# Índice General

<b>Capítulo 1 Modelización y ambiente Virtuales</b>	
1.1 Realidad Virtual.....	6
1.2 Arquitectura en realidad virtual .....	7
1.3 Inmersión, presencia y realismo .....	10
1.3.1 ¿Qué es la presencia?.....	11
1.3.2 La inmersión .....	11
1.4 Realismo en entornos virtuales y arquitectónicos .....	13
1.5 Metodología usada en el desarrollo del entorno virtual arquitectónico .....	14
Conclusiones del Capítulo.....	16
<b>Capítulo 2 Modelado de los edificios de la Facultad de Ciencias de la Computación</b>	
2.1 AC3D.....	17
2.2 Vistas Ortográficas (2D) .....	18
2.3 Usando Imágenes de Fondo.....	18
2.4 Modelado de un Edificio utilizando la rotoscopia .....	19
2.5 Texturizado de Objetos 3D.....	23
2.5.1 Mapeo UV.....	23
2.5.2 AC3D Texture Coordinate Editor.....	24
2.5.3 Texturizando nuestro edificio.....	27
Conclusiones del Capítulo .....	30
<b>Capítulo 3 Adaptación de un archivoBVH a un personaje</b>	
Conclusiones del capítulo. ....	43
<b>Capítulo 4 Resultados</b>	
4.1 Conformación de la Escena.....	44
4.1.1 SkyBox .....	44
4.1.2 Uso de Terrenos .....	45
4.1.3 Integración de los Edificios de la Facultad.....	46
4.1.4 Sistema de Locomoción para Unity.....	47
4.1.4.1 Configuración básica de la escena.....	47
4.1.4.2 Configuración del sistema de locomoción.....	48
4.1.5 Luces en nuestra Escena .....	52
4.1.5.1 Luces direccionales.....	53
4.1.6 Uso de WoW Character Movement Camera.....	54
4.2 Resultados.....	55
Conclusiones del Capítulo .....	58
<b>Capítulo 5 Conclusiones y trabajo futuro</b>	
5.1 Conclusiones .....	59
5.2 Trabajo Futuro.....	59
BIBLIOGRAFÍA.....	60

## Indice de Figuras

Figura 1.1: Ivan Sutherland Espada de Democles .....	6
Figura 1.2: Habitación modelada en 3D. ....	8
Figura 1.3: Primer Piso Casa Modelada en 3D. ....	10
Figura 1.4: Inmersión Completa Simulador de Automóvil.....	12
Figura 1.5: Complejo Escolar para Recorrido Virtual. ....	13
Figura 1.6: Etapas de desarrollo orientado a prototipos. ....	15
Figura 2.1: Interfaz AC3D.....	17
Figura 2.2: Rotoscopia en AC3D. ....	18
Figura 2.3: Planos arquitectónicos del edificio 135. ....	19
Figura 2.4 Rotoscopia planos del edificio 135. ....	20
Figura 2.5: Primera pared Edificio 135.....	21
Figura 2.6: Selección de los vértices. ....	21
Figura 2.7: Extrusión de los vértices. ....	22
Figura 2.8: Creación de un primer cuarto.....	22
Figura 2.9: Creación de muro independiente.....	23
Figura 2.10: Ejemplo texturizado UV Map Esférico. ....	24
Figura 2.11: AC3D Texture Coordinate Editor.....	24
Figura 2.12: Selección primera cara. ....	25
Figura 2.13: Remap primera cara .....	26
Figura 2.14: Mapeo UV cubo.....	26
Figura 2.15: Mapeo UV edificio 135.....	27
Figura 2.16: Texturizado en photoshop del edificio 135.....	28
Figura 2.17: Modelo Edificio 136. ....	28
Figura 2.18: Modelo Edificio 135. ....	29
Figura 2.19: Modelo Edificio 104C.....	29
Figura 3.1: Menú Importar archivo .bvh Softimage.....	31
Figura 3.2: Selección del archivo .bvh.....	32
Figura 3.3: Pose T archivo .bvh.....	32
Figura 3.4: Menú Importar archivo .obj.....	33
Figura 3.5: Ajuste longitud de pies. ....	34
Figura 3.6: Ajuste longitud piernas.....	34
Figura 3.7: Ajuste longitud columna, cuello y cabeza.....	35
Figura 3.8: Ajuste hombro, brazo y mano.....	35
Figura 3.9: Menú Enveloping. ....	36
Figura 3.10: Envelope Weights. ....	37
Figura 3.11: Errores en la influencia del hueso.....	37
Figura 3.12: Corrección mediante menú Reassign Locally. ....	38
Figura 3.13: Resultado corrección de la influencia.....	38
Figura 3.14: Corrección mediante menú Weight Editor.....	39
Figura 3.15: Selección de huesos. ....	40
Figura 3.16: Plot rotaciones de huesos.....	40
Figura 3.17: Parámetros del plot.....	41
Figura 3.18: Curvas de animación de un hueso.....	41
Figura 3.19: Parámetros exportación archivo .fbx.....	42

Figura 4.1: Skybox Unity 3D.....	44
Figura 4.2: Terrain Unity 3D.....	45
Figura 4.3: Parámetros Terrain Unity 3D.....	46
Figura 4.4: Carpeta Assets Objetos .3ds.....	47
Figura 4.5: Character Controller Unity 3D.....	48
Figura 4.6: Leg Controller Unity 3D.....	49
Figura 4.7: Parámetros Leg Controller.....	50
Figura 4.8: Ajuste de ambas piernas.....	51
Figura 4.9: Configuración de animaciones.....	51
Figura 4.10: Luz tipo punto.....	52
Figura 4.11: Luz direccional.....	53
Figura 4.12: Luz direccional al aire libre.....	53
Figura 4.13: WoW Character Movement Camera.....	54
Figura 4.14: Zoom out WoW Camera.....	55
Figura 4.15: Resultado Final I.....	56
Figura 4.16: Resultado Final II.....	56
Figura 4.17: Resultado Final III.....	57
Figura 4.18: Resultado Final IV.....	57

# Capítulo 1

## Modelización y Ambientes Virtuales

### 1.1 Realidad Virtual

Para entender mejor el concepto de realidad virtual, primero se definirán los significados de las palabras “realidad” y “virtual”. El diccionario de la Real Academia Española (<http://www.rae.es>) define virtual como “Que tiene virtud para producir un efecto, aunque no lo produce de presente” o “Que tiene existencia aparente y no real”; y realidad como “Existencia real y efectiva de algo”. También define realidad virtual como “Representación de escenas o imágenes de objetos producidas por un sistema informático, que da la sensación de su existencia real”.

El concepto de Realidad Virtual (RV) surgió en 1965 cuando Ivan Sutherland (hoy miembro de Sun Microsystems Laboratories) publicó un artículo titulado “The Ultimate Display”. También fue Sutherland quien construyó el primer casco de realidad virtual, que en su momento se llamó “Espada de Damocles”, dado el diseño brusco que tenía en ese entonces (<http://telematica.cicese.mx/computo/super/cicese2000/realvirtual/>). Hoy en día no existe una descripción clara y oficial de lo que es RV, lo que ha causado que el término sea usado en muchas otras aplicaciones como el cine, los videojuegos, simuladores y demás. Algunos autores han tratado de describir el término de RV como “Medio ambiente interactivo, tridimensional, generado por computador, en el cual se sumerge a una persona” o “Simulación de un ambiente tridimensional generada por computadoras, en el que el usuario es capaz tanto de ver como de manipular los contenidos de ese ambiente” (Roehl, 1996, citado en Escarpín,2000). Ambas definiciones concuerdan en que realidad virtual es un ambiente tridimensional generado por computadora (hardware y software), que simula objetos o ambientes reales en el cual los usuarios pueden interactuar con ellos. La interacción lograda y la visualización del entorno virtual logran hacer que el usuario experimente diferentes niveles de inmersión.

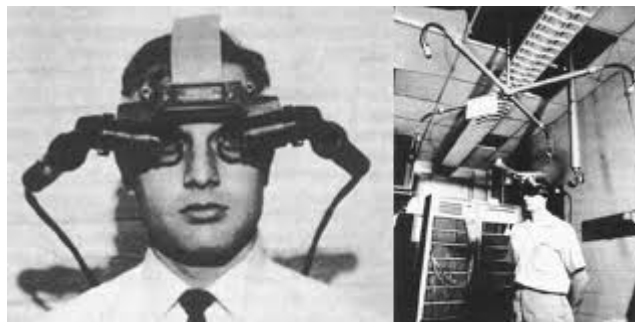


Figura 1.1: Ivan Sutherland Espada de Democles

Aukstakalnis y Blatner (1992) comentan que: “Ha sido demostrado que usar sonido para proporcionar información alternativa o suplementaria a un usuario de computadora puede aumentar de manera considerable la cantidad de información que ellos pueden ingerir”, lo anterior muestra la importancia de incluir sonido 3D en los entornos virtuales para lograr una mayor inmersión en el entorno. Los sonidos deben reaccionar de la misma manera cómo reaccionarían en el mundo real, con efectos de desvanecimiento, eco, perturbación, y todos aquellos factores que intervienen desde el origen del sonido hasta la recepción del mismo por parte del usuario.

Para poder visualizar entornos virtuales e interactuar con los mismos, generalmente se requieren dispositivos de despliegue eficaz, tales como:

Lentes LCD, los cuales utilizan como base al cristal líquido de cuarzo para presentar imágenes de la computadora. Despliegues montados en la cabeza (HMD), los cuales son dispositivos que cubre ambos ojos y calcula en tiempo real los gráficos estereoscópico generados por la computadora huésped, además que proporciona información sobre los movimientos de la cabeza para que la computadora genere perspectivas apropiadas del mundo virtual de acuerdo a la posición del usuario. Así mismo, para obtener la interacción deseada en un entorno virtual se emplean diferentes tipos de dispositivos que permiten a los usuarios sentir ciertos aspectos del mismo, como son guantes, dermatoesqueletos, mayordomos, plataformas simuladoras de movimiento, etc. A estos dispositivos periféricos se les conoce como formas de retroalimentación (Perry, Smith y Yang, 2001).

La realidad virtual ha ido mejorando enormemente, debido a los altos requerimientos que se exigen en la actualidad, tal es el caso de los simuladores vuelo comerciales, los cuales requieren sean más realistas e inmersivos, dado que el entrenamiento de los pilotos es un elemento crucial para la seguridad de las personas y aerolíneas, se requiere que los simuladores de vuelo cuenten con ambientes virtuales precisos, actualizados, y altamente realistas. Tal es el caso del simulador de vuelos EP-1000CT desarrollado por Evans y Sutherland, el cual cuenta con áreas geográficas de alto detalle, una enorme librería 3D de aviones, simulación de humo, nubes, fuego, lluvia, etc. Todos estos factores crean un entorno más realista que capacitará mejor a los pilotos comerciales. (EvansSutherland, 2002).

## 1.2 Arquitectura en realidad virtual

La práctica arquitectónica se incluye como una de las materias más inmersas en este cambio tecnológico. Las primeras experiencias de la arquitectura asistida por computadora se dan en la década de los 80 cuando se aplican los programas CAD (Computer Aided Design). Hacia los últimos años se hace referencia a una arquitectura virtual realizada en un espacio tridimensional, en la cual se concibe una arquitectura aparente pero no real (Foley et al., 1994; Hearn y Baker, 1994).

A las aplicaciones de la arquitectura virtual comúnmente se les conocen como “paseos” o “paseos virtuales”, ya que principalmente se enfocan en la visualización de espacios tridimensionales, además que dichas aplicaciones no incluyen efectos complejos de sonido o tacto. Dichos paseos son de gran ayuda para que los diseñadores, decoradores, ingenieros, clientes y cualquier otra persona involucrada en la obra arquitectónica trabajen conjuntamente en el aspecto final de la arquitectura. Uno de los mayores problemas a los que se enfrentan los arquitectos es la definición de espacios volumétricos en un dibujo 2D, además de que la mayoría de la gente tiene dificultades para leerlo y poderlo asimilar tal como lo haría el arquitecto.

Para compensar esto, los arquitectos incluyen alzadas en las estructuras de los anteproyectos, cuyo propósito es ayudar al cliente a visualizar mejor lo que en el plano se especifica. Aun así, no son suficientes para lograr la comprensión total del plano (Rangel y Santana, 1999).

Mediante el uso de la realidad virtual, el arquitecto puede avanzar enormemente desde el anteproyecto a las alzadas, a través de representaciones en 2D y 3D creando una mejor percepción del plano. A esto se le conoce como construcciones virtuales, en las cuales, los participantes pueden entrar en ellas, incluso antes de que un edificio sea construido, permitiendo, tanto al cliente como al arquitecto, la posibilidad de modificar, corregir y rectificar rápidamente los planos mientras se pasean por las habitaciones en el modelo tridimensional generado por computadora.



Figura 1.2: Habitación modelada en 3D.

La arquitectura apoyada en la realidad virtual ha tenido mucho impacto en todo el mundo, convirtiéndose en una herramienta esencial para los actuales arquitectos, ya que les permite visualizar antes de comenzar su construcción.

En el artículo titulado “Arquitectura virtual” se comenta que la arquitectura virtual muestra nuevos alcances, debido a que apoya la exploración, identificación y análisis de una edificación arquitectónica en forma digital, donde el usuario tiene una inmersión total, con la opción de volar o desplazarse en forma similar a como lo haría en la realidad. También es posible realizar cambios de ambientación y de objeto, lo que permite una mejor apreciación de los espacios arquitectónicos (Casas y Lucent, 2004).

En entrevista realizada, Geneviève define arquitectura virtual como una “Construcción de objetos arquitectónicos tridimensionales utilizando herramientas de cómputo, los cuales pueden ser visualizados y vividos con varios grados de realismo” (Casas y Lucent, 2004). En un artículo (<http://www.blues.uab.es/home/material/apunts/M02020/lectura1.htm>) en Internet se menciona que, gracias al uso de maquetas virtuales, puede percibirse perfectamente la espacialidad, de la cual carecían los anteriores sistemas de representación. (Levis, 2000).

La arquitectura virtual ofrece muchas ventajas, ya no solo se limita a mostrar arquitectura actual, sino que también puede ser usada para mostrar riquezas arquitectónicas históricas, tal es el caso del proyecto Santiago 2000 virtual city, el cual produce un ambiente virtual que recrea la Plaza de Obradoiros Santiago y algunas otras áreas que la rodean. Este proyecto fue visualizando usando el proyecto NAVE (ambientes virtuales inmersivos a bajo costo), el cual demuestra que una computadora personal puede soportar en tiempo real alta calidad, grandes resoluciones e imagen estereoscópica, mediante tarjetas aceleradoras de gráficos (RIVA TNT 2), y componentes comunes en una computadora de escritorio (tarjeta de sonido SBLIVE, Procesadores PIII, etc.), usando para su despliegue proyectores VREX. (Pair et al., 2000).

Otra de las ventajas que ofrece la arquitectura virtual es que permite descubrir problemas potenciales mucho antes de que comience la construcción, ahorrando así tiempo, dinero y esfuerzo.

En una noticia publicada en Internet se dice que: “Gracias al uso de aplicaciones interactivas, el comprador podrá conocer virtualmente su futura vivienda, observando la funcionalidad y la calidad de los espacios diseñados, lo que, sin duda, le ayudará a adquirir seguridad de cara a realizar su compra” (Las Provincias, 2004).

La arquitectura virtual puede ser de gran ayuda y apoyo a los arquitectos, ya que permite visualizar con bastante exactitud la obra final, incluso mucho antes de plasmar la obra en planos. Tal es el caso de un Instituto Tecnológico en Valencia, llamado Instituto Tecnológico de la Construcción (AIDICO), el cual, con el objeto de dar respuesta al creciente interés del sector hacia estas nuevas formas de presentación, cuenta con un departamento que se dedica exclusivamente a la creación de presentaciones virtuales en el sector de la construcción (Las Provincias, 2004).



Figura 1.3: Primer Piso Casa Modelada en 3D.

Otra área en la que se muestra especial interés en la arquitectura virtual, es la reconstrucción de arquitectura prehistórica, tal es el caso de la Universidad del Estado de Georgia, la cual mediante su artículo titulado “Realism vs. Reality: Creating virtual reconstructions of prehistoric architecture” (<http://sipapu.gsu.edu/SAA00/>) muestra la importancia del realismo en la representación de entornos virtuales arquitectónicos prehistóricos.

## 1.3 Inmersión, presencia y realismo

### 1.3.1 ¿Qué es la presencia?

Actualmente, no existe una definición real y mundialmente aceptada para definir la presencia, existe un amplio campo de estudio sobre la presencia, sin embargo, mucha literatura define a la presencia como un efecto de transportación, un sentimiento de estar transportado o presente en un entorno virtual inmersivo, una sensación de “estar allí” (Schuemie et al., 2001). Otros autores la definen como “Un fenómeno subjetivo tal como la sensación de estar en un entorno virtual” (Slater y Wilbur. 1995).

### 1.3.2 La inmersión

La presencia comúnmente se le relaciona con la inmersión, la cual, algunos autores la identifican como la magnitud en la que los sentidos son comprometidos por el ambiente virtual (Lombard, M., y Ditton, T. (2000). Otros autores la definen como un estado psicológico que se caracteriza por la percepción de uno mismo se encuentra interactuando en un entorno virtual” (Witmer y Singer. 1998).

La inmersión y la presencia son dos términos que comúnmente se confunden, ya que no existe una definición clara y mundialmente aprobada para cada una de éstos. Pero ambos se relacionan con la sensación de que el usuario se encuentra presente en el entorno virtual, esto es, que crea lo más posible que lo que está viendo es real, que forma parte del entorno. Ambas sensaciones dependerán de diversos factores como pueden ser la calidad visual del entorno, la consistencia, los dispositivos periféricos usados, la interacción del usuario con el entorno, el tipo de visualización del entorno, y unos de los más importantes, el realismo, la interactividad en el ambiente virtual y la velocidad de respuesta del mismo (J. Martijn y A.P.G., 1999).



Figura 1.4: Inmersión Completa Simulador de Automóvil.

El realismo se refiere a la calidad o detalle de los objetos en el entorno virtual, comparándolos con los objetos de la vida real y como estos interactúan con el usuario. El realismo se logra empleando modelos complejos (muchos polígonos), texturas de alta definición y efectos visuales de alto impacto como relieves en los objetos, sombras realistas, simulación de fluidos y partículas, entre otros. Sin embargo, estos elementos dan como resultado un entorno virtual complejo y por consiguiente difícil de manipular por equipos convencionales (Universidad EAFI T, 1999), es por eso que para poder mostrar alto realismo en los entornos virtuales, se hace uso de las tarjetas aceleradoras de gráficos.

Ammoura (2001) menciona que el factor más importante (aún más que la calidad de las imágenes) para lograr una inmersión en el ambiente virtual, es la velocidad con la que la computadora redibuja constantemente los gráficos que se encuentran a su alrededor, lo que hace sentir al usuario que realmente se encuentra moviéndose en el ambiente, esta velocidad se mide en FPS (Frames o Cuadros Por Segundo) que significa cuantas imágenes del ambiente virtual es capaz de generar la computadora por segundo.

Para lograr una buena velocidad de respuesta o altos FPS, se requiere de una máquina de renderizado poderosa (render engine), la cual interpreta las entradas del usuario (movimiento de cámara, interacción con los objetos, etc.) y dibuja muy rápidamente la escena, con respecto a estas entradas, desde la posición del usuario para después presentar

la imagen final al usuario. Una velocidad de actualización de 60 FPS o superior es deseable para lograr un buen efecto de inmersión (Regan y Pose, 1995).

#### 1.4 Realismo en entornos virtuales y arquitectónicos

El realismo en los entornos virtuales arquitectónicos es un factor clave para su aceptación por parte del usuario, creándole el sentimiento de presencia real en el entorno y haciéndole sentir que forma parte de él.

Existen varias formas de incrementar el realismo en entornos virtuales arquitectónicos, una de ellas es mediante el uso de elementos (objetos, personas, etc.) más complejos y detallados que se logran incrementando la cantidad de polígonos que componen los objetos; sin embargo una gran cantidad de polígonos no suele ser siempre la solución más idónea, ya que, al incrementar este número, se incrementan también los cálculos y el tiempo de procesamiento necesarios para poder visualizar el mismo objeto, lo cual, de ser excesivo, puede provocar un decremento en la fluidez de la representación y los movimientos del entorno virtual, fracasando en el intento de que el usuario perciba el realismo y la inmersión deseadas (Ammoura 2001).



Figura 1.5: Complejo Escolar para Recorrido Virtual.

Con el fin de incrementar el realismo en los entornos virtuales, se recrearon cientos de efectos y técnicas de graficación que se utilizan en éstos y añaden un nivel de realismo extra a las escenas 3D, muchos de estos efectos son recreados mediante el uso casi exclusivo del hardware gráfico, lo que permite su rápida visualización. Existe una gran variedad de efectos visuales soportados por hardware, de los cuales no todos pueden ser

usados en el mismo entorno, esto dependerá del tipo de entorno que se esté creando, de la sensación que se desea plasmar en el usuario y de los objetos en éste, ya que algunos efectos sólo pueden ser aplicados a cierto tipo de objetos en específico (agua, metales, cielo, etc.).

## 1.5 Metodología usada en el desarrollo del entorno virtual arquitectónico

Después de estudiar diversas tecnologías en el desarrollo de software, se optó por seleccionar la metodología de “Desarrollo Orientado a Prototipos” (Mcconnell, 1997; Preece et al., 1994), ya que es ésta la que se considera más adecuada y orientada a lo que se pretende realizar.

Con este tipo de metodología, el usuario final (el cliente, futuro dueño de la arquitectura) participa constantemente en el desarrollo del prototipo, lo cual es útil cuando éste no tiene una idea clara y precisa de lo que desea. El cliente puede aportar retroalimentación para el desarrollo de la siguiente versión del prototipo, hasta que quedar satisfecho con los resultados.

Cuando se desarrolla una arquitectura (en este caso viviendas), muchas veces el cliente tiene ideas o nociones de cómo le gustaría que fuese su casa, por lo tanto, el arquitecto debería de tomar en cuenta estas ideas durante el desarrollo del diseño de la arquitectura, para que al término de ésta, el cliente se sienta satisfecho y contento con la apariencia final de su casa. El cliente puede ir aportando sus ideas durante el diseño de la vivienda, por ejemplo, opinando en el tipo de fachada, la definición de los espacios interiores (cocina, baños, recamaras, etc.), para que al terminar el diseño, el cliente este completamente de acuerdo en la apariencia final de la casa, y posteriormente, proceder con una construcción real de la vivienda. Incluso la misma arquitectura virtual se podría emplear en la decoración del interior de la vivienda (seleccionando colores, tipos de piso, etc.).

A continuación se muestra un diagrama de las etapas de desarrollo usando la metodología de desarrollo orientado a prototipos.

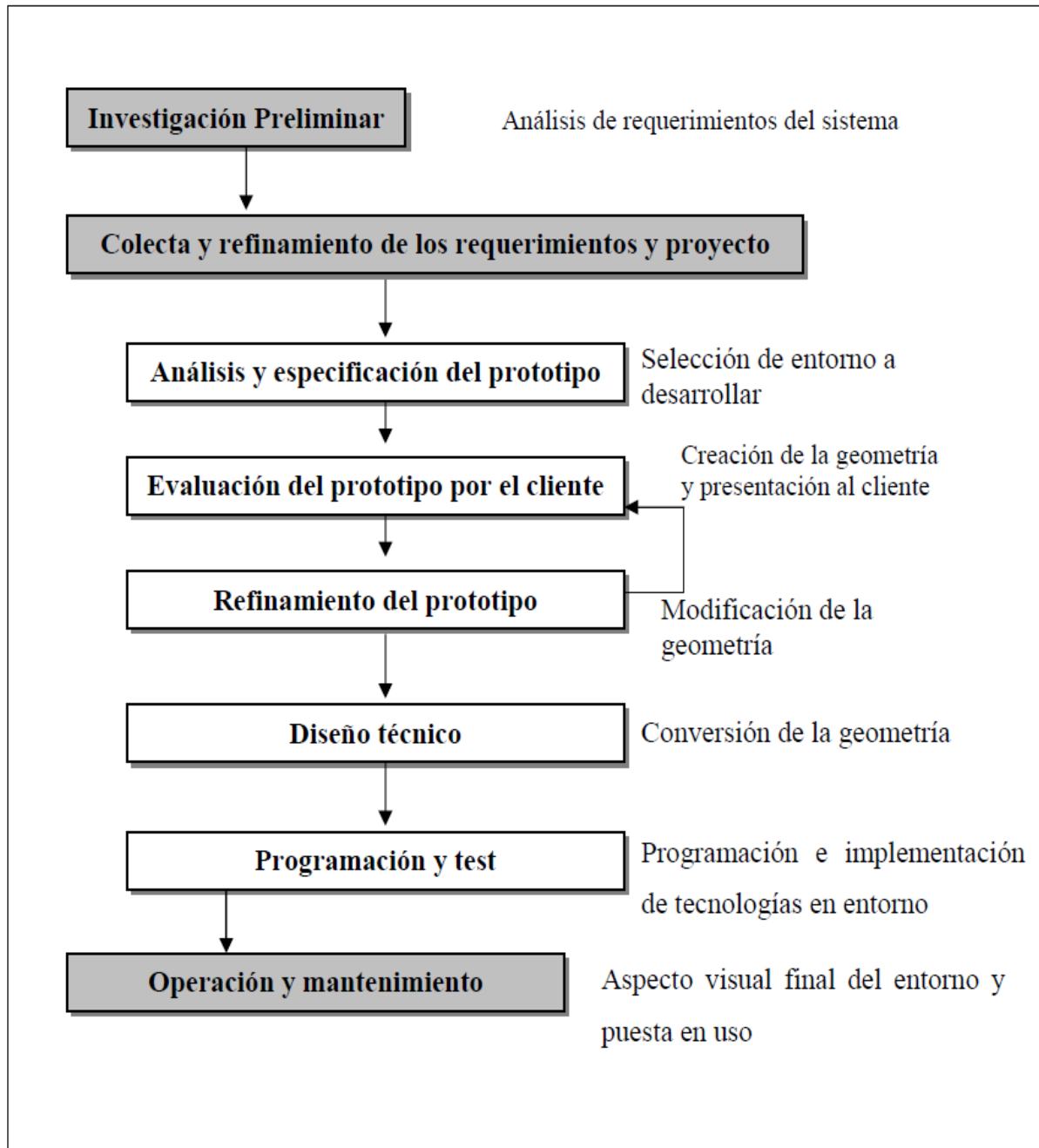


Figura 1.6: Etapas de desarrollo orientado a prototipos.

Se sugiere esta metodología, cuando se desea crear una arquitectura que no existe, ya que permite que al término de ésta, el cliente se encuentre completamente satisfecho, ya que sus aportaciones se ven reflejadas en la apariencia final de la arquitectura. Ya que la arquitectura desarrollada para esta tesis ya existía, no se requirió de grandes aportaciones del cliente en la etapa de la estructura de la arquitectura, únicamente aquellas relacionadas

con la decoración del interior de la vivienda, como tipos de pisos, colores de paredes, adornos, amueblado, etc.

## Conclusiones del Capítulo

En este capítulo, presentamos una investigación en la literatura, primeramente sobre la definición de que es la realidad virtual, así como el inicio de la realidad virtual. Después se presenta la investigación de como la arquitectura se va haciendo presenta en la realidad virtual primeramente a través de imágenes 2D y luego por recorridos virtuales y ambientes 3D. Es importante mencionar, hablando de realidad virtual, los sistemas de inmersión que existen, ya que si no existe un sistema de inmersión mínimo o básico, no podemos hablar de realidad virtual. En este capítulo, se presenta la definición de inmersión.

Como puntos finales de este capítulo, se menciona como se desarrolló el realismo en entornos virtuales para que la experiencia de la realidad virtual tenga mayor credibilidad al usuario, así también se presenta la metodología correcta de cómo se tiene que desarrollar un ambiente virtual.

Como conclusión importante del capítulo, se presenta una investigación de los términos que son necesarios para comprender, en que consiste este trabajo de tesis. Comprendiendo estos términos, se puede entender mejor, que detrás de la Facultad Virtual, hay mucha literatura y teoría que se tomó en cuenta para su construcción y que no solo es un proyecto sin beneficios a la comunidad universitaria sino que tiene muchas aportaciones y que se comprende mejor, teniendo una base fuerte de conocimientos.

## Capítulo 2

### Modelado de Edificios de la Facultad de Ciencias de la Computación

*El modelado de objetos 3D, no solamente, es una parte fundamental en el proceso de creación de 3D, sino es que la más importante, ya que sin modelado 3D no se podría tener una representación digital del objeto. La creación de un objeto en 3D requiere de técnicas de modelado que ayudan a la creación del objeto 3D en sus proporciones correctas. Es importante mencionar que la creación de un objeto 3D cualquiera, no solo es su modelado, sino también su proceso de creación de mapas UV así como su texturizado y que en este capítulo se describirán estos procesos.*

#### 2.1 AC3D

AC3D ha sido diseñado para hacer la construcción de objetos 3D rápida y fácil. Es usado para crear modelos 3D para simulaciones, juegos, imágenes obtenidas mediante trazado de rayos, y para la visualización de datos científicos y generales.

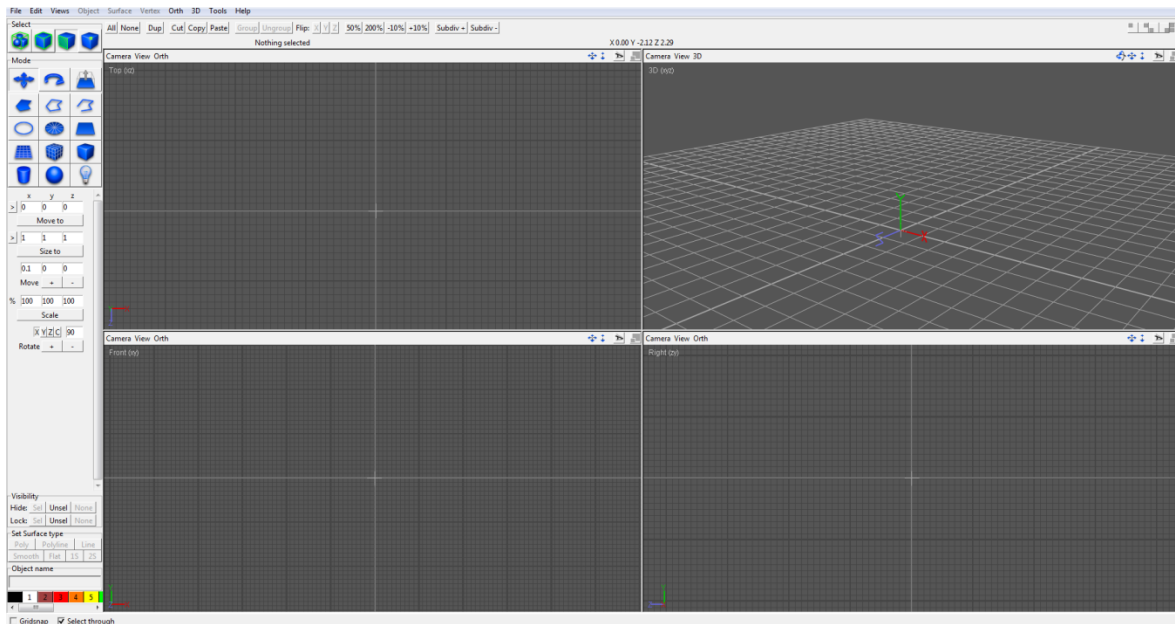


Figura 2.1: Interfaz AC3D.

## 2.2 Vistas Ortográficas (2D)

Cuando observamos objetos, por lo general, distinguimos tres dimensiones; con anchura, profundidad, y altura o con longitud, (anchura) y altura, dependiendo de la forma y proporción del objeto.

Las vistas ortográficas se utilizan en el dibujo técnico para describir de manera íntegra y exacta las formas de los objetos. La palabra "ortográfica" se deriva de dos palabras griegas; orto, que significa bien, correcto, en ángulo recto; y grados, escribir o describir con líneas de dibujo. Una vista ortográfica es la que se observa al mirar en forma directa un lado o "cara" de un objeto. Cuando se observa directamente la cara frontal, se distinguen: ancho y altura, dos dimensiones; pero no la tercera dimensión, profundidad. Cada vista ortográfica proporciona dos de las tres dimensiones principales.

## 2.3 Usando Imágenes de Fondo

Cualquier ventana ortográfica (2D) puede mostrar una imagen de fondo (background image). Esto puede ser útil para trazar formas o crear objetos de acuerdo a una imagen escaneada por ejemplo: un plano/mapa. Las imágenes de fondo se activan y desactivan desde el menú Orth en la parte superior de una ventana ortográfica. Todos los formatos de imagen que son soportados por los cargadores de texturas de AC3D pueden ser usados como imágenes de fondo. Esta técnica se llama rotoscopia.

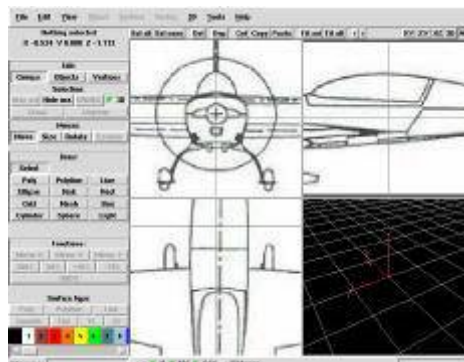


Figura 2.2: Rotoscopia en AC3D.

## 2.4 Modelado de un Edificio utilizando la rotopscopia

Como primer paso, se adquieren los planos arquitectónicos, del edificio, para poderlo utilizarlos de rotopscopia. Esto nos servirá para hacer las proporciones correctas del edificio.

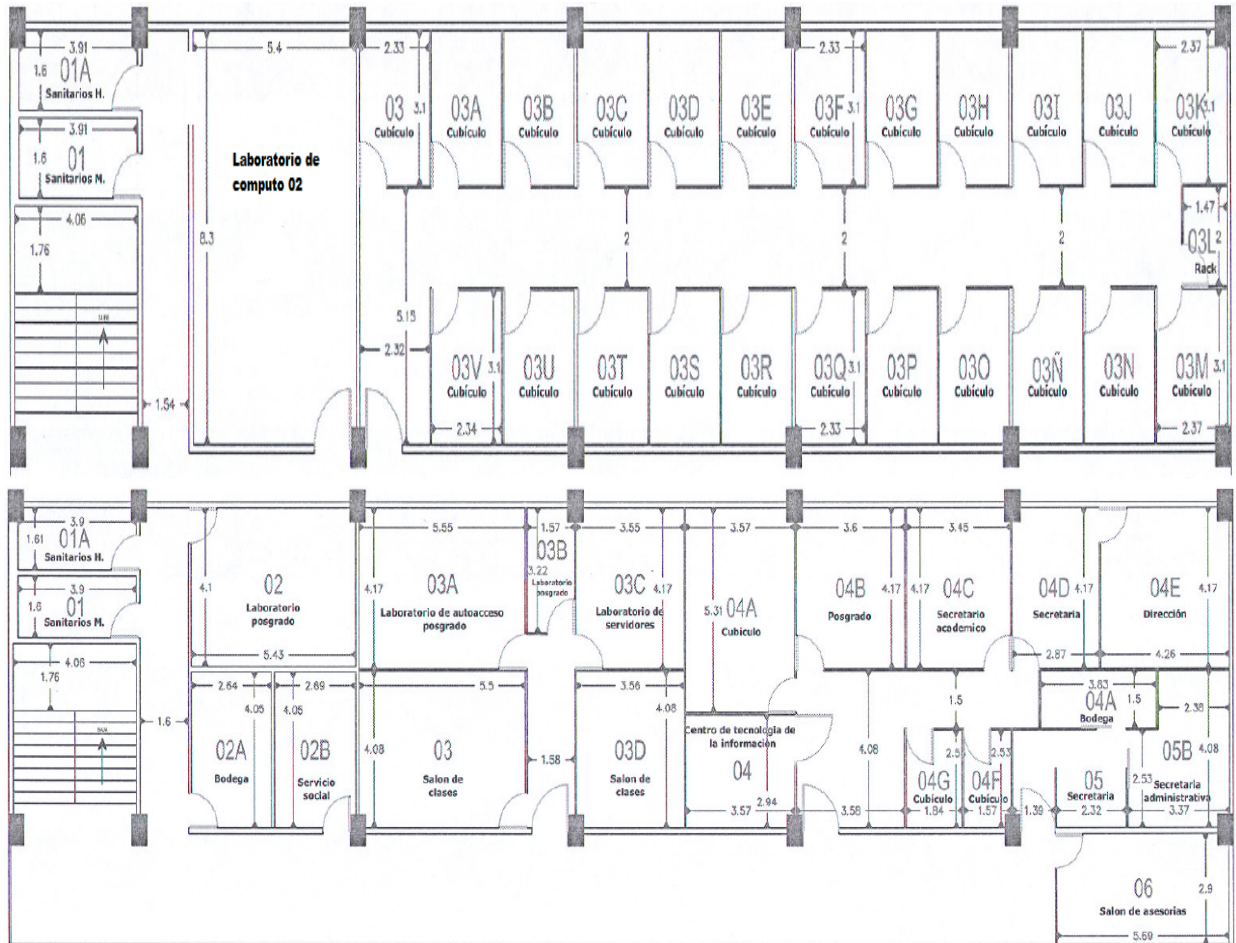


Figura 2.3: Planos arquitectónicos del edificio 135.

Como segundo paso, ponemos los planos arquitectónicos de rotopscopia en las ventanas ortográficas, en este caso en la vista del top de AC3D.

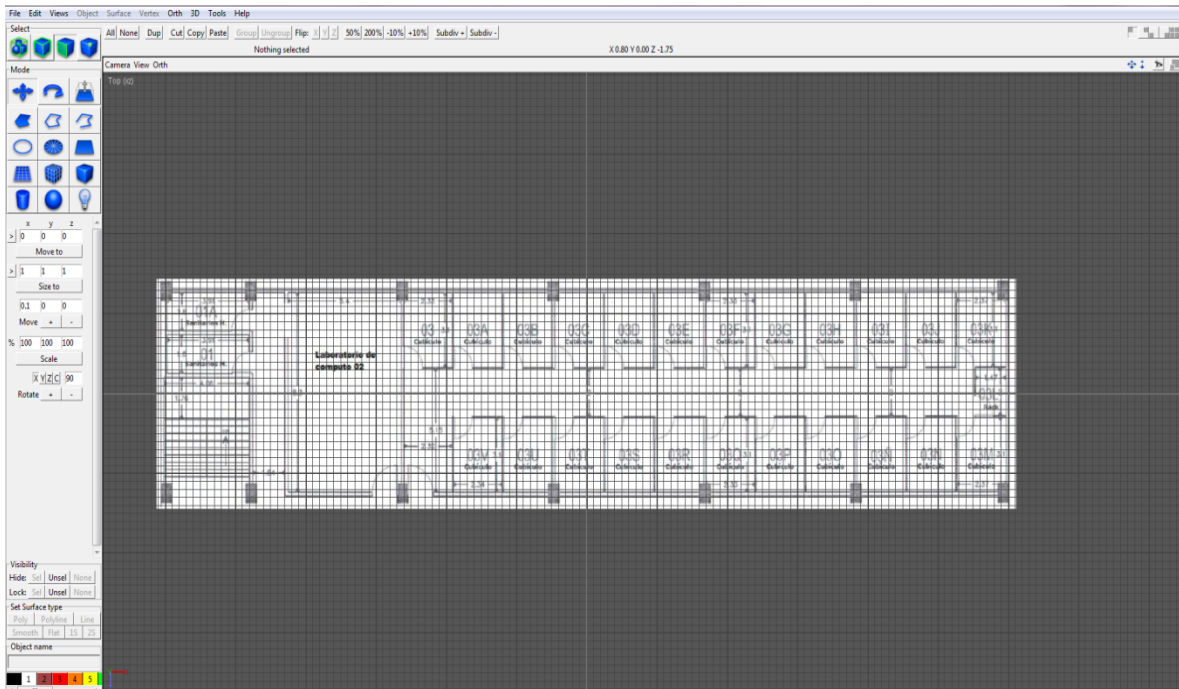


Figura 2.4 Rotoscopia planos del edificio 135.

Ajustamos nuestra el tamaño de la cuadrícula de acuerdo a la unidad de nuestro plano, para que una cuadrícula sea equivalente a la unidad de nuestro plano, en este caso, una cuadrícula sea un metro.

Después colocamos un cubo en el exterior de nuestro plano arquitectónico, y lo ajustamos moviendo los vértices, de tal manera que la longitud del plano, sea la misma longitud de nuestro cubo. Esto lo realizamos hasta encontrarnos con el primer muro del plano, como se muestra en la imagen.

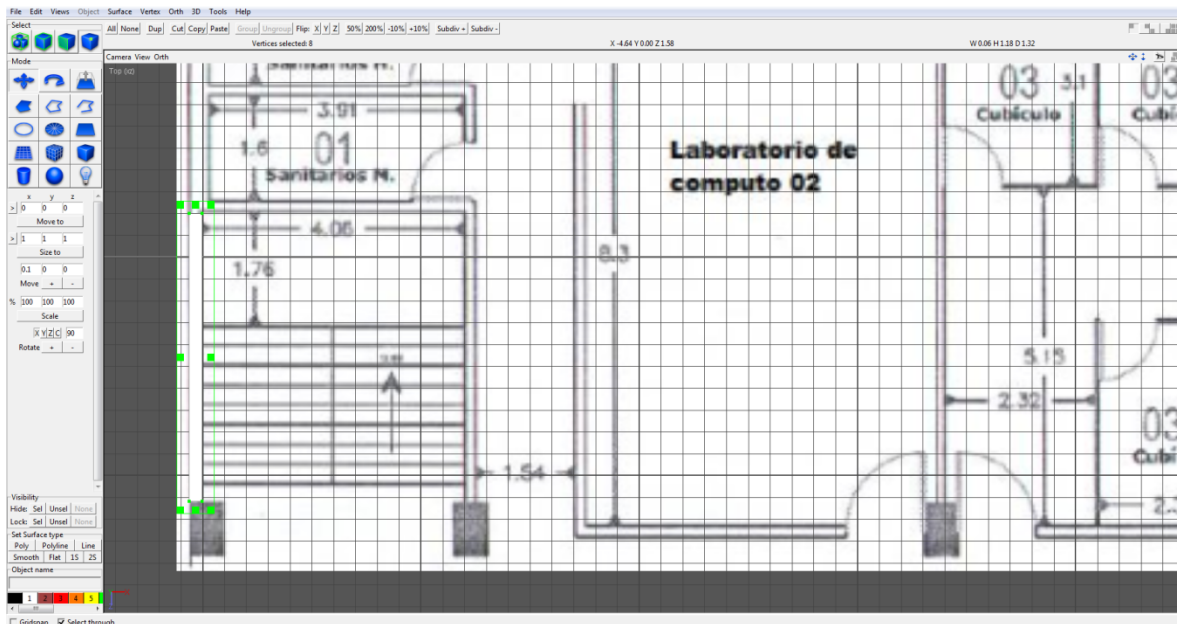


Figura 2.5: Primera pared Edificio 135.

En nuestro panel de selección podemos seleccionar, aristas, vértices y caras de nuestro polígono. Esto nos servirá para poder manipular nuestro objeto y realizar las formas que queramos con el polígono.

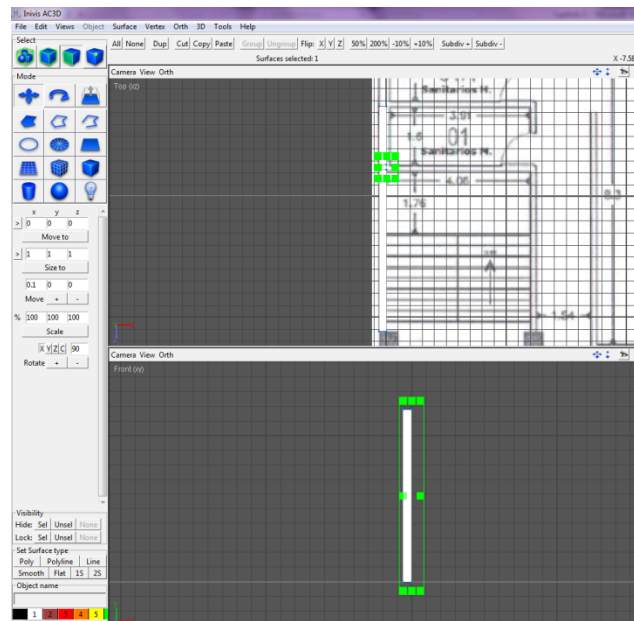


Figura 2.6: Selección de los vértices.

Después seleccionamos los vértices que quedaría en el muro perpendicular al que nos topamos, y realizamos el Extrude de la cara del polígono y lo alargamos al final del muro,

esto para que podamos tener una cara de polígono extra que nos servirá para realizar el muro perpendicular.

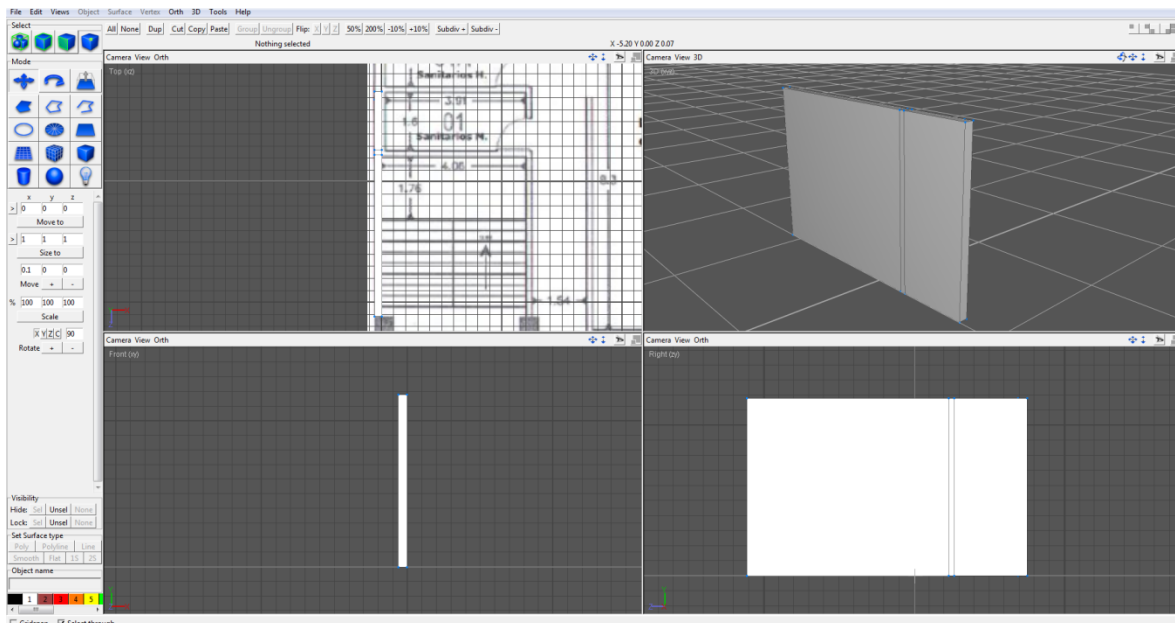


Figura 2.7: Extrusión de los vértices.

De esta forma, realizamos nuestro primer cuarto del edificio, con un solo polígono.

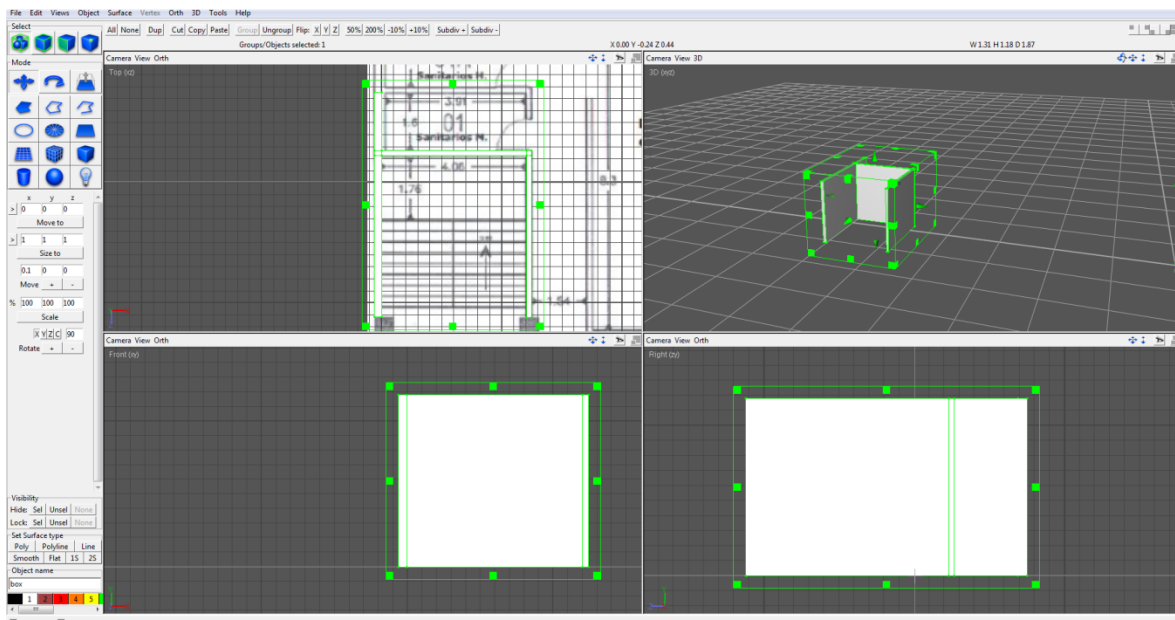


Figura 2.8: Creación de un primer cuarto.

Existen muros independientes que no se podrán crear, del mismo polígono del exterior, en este caso se tendrán que crear polígonos independientes, como es el caso siguiente, el objeto que está en selección.

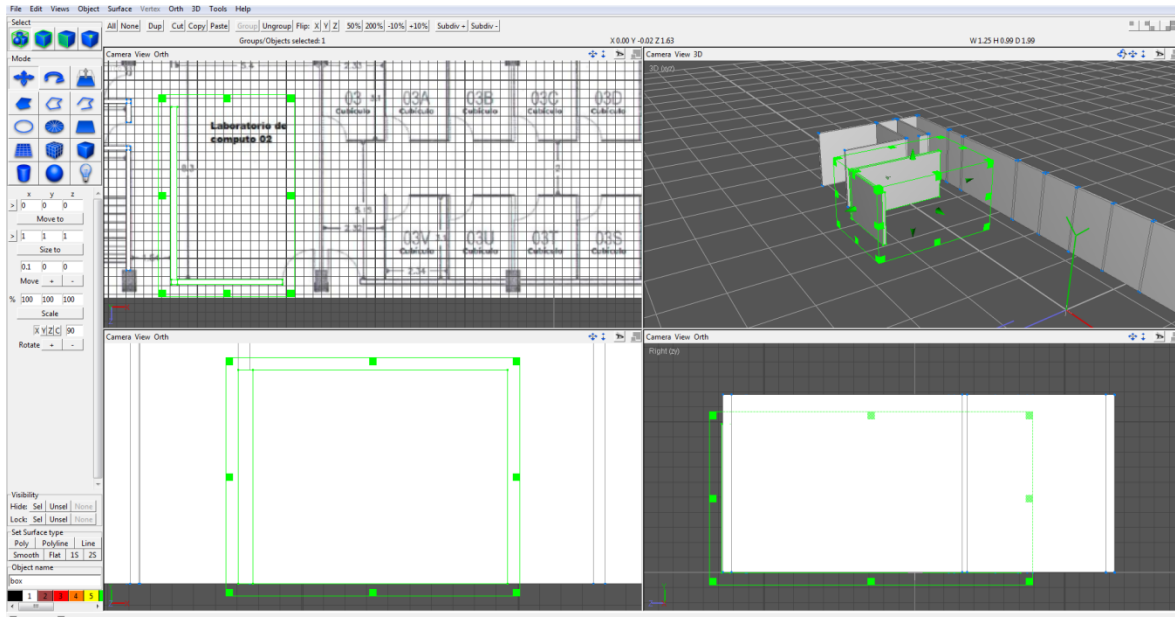


Figura 2.9: Creación de muro independiente.

## 2.5 Texturizado de Objetos 3D

### 2.5.1 Mapeo UV

El mapeo UV o UV mapping, por su traducción al inglés, es el proceso de hacer una representación 2D de un modelo 3D. Este proceso proyecta un mapa de textura en un objeto 3D. Las letras “U” y “V” indican los ejes de la textura 2D, ya que los ejes “X”, “Y” y “Z” se utilizan para referirse a los ejes del objeto 3D en el espacio modelado.

El texturizado UV permite a los polígono que forman un objeto 3D colocar varias imágenes. La imagen se llama mapa de textura. Las coordenadas UV son aplicadas a las caras del polígono, no por vértice. Esto significa que un vértice común puede tener diferentes coordenadas UV en cada uno sus triángulos, de modo que los triángulos adyacentes se pueden cortar aparte y posicionados en diferentes áreas del mapa de textura.

El proceso de asignación de UV en su forma más simple requiere tres pasos:

- Desenvolver la malla ( objeto 3D)
- La creación de la textura.
- La aplicación de la textura.

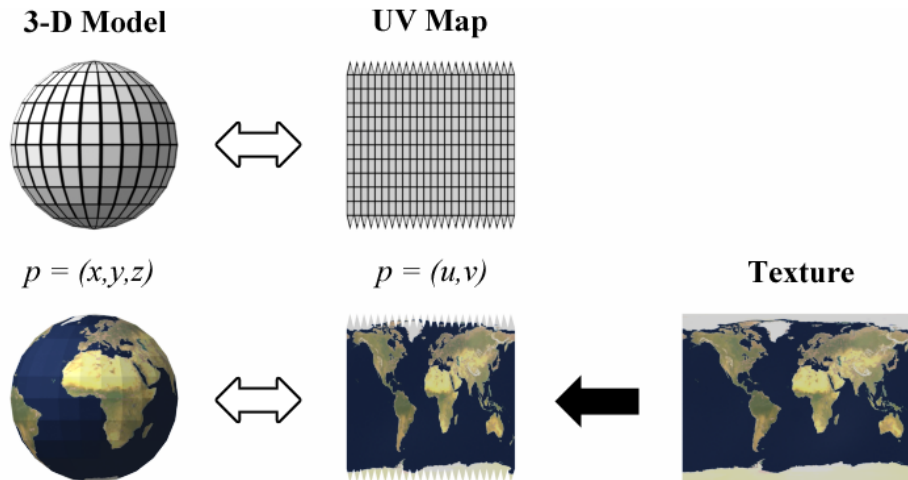


Figura 2.10: Ejemplo texturizado UV Map Esférico.

### 2.5.2 AC3D Texture Coordinate Editor.

Es la herramienta integrada en AC3D que proporciona un control total sobre las coordenadas de textura que se aplican sobre las superficies en los objetos.

Es importante darse cuenta que estas imágenes representan las coordenadas que se asignan a cada vértice referenciado en cada superficie, no tienen relación con la disposición real de los vértices del objeto.

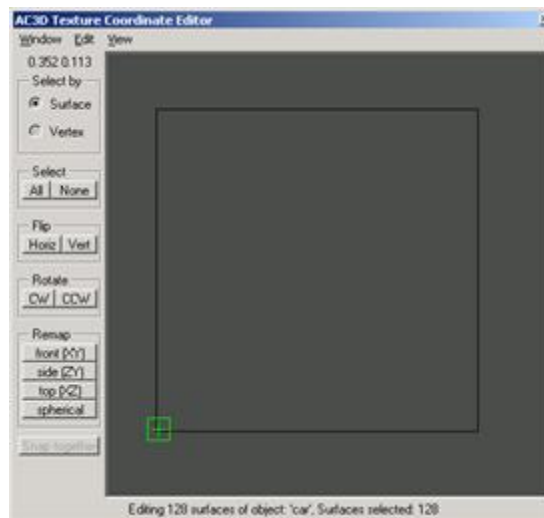


Figura 2.11: AC3D Texture Coordinate Editor.

Como ya explicamos que nuestro edificio, comenzó con un cubo común y corriente, explicaremos como realizar el mapeo uv de un cubo. Primero debemos seleccionar en cualquiera de nuestras vistas, una cara del cubo, como se muestra la imagen.

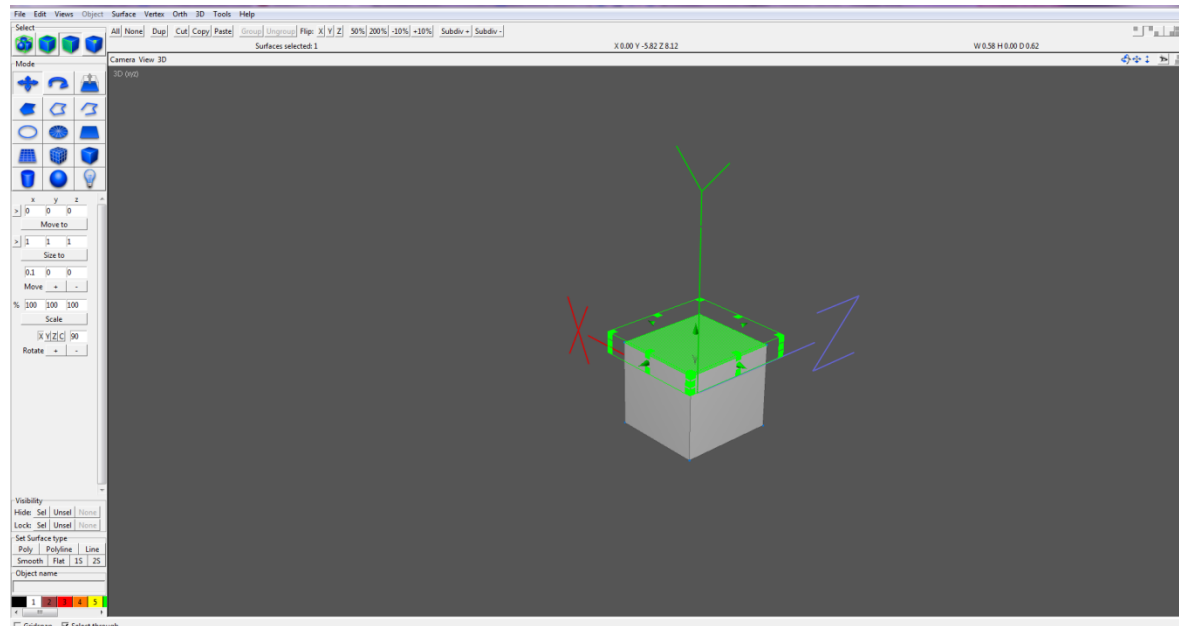


Figura 2.12: Selección primera cara.

En este caso seleccionamos la cara superior del polígono y abrimos nuestro Texture Coordinate Editor (F 10). Ya con el TCE abierto, en las opciones de “Remap” seleccionamos la opción de “top” esto porque la cara superior del polígono están perpendicular hacia el eje Y y no hacia los otros ejes X o Z, lo mismo pasa para la cara inferior del cubo. Después escalamos nuestro primer mapeo y lo colocamos en alguna posición del TCE de tal manera que podamos utilizar el espacio completo para las demás caras del cubo.

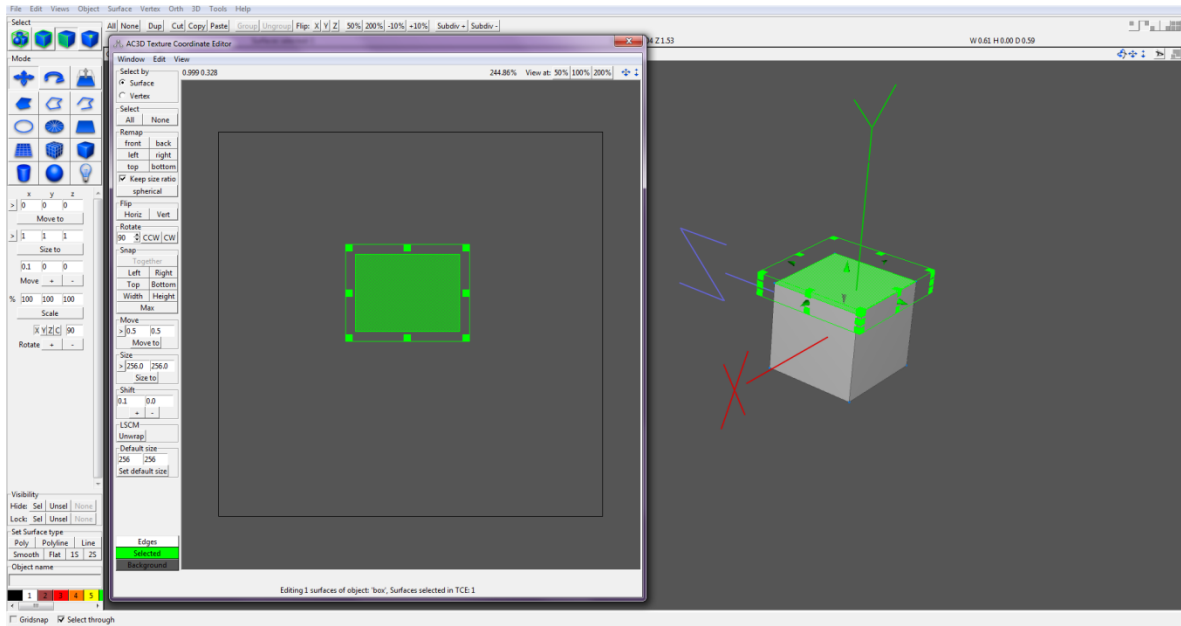


Figura 2.13: Remap primera cara

Así continuamos con las otras caras del polígono, las caras laterales, utilizamos la opción Remap -> Right (o Left, tienen la misma función) y para caras frontales la opción Remap -> Front (o Back)

De modo que nuestra representación 2D del cubo sea la siguiente:

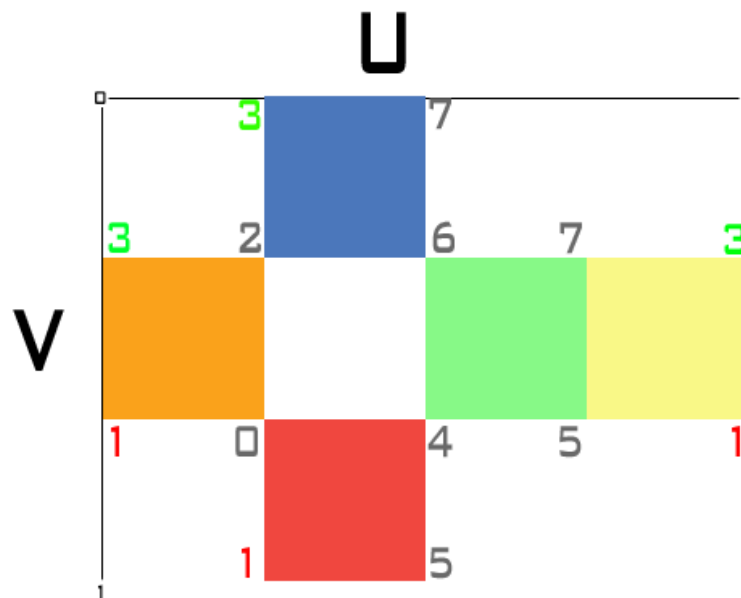


Figura 2.14: Mapeo UV cubo.

### 2.5.3 Texturizando nuestro edificio

De la misma forma que realizamos el mapeo UV de nuestro cubo, realizamos el mapeo UV del edificio, las paredes de nuestro edificio que están viendo de frente, tendrán que llevar Remap -> front, los techos y suelos un Remap-> top.

Ya con nuestro mapa UV de nuestro edificio, ya lo podemos pasar a cualquier editor de imágenes y utilizar el mapa UV de roscopia y texturizar el edificio. En este caso utilizamos Adobe Photoshop para poder realizar la textura completa. Una vez completada la textura podemos aplicar la textura, desde Object -> Texture -> Load Texture.

Con estos pasos concluye el proceso de cómo se construyeron los edificios de la Facultad de Ciencias de la Computación.

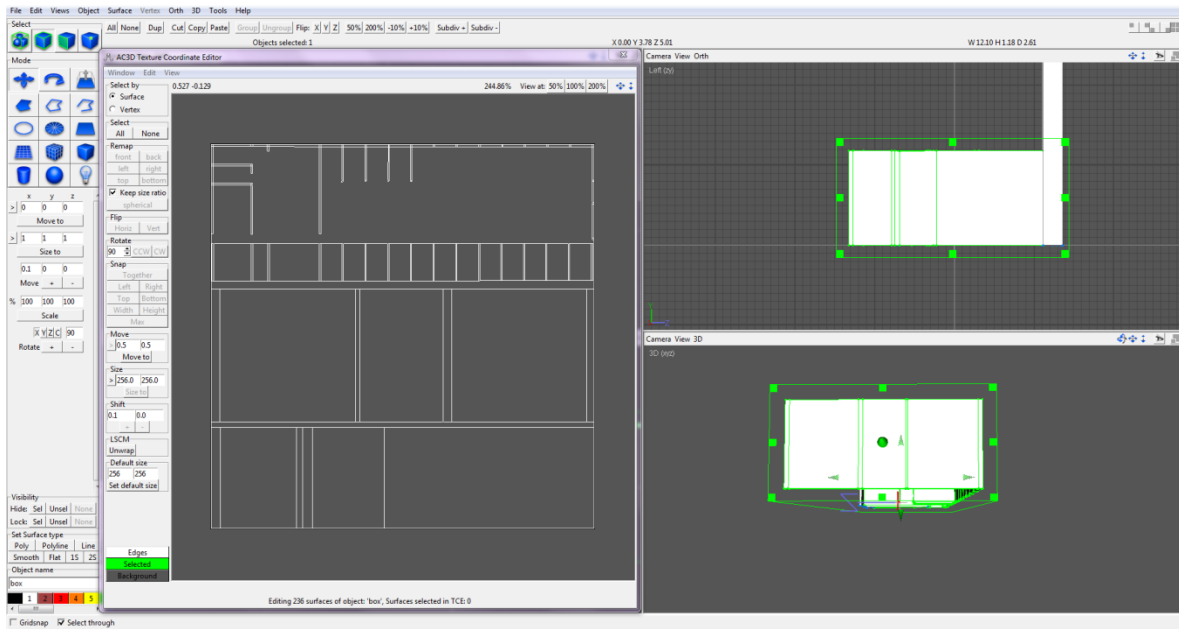


Figura 2.15: Mapeo UV edificio 135.

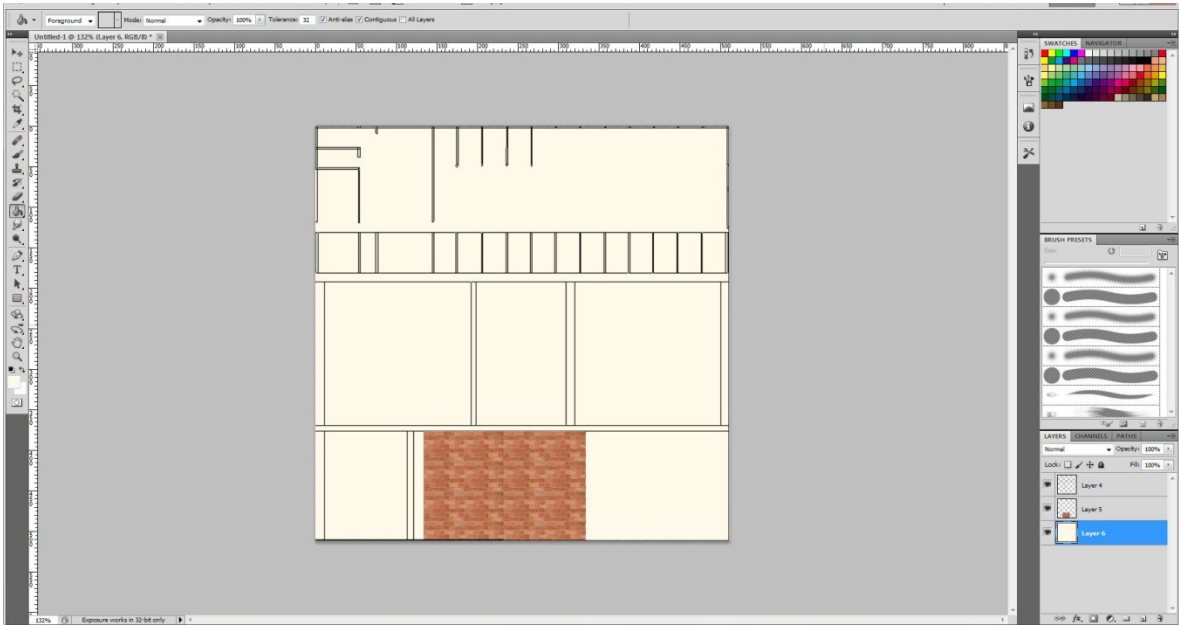


Figura 2.16: Texturizado en photoshop del edificio 135.



Figura 2.17: Modelo Edificio 136.

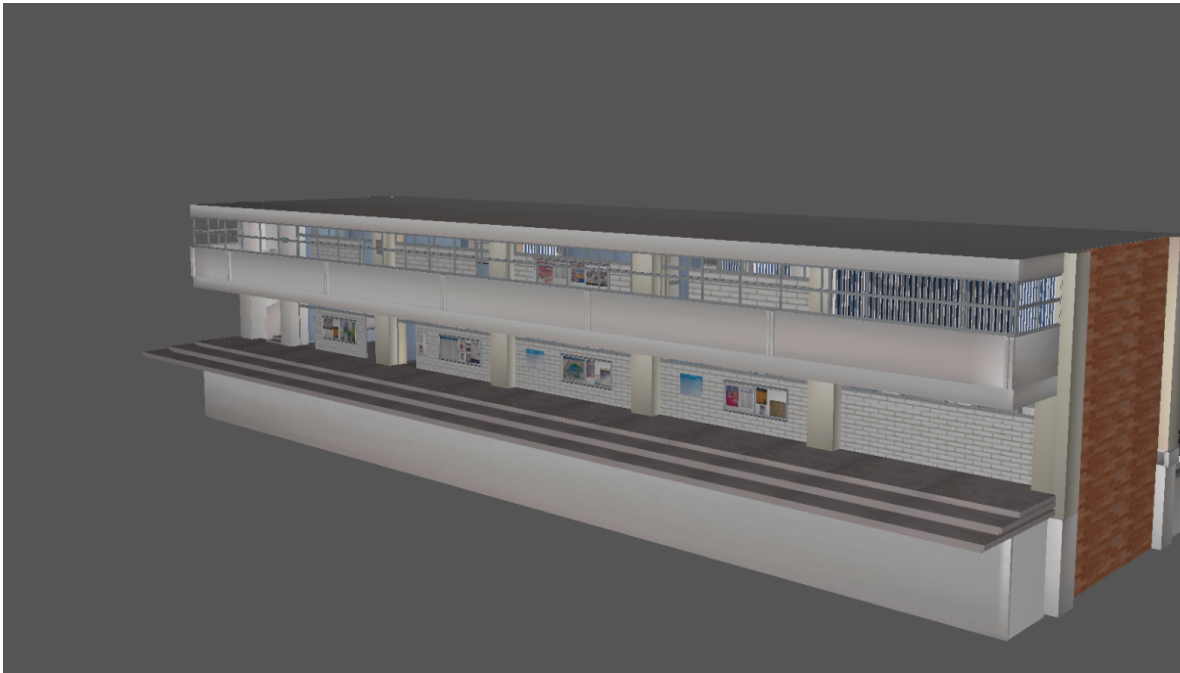


Figura 2.18: Modelo Edificio 135.



Figura 2.19: Modelo Edificio 104C.

## Conclusiones del Capítulo

La conclusión más importante de este capítulo, es la descripción paso a paso, que se tiene que realizar para tener nuestro modelo 3D ya que estos pasos, no solo son aplicados a los edificios arquitectónicos, sino también, se puede aplicar a la creación de cualquier objeto 3D.

- La rotoscopia nos ayudará a realizar nuestro objeto con las proporciones correctas.
- El proceso de Mapeo UV del objeto nos ayudará a ponerle una textura a nuestro modelo 3D de forma precisa.
- El texturizado muestra mayor realismo a nuestro modelo 3D, con imágenes que pueden ser la textura del objeto.

Estos modelos, así como su textura se exportan a Unity sin ningún problema en formato .3ds

Por cuestiones de espacio en la redacción del documento de tesis, no se describe el proceso de modelado de todos los edificios, ya que el proceso es el mismo para todos.

## Capítulo 3

### Adaptación de un Archivo BVH a un personaje

*En este capítulo se describirá la forma de importación de un archivo Motion Capture en formato .bvh, a Softimage, así como la integración del archivo al personaje y su exportación en el formato que Unity 3d requiere para aplicar el Sistema de Locomoción. Hay que recordar que la captura de movimientos o MOCAP son términos utilizados para describir el proceso de registrar el movimiento y la traducción de ese movimiento a un modelo digital.*

Para importar el archivo que contiene la captura de movimiento utilizando el software Softimage 2012, damos clic en el botón “File” seleccionamos en el menú “Import” y le damos clic en la opción “Biovision File”.

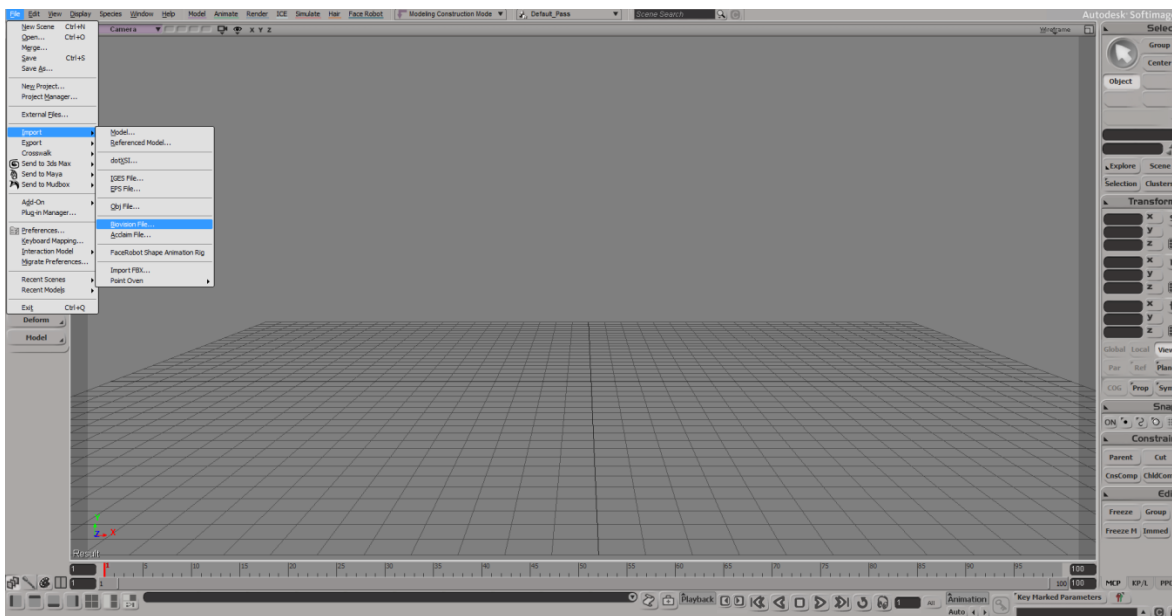


Figura 3.1: Menú Importar archivo .bvh Softimage.

Inmediatamente nos sacará un cuadro de búsqueda para seleccionar el archivo que deseamos importar.

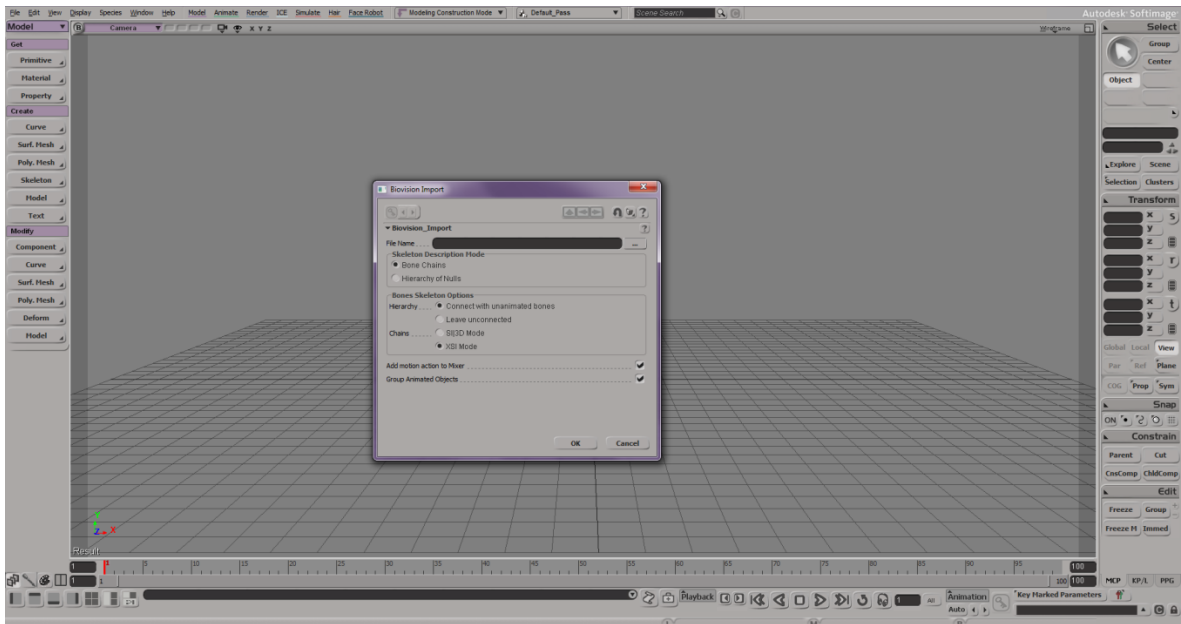


Figura 3.2: Selección del archivo .bvh.

Una vez abierto el archivo de captura de movimiento, debemos ajustar el esqueleto a una posición igual que nuestro personaje. En el proceso de modelado de un personaje en 3d, la pose para presentar un personaje es la “pose T” o “T Pose”. Los archivos BVH tienen una pose inicial, por lo general la pose T, para hacer el ajuste del personaje a la captura de movimiento. En este caso seleccionamos la pose T como se muestra en la siguiente imagen.

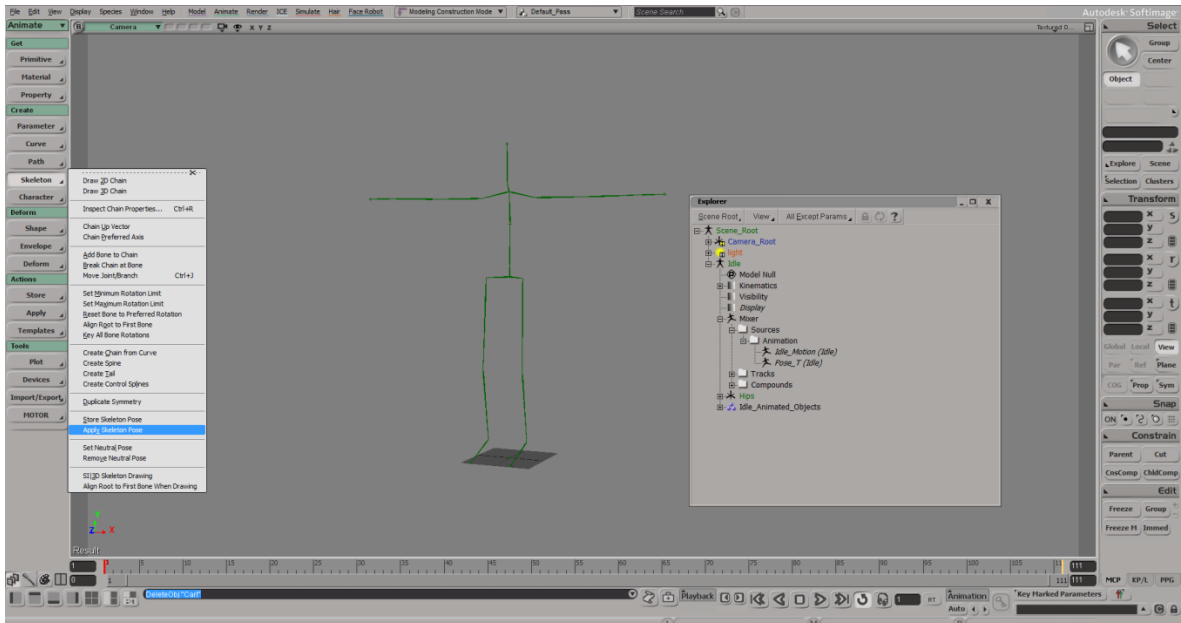


Figura 3.3: Pose T archivo .bvh.

Después podemos importar un personaje cualquiera. En este caso el formato de objetos 3d que más se utiliza para exportar objetos 3d y que todo software reconoce, es el formato “.obj”. Lo importamos desde el menú “File”, seleccionamos en el menú “Import” y le damos clic en el botón “OBJ file...” Inmediatamente nos sacará un cuadro de búsqueda para seleccionar el archivo que deseamos importar

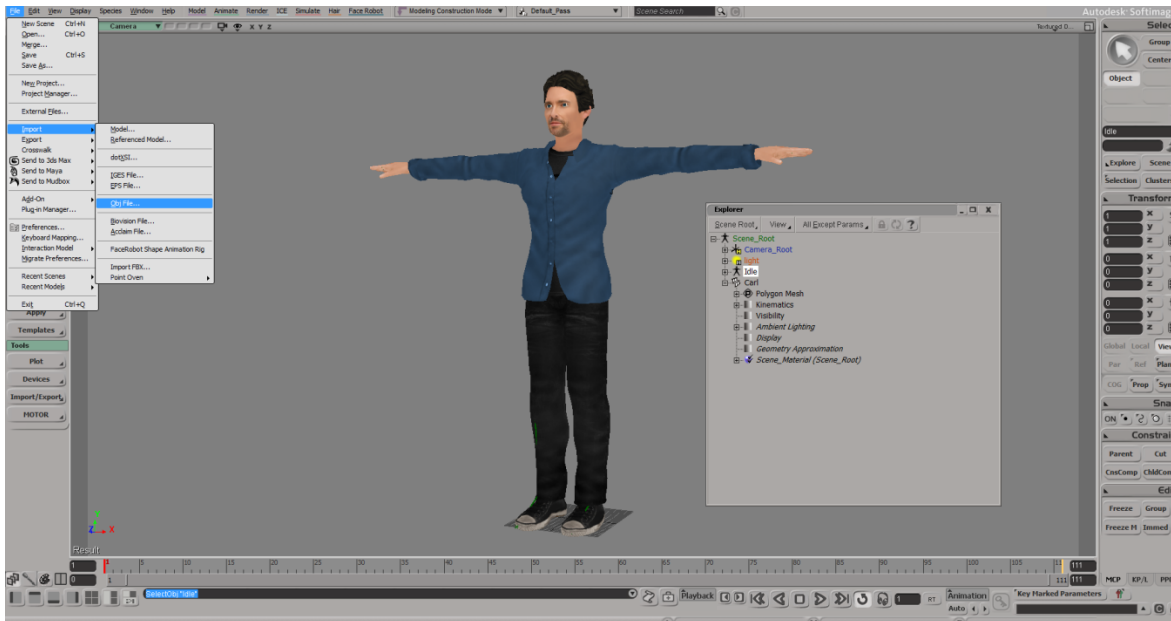


Figura 3.4: Menú Importar archivo .obj.

La mayoría de las veces, nuestro personaje, no tiene las mismas dimensiones que nuestro esqueleto del archivo de captura de movimiento. En este caso debemos ajustar los huesos de nuestro esqueleto al personaje, este proceso se debe hacer en modo “Alambre” o “Wireframe” para tener una mejor visión del ajuste.

Las vistas ortogonales de perfil y de frente nos sirven para realizar el ajuste. Primero ajustamos los huesos de la cadera hacia abajo, lo que son las piernas y los pies.

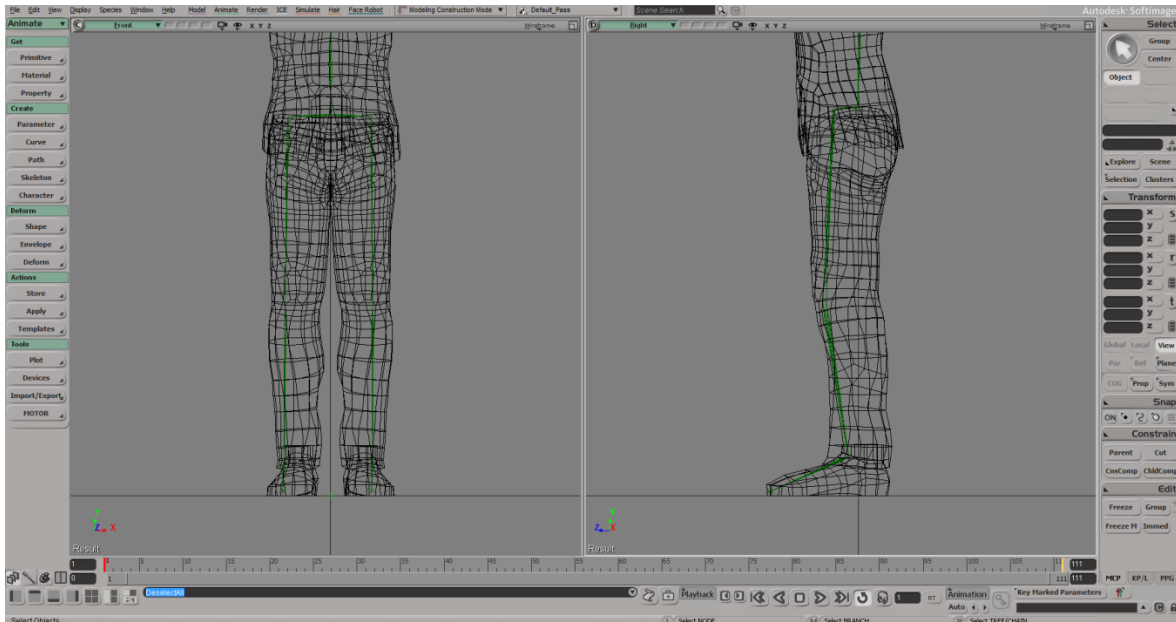


Figura 3.5: Ajuste longitud de pies.

Algo que hay que tomar en cuenta es que el ajuste del esqueleto se realiza por medio de la longitud de los huesos, nunca utilizando la transformación de Escalado del hueso.

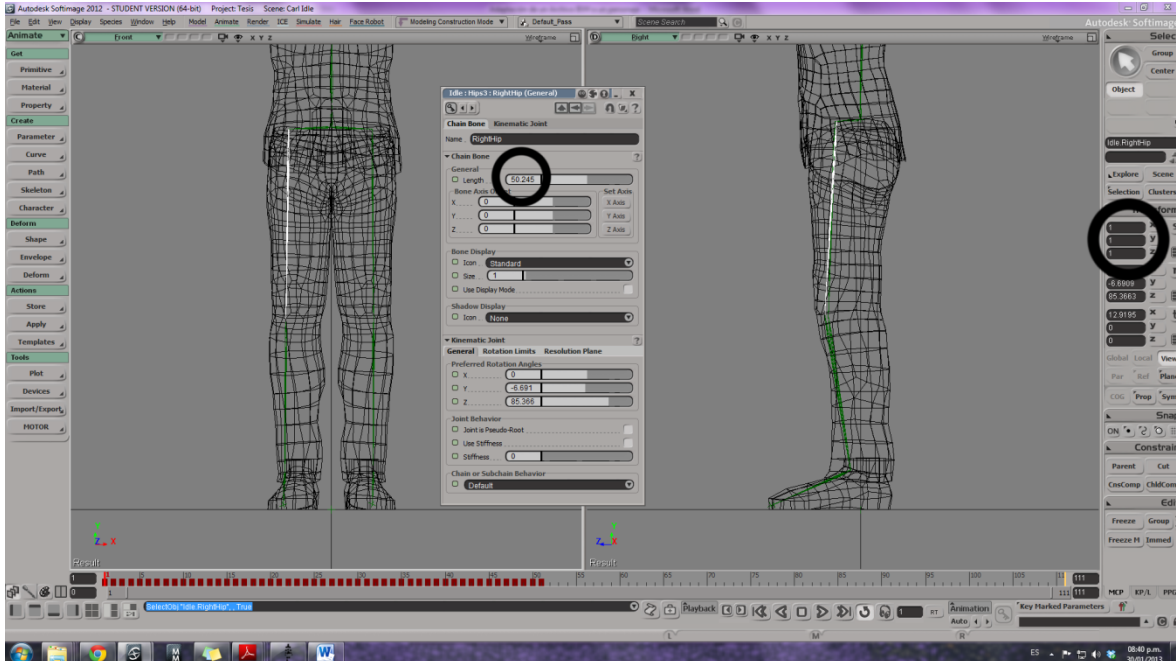


Figura 3.6: Ajuste longitud piernas.

Después ajustamos los huesos de la espina del personaje, así como la del cuello, cabeza como muestra la imagen siguiente.

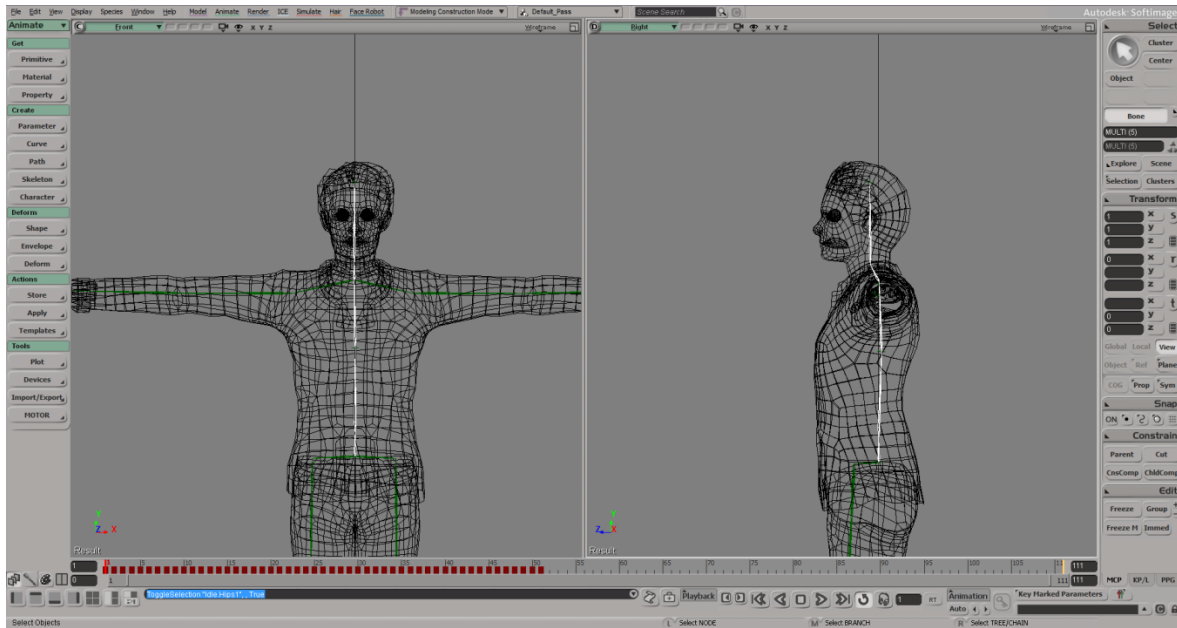


Figura 3.7: Ajuste longitud columna, cuello y cabeza.

Terminamos con la longitud de los brazos, antebrazos y manos. Para ajustar estos huesos, las vistas ortogonales de frente y de superior, nos dan una mejor referencia para el ajuste.

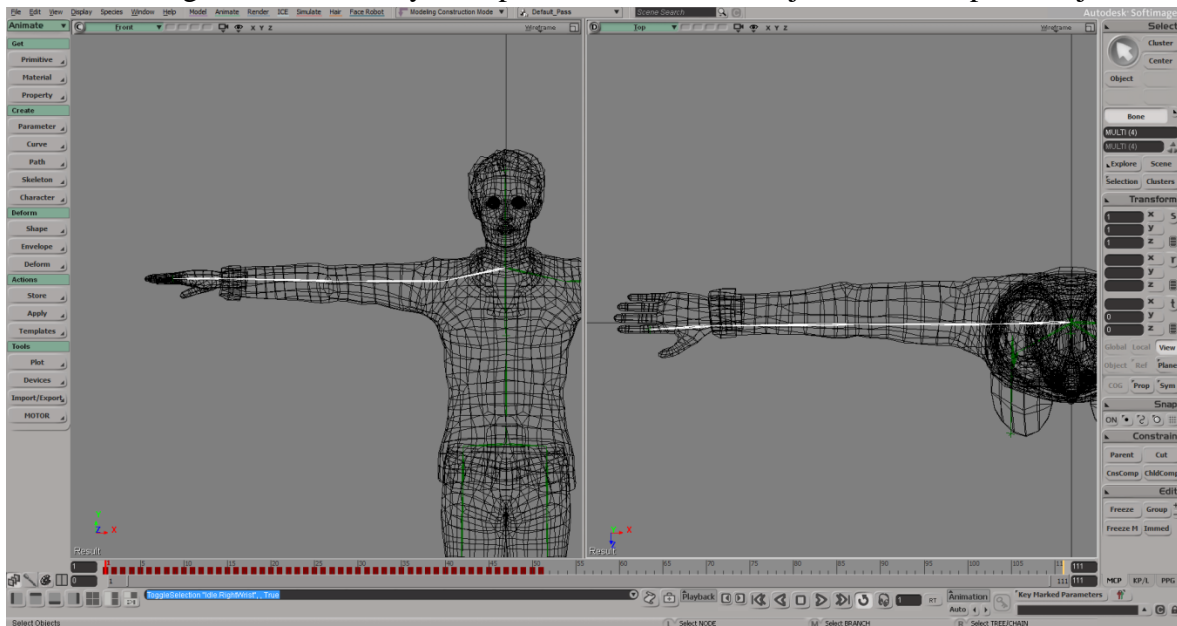


Figura 3.8: Ajuste hombro, brazo y mano.

Ya que tenemos el esqueleto, ajustado, lo que hace falta es asignarle el esqueleto a nuestro objeto, este proceso se llama en Softimage, set Envelope. Enveloping es el proceso de deformación de objetos por medio de los huesos del esqueleto. Se pueden ajustar que puntos del objeto son deformados por los huesos del esqueleto y que valor de influencia tiene cada punto. El proceso de Enveloping se define en la siguiente imagen.

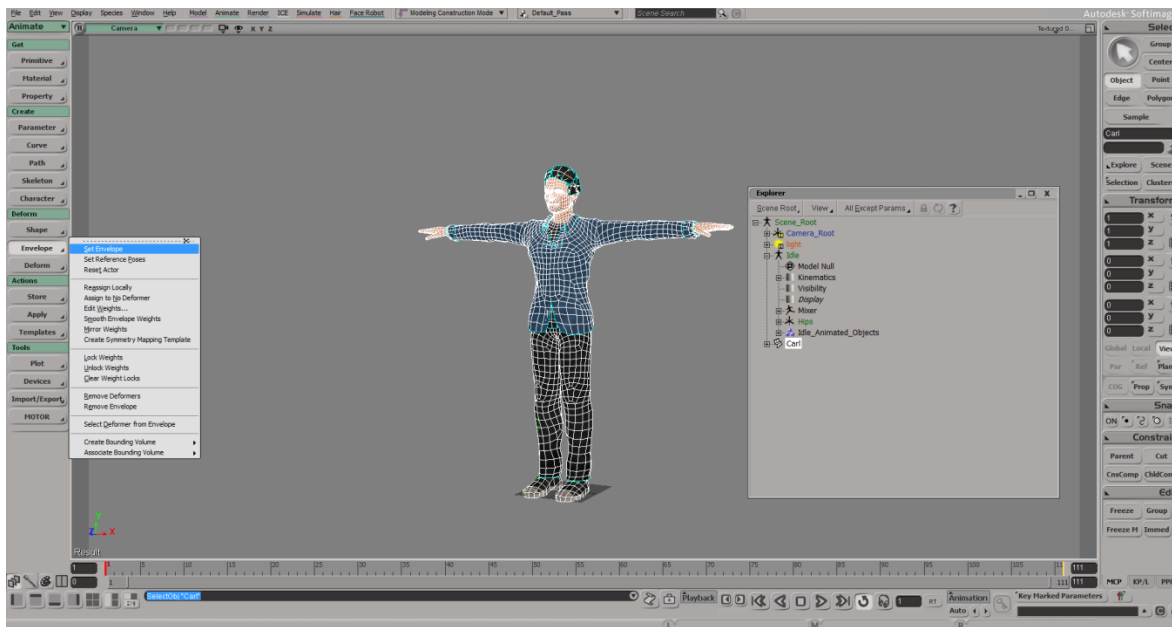


Figura 3.9: Menú Enveloping.

Nuestro personaje ahora aparece de varios colores, lo cual representa la influencia que tiene cada hueso en el personaje. Cada color representa un hueso del esqueleto.

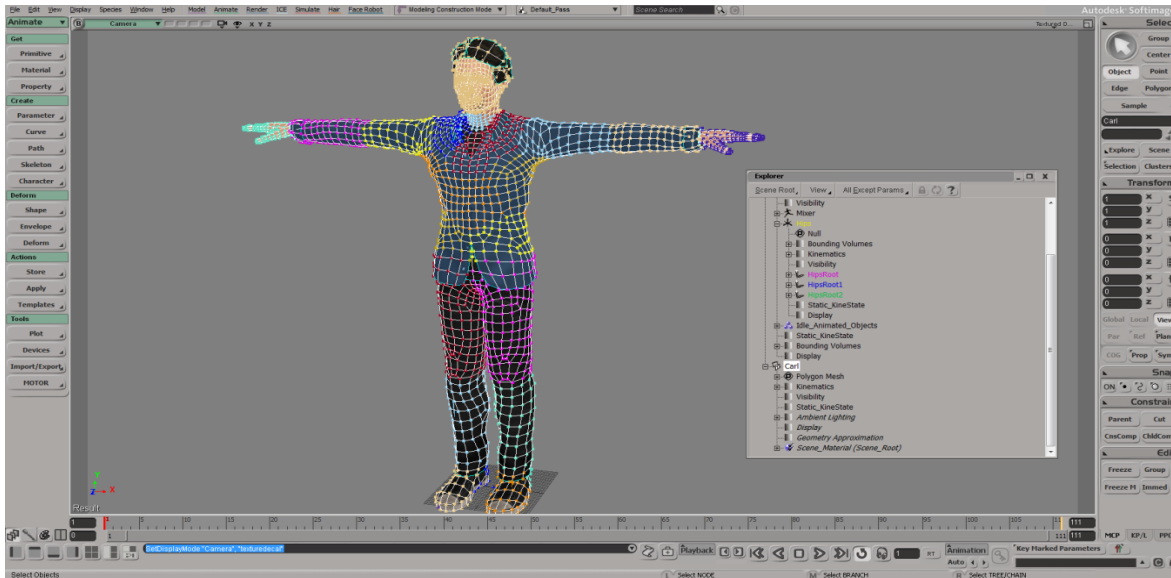


Figura 3.10: Envelope Weights.

El proceso de Enveloping no termina aquí, ya que el programa no sabe con total claridad la influencia que tiene cada hueso en el objeto, a continuación se presenta un ejemplo de este problema. Y en las líneas siguientes se muestra cómo arreglar estos detalles.

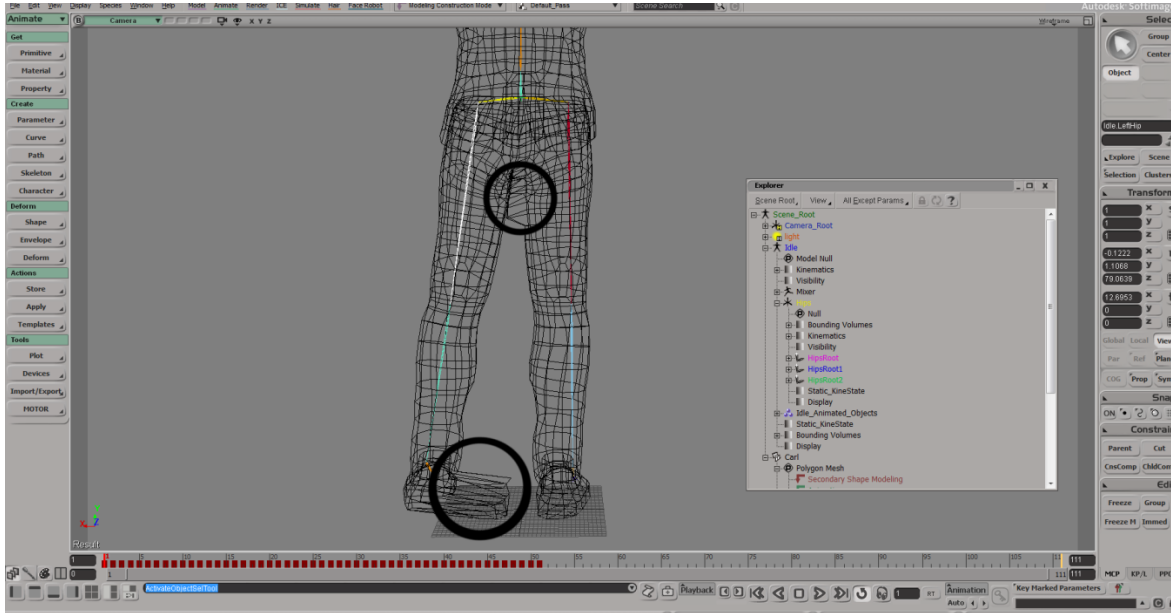


Figura 3.11: Errores en la influencia del hueso.

Existen varias formas de arreglar estos detalles de la influencia de los puntos a los huesos. Uno de ellos es seleccionando los puntos que no están correctamente influenciados y nos vamos a la opción Envelope -> Reassign Locally como lo muestra en la figura siguiente.

Después seleccionamos el hueso que queremos que tenga la nueva influencia. En este caso seleccionamos el hueso del pie “LeftAnkle” y veremos que la influencia será correcta.

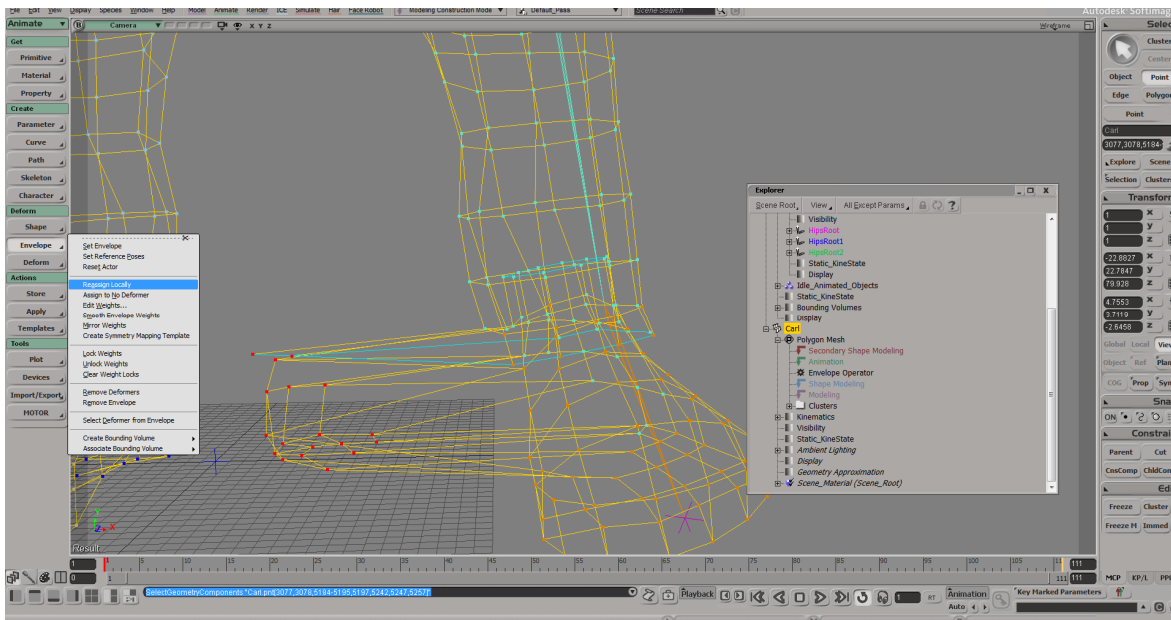


Figura 3.12: Corrección mediante menú Reassign Locally.

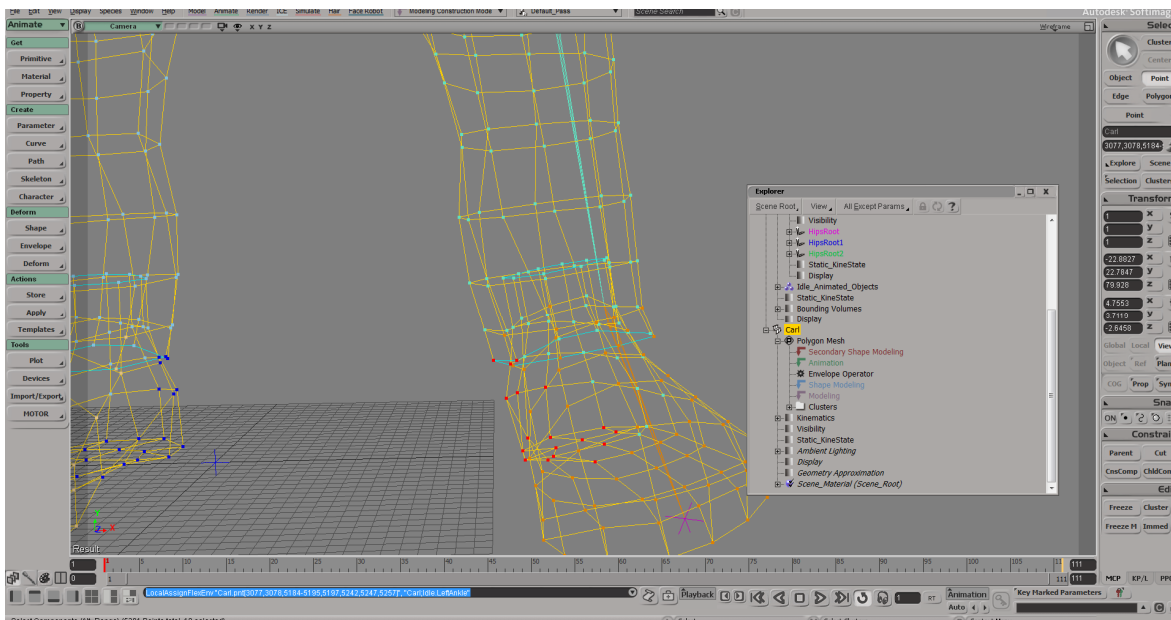


Figura 3.13: Resultado corrección de la influencia.

Otra forma de corregir es mediante el Weight Editor Panel, este panel, nos indica la influencia del punto que seleccionamos, la representación de la influencia nos la presenta en números y mediante la barra de aumento podemos agregar o quitar influencia del punto y ponerlo en otro hueso. En la imagen vemos que los puntos tienen una influencia de un objeto llamado “Idle” del 100 y que del hueso LeftAnkle tienen influencia 0. En este caso seleccionamos la Columna de LeftAnkle y mediante la barra de aumento, subimos la influencia a 100 sobre el LeftAnkle o como mejor nos convenga. Para que podamos ver y manipular el Weight Editor, seleccionamos los puntos del objeto y presionamos “Ctrl + e”.

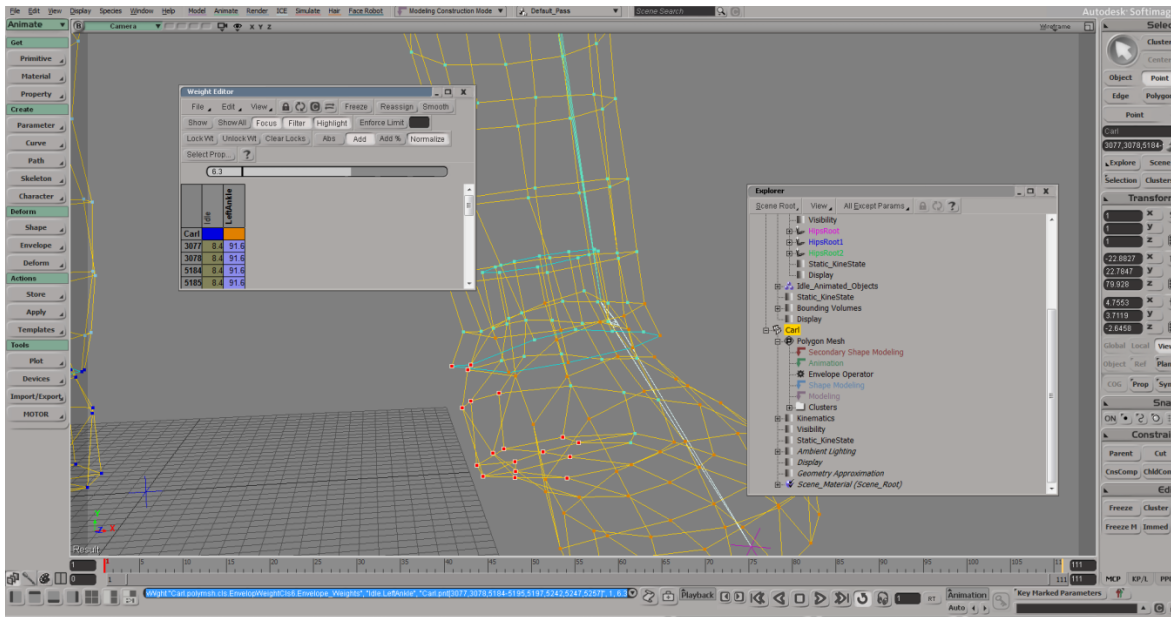


Figura 3.14: Corrección mediante menú Weight Editor.

Una vez que terminemos de hacer el proceso de Enveloping de nuestro personaje, lo que queda por hacer es, realizar un plot de los huesos del esqueleto para poder exportarlo como un archivo .fbx.

El primer paso es activar en nuestro filtro de selección, indicamos seleccionar solamente “Bone”, como lo muestra en la imagen y hacer un recuadro que seleccione los huesos del personaje

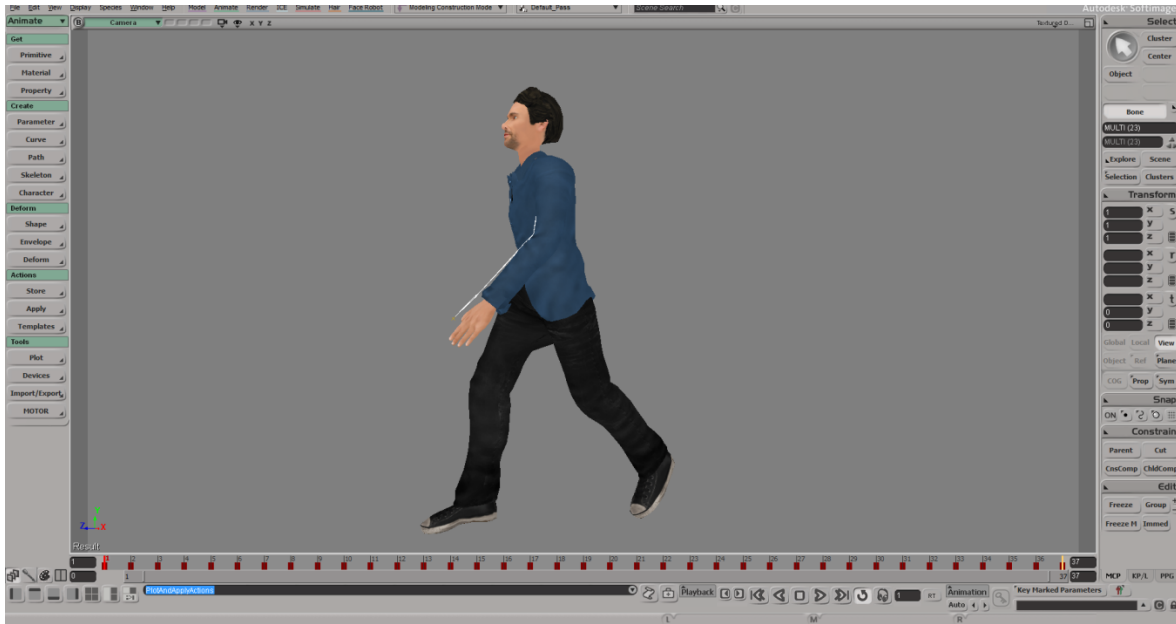


Figura 3.15: Selección de huesos.

Después debemos de seleccionar en la sección de Animate, el menú “Plot” y seleccionar “Rotations” como lo muestra la imagen.

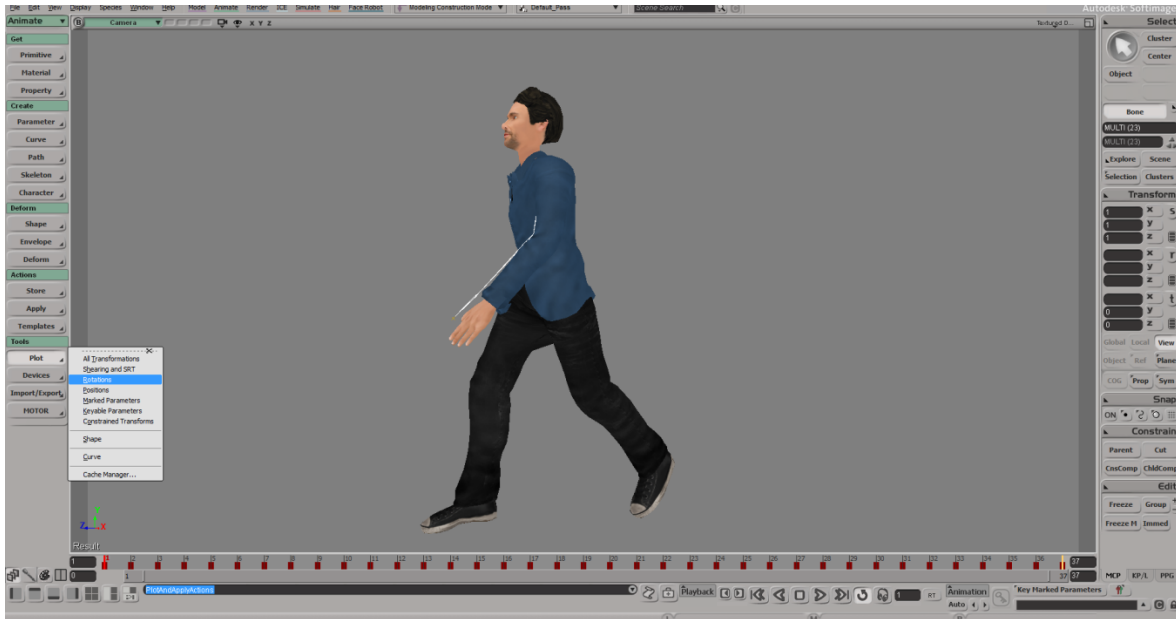


Figura 3.16: Plot rotaciones de huesos.

Ya realizando las indicaciones anteriores para las rotaciones de los huesos, debemos nombrar la acción que estamos realizando, en este caso “Walk” después debemos indicar de que Frame de inicio a que Frame de fin queremos el plot de los huesos.

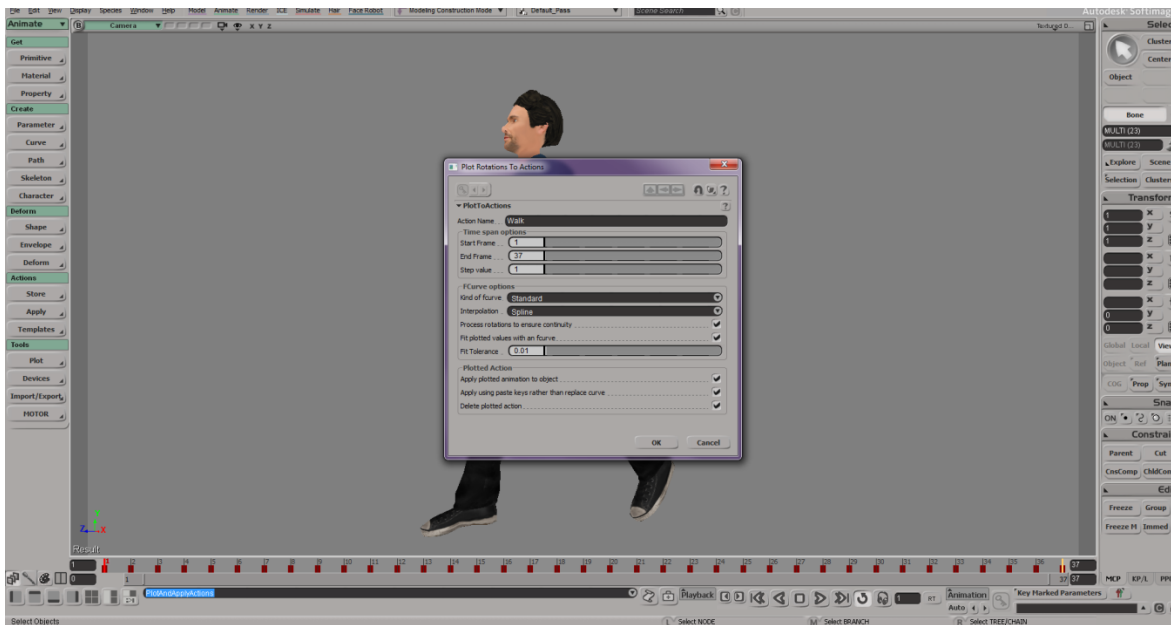


Figura 3.17: Parámetros del plot.

La importancia de realizar el Plot de las rotaciones de los huesos, se debe a que este proceso nos traducirá el movimiento de un archivo .bvh a mostrar las curvas de animación de los huesos, estas curvas de animación servirán en Unity o cualquier programa de videojuegos para hacer la interpolación entre una animación a otra.

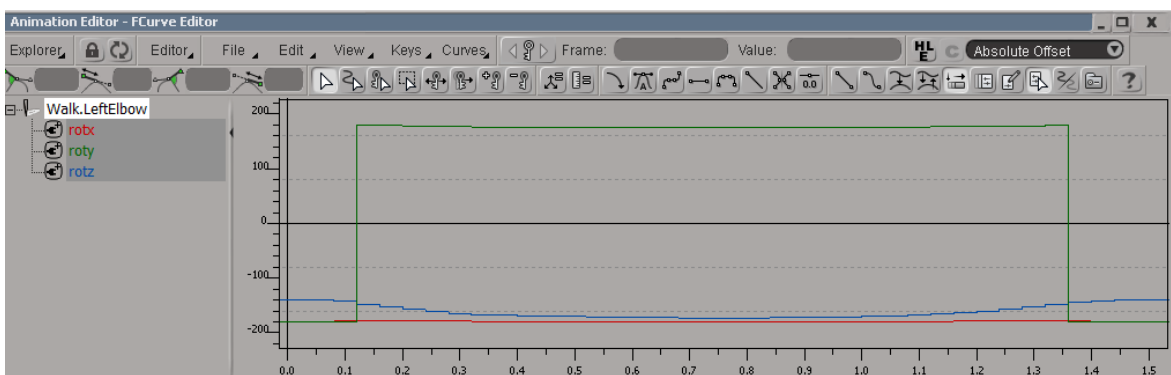


Figura 3.18: Curvas de animación de un hueso.

Con esto ya podemos realizar el Export para crear nuestro archivo .fbx que Unity podrá interpretar para el sistema de locomoción.

Los pasos son clic en el menú “File” -> “Export” -> “Export fbx”, nos pedirá el nombre del archivo .fbx así como la ruta donde lo queremos guardar, después nos mostrara el panel con las opciones para exportar el archivo.fbx, debemos dejar todas las opciones excepto en la pestaña de “Include” las opciones de “Light” y “Cameras” ya que solo nos interesa nuestro personaje y su animación.

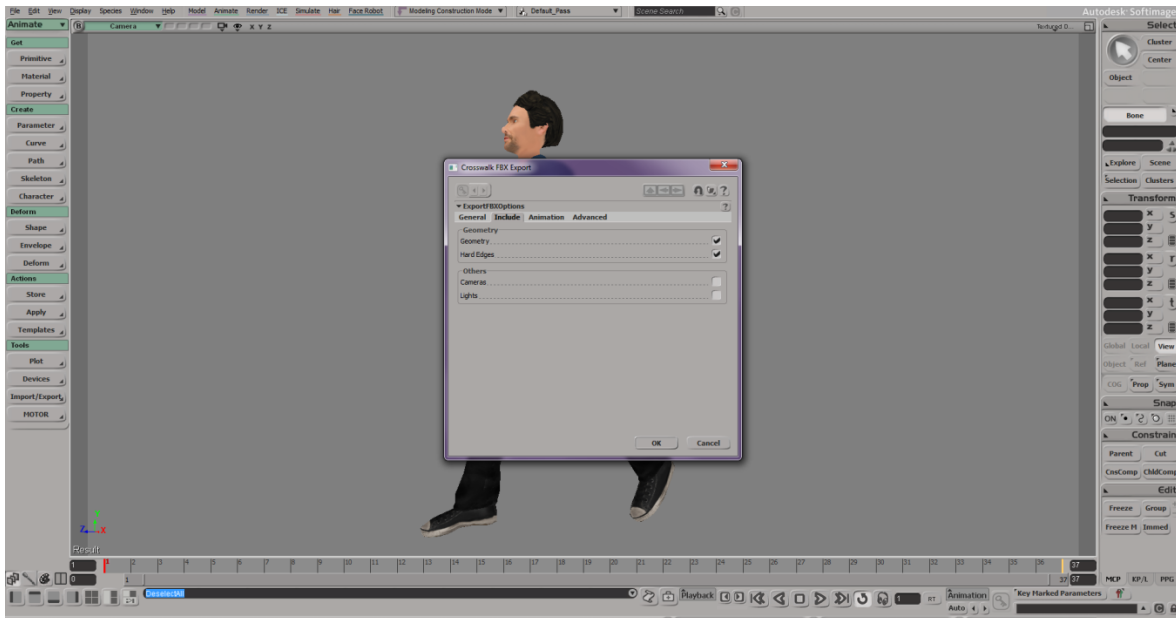


Figura 3.19: Parámetros exportación archivo .fbx.

Nota: Unity nos pide mínimo 2 animaciones para poder integrar un personaje a una escena. Una animación *Idle*, que es cuando el personaje está de pie y no está caminando o realizando otra acción, y la otra animación es un *Walk Cycle* o la mecánica de caminata del personaje.

## Conclusiones del capítulo.

En este capítulo es donde radica la principal aportación de este trabajo de tesis, que es realizar los archivos .fbx, a partir de archivos .bvh, para Unity 3D, ya que la Captura de Movimiento, adjunto a nuestro personaje, no se puede poner directamente en un archivo .fbx, sino que requiere de ciertos pasos para que nuestro personaje pueda ser exportado Unity 3d. Entre estos pasos, los más importantes son el Plot de la rotación de los huesos, así como la configuración de los parámetros del archivo .fbx para que podamos exportar, nuestro personaje, la textura y la animación.

Se decidió realizar el trabajo de animación en Softimage ya que el uso de su interfaz es muy sencillo de manejar. Y tiene todos los elementos para poder realizar las animaciones con una completa compatibilidad con Unity 3D. Otra ventaja es que Softimage cuenta con una versión estudiantil libre para el uso de proyectos.

## Capítulo 4

### Resultados

#### 4.1 Conformación de la Escena

##### 4.1.1 SkyBox

Skyboxes son una envoltura alrededor de toda la escena que más allá de mostrar la variedad de su mundo, con el fin de dar la impresión de escenario complejo en el horizonte. Son una caja de 6 texturas, uno para cada dirección primaria (+/-X, +/-Y, +/-Z).

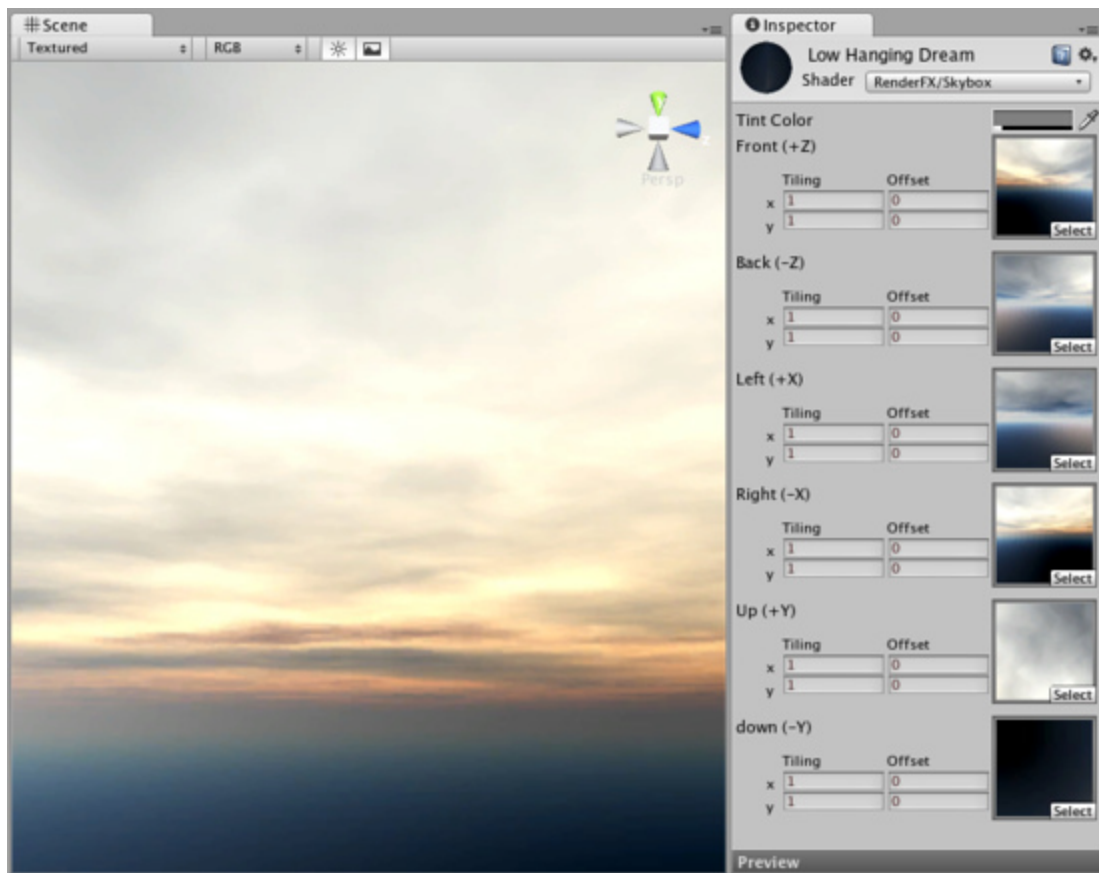


Figura 4.1: Skybox Unity 3D.

El material utilizado para representar el Skybox, que consta de 6 texturas. Este material debe utilizar el sombreado Skybox, y cada una de las texturas se debe asignar a la dirección global adecuada.

### 4.1.2 Uso de Terrenos

Para crear un terreno puede ser creado a partir del terreno-> Crear terreno. Esto agregará un terreno para su proyecto y vistas de jerarquía. El nuevo terreno se verá así en la vista de escena:

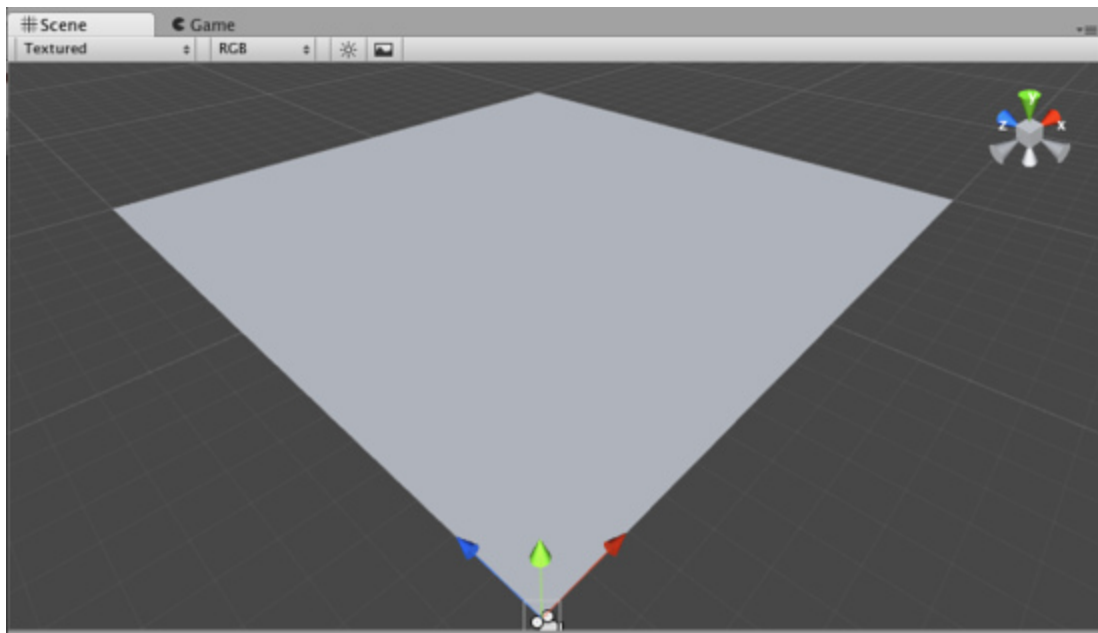


Figura 4.2: Terrain Unity 3D.

Los pinceles sirven para pintar y manipular el terreno. Si desea cambiar la posición de un terreno, se puede modificar su posición Transformar valores en el inspector. Esto le permite mover el terreno alrededor, pero no se puede rotar o escalar ella. Para navegar por el terreno presionamos la letra F, cuando se presiona F, donde se sitúa el puntero del ratón se moverá hacia el centro de la vista de escena. Esto le permite retocar un área, y rápidamente hacer un zoom a un área diferente y cambiar algo más.

Con el terreno seleccionado, se puede ver en el **Inspector** de ver algunas herramientas de edición de terreno. Cada botón rectangular es una herramienta de terreno diferente. Hay herramientas para cambiar la altura, pintar mapas splat, o adjuntar detalles como árboles o rocas.

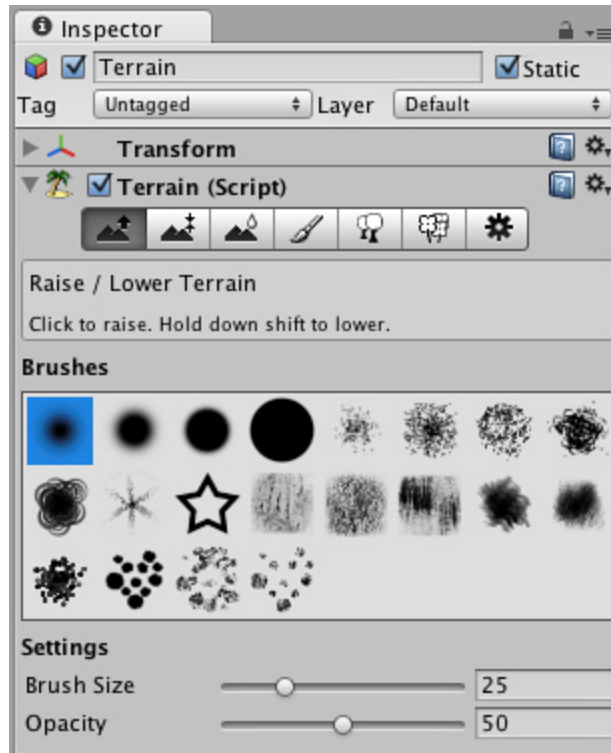


Figura 4.3: Parámetros Terrain Unity 3D.

### 4.1.3 Integración de los Edificios de la Facultad

Una vez que tenemos nuestro archivo .3ds lo que tenemos que hacer para integrarlo a Unity, es simplemente arrastrarlo a la carpeta Assets de nuestro proyecto y con eso es suficiente para integrarlo a nuestra escena. Unity nos muestra nuestros Assets u objetos (ya sea fbx, .3ds, .dae, etc...) en un modo de vista de Icono para facilitar su búsqueda y selección.

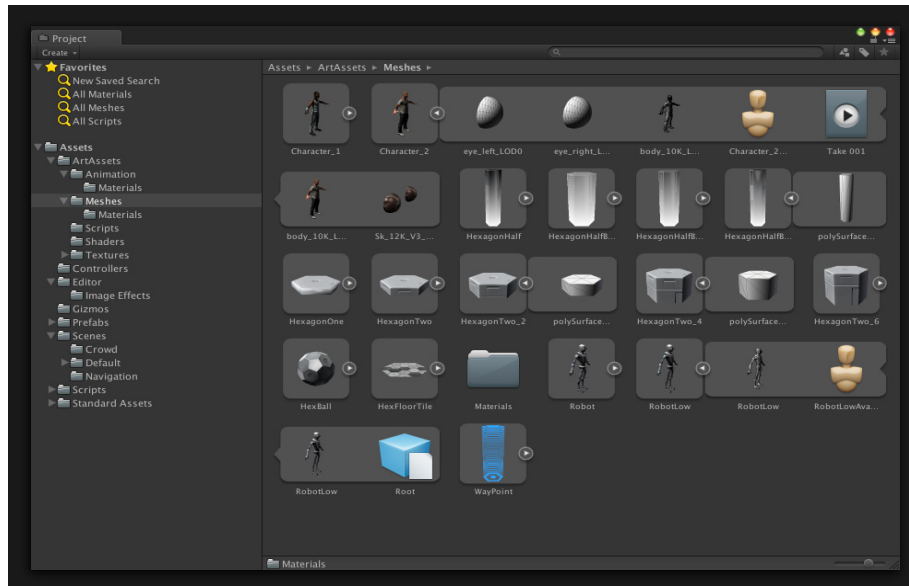


Figura 4.4: Carpeta Assets Objetos .3ds.

#### 4.1.4 Sistema de Locomoción para Unity

Para integrar nuestro personaje a la escena es necesario trabajar con el sistema de locomoción, para esto, Unity combina automáticamente nuestro keyframed o la captura de movimiento (ciclos de caminar y correr) y se ajustan los movimientos de los huesos de las piernas para garantizar que los pies estén correctamente ajustados con el suelo. El sistema puede ajustar animaciones hechas para una velocidad específica y la dirección sobre una superficie plana a cualquier velocidad, la dirección y la curvatura, en cualquier superficie, incluyendo medidas arbitrarias y pendientes. A continuación describimos lo que es necesario para fusionar nuestro personaje, con el sistema de locomoción.

##### 4.1.4.1 Configuración básica de la Escena.

Primero Importamos nuestro archivo .fbx a nuestra carpeta de Assets de nuestro personaje. Se agrega un Character Controller al personaje para manejar las colisiones, también se ajustaron los valores para que quedara alrededor del torso de nuestro personaje y abarcara desde la cabeza hasta los pies. Una vez hecho esto agregamos dos scripts: Normal Character Motor y Platform Character Controller estos están ubicados en Assets > Character Controller Scripts; el primero nos sirve como un manejador de movimientos y el segundo para manejar la entrada.

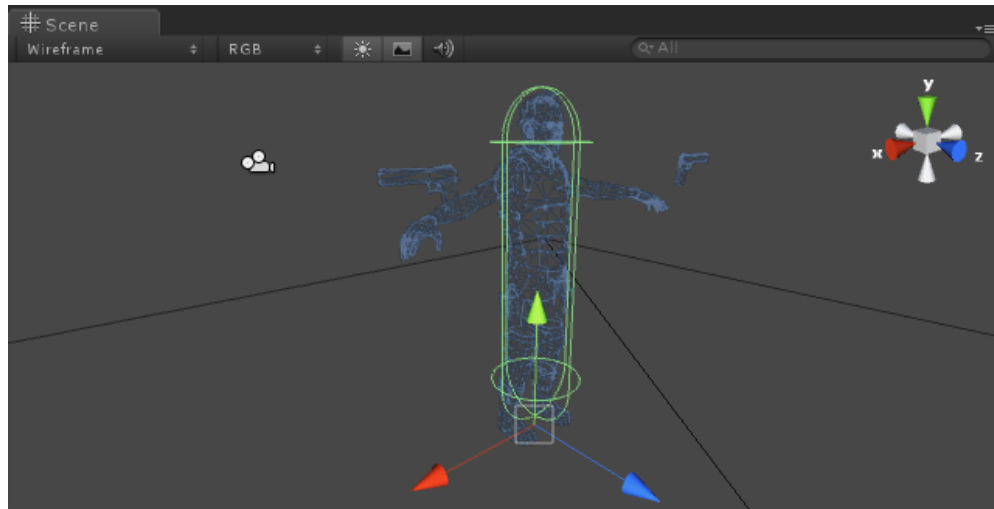


Figura 4.5: Character Controller Unity 3D.

Nos tenemos que asegurar que el **Character Controller** se ajusta al personaje de tal manera que los pies queden por debajo de este. Para comprobar que este bien podemos poner el modo de reproducción; si todo va bien se podrá mover el personaje por toda nuestra superficie de forma estática es decir sin que se dé una fluidez de movimientos, en caso contrario el personaje traspasara la plataforma y caerá.

#### 4.1.4.2 Configuración del sistema de locomoción.

Para configurar el sistema de locomoción primero debemos asegurarnos que existe la opción **Locomotion Initialization** esto lo podemos verificar en la barra de menú en Tools > Locomotion Initialization, esta opción debe aparecer automáticamente por que el script LocomotionEditorClass.cs lo genera por sí solo.

Cuando se utiliza el personaje por primera vez tiene una animación de componentes por defecto (si el personaje no la tiene algo está mal); además se tiene que desactivar la opción de **Play automatically**.

Se agregó un **Leg Controller** al personaje que se encuentra en Assets >Locomotion System > Leg Controller, este debe estar en el mismo objeto que tiene la animación componente.

Una vez agregado el controlador de la pierna éste se tiene que ajustar a la altura del plano de la tierra; una **cruc verde o gizmo** aparece en el nivel de los pies del personaje.

Además se tiene que tomar en cuenta que la altura se mide en unidades antes de aplicar la transformación del carácter. Esto es para asegurarse de que puede utilizar otra posición, rotación y escala del personaje más adelante sin tener que cambiar los controladores de la pierna.

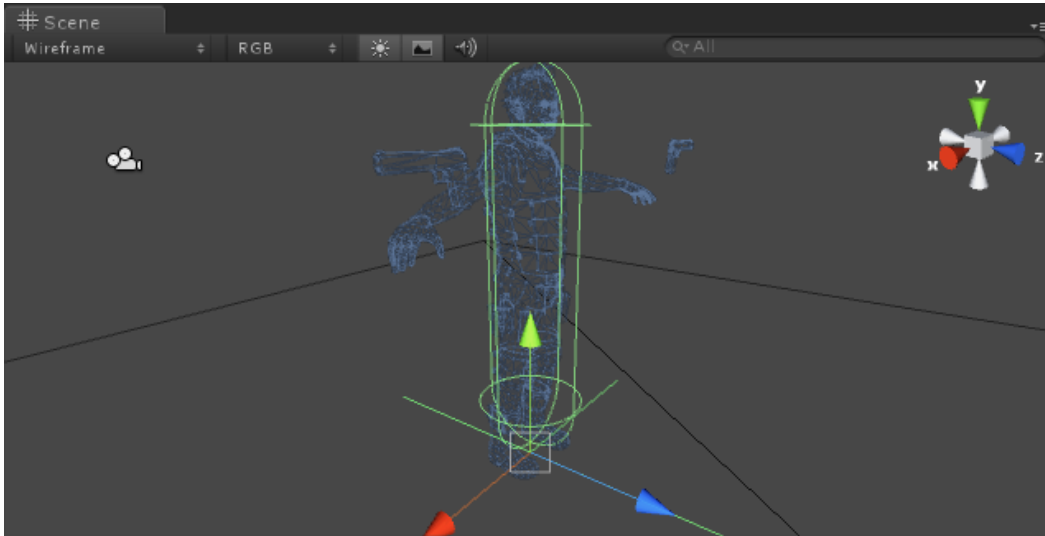


Figura 4.6: Leg Controller Unity 3D.

Una vez hecho esto arrastramos la animación **idle** (idle.fbx) en la variable **Grounded Pose**. Una forma rápida de llegar a las animaciones en el panel del proyecto es hacer clic en el clip de animación en la animación. Para nuestro personaje usaremos idle1.

La raíz del hueso debe ser el hueso que es la raíz de la jerarquía de animación. Por defecto se establece en el hueso principal del hueso de la cadera, si no se coloca manualmente arrastrando el **reference** de nuestro personaje.

Después expandimos la opción de Legs sobre la sección del controlador de las piernas. Esta sección debe contener los datos de configuración para todas las piernas.

Al hacer clic en **Add Leg** se expande una serie de opciones en la cual se ven las propiedades de la pierna y su configuración y se tienen que acomodar de la siguiente forma:

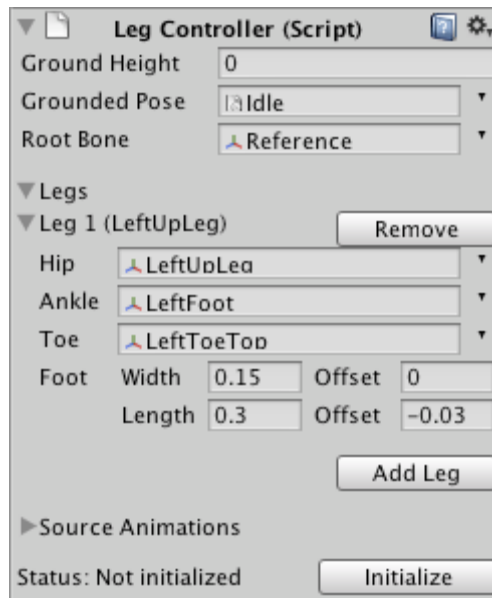


Figura 4.7: Parámetros Leg Controller.

Hips→ Se colocara el hueso que está pegado a la cadera es decir el hueso superior de la pierna.

De acuerdo a la jerarquía del esqueleto del personaje este se encuentra en Character > Reference > Hips >LeftUpLeg

Ankle→Aquí se colocara el hueso del tobillo. La ruta para este hueso se encuentra en Character > Reference > Hips >LegtUpLeg >LeftLeg >LeftFoot.

Se tiene que tomar en cuenta que la altura del tobillo (Ankle) realmente debe sostener cualquier hueso que controle la rotación del pie. Para los seres humanos, que será el tobillo, pero para algunos animales el tobillo está situado bastante alto en la pierna, y el hueso controla directamente el pie (o pata o pezuña). Una vez hecho esto se tiene que ajustar el Ancho, Largo y Compensaciones del pie. Un rectángulo verde gizmo se dibuja alrededor del pie que se puede utilizar para verificar las dimensiones y que la distancia sea correcta. Una vez más las mediciones están en unidades anteriores a la ampliación.

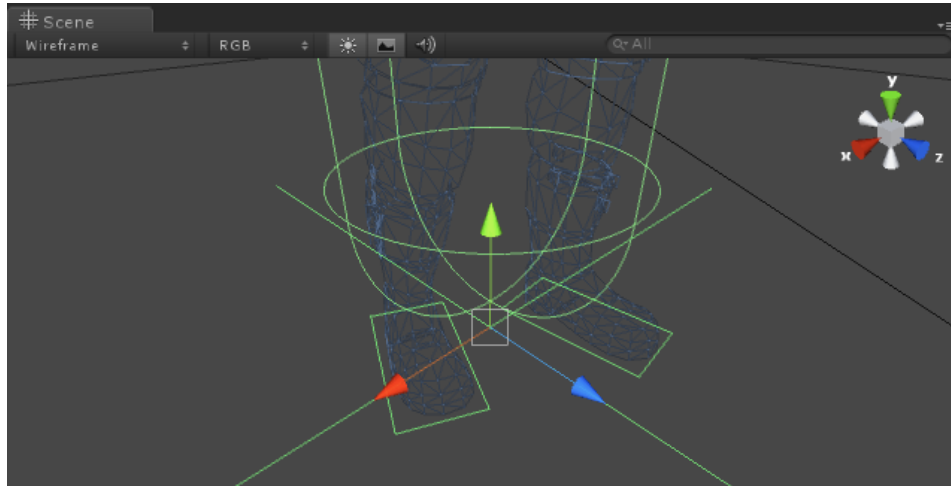


Figura 4.8: Ajuste de ambas piernas.

Cuando se esté satisfecho con la primera pierna, se tiene que hacer clic en el botón **Add Leg** para agregarle las propiedades de las piernas restantes del personaje. Las piernas restantes copian automáticamente los datos del pie **Ancho**, **Largo** y **Compensaciones** de la primera pierna, por lo que ya no es necesario configurar de nuevo los pies si los otros son similares a la primera.

Para agregar más animaciones de nuestro personaje, debemos verificar en el Leg Controller, en la opción de Source Animations que la configuración sea la siguiente:

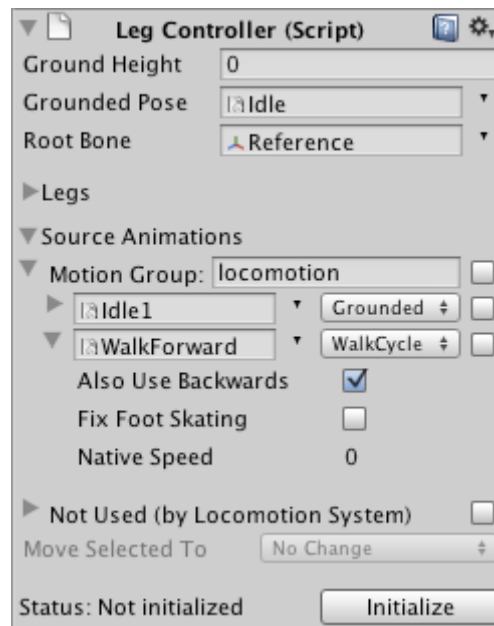


Figura 4.9: Configuración de animaciones.

Ahora agregamos un script **Leg Animator** en el personaje (este se encuentra en Assets > Locomotion System > Leg Animator).

Finalmente damos clic en el botón **Play** y podemos mover nuestro personaje con las flechas de dirección alrededor de nuestra escena.

#### 4.1.5 Luces en nuestra Escena

Luces son una parte esencial de cada escena. Si bien los objetos con sus texturas definen la forma y el aspecto de una escena, las luces definen el color y el estado de ánimo de su entorno 3D. En este tipo de ambientes es muy probable trabajar con más de una luz en la escena. Las luces se pueden utilizar para simular el sol, la luz ardiente fósforo, lámparas de mano, arma de fuego o explosiones, sólo por nombrar algunos ejemplos.

Hay cuatro tipos básicos de luz en la Unidad. Cada tipo puede ser personalizado para adaptarse a sus necesidades.

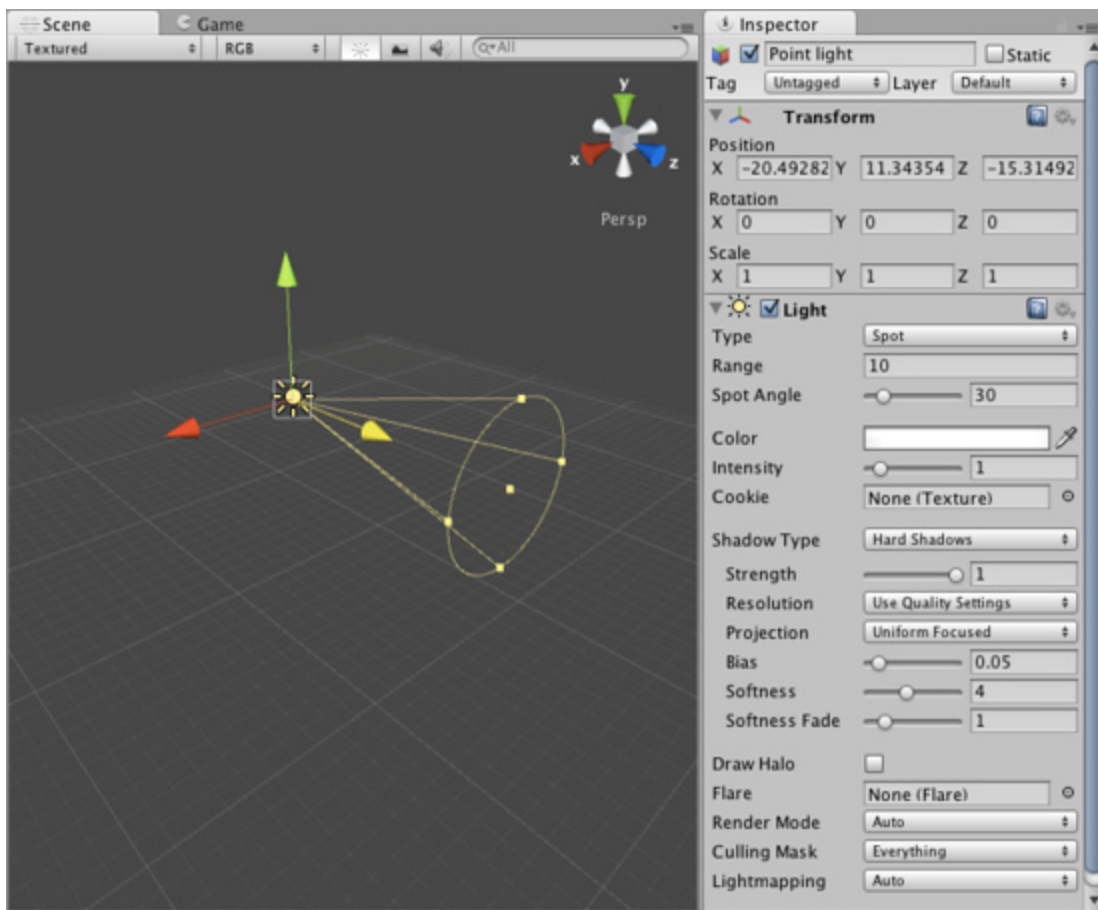


Figura 4.10: Luz tipo punto.

### 4.1.5.1 Luces direccionales

Luces direccionales se utilizan principalmente en las escenas al aire libre para el sol y la luna. La luz afecta a todas las superficies de los objetos en la escena. Son los menos caros en el procesador gráfico.

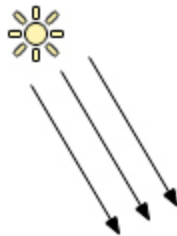


Figura 4.11: Luz direccional.

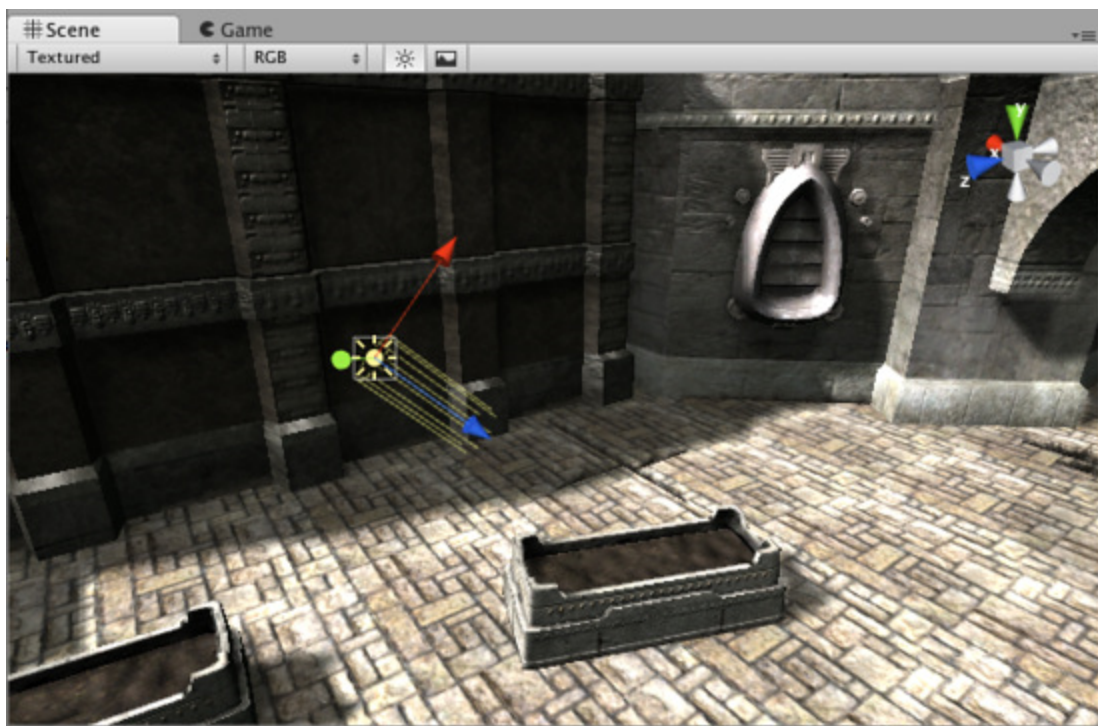


Figura 4.12: Luz direccional al aire libre.

#### 4.1.6 Uso de WoW Character Movement Camera

El WoW Character Movement Camera es un script para el manejo de la cámara para un personaje. Este script, no maneja directamente la vista de la cámara como si la cámara estuviera en la cabeza del personaje, sino que nos permite ver al personaje totalmente, simplemente maneja un constrain o una restricción de posición y rotación esto quiere decir que la cámara realizara el movimiento de traslación como lo haga el personaje. En este caso para el movimiento de rotación de la cámara, la cámara toma como punto de centro al personaje para que el personaje siempre sea enfocado por la cámara.

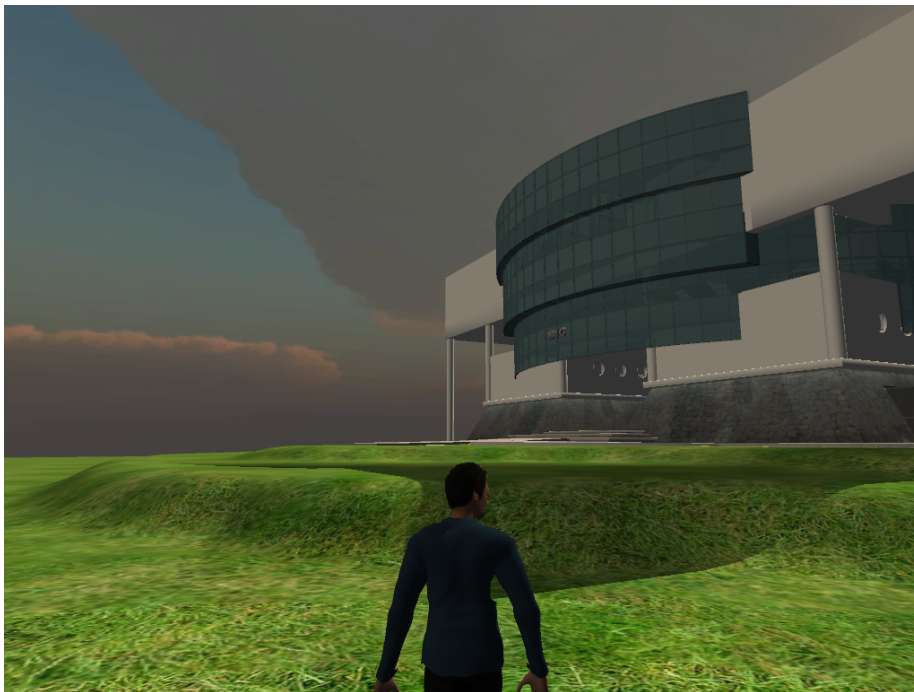


Figura 4.13: WoW Character Movement Camera.

En nuestra aplicación, manteniendo clic izquierdo, podemos rotar la vista de la cámara cumpliendo la restricción de siempre enfocar al personaje pero nos permite ver nuestra escena. Con el botón scroll del mouse o clic de en medio se hace un zoom hacia el personaje.

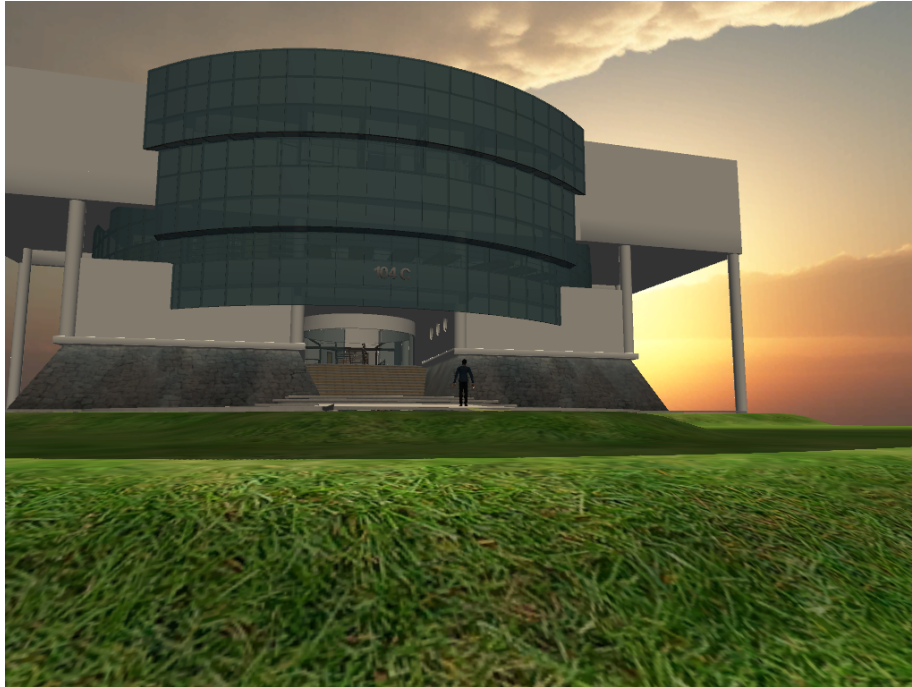


Figura 4.14: Zoom out WoW Camera.

## 4.2 Resultados

A continuación se presenta una serie de imágenes, donde se muestra el resultado de la integración de los elementos mencionados anteriormente, dando como resultado un ambiente virtual en el cual podemos recorrer los espacios de la Facultad de Ciencias de la Computación. La ejecución del ambiente virtual tiene como motor gráfico a Unity Web Player, el cual nos permite colocar nuestra aplicación, en este caso nuestro ambiente virtual, en una página web, dando como ventaja, la posibilidad de que cualquier persona con Internet pueda navegar en nuestro ambiente virtual. El primer par de las imágenes podemos observar el exterior del edificio 140 C, el siguiente par de imágenes muestra el interior del edificio, con su inmueble.



Figura 4.15: Resultado Final I.

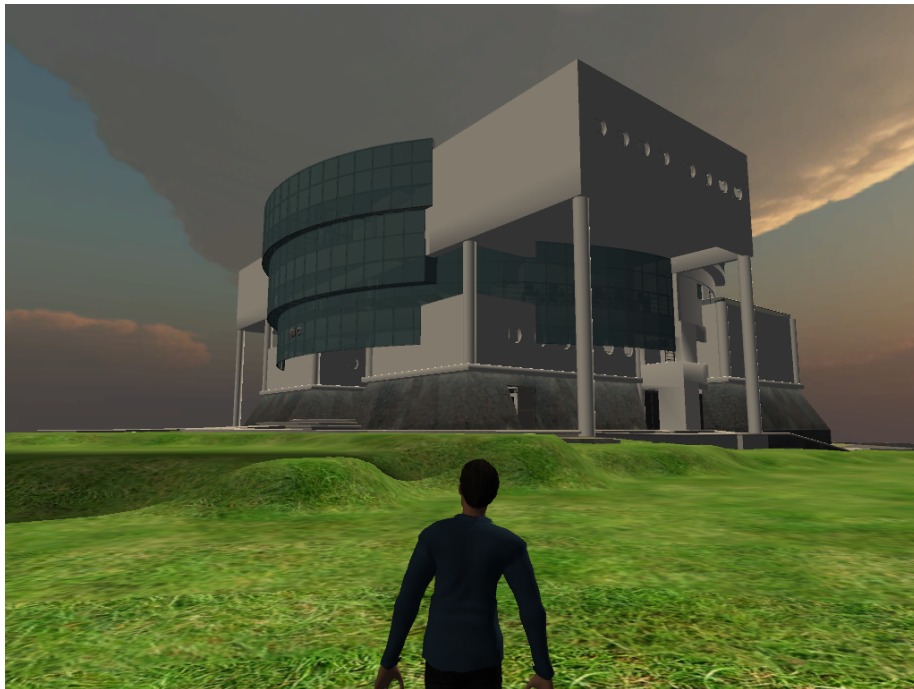


Figura 4.16: Resultado Final II.



Figura 4.17: Resultado Final III.



Figura 4.18: Resultado Final IV.

## Conclusiones del Capítulo

La conclusión de este capítulo, es la más importante ya que en este capítulo es donde se muestra el resultado final de la Facultad Virtual y que engloba todo el trabajo previo en una escena completa, donde está, el personaje y sus ciclos de animación, edificios, el ambiente, interiores de los edificios, el manejo de la cámara, cielo, etc.

En este capítulo, ya se puede ver un ambiente virtual completo, con un personaje principal y que puede navegar en diferentes edificios e interiores de los edificios, tal cual como se propuso en el objetivo general.

## Capítulo 5

### Conclusiones y Trabajo futuro

#### 5.1 Conclusiones

El objetivo principal del trabajo de tesis fue realizar el ambiente virtual de la Facultad de Ciencias de la Computación y poder navegar los espacios de la Facultad como así también integrar varias herramientas, en las que se utilizó AC3D como herramienta de modelado 3D y Unity como motor gráfico; posteriormente, se hace una amplia explicación de la creación de los edificios, el proceso de texturizado, así como también el ajuste del archivo de captura de movimiento al personaje y la creación de un Ambiente Virtual de la Facultad de Ciencias.

Un punto importante, que podemos resaltar, es que una sola herramienta no es suficiente para el desarrollo de un ambiente virtual, como también, una sola persona no es suficiente para desarrollar una ambiente de calidad comercial.

Otra conclusión es sobre la captura de movimiento, que ofrece muchas ventajas, así como desventajas. Pero es un hecho que hoy en día la captura de movimientos es una técnica que utiliza la animación para obtener secuencias en video juegos o en filmaciones.

#### 5.2 Trabajo Futuro

Como trabajos futuros, nos proponemos los siguientes puntos:

- Creación de otros personajes secundarios. Que no solo exista en el Ambiente Virtual un solo personaje sino también personajes secundarios para la interacción con nuestro personaje principal.
- Interacción con diálogos en multimedia, no solo con otros personajes sino con objetos que estén en este Ambiente Virtual, ya sean objetos que muestren información específica o diálogos que nos ayuden a orientarnos en este Ambiente Virtual, así como sucede en la vida real como, museos, parque temáticos, módulos de información, etc.
- Aplicaciones en diferentes plataformas. Poder generar una aplicación no solo para un sistema operativo sino que nuestra Ambiente Virtual pueda funcionar tanto en un PC como en un sistema operativo Android.
- Sistemas de Inmersión. En este puntos nos referimos a que nuestra aplicaciones pueda funcionar con diferentes sistema de inmersión, ya sea un simple game pad, un mouse inalámbrico, un control para Xbox, lentes de video para que nuestras realidad virtual, en nuestro Ambiente Virtual, sea de Inmersión completa y no se quede en una realidad virtual de escritorio.

## BIBLIOGRAFÍA

Aukstakalnis, S., y Blatner, D. 1992. *Silicon Mirage The Art and Science of Virtual Reality*. Berkeley, California: Peachpit Press, Inc.

Ammoura, A. (2001, Enero 02). DIVE-ON: De la Base de Datos a la Realidad Virtual. *ACM Crossroads*. Extraído el 24 de Marzo de 2004 desde <http://www.acm.org/crossroads/espanol/xrds7-3/diveon.html>

Casas, A., y Lucent, G. (2004). *Arquitectura Virtual*. Disponible en: <http://www.enterate.unam.mx/Articulos/2004/Febrero/arqvirtual.htm> [2004, Marzo 19].

EvansSutherland (2002). Manual del simulador de vuelo EP1000CT. Evans & Sutherland Computer Corporation, Salt Lake City, UT.

Las Provincias (2004). *La arquitectura Virtual, última arma de los constructores*. Disponible en: <http://canales.lasprovincias.es/vivienda/noticia459.html> [2004, Abril 10]

Levis, D. (2000). *Realidad virtual y educación*. Disponible en: <http://www.blues.uab.es/home/material/apunts/M02020/lectura1.htm> [2004, Marzo 23].

Lombard, M., & Ditton, T. (2000). Measuring presence: A literature-based approach to the development of a standardized paper-and-pencil instrument. Ponencia presentada en el *Taller Presence 2000*, Marzo 27–28,

Mcconnell, S. (1997). *Desarrollo y Gestión de Proyectos Informáticos*. México, D.F.: McGraw Hill.

Pair,J., Jensen, C. Wilson,J. Hodges, L., Gotz,D., FloresJ. (2000). The NAVE Design and Implementation of a Non-Expensive Immersive Virtual Environment. *Memorias del SIGGRAPH*, Sketches & Applications.

Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., & Carey, T. (1994). *Human-Computer Interaction*. Wokingham: Addison-Wesley.

Perry, L. D.S., Smith, M. C. y Yang, S. (1997, Primavera) Una Investigación sobre Interfaces Actuales de Realidad Virtual. *ACM Crossroads*. Extraído el 24 de Marzo de 2004 desde <http://www.acm.org/crossroads/espanol/xrds3-3/vrhci.html>

Rangel, N. y Santana, Jorge L. (2004). *Áreas de aplicación de la realidad virtual* Extraído el 24 de Marzo de 2004 desde <http://www.utp.ac.pa/seccion/topicos/realidad/cap4.htm>

Schuemie M. y van der Mast C. (1999). Presence: Interacting in VR?. University of Twente. *Memorias del TWLT 15*, Universidad de Twente, pp.213-217.

Slater, M., & Wilbur, S. (1995). A framework for immersive virtual environments (FIVE):Speculations on the role of presence in virtual environments. *Presence*. 6(6).

Universidad EAFIT. (1999). *Ambientes Virtuales Colaborativos aplicados a la Educación*. Medellín, Colombia. Disponible en: <http://arcadia.eafit.edu.co/virtualc/informe/html/informe.html>

<http://docs.unity3d.com/Documentation/Components/terrain-UsingTerrains.html>

<http://docs.unity3d.com/Documentation/Components/class-Skybox.html>

<http://docs.unity3d.com/Documentation/Manual/Lights.html>

Witmer, B.G., & Singer, M.J. (1998). Measuring presence in virtual environments: A presence questionnaire. *Presence*.7(3).