

**Universidad de las Ciencias Informáticas
Facultad 5**



**Trabajo de Diploma para optar por el título de Ingeniero en
Ciencias Informáticas**

**Título: Paseo virtual del docente 5 utilizando el motor de
video-juegos Unity 3D.**

Autor: Mauricio Monné Argilagos.

Tutor: MSc. Liudmila Pupo Peña

La Habana, Junio de 2015
"Año 57 de la Revolución"

“Exitus acta probat”

DECLARACIÓN JURADA DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al Centro de Informática Industrial (CEDIN) de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año 2015.

Firma del autor:

Mauricio Monné Argilagos

Firma del tutor:

MSc. Liudmila Pupo Peña

DATOS DE CONTACTO

Autor

Mauricio Monné Argilagos

Universidad de las Ciencias Informáticas, Ciudad de la Habana, Cuba

Correo: mmonne@estudiantes.uci.cu

Tutor

MSc. Liudmila Pupo Peña

Universidad de las Ciencias Informáticas, Ciudad de la Habana, Cuba

Correo: lpupo@uci.cu

DEDICATORIA

Se la dedico especialmente a mi mamá que siempre ha estado pendiente a mis estudios durante toda mi vida y se ha dedicado y entregado por completo a mi bienestar.

A mi papá y mi hermanito por su preocupación y aliento durante todo el proceso.

Al Dani por la ayuda incalculable que me brindó en el desarrollo de la tesis, sin el nada de esto hubiera sido posible.

A la "COMUNA", en donde he pasado días inolvidables, de los cuales tengo vagos recuerdos de algunos!!!

A Patricio, a mis jimaguas (no pongo ninguno de sus nombres para evitar una posible discusión por el orden de aparición), a mi limoncito Carla, a Luis Mauri mi enano del alma, al Toni el mejor piloto de todos los tiempos, a Edian con quien siempre se aprende algo nuevo, demostrando que un pomo no sirve solo para tomar agua, a mi Panchito que siempre le voy a desear toda la suerte del mundo, a Alberto barcelonista empedernido.

A todos los que pasaron conmigo tantas noches dando DPS en la red.

A mis padrinos que siempre me tiran el salvavidas cuando las cosas se ponen feas.

A Karel que más que oponente fue un tutor, co-tutor y tesista.

A los que ya no están pero siempre serán recordados con cariño.

Y por último pero no menos importante a Amy, Mia, Mitzumi y al insoportable de Polo.

RESUMEN

Los paseos virtuales, son el modo de acercar y facilitar a los usuarios un entorno o espacio concreto con el fin de que lo mostrado sea fiel ejemplo de lo que allí van a encontrar. Los paseos virtuales interactivos añaden además, la posibilidad al visitante de moverse entre los distintos espacios que la componen, así como el detallar los elementos importantes que en él se consideren. En el Centro de Entornos Interactivos 3D (VERTEX), perteneciente a la Universidad de las Ciencias Informáticas, no se cuenta con un entorno virtual interactivo del docente 5 que incluya todas sus características arquitectónicas y a la vez refleje el nivel de las aplicaciones gráficas que se desarrollan en este centro. Es por eso que el presente trabajo tiene como objetivo el desarrollo de un paseo virtual interactivo del docente 5, que permite visitar las áreas exteriores e interiores, incorporando elementos de inteligencia artificial y un nivel gráfico adecuado.

Palabras clave: gráficos por computadoras, entorno virtual, paseo virtual interactivo.

ÍNDICE DE CONTENIDO

INTRODUCCIÓN 1

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA 4

1.1 Paseos Virtuales4

 Realidad virtual4

 Visitas o paseos virtuales4

 Evolución de los paseos virtuales.....5

 Características fundamentales.....6

 Aplicaciones de paseos virtuales7

1.2 Elementos de inteligencia artificial orientados a paseos virtuales8

1.3 Motor de Videojuegos9

1.4 Entorno de desarrollo9

 Metodologías del desarrollo10

 Herramientas y tecnologías12

1.5 Conclusiones parciales del capítulo14

CAPÍTULO 2. CARACTERÍSTICAS Y DISEÑO 15

2.1 Propuesta de solución.....15

2.2 Análisis del dominio18

 Descripción del modelo de dominio18

2.3 Fase de Exploración.....19

 Historias de usuarios19

2.4 Fase de Planificación21

 Estimación de esfuerzo por HU.....22

 Plan de Iteraciones.....22

 Plan de duración de las iteraciones23

 Plan de entrega23

2.5 Fase de Diseño24

 Arquitectura del sistema.....24

 Tarjetas Clase-Responsabilidad-Colaboradores (CRC).....25

2.6 Conclusiones parciales27

CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBAS	28
3.1 Fase de Producción	28
Iteraciones en el desarrollo del producto	28
3.2 Implementación	30
Componentes	31
3.3 <i>Scripts</i>	33
3.4 Pruebas	35
Pruebas de aceptación	35
3.5 Conclusiones parciales	40
CONCLUSIONES GENERALES.....	41
RECOMENDACIONES.....	42
REFERENCIAS BIBLIOGRÁFICAS	43
GLOSARIO DE TÉRMINOS	¡Error! Marcador no definido.
ANEXOS.....	¡Error! Marcador no definido.

ÍNDICE DE TABLAS

Tabla 2 Historia de usuario “Crear el escenario virtual”	20
Tabla 3 Historia de usuario “Desplazamiento del avatar”	20
Tabla 4 Historia de usuario “Elementos de Inteligencia Artificial”	21
Tabla 5 Puntos estimados para las historias de usuario	22
Tabla 6 Plan de duración de las iteraciones.....	23
Tabla 7 Plan de entregas	23
Tabla 8 Tarjeta CRC “Main”	25
Tabla 9 Tarjeta CRC “SceneManager”	25
Tabla 10 Tarjeta CRC “Avatar”	26
Tabla 11 Tarjeta CRC “LocomotionHandler”	26
Tabla 12 Tarjeta CRC “Scenes”	26
Tabla 13 Tarjeta CRC “Actor”	26
Tabla 14 Tarjeta CRC “PerceptionManager”	26
Tabla 15 Historia de usuario correspondiente a la Iteración 1	28
Tabla 16 Primera tarea asociada a la historia de usuario “Crear el escenario virtual”	29
Tabla 17 Primera tarea asociada a la historia de usuario “Crear el escenario virtual”	29
Tabla 18 Historia de usuario correspondiente a la Iteración 2	29
Tabla 19 Tarea asociada a la historia de “Desplazamiento del avatar”	29
Tabla 20 Historia de usuario correspondiente a la Iteración 3	30
Tabla 21 Tarea asociada a la historia de “Elementos de Inteligencia Artificial”	30
Tabla 22 <i>Script</i> de cámara " <i>CameraController</i> "	33
Tabla 23 <i>Script</i> del avatar " <i>CharacterAnimator</i> "	34
Tabla 24 <i>Script</i> del avatar " <i>GraphicUserInterface</i> "	34
Tabla 25 <i>Script</i> de objeto " <i>Open_Door</i> "	34
Tabla 26 <i>Script</i> de avatar de "IA"	34
Tabla 27 Caso de prueba 1 sobre la historia de usuario 1	37
Tabla 28 Caso de prueba 1 sobre la historia de usuario 2	37
Tabla 29 Caso de prueba 2 sobre la historia de usuario 2	37
Tabla 30 Caso de prueba 3 sobre la historia de usuario 2	38
Tabla 31 Caso de prueba 4 sobre la historia de usuario 2	38
Tabla 32 Caso de prueba 5 sobre la historia de usuario 2	38
Tabla 33 Caso de prueba 1 sobre la historia de usuario 3	39
Tabla 34 Caso de prueba 2 sobre la historia de usuario 3	39
Tabla 35 Caso de prueba 3 sobre la historia de usuario 3	40

INTRODUCCIÓN

Con el creciente desarrollo de los gráficos por computadoras la recreación virtual de un espacio o área geográfica de nuestra realidad se ha incrementado considerablemente. Diversos son los ejemplos que podemos encontrar ya sea en aplicaciones de escritorio, video juegos o en la web.

Los paseos virtuales son el modo de acercar y facilitar a los usuarios a un entorno o espacio concreto con el fin de que lo mostrado sea fiel ejemplo de lo que allí van a encontrar. Los paseos virtuales interactivos añaden además, la posibilidad al visitante de moverse entre los distintos espacios que lo componen, así como detallar los elementos importantes que en él se consideren.

Los paseos virtuales deberán tener como objetivo aumentar notablemente la permanencia del usuario, y en consecuencia, su atracción e interés por el lugar fotografiado. Sin embargo todo esto depende de muchos otros factores, por lo que no basta ser solo "paseo virtual" para lograr el objetivo.

En una visita virtual el usuario percibe el espacio esférico con una vista totalmente verosímil y natural, tal como es en la realidad desde cualquier computadora. Este es el gran atractivo, el de convidar al navegante con la posibilidad de realizar una visita virtual al lugar, con la sensación de estar allí. Son una forma de publicidad altamente efectiva y atractiva para los usuarios.

En el Centro de Entornos Interactivos 3D (VERTEX), perteneciente a la Universidad de las Ciencias Informáticas (UCI), se han realizado productos afines a paseos virtuales, ejemplo de ello es el paseo virtual del prado de las esculturas llevado a cabo por profesionales y estudiantes de este centro. Este paseo virtual fue realizado con el motor de video-juegos Unity 3D por las ventajas que presenta para el desarrollo de este tipo de productos. Sin embargo aún no se cuenta con un entorno virtual interactivo del docente 5 que incluya todas sus características arquitectónicas y a la vez refleje el nivel de las aplicaciones gráficas que se desarrollan en este centro.

Por lo anterior se propone resolver el siguiente **problema**: ¿Cómo crear un paseo virtual interactivo del docente 5 utilizando el motor de video-juego Unity 3D?

El **objeto de estudio**, se enmarca en los paseos virtuales. Se define como **campo de acción**, los paseos virtuales interactivos.

Se plantea como **objetivo general** desarrollar un paseo virtual interactivo del docente 5, que permita visitar las áreas exteriores e interiores, incorporando elementos de inteligencia artificial y gráficos de alto nivel.

Tareas

Para el cumplimiento de los objetivos trazados se realizaran las siguientes tareas.

- Realización de un estudio del estado del arte en el desarrollo de paseos virtuales.
- Identificación de las herramientas y lenguajes para el desarrollo de la aplicación.
- Análisis y desarrollo de la documentación ingenieril de la aplicación para la creación de paseos virtuales según la metodología seleccionada.
- Modelación del escenario 3D.
- Implementación de las funcionalidades principales de la Inteligencia Artificial (IA) con la que va a contar el paseo.
- Integración de los modelos en 3D y la Inteligencia Artificial en el editor de video-juegos Unity 3D.
- Realización de pruebas al sistema.

Resultados a obtener

Una aplicación que permita recorrer virtualmente el docente 5, mostrando todos los elementos de su arquitectura exterior e interior de forma interactiva, brindándole al usuario la posibilidad de paseos guiados así como datos e informaciones sobre las lugares donde se encuentra.

Métodos teóricos

Histórico y Lógico: se empleó para la fundamentación y sistematización de los aspectos teóricos observados en el desarrollo de la investigación acerca de los paseos virtuales, la IA y demás elementos vinculados a la temática del trabajo, así como para la comprensión de los antecedentes y su evolución como base para la investigación acerca de los paseos virtuales.

Analítico - Sintético: Durante el proceso de investigación permitió descubrir los elementos esenciales, concepciones y conceptos en torno al objeto de investigación.

Modelación: Se empleó para la elaboración de la aplicación teniendo en cuenta los elementos teóricos y prácticos de los entornos virtuales.

Métodos empíricos

Observación: Se empleó como método de referencia al observar distintos paseos virtuales que sirvieron de material de análisis y comparación para definir las características y elementos fundamentales para el cumplimiento de la propuesta del autor.

Experimento: Este método se aplicó para la validación de los resultados obtenidos al finalizar el proceso de desarrollo mediante pruebas realizadas a la aplicación.

Estructura en capítulos del documento:

En el presente documento se aborda la investigación cuyo diseño metodológico se ha descrito. Su desarrollo comienza con un capítulo que describe la fundamentación teórica de dicha investigación, el cual incluye un estudio del estado del arte y una descripción de las herramientas disponibles y los criterios que se tuvieron en cuenta para el desarrollo de la solución propuesta. A continuación un capítulo donde se abordan las fases de Exploración y Planificación propias de la metodología de desarrollo utilizada en la implementación, los planes de entrega y de duración de las mismas a lo largo del desarrollo del producto y finalmente se realiza el diseño de la solución. Se concluye con un capítulo en el cual se describen los métodos utilizados para validar la solución desarrollada.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

En el presente capítulo se hace una investigación sobre los Paseos Virtuales, se muestra la evolución de los mismos y se exponen sus características fundamentales. Se aborda también cómo se aplica la Inteligencia Artificial (IA) en los paseos virtuales. Se realiza además un estudio de las herramientas y metodologías a utilizar en el proceso de desarrollo de la aplicación.

1.1 Paseos Virtuales

Para abordar el tema de los paseos virtuales se toman como referencia algunos conceptos relacionados a la materia en cuestión para entenderla.

Realidad virtual

Según Sherman y Craig (2003), “la realidad virtual se define como un espacio tridimensional gráfico, táctil, visual o auditivo (también llamado ambiente o “mundo” virtual) generado por computadora, donde los usuarios pueden interactuar y navegar en ese espacio utilizando aparatos especiales” (1).

La Enciclopedia Británica define a la realidad virtual como: La utilización de la modelación y simulación que habilitan a las personas a interactuar con una visión tridimensional de la realidad o a través de sensores ambientales (2).

Para Manetta C. y R. Blade (1995) “la realidad virtual es un sistema de computación usado para crear un mundo artificial en el cual el usuario tiene la impresión de estar en él, posee la habilidad de navegar y manipular objetos en él” (3).

Visitas o paseos virtuales

Se conoce como visita virtual a la simulación de un espacio real, esta se compone de dos o varias imágenes dimensionales unidas mediante un *software* especial, creando una imagen interactiva que el concurrente puede controlar mediante el ordenador, visualizándolo en 360°. Lo que permite decir que una visita virtual es lo más parecido a estar realmente en el sitio. Un paseo virtual es una simulación de una localidad existente (4).

Evolución de los paseos virtuales

Para entender que es un paseo virtual se puede dejar de mencionar la fuente del cual el mismo nace, se estaría hablando de la Realidad Virtual (VR). El concepto de realidad virtual surgió en 1965, cuando en su artículo Ivan Sutherland dice que “La pantalla es una ventana a través de la cual uno ve un mundo virtual. El desafío es hacer que ese mundo se vea real, actúe real, suene real, se sienta real”. Sería él, el creador del primer casco visor de realidad virtual utilizando tubos de rayos catódicos (uno para cada ojo) y de un sistema mecánico de seguimiento. Posteriormente en 1968 junto con David Evans se dieron los primeros pasos en la creación de los mundos o paseos virtuales de la actualidad al crear el primer generador de escenarios con imágenes tridimensionales, datos almacenados y aceleradores.

Tan solo un año después, en 1969, Myron Krueger creó el denominado “*Artificial Reality*” que permitía la interacción con elementos creados virtualmente (5).

En la década de los setenta, los gráficos generados por ordenador habían reemplazado los videos en los simuladores. Estos vuelos simulados operaban en tiempo real, aunque los gráficos eran bastante primitivos. En 1979, los militares empezaron a experimentar con cascos de simulación. A principios de los ochenta, una gran mejora en el *software*, *hardware* y las plataformas de movimiento, permitían a los pilotos navegar por detallados mundos virtuales.

Naturalmente la industria militar no era la única interesada en los gráficos por ordenador, la industria del entretenimiento también empezó a utilizar esta tecnología, alcanzando grandes éxitos.

El guante virtual o “*dataglove*” fue una de las variantes en el mundo de la realidad virtual, consistía en una interfaz ordenador que detectaba movimientos con la mano. Este fue creado para producir música haciendo gestos a un sintetizador musical. Una división de la Nasa es uno de los primeros clientes que utilizan este dispositivo para sus experimentos con entornos o paseos virtuales (5).

A principios de los ochenta, Andy Lippman junto con un grupo de investigadores, desarrollaron el primer mapa interactivo virtual de la ciudad de Aspen, Colorado. La grabación fue realizada por medio de cuatro cámaras, tomando una foto cada tres metro y las reproducían a 30 fotogramas por segundo, simulando una velocidad de 330 km/h que más tarde se reduciría a 110 km/h.

En 1991, la compañía W. Industries desarrollaron los "Virtuality". El equipo incluía unos cascos y gafas de visionado. En este mismo periodo aparecerán modelos emulando cabinas de vuelo o conducción. En este mismo año, sale el primer programa destinado a los usuarios para la confección de ambientes virtuales 3D. También se estrena un programa infantil de televisión llamado "El rescate del talismán" en el cual los concursantes debían guiar a un compañero con los ojos vendados por unos escenarios virtuales (5).

En 1995, aparece la primera formulación del VRML (del inglés *Virtual Reality Modeling Language*. "Lenguaje para Modelado de Realidad Virtual").

En 1997, se desarrolla para la US Army's STRICOM un dispositivo que permite caminar, correr y moverse en un reducido espacio en todas las direcciones posibilitando experimentar el movimiento real en un cabina.

En 2003, se crea el famoso mundo virtual en 3D "Second Life" donde por medio de un programa pc, los usuarios o residentes, pueden moverse por él, relacionarse, modificar su entorno y participar en su economía (5).

Características fundamentales

- Conformado por fotografías consecutivas de alta dimensión, enfatizadas en la calidad de los colores de la imagen, sus contrastes, enfoques y profundidades, las cuales después de realizado un tratamiento se adjuntan conformando una panorámica.
- Visión de la realidad con un gran flujo de detalles de todo el entorno.
- Reducen el miedo de los usuarios o clientes a lo desconocido.
- Recorrido libre por el entorno virtual.
- Transmiten con gran realismo el espacio representado, evidenciando sus características reales.
- Permiten al usuario la inmersión en un espacio.
- Permiten la interactividad del usuario con el contenido representado, brindándole la posibilidad de hacer exploraciones, manipular las estancias, objetos, edificaciones, entre otras.

Aplicaciones de paseos virtuales

Con el creciente desarrollo de las tecnologías y los productos de la realidad virtual con avanzadas herramientas que permiten su desarrollo continuo, detallado e interactivo; en varias esferas de la sociedad actual se ponen en práctica estas aplicaciones, trayendo consigo un gran impacto y resultados positivos. A continuación se describen algunas de las principales aplicaciones de los paseos virtuales y su utilidad en la actualidad.

○ **Medicina**

Se utiliza en la psicología para realizar tratamientos de fobias y traumas. Este tratamiento se basa en un sillón que está colocado sobre una plataforma en movimiento donde el paciente usa un casco con unos auriculares, en este espacio se enfrenta a través de una simulación tridimensional al problema que le causa fobia, como miedo a los insectos, a las alturas, a volar, entre otros.

○ **Educación**

Museos/planetarios: Estos centros realizan exposiciones virtuales donde se pueden hacer recorridos por templos antiguos, palacios, galaxias, aprender de diversas áreas de conocimiento, entre otras. En algunos de los proyectos realizados en los centros, se experimenta con situaciones más cotidianas o con las que los visitantes pueden identificarse. Por ejemplo se puede diseñar una montaña rusa y posteriormente experimentar el viaje como si físicamente se estuviera en la montaña, así, mientras se disfruta del viaje se puede aprender de leyes de física. Otro de los enfoques que se le da a la realidad virtual, es el de experimentar visitas virtuales a lugares o edificaciones antiguas que por alguna razón no están disponibles al usuario (destrucción, restauración).

○ **Arquitectura**

Arquitectura Virtual: sirve para generar espacios que solamente se pueden visitar digitalmente, entre los que destacan los Museos Virtuales y *campus* universitarios

Patrimonio Arquitectónico Virtual: que se orienta al rescate y recuperación, en el medio virtual, de aquellas edificaciones desaparecidas o en proceso de serlo.

Modelado virtual de diseños de casas, edificios, planos y maquetas elaborados en modelos tridimensionales, donde se puede contemplar de manera real los diseños, inclusive adentrarse en ellos y poder recorrerlos, dando como consecuencia una visión más clara de las ideas que se tratan de expresar.

- **Ocio**

En la industria del entretenimiento está presente en los videojuegos 3D, donde el usuario recorre virtualmente diversos mundos, esta es la forma más conocida por los clientes.

- **Ejército y Fuerzas de seguridad**

Simuladores de vuelo: son hoy en día una herramienta fundamental para el entrenamiento de los pilotos. Son sistemas muy sofisticados y costosos que incorporan todo tipo de interfaces para simular las situaciones reales dentro de un avión, así como las distintas maniobras.

Entrenamiento de soldados: los ejércitos de muchos países usan los videojuegos para crear simulaciones virtuales de entrenamiento para sus soldados (además de las de vuelo antes mencionadas) y crear estrategias o planes de guerra. El punto de vista es siempre en primera persona, un recurso que utilizan algunos populares juegos de guerra para ser más realistas. (Con esta se simulan perfectamente situaciones reales que pueden ocurrir en el campo de combate).

1.2 Elementos de inteligencia artificial orientados a paseos virtuales

En la actualidad la inteligencia artificial ha emergido como una tecnología práctica con exitosas aplicaciones en muchos campos, desde áreas de propósito general como la percepción o el razonamiento, hasta áreas específicas como ingeniería del conocimiento, planificación, juegos y otros. Por ello es usual que investigadores de otros campos vayan poco a poco introduciéndose en la “IA”, un área que cuenta con técnicas y herramientas que les permiten sistematizar y automatizar el trabajo que ha ocupado gran parte de sus vidas. Por otro lado, la “IA” es aplicable a aquellas áreas relacionadas con las tareas del intelecto humano: esto hace que sea un campo genuinamente universal, y que los investigadores tengan un amplio rango de áreas donde aplicar sus metodologías (6).

La aplicación de la inteligencia artificial en un paseo virtual es fundamental para aportar al usuario una experiencia más vivida del lugar que está visitando. Ejemplo de estos elementos son avatares artificiales

recorriendo el entorno de forma autónoma, voces, sonidos ambiente, colisiones y autocorrección de movimiento.

1.3 Motor de Videojuegos

Unity 3D: es un motor de gráficos 3D para PC (*Personal Computer* o Computadora Personal) y Mac que viene empaquetado como una herramienta para crear juegos, aplicaciones interactivas, visualizaciones y animaciones en 3D y tiempo real. Unity puede implementar contenido para múltiples plataformas como PC, Mac, Nintendo Wii e iPhone. El motor también puede publicar juegos basados en web, visualizando las aplicaciones mediante su plugin “*Unity Web Player*” (7).

Unity 3D provee de un editor visual muy útil y completo, brinda diversas facilidades para importar modelos 3D, texturas, sonidos, entre otros, para después ir trabajando con ellos. Además incluye la herramienta de desarrollo MonoDevelop con la que se puede crear scripts en JavaScript, C# y un dialecto de Python llamado Boo, el cual extiende la funcionalidad del editor (8).

Unity es un ecosistema de desarrollo de juegos, un poderoso motor de renderizado totalmente integrado con un conjunto completo de herramientas intuitivas y flujos de trabajo rápidos para crear contenido 3D interactivo; publicación multiplataforma sencilla y activos de calidad. Para desarrolladores independientes y estudios, el ecosistema democratizador de Unity disminuye el tiempo y los costos necesarios para crear juegos y entornos virtuales (9).

Unity 3D fue escogido para el desarrollo de esta aplicación por todas estas características y funcionalidades que brinda, además de ser el motor gráfico con que el centro desarrolla sus productos.

1.4 Entorno de desarrollo

Para el desarrollo del paseo virtual que se propone en el presente trabajo, se seleccionó la metodología que guiará el proceso de desarrollo, así como un conjunto de herramientas y tecnologías a utilizar. A continuación se presentan sus principales características y la justificación de su selección.

Metodologías del desarrollo

Para la selección de la metodología que guiará el proceso de desarrollo del producto, se realizó un estudio de las diferentes metodologías existentes con el objetivo de seleccionar la más apropiada para el desarrollo de esta investigación.

Una metodología de desarrollo de *software* es una serie de técnicas, herramientas y procedimientos que deben seguirse para desarrollar un *software* (10). Son utilizadas principalmente para establecer las etapas de un proyecto, decidir que tareas se llevarán a cabo en cada etapa y para controlar y gestionar un proyecto. Sirven de sostén para la creación de *software*.

Según Booch, Jacobson y Rumbaugh los padres de la metodología RUP¹, una metodología de ingeniería del *software* es un proceso para producir *software* de forma organizada, empleando una colección de técnicas y convenciones de notación predefinidas (11).

Por otra parte Mario Piattini la conceptualiza como un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar un nuevo *software* (12).

Las características de las metodologías ágiles las convierten en candidatas perfectas para el desarrollo de la presente investigación, debido a que se cuenta con un proyecto de pequeña escala y un tiempo de desarrollo corto. A continuación se hace un análisis de las diferentes metodologías ágiles, que más se utilizan en la actualidad, con el objetivo de escoger la metodología que guiará el proceso de desarrollo de la solución propuesta.

Extreme Programming (XP)

Es una metodología de desarrollo de *software* creada por Kent Beck, centrada principalmente en la simplicidad, la comunicación y el reciclado continuo de código, preocupándose por el aprendizaje de los desarrolladores y propiciando la existencia de un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. Esta metodología es adecuada para proyectos con requisitos imprecisos y cambiantes, donde existe un alto riesgo técnico (13).

¹ Proceso Unificado Racional, del inglés *Rational Unified Process*.

XP se basa fundamentalmente en ser ligera, cercana al desarrollo, basada en Historias de Usuario (HU), buena comunicación con el cliente, el código pertenece a todos, programación por parejas y pruebas como base de la funcionalidad. Las HU están descompuestas en tareas de programación y asignadas a los programadores para ser implementadas durante una iteración (14).

Características fundamentales de XP:

- En vez de planificar, analizar y diseñar para el futuro distante, realizar estas actividades durante el proceso de desarrollo.
- Metodología basada en prueba y error.
- Fundamentada en Valores y Prácticas.
- Pocos artefactos.
- Pocos roles.
- Está orientada hacia quien produce y usa el *software*.
- Se aplica en grupos pequeños y muy integrados, menos de 10 integrantes.
- Combina las que han demostrado ser las mejores prácticas para desarrollar *software*, y las lleva al extremo.
- Los programadores trabajan por parejas (dos delante del mismo ordenador) y se intercambian las parejas con frecuencia (un cambio diario).
- Pruebas continuas, frecuentemente repetidas y automatizadas.
- El código es compartido entre todo el equipo.

Metodología seleccionada para el desarrollo del componente

XP es la metodología seleccionada por que sus características favorecen el desarrollo de la solución. Entre las cuales están: la realimentación continua entre el cliente y el equipo de desarrollo, la producción de versiones del sistema de manera rápida, el diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto y además, soporta el cambio de requerimientos.

Se puede considerar la programación extrema como la adopción de las mejores metodologías de desarrollo de acuerdo a lo que se pretende llevar a cabo con el proyecto, y aplicarlo de manera dinámica durante el ciclo de vida del *software*.

El ciclo de vida ideal al emplear la metodología XP consta de seis fases (15), de las cuales se abordarán en el presente trabajo las primeras cuatro fases: **exploración, planificación, iteración y producción**, las cuales son descritas a continuación:

- **Exploración:** en esta fase se realiza la reunión con los clientes y en la cual son recogidas las llamadas historias de usuarios que determinan los requerimientos a implementar por los programadores. Además se utiliza este período para la familiarización por parte del equipo desarrollador con las tecnologías, herramientas y prácticas a utilizarse por el proyecto.
- **Planificación de la Entrega (*release*):** en esta fase son establecidas por el cliente las prioridades para los requerimientos recogidos, realizándose una estimación del esfuerzo necesario en cada una de ellas. En este período son plasmadas las iteraciones a realizarse a través de un Plan de Entrega.
- **Iteraciones:** fase utilizada para desarrollar las iteraciones definidas en el Plan de Entrega, donde se escogen las arquitecturas afines a cada iteración, así como las iteraciones que fuerzan la creación de dichas arquitecturas.
- **Producción:** fase utilizada para la realización de revisiones de rendimiento, pruebas y para la inclusión de nuevas características al producto, antes de ser trasladado al entorno del cliente.

Herramientas y tecnologías

Las herramientas y tecnologías que se seleccionaron para llevar a cabo la realización del entorno virtual se describen a continuación.

Lenguaje de Modelado

El Lenguaje de Modelado Unificado (UML, por sus siglas en inglés de *Unified Modeling Language*) se define como un lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de *software*. Es un sistema de notación destinado a los sistemas de modelado que utilizan conceptos orientados a objetos. Es el estándar mundial que utilizan los desarrolladores, autores y proveedores de Herramientas para Ingeniería de Software Asistida por Computación (CASE) (16).

Herramienta CASE

Visual Paradigm: es una herramienta para la Ingeniería de Software Asistida por Computación (CASE). Esta ha sido concebida para soportar el ciclo de vida completo del proceso de desarrollo del *software* a través de la representación de todo tipo de diagramas. Fue diseñado para una amplia gama de usuarios interesados en la construcción de sistemas de *software* de forma fiable a través de la utilización de un enfoque Orientado a Objetos. Esta herramienta mediante la utilización de UML colabora en gran medida con el desarrollo de la aplicación desde la planificación, el análisis, el diseño y la generación de la documentación (17).

Selección entre Blender y Autodesk 3ds Max.

Blender

Blender es un programa de modelado en 3D, apoyado por varias herramientas, es multiplataforma. Está orientado a artistas y profesionales del diseño y multimedia, puede ser usado para crear, visualizaciones 3D estáticas o vídeos de alta calidad. También incorpora un motor de 3D en tiempo real el cual permite la creación de contenido tridimensional interactivo que puede ser reproducido de forma independiente. Es compatible con todas las versiones de Windows, Mac OS X, GNU/Linux, Solaris, FreeBSD e IRIX (18).

Autodesk 3ds Max

Autodesk 3ds Max y Autodesk 3ds Max Design proporcionan potentes herramientas integradas de modelado, animación y renderización en 3D que permiten a los artistas y los diseñadores dedicar más energía a la creatividad en lugar de a las dificultades técnicas. Aunque ambos productos comparten la tecnología principal, uno ofrece herramientas especializadas a los desarrolladores de juegos, creadores de efectos visuales, diseñadores de gráficos de movimiento y otros profesionales de la creatividad que trabajan en el diseño de medios, mientras que el otro está concebido específicamente para los arquitectos, diseñadores, ingenieros y especialistas en visualización (19).

Se decide utilizar Autodesk 3ds Max para la creación de los modelos por las facilidades en el trabajo de las coordenadas de mapeado y la curva de aprendizaje es menor con respecto a Blender. Además ya se había modelado varias estructuras haciendo uso de la herramienta.

1.5 Conclusiones parciales del capítulo

- Se realizó un estudio de los conceptos fundamentales relacionados con el marco teórico de la presente investigación y se determinaron los aspectos que más aportan al producto.
- Se seleccionó la metodología XP para guiar el proceso de desarrollo de *software* debido a los cambios específicos que propone el cliente.
- Al utilizar Autodesk 3ds Max se permite utilizar los modelos previos y exportarlos a Unity 3D sin pérdida de información.

CAPÍTULO 2. CARACTERÍSTICAS Y DISEÑO

En el presente capítulo se brinda una descripción del proceso de creación del producto que abarca el presente trabajo. Además se abordan las fases de Exploración y Planificación propias de la metodología de desarrollo utilizada en la implementación. En las cuales se exponen las historias de usuario seccionadas en iteraciones, los planes de entrega y de duración de las mismas a lo largo del desarrollo del producto. Finalmente se realiza el diseño de la solución.

2.1 Propuesta de solución

Para el cumplimiento del objetivo del trabajo, se propone como propuesta de solución la creación de un paseo virtual interactivo del docente 5 el cual tiene áreas exteriores e interiores además de objetos dinámicos (avatares, carteles interactivos y puertas) y estáticos que aportan sensación de realismo al entorno con que el usuario interactúa. Las áreas poseen detalles y elementos arquitectónicos del docente con similitud.

Para la realización del modelado de los objetos que componen las escenas existen dificultades para lograr un alto nivel de semejanza con la arquitectura del docente. Por lo que es necesario tomar fotos del edificio, tanto a su entorno exterior como sus áreas interiores, de estas fotos se obtienen las texturas reales que posteriormente se aplican al texturizado de los modelos en 3D.



Figura 1 Texturas de paredes

En la figura 1 se aprecia un ejemplo de textura de una pared dentro del docente. Para lograr el resultado se procede a tomar una foto de la pared real. Luego se le asigna un material con la textura al modelo y se le definen las coordenadas de mapeado con alineación en forma de cubo al objeto para que abarque toda la pared de manera uniforme.



Figura 2 Texturas del techo

En la figura 2 se aprecia un ejemplo de textura en el techo del docente, pero en este caso se aplican coordenadas de mapeado con alineación de plano.

Al terminar el trabajo con los modelos, estos se exportan al formato .FBX, el cual es compatible con el motor de video-juegos Unity 3D. Una vez exportados los modelos se procede a la etapa de ensamblado de las escenas que tiene el paseo virtual. En este proceso se aprecia que no se pierde la información de los objetos modelados, aunque si se detectan problemas con el tamaño de varios modelos en el entorno, los cuales se proceden a escalar.



Figura 3 Modelos del paseo virtual

En la figura 3 se aprecian que los modelos están escalados correctamente. Inicialmente los monitores tenían mayor tamaño que las mesas, por lo que se procede al ajuste.

Para el mejoramiento visual del producto se incluyeron iluminaciones en varias áreas del paseo virtual. Estas iluminaciones son apreciadas durante toda la navegación del usuario dentro del entorno. También se le incluye diferentes sonidos en varios lugares del paseo, los cuales brindan la sensación de sonidos ambientales realistas.



Figura 4 Iluminación del interior del laboratorio

En la figura 4 se muestra un ejemplo de iluminación de un laboratorio evidenciándose los efectos de sombras y oclusión ambiental.

2.2 Análisis del dominio

La metodología XP no propone un modelo de dominio, aunque distintos autores recomiendan mantener diferentes artefactos específicos que pueden ser útiles durante el proceso de desarrollo para visualizar el progreso del sistema, siempre y cuando el tiempo y esfuerzo dedicado a mantenerlos sea menor que el dedicado al desarrollo. Criterio por el cual se decide mantener el modelo de dominio para una mejor descripción y entendimiento del sistema. En la Figura 1 se muestra el modelo de dominio correspondiente al entorno de trabajo en que se realiza la creación de paseo virtual que se lleva a cabo en el presente trabajo, reflejando cada uno de los elementos que conforman el proceso.

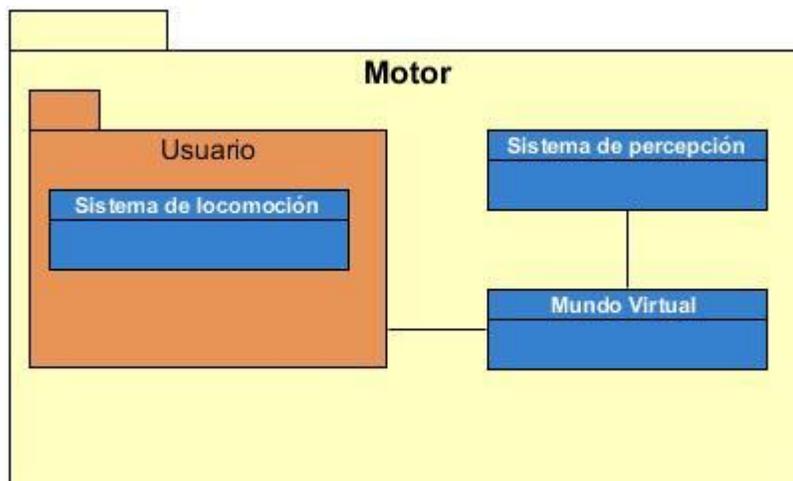


Figura 5 Modelo de dominio

Descripción del modelo de dominio

- **Módulo principal “Motor”:** es el que se encarga de crear y ejecutar cada objeto virtual a partir de archivos, que contienen el comportamiento de los mismos.
- **Mundo Virtual:** controla todos los elementos válidos dentro de la lógica del entorno virtual.
- **Usuario:** representa al avatar con sus características propias.
- **Sistema de percepción:** brinda al agente inteligente la posibilidad de interactuar con los diferentes elementos presentes a su alrededor.

- **Sistema de Locomoción:** es el encargado de realizar el movimiento físico del agente en el entorno virtual.

2.3 Fase de Exploración

En esta fase, se define el alcance del proyecto, además de realizar la entrevista con los clientes, definiéndose las llamadas Historias de Usuarios (HU) y con ellas la determinación de los requisitos planteados por el cliente, así como la que el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto (20).

En esta fase, el cliente define lo que necesita mediante la redacción de sencillas historias de usuarios. Los programadores estiman los tiempos de desarrollo sobre la base de esta información.

Historias de usuarios

La definición de las historias de usuario permite especificar los requerimientos del *software*. En ellas se describen brevemente las características con las que el producto debe contar, o sea, los requisitos planteados por el cliente. Su utilización debe ser flexible y lo más comprensible posible, permitiendo su implementación por los programadores en poco tiempo (20).

La respuesta a las HU debe estar provista de metas o tareas de programación para el desarrollador, además de que cada historia terminada debe contar con una serie de pruebas de unidad que aseguren el correcto funcionamiento de los componentes. Una vez terminada cada historia, el sistema debe ser presentado al cliente para la realización de pruebas de aceptación de la funcionalidad implementada. Además, al reunirse un número suficiente de funcionalidades válidas de la aplicación, se produce una liberación del producto, la cual es una versión funcional de la aplicación que aporta valor al negocio y que debe ser mantenida a la par que se desarrollan las funcionalidades siguientes.

Las historias de usuarios quedan estructuradas de la siguiente manera:

- **Nombre:** nombre descriptivo de la HU.
- **Prioridad:** grado de prioridad que le asigna el cliente a la HU en dependencia de su importancia. Los valores que puede tomar son: alta, media o baja.
- **Complejidad:** grado de complejidad que le asigna el desarrollador a la HU luego de analizarla. Los valores que puede tomar son: alta, media o baja.

- **Estimación:** unidades de tiempo estimadas por el desarrollador para darle cumplimiento a la HU.
- **Iteración:** número de la iteración en la cual será implementada la HU.
- **Descripción:** descripción simple que brinda el cliente sobre lo que debe hacer la funcionalidad en cuestión.

Adicionalmente a cada HU se le asigna un número para facilitar su identificación por parte del equipo de desarrollo.

Durante esta fase se identificaron 3 historias de usuarios, cada una de ellas respondiendo a las diferentes funcionalidades solicitadas por el cliente y dando una idea de cómo debe ser su posterior implementación. A continuación se describen las historias de usuarios identificadas.

Tabla 1 Historia de usuario “Crear el escenario virtual”

Historia de usuario	
Número: 1	Nombre: Crear el escenario virtual
Prioridad en Negocio: Alta	Complejidad en Desarrollo: Alta
Puntos Estimados: 3	Iteración Asignada: 1
Descripción: Se crea el escenario virtual y los módulos de interacción del avatar con los objetos dinámicos y estáticos. Además de la verificación del correcto funcionamiento de las colisiones, las texturas, las luces, el sonido y los detalles arquitectónicos del docente.	
Observaciones: Escenario virtual donde el avatar se desplazará.	

Tabla 2 Historia de usuario “Desplazamiento del avatar”

Historia de usuario	
Número: 2	Nombre: Desplazamiento del avatar
Prioridad en Negocio: Alta	Complejidad en Desarrollo: Alta
Puntos Estimados: 2	Iteración Asignada: 2
Descripción: El avatar del usuario se podrá desplazar por el escenario virtual. Este va a ser dirigido por instrucciones que serán entradas por teclado y ratón. También comprobará las colisiones con los demás objetos del escenario virtual que lo rodean.	
Observaciones: El desplazamiento del avatar será dirigido mediante el teclado y el ratón.	

Tabla 3 Historia de usuario “Elementos de Inteligencia Artificial”

Historia de usuario	
Número: 3	Nombre: Elementos de Inteligencia Artificial
Prioridad en Negocio: Alta	Complejidad en Desarrollo: Alta
Puntos Estimados: 3	Iteración Asignada: 3
Descripción: Elementos con comportamientos autónomos dentro del escenario virtual. Tendrán un comportamiento determinado por el <i>script</i> que se le aplique.	
Observaciones: Los avatares interactivos y decorativos que aportan vida y realismo a la simulación.	

2.4 Fase de Planificación

Durante la fase de planificación se realiza una estimación del esfuerzo para generar cada HU. XP no tiene métricas predeterminadas, sino que son libres, lo que permite utilizar cualquier criterio para medir el alcance del proyecto. La métrica a utilizar en este caso es la que toma como medida el punto, el cual se considera como una semana ideal de trabajo, donde se trabaja el tiempo planeado sin ningún tipo de interrupción. La estimación incluye todo el esfuerzo asociado a la implementación de HU.

Es una fase corta, en la que el cliente y el grupo de desarrolladores acuerdan el orden en que deberán implementarse las UH, y, asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones grupales de planificación. La planificación no debe ser estricta puesto que hay muchas variables en juego, debe ser flexible para poder adaptarse a los cambios que puedan surgir. Una buena estrategia es hacer planificaciones detalladas para unas pocas semanas y planificaciones mucho más abiertas para unos pocos meses.

Estimación de esfuerzo por HU

La siguiente tabla contiene la estimación de esfuerzo por historia de usuario (HU) según el orden a realizar. Cada estimación incluye el esfuerzo asociado a la implementación de la HU, la misma se expresa utilizando como medida una puntuación, que es directamente proporcional al esfuerzo.

Tabla 4 Puntos estimados para las historias de usuario

Historia de usuario	Puntos estimados
Crear el escenario virtual	3
Desplazamiento del avatar	2
Elementos de Inteligencia Artificial	3

El tiempo total estimado para el desarrollo del producto es de 8 puntos de estimación, que equivalen a 2 meses ideales de trabajo.

Plan de Iteraciones

Una vez identificadas las HU del producto y estimado el esfuerzo dedicado a la realización de cada una de ellas, se procede a la planificación de la etapa de implementación del proyecto. Se decide realizar esta etapa en tres iteraciones, las cuales se detallan a continuación.

Iteración 1

Esta iteración tiene como objetivo la implementación de la HU referente a la creación del escenario virtual. Al final se contará con una primera versión de prueba, la cual será mostrada al cliente con el objetivo de obtener una retroalimentación para el grupo de trabajo.

Iteración 2

El objetivo de esta iteración es la implementación de la HU referente al desplazamiento del avatar. Al finalizar se contará con una versión de prueba, que contará con los procedimientos que le permitirán al usuario recorrer el entorno virtual. Dicha versión será mostrada al cliente con el objetivo de realizar cambios necesarios en base a la opinión del mismo.

Iteración 3

Durante el transcurso de esta iteración, se implementará la HU referente a la aplicación de IA en los objetos con los que interactuará el usuario. Al finalizar se contará con una primera versión final del producto, el cual será puesto en funcionamiento durante un período de tiempo para evaluar su desempeño.

Plan de duración de las iteraciones

Como parte del ciclo de vida de un proyecto utilizando XP se crea el plan de duración de cada una de las iteraciones, en este caso se hace para el único equipo de desarrollo con que se cuenta. Este plan tiene como meta mostrar la duración de cada iteración, así como el orden en que serán implementadas las HU en cada una de las mismas.

Tabla 5 Plan de duración de las iteraciones

Iteración	Orden de las historias de usuario a implementar	Duración de la iteración
Iteración 1	Crear el escenario virtual	3 semanas
Iteración 2	Desplazamiento del avatar	2 semanas
Iteración 3	Elementos de Inteligencia Artificial	3 semanas

Plan de entrega

A continuación se presenta el plan de entregas ideado para la fase de implementación. Como resultado del mismo se obtienen versiones del producto al finalizar cada iteración en la fecha aproximada que se indica en la siguiente tabla.

Tabla 6 Plan de entregas

Producto	Fin 1ra Iteración 1era semana de Mayo	Fin 2da Iteración 4ta semana de Mayo	Fin 3ra Iteración 3ra semana de Junio

Paseo virtual del docente 5.	0.1	0.2	1.0
------------------------------	-----	-----	-----

2.5 Fase de Diseño

El diseño adecuado para el *software* es aquel que supera con éxito todas las pruebas, no tiene lógica duplicada, refleja claramente la intención de implementación del equipo de desarrollo y tiene el menor número posible de clases y métodos. El diseño crea una estructura para la organización de la lógica del sistema y permite que sea escalable con cambios en un solo lugar. Los diseños se crean con el mayor grado de sencillez posible, dividiéndolos en varias partes que pudieran tornarse complejas.

Arquitectura del sistema

Expresa una organización o esquema estructural fundamental para sistemas *software*. Proporciona un conjunto de subsistemas predefinidos, especifica sus responsabilidades, e incluye una guía para organizar las relaciones entre ellos.

El producto se desarrolla sobre el motor de video-juegos Unity 3D y posee una arquitectura en 3 capas que se describen a continuación:

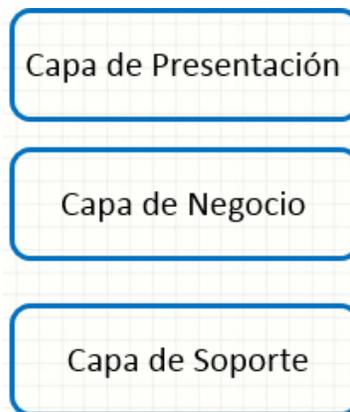


Figura 6 Arquitectura

- **Capa de Presentación:** Es la capa con la que interactúa el usuario y está formada por un menú de herramientas localizado en la barra de Menú principal de Unity 3D donde se encuentran los datos

de configuración de los elementos del sistema que se muestran dentro del panel Inspector de Objetos.

- **Capa de Negocio:** Está formada por las entidades, que representan objetos que van a ser manejados o utilizados por toda la aplicación.
- **Capa de Soporte:** Contiene las bibliotecas para la física, los gráficos, el sonido, y el almacenamiento de los datos brindados por el motor gráfico de Unity 3D.

Tarjetas Clase-Responsabilidad-Colaboradores (CRC)

Las tarjetas CRC (Clase, Responsabilidad, Colaborador) son una herramienta de reflexión en el diseño de *software* orientado a objetos. Fueron propuestas por Ward Cunningham y Kent Beck. Se utilizan normalmente cuando primero se determinan las clases que se necesitan y cómo van a interactuar, después se implementa la solución (21). A continuación se muestran las tarjetas CRC que fueron confeccionadas:

Tabla 7 Tarjeta CRC “Main”

Tarjeta CRC	
Clase: Main	
Responsabilidades	Colaboraciones
Clase principal encargada de agrupar todas las funcionalidades con las que va a contar el sistema.	SceneManager

Tabla 8 Tarjeta CRC “SceneManager”

Tarjeta CRC	
Clase: SceneManager	
Responsabilidades	Colaboraciones
Clase encargada del manejo y administración de las escenas exteriores e	Scenes Avatar

interiores.	
-------------	--

Tabla 9 Tarjeta CRC “Avatar”

Tarjeta CRC	
Clase: Avatar	
Responsabilidades	Colaboraciones
Clase responsable del control de los personajes que se moverán por él escenario virtual.	LocomotionHandler

Tabla 10 Tarjeta CRC “LocomotionHandler”

Tarjeta CRC	
Clase: LocomotionHandler	
Responsabilidades	Colaboraciones
Clase responsable de los movimientos del avatar.	

Tabla 11 Tarjeta CRC “Scenes”

Tarjeta CRC	
Clase: Scenes	
Responsabilidades	Colaboraciones
Clase encargada de controlar los elementos de la escena.	Actor PerceptionManager

Tabla 12 Tarjeta CRC “Actor”

Tarjeta CRC	
Clase: Actor	
Responsabilidades	Colaboraciones
Elementos dinámicos y estáticos que forman parte de la escena.	

Tabla 13 Tarjeta CRC “PerceptionManager”

Tarjeta CRC	
Clase: PerceptionManager	
Responsabilidades	Colaboraciones
Clase encargada de controlar las acciones relacionadas con la percepción de los objetos y avatares en las colisiones.	

2.6 Conclusiones parciales

Se realiza la descripción mediante historias de usuarios contando con tres HU precisadas por el cliente, estas serán implementadas en tres iteraciones. Fueron definidas las tarjetas CRC necesarias para establecer el orden de implementación de las HU.

Al culminar el presente capítulo, se obtuvieron los artefactos generados por la metodología empleada, los cuales permitieron una mejor comprensión de las funcionalidades, elementos y guía de trabajo a seguir por los desarrolladores durante todo el ciclo de desarrollo del software. Se definió la arquitectura sobre la cual se construye la aplicación.

CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBAS

En el presente capítulo se detallan las tres iteraciones llevadas a cabo durante la etapa de codificación del sistema, debido a que XP plantea que la implementación de un *software* debe realizarse de forma iterativa. Obteniendo en cada iteración un producto funcional el cual va a ser evaluado por el cliente, para luego su continuo desarrollo, cumpliendo con las expectativas recomendadas por el mismo. Además, se definen las pruebas a realizarle al sistema para comprobar su funcionamiento.

3.1 Fase de Producción

En esta fase se genera todo el código fuente necesario para satisfacer las HU definidas para la solución y se describen todas las tareas realizadas en cada iteración. Todo el trabajo de la iteración es expresado en tareas de programación.

Iteraciones en el desarrollo del producto

En esta fase se genera todo el código fuente necesario para satisfacer las historias de usuario definidas para la solución y se describen todas las tareas realizadas en cada iteración.

Una vez realizada la planificación, se determinaron tres iteraciones de desarrollo sobre el producto, obteniendo como resultado un producto con todas las restricciones y características deseadas por el cliente para luego ser utilizado. A continuación se detallan cada una de las iteraciones, mostrando las tareas para cada una de ellas.

Iteración 1

Tabla 14 Historia de usuario correspondiente a la Iteración 1

No HU	Historia de usuario	Tiempo de implementación	
		Estimación	Real
1	Crear el escenario virtual	3	3

Tareas de las HU pertenecientes a la primera iteración

Tabla 15 Primera tarea asociada a la historia de usuario "Crear el escenario virtual"

Tarea por historia de usuario	
Número de la tarea: 1	Número de historia de usuario: 1
Nombre de la tarea: Modelar la parte externa del escenario virtual.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 20/3/2015	Fecha fin: 31/3/2015
Descripción: Modelar en Autodesk 3ds Max la vista exterior del docente.	

Tabla 16 Primera tarea asociada a la historia de usuario "Crear el escenario virtual"

Tarea por historia de usuario	
Número de la tarea: 2	Número de historia de usuario: 1
Nombre de la tarea: Modelar la parte interna del escenario virtual.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 5/4/2015	Fecha fin: 17/4/2015
Descripción: Modelar en Autodesk 3ds Max la vista interior del docente.	

Iteración 2

Tabla 17 Historia de usuario correspondiente a la Iteración 2

No HU	Historia de usuario	Tiempo de Implementación	
		Estimación	Real
2	Desplazamiento del avatar	2	2

Tareas de las historias de usuarios pertenecientes a la segunda iteración

Tabla 18 Tarea asociada a la historia de "Desplazamiento del avatar"

Tarea por Historia de Usuario	
Número de la tarea: 3	Número de historia de usuario: 2
Nombre de la tarea: Desplazamiento del avatar por el entorno virtual.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha inicio: 24/4/2015	Fecha fin: 8/5/2015
Descripción: Desplazamiento del avatar por el escenario virtual mediante el teclado y el ratón.	

Iteración 3

Tabla 19 Historia de usuario correspondiente a la Iteración 3

No HU	Historia de usuario	Tiempo de Implementación	
		Estimación	Real
3	Elementos de Inteligencia Artificial	3	3

Tareas de las historias de usuarios pertenecientes a la tercera iteración

Tabla 20 Tarea asociada a la historia de “Elementos de Inteligencia Artificial”

Tarea por Historia de Usuario	
Número de la tarea: 4	Número de historia de usuario: 3
Nombre de la tarea: Aplicar inteligencia artificial a los objetos que responden a una determinada acción del avatar.	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Fecha inicio: 10/5/2015	Fecha fin: 31/5/2015
Descripción: Desplazamiento de los personajes, detección de colisión con objetos inmóviles y con personajes estáticos.	

3.2 Implementación

Cada una de las estructuras utilizadas en este paseo virtual está compuestas por *gameObject* similares. Estos funcionan como contenedores donde se les adjuntan componentes y *script* que son los que dictan el

funcionamiento de los mismos. Convirtiéndose en parámetros modificables de forma visual. Todos estos *gameObject* se comunican entre ellos de forma pública a través de *MonoBehaviour*, que es la clase base de la que derivan todos los scripts.

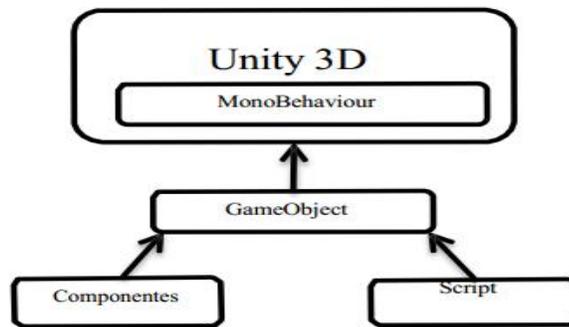


Figura 7 Relación entre el núcleo de Unity 3D y los objetos que componen la escena

Componentes

Los componentes que provee Unity 3D para la interacción en las escenas se muestran en formas diversas. Pueden ser para crear comportamiento, definiendo apariencia, e influenciando otros aspectos de la función de un objeto en el juego. Los componentes comunes de producción de juego vienen contruidos dentro del Unity 3D, desde el *Rigidbody*, hasta elementos más simples, como luces, las cámaras, los emisores de partículas y otros, que permiten estructurar de forma rápida un escenario (22).

Componentes de visión

- *Camera*

Las cámaras son los dispositivos que capturan y muestran el mundo al jugador. Mediante la personalización y la manipulación de cámaras, se puede hacer la presentación del juego verdaderamente única. El número de cámaras a utilizar en una escena es ilimitado y la configuración puede hacerse en cualquier orden, en cualquier lugar de la pantalla, o sólo ciertas partes de la pantalla (23).

Componentes de audio

- *Audio Source*

La fuente de audio reproduce un clip de audio en la escena. Si el clip es un clip de audio 3D, la fuente se reproduce en una posición determinada y se atenúan con la distancia. El audio se puede transmitir a cabo entre los altavoces (estéreo a 7.1) (*Spread*) y se transforma entre 3D y 2D. Esto puede ser controlado con la distancia con las curvas de difuminación. Solo por filtros individuales se pueden aplicar a cada fuente de audio para una experiencia de audio aún más rica.

- *Audio Listener*

El *Audio Listener* actúa como un dispositivo de entrada de micrófono. Se recibe información desde cualquier fuente de audio que figura en la escena y reproduce sonidos a través de los altavoces del ordenador. Si un detector de audio está dentro de los límites de una zona, se aplica a todos los sonidos audibles en la escena. Por otra parte, los efectos de audio se pueden aplicar a la escucha y se aplicarán a todos los sonidos audibles en la escena.

Componentes de colisiones

- *Box Collider*

Un *Box Collider* es una primitiva básica de colisión en forma de cubo (24). El *Box Collider* puede cambiar de tamaño en diferentes formas de prismas rectangulares. Funciona muy bien para las puertas, paredes y plataformas.

- *Capsule Collider*

El *Capsule Collider* se compone de dos medias esferas unidas por un cilindro. Se trata de la misma forma que la cápsula primitiva. Puede ajustar radio y la altura del *Capsule Collider* independientemente unos de otros. Se utiliza en el controlador de caracteres y funciona bien para los polos, o se puede combinar con otros *Colliders* para formas inusuales.

- *Sphere Collider*

El Sphere Collider es una colisión básica en forma de esfera primitiva (25). El *Sphere Collider* puede ser redimensionar a escala uniforme, pero no a lo largo de cada eje. Funciona muy bien para la caída de piedras, pelotas de ping-pong y otros cuerpos en forma esférica.

- *Character Controller*

El *Character Controller* se utiliza principalmente para el control del reproductor de tercera persona o en primera persona que no haga uso de física *Rigidbody*. Se trata simplemente de una cápsula que se puede utilizar para moverse en la dirección de un guión. El controlador entonces lleva a cabo el movimiento limitado por colisiones. Se desliza a lo largo de las paredes, subir las escaleras y caminar en pendientes dentro de su límite.

Componentes de animación

- *Animation*

Cualquier *GameObject* que tiene un avatar también tendrá un componente animador, que es la relación entre el carácter y su comportamiento. Este componente referencia a componentes de un controlador animador que se utiliza para configurar el comportamiento del personaje. Esto incluye la instalación de máquinas de estado, se mezclan árboles, y los eventos que deben ser controlados desde el guión.

3.3 Scripts

El *Scripting* es una parte esencial que define el comportamiento del juego (o las normas) en Unity.

Scripts de la cámara

Tabla 21 *Script* de cámara " *CameraController* "

Nombre
<i>CameraController</i>
Descripción
Controla el comportamiento de la cámara la cual perseguirá en todo momento al personaje. También detecta las colisiones con los objetos de etiqueta "paredes" para evitar que estos se interpongan entre la cámara y el personaje.

Scripts del avatar

Tabla 22 Script del avatar "CharacterAnimator"

Nombre
CharacterAnimator
Descripción
Este <i>script</i> controla las animaciones que el personaje debe realizar en cada acción que el usuario decida hacer y la mezcla de las mismas.

Tabla 23 Script del avatar "GraphicUserInterface"

Nombre
GraphicUserInterface
Descripción
Este <i>script</i> se encarga de crear toda la interfaz de usuario referente a las escenas en las que el personaje posea acción de la aplicación así como guardar los objetos y mostrar sus características.

Scripts de objetos

Tabla 24 Script de objeto "Open_Door"

Nombre
Open_Door
Descripción
Este <i>script</i> activa una animación del objeto al que pertenece dependiendo de la distancia en la que se encuentra el personaje de él. En este caso se abrirá la puerta si se acerca o se cerrará si se aleja.

Scripts de avatares de IA

Tabla 25 Script de avatar de "IA"

Nombre
AiPatrol

Descripción
Este <i>script</i> se encarga del desplazamiento, colisión y percepción de los personajes independientes del entorno virtual.

3.4 Pruebas

En la fase de producción se requieren pruebas adicionales y revisiones de rendimiento antes de que el producto sea trasladado al entorno del cliente.

Uno de los pilares fundamentales de XP es el proceso de pruebas, el cual anima a los desarrolladores a probar constantemente tanto como sea posible (26). Esto contribuye a elevar la calidad de los productos desarrollados y a la seguridad de los programadores a la hora de introducir cambios o modificaciones. La metodología XP plantea las pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que la funcionalidad sea la esperada por el cliente.

Pruebas de aceptación

Las pruebas de aceptación XP, también las llamadas pruebas del cliente son especificadas por el mismo y se centran en las características y funcionalidades generales del sistema que son visibles y revisables por parte del cliente. Las pruebas de aceptaciones derivan de las HU que se han implementado como parte de la liberación del *software* (27).

Durante las iteraciones las HU seleccionadas serán traducidas a pruebas de aceptación. En ellas se especifican, desde la perspectiva del cliente, los escenarios para probar que una HU ha sido implementada correctamente. El objetivo final de las mismas es garantizar que los requerimientos han sido cumplidos y que el producto es aceptable.

Como resultado de las pruebas de aceptación se obtendrán artefactos descritos en tablas, estas contarán con los siguientes campos:

- **Código:** servirá como identificador de la prueba realizada, a su vez será sugerente al nombre de la prueba a la que hace referencia.
- **HU:** tendrá el nombre de la historia de usuario a la que hace referencia la prueba a realizar.

- **Nombre:** nombre que se le da a la prueba a realizar.
- **Descripción:** se describe la funcionalidad que se desea probar.
- **Condiciones de Ejecución:** mostrará las condiciones que deben cumplirse para poder llevar a cabo el caso de prueba, estas condiciones deben ser satisfechas antes de la ejecución del caso de prueba para que se puedan obtener los resultados esperados.
- **Entradas / Pasos de Ejecución:** se hará la descripción de cada uno de los pasos seguidos durante el desarrollo de la prueba, se tendrá en cuenta cada una de las entradas que hace el usuario con el objetivo de ver si se obtiene el resultado esperado.
- **Resultado esperado:** se hará una breve descripción del resultado que se espera obtener con la prueba realizada.
- **Evaluación de la prueba:** acorde al resultado de la prueba realizada se emitirá una evaluación sobre la misma. Esta evaluación tendrá uno de los tres resultados que a continuación se describen:
 - Bien: cuando el resultado de la prueba es exactamente el esperado por el usuario.
 - Parcialmente bien: cuando el resultado no es completamente el esperado por el cliente o usuario de la aplicación y muestra resultados erróneos o fuera de contexto.
 - Mal: cuando el resultado de la prueba realizada genera un error de codificación en la aplicación o muestra como resultado elementos no deseados o fuera de contexto, trayendo como consecuencia que la funcionalidad requerida por el cliente no tenga resultado, lo que invalida también la UH.

Resultados de las pruebas de aceptación

Las pruebas de aceptación se llevaron a cabo durante las tres iteraciones establecidas para la modelación e implementación del producto. En cada iteración, entre el equipo de desarrollo y el cliente, se realizaron varias pruebas de este tipo, al principio se detectaron varias no conformidades en cuanto al desplazamiento del avatar y a la aplicación de la inteligencia artificial a los objetos que responderán a determinadas acciones del usuario.

A continuación se muestran los casos de pruebas de aceptación realizados para las HU que fueron establecidas en el capítulo anterior, en los cuales se evidencia el correcto funcionamiento del producto.

Tabla 26 Caso de prueba 1 sobre la historia de usuario 1

Caso de Prueba de Aceptación	
Código: P1_ HU1	Historia de Usuario: 1
Nombre: Comparación visual del entorno virtual.	
Descripción: Prueba para verificar el nivel de realismo del entorno virtual.	
Condiciones de Ejecución: Deben haberse tomado fotos y realizado videos del docente 5 previamente.	
Entradas/Pasos de Ejecución: Seleccionar cada foto o video correspondiente a cada parte del entorno virtual. Verificar mediante la apreciación visual el nivel de semejanza.	
Resultado Esperado: Nivel de semejanza alto.	
Evaluación de la Prueba: Bien.	

Tabla 27 Caso de prueba 1 sobre la historia de usuario 2

Caso de Prueba de Aceptación	
Código: P1_ HU2	Historia de Usuario: 2
Nombre: Verificación del desplazamiento del avatar hacia la izquierda.	
Descripción: Prueba para verificar el desplazamiento del avatar hacia la izquierda.	
Condiciones de Ejecución: El usuario debe haber accedido al entorno virtual.	
Entradas/Pasos de Ejecución: El usuario presiona la tecla “Flecha izquierda” del teclado.	
Resultado Esperado: El avatar se desplaza hacia la izquierda.	
Evaluación de la Prueba: Bien.	

Tabla 28 Caso de prueba 2 sobre la historia de usuario 2

Caso de Prueba de Aceptación	
Código: P2_ HU2	Historia de Usuario: 2
Nombre: Verificación del desplazamiento del avatar hacia la derecha.	

Descripción: Prueba para verificar el desplazamiento del avatar hacia la derecha.
Condiciones de Ejecución: El usuario debe haber accedido al entorno virtual.
Entradas/Pasos de Ejecución: El usuario presiona la tecla “Flecha derecha” del teclado.
Resultado Esperado: El avatar se desplaza hacia la derecha.
Evaluación de la Prueba: Bien.

Tabla 29 Caso de prueba 3 sobre la historia de usuario 2

Caso de Prueba de Aceptación	
Código: P3_ HU2	Historia de Usuario: 2
Nombre: Verificación del desplazamiento del avatar hacia atrás.	
Descripción: Prueba para verificar el desplazamiento del avatar hacia atrás.	
Condiciones de Ejecución: El usuario debe haber accedido al entorno virtual.	
Entradas/Pasos de Ejecución: El usuario presiona la tecla “Flecha abajo” del teclado.	
Resultado Esperado: El avatar se desplaza hacia atrás.	
Evaluación de la Prueba: Bien.	

Tabla 30 Caso de prueba 4 sobre la historia de usuario 2

Caso de Prueba de Aceptación	
Código: P4_ HU2	Historia de Usuario: 2
Nombre: Verificación del desplazamiento del avatar hacia adelante.	
Descripción: Prueba para verificar el desplazamiento del avatar hacia adelante.	
Condiciones de Ejecución: El usuario debe haber accedido al entorno virtual.	
Entradas/Pasos de Ejecución: El usuario presiona la tecla “Flecha arriba” del teclado.	
Resultado Esperado: El avatar se desplaza hacia adelante.	
Evaluación de la Prueba: Bien.	

Tabla 31 Caso de prueba 5 sobre la historia de usuario 2

Caso de Prueba de Aceptación

Código: P5_ HU2	Historia de Usuario: 2
Nombre: Verificación de la visualización del entorno virtual mediante el “Mouse”.	
Descripción: Prueba para verificar la visualización que tendrá el avatar del entorno virtual mediante el movimiento del “Mouse”	
Condiciones de Ejecución: El usuario debe haber accedido al entorno virtual.	
Entradas/Pasos de Ejecución: El usuario mueve el “Mouse” hacia una dirección del entorno virtual.	
Resultado Esperado: Se muestran los objetos que se encuentran en la dirección en la que sea movido el “Mouse”.	
Evaluación de la Prueba: Bien.	

Tabla 32 Caso de prueba 1 sobre la historia de usuario 3

Caso de Prueba de Aceptación	
Código: P1_ HU3	Historia de Usuario: 3
Nombre: Verificación de la inteligencia artificial del tipo de objeto puerta.	
Descripción: Prueba para verificar la inteligencia artificial del tipo de objeto puerta.	
Condiciones de Ejecución: El usuario debe haber accedido al entorno virtual.	
Entradas/Pasos de Ejecución: El usuario se desplaza hacia un laboratorio del entorno virtual y se posiciona frente a la puerta del mismo.	
Resultado Esperado: Se observa que la puerta se abre o se cierra en dependencia de la distancia del usuario a la misma.	
Evaluación de la Prueba: Bien.	

Tabla 33 Caso de prueba 2 sobre la historia de usuario 3

Caso de Prueba de Aceptación	
Código: P2_ HU3	Historia de Usuario: 3
Nombre: Verificación de la inteligencia artificial del tipo de objeto personaje independiente.	
Descripción: Prueba para verificar la inteligencia artificial del tipo de objeto personaje independiente.	

Condiciones de Ejecución: El usuario debe haber accedido al entorno virtual.
Entradas/Pasos de Ejecución: El usuario se desplaza por el entorno virtual para comprobar las funcionalidades de movimiento de los personajes independientes.
Resultado Esperado: Se muestra que el personaje independiente es capaz de bordear y colisionar con el personaje principal.
Evaluación de la Prueba: Bien.

Tabla 34 Caso de prueba 3 sobre la historia de usuario 3

Caso de Prueba de Aceptación	
Código: P3_ HU3	Historia de Usuario: 3
Nombre: Verificación de la inteligencia artificial del tipo de objeto personaje.	
Descripción: Prueba para verificar la inteligencia artificial del tipo de objeto personaje.	
Condiciones de Ejecución: El usuario debe haber accedido al entorno virtual.	
Entradas/Pasos de Ejecución: El usuario interactúa con el personaje encargado de cada laboratorio.	
Resultado Esperado: El personaje brinda información del lugar en el cual se encuentra el usuario dentro del entorno virtual.	
Evaluación de la Prueba: Bien.	

3.5 Conclusiones parciales

En el presente capítulo se definieron las tareas a desarrollar pertenecientes a cada HU en cada una de las tres iteraciones anteriormente planificadas. Estas tareas fueron desarrolladas de acorde al tiempo planificado para las mismas, posibilitando el cumplimiento del plan de entrega del producto, descrito en el capítulo 2. Por último se realizaron pruebas al sistema para comprobar su funcionalidad, llevándose a cabo las pruebas de aceptación, las cuales fueron seleccionadas ya que las mismas son llevadas a cabo por el cliente y es el mejor evaluador que puede tener el producto, ya que el mismo, puede verificar que el sistema cumpla con sus expectativas.

CONCLUSIONES GENERALES

La utilización del motor de videojuegos Unity 3D permitió desarrollar un paseo virtual interactivo del Docente 5 de la UCI, que permite visitar las áreas exteriores e interiores del mismo, incorporando elementos de inteligencia artificial y un nivel gráfico adecuado.

RECOMENDACIONES

- Incorporar más elementos de inteligencia artificial para una mejor experiencia dentro del entorno virtual.
- Integrar el producto con el paseo virtual que se desarrolla en la Facultad 5 de la UCI.

REFERENCIAS BIBLIOGRÁFICAS

1. **García Ruiz, D.en C. Miguel Ángel** . Aplicaciones de la Realidad Virtual e Inteligencia Artificial. *UOIT METIS Research Group*. [En línea] [Citado el: 20 de Enero de 2014.] http://www.hrl.uoit.ca/~miguelga/realidad_virtual.pdf.
2. **Ruíz Valera, Ing. Lidea**. Fundamentos de la Realidad virtual. *Slideshare*. [En línea] [Citado el: 2014 de Enero de 20.] <http://www.slideshare.net/docentecis/realidad-virtual-26156517>.
3. **Conceptos de innovación**. [En línea] 2011. [Citado el: 2014 de Enero de 20.] http://valladolid.iver.es/wiki/index.php/REALIDAD_VIRTUAL.
4. **Envista**. [En línea] [Citado el: 2014 de Enero de 20.] <http://www.envista.es/Visitas-Virtuales.html>.
5. **Visita virtual - EcuRed. EcuRed**. [En línea] [Citado el: 2014 de Enero de 20.] http://www.ecured.cu/index.php/Visita_virtual.
6. **Vicente Botti, Federico Barber**. Novática. [En línea] [Citado el: 2014 de Enero de 22.] <http://www.ati.es/novatica/1998/131/pres131.html>.
7. **Identi . Identi**. [En línea] 2009-2014. [Citado el: 2014 de Enero de 23.] <http://www.identi.li/index.php?topic=55302>.
8. **PolyFactor**. [En línea] [Citado el: 2014 de Enero de 23.] <http://www.polyfactor.com/page/2/>.
9. **DirectorioW**. [En línea] 2006-2011. [Citado el: 2014 de Enero de 25.] http://www.directoriow.com/c_Programas_Windows_Linux_Mac_6.html.
10. **Scribd**. [En línea] [Citado el: 7 de Febrero de 2014.] <http://es.scribd.com/doc/12983329/Metodologia-de-Desarrollo-de-Software>.
11. **G. Booch, I.J., J. Rumbaugh**. *El Proceso Unificado de desarrollo de software*. 2000.
12. **Piattini, M**. *Análisis y diseño detallado de aplicaciones informáticas de gestión*. s.l. : Ra-Ma Editorial, S.A.
13. **Ken Schwaber, M.B**. *Agile Software Development with Scrum*. 2001.
14. **Patricio Letelier, M.C.P**. *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. 2009.
15. **XProgramming.com** An Agile Software Development Resource. [En línea] 1999 - 2013. [Citado el: 25 de Febrero de 2014.] <http://xprogramming.com/index.php>.
16. **Object Management Group - UML**. [En línea] [Citado el: 17 de Marzo de 2014.] <http://www.uml.org/>.

17. Visual Paradigm. [En línea] [Citado el: 25 de Marzo de 2014.] <http://www.visual-paradigm.com/>.
18. <http://www.blender.org>. [En línea] 2015.
http://wiki.blender.org/index.php/Doc:2.6/Manual/Render/Cycles/GPU_Rendering.
19. 2cCAD Global group. [En línea] [Citado el: 2014 de Abril de 5.]
http://www.2acad.net/index.php?option=com_content&view=article&id=86&Itemid=457.
20. Tangient LLC. [En línea] [Citado el: 2014 de Abril de 10.] <http://programacion-extrema.wikispaces.com/5.+Ciclo+de+vida+y+fases>.
21. Scribd Inc. Scribd. [En línea] [Citado el: 2014 de Abril de 20.] <http://es.scribd.com/doc/45010504/Tarjetas-CRC>.
22. Unity3D para la Rehabilitación Cognitiva contra el Alzheimer . [En línea] [Citado el: 2014 de Mayo de 25.]
<http://riunet.upv.es/bitstream/handle/10251/17797/Memoria.pdf.txt?sequence=3>.
23. Scribd Inc. Scribd. [En línea] [Citado el: 21 de Mayo de 2014.] <http://es.scribd.com/doc/201920605/Manual-de-Unity>.
24. unity3d.com. [En línea] 2015. <http://docs.unity3d.com/es/current/Manual/class-BoxCollider.html>.
25. unity3d.com. [En línea] 2015. <http://docs.unity3d.com/es/current/Manual/class-SphereCollider.html>.
26. J. J. Gutiérrez, M. J. Escalona, M. Mejías, J. Torres. PRUEBAS DEL SISTEMA EN PROGRAMACIÓN. *Departamento de Lenguajes y Sistemas Informáticos, Universidad de Sevilla*. [En línea] [Citado el: 24 de Mayo de 2014.]
http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf.
27. Metodología XP. [En línea] [Citado el: 25 de Mayo de 2014.]
<https://sites.google.com/site/xpmetodologia/marco-teorico/funcionamiento>.