

Especificación de Requisitos

Proyecto de Grado de ISC
Videojuego
(PUCMM)

Índice General

1	Introducción.....	4
1.1	Propósito.....	4
1.2	Ámbito del Sistema	4
1.3	Definiciones, Acrónimos y Abreviaturas	6
1.4	Referencias	6
1.5	Visión General del Documento	7
2	Descripción General	8
2.1	Perspectiva del Producto	8
2.2	Funciones del Producto.....	9
2.2.1	Tipo de Juego	10
2.2.2	Modo de un jugador y entidades de AI	11
2.2.3	Modalidad de multi-jugador en línea	11
2.2.4	Complejidad del Juego	12
2.3	Características de los Usuarios	13
2.4	Restricciones.....	14
3	Requisitos Específicos	15
3.1	Requerimientos Funcionales.....	15
3.1.1	Requerimientos Sobre el Tipo de Juego	15
3.1.2	Requerimientos del Modo de un jugador y entidades de AI.....	16
3.1.3	Requerimientos de la Modalidad de Multi-jugador en Línea	17
3.1.4	Requerimientos de la Complejidad del Juego	18
3.1.5	Elementos Visuales y de Sonido	18
3.2	Requisitos de Rendimiento	19
3.3	Atributos del Sistema.....	20
4	Anexos	21
4.1	Diagramas UML.....	21
4.1.1	UML – Caso de Uso.....	21
4.1.2	UML – Clases	22
4.1.3	UML – Diagrama de Actividades	23
4.1.4	UML – Secuencia	24

4.2	Diagrama de Despliegue.....	25
4.3	Diagrama Físico	26
4.4	UED – Diseño de Ambiente de Usuario	27

1 Introducción

Este documento es la Especificación de Requisitos de Software (ERS) correspondiente al proyecto de Videojuegos, desarrollado en la PUCMM. Haciendo énfasis ligeramente en la descripción del proyecto, este será un juego estilo tirador en tercera persona, el cual tendrá dos modalidades de juego (modo de un jugador y multi-jugador).

En el presente documento se pueden encontrar diversas secciones, que recompilan detalles relevantes al proyecto mencionado. Las secciones contempladas en este documento son las contenidas en la especificación de la IEEE para un documento de este tipo, es decir: propósito, ámbito del sistema, definiciones, referencias y visión general del documento.

1.1 Propósito

El presente documento está dirigido a los jurados y asesores de proyectos de grado de la PUCMM. Además se dirige al equipo de desarrollo del proyecto mismo, puesto que en el presente escrito se manifiestan requerimientos del sistema y decisiones de diseño e implementación.

1.2 Ámbito del Sistema

El videojuego que se desarrollara siguiendo los requerimientos que serán expuestos en este documento, hasta ahora posee el nombre de PLATKILL. Este videojuego se busca desarrollar para que funcione perfectamente en ambientes de PC.

PLATKILL, será un juego que les permitirá a sus usuarios acceder a dos distintas modalidades de juego, siempre regidas por un estilo de tirador en tercera persona. Las formas de juego serán: modo de un único jugador y multi-jugador, en este ultimo los jugadores adversarios pondrán a prueba sus habilidades y destrezas para ver quien resulta ganador de cada partida. Dentro de una partida, ya sea en el modo de un solo jugador o en el multi-jugador, los usuarios tendrán acceso a distintos tipos de personajes y así mismo a tipos de armas, la elección de estas será

crucial para poder salir victorioso de cada partida. Con respecto a las partidas, las reglas que regirán las mismas se pretende que sean las siguientes: el usuario podrá elegir si desea que una partida termine luego de una determinada cantidad de tiempo, o una vez que uno de los jugadores de la misma alcance una determinada cantidad de puntos; estos factores serán los tomados para determinar quién es el ganador de la partida.

1.3 Definiciones, Acrónimos y Abreviaturas

A continuación se presenta una lista de algunos términos y abreviaturas empleadas en el presente escrito:

- PUCMM – Pontificia Universidad Católica Madre y Maestra.
- ISC – Ingeniería en Sistemas y Computación.
- XNA – Es una estructura de trabajo desarrollada por Microsoft, enfocada especialmente al desarrollo de videojuegos.
- BEPUPHYSICS – Es un motor encargado de manejar los cálculos relativos a la física de los cuerpos. Está orientado a sistemas que simulan un ambiente en tres dimensiones.
- LIDGREN - Es una librería para el lenguaje de programación CSharp orientada al manejo de envío y captura de paquetes a través de redes.
- PLATKILL – Nombre provisional del proyecto en cuestión.
- AI - Inteligencia Artificial.

1.4 Referencias

IEEE Recommended Practice for Software Requirements Specification. ANSI/IEEE STD. 830-1998.

1.5 Visión General del Documento

Este documento respeta la estructura dada por la IEEE para documentos de especificación de requisitos, en su estándar: ANSI/IEEE STD 830-1998. Algunos de las secciones que se omitieron de dicho estándar se consideran no aplicables para este proyecto, o bien no se consideran relevantes para los propósitos que se persiguen con este proyecto.

Este documento se desvía un tanto del estándar establecido en la IEEE en la sección número 3. El motivo de esta desviación es facilitar la comprensión de los temas que se exponen en la dicha sección y garantizarle al lector una lectura rápida y clara.

2 Descripción General

En esta sección se realiza una descripción de alto nivel del juego PLATKILL, se enfoca especialmente en las características que poseerá y así mismo las restricciones que tendrá. Además, se describe el perfil de los usuarios a los que el videojuego estará enfocado.

2.1 Perspectiva del Producto

El producto final que será el fruto de la realización de este proyecto, interactuara con cuatro otros sistemas. Estos sistemas se listan a continuación:

- **BEPUPHYSICS:** Este es un motor virtual enfocado en la simulación de cálculos físicos. Es decir este sistema permite realizar simulaciones del comportamiento de los cuerpos físicos en un entorno real bastante acertado. Sera empleado en todo momento en la ejecución del programa, puesto que esto permite crear una ilusión de realismo dentro del juego.
- **LIDGREN:** Es un sistema de envío y recibido de paquetes mediante el protocolo UDP. Este sistema contempla una serie de requerimientos, como predicción de latencia entre otras cosas útiles en el desarrollo de juegos multi-jugador a través de redes informáticas.
- La versión final de este producto, también interactuar en el modo multi-jugador con un servidor el cual gestionara las acciones que realicen todos los jugadores que estén conectados al mismo.

2.2 Funciones del Producto

Con el desarrollo del juego, PLATKILL, se busca ofrecerles a los jugadores un ambiente entretenido que les permita:

- **Tipo de Juego:** La creación de un juego que combine elementos de un tirador en tercera persona con un juego de plataforma, al estilo de un duelo a muerte.
- **Modo de un jugador y entidades de AI:** La creación de una inteligencia artificial que simule hasta cierto punto, la experiencia que se tendría al interactuar con un oponente humano.
- **Modalidad de multi-jugador en línea:** El juego debe poder ser accedido por múltiples jugadores por medio de una red de comunicación.
- **Complejidad del juego:** Emplear distintos elementos atractivos para los jugadores, con esto se busca elevar el grado de complejidad del juego y permitirle a los jugadores, personalizar su modo de juego. Con estos elementos se quiere dejar dicho, distintas estadísticas que modifiquen el modo de juego, armas distintas, y además distintos personajes.

2.2.1 Tipo de Juego

Como ya se ha dicho, con el desarrollo de este juego se pretende involucrar los conceptos de dos géneros muy populares de juegos, el tipo plataforma y el tipo tirador en tercera persona, en un ambiente que conduzca a los usuarios a competir. Dentro de una partida de PLATKILL, los usuarios deberán adquirir bienes para sacarle ventaja a sus oponentes. Los bienes serán básicamente ítems y herramientas virtuales que le permitan a un jugador salir airoso de una partida. La contienda entre los jugadores o el AI, se realizara involucrando armas de fuego y armas blancas (teniendo en cuenta siempre el concepto de caricaturas y de cierta forma poco realismo) como popularmente se realiza en juegos de tirador en tercera persona.

2.2.2 Modo de un jugador y entidades de AI

En cuanto a los jugadores de AI, con estos se busca simular un cierto grado de dificultad para ir entrenando a los usuarios, que lo deseen, en las mecánicas del juego, y que luego procedan a dificultades más altas.

Para la implementación del motor de la inteligencia artificial que se creara, se tiene previsto de ante mano emplear algoritmos heurísticos que dan la oportunidad de que el motor del juego pueda encontrar en tiempos aceptables respuestas aceptables dentro del juego. Cosas como que camino elegir, a que enemigos atacar, que ítems tomar, entre otras cosas similares, serán elegidas por el motor del juego empleando algoritmos como estos.

2.2.3 Modalidad de multi-jugador en línea

El modo de multi-jugador del producto final involucrara un servidor central que maneje todas las conexiones e instancias de los juegos que se estén llevando a cabo en los distintos extremos de la red. Para esta implementación se empleara una topología de red estilo cliente servidor, en la cual todos los jugadores con una copia del cliente del juego podrán acceder a dicho servidor y encontrar, se espera, a muchos otros jugadores para ponerse en contacto y llevar a cabo batallas dentro del juego. Con esto en mente, se eligió una librería escrita en el lenguaje de programación CSharp que facilita el manejo de las comunicaciones dentro de programas informáticos. Dicha librería es conocida como LIDGREN. Esta librería emplea el protocolo UDP para realizar las conexiones y posee una serie de controles para garantizar el envío de paquetes, además de una gama bastante amplia de herramientas dedicadas para los fines de lugar.

2.2.4 Complejidad del Juego

Buscando elevar el grado de complejidad del juego y así mismo su dinamismo, se diseñaron una serie de estadísticas que dictaran como trabajarán algunas de las mecánicas primarias del juego, tales como: el daño que realizarán las armas, la cantidad de puntos de vida de un jugador o una entidad de AI, la cantidad de saltos que se podrán realizar, el número de puntos de vida de un jugador o AI, entre otras. A continuación se presentan las estadísticas que se emplearán:

- **Poder Cuerpo a Cuerpo:** Esta estadística aumentará el daño que realiza un jugador con armas de cuerpo a cuerpo (espadas, cuchillos, etc.).
- **Poder a distancia:** Esta estadística aumentará el daño que realiza un jugador con armas de distancia (pistolas, arcos, etc.).
- **Aguante:** Esta estadística determinará que tan rápido se regenera la barra de aguante de cada jugador.
- **Velocidad:** Este manejará la velocidad de movimiento base de los jugadores.
- **Defensa:** Su función es disminuir el daño que recibe un jugador de armas de rango y de cuerpo contra cuerpo.
- **Vida:** Determina la cantidad de puntos de vida que tendrá un jugador.

2.3 Características de los Usuarios

Los usuarios a los que se está enfocando este juego pueden pertenecer a cualquier rango de edades. Con este juego se intentara involucrar a la mayor cantidad de personas posibles, en un ambiente que todos puedan disfrutar y pasar un rato divertido. Es por esto que muchos elementos que se incluirán en el juego serán un tanto caricaturesco, para que jugadores más jóvenes puedan jugarlo, y así mismo se le cause gracia a audiencias de mayor edad. El sistema deberá ser lo suficientemente intuitivo de utilizar, de forma que nuevos jugadores puedan rápidamente adentrarse a competir ya sea contra enemigos controlados por el ordenador o por otro contrincante humano.

2.4 Restricciones

Esta subsección esta principalmente dirigida a los desarrolladores del proyecto. Acá se mencionan detalles estrictamente técnicos relevantes para la realización del proyecto.

Los desarrolladores deberán:

- Tener dominio del lenguaje CSharp.
- Conocer el entorno de desarrollo que provee Visual Studio 2012.
- Poseer conocimientos de técnicas empleadas en el desarrollo de videojuegos en tres dimensiones.
- Conocer el marco de desarrollo XNA.
- Aprender a emplear el motor BEPUPHYSICS y así mismo la librería LIDGREN.

3 Requisitos Específicos

Esta sección esta principalmente enfocada a los diseñadores del juego, desarrolladores, artistas gráficos y de sonido. En la presente sección de este manifiesto, se plasman de manera más explícita y detallada los requerimientos del sistema. El propósito de esto, es proveerles a los diseñadores del juego una idea acertada de lo que se quiere construir.

Se listaran a forma de lista los requerimientos funcionales, de modo de que sean más legibles y así mismo más entendibles.

3.1 Requerimientos Funcionales

Seguido en las próximas subsecciones se presentan los requerimientos funcionales que deberán se implementados en cada una de las diversas áreas del videojuego.

3.1.1 Requerimientos Sobre el Tipo de Juego

- Deberá ser implementada una cámara que persiga a los jugadores en tercera persona.
- El juego tendrá que permitirle a los jugadores apuntas sus ataques mediante un sistema de miras en tercera persona.
- Los mapas del juego deberá poseer obstáculos, de forma similar a como es en un juego de plataformas.
- Además, dentro de las partidas los usuarios podrán adquirir ítems especiales que les permitan tomar ventaja sobre otros jugadores.

3.1.2 Requerimientos del Modo de un jugador y entidades de AI

Para crear el motor que maneje las decisiones que la AI decida tomar se empleara una librería llamada "Simple AI for XNA". Esta librería posee una implementación del algoritmo A*.

A* se clasifica dentro de los algoritmos de búsqueda en grafos. El algoritmo A* encuentra, siempre y cuando se cumplan unas determinadas condiciones, el camino de menor coste entre un nodo origen y uno objetivo. Este tipo de algoritmo resulta ideal en el desarrollo de AI para juegos.

La implementación del algoritmo mencionado deberán proporcionar las siguientes características:

- Los tiempos de respuesta de los algoritmos que sean implementados en el AI deben ser aceptables dentro de los parámetros que posteriormente se establezcan.
- Las respuestas de las entidades de AI deben ser fluidas.
- Los jugadores deberán poder establecer antes de iniciar las partidas, los estilos de juego, ítems y dificultad de los oponentes antes de empezar una partida. También podrán indicar el límite de enemigos manejados por la AI, hasta un límite que será establecido en futuras revisiones del juego.
- Las entidades de inteligencia artificial deberán ser capaces de proveer un desafío interesante para los jugadores. Si esto no se satisface el juego será muy simple y los jugadores perderán interés rápidamente, al menos en lo que al modo de un solo jugador se refiere.

3.1.3 Requerimientos de la Modalidad de Multi-jugador en Línea

- Debe proveérsele a los jugadores que opten participar en esta modalidad, la posibilidad de rápidamente unirse a partidas en línea de otros jugadores.
- El usuario podrá ver antes de unirse a una partida informaciones relevantes sobre la misma, tales como el estilo de juego, reglas del juego y el número de jugadores que actualmente se encuentran dentro de la partida.
- Los jugadores deberán poder comunicarse dentro de las partidas con sus aliados o enemigos por medio de un sistema de chat.
- Se le debe dar la capacidad al jugador que cree una partida de establecer las reglas del juego previamente.
- El diseño de la arquitectura de red que se emplee para garantizar la comunicación entre el servidor del juego y los clientes del mismo deberá ser lo suficientemente capaz de mantener una comunicación estable y constante, de modo que la latencia que se produzca entre el servidor y los clientes sea mínima. Esto es de vital importancia puesto que al tratarse de un juego que involucrara a múltiples jugadores el retardo más mínimo en el flujo de información entre el servidor y los clientes producirá una experiencia desagradable para los usuarios.

3.1.4 Requerimientos de la Complejidad del Juego

- Dentro del juego los personajes deberán poder hacer uso de armas tales como: armas de fuego, ballestas y armas blancas (cuchillos, espadas).
- Cada arma deberá tener características especiales de modo que se le ofrezca variedad dentro del juego a cada usuario.
- Existirán diversos personajes, los cuales deberán destacarse en ciertas áreas, específicas a las mecánicas del juego.
- Deberán implementarse las estadísticas que son mencionadas en la sección 2.2.4 del presente documento. Estas estadísticas deberán ser balanceadas para brindarle a los usuarios una experiencia justa.
- En cuanto, a la dificultad de las entidades manejadas por la AI. Los jugadores deberán poder elegir entre una serie de dificultades preestablecidas por los desarrolladores.
- Las dificultades se distinguirán una de otra, básicamente en decir que tan precisas serán las acciones que realizaran los jugadores controlados por el motor de AI.

3.1.5 Elementos Visuales y de Sonido

- Es necesario crear entornos visuales agradables y llamativos de modo que los jugadores que prueben nuestro videojuego se sientan atraídos al mismo y sean motivados a descubrir todo lo que el juego les ofrezca.
- Además, debemos proveer una interfaz agradable, que le permita a los jugadores reaccionar rápida y eficientemente a los eventos que ocurran dentro del juego.
- Los modelos de los personajes, entornos en donde se desarrolle el juego y otros elementos gráficos deben tener un estilo similar de modo que aparenten pertenecer a un mismo universo.
- Tener pendiente que los modelos, texturas y efectos visuales deben tener un estilo un tanto caricaturesco.
- La calidad de los sonidos (efectos de sonido y música) debe ser la mejor posible, esto es para garantizarle una buena experiencia audiovisual al usuario.
- Adicionalmente, como serán manipulados por el marco de desarrollo, los elementos de audio, deberán ser claro y agradables.

3.2 Requisitos de Rendimiento

El modo de juego de multi-jugador del videojuego (PLATKILL) deberá poder soportar diferentes salas en donde se podrá tener un máximo de 4 jugadores por sala. Esto podría cambiar. La latencia dentro de la red de PLATKILL deberá ser lo más baja posible, preferiblemente mantener una latencia promedio de 100 milisegundos entre los paquetes, de lo contrario el juego posiblemente no se vea fluido. Esto último es difícil de controlar puesto que en esto se involucran la capacidad de las redes en donde se ejecute el juego.

En cuanto a los requerimientos de un cliente del juego, el ordenador que los ejecute deberá contar como mínimo con lo siguiente:

- Sistema Operativo: Windows 7 o 8 de 32 bits.
- Memoria RAM: 512MB
- Procesador: 2 GHZ
- Disco Duro: 1GB disponible
- GPU: 128 MB, soporte para DIRECTX

3.3 Atributos del Sistema

En esta sección se listan los atributos de calidad con los que deberá contar el juego una vez creado. Los atributos que se buscan son:

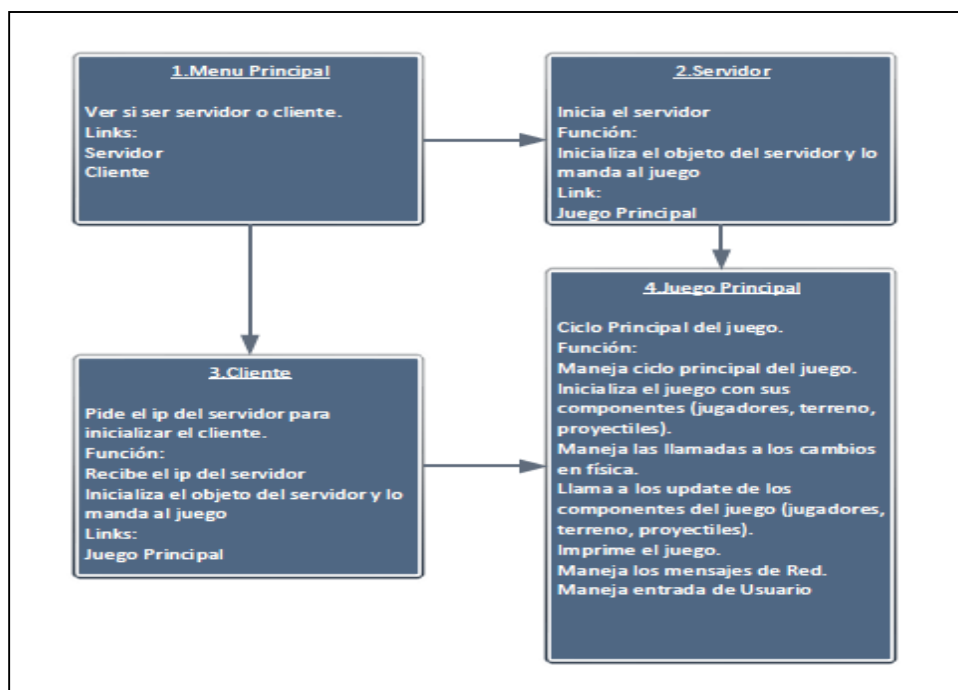
- **Usabilidad:** Las opciones del juego deben ser fácilmente entendibles por un individuo de cualquier edad. Y así mismo el diseño del juego debe ser muy captable por todos.
- **Disponibilidad:** El modo de multi-jugador deberá de estar disponible la mayor cantidad de tiempo posible de forma que esté disponible a cualquier hora que un jugador desee acceder a una partida.
- **Performance:** El cliente del juego no deberá consumir más recursos de los necesarios. Se pretende emplear los recursos que son planteados en la pasada sección de este documento. Además, los tiempos de respuesta de las acciones que realice un usuario deben ser mínimos.

4 Anexos

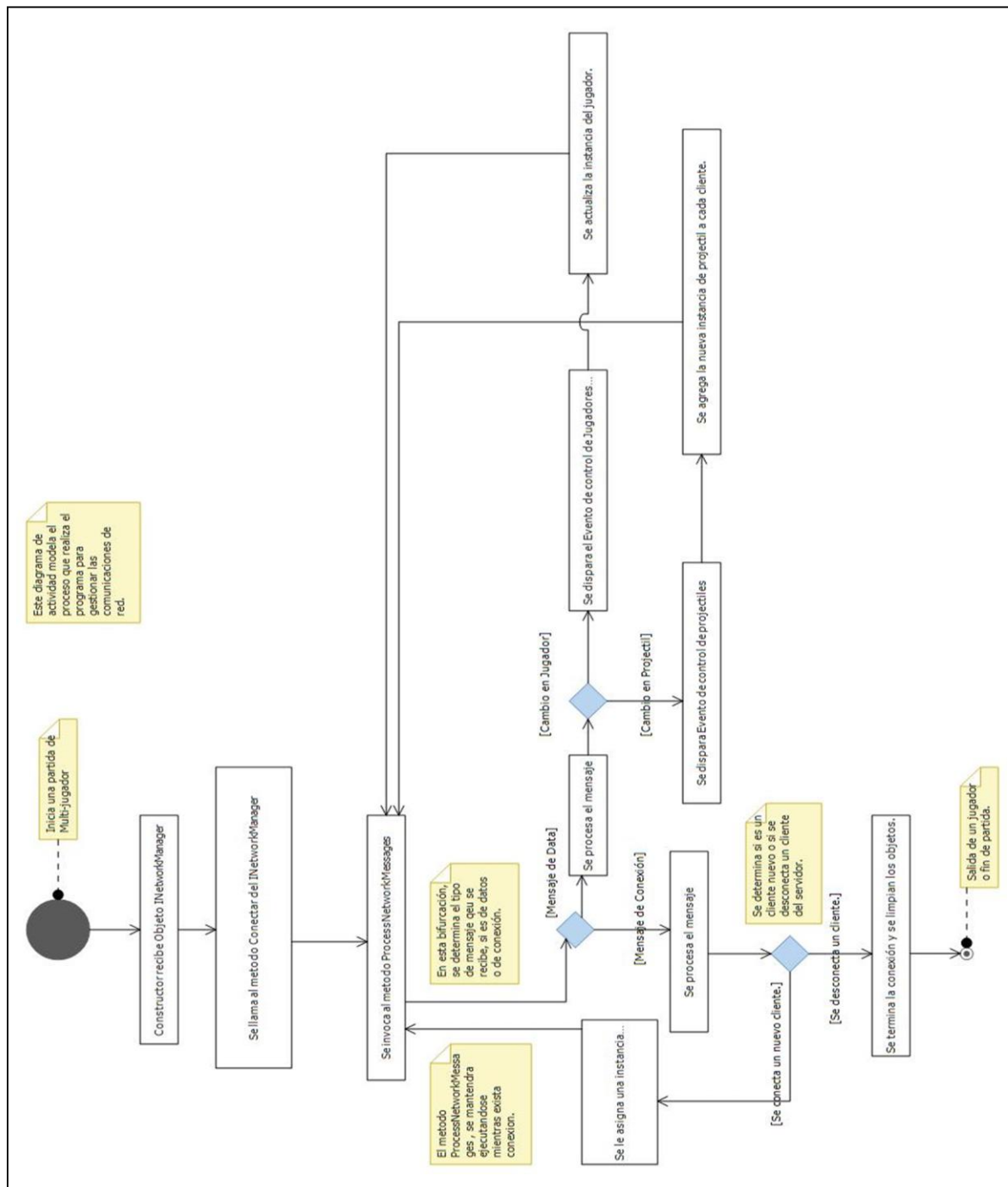
En la presente sección de anexos se muestran algunas imágenes correspondientes a los diagramas que son producto de la fase de diseño del sistema o producto.

4.1 Diagramas UML

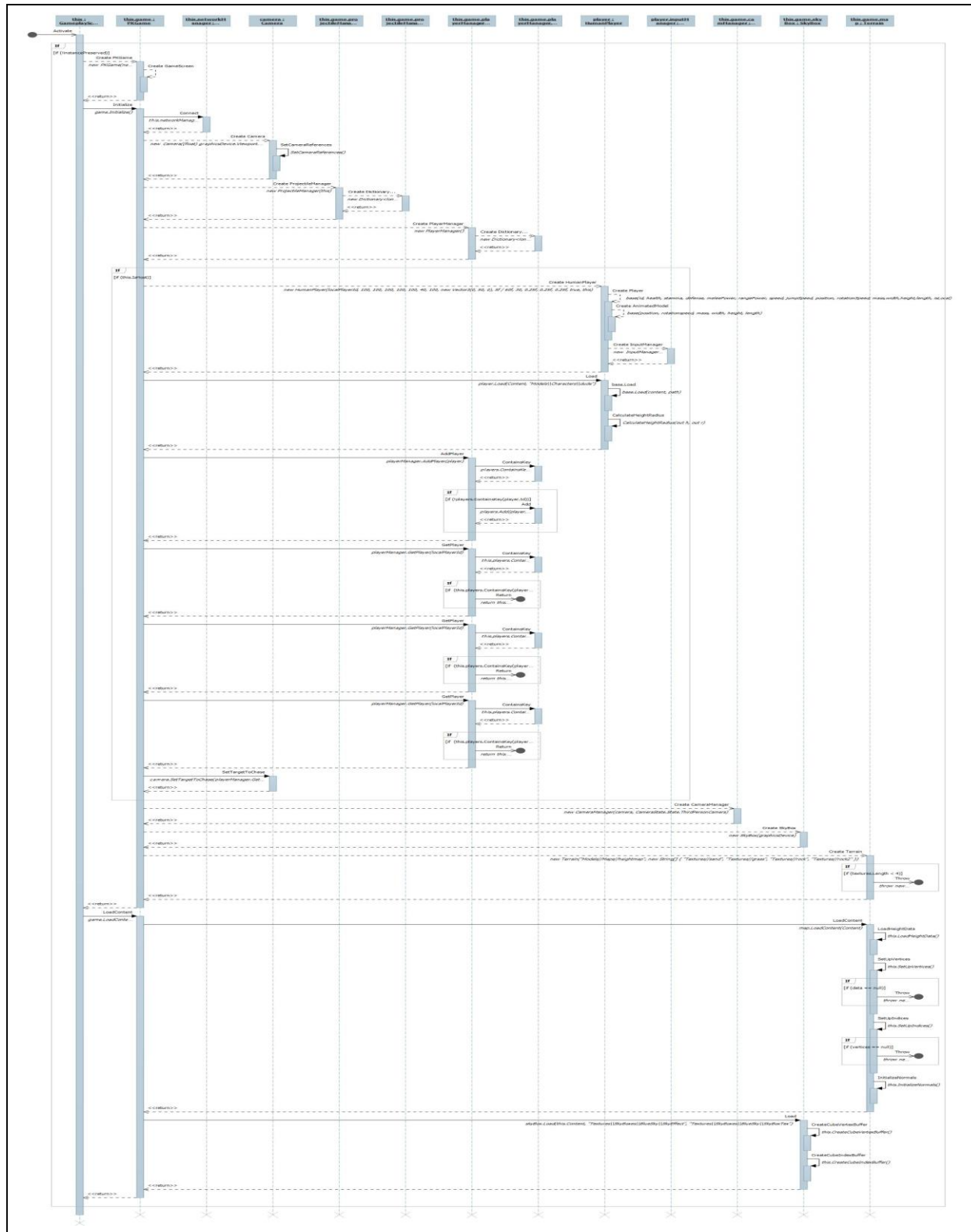
4.1.1 UML – Caso de Uso



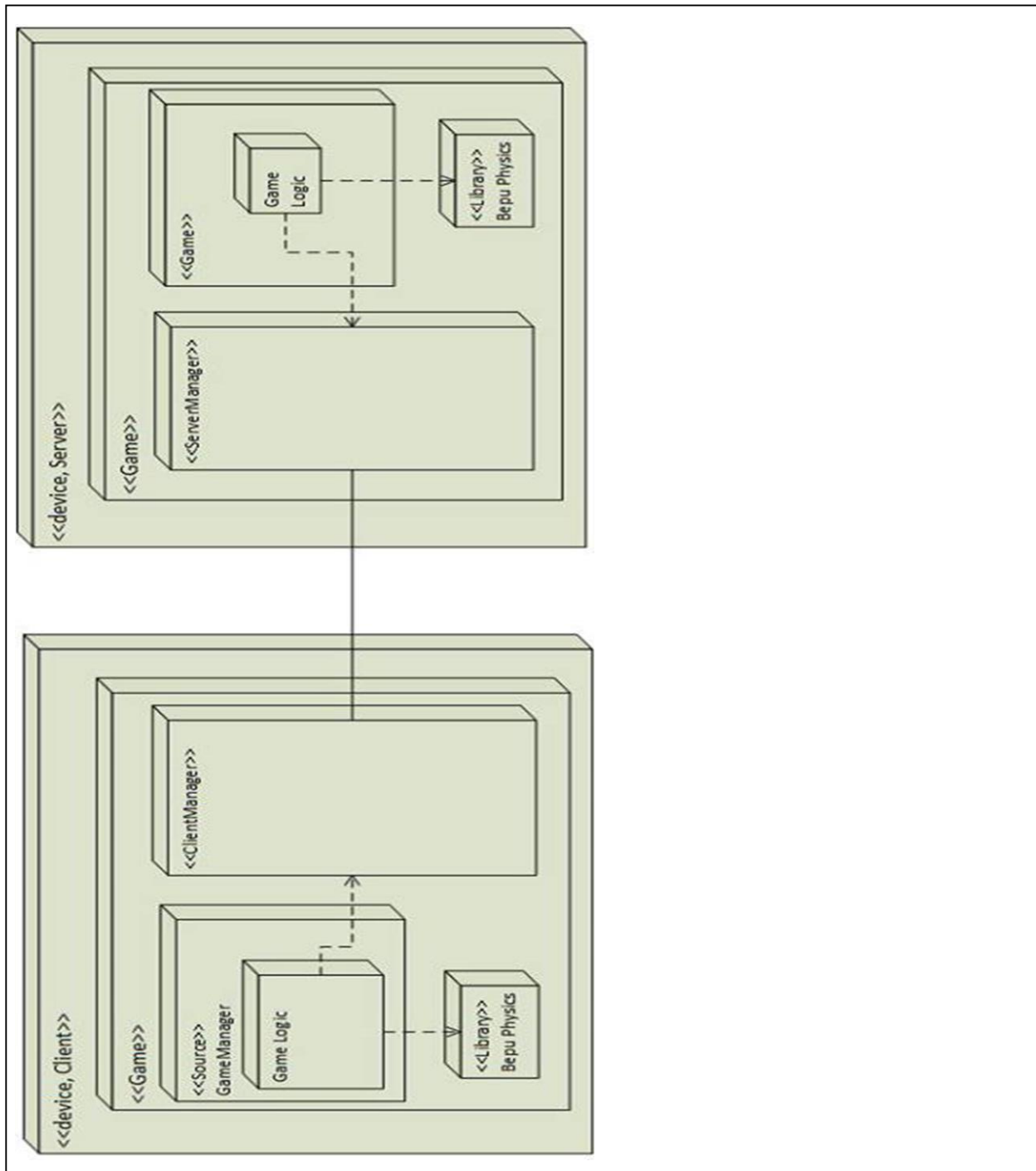
4.1.3 UML – Diagrama de Actividades



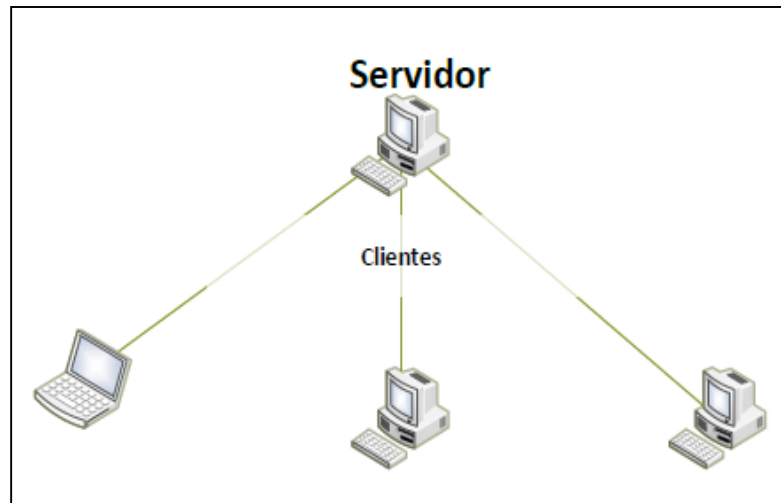
4.1.4 UML – Secuencia



4.2 Diagrama de Despliegue



4.3 Diagrama Físico



4.4 UED – Diseño de Ambiente de Usuario

