

UNIVERSIDAD NACIONAL AUTONOMA DE NICARAGUA, MANAGUA  
RECINTO UNIVESITARIO "RUBEN DARIO"  
FACULTAD DE CIENCIAS E INGENIERIAS  
DEPARTAMENTO DE COMPUTACION



TEMA:

**Programación de Videojuegos**

SUBTEMA:

**“Desarrollo de un Videojuego de perspectiva tercera persona, en ambiente 3D para PC, en la plataforma Windows, en el primer semestre 2013”**

PRESENTADO POR:

Br. José Ramón Duran Ramírez

Br. Meyling A. Lara Narváez

Br. Jaeddsen Jeannick Sánchez Arana

TUTOR:

Msc. Juan De Dios Bonilla

MANAGUA, NICARAGUA 31 DE OCTUBRE DEL 2013



## **Dedicatoria**

*Dedicamos este proyecto a nuestros padres, por su paciencia y apoyo incondicional, por brindar los medios para llevar este proceso de aprendizaje a término y por ser símbolo de superación y entrega; a nuestros amigos y compañeros por motivarnos con su energía; y demás personas que directa o indirectamente hicieron posible la culminación de esta etapa.*

## **Agradecimientos**

Con la realización de este proyecto se pone punto y final a todo un largo camino que ha sido la carrera de Licenciatura en Computación, mediante estas líneas queremos agradecer todo el apoyo que hemos tenido para conseguir finalizar.

Primeramente queremos dar las gracias a Dios por nuestra salud y por darnos fortaleza para salir adelante cada día.

A nuestros padres, por todo su amor y apoyo incondicional.

A Msc. Juan de Dios Bonilla, tutor del proyecto, por todos sus consejos y tiempo invertido en ayudarnos a terminar con éxito este proyecto fin de carrera.

A Darwin Rocha por su apoyo brindado a lo largo del proyecto.

Al Lic. Luis Miguel Martínez por su colaboración en el desarrollo metodológico del proyecto.

Y a las demás personas que hicieron posible llevar a buen término este trabajo de grado.

## **Resumen**

Este documento de tesis titulado desarrollo de videojuegos, introduce al lector en los conceptos básicos y la historia de los videojuegos desde sus inicios hasta la época actual, haciendo un recuento de los juegos más representativos durante los comienzos de esta industria.

El documento muestra contenido acerca de género de videojuegos, su clasificación y la posición de la computadora personal PC como plataforma de juego.

Los objetivos propuestos son el desarrollo de un videojuego en todos sus aspectos desde elaborar el guión, diseñar un escenario hasta establecer las mecánicas de juego e implementarlas en la aplicación

En este documento está plasmado el desarrollo de un videojuego de disparos en perspectiva de tercera persona para PC y desarrollado en el motor para videojuegos Unreal Development Kit, describiendo la metodología utilizada su desarrollo y los resultados obtenidos.

## Contenido

I.	Introducción .....	1
II.	Justificación .....	2
III.	Objetivos .....	3
IV.	Marco Conceptual .....	4
4.1	Videojuegos.....	4
4.1.1	Concepto de videojuego.....	4
4.1.2	Historia de los videojuegos .....	4
4.1.3	Géneros de videojuegos.....	22
4.1.4	Sistema de clasificación de contenido de videojuegos .....	45
4.2	Videojuegos para la computadora personal (PC) .....	52
4.2.1	Industria de videojuegos para PC presente y futuro.....	53
4.2.2	Hardware para Videojuegos en PC .....	56
4.2.3	Computación Grafica .....	60
4.3	Game Engines-Motores de videojuegos.....	79
4.3.1	Historias de los motores de videojuegos .....	80
4.3.2	Game Middleware .....	82
4.3.3	Tendencias actuales .....	83
4.3.4	Tipos de Motores .....	84
4.3.5	Lista de motores .....	85
4.3.6	Unreal Engine.....	88
4.4	Ingeniería del Software.....	97
4.4.1	Procesos para el desarrollo de Software.....	98
4.4.2	Un proceso para el desarrollo de videojuegos: “Huddle” .....	99
V.	Diseño metodológico.....	110
5.1	Tipo de estudio .....	110
5.2	Estudio de factibilidad .....	110
5.2.1	Factibilidad técnica.....	110
5.2.2	Factibilidad Económica.....	111
5.3	Proceso metodológico .....	112
5.3.1	Fase de preproducción del videojuego .....	113
5.3.2	Fase de Producción del videojuego.....	114

5.3.3	Fase Postmorten .....	115
5.4	Desarrollo del Videojuego .....	115
5.4.1	Diseño de niveles .....	115
5.4.2	Creación de materiales .....	115
5.4.3	Personaje y enemigos .....	116
5.4.4	Programación de eventos .....	116
5.5	Diagramas del Lenguaje unificado de modelado .....	118
5.5.1	Diagramas de casos de usos .....	118
5.5.2	Diagramas de secuencia .....	118
5.5.3	Diagrama de clases del videojuego .....	119
VI.	Resultados .....	120
VII.	Conclusiones .....	180
VIII.	Recomendaciones .....	181
IX.	Bibliografía .....	182
X.	Anexos .....	185
10.1	Cotización del equipo .....	185
10.2	Manual de usuario de videojuego Soul of wars .....	186

## Indice de tablas

Tabla 1. Clasificaciones normales de ESBR .....	47
Tabla 2. Clasificaciones Restringidas de ESBR .....	48
Tabla 3. Otras clasificaciones sin restricciones de ESBR .....	49
Tabla 4. Clasificación en desuso de ESBR .....	49
Tabla 5. Tipos de memoria para tarjetas gráficas vigentes en el mercado .....	58
Tabla 6. Motores Free y Open Source .....	86
Tabla 7. Motores Freeware .....	87
Tabla 8. Motores Propietarios .....	88
Tabla 9. Plantilla del Documento de Diseño .....	106
Tabla 10. Plantillas de los Documentos de Producción .....	108
Tabla 11. Plantilla del Reporte Postmortem .....	109
Tabla 12. Sprint Plan .....	155
Tabla 13. Feature Log .....	156
Tabla 14. Sprint Backlog .....	157



## Indice de gráficos

Imagen 1. Usuarios de videojuegos por región y plataforma .....	54
Imagen 2. Ingresos globales videojuegos 2012 .....	55
Imagen 3. Entorno Unreal Engine .....	89
Imagen 4. Componentes de Unreal Engine .....	90
Imagen 5. Interprete UnrealScript .....	92
Imagen 6. Las tres fases del proceso Huddle .....	99
Imagen 7. Modelo de fase Preproducción.....	100
Imagen 8. Modelo de fase Producción Huddle.....	102
Imagen 9. Modelo de fase Postmortem.....	104
Imagen 10. Etapas del proceso metodológico utilizado .....	112
Imagen 11. Resultados en la fase de preproducción .....	111
Imagen 12. Resultados en preproducción .....	114
Imagen 13. Resultado de la fase Postmortem .....	115
Imagen 14. Mapa de nivel 1: Cárcel vista plano.....	158
Imagen 15. Mapa de nivel 1: Cárcel vista modo juego.....	158
Imagen 16. Mapa de nivel 2: Laboratorio vista plano .....	159
Imagen 17. Mapa de nivel 2: Laboratorio vista modo juego .....	160
Imagen 18. Mapa de Nivel 3: Campo militar vista plano .....	161
Imagen 19. Mapa de Nivel 3: Campo militar vista modo juego.....	162
Imagen 20. Materiales creados .....	163
Imagen 21. Personaje .....	163
Imagen 22. Función de cambio a tercera persona .....	164
Imagen 23. Creación de enemigos.....	164
Imagen 24. Función abrir puerta con llave .....	165
Imagen 25. Función ascensor .....	166
Imagen 26. Animaciones .....	167
Imagen 27. Caso de uso jugador ejecuta el videojuego .....	168
Imagen 28. Diagrama de secuencia jugador ejecuta el videojuego .....	173
Imagen 29. Diagrama de secuencia jugador enfrenta enemigo .....	174

Imagen 30. Diagrama de secuencia jugador y puerta con llave enemigo .....	175
Imagen 31. Diagrama de clases del Videojuego .....	176

## **I. Introducción**

El creciente aumento del uso de dispositivos tanto PC, como móviles inteligentes, tabletas, consolas etc., ha dado pie al auge de la industria de los videojuegos, hoy está presente en la industria del entretenimiento comercial compitiendo a la par del cine y de otros tipos de entretenimiento.

El desarrollo de videojuegos es la actividad por la cual se diseña y crea un videojuego, desde el concepto inicial hasta el videojuego en su versión final. Ésta es una actividad multidisciplinaria, que involucra profesionales de la informática, el diseño, el sonido y comercio.

Cabe señalar que mundialmente hay una oferta en la parte creativa de videojuegos que está creciendo de forma acelerada. En Nicaragua poco o nada se puede recapitular acerca de este tipo de desarrollo de software, reduciéndose a unos cuantos proyectos de desarrolladores independientes y ninguna empresa que sobre salga en el mercado nacional.

El presente trabajo pretende incursionar en este tipo de aplicaciones, consistiendo en el desarrollo de un video juego para PC., dando un aporte innovador en un área del desarrollo comercial relativamente nueva en Nicaragua.

## **II. Justificación**

A través del desarrollo de este proyecto se pretende contribuir a la mejora de la oferta laboral en el área de programación en Nicaragua, ya que muchas de las empresas dedicadas al desarrollo de software están enfocadas en ciertos segmentos específicos de esta área como lo es el desarrollo de sistemas de información y desarrollo web y han dejado de un lado el campo de los videojuegos.

En la actualidad hay más de 1,000 millones de jugadores de PC en todo el mundo, un número que sigue creciendo a medida que más de estos equipos se conectan a la red, lo que crea una potencial oportunidad de negocios, que nuestro país perfectamente pudiera aprovechar.

Por esta razón, se ha tomado como plataforma de aplicación la computadora Personal y se ha escogido el género de disparos en perspectiva de tercera persona, por ser uno de los más populares y comercializados en esta industria.

Con este proyecto se pretende sentar un precedente que cambie el universo común de la comunidad de programadores en nuestro país.

### **III. Objetivos**

#### **Objetivo general**

- Desarrollar un videojuego en tercera persona, ambiente 3D para computadora Personal y Sistema Operativo Windows.

#### **Objetivos específicos**

- Elaborar el guión del videojuego Soul of Wars
- Diseñar ambiente y escenario del Videojuego
- Programar las mecánicas de juego basadas en el guión

## **IV. Marco Conceptual**

### **4.1 Videojuegos**

#### **4.1.1 Concepto de videojuego**

Un video juego es un programa creado para divertir o educar, basado en la interacción entre una persona y un aparato electrónico. Estos recrean entornos virtuales en los cuales el jugador puede controlar a un personaje o cualquier otro elemento de dicho entorno con un objetivo y unas reglas determinadas. (Dominguez Marin, 2007)

#### **4.1.2 Historia de los videojuegos**

Los videojuegos, como todo lo ligado a la tecnología ha tenido una meteórica y abrumadora evolución en muy pocos años, donde hemos pasado de apenas unos rudimentarios medios disponer de una tecnología capaz de recrear realidades virtuales asombrosas. (González, 2011)

Hay numerosos debates acerca de quién creó el primer videojuego, la respuesta depende en gran medida de la definición de videojuego. La evolución de los videojuegos presenta cruces entre varias industrias, incluyendo la científica, la informática, la industria del entretenimiento y la electrónica de consumo. La siguiente información muestra una línea de tiempo con los sucesos más importantes en la historia de los videojuegos desde sus comienzos hasta la actualidad.

#### **1947: Dispositivo de entretenimiento de tubos de rayos catódicos.**

El más antiguo juego electrónico interactivo conocido fue creado por Thomas T. Goldsmith Jr. y Estle Ray Mann en un tubo de rayos catódicos. El juego era un simulador de misiles inspirado por los radares usados en la Segunda Guerra Mundial. Utilizaba circuitos analógicos, no digitales y las superposiciones de pantalla fueron utilizadas para los objetivos ya que los gráficos no estaban

disponibles en ese momento. Empleaba ocho tubos de vacío para simular el lanzamiento de un misil a una serie de objetivos y disponía de un par de mandos para ajustar la velocidad y la curva de la trayectoria. Los objetivos estaban directamente pintados sobre el tubo, que sólo mostraba un punto: el misil. Sin duda su mecánica jugable lo acerca a lo que sería un videojuego, pero no se le puede considerar como tal porque no mostraba ningún gráfico en pantalla. (Herrera, 2010)

### **1950: Se publica el diseño de un programa para jugar ajedrez**

Durante la Segunda Guerra Mundial, el prestigioso matemático británico Alan Turing y el experto en computación estadounidense Claude Shannon trabajaron juntos descifrando los códigos secretos usados por el ejército nazi con la máquina Enigma y con los codificadores de teletipos FISH. Fruto de esta unión, Turing y Shannon junto a otros matemáticos como como Alonzo Church y Kurt Gödel establecieron las bases de la teoría de la computación, donde señalaban la Inteligencia artificial como el campo más importante hacia el que había que dirigir todos los esfuerzos de investigación en un futuro.

Gracias a estas investigaciones en el campo de la inteligencia artificial, en marzo del 1950 Claude Shannon presentó junto al economista y matemático D. G. Champernowne un artículo llamado “Programming a Computer for Playing Chess” en la revista Philosophical Magazine, en dicho artículo se especificaban las primeras técnicas y algoritmos necesarios para crear programas de ajedrez, pero en ese momento no había ningún ordenador con el suficiente potencial para poder ejecutarlo. (Lutznern8, 2013)

### **1951: Creación del concepto de videojuego**

Un joven ingeniero de televisión de 29 años llamado Ralph Baer trabajó en Loral, una compañía de televisión. Su jefe el ingeniero Sam Lackoff le pidió construir el mejor aparato de televisión en el mundo. El diseño de un televisor era una tarea fácil para Ralph y quería añadir un nuevo concepto que su jefe no entendía: jugar en la televisión. El concepto del videojuego nació, pero no pudo ser implementado

desde que el jefe rechazó la idea. Por lo tanto, Ralph Baer en consecuencia se acredita como el inventor del videojuego, en muchas ocasiones es nombrado “El Padre de los videojuegos” siendo reconocido por sus grandes contribuciones a los juegos y la industria de los videojuegos. (Winter, 2013)

### **1951: La recreación del juego Nim en un dispositivo de nombre Nimrod**

En mayo de 1951, el australiano John Bennett de la compañía Ferranti International, una empresa de ingeniería eléctrica que además comercializó el primer computador de la historia (Ferranti Mark I en 1951), presenta la computadora Nimrod en el Festival Británico, una exposición nacional para mostrar avances en distintas disciplinas como la ciencia o la arquitectura. Sus creadores plantearon la recreación del juego Nim en un dispositivo de nombre Nimrod. El Nim es un juego de estrategia (se cree que de origen chino) en el que dos jugadores se turnan para retirar objetos dispuestos en varios montones diferentes y en el que, al final, pierde el jugador que acaba retirando el último objeto del tablero. El computador jugaba al Nim contra un ser humano y lógicamente tenía programado el algoritmo de resolución del juego. Nimrod utilizaba un panel con luces a modo de display y en el que se representaban los "montones" y cada luz simbolizaba un objeto, al carecer de pantalla, es considerado como un juego electrónico y no un videojuego, pero fue la primera computadora diseñada específicamente para jugar un “juego”, de ahí su importancia en la historia. (Velasco, Nimrod, un videojuego de 1951, 2011)

### **1951: Un programa de ajedrez en una computadora**

Fue hasta en noviembre de 1951, el inventor alemán Dietrich Prinz pudo escribir el programa original de ajedrez (Publicado marzo del 1950 junto D. G. Champernowne en la revista Philosophical Magazine) (Lutznier8, 2013), este fue el primero que podía ejecutarse en una computadora electrónica real la Ferranti Mark 1. Este programa carecía de la posibilidad de jugar un juego completo de ajedrez, Prinz creó un programa limitado que podía encontrar la mejor jugada en un juego de ajedrez solucionando problemas que terminaban en jaque mate en



dos jugadas. Al ser una partida simulada y no disponer de gráficos, tampoco se puede considerar como el primer videojuego, aunque tiene un enorme valor de cara a la evolución de este medio. (PC World, 2013)

### **1952: X y 0**

Como parte de su tesis sobre la interacción hombre-máquina, Alexander S. Douglas programó el primer juego de computadora en usar una pantalla gráfica digital: X y 0 el programa permitía enfrentar un jugador humano contra la máquina EDSAC (Electronic Delay Storage Automatic Calculator) en la Universidad de Cambridge. X y 0 no tuvo ningún tipo de popularidad ni repercusión porque aparte de ser una tesis de investigación, solo podía ser jugado en la EDSAC en Cambridge, pues era la única que existía en todo el mundo. Es el primer juego de computadora en usar una pantalla gráfica digital, por lo que al mostrar gráficos, tiene todos los requisitos para poder considerarlo como el primer videojuego de la historia: una maquina (EDSAC), un mando (se jugaba con un dial de teléfono), una pantalla y un programa. Sin embargo, hay algunas voces discordantes con tal afirmación, pues prefieren considerarlo como un programa gráfico por ordenador ya que no tenía vídeo en movimiento. Quizá el gran desconocimiento y poca importancia de ser “solo” una tesis es lo que ha provocado la relegación de X y 0 frente al que la mayoría considera oficialmente el primer videojuego de la historia. (Lutznner8, 2013)

### **En 1958: Tennis for Two**

Después de todos estos primeros pasos, llegó el momento para Tennis for Two, el considerado oficialmente como primer videojuego de la historia pese a las discrepancias existentes con el X y 0. Willy Higginbotham era un reconocido físico que trabajaba en los laboratorios Brookhaven National, Higginbotham, decidió hacer algo entretenido para los visitantes en ocasión de una celebración anual de la compañía llamada el día anual del visitante, para ello creó un sencillo juego de tenis utilizando un osciloscopio, una computadora analógica y unos botones muy básicos. Así nació Tennis for Two, fue el 18 de Octubre del año 1958 que el juego

fue expuesto, usaba una perspectiva lateral en la que se podía observar una línea horizontal que representa el campo de juego y otra pequeña vertical en el centro del campo que hacía de red. Pese a las colas que se formaran el día de su presentación y las buenas críticas recibidas, Tennis for Two fue creado como un simple experimento para entretener a los visitantes del laboratorio, así que un año más tarde, en 1959, se realizó una última sesión de puertas abiertas al público antes de ser desmontado para poder usar sus piezas en proyectos más serios y de mayor importancia. (Winter, 2013)

### **1959: Mouse in the Maze, Tic-Tac-Toe (X y O)**

En este año una colección de programas gráficos interactivos fueron creados para la computadora experimental TX-0 en el MIT. Éstos incluían el Mouse in the Maze (Ratón en el laberinto) y el Tic-Tac-Toe (X y O). Mouse in the Maze permitía a los usuarios utilizar un lápiz óptico para colocar muros en el laberinto, puntos que representaban bits de queso y (en algunas versiones) vasos de martini. Un ratón virtual representado por un punto era entonces liberado y se movía a través del laberinto para encontrar los objetos. Tic-Tac-Toe usaba el lápiz óptico para jugar un juego simple de ceros y cruces contra la computadora. (Donovan, 2010)

### **1961: SpaceWar**

En el año 1961, Steve "Slug" Russell, Martin "Shag" Graetz y Wayne Wiitanen, idearon un juego para computador que querían desarrollar en la PDP-1 del MIT (el primer computador fabricado en serie por la compañía Digital Equipment). Después de 200 horas de trabajo Steve, fue capaz de escribir el código de la primera versión de un videojuego, el Spacewar!, que se jugaba en un monitor CRT que se usaba, en principio, para testear algunas funciones en el computador. La primera versión del juego se terminó en febrero de 1962 e incluía un fondo estrellado que se generaba de manera aleatoria. El juego se hizo muy popular en el MIT y se fueron añadiendo nuevas características por parte de otros estudiantes, el juego siguió popularizándose y otros centros de investigación usando el código fuente original trabajaron en sus propias variantes. Tal fue el

éxito del juego que el fabricante del PDP-1 incluyó en su fabricación el código del juego, algo que siguió manteniendo en el PDP-10 y en el PDP-11. Spacewar fue uno de los primeros juegos que se incluyeron de serie en un computador de propósito general. (Velasco, Historia de la tecnología: Spacewar! el videojuego que nació en el MIT, 2011)

### **1966: Fox y hounds**

En ese año Ralph Baer trabajaba como diseñador jefe de Sanders Associates, una empresa que trabajaba para el ejército, Baer reconsideró una idea que había abandonado unos años antes: un dispositivo que, conectado a un simple televisor, permitiese jugar al espectador con su aparato. Ralph Baer empezó a desarrollar junto a Albert Maricon y Ted Dabney, un proyecto de videojuego llamado “Fox and Hounds” dando inicio al videojuego doméstico. En marzo de 1967 finalizaron un primer prototipo que incorporaba ya una serie de juegos, entre los que se encontraban el ping-pong y un juego para dos jugadores en el que ambos debían acorralar al contrario. Baer y sus colaboradores también diseñaron un rifle que, conectado al dispositivo, permitía disparar a una serie de objetivos. A finales de 1967 el proyecto estaba casi completado y atrajo la atención de TelePrompster Corporation, una compañía de televisión por cable cuyos ejecutivos habían visto el aparato durante una visita a las instalaciones de Sanders. Tras unas negociaciones que duraron dos meses no se llegó a ningún acuerdo, y las ideas de Baer fueron relegadas por segunda vez al olvido. (Sellers, 2001)

### **1971: Galaxy Game**

A finales de la década de los 60 Bill Pitts, un estudiante de la Universidad de Stanford fascinado por Spacewar! tuvo la idea de hacer una versión del juego que funcionase con monedas para su explotación en los salones recreativos. Junto a Hugh Tuck, un amigo de la secundaria se unieron para construir un prototipo. En 1971 ambos formaron Computer Recreations, Inc con el propósito de construir una versión operada con monedas de Spacewar, tras tres meses y medio de trabajo habían finalizado la máquina, pero decidieron cambiar el título del programa a

Galaxy Game, la máquina fue instalada en junio de 1972 en el Coffe House de Tresidder Union, cerca de la Universidad de Stanford, y allí permaneció con bastante éxito hasta 1979. (Kent, 2001)

### **1971: Computer Space**

Nolan Bushnell, al igual que Bill Pitts (Galaxy Game), había descubierto Spacewar durante sus días de estudiante en la Universidad de Utah, tenía un interés genuino en el negocio del entretenimiento, en el verano de 1971 Bushnell se asoció junto a otro ingeniero Ted Dabney bajo el nombre de Syzygy Engineering para presentar un prototipo a Nutting Associates, una empresa de aparatos electrónicos. Nutting mostró interés hacia el proyecto y se encargó de la fabricación de un primer modelo, usando una carcasa de diseño futurista. En noviembre de 1971, el primer Computer Space se instaló en el bar Dutch Goose, cerca del campus de la Universidad de Stanford y obtuvo un éxito inmediato entre los estudiantes, pero el sistema de control y el objetivo de las partidas resultaban muy complicados para el público no universitario, Bushnell y Dabney finalizaron su contrato con Nutting Associates y el 27 de junio de 1972, por problemas de derechos de autor cambiaron el nombre de su empresa por el de Atari. (Kent, 2001)

“Computer Space” fue el primer juego arcade de la historia., arcade es el término genérico de las máquinas recreativas de videojuegos disponibles en lugares públicos de diversión. (Sellers, 2001)

### **1972: “Magnavox Odyssey”**

Ralph Baer no había desistido en sus empeños, había continuado presentando su proyecto a distintas compañías para su posible comercialización. Bill Enders, trabajaba para Magnavox y había quedado impresionado con las primeras demostraciones de Baer, convenció a otros directivos de la empresa para que dieran una oportunidad. Magnavox firmó un acuerdo con Sanders Associates, la empresa para la que trabajaban Baer y sus colaboradores y se aprobó definitivamente la fabricación del producto. En abril de 1972 la firma presentó la

nueva máquina a la prensa y a sus distribuidores. Al mismo tiempo se presentó el primer accesorio de la máquina, un rifle de plástico de buena apariencia, y diez juegos adicionales, se consiguieron vender cerca de 130 000 unidades en la campaña de navidad, todo un éxito, que atrajo la atención de numerosos emprendedores. (Sellers, 2001)

### **1972: “Pong”**

La ascensión de los videojuegos llegó con la máquina recreativa Pong, el sistema fue diseñado por Al Alcom para Nolan Bushnell, que se convirtió en el primer título de la recién creada Atari, El juego se presentó en 1972 y fue la piedra angular del videojuego como industria. Pong se considera la versión comercial de Tennis For Two. De repente el país se encontraba inundado de máquinas Pong, así como de copias manufacturadas por compañías de la competencia. La japonesa Taito lanzó al mercado oriental su propia versión del juego, y lo mismo ocurrió en países como Francia o Italia. El enorme éxito de la máquina de Atari impulsó las ventas de la Odyssey, la consola que le había dado origen, y a finales de 1974 había cerca de 100 000 máquinas arcade solamente en Estados Unidos que generaban más de 250 millones de dólares anualmente. La industria de los videojuegos había nacido definitivamente (Sellers, 2001)

### **1973: El liderazgo de Atari**

Quince compañías se habían lanzado al negocio de los videojuegos, un negocio que un año antes estaba exclusivamente en manos de Atari. Los videojuegos de estas compañías no dejaban de ser simples copias de Pong, mientras que la compañía que había creado el juego original seguía aportando nuevas innovaciones, Atari comercializó nuevos éxitos como Space Race, Rebound o Gotcha (1973), Quadrapong, Touch Me, Tank, Qwak o Gran Trak 10 (1974), cada uno de los cuales suponía en la práctica la inauguración de un nuevo género. (Sellers, 2001)

### **1975: Atari Pong**

Tras la salida en arcade de Pong, Atari creó su primera consola: Atari Pong. El nuevo producto, que permitía jugar a Pong en casa, lo cual por aquellos tiempos era lo más parecido a llevarte la máquina recreativa a casa. Aun así la idea no fue muy bien acogida ni rentable al disponer de un sólo y único juego sin ninguna otra posibilidad. (Garcia, 2012)

### **1977: Atari 2600**

Atari lanzó la consola que le propulsó directo al éxito: Atari 2600. Entre sus novedades se encontraba la innovación de poder cambiar de juego por medio del intercambio de cartuchos. Sólo tenía 8 bits de potencia, pero eran más que suficientes para aquella época y para combatir la competencia. Se mantuvo al pie del cañón liderando a sus rivales durante muchos años gracias a su amplio catálogo de juegos. Unos años después de su salida sacó una nueva versión junior que remató la jugada. Con su éxito, Atari se permitió comprar las licencias de películas tales como E.T o Indiana Jones, lo que le aseguró el puesto. Se mantuvo en el mercado hasta el año 1990 por lo que se ha convertido en la consola hasta el momento, que más tiempo ha permanecido activa en el mercado. Destronada por la NES (Nintendo). (Lutznern8, 2013)

### **1978: Magnavox-Odyssey2**

Magnavox-Odyssey2 se dispuso a combatir cara a cara con Atari 2600. Fue su rival más fuerte, pero nunca alcanzó la popularidad de la 2600. La gran novedad de esta consola, única en la generación de 8 bits, era incluir un teclado emulando un ordenador. (Lutznern8, 2013)

### **1980-87: La década de los 8 bits**

Los años 80 comenzaron con un fuerte crecimiento en el sector del videojuego alentado por la popularidad de los salones de máquinas recreativas y de las primeras videoconsolas aparecidas durante la década de los 70. Por otro lado en las máquinas recreativas triunfaron juegos como el famoso Pacman. El mercado

de videojuegos desarrollado en la década de 1980 estuvo caracterizado por la atmósfera de creatividad que reinaba una industria que todavía estaba dando sus primeros pasos. (Facultad de Informatica de Barcelona, 2010)

A principios de 1980 apareció en Europa el Sinclair ZX80, un nuevo invento del británico Clive Sinclair que prometía revolucionar el mercado. En 1981 Sinclair sacó al mercado su ZX81, un modelo mejorado que incorporaba 1Kb de RAM (ampliables a 64) y que por su reducido precio, supuso la punta de lanza para la penetración definitiva de los ordenadores personales en los hogares europeos. Los primeros juegos para el sistema no tardaron en aparecer y posteriormente salió a la venta el ZX Spectrum y que se establecería como uno de los ordenadores más populares de Europa. (Kent, 2001)

En 1983 comenzó la llamada crisis del videojuego, afectando principalmente a Estados Unidos y Canadá ocasionada por la saturación del mercado con numerosos títulos de muy baja calidad, llegando éstos a ocasionar una notable reducción en las ventas de videojuegos y una enorme pérdida de la confianza en los clientes. La crisis duró aproximadamente dos años, llegando incluso a sembrar numerosas dudas sobre la viabilidad a largo plazo de la industria del videojuego. (Consalvo, 2006)

En 1983 se lanza en Japón la Famicom, un modelo muy innovador que disponía entre su catálogo de varios títulos clásicos, que dieron lugar a algunas de las más famosas series de personajes de los videojuegos, y que estaba destinado a sentar las bases del mundo de las videoconsolas domésticas modernas, la consola catapultó a Nintendo al éxito.

A partir de 1985 se produjo el asalto de Nintendo al mercado estadounidense tras la crisis del videojuego de 1983. La Famicom sufrió varias modificaciones, siendo rebautizada como NES (Nintendo Entertainment System), y como punta de lanza de su catálogo de videojuegos se presentó la nueva creación de Shigeru Miyamoto, Super Mario Bros cambiando el curso de la historia, vendiendo no sólo millones de copias, sino también las millones de consolas que eran necesarias

para jugarlo. Este es el primer video juego que rompe con la estructura de bucle de pantalla, tan solo basada en obtener una buena puntuación.

Súper Mario Bros supone un estallido de creatividad, rápidamente emulado por otras compañías, que dará lugar a los videojuegos que conocemos hoy día. Para 1987 había quedado claro que el mercado de las videoconsolas había vuelto para quedarse centralizándose el desarrollo de videojuegos en Japón. (Kent, 2001)

### **Época de los 16 bits**

La cuarta generación de consolas, comúnmente conocida como la "era de los 16 bits" empezó el 30 de octubre de 1987 cuando la compañía japonesa Nippon Electric Company lanzó al mercado la consola PC Engine.

En 1988 Sega lanza su consola Sega MegaDrive en Japon, Rebautizada en Estados Unidos como Sega Genesis.

En 1991 Nintendo lanzó al mercado su consola SNES (Super Nintendo Entertainment System), obteniendo nuevamente un rotundo éxito.

En esta era los personajes de los juegos que habían impactado al público, se hicieron aún más populares, como Mario y Sonic the Hedgehog.

Otras compañías también lanzaron consolas al mercado pero no tuvieron éxito. Esta generación vivió un periodo de intensa guerra comercial, principalmente entre MegaDrive y el SNES.

En esta época, el catálogo de Juegos que poseía Nintendo se diversifico mucho más que en la época anterior, haciendo incursiones revolucionarias en los géneros de RPG (juegos de rol), plataformas, Aventuras, Lucha y carreras. (Kent, 2001)

### **Época del 3D: 32 y 64 bits**

Mientras existía un interés del público por la aparición de nuevas tecnologías como el CD-ROM, que puso un salto importante en la industria. Empiezan a aparecer nuevos géneros de videojuegos gracias a la puesta en marcha de



nuevas tecnologías, entre ellos los primeros desarrollos que buscaban un entorno tridimensional.

Se publican “WOLFESTEIN” y “DOM” para PC, los primeros videojuegos que intentan emular un entorno “tridimensional” mediante técnicas como el Ray-Casting. Apareciendo con ellos un nuevo género, FPS (First Person Shooter). Incluyendo adicionalmente una temática adulta, que rápidamente genera una polémica social.

Así mismo también en PC, aparecen los primeros videojuegos integros 3D como “4D Boxing”, y mezcla de 3D en tiempo real y prerenderizado como “Alone In The Dark”.

La industria de inmediato adopta los video juegos 3D creando la “generación de 32 bits” con las consolas Play Station de Sony, Saturn de Sega y la “Generacion de 64 bits” con Nintendo 64 y Atari Jaguar.

El Nintendo 64 fue la tercera videoconsola de sobremesa de Nintendo, desarrollado para suceder al Super Nintendo, esta incorpora en su arquitectura un procesador principal de 64 bits.

En cuanto a los juegos, el más reseñable por su éxito de crítica y público fue Super Mario 64. Desarrollado por Shigeru Miyamoto y lanzado junto con la consola, su sistema de juego sentó bases en mayor parte de los juegos 3D de su estilo. El otro título popular de la consola fue The Legend of Zelda: Ocarina of Time, también de Shigeru Miyamoto, primer juego en obtener una puntuación perfecta en la prestigiosa revista japonesa Famitsu y habitual ganador de premios a "mejor juego de la historia".

Aparecen las primeras tarjetas gráficas con aceleración 3D.

La consola de Sony apareció tras un proyecto iniciado con Nintendo (denominado SNES PlayStation), que consistía en un periférico para SNES con lector de CD. Al final Nintendo rechazó la propuesta de Sony, puesto que Sega había desarrollado algo parecido sin tener éxito, y Sony lanzó independientemente PlayStation.

Por su parte los arcade comenzaron un lento pero imparable declive según aumentaba el acceso a consolas y ordenadores más potentes. Por este motivo los fabricantes, idearon nuevos conceptos difícilmente adaptables a las casas, como pistas de baile y asientos de coche. Lo cual encareció notablemente los procesos de creación, motivo por el cual muchos fabricantes dejaron ese sector.

La popularización de internet y su accesibilidad, facilitaron los juegos multijugador via online, creándose importantes comunidades denominadas a menudo “clanes”. Siendo los responsables de la creación de un nuevo género, MMORPG (Massive Multiplayer Online Role-Playing Games), creándose Ultima Online de Origin y una gran comunidad asociada a lo que viene denominándose “Mundos Persistentes”.

En 1998 aparece una nueva consola, Dreamcast de Sega, iniciándose con ella la “generación de 128 Bits”. (González, 2011)

## **2000: Nuevo siglo**

Esta década está marcada como la plenitud de los videojuegos como industria, donde se supera a la industria del cine en cuanto a facturación en puntos de venta.

En la historia de los videojuegos, la era de sexta generación que fue de 1998 a 2009 y a veces referido como la era de 128 bits, se refiere a las computadoras, los videojuegos, consolas de videojuegos y dispositivos portátiles de vídeo juegos, disponibles a finales del siglo XX. (Melissinos, 2012)

Esta era comenzó el 27 de noviembre de 1998, con el lanzamiento en japon del Sega Dreamcast posteriormente PlayStation 2 en marzo de 2000 y el GameCube y Xbox en 2001. (Melissinos, 2012)

Dreamcast de Sega fue la primera consola de la generación y ha introducido una serie de innovaciones, incluyendo juegos en Internet como una característica opcional a través de su módem integrado y un navegador web. También fue la primera consola de sobremesa en mostrar máxima resolución SD (Standard Definition). (Melissinos, 2012)

Sony PlayStation 2 logra el dominio de ventas en esta generación, convirtiéndose en la consola más vendida en la historia alcanzando los 157.68 millones de unidades vendidas, La consola es también la más longeva de esta generación y la segunda en toda la historia de los videojuegos, alcanzando 12 años y 10 meses siendo descontinuada a principios de 2013.

La PlayStation 2 también fue capaz de reproducir DVD y era retro compatible con los juegos de PlayStation. Para esta consola se publicaron juegos clave que le permitieron superar a los lanzamientos de sus competidores. (Melissinos, 2012)

Nintendo Gamecube tuvo problemas en esta generación, particularmente por la imagen familiar e infantil que desarrolló durante la década de 1990. A pesar de ganar una base de fans leales, esta no pudo darle una ventaja frente a la Xbox y PlayStation 2, que capturó la audiencia que buscaba títulos más maduros que Nintendo no poseía. Nintendo también hizo pocos progresos en los juegos en línea, como resultado el Nintendo GameCube no logró igualar las ventas de su predecesora, la Nintendo 64. (Melissinos, 2012)

La Xbox de Microsoft atrajo a una gran base de fans y un fuerte apoyo de terceros en los Estados Unidos y Europa, convirtiéndose en una marca reconocida entre la corriente principal. Microsoft ofreció servicios en línea, para los usuarios de Xbox, estos servicios fueron nombrados como Xbox Live el cual resultó particularmente exitoso, lo que llevó a Sony a aumentar las capacidades en línea de la consola PlayStation 2. Xbox Live también dio a la Xbox una ventaja sobre la GameCube, que tenía una carencia casi total de juegos en línea. Su capacidad de conectar a los jugadores en partidos en línea, fue un factor importante en permitir al Xbox ganarse un lugar en el mercado occidental, especialmente en el género de primera persona (First Person Shooter). (Melissinos, 2012)

La Microsoft Xbox ha vendido más de 24.65 millones de unidades hasta que fue retirada del mercado en 2008. En Japón, las ventas de Xbox eran muy pobres, debido en parte a la incapacidad de Microsoft de atraer a los principales desarrolladores japoneses y a las franquicias de juegos, sumado a esto, el tamaño

de la consola física no encajaba en los estándares estéticos locales y la lealtad a las marcas japonesas como Sony y Nintendo fueron factores importantes también. (Melissinos, 2012)

Durante la sexta generación, la consola de juegos portátil tuvo un mercado muy amplio, pero dominado en su mayoría por Nintendo con el lanzamiento en 2001 de la Game Boy Advance, que contó con muchas mejoras y nuevas funciones en el Game Boy. (Melissinos, 2012)

El campo de los Juegos online en generaciones anteriores habían sido de dominio exclusivo de los juegos de PC, se hizo más prominente en las consolas de videojuegos durante esta generación. Dreamcast inició este cambio incorporando un módem, un software de navegación por Internet y la capacidad de jugar ciertos juegos en línea. La PlayStation 2, Xbox y GameCube también ofrecen juegos en línea. (Melissinos, 2012)

La sexta generación vio el inicio de su fin cuando el Xbox fue sucedido por el Xbox 360 a finales de 2005, Dreamcast fue discontinuado en 2001, GameCube fue discontinuado en 2007 y Xbox se interrumpió definitivamente en 2008. La PlayStation 2 aún se producía después de la sexta generación y se suspendió a principios de 2013. (Melissinos, 2012)

La séptima generación de videoconsolas es la generación que incluye las consolas lanzadas desde finales de 2005 por Nintendo, Microsoft y Sony. Comenzó el 22 de noviembre de 2005 con el lanzamiento de la Xbox 360 de Microsoft y continuó con el lanzamiento de la PlayStation 3 de Sony el 11 de noviembre de 2006 y de la Wii de Nintendo el 19 de noviembre de 2006 siendo la de mayor éxito la Wii de Nintendo. (Wolf, 2007)

Cada nueva consola introdujo un nuevo tipo de avance en la tecnología. La Xbox 360 ofrece juegos con resoluciones de alta definición. PlayStation 3 ofrece además de juegos de alta definición, la reproducción de contenido multimedia en alta definición también a través de su reproductor de Blu-ray Disc. Wii se centra en

la integración de controles con sensores de movimiento, así como palancas de mando. (Wolf, 2007)

Nintendo entró en esta generación con un nuevo planteamiento, la compañía planea atraer a los Jugadores más dedicados (Hardcore Gamers), a jugadores casuales y a no jugadores, centrándose en nuevas experiencias de juego y nuevas formas de interacción en lugar de gráficos de vanguardia y tecnología costosa. (Wolf, 2007)

La Xbox360 Fue desarrollada en colaboración con IBM y ATI. Como principales características, están su unidad central de procesamiento basado en un IBM PowerPC y su unidad de procesamiento gráfico que soporta la tecnología de Shaders Unificados. El sistema incorpora un puerto especial para agregar un disco duro externo y es compatible con la mayoría de los aparatos con conector USB gracias a sus puertos USB 2.0. Los accesorios de este sistema pueden ser utilizados en una computadora personal como son los controles y el volante. (Johnson, 2006) (Wolf, 2007)

El lanzamiento temprano de Xbox360 llegó con algunos problemas técnicos que aparecieron en una gran parte de las unidades vendidas. El problema más conocido es el "anillo rojo de la muerte " y error E74 que recibió y aun recibe una gran cantidad de atención debido a que una gran parte de usuarios han tenido que reemplazar las consolas múltiples veces. Microsoft trató de abordar este problema ofreciendo una garantía de tres años en todas las consolas afectadas y la reparación de forma gratuita. (Wolf, 2007)

PlayStation 3 fue la primera consola en utilizar discos Blu-ray como medio de almacenamiento primario. Una de las principales características de la consola es su servicio unificado de juegos en línea, la PlayStation Network (Wright, 2010).

Fue inicialmente disponible con discos duros de 20 y 60 GB, llegando a tener hasta 500 GB en modelos posteriores. Todos los modelos de PS3 tienen discos duros SATA de 2.5. (Wright, 2010)

La PlayStation 3 utiliza un microprocesador celular, diseñado por Sony, Toshiba e IBM, el procesamiento de gráficos está a cargo de la NVIDIA RSX Reality Synthesizer, que puede producir resoluciones de 480i / 576i SD hasta 1080p HD . La PlayStation 3 tiene 256 MB de memoria principal y 256 MB GDDR3 de memoria de video. (Wright, 2010)

Se prevé que la octava generación de videoconsolas se enfrentará a la dura competencia de los teléfonos inteligentes y tabletas. (Forster, 2011)

Oficialmente la octava generación de consolas inició el 18 de noviembre de 2012 cuando Nintendo sacó a la venta el Wii U. El control principal de la Wii U cuenta con una pantalla táctil incorporada que funciona como una pantalla interactiva auxiliar de manera similar a la Nintendo DS/3DS, o incluso como la pantalla principal en sí, permitiendo que los juegos sean utilizados sin la necesidad de una pantalla extra conectada a la consola. (Forster, 2011)

Una segunda consola de sobremesa, Ouya, fue anunciada. Será la primera videoconsola de Android y saldrá a la venta en 2013. (Forster, 2011)

El 20 de febrero de 2013, Sony anunció la PlayStation 4 en una conferencia de prensa en la ciudad de Nueva York y se dará a conocer durante el último trimestre de 2013. La nueva consola pone un fuerte énfasis en las características que rodean la interacción social. Contará con un procesador AMD. Otras características de hardware notables de la PS4 es que incluyen 8 GB de memoria unificada en forma de GDDR5, una unidad de disco Blu-ray Disc más rápido y los chips personalizados dedicados a tareas de procesamiento de audio, vídeo y de fondo. (Forster, 2011)

Entre las nuevas aplicaciones y servicios, Sony planea lanzar la aplicación PlayStation App, permitiendo a los que tengan un PS4 convertir los teléfonos inteligentes (Smartphones) y las tabletas en una segunda pantalla para mejorar la jugabilidad. La compañía también planea debutar con un servicio llamado Gaikai basado en la nube, que alojara contenidos y juegos descargables. La

incorporación del botón "compartir" en el nuevo control hará posible ver el juego transmitido en vivo en las redes sociales. (Forster, 2011)

El 21 de mayo de 2013 Microsoft anunció la Xbox One la cual contará con un mayor enfoque en el entretenimiento, incluyendo la capacidad de pasar la programación de televisión de un decodificador de señales a través de HDMI, utilizará una función de guía electrónica de programación y la capacidad de realizar múltiples tareas de manera similar a Windows 8. Poseerá una función de control de voz que permitirá a los usuarios controlar las funciones de Xbox a través de comandos de voz. Todo el control de voz se coordina a través del Kinect y con ello la capacidad de Skype también se convertirá en una función de la nueva Xbox. (Forster, 2011)

El control también contará con una zona sensible al tacto, similar a la función en el nuevo control de Sony PS4. El área de contacto se utilizará para complementar la barra de sensor Kinect y también podrá hacer clic. (Forster, 2011)

## **Resumen**

Todo lo que hemos descrito se podría resumir en que disfrutamos de una industria joven con cerca de 25 – 30 años de vida donde hemos vivido ocho generaciones de consolas hasta el momento.

Los videojuegos son un mercado de alto riesgo debido a los tiempos de producción, plataformas, variabilidad de los intereses del público y la competencia.

Desde el principio Nintendo aparece como la compañía con mayor número de logros, así como potencial creativo y comercial de nuevas tendencias.

En los últimos años hemos vivido por fin el gran cambio social de dejar de considerar el videojuego como un juguete a ser considerado como una plataforma de entretenimiento. Y también ha dejado de ser una opción de ocio para ser una oportunidad de negocio.

### **4.1.3 Géneros de videojuegos**

Los géneros de videojuegos son las categorías que se utilizan para clasificar y organizar a los videojuegos por elementos clave de su jugabilidad, tales como la forma de juego, los controles y el objetivo. A diferencia de otros medios como la literatura o el cine, los géneros de los videojuegos no toman en cuenta aspectos relacionados al argumento o la temática situacional de las historias presentadas. (Despain, 2009)

La clasificación por géneros es un sistema que se basa en las coincidencias que existen entre distintos videojuegos por su jugabilidad básica y se ha ido formando a medida que fueron surgiendo numerosos juegos con características en común que permitían agruparlos. De esta manera, los videojuegos con una forma de juego parecida constituyen géneros que se pueden dividir nuevamente en subgéneros cada vez más específicos. Un videojuego no necesariamente debe pertenecer a un único género, muchos títulos presentan elementos combinados de dos o más géneros distintos, en estos casos se considera al género que predomina para categorizarlo.

La clasificación por géneros presenta muchos inconvenientes, principalmente porque cada medio especializado (sitio de Internet, revista, programa) asume un criterio distinto para organizar las categorías debido a que no existe un sistema unificado. Otra dificultad es que muchos géneros surgen por la denominación popular de manera que se crean categorías muy ambiguas o abarcativas que coinciden o chocan con otras (como "juego de naves", "juego de peleas callejeras", "hack'n slash", "juego de guerra", "multijugador masivo") y que son nombres comúnmente mencionados pero pueden agrupar a juegos sumamente distintos. De todas maneras, existe un consenso general para identificar a un juego como parte de un género, debido a que la mayoría de los títulos comerciales se diseñan en base a los géneros más populares existentes. (Despain, 2009)



#### **4.1.3.1 Géneros basados en la acción en tiempo real**

Estos géneros suelen ser agrupados dentro del macrogénero conocido como "Juegos de acción". En ellos, el jugador generalmente debe controlar a un personaje, vehículo u objeto en tiempo real usando el mando para moverlo dentro de un escenario, el objetivo es tradicionalmente llegar hasta una meta, encontrar objetos, ganar puntos, derrotar a los enemigos o una combinación de estos.

##### **4.1.3.1.1 Arcade**

Llamados así porque mantienen los principios de los antiguos juegos de Arcade, basados en el entretenimiento inmediato y una jugabilidad sencilla y adictiva. Su presencia está muy extendida en los sistemas móviles. Estos videojuegos presentan controles muy simples, niveles de corta duración y los objetivos son fáciles de comprender al instante. Se caracterizan por tener una gran cantidad de niveles cortos en donde la dificultad va aumentando rápidamente al progresar. (Despain, 2009)

##### **Principales subgéneros:**

**a) Rompebloques:** Género que deriva del videojuego Breakout. El jugador manipula una paleta que sólo puede moverse a los lados y debe usarla para golpear una pelota que va rompiendo los bloques que forman el nivel. El objetivo es destruir todos los bloques y evitar que la pelota se vaya fuera del área de juego. Ejemplos: Breakout, Arkanoid, Kirby's Block Ball.

**b) Laberinto:** Incluye a los videojuegos con escenarios en forma de laberinto, en donde las paredes marcan los límites del área de juego. Es muy común que estos juegos contengan elementos de persecución en donde el personaje debe escapar de los enemigos y en muchos casos no puede atacarlos. Ejemplos: Pac-Man, Ms. Pac-Man, Mappy, Devil World.

**c) Velocidad:** En esta clase de juegos, el jugador controla a un personaje o vehículo que avanza a gran velocidad sin posibilidad de detenerse. El objetivo es

generalmente recorrer un escenario evitando obstáculos y enemigos, muchos incluyen formas de ataque como disparar proyectiles a los enemigos. Ejemplos: Antarctic Adventure, Kid Klown in Crazy Chase, The 3-D Battles of World Runner, Paperboy.

**d) Camino de obstáculos:** Género muy parecido al de plataformas pero con un control mucho más reducido sobre el personaje. El objetivo es conducir a un objeto a través de un escenario, tratando de no colisionar con los obstáculos y teniendo un límite de tiempo para llegar a la meta. Ejemplos: Marble Madness, Super Monkey Ball, Cameltry, Kuru Kuru Kururin.

#### **4.1.3.1.2 Plataformas**

Un género muy popular en donde la principal misión consiste en conducir a un personaje dentro de un escenario que presenta numerosas dificultades, incluyendo obstáculos, precipicios o enemigos, con la finalidad de alcanzar una meta. La característica que unifica a este género es la presencia de la función de saltar. (Despain, 2009)

##### **Principales subgéneros:**

**a) Comical action:** Estos son juegos que generalmente presentan escenarios de pantalla fija en donde el jugador se mueve dentro un nivel cerrado con varias plataformas y debe derrotar a todos los enemigos para poder avanzar a la siguiente fase. En este género se incluyen a muchos de los antiguos juegos de plataformas de Arcade. Ejemplos: Bubble Bobble, Snow Bros., Tumble Pop.

**b) Plataformas con vista lateral:** Son juegos de plataformas en donde el jugador tiene un punto de vista lateral del escenario y que restringe al personaje a un movimiento bidimensional, avanzando hacia la izquierda o hacia la derecha. Ejemplos: Donkey Kong, Super Mario Bros, Adventure Island, Sonic the Hedgehog.

**c) Plataformas 3D:** Una evolución de los plataformas de vista lateral. Los primeros juegos de este género presentaban gráficos en 2D con una perspectiva isométrica para permitir al jugador mover a su personaje en un entorno tridimensional hacia todas las direcciones. Ejemplos: Fairlight, Sonic 3D Blast, Batman (1986). Con la mejora de la capacidad gráfica de las consolas se pudieron construir verdaderos mundos tridimensionales a partir de gráficos poligonales. Estos permiten una gran libertad de movimientos del personaje, la posibilidad de ajustar el ángulo de la cámara y el recorrido de enormes escenarios en un mundo abierto o siguiendo un camino restringido. Ejemplos: Super Mario 64, Crash Bandicoot, Super Mario, Galaxy, Sonic Adventure.

#### **4.1.3.1.3 Disparos**

Los videojuegos de disparos son aquellos en los que el jugador asume el rol de una persona, vehículo o nave y el objetivo principal es atacar a objetivos diversos disparándoles proyectiles. (Despain, 2009)

#### **Principales subgéneros:**

**a) Shoot 'em up:** En los juegos de este tipo, el jugador controla a un personaje o vehículo de combate que puede disparar ráfagas generalmente ilimitadas de proyectiles a todos los enemigos que salen en pantalla para destruirlos. Suelen usar un plano en dos dimensiones y se los divide en dos grandes grupos según la perspectiva de la cámara: los de vista cenital (19XX (saga), Centipede, Galaxian) y los de vista lateral (Gradius, Thunder Force IV, Forgotten Worlds).

**b) Correr y disparar:** Este clase de juegos combina los elementos de los Shoot 'em up con los juegos de plataformas. El jugador controla a un personaje que puede caminar y saltar a lo largo del escenario mientras dispara frenéticamente a todos los enemigos que van apareciendo. Ejemplos: Contra, Metal Slug, Heavy Barrel.

**c) Rail Shooter:** Estos videojuegos siguen la premisa de los Shoot 'em up, con la gran diferencia de que la pantalla muestra al jugador visto desde atrás, teniendo una vista panorámica del área de juego. Se caracteriza por tener un recorrido preestablecido, el jugador solo puede mover al personaje a lo largo de la pantalla mientras este avanza automáticamente. Ejemplos: Sky Destroyer, Star Fox, Sin and Punishment: Hoshi no Keishosha, Panorama Cotton.

**d) Galería de tiro:** Este género presenta a toda la pantalla como el área a disparar y a los objetos o enemigos que se mueven dentro de ella como los blancos. Muchos videojuegos de este género suelen incluir un dispositivo con forma de arma de fuego para que el jugador pueda apuntar y disparar hacia la pantalla, o de lo contrario, utilizan un cursor que funciona como la mirilla del arma. Ejemplos: Duck Hunt, Virtua Cop, Terminator 2: Judgment Day, Cabal

**e) Disparos en primera persona (First Person Shooter):** Es un género de disparos en 3D en donde la acción se desarrolla desde la perspectiva del personaje protagonista sea humano u otro tipo. El objetivo es recorrer un escenario laberíntico repleto de enemigos, tratando de cumplir un objetivo que por lo general es llegar a una meta o vencer a un jefe. Algunos juegos dentro de este género permiten cambiar la perspectiva a tercera persona viéndose la espalda del protagonista. Ejemplos: Wolfenstein 3D, Doom, Half-Life, Quake

**f) Disparos en tercera persona:** Es un género de disparos en 3D en donde la cámara muestra al jugador de espaldas mientras recorre el escenario y ataca a los enemigos. Se diferencian de los de 1º persona por tener una mayor movilidad del personaje, combinando por lo general elementos de otros géneros como las plataformas, las aventuras o el sigilo. Ejemplos: Tomb Raider, Max Payne, Jet Force Gemini.

#### **4.1.3.1.4 Peleas**

Los videojuegos de pelea o lucha son todos aquellos que incluyen combates directos entre personajes, usando los puños o armas. El objetivo es siempre vencer atacar al rival hasta lograr vencerlo. En este género no se incluyen a aquellos juegos que representan deportes de combate como el boxeo ya que sus principios son distintos. (Despain, 2009)

##### **Principales subgéneros:**

**a) Lucha versus:** Juego de lucha más tradicional que presenta combates cerrados, casi siempre entre dos personajes, que aparecen enfrentados cara a cara en tiempo real. El objetivo es golpear al rival múltiples veces para ir agotando su barra de energía, cuando esta se acaba el personaje cae derrotado. Se caracterizan por tener una interacción nula o escasa con el escenario. Ejemplos: Street Fighter II, Tekken, Samurai Showdown, Mortal Kombat.

**b) Lucha Free-for-All:** Juego de lucha que presenta combates entre dos o más personajes y que se desarrollan en escenarios más elaborados, en donde los competidores pueden moverse libremente, y usar los elementos y objetos presentes como parte de su estrategia. Ejemplos: Power Stone 2, Super Smash Bros.

**c) Beat 'em up:** Estos juegos consisten en controlar a un personaje y comenzar a avanzar a lo largo de un escenario derrotando a todos los enemigos que van apareciendo. Los escenarios se caracterizan por ser muy planos y lineales, presentando pocos o ningún elemento de plataformas. Muchos Beat 'em ups permiten a dos o más jugadores jugar en forma cooperativa para hacer frente a los enemigos. Ejemplos: Double Dragon, Spartan X, Final Fight, Streets of Rage.

**d) Combate por turnos:** Un estilo de lucha poco convencional similar a los combates de RPG en donde los competidores deben escoger un único movimiento por cada turno (atacar, esquivar, usar un especial). Además influyen los factores

estadísticos como la fuerza, la velocidad o la defensa y también el azar. Ejemplos: Yuu Yuu Hakusho (SNES), Pokemon Stadium, Kishin Douji Zenki: Denei Raibu.

#### **4.1.3.2 Géneros basados en el desarrollo argumental y la exploración**

Los llamados juegos de aventura son un macrogénero que agrupa a los videojuegos que están basados primordialmente en el relato de una historia y que se desarrollan mediante la exploración de un mundo abierto al jugador, que le permite la toma de decisiones y que lo enfrenta a desafíos que debe resolver principalmente utilizando el ingenio y el conocimiento de las reglas del juego. (Despain, 2009)

##### **4.1.3.2.1 Aventura conversacional**

Son los más antiguos juegos de aventura que fueron muy populares en los primeros ordenadores personales, debido a las limitaciones de la tecnología, estos juegos estaban contruidos únicamente por textos. Aunque eventualmente se agregaron imágenes, estas no influyen en la forma de juego y son meramente ilustrativas. La situación se le presenta al jugador mediante relatos, como si estuviera leyendo una historia, y tiene la posibilidad de interactuar con el juego y avanzar en la trama insertando comandos con el teclado. Ejemplos: Colossal Cave Adventure, The Hitchhiker's Guide to the Galaxy, Zork. (Despain, 2009)

#### **Principales subgéneros:**

**a) Novela visual:** Las novelas visuales son un tipo de videojuegos para ordenadores muy prolíficos en Japón que se desarrollan completamente mediante textos que relatan la historia y los diálogos de los personajes y que van acompañados por imágenes con una estética muy característica del animé para adolescentes. Ofrecen por lo general un grado muy bajo de interacción, el jugador se limita sólo a escoger opciones que pueden afectar el desarrollo de la historia y conseguirle eventos especiales y finales alternativos. Ejemplos: Tomoyo After: It's a Wonderful Life, Doukyuusei, Neon Genesis Evangelion: Iron Maiden.

**b) Aventura gráfica:** Este género marca una evolución de los juegos de aventura, al introducir gráficos que acompañaban a los textos para lograr una interfaz más amistosa para el jugador. A medida que las aventuras gráficas fueron evolucionando, se le fueron agregando más elementos de acción como la posibilidad de caminar y mover al personaje dentro de un mundo en tiempo real, en lugar de solamente escoger opciones para hacerlo avanzar.

#### **Principales subgéneros de la Aventura Gráfica:**

**Point-and-click:** Un estilo de aventura gráfica diseñado en base al uso del mouse para jugar. Los jugadores aquí controlan un cursor con el que pueden escoger una variedad de opciones para que el personaje las ejecute (Abrir, Hablar, Usar, Tirar) y luego escoger sobre que objeto del escenario ejecutará la opción. Ejemplos: Maniac Mansion, Clock Tower, The Secret of Monkey Island.

**Aventura en primera persona:** En este tipo de aventura gráfica el jugador tiene una perspectiva en primera persona del mundo a explorar, se caracterizan generalmente por una forma de juego muy abierta y libre en donde no hay enemigos o peligros inmediatos. Ejemplos: Myst, Atlantis: The Lost Tales, Metroid Prime.

**Video interactivo:** Este es un género de videojuegos que comenzó a surgir con la tecnología de grabación de datos en laserdics y CDs y que permitía por primera vez mostrar videos extensos en tiempo real. Este tipo de juegos son siempre de jugabilidad muy limitada, el jugador ve una película y cada tanto se le presentan opciones o en algunos casos debe presionar botones en el momento indicado, dependiendo de lo que escogió se mostrará una nueva escena en donde sigue jugando o pierde. Ejemplos: Dragon's Lair, Brain Dead 13, Space Ace.

#### **4.1.3.2.2 Juegos de rol**

Popularmente conocidos como RPG, este género está basado en los juegos de rol de mesa. Por lo general, el jugador inicia una partida en la que escoge un equipo de personajes, cada uno con sus propias características y habilidades, expresadas en forma de datos estadísticos (salud, magia, ataque, defensa, sabiduría). El juego

en general se divide en explorar pueblos, en donde el equipo puede abastecerse y comprar distintos objetos y en explorar mazmorras o calabozos, que son escenarios laberínticos repletos de enemigos en donde los personajes combaten para ganar experiencia y dinero y así se van fortaleciendo. (Despain, 2009)

**Principales subgéneros:**

**a) RPG occidental:** Este es el subgénero más clásico de juegos de rol. Se caracterizan por una forma de juego no lineal basada en la exploración e interacción con personajes y el uso de numerosas estadísticas para determinar las capacidades de cada personaje. En estos juegos el jugador puede crear su grupo de personajes a gusto y el objetivo se basa en cumplir distintas misiones que incluyen ingresar en mazmorras laberínticas repletas de monstruos. Es muy común que los más antiguos usen una perspectiva en primera persona. Ejemplos: Wizardry: Proving Grounds of the Mad Overlord, Digital Devil Story: Megami Tensei, Advanced Dungeons & Dragons: Eye of the Beholder.

**b) RPG japonés:** Este tipo de RPGs se caracterizan por el predominio de una historia muy elaborada que define el rumbo a seguir. El transcurso del juego normalmente se centra en un protagonista que comienza siendo muy débil, pero que se va fortaleciendo y va sumando compañeros a lo largo de la aventura. Los personajes a controlar están predefinidos y cada uno juega un papel en la historia, apareciendo en distintos tramos del juego. Tradicionalmente utilizan un sistema de combate por turnos. Ejemplos: Dragon Quest, Final Fantasy, Chrono Trigger, Earthbound.

**c) Roguelike:** Los videojuegos Roguelike son juegos tradicionalmente de ordenador, ambientados en mazmorras bidimensionales, en su mayoría con texto simple o "gráficos" ASCII, aunque el suceso de este género llevo a la creación de juegos de consola con gráficos más convencionales. Los juegos roguelike se caracterizan por tener niveles de mazmorra generados de forma aleatoria, lo que les da una mayor variedad que los videojuegos en que siempre aparecen los



mismos niveles. Muchos de ellos tienen algunos niveles estáticos, que suelen ser niveles especiales o un nivel único (como puede serlo el nivel situado por encima de las mazmorras). Ejemplos: Rogue, NetHack, Pokémon Mundo Misterioso: Equipo de Rescate Azul, DoomRL.

**d) RPG táctico:** Muy similares en diseño a los RPGs japoneses, su principal característica es su sistema de batalla, que combina elementos típicos de los RPGs con la estrategia. El jugador controla a un equipo muy numeroso de personajes, cada uno con habilidades y atributos particulares y debe moverlos por turnos en un campo de batalla repleto de enemigos, usando las tácticas y formaciones que considere favorables. Ejemplos: Shining Force, Fire Emblem, Super Robot Taisen, Front Mission.

**e) RPG de acción:** Esta clase de videojuegos presentan todos los elementos típicos de los RPG, con la gran diferencia de que cambian el sistema de combates por turnos por un combate en tiempo real en donde el jugador tiene control total sobre su personaje al luchar. Estos juegos suelen tener a un protagonista solitario aunque algunos incluyen compañeros controlados por el CPU o por un segundo jugador. Es muy común la presencia de elementos de videojuegos de plataformas y de ingenio. Ejemplos: Terranigma, Secret of Mana, Illusion of Gaia.

#### **4.1.3.2.3 Aventura de acción**

Este género incluye a los videojuegos que combinan elementos propios de los juegos de aventura (exploración, resolver intrigas, interacción con el escenario y personajes) con elementos característicos de los videojuegos de acción (combate en tiempo real, corridas, saltos y destrezas físicas). Ejemplos: The Legend of Zelda, Devil May Cry, Star Fox Adventures. (Despain, 2009)

**Principales subgéneros:**

**a) Plataformas de exploración:** Este género presenta una combinación de elementos de juegos de plataformas con los de los juegos de aventura de acción. Se caracterizan el uso de escenarios más grandes, que se han de recorrer varias veces y en varias direcciones, una evolución continúa de las habilidades del personaje y generalmente poseen un argumento y ambientación mucho más desarrolladas. Ejemplos: Metroid, Another World, Prince of Persia.

**b) Sigilo:** Este subgénero incluye a los videojuegos en los que predominan el sigilo y el avance táctico por sobre la confrontación directa. El jugador controla a un personaje que se ubica en un escenario con varios enemigos y debe cumplir una determinada misión tratando de no llamar la atención, de ser necesario debe eliminar al enemigo tomándolo por sorpresa y sin alertar a los demás. Son muy comunes los juegos de sigilo que combinan elementos de los videojuegos de disparos en tercera persona. Ejemplos: Metal Gear Solid, Hitman, Tenchu: Stealth Assassins, Splinter Cell.

**c) Survival horror:** Esta es una clase de videojuegos de aventura con una temática muy influenciada por el cine de terror. El jugador controla a un personaje que se ve inmerso en una situación dramática en la que es acechado por enemigos sobrenaturales y otros peligros. Este tipo de juegos se centra en la exploración y la resolución de problemas mediante objetos y pistas, el objetivo por lo general es escapar a salvo del escenario y rescatar a otros personajes. Ejemplos: Alone in the Dark, Resident Evil, Silent Hill.

**d) Conducción por misiones:** Este subgénero es un híbrido entre los videojuegos de carreras y los de aventura de acción. En ellos hay que conducir vehículos a lo largo de escenarios, pero en lugar de simplemente competir o llegar a la meta, hay que cumplir con misiones específicas que suelen involucrar situaciones de espionaje, persecuciones y conducir por zonas de alto riesgo. Ejemplos: Knight Rider: The Game, Driver, Midtown Madness.

### **4.1.3.3 Géneros basados en el ingenio y la coordinación**

Hay varios géneros de videojuegos que requieren de la destreza mental del jugador para cumplir con los objetivos requeridos y así poder ganar. Estos géneros explotan capacidades como los reflejos, la coordinación, la memoria y principalmente la capacidad de resolver problemas. (Despain, 2009)

#### **4.1.3.3.1 Puzzle**

Conocidos también como videojuegos de lógica o de inteligencia. Este género incluye a los juegos en donde cada nivel se presenta como una situación problemática al jugador, que este debe resolver siguiendo las reglas impuestas mediante el uso de su razonamiento lógico. Estos juegos suelen tener un límite de tiempo que impide al jugador tardarse demasiado y también chances ilimitadas, puesto que el jugador suele perder una gran cantidad de veces antes de llegar a la solución. Ejemplos: Buscaminas, Picross, World of Goo, Adventures of Lolo, Shove It! The Warehouse Game.

#### **4.1.3.3.2 Puzzle de acción**

Este tipo de puzzles son videojuegos derivados del Tetris, en donde hay que acomodar numerosos bloques para formar una secuencia determinada y así hacerlos desaparecer, el jugador tiene que hacerlo rápidamente para evitar que la pantalla se llene de bloques. A partir del lanzamiento de Puyo Puyo se hicieron muy populares los puzzles del tipo competitivo. Ejemplos: Tetris, Puyo Puyo, Klax, Panel de Pon, Columns

#### **4.1.3.3.3 Didáctico**

Se incluye en el género didáctico o educativo a los videojuegos diseñados para fomentar el aprendizaje. Los más comunes son los videojuegos dedicados para niños en edad preescolar y escolar que incluyen desafíos matemáticos o gramaticales, lecciones de historia y geografía, para enseñar idiomas, ejercitar la mente o simular labores reales. Ejemplos: Mario is Missing, Captain Novolin, Math Blaster: Episode 1, EMIT.

#### **4.1.3.3.4 Preguntas**

Esta clase de videojuegos está inspirada en los programas televisivos de concursos sobre preguntas y respuestas. Algunos juegos de preguntas son simples simuladores de estos concursos televisivos y el objetivo es responder preguntas antes que los rivales para sumar puntos y salir vencedor al terminar el programa. Otros presentan un argumento más creativo en donde el jugador se sumerge en una historia en donde debe responder preguntas correctamente para poder seguir avanzando y llegar al final. Las preguntas por lo general contienen trivia sobre cultura general como historia, geografía, cine y televisión, deportes, celebridades, etcétera. Ejemplos: Jeopardy!, Capcom World 2, Quiz & Dragons, Quiz Nanairo Dreams: Nijirochi no Kiseki

#### **4.1.3.3.5 Musical**

Los videojuegos musicales se basan en la interacción del jugador con una pieza musical determinada. El objetivo comúnmente es seguir el ritmo de la música en tiempo real, representada en la pantalla por unos indicadores que muestran la secuencia de comandos que el jugador debe seguir con rigurosa precisión tratando de no cometer errores. Ejemplos: Dance Revolution, Rock Band, Guitar Hero, Donkey Konga.

#### **4.1.3.3.6 Ejercicios**

Género que incluye a los videojuegos especialmente diseñados para que el usuario haga ejercicio físico como forma de juego. En el pasado, como los controles tradicionales no podían detectar el movimiento estos juegos eran lanzados junto con periféricos especiales, como alfombras y plataformas para detectar el trabajo de las piernas. Con el auge de las consolas con controles sensores de movimiento se hizo más sencillo producir videojuegos que pudieran reconocer el ejercicio del jugador. Ejemplos: Power Pad, Exertris, Wii Fit, Zumba Fitness.

#### **4.1.3.4 Géneros basados en el pensamiento estratégico y la administración**

Los géneros de juegos de video denominados estratégicos y simuladores presentan una forma de juego en donde al jugador se le presenta una situación concreta en una realidad determinada, en donde su objetivo es administrar los recursos de que dispone para lograr un desarrollo progresivo de sus bienes. (Despain, 2009)

##### **4.1.3.4.1 Estrategia**

Los videojuegos de estrategia son aquellos en donde la forma de juego requiere el uso de un pensamiento táctico y la planificación de acciones para alcanzar la victoria. Estos juegos suelen caracterizarse por dar una gran libertad al jugador para diseñar su camino a seguir. (Despain, 2009)

##### **Principales subgéneros:**

**a) Artillería:** Juegos de combate en los que dos o más bandos de varias unidades se enfrentan disparando cañonazos y otros tipos de proyectiles. Se desarrollan por turnos en donde el jugador tiene un determinado tiempo para escoger a un blanco y tratar de apuntarle de forma correcta, se toma en consideración factores como el ángulo de disparo, la distancia y el viento antes de ejecutar un ataque. Ejemplos: Artillery, Worms, Scorched Earth.

**b) Estrategia por turnos:** Estos juegos presentan a dos o más bandos competidores, representando generalmente a naciones en conflicto. Algunos títulos presentan únicamente batallas en el campo de guerra en donde hay que vencer al enemigo. Mientras que otros son mucho más complejos y permiten llevar a cabo varios aspectos del gobierno, incluyendo el conflicto militar, las negociaciones y alianzas con otros bandos y la administración económica. Tienen como objetivo la expansión del territorio. Ejemplos: Romance of the Three Kingdoms, Advance Wars, The Battle for Wesnoth.

**c) Estrategia en tiempo real:** También llamados RTS, son juegos en donde la misión es controlar a un ejército de personajes en tiempo real y sin pausa para la toma de decisiones. Las misiones generalmente consisten en avanzar y expandir al ejército a lo largo de un territorio, tomando las bases enemigas y a la vez construyendo bases a modo de incrementar los recursos disponibles. El jugador debe estar atento a distintos hechos que ocurren al mismo tiempo y dar órdenes rápidas para superar las dificultades. Ejemplos: Dune, Command & Conquer, Warcraft, StarCraft.

**d) Construcción de imperios:** También conocidos como 4X, son juegos en los que el jugador toma el mando de un imperio, dando prioridad a los aspectos como la economía, diplomacia e investigación. Pueden tener toques de táctica militar aunque no a un nivel tan extendido como otros subgéneros de la estrategia. Pueden ser por turnos o en tiempo real. Ejemplos: Hearts of Iron, Empire Earth, Civilization.

#### **4.1.3.4.2 Simulación**

Este género agrupa a los juegos que simulan una realidad determinada que puede ser semejante a la vida real o presentar un mundo ficticio en donde el jugador interactúa con el entorno haciendo uso de recursos y bienes.

#### **Principales subgéneros:**

**a) Construcción y administración:** Juegos basados en la construcción de una propiedad y la administración económica de la misma. El desarrollo del juego se centra en el aspecto de la administración de los recursos y la construcción edilicia para mejorar o expandir la propiedad, teniendo como fin ganar clientes y dinero para continuar con el crecimiento. También hay juegos que permiten administrar una ciudad, controlando aspectos como la seguridad, la salud y la industria a medida que va creciendo el número de habitantes. Ejemplos: SimCity, Theme Park, Caesar.

**b) Simulador de negocios:** Género de estrategia que se centra en el manejo del dinero, dejando de lado la construcción. Este tipo de videojuegos simula el mundo de los negocios. El videojugador cuenta con un capital determinado y debe tratar de generar ganancias invirtiendo su capital sabiamente y negociando con los personajes del juego. Ejemplos: Aerobiz Supersonic, Railroad Tycoon, Capitalism.

**c) Simulador de oficio:** Este tipo de simuladores se asemeja a los juegos de negocios, pero tienen como objetivo el desempeño exitoso de la actividad por sobre la ganancia de dinero. La forma de juego se focaliza principalmente en el uso correcto de herramientas frente a distintas situaciones. Entre los simuladores de oficio se pueden encontrar videojuegos en donde el usuario asume el rol de granjero, médico o cocinero, entre otros. Ejemplos: Harvest Moon, Cooking Mama, Trauma Center.

**d) Pesca:** Un género que incluye a simuladores de la actividad de pesca deportiva. Se asemeja mucho a los simuladores de oficio aunque el aspecto administrativo es escaso y se concentra en la técnica del jugador y el éxito competitivo representado por la calidad los ejemplares capturados. Los juegos de pesca tratan siempre de ofrecer una experiencia realista permitiendo al jugador una gran libertad para escoger su equipo y navegar por los ríos, simulan entre otras cosas las condiciones climáticas y el comportamiento de los peces. Ejemplos: The Black Bass, River King, The Blue Marlin

**e) Manager deportivo:** Los videojuegos de manager deportivo colocan al jugador al mando administrativo de un club o equipo deportivo, en estos juegos el jugador debe buscar el éxito deportivo de su equipo además de lograr ganancias y debe tomar decisiones en aspectos como la táctica deportiva, transferencia de jugadores y manejo de la economía. También existen juegos que se basan en el manejo de la carrera de un único deportista, teniendo que armar su calendario de entrenamiento y competencias. Ejemplos: Football Manager, Premier Manager, FIFA Manager.

**f) Apuestas:** También conocidos como juegos de azar. Los videojuegos de apuestas son aquellos en donde el jugador comienza con una suma de dinero reducida y debe lograr enriquecerse participando en el mundo de las apuestas y los juegos de azar. Dentro del género de apuestas, los más comunes son los simuladores de casino, en donde hay la posibilidad de apostar y también de jugar en los distintos desafíos. En Japón también son muy populares los juegos de apuestas de carreras de caballos y deportes varios, en donde raramente hay participación del jugador dentro de la carrera en cuestión y solo se centran en la predicción. Ejemplos: Caesar's Palace, Casino Kid, Derby Stallion.

**g) Simulador de vehículo:** Este género agrupa a los distintos videojuegos que introducen al jugador en el control de un vehículo de la manera más realista posible, incluyendo aspectos como el manejo, el mantenimiento, las limitaciones de la física, la maniobrabilidad. Los más populares son los simuladores de vuelo, que permiten tener la experiencia real de controlar un avión de pasajeros o una nave de guerra, también son comunes los simuladores de trenes, vehículos militares y navales. Algunos simuladores además se basan en vehículos futuristas como naves espaciales y mechas. Los videojuegos que representan vehículos de carreras suelen incluirse en un género distinto (Carreras) porque en ellos prima la competitividad y la acción rápida por sobre la experiencia de manejo y el realismo. Ejemplos: Microsoft Flight Simulator, Train Simulator, Star Trek: The Next Generation: Advanced Holodeck Tutorial.

**h) Simulador de vida:** Este tipo de videojuegos simulan el cuidado y mantenimiento de seres vivos virtuales, estos pueden ser animales reales o fantásticos y también personas. El objetivo es garantizar el correcto desarrollo del ser que se está cuidando. Por ejemplo, los simuladores de mascotas requieren cuidar aspectos como la alimentación, el ejercicio y la felicidad. Los simuladores de personas son también agrupados en la categoría de "simuladores sociales", ya que incluyen aspectos de la vida cotidiana como el trabajo, los pasatiempos y las relaciones interpersonales. También existen los denominados "simuladores de



dios", en donde el jugador puede crear y controlar el desarrollo de seres vivos y su hábitat en el rol de un ser omnipotente. Ejemplos: The Sims, Spore, Tamagotchi, Petz.

#### **4.1.3.5 Géneros adaptados de juegos, deportes y disciplinas reales**

##### **4.1.3.5.1 Deportes**

Los videojuegos deportivos están basados o son una simulación virtual de las distintas disciplinas deportivas reales. (Despain, 2009)

##### **Principales subgéneros:**

**a) Simulador de deportes:** Este subgénero incluye a todos los videojuegos que están basados en un deporte real. Los simuladores de deportes además se agrupan en subcategorías de modo que cada una representa a un deporte específico (fútbol, tenis, baloncesto, juegos olímpicos, etcétera.). Aunque la mayoría de los juegos representan a un único deporte, algunos se dedican a representar a varios deportes, aunque de manera más sencilla. Se puede destacar además que algunos juegos deportivos se especializan en representar al deporte de la manera más realista posible, mientras que otros agregan elementos fantásticos y personajes sobrehumanos para aumentar la acción y la velocidad del juego. Ejemplos: Videojuegos de fútbol, International Superstar Soccer Deluxe, Pro Evolution Soccer, FIFA, de Béisbol, R.B.I. Baseball, Jikkyou Powerful Pro Yakyuu, Tenis Virtua Tennis, Mario Tennis, Golf Everybody's Golf, Tiger Woods PGA Tour, Baloncesto NBA Jam, NBA Live, Billar, Lunar Ball, Virtual Pool 64.

**b) Deportes de combate:** Este subgénero de los juegos deportivos agrupa a los videojuegos que representan competencias deportivas de combate y/o artes marciales reales, se diferencian de los juegos de lucha porque los personajes están limitados por las leyes de la física, pudiendo utilizar solamente movimientos humanamente posibles y por el reglamento del deporte en cuestión. Los distintos

deportes de combate que fueron adaptadas en videojuegos son: Boxeo (Punch-Out!!, Fight Night); Lucha libre (Saturday Night Slam Masters, Fire Pro Wrestling, WWF Super WrestleMania); Artes marciales mixtas (UFC 2009 Undisputed, EA Sports MMA); Judo (Brian Jacks Uchi Mata, Moero 7!! Juudou Warriors); Sumo (64 Oozumou, Tsuppari Oozumou), entre otros.

**c) Deportes ficticios:** Algunos videojuegos representan deportes totalmente inventados, inexistentes en la realidad, aunque siempre tienen como base a elementos o reglas de los deportes reales. Los juegos de deportes ficticios por lo general transcurren en mundos o situaciones fantásticas que les permite quebrar las reglas de la física que limitan a los deportes reales. Existen numerosas clases de deportes ficticios: Los deportes cómicos, que son básicamente una parodia caricaturesca de los deportes reales; los deportes mágicos, en donde elementos de la fantasía y lo sobrenatural forman parte del juego; los deportes virtuales, en donde los jugadores se "meten" en la computadora y compiten en un mundo de realidad virtual; y los deportes futuristas, en donde se utiliza la tecnología y elementos mecánicos para mejorar al deporte y hacerlo más violento y emocionante. Ejemplos: Speedball, Kirby's Dream Course, Harry Potter: Quidditch World Cup.

#### **d) Carreras**

Los videojuegos de carreras son aquellos en donde el jugador controla a un personaje o vehículo que compite en una carrera contra otros vehículos a lo largo de una pista. (Despain, 2009)

### **Principales subgéneros:**

**Carreras estilo Arcade:** Este es el género de carreras más antiguo en donde el objetivo principal es mantenerse a salvo y evitar las colisiones. La mayoría de estos juegos incluyen un tiempo limitado en el que el jugador debe tratar de alcanzar la meta o punto intermedio, si no lo logra pierde automáticamente. Como variante hay juegos que utilizan combustible en lugar de tiempo y el jugador debe recoger contenedores durante la carrera para evitar que se agote. Ejemplos: Pole Position, OutRun, Super Hang-On.

**Carreras deportivas:** Este género es el que mejor representa a las competiciones reales de vehículos. El jugador debe competir en una pista contra varios corredores en igualdad de condiciones y tratar de llegar primero a la meta. Al igual que en los campeonatos de carreras, el objetivo es sumar puntos a lo largo de varios circuitos para lograr tener la mejor calificación al terminar el torneo y coronarse campeón. Existen numerosas subdivisiones de este género según el tipo de vehículos que compiten (Formula-1, rally, motos, jet ski). Muchos de estos videojuegos representan competiciones, pilotos, vehículos y pistas reales. También existen aquellos que presentan vehículos ficticios, como las carreras futuristas, en donde los vehículos son máquinas fantásticas que viajan a enormes velocidades en pistas con muchos peligros. Ejemplos: F1 Challenge, Top Gear, Need for Speed, F-Zero.

**Carreras de combate:** Los videojuegos de carreras de combate también se basan en la competición de varios vehículos por llegar primero a la meta, aunque la mayor diferencia está en el agregado de armas o ítems que los competidores pueden usar para obtener ventaja. Los controles suelen ser más simples que en los demás juegos de carreras de modo que los jugadores que reciben daño o chocan se puedan recuperar rápidamente. Muchos de estos videojuegos, principalmente los de carreras de karts, incluyen pistas fantásticas, llenas de trampas y obstáculos que aumentan considerablemente el grado de acción. Ejemplos: Road Rash, Super Mario Kart, Rock'n Roll Racing.

#### **4.1.3.5.2 Juegos de mesa**

Los juegos de mesa o de tablero están directamente basados en los juegos de mesa reales. (Despain, 2009)

##### **Principales subgéneros:**

**a) Juego de tablero:** Los juegos de tablero son el tipo más tradicional de juego de mesa. Este tipo de juegos se caracterizan por jugarse sobre un tablero especial que presenta los dibujos de los casilleros en donde se colocan las fichas para jugar. Cada juego tiene sus propias reglas, está siempre diseñado para dos o más jugadores y se desarrolla por turnos en donde cada jugador debe realizar un movimiento. Los más tradicionales suelen basarse puramente en la estrategia (ajedrez, shogi, reversi), mientras que los más modernos suelen agregar elementos de azar como lanzar dados o sacar cartas para obtener algún beneficio o prenda. Ejemplos: Battle Chess, Mahjong, Monopoly.

**b) Juego de la oca:** Este es un tipo especial de juego de tablero que se basa en el tradicional juego de la oca, pero con el agregado de muchos elementos fantásticos propios de los videojuegos que no se pueden representar en la realidad. Están armados casi siempre para cuatro o más jugadores. El juego se lleva a cabo en un complejo tablero que representa un camino, dividido en numerosos casilleros. Todos los jugadores parten desde un mismo punto y deben lanzar uno o dos dados para determinar cuántos casilleros pueden avanzar, es común que por caer en algún casillero determinado el jugador reciba un premio o un castigo, el objetivo es siempre tratar de llegar a la meta, que es el casillero final. Es también muy común el agregado de minijuegos que se activan al caer en determinados casilleros. Ejemplos: Kiteretsu Daihyakka: Choujikuu Sugoroku, Momotaro Dentetsu, Mario Party.

**c) Juego de naipes:** Este tipo de juegos son casi siempre adaptaciones de juegos de naipes o cartas reales. Los juegos de naipes se caracterizan por jugarse siempre con una baraja con una cantidad invariable de naipes y que son mezcladas antes de jugar de modo que las cartas que reciba cada jugador dependan puramente del azar. La mayoría de los videojuegos están basados en los juegos de la baraja inglesa, debido a que esta es la más extendida a nivel mundial por los casinos. Los juegos de naipes competitivos están muy ligados con el mundo de las apuestas y los casinos y resulta muy común que estén integrados con los videojuegos de apuestas. Ejemplos: World Series of Poker Pro Challenge, Johnny Midnight BlackJack, Solitario, Carta Blanca.

**d) Juego de cartas coleccionables:** Estos son un tipo especial de juego de cartas, con elementos de los juegos de rol, en donde los jugadores usan las cartas para representar una batalla entre personajes. Se caracteriza porque cada carta es única y tiene sus propias características, función y efecto dentro del juego, que lleva escrita sobre la misma. Cada juego de cartas coleccionables utiliza su propia baraja, en donde el jugador puede confeccionar y escoger su propio mazo para utilizar en cada partida. Ejemplos: Pokemon Trading Card Game, Yu-Gi-Oh, Duel Masters.

#### **4.1.3.5.3 Juegos mecánicos**

**a) Pinball:** Estos videojuegos están basados en las máquinas de Pinball que se solían encontrar en los salones recreativos. Algunos pinball tratan de ser lo más realista posible, simulando la física y los elementos reales de la máquina. Por el contrario, otros videojuegos dejan de lado el realismo para crear pinballs fantásticos agregando elementos como enemigos, poderes, portales y minijuegos. En los juegos de Pinball el objetivo es siempre sumar la mayor cantidad de puntos posibles. Ejemplos: Pinball, Sonic Spinball, Dragon's Revenge.

**b) Pachinko:** Los videojuegos de este género representan las máquinas de juego japonesas llamadas Pachinko. Las máquinas son similares a un Pinball, con la diferencia de que no hay paletas, la máquina se encarga de lanzar desde lo alto una gran cantidad de bolitas que van cayendo y golpeando los distintos topes a su paso, el jugador sólo puede ajustar la potencia del disparo y debe tratar de que las bolitas caigan en las zonas afortunadas para ganar más bolitas que luego se canjean por premios. Ejemplos: Pachinko Sexy Reaction, Super Pachinko Taisen, Pachinko Kuunyan.

#### **4.1.3.6 Videojuegos que incluyen varios géneros distintos**

**4.1.3.6.1 Minijuegos:** Hay ciertos videojuegos que están conformados por un conjunto de varios juegos distintos conocidos como minijuegos y que suelen ser más sencillos que un juego normal. Los minijuegos pueden pertenecer a cualquier género, pero por lo general son de los géneros de acción más sencillos y rápidos de asimilar. Ejemplos: Action 52, WarioWare Inc: Mega Microgame, Three Wonders, Wii Sports.

**4.1.3.6.2 Recopilación:** Se llama recopilación a la publicación de dos o más juegos que originalmente se comercializaron por separado dentro de un mismo título. Los juegos de recopilación normalmente incluyen videojuegos de generaciones anteriores que por resultar anticuados, se incluyen varios en un mismo paquete para resultar comercialmente más atractivos. También se suelen comercializar packs de juegos de la misma generación que antes se vendían por separado, a modo de oferta. Estos son conocidos como "2 en 1" o "3 en 1" según la cantidad.

#### **4.1.3.7 Géneros de no videojuegos**

Estos géneros incluyen a determinados tipos de software interactivo lanzados para consolas de videojuegos sin ser juegos realmente. (Despain, 2009)

**4.1.3.7.1 Creación artística:** Género que incluye a las aplicaciones que permiten al jugador crear algún tipo de arte, como ser dibujar, colorear, crear animaciones, componer música. Ejemplos: Mario Paint, Sid Meier's C.P.U. Bach, Tiny Toon Adventures: Cartoon Workshop.

**4.1.3.7.2 Horóscopo:** Algunos juegos están diseñados únicamente para presentar el horóscopo del personaje basándose en la astrología, incluyendo una presentación y menús muy parecidos a los de un videojuego. Ejemplos: '89 Dennou Kyuusei Uranai, Taboo: The Sixth Sense, Ai Sensei no Oshiete: Watashi no Hoshi.

**4.1.3.7.3 Video:** En este género se incluyen a títulos publicados especialmente para consolas de videojuegos, conteniendo solamente videos para reproducir, sin ningún tipo de interacción del usuario. Ejemplos: Pokémon: Johto Photo Finish, Game Boy Advance Video, Sonic X: Game Boy Advance Video Volume 1, Valis Visual Collection.

**4.1.3.7.4 Aplicación:** Utilidades especialmente creadas para la recreación, la administración de videojuegos, de videoconsolas o algún otro tipo de actividad. Ejemplos: 3DO Game Guru, Pokémon Box: Ruby and Sapphire, Mario Calculator.

#### **4.1.4 Sistema de clasificación de contenido de videojuegos**

Un sistema de clasificación de contenido de videojuegos es un sistema usado para la clasificación de videojuegos en grupos idóneos relacionados. La mayoría de estos sistemas están asociados con y/o patrocinados por un gobierno y a veces forman parte del sistema de clasificación de películas local. La utilidad de estas clasificaciones ha sido puesta en duda por estudios que han llegado a conclusiones tales como que el 90% de los adolescentes declaran que sus padres "nunca" verifican las clasificaciones antes de permitirles alquilar o comprar

videojuegos, por esto tales reclamos se han realizado para "arreglar" los sistemas de clasificación existentes.

Siendo el sistema de clasificación más importante y más ampliamente utilizado el ESRB

#### **4.1.4.1 Entertainment Software Rating Board ESRB**

Es un sistema norteamericano para clasificar el contenido de los videojuegos y asignarle una categoría dependiendo de su contenido.

Fue establecido en 1994 por la Entertainment Software Association (ESA), la anteriormente llamada Interactive Digital Software Association (IDSA). En 2009, había asignado 19.130 categorías a juegos enviados por más de 350 distribuidoras.

ESRB realiza de forma independiente clasificaciones, entregando lineamientos y los principios de privacidad para la industria de los videojuegos. Primero clasifica los videojuegos según su contenido de violencia física o verbal y otros elementos como el contenido sexual. Esta clasificación orienta y ayuda a los padres y consumidores a elegir los videojuegos que son correctos para su familia, idea propiciada tras la aparición del video juego Mortal Kombat. (Entertainment Software Rating Board, About ESRB, 2013)

##### **4.1.4.1.1 Clasificación ESRB**

Los símbolos que usa la ESRB son letras alfabéticas estilizadas que tienen la intención de indicar la etapa para la que es adecuado el juego. La ESRB usa actualmente 7 clasificaciones diferentes. (Entertainment Software Rating Board, ESRB Ratings Guide, 2013)



Tabla 1. Clasificaciones normales de ESRB

Clasificación ESRB	Nombre	Año de Entrada	Edad Adecuada	Descripción de las clasificaciones
EC	Early Childhood	1994	De 3 a 5 años	Material con contenido apto para niños. Son específicamente desarrollados para niños pequeños y usualmente son de orientación educacional.
E	Everyone	1998	Todas las edades o 6 años en adelante	Los títulos comprendidos dentro de esta categoría, pueden contener algo de animación, violencia fantástica o mínima o el uso de insultos suaves.
E10+	Everyone 10 and up (Todas las personas mayores de 10 años)	2005	10 años en adelante	Las obras dentro de esta categoría contienen animaciones, más violencia leve o fantástica, insultos regulares o sangre en temas sugerentes.
T	Teen (Adolescentes)	1994	13 años, en adelante	Los productos de este género contienen violencia moderada, temas sugerentes, humor crudo, sangre, juegos de azar simulados, o uso ligero de lenguaje fuerte.

**Tabla 2. Clasificaciones Restringidas de ESRB**

Clasificación ESRB	Nombre	Año de Entrada	Edad Adecuada	Descripción de las clasificaciones
M	Mature	1994	17 (Menores deben ir acompañados de un adulto)	Las obras de esta categoría muestran violencia, sangre y horror, temas sexuales o insultos. Muchos vendedores aplican políticas sobre no vender juegos con este ordenamiento a menores de edad sin la presencia y aprobación de algún tutor. Desde el año 2003, el símbolo lleva la leyenda 17+ al lado de la palabra Mature, a pesar de que estos juegos están destinados a personas mayores de 17 años. Algunos ejemplos de juegos de esta categoría son: Conker's Bad Fur Day y Mortal Kombat
AO	Adults Only (18+)(Adultos únicamente)	1994	18 (No se admiten menores)	Los títulos en esta categoría incluyen escenas prolongadas de violencia extrema o temas sexuales y desnudez. La categorización AO es tema de controversia por las exageradas restricciones que impone a algunos juegos. Por ejemplo, Thrill Kill, Manhunt 2. También es usado comúnmente en juegos que presentan escenas de hentai. Empresas como Microsoft, Sony o Nintendo no permiten que juegos calificados con este símbolo salgan para sus consolas.

Tabla 3. Otras clasificaciones sin restricciones de ESRB

Clasificación ESRB	Nombre	Año de Entrada	Edad Adecuada	Descripción de las clasificaciones
RP	Rating Pending (Clasificación pendiente)	1994	Ninguna	Aparece en juegos enviados a la ESRB y que esperan una clasificación final. Sin embargo, una vez calificados, toda la publicidad prelanzamiento debe contener la clasificación oficial del juego puesta por la ESRB. Algunos juegos, dependiendo de la intensidad de su contenido, pueden tener una renuncia de responsabilidad en la que aparece "Puede tener contenido inapropiado para menores" para muchos juegos que están clasificados con T (Teen), M (Mature) y AO (Adults Only).

Tabla 4. Clasificación en desuso de ESRB

Clasificación ESRB	Nombre	Año de Entrada	Edad Adecuada	Descripción de las clasificaciones
KA	Kids to Adults (Desde niños hasta adultos)	1994–1997	6 años	Material con contenido propicio para edades de 6 o más años. Estos títulos eran diseñados para niños y adultos de varias edades y gustos. Pueden contener violencia mínima, comicidad o algo de lenguaje crudo. Fue reemplazada por Everyone el 1 de enero de 1998

#### **4.1.4.1.2 Descripciones y definiciones de la ESRB**

Los descriptores de contenido aparecen en la parte posterior de la caja del juego y en anuncios impresos o sitios web que hagan referencia al juego. La ESRB tiene más de veinte descriptores de contenido (Entertainment Software Rating Board, ESRB Ratings Guide, 2013), como referencia al alcohol, violencia y desnudez. Abajo, los descriptores de contenido usados en la clasificación:

**Alcohol Reference (Referencia al alcohol):** Referencia e imágenes de bebidas alcohólicas.

**Animated Blood (Animación de sangre):** Representaciones decoloradas o no realistas de sangre.

**Blood (Sangre):** Representaciones de sangre.

**Blood and gore (Derramamiento de sangre):** Representaciones de sangre o de mutilación de partes del cuerpo.

**Cartoon Violence (Violencia de caricatura):** Acciones violentas que incluyen situaciones y personajes caricaturescos. Puede incluir violencia en la cual un personaje sale ileso después de que la acción se llevó a cabo.

**Comic Mischief (Travesuras cómicas):** Representaciones o diálogo que impliquen payasadas o humor sugestivo.

**Crude Humor (Humor vulgar):** Representaciones o diálogo que implique bromas vulgares, incluido el humor tipo “baño”.

**Drug References (Referencias a drogas):** Referencia o imágenes de drogas ilegales.

**Fantasy Violence (Violencia de fantasía):** Acciones violentas de naturaleza fantástica que incluyen personajes humanos y no humanos en situaciones que se distinguen con facilidad de la vida real.

**Intense Violence (Violencia intensa):** Representaciones gráficas y de apariencia realista de conflictos físicos. Puede comprender sangre excesiva o realista, derramamiento de sangre, armas y representaciones de lesiones humanas y muerte.

**Language (Lenguaje):** Uso de lenguaje soez de moderado a intermedio.

**Lyrics (Letra de canciones):** Referencias moderadas de lenguaje soez, sexualidad, violencia, alcohol o uso de drogas en la música.

**Mature Humor (Humor para adultos):** Representaciones o diálogo que contienen humor para adultos, incluidas las alusiones sexuales.

**Mild Violence (Violencia moderada):** Escenas de personajes dibujados en situaciones no seguras o violentas.

**Nudity (Desnudez):** Representaciones gráficas o prolongadas de desnudez.

**Partial Nudity (Desnudez parcial):** Representaciones breves o moderadas de desnudez.

**Real Gambling (Apuestas reales):** El jugador puede apostar, incluso colocar apuestas con dinero o divisas de verdad.

**Sexual Content (Contenido sexual):** Representaciones no explícitas de comportamiento sexual, tal vez con desnudez parcial.

**Sexual Themes (Temas sexuales):** Alusiones al sexo o a la sexualidad.

**Sexual Violence (Violencia sexual):** Representaciones de violaciones o de otros actos sexuales violentos.

**Simulated Gambling (Apuestas simulados):** El jugador puede apostar sin colocar apuestas con dinero o divisas reales.

**Strong Language (Lenguaje fuerte):** Uso explícito o frecuente de lenguaje soez.

**Strong Lyrics (Letra de canciones fuertes):** Alusiones explícitas o frecuentes de lenguaje soez, sexo, violencia o uso de alcohol o drogas en la música.

**Strong Sexual Content (Contenido sexual fuerte):** Alusiones explícitas o frecuentes de comportamiento sexual, tal vez con desnudez.

**Suggestive Themes (Temas insinuantes):** Referencias o materiales provocativos moderados.

**Tobacco Reference (Referencia al tabaco):** Referencia o imágenes de productos de tabaco.

**Use of Alcohol (Uso de alcohol):** Consumo de alcohol bebidas alcohólicas.

**Use of Drugs (Uso de drogas):** Consumo o uso de drogas.

**Use of Tobacco (Uso de tabaco):** Consumo o uso de productos de tabaco.

**Violence (Violencia):** Escenas que comprenden un conflicto agresivo. Pueden contener desmembramiento sin sangre.

**Violent References (Referencias violentas):** Alusiones a actos violentos.

## **4.2 Videojuegos para la computadora personal (PC)**

El mundo de los videojuegos para PC nació en los 70, cuando aún era casi impensable la sola idea de un gráfico. Durante la mayor parte de esa década, los ordenadores eran equipos costosos cuyos fines eran casi estrictamente profesionales.

Los juegos de esa década y muchos de los que vinieron después nacieron en el seno de las comunidades de programadores e ingenieros de software que desarrollaban juegos en sus ratos libres, pero no se puede hablar de una auténtica industria del videojuego de PC.

Durante los 80 y 90, los PC se consolidaron como plataforma de juego con centenares de títulos. Pero si hay una característica que ha prevalecido en el mundo de los juegos de PC es su vínculo a un perfil de usuario más técnico.

#### **4.2.1 Industria de videojuegos para PC presente y futuro**

En los 90 y 2000, las consolas volvieron a resurgir y hoy son el mercado mayoritario para la industria de los videojuegos, pero el mercado de juegos en PC sigue ostentando un público fiel, altamente técnico y aficionado a géneros, como la estrategia en tiempo real, que no han logrado adaptarse satisfactoriamente a consola. (Zackariasson, 2012)

En el ámbito de la distribución, es llamativo como las compañías han pasado de cobrar por el juego en sí, a cobrar por su uso. Buenos ejemplos de ellos son títulos masivos online donde la descarga del juego es gratuita o testimonial y lo importante es el pago por disfrutar del juego online en sus servidores. También son frecuentes los micro pagos por descarga de materiales extra o expansiones.

La nueva década llega cargada de novedades para el mundo de los videojuegos. Las nuevas tecnologías touch abren las puertas a nuevos dispositivos como los tablets pero no parece que los complejos juegos de PC y sus sistemas de control vayan a ceder fácilmente a sus primos táctiles. En todo caso las nuevos interfaces acabarán enriqueciendo el Juego en PC. Los retos actuales llegan más bien de la mano de la distribución y de competir en un entorno donde es frecuente que la competencia sea gratuita. (Zackariasson, 2012)

Al final, la clave de la supervivencia de los juegos en PC ha sido siempre su extraordinaria capacidad de adaptación, una comunidad unida y fiel, y el hecho de habitar un dispositivo que no sirve sólo para jugar ya que el teclado y el ratón

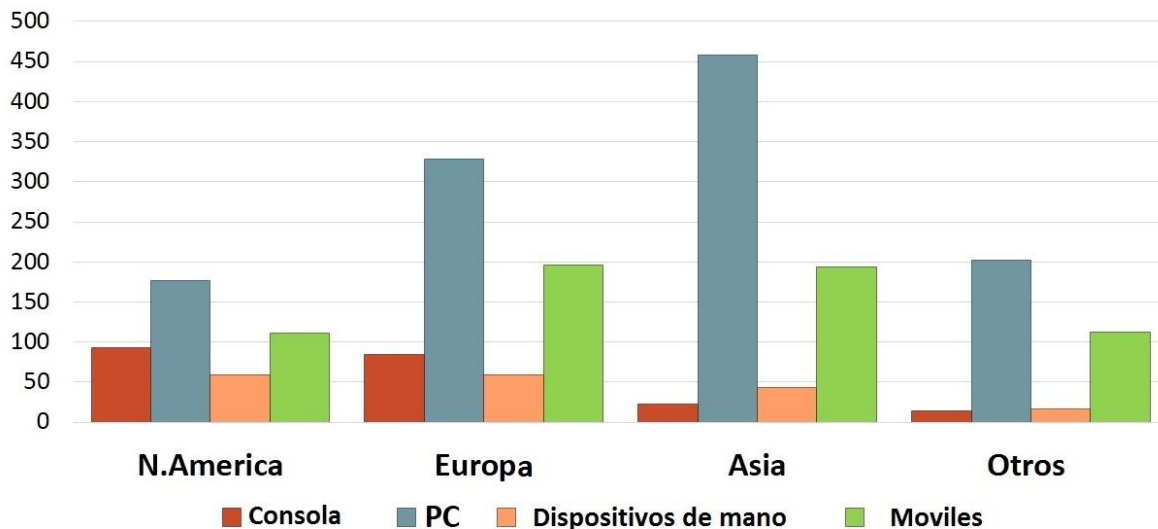
siguen siendo las armas predilectas en todos los torneos profesionales de videojuegos. (Zackariasson, 2012)

La PC Gaming Alliance, un consorcio de la industria sin fines de lucro dedicada a la Popularización del Gaming PC ha entregado los resultados anuales de la investigación realizada por la consultora DFC Intelligence.

Según los últimos datos ofrecidos por la PC Gaming Alliance, el mercado global de los videojuegos de PC volvió a subir durante el pasado 2012. En concreto el aumento ha sido del 8% con respecto a los datos de 2011, alcanzando un nuevo récord situado en los 20 mil millones de dólares. La PCGA, además, asegura que esto representa un crecimiento del 90% con respecto a los datos del primer informe de este tipo, llevado a cabo en el año 2008. (PC Gaming Alliance, 2012)

Según el analista de DFC David Cole a pesar de la atención de los medios a juegos para móviles y juegos de redes sociales que luchan, en la actualidad hay más de mil millones de jugadores de PC en todo el mundo (imagen 1) y ese número seguirá creciendo a medida que más ordenadores hayan conectados en línea, esto lo podemos (PC Gaming Alliance, 2012)

**Imagen 1. Usuarios de videojuegos por región y plataforma**



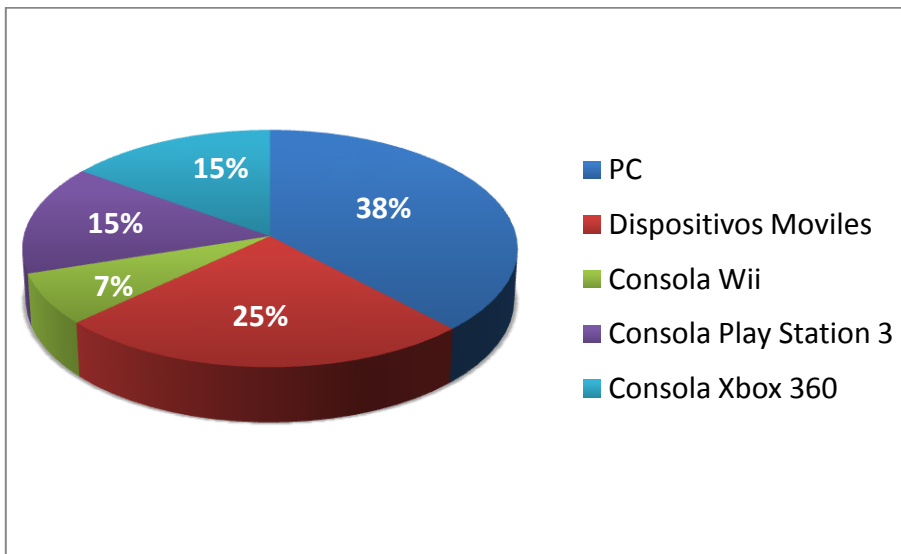


El informe de la PCGA además describe que China continúa siendo el mercado más grande y con mayor crecimiento en cuanto a juegos de PC con un récord de ingresos en 2012 de 6,800 millones dólares, lo que supone un crecimiento del 9%. Otros mercados que han crecido durante el año 2012 son Corea, Japón, Estados Unidos, Reino Unido y Alemania. (PC Gaming Alliance, 2012)

Finalmente el informe concluye asegurando que el mercado de videojuegos de PC seguirá creciendo a una tasa de crecimiento anual compuesto del 6% hasta alcanzar los 25,700 millones de dólares para el año 2016, principalmente debido al mayor acceso a conexiones de banda ancha y la facilidad de uso que ofrecen tanto la distribución digital como los métodos de pago globales. (PC Gaming Alliance, 2012)

Según el estudio, el mercado de consumo de los videojuegos quedara dividido de la siguiente manera: el 38% provendrá de los PC, seguido de las consolas con un 37% mientras que los dispositivos móviles se quedarán con un 25% como se observa en la imagen 1 (PC Gaming Alliance, 2012)

**Imagen 2. Ingresos globales videojuegos 2012**



## **4.2.2 Hardware para Videojuegos en PC**

### **4.2.2.1 Tarjeta Grafica**

Una tarjeta gráfica, tarjeta de vídeo, placa de vídeo, tarjeta aceleradora de gráficos o adaptador de pantalla, es una tarjeta de expansión para una computadora u ordenador, encargada de procesar los datos provenientes de la CPU y transformarlos en información comprensible y representable en un dispositivo de salida, como un monitor o televisor. (Grimsdale, 2011)

Las tarjetas gráficas no son dominio exclusivo de las Computadoras Personales, contaron o cuentan con ellas dispositivos como los Commodore Amiga, Apple II, Apple Macintosh, Spectravideo SVI-328, equipos MSX y por supuesto, así como también las videoconsolas modernas como la Wii U, la Playstation 3 y la Xbox360. (Grimsdale, 2011)

Los componentes principales de una tarjeta gráfica son: GPU (graphics processing unit) y Memoria gráfica de acceso aleatorio

**4.2.2.1.1 GPU (graphics processing unit):** Es un procesador (como la CPU) dedicado al procesamiento de gráficos; su razón de ser es aligerar la carga de trabajo del procesador central y por ello, está optimizada para el cálculo en coma flotante, predominante en las funciones 3D. La mayor parte de la información ofrecida en la especificación de una tarjeta gráfica se refiere a las características de la GPU, pues constituye la parte más importante de la tarjeta gráfica, así como la principal determinante del rendimiento. (Grimsdale, 2011)

Tres de las más importantes de sus características son la frecuencia de reloj del núcleo, el número de procesadores shaders y el número de pipelines (vertex y fragment shaders), encargadas de traducir una imagen 3D compuesta por vértices y líneas en una imagen 2D compuesta por píxeles.

### **a) Elementos generales de una GPU**

**Shaders:** Es el elemento más notable de potencia de una GPU. Son una evolución natural de los antiguos pixel shader (encargados de la rasterización de texturas) y vertex shader (encargados de la geometría de los objetos), los cuales anteriormente actuaban de forma independiente. Los shaders unificados son capaces de actuar tanto de vertex shader como de pixel shader según la demanda, aparecieron en el 2007 con los chips de Nvidia incrementando la potencia drásticamente respecto a sus familias anteriores. (Grimsdale, 2011)

**ROP:** Se encargan de representar los datos procesados por la GPU en la pantalla, además también es el encargado de los filtros.

#### **4.2.2.1.2 Memoria gráfica de acceso aleatorio**

Son chips de memoria que almacenan y transportan información entre sí, no son determinantes en el rendimiento máximo de la tarjeta gráfica, pero bien unas especificaciones reducidas pueden limitar la potencia de la GPU. Existen de dos tipos, Dedicada, es cuando la tarjeta gráfica o la GPU dispone exclusivamente para sí esas memorias, ésta manera es la más eficiente y la que mejores resultados da y la compartida cuando se utiliza memoria en detrimento de la memoria RAM, ésta memoria es mucho más lenta que la dedicada y por tanto su rendimiento es menor. (Boreskov, 2013)

### **a) Las características de memoria gráfica de una tarjeta gráfica**

**Capacidad:** La capacidad de la memoria determina el número máximo de datos y texturas procesadas, esta capacidad se mide en Bytes. Una capacidad insuficiente se traduce en un retardo a espera de que se vacíen esos datos. Sin embargo es un valor muy sobrevalorado como estrategia recurrente de márketing para engañar al consumidor, tratando de hacer creer que el rendimiento de una tarjeta gráfica se mide por la capacidad de su memoria. (Boreskov, 2013)

**Interfaz de Memoria:** También denominado Bus de datos, Es una característica importante y determinante, junto a la velocidad de la memoria, a la cantidad de datos que puede transferir en un tiempo determinado, denominado ancho de banda. Una analogía al ancho de banda se podría asociar al ancho de una autopista o carriles y al número de vehículos que podrían circular a la vez. La interfaz de memoria se mide en bits. (Boreskov, 2013)

**Velocidad de Memoria:** Es la velocidad a la que las memorias pueden transportar los datos procesados, por lo que es complemento a la interfaz de memoria para determinar el ancho de banda total de datos en un tiempo determinado. Continuando la analogía de la circulación de los vehículos de la autopista, la velocidad de memoria se traduciría en la velocidad máxima de circulación de los vehículos, dando resultado a un mayor transporte de mercancía en un mismo periodo de tiempo. La velocidad de las memorias se mide en Hertzios (su frecuencia efectiva) y se van diseñando tecnologías con más velocidad. (Boreskov, 2013)

**Ancho de banda:** Es la tasa de datos que pueden transportarse en una unidad de tiempo. Un ancho de banda insuficiente se traduce en un importante limitador de potencia de la GPU. Habitualmente se mide en "Gigabytes por segundo" (GB/s). (Boreskov, 2013)

**Tabla 5. Tipos de memoria para tarjetas gráficas actualmente vigentes en el mercado**

Tipo de Memoria	Frecuencia Efectiva(Mhz)	Ancho de Banda(Gb/s)
GDDR	166-950	1.2-30.4
GDDR2	533-1000	8.5-16
GDDR3	700-1700	5.6-54.4
GDDR4	1600-1800	64-86.4
GDDR5	3200-7000	24-448

#### **4.2.2.1.3 RAMDAC**

Es un conversor de señal digital a analógico de memoria RAM. Se encarga de transformar las señales digitales producidas en el ordenador en una señal analógica que sea interpretable por el monitor. Dada la creciente popularidad de los monitores de señal digital, el RAMDAC está quedando obsoleto, puesto que no es necesaria la conversión analógica si bien es cierto que muchos conservan conexión VGA por compatibilidad. (Boreskov, 2013)

#### **4.2.2.1.4 Salidas de una tarjeta grafica**

Los sistemas de conexión más habituales entre la tarjeta gráfica y el dispositivo visualizador (como un monitor o un televisor) son:

**a) SVGA/Dsub-15:** Estándar analógico diseñado e implementado desde los años 1990, diseñado para dispositivos CRT, sufre de ruido eléctrico y distorsión por la conversión de digital a analógico y el error de muestreo al evaluar los píxeles a enviar al monitor. Se conecta mediante pines. Su utilización continúa muy extendida al día de hoy, aunque claramente muestra una reducción frente al DVI en los últimos años. (Grimsdale, 2011)

**b) DVI:** Sustituto del anterior, pero digital, fue diseñado para obtener la máxima calidad de visualización en las pantallas digitales o proyectores. Se conecta mediante pines. Evita la distorsión y el ruido al corresponder directamente un píxel a representar con uno del monitor, en la resolución nativa del mismo. Cada vez más adoptado, aunque compite con el HDMI, pues el DVI no es capaz de transmitir audio. (Grimsdale, 2011)

**c) HDMI:** Tecnología propietaria transmisora de audio y vídeo digital de alta definición cifrado sin compresión en un mismo cable. Se conecta mediante patillas de contacto. No está pensado inicialmente para monitores, sino para Televisiones, por ello no apaga la pantalla cuando deja de recibir señal y debe hacerse manualmente en caso de monitores. (Grimsdale, 2011)

### **4.2.3 Computación Grafica**

La computación gráfica(también llamada CG) o gráficos por computadora es el campo de la informática visual, donde se utilizan computadoras tanto para generar imágenes visuales sintéticamente como integrar o cambiar la información visual y espacial probada del mundo real.

Este campo puede ser dividido en varias áreas: Interpretado 3D en tiempo real (a menudo usado en juegos de vídeo), animación de computadora, captura de vídeo y creación de vídeo interpretado, edición de efectos especiales (a menudo usado para películas y televisión), edición de imagen y modelado (a menudo usado para ingeniería y objetivos médicos). (Lueptow, 2007)

#### **4.2.3.1 Gráficos 2D por computadora**

El primer avance en la computación gráfica fue la utilización del tubo de rayos catódicos. Hay dos acercamientos a la gráfica 2d: vector y gráficos raster. La gráfica de vector almacena datos geométricos precisos, topología y estilo como posiciones de coordenada de puntos, las uniones entre puntos (para formar líneas o trayectos) y el color, el grosor y posible relleno de las formas. (Wong, 1972)

En la mayor parte de casos una imagen de vectores tiene que ser convertida a una imagen de trama o raster para ser vista.

#### **4.2.3.2 Rasters o Mapa de bits**

Los gráficos de tramas o raster (llamados comúnmente Mapa de bits) es una estructura o fichero de datos que representa una

rejilla rectangular de píxeles o puntos de color, denominada matriz, que se puede visualizar en un monitor, papel u otro dispositivo de representación. (Landa, 2006)

A las imágenes en mapa de bits se las suele definir por su altura y anchura (en píxeles) y por su profundidad de color (en bits por píxel), que determina el número de colores distintos que se pueden almacenar en cada punto individual, y por lo tanto, en gran medida, la calidad del color de la imagen.

Cada pixel tiene un valor específico como por ejemplo brillo, transparencia en color o una combinación de tales valores.

Los gráficos en mapa de bits se distinguen de los gráficos vectoriales en que estos últimos representan una imagen a través del uso de objetos geométricos como curvas de Bézier y polígonos, no del simple almacenamiento del color de cada punto en la matriz. El formato de imagen matricial está ampliamente extendido y es el que se suele emplear para tomar fotografías digitales y realizar capturas de vídeo. Para su obtención se usan dispositivos de conversión analógica-digital, tales como escáneres y cámaras digitales. (Landa, 2006)

Una imagen de trama tiene una resolución finita de un número específico de filas y columnas. Las demostraciones de computadora estándares muestran una imagen de trama de resoluciones como 1280 (columnas) x 1024 (filas) píxeles. Hoy uno a menudo combina la trama y lo gráficos vectorizados en formatos de archivo compuestos (pdf, swf, svg). (Landa, 2006)

#### **4.2.3.3 Gráficos 3D por computadora**

Un gráfico 3D difiere de uno bidimensional principalmente por la forma en que ha sido generado. Este tipo de gráficos se originan mediante un proceso de cálculos matemáticos sobre entidades geométricas tridimensionales producidas en un ordenador y cuyo propósito es conseguir una proyección visual en dos dimensiones para ser mostrada en una pantalla o impresa en papel. (Gortler, 2012)

En general, el arte de los gráficos tridimensionales es similar a la escultura o la fotografía, mientras que el arte de los gráficos 2D es análogo a la pintura. En los programas de gráficos por computadora esta distinción es a veces difusa: algunas aplicaciones 2D utilizan técnicas 3D para alcanzar ciertos efectos como iluminación, mientras que algunas aplicaciones 3D primarias hacen uso de técnicas 2D. (Gortler, 2012)

#### **4.2.3.3.1 Fases para la creación de elementos o gráficos tridimensionales**

**a) Modelado:** La etapa de modelado consiste en ir dando forma a objetos individuales que luego serán usados en la escena. Puede figurarse como estructuras de alambre, recubiertas de papel de colores. El proceso es realizar la malla de manera simple, para luego crear el material por el cual le daremos sus características tales como metal, barro, agua, etc. (Gortler, 2012)

Por tanto, primero se deben construir un modelo, para ello hay técnicas de modelo comunes, en las cuales se encuentran:

Modelado poligonal o subdivisión de superficies, NURBS (un modelo matemático muy utilizado en la computación gráfica para generar y representar curvas y superficies).

**b) Iluminación:** Creación de luces de diversos tipos puntuales, direccionales en área o volumen, con distinto color o propiedades. Esto es la clave de una animación. (Gortler, 2012)

Gran parte de la iluminación en 3D requiere del entendimiento físico de la luz en la realidad, este entendimiento puede ir desde lo más básico en el tema como por ejemplo el concepto de iluminación global hasta comportamientos complejos y extraños de la luz como la dispersión en superficies y subsuperficies. (Gortler, 2012)



## **Iluminación global**

La Iluminación Global, Iluminación indirecta, o GI (Global Illumination), se conoce a un conjunto de algoritmos que tratan de simular o aproximar como la luz emitida por alguna fuente, rebota en cada superficie de la escena iluminando espacios que la luz directa producida por la fuente no alcanzaría a iluminar. Los primeros Algoritmos de Iluminación indirecta buscaban simular la luz como fotones, de ahí uno de los algoritmos más implementados es el llamado Photon Map, existen otros algoritmos como el de Quasi Montecarlo, o algoritmos basados en Irradiance Caching, que dependiendo del motor de render con el que se trabaja tienen nombres diferentes e implementaciones propias. (Kerlow, 2009)

## **c) Animación**

Los objetos se pueden animar en cuanto a

**Transformaciones básicas:** en los tres ejes X, Y, Z, rotación, escala y traslación.

**Mediante esqueletos:** A los objetos se les puede asignar un esqueleto, una estructura central con la capacidad de afectar la forma y movimientos de ese objeto. Esto ayuda al proceso de animación, en el cual el movimiento del esqueleto automáticamente afectará las porciones correspondientes del modelo.

**Mediante deformadores:** ya sean cajas de deformación (lattices) o cualquier deformador que produzca, por ejemplo, una deformación sinusoidal.

**Dinámicas:** Para simulaciones de ropa, pelo, dinámicas rígidas de objeto.

La animación es muy importante dentro de los gráficos porque en estas animaciones se intenta imitar a la realidad misma, por esto es un trabajo que usualmente requiere muchas horas. (Kerlow, 2009)

**d) Renderizacion:** Renderizado (render en inglés) es un término usado en jerga informática para referirse al proceso de generar una imagen o vídeo mediante el cálculo de iluminación GI partiendo de un modelo en 3D. Este término técnico es

utilizado por los animadores o productores audiovisuales (CG) y en programas de diseño en 3D como por ejemplo 3DS Max, Maya, Blender, etc (O'Connor, 2010)

El proceso de renderizado se desarrolla con el fin de generar en un espacio 3D formado por estructuras poligonales, una simulación realista del comportamiento tanto de luces, texturas y materiales (agua, madera, metal, plástico, tela, etcétera) como también de los comportamientos físicos de la animación es el caso de la simulación de colisiones y fluidos, simulando ambientes y estructuras físicas verosímiles.

Las técnicas van desde las más sencillas, como el rénder de alambre (wireframe rendering), pasando por el rénder basado en polígonos, hasta las técnicas más modernas como el scanline rendering, el trazado de rayos, la radiosidad o el mapeado de fotones. (O'Connor, 2010)

El software de rénder puede simular efectos cinematográficos como ellens flare, la profundidad de campo, o el motion blur (desenfoque de movimiento). Estos artefactos son, en realidad, un producto de las imperfecciones mecánicas de la fotografía física, pero como el ojo humano está acostumbrado a su presencia, la simulación de dichos efectos aportan un elemento de realismo a la escena. Se han desarrollado técnicas con el propósito de simular otros efectos de origen natural, como la interacción de la luz con la atmósfera o el humo. Ejemplos de estas técnicas incluyen los sistemas de partículas que pueden simular lluvia, humo o fuego, el muestreo volumétrico para simular niebla, polvo y otros efectos atmosféricos y las cáusticas para simular el efecto de la luz al atravesar superficies refractantes.

El proceso de rénder necesita una gran capacidad de cálculo, pues requiere simular gran cantidad de procesos físicos complejos. La capacidad de cálculo se ha incrementado rápidamente a través de los años, permitiendo un grado superior de realismo en los rénders. Los estudios de cine que producen animaciones generadas por ordenador hacen uso, en general, de lo que se conoce como

render farm (granja de render) para acelerar la producción de fotografías. (O'Connor, 2010)

Una de las partes más importantes de los programas dedicados a la renderización es el motor de renderizado, el cual es capaz de realizar complejos cálculos como radiosidad, raytrace (trazador de rayos), canal alfa, reflexión, refracción o iluminación global (GI). Que permitirá que la simulación de condiciones físicas y lumínicas sean lo suficientemente realistas, llegando en muchos casos a ser difícil diferenciar una fotografía de un "dibujo". (O'Connor, 2010)

Cabe destacar que aun así, son programas de una gran complejidad de uso con una curva de aprendizaje muy elevada ya que no son intuitivos ni automatizados requiriendo una gran pericia de sus operadores para llegar a resultados óptimos.

Cuando se trabaja en un programa de diseño 3D por computadora, por lo general los resultados no pueden ser visualizados en tiempo real, no obstante la última generación de programas de renderizado han comenzado a modificar esta realidad apoyándose en las actuales placas de vídeo con procesadores dedicados permitiendo visualizar en tiempo real el acabado final de una escena 3D, aun así esto solo se suele aplicar a escenas sencillas no siendo aún aplicable para escenas de mayor complejidad ya que esto requiere una potencia de cálculo demasiado elevada por lo que se opta por crear el entorno 3D con una forma de visualización más simple para luego generar el lento proceso de renderización y así conseguir los resultados finales deseados. El tiempo de render depende en gran medida de los parámetros establecidos en los materiales y luces, así como de la configuración del motor de render. (O'Connor, 2010)

Normalmente cada aplicación de 3D cuenta con su propio motor de renderizado, pero cabe aclarar que existen plugins que se dedican a hacer el cálculo dentro del programa, utilizando fórmulas especiales. Es el caso de los conocidos motores V-Ray y Mental Ray, actualmente, los más populares dentro de los motores de renderizado. En el caso de los videojuegos, normalmente se utilizan imágenes "pre-renderizadas" para generar las texturas y así ayudar al equipo ya sea una

consola o un pc a trabajar en el entorno virtual con mucha más fluidez, (aun así los actuales juegos tiene una gran demanda tanto de procesador, placa de vídeo y memoria Ram)

**e) Shading (Sombreado):** El proceso de sombreado o shading implica la simulación de (o más exactamente; el cálculo) como las caras de un polígono se comportarán cuando es iluminado por una fuente de la luz virtual. El cálculo exacto varía según no sólo que datos están disponibles sobre la cara sombreada, sino también la técnica de sombreado.

Generalmente este afecta propiedades de la especularidad y valores de intensidad, reflexión y transparencia. (O'Connor, 2010)

- Bump mapping (Correlación de relieve): Es una técnica en la cual la dirección hacia donde apunta un polígono solía simular superficies desiguales o arrugadas y con relieve.
- Ray Tracing (Trazador de rayos): Un método basado en los principios físicos de la óptica geométrica que puede simular reflexiones múltiples y la transparencia.
- Radiosidad: una técnica para la iluminación global que usa la teoría de transferencia de radiación para simular la iluminación reflejada indirecta en escenas con superficies difusas.

**f) Texturizado:** Las superficies poligonales o secuencia de caras, pueden contener datos correspondientes de más de un color, pero en el software más avanzado, pueden ser una lona virtual para una imagen, u otra imagen rasterizada. Tal imagen es colocada en una cara, o la serie de caras y es llamada Textura.

Las texturas añaden un nuevo grado de personalización en cómo las caras y los polígonos cuidarán la forma en que serán sombreados, según el método de sombreado, y como la imagen es interpretada durante el sombreado. (O'Connor, 2010)

#### **4.2.3.3.2 Interfaces de programación de Aplicaciones para el diseño 3D**

Los gráficos 3D se han convertido en algo muy popular, particularmente en videojuegos, al punto que se han creado interfaces de programación de aplicaciones (API) especializadas para facilitar los procesos en todas las etapas de la generación de gráficos por computadora. Estas interfaces han demostrado ser vitales para los desarrolladores de hardware para gráficos por computadora, ya que proveen un camino al programador para acceder al hardware de manera abstracta, aprovechando las ventajas de tal o cual placa de video. (O'Connor, 2010)

Las siguientes interfaces para gráficos por computadora son particularmente las más populares:

**a) OpenGL:** Es una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos 2D y 3D. La interfaz consiste en más de 250 funciones diferentes que pueden usarse para dibujar escenas tridimensionales complejas a partir de primitivas geométricas simples, tales como puntos, líneas y triángulos. Fue desarrollada originalmente por Silicón Graphics Inc. (SGI) en 1992 y se usa ampliamente en CAD, realidad virtual, representación científica, visualización de información y simulación de vuelo. También se usa en desarrollo de videojuegos, donde compite con Direct3D en plataformas Microsoft Windows. (Shereiner, 2013)

OpenGL tiene dos propósitos esenciales:

Ocultar la complejidad de la interfaz con las diferentes tarjetas gráficas, presentando al programador una API única y uniforme.

Ocultar las diferentes capacidades de las diversas plataformas hardware, requiriendo que todas las implementaciones soporten la funcionalidad completa de OpenGL (utilizando emulación software si fuese necesario).

El funcionamiento básico de OpenGL consiste en aceptar primitivas tales como puntos, líneas y polígonos, y convertirlas en píxeles. Este proceso es realizado por una pipeline gráfica conocida como Máquina de estados de OpenGL.

La mayor parte de los comandos de OpenGL bien emiten primitivas a la pipeline gráfica o bien configuran cómo la pipeline procesa dichas primitivas. Hasta la aparición de la versión 2.0 cada etapa de la pipeline ejecutaba una función prefijada, resultando poco configurable. A partir de la versión 2.0 algunas etapas son programables usando un lenguaje de programación llamado GLSL. (Shereiner, 2013)

OpenGL es una API basada en procedimientos de bajo nivel que requiere que el programador dicte los pasos exactos necesarios para renderizar una escena. Esto contrasta con las API descriptivas, donde un programador sólo debe describir la escena y puede dejar que la biblioteca controle los detalles para representarla. El diseño de bajo nivel de OpenGL requiere que los programadores conozcan en profundidad la pipeline gráfica, a cambio de darles libertad para implementar algoritmos gráficos novedosos. (Shereiner, 2013)

OpenGL ha influido en el desarrollo de las tarjetas gráficas, promocionando un nivel básico de funcionalidad que actualmente es común en el hardware comercial, algunas de esas contribuciones son:

- Primitivas básicas de puntos, líneas y polígonos rasterizados.
- Una pipeline de transformación e iluminación.
- Z-buffering.
- Mapeado de texturas.
- Alpha blending
- 

**b) GDI:** Graphics Device Interface (GDI), es uno de los tres componentes o subsistemas de la interfaz de usuario de Microsoft Windows. Trabaja junto con el núcleo y la API de Windows. (Petzold, 2007)

Esta Interfaz de programación de aplicaciones se encarga del control gráfico de los dispositivos de salida como los monitores o las impresoras. Las tareas más comunes de GDI son el dibujo de líneas, curvas, polígonos y el manejo de paletas. (Petzold, 2007)

GDI no se encarga del dibujo de los menús, ventanas, etc. este es un trabajo especial para el user32.dll (una biblioteca de la API de Windows exclusiva para estas funciones).

Tal vez la más significativa capacidad del GDI sobre métodos más directos de acceder al hardware es la escalabilidad de posibilidades y la abstracción que se puede conseguir con los dispositivos de salida. Usando GDI es muy fácil dibujar en varios dispositivos, como es en la pantalla o en una impresora, esta capacidad es el objetivo principal del WYSIWYG de Microsoft Windows. (Petzold, 2007)

La mayoría de los videojuegos simples se pueden crear usando GDI. Sin embargo, no es lo suficientemente poderosa para la creación de videojuegos complejos debido a la falta de sincronización con el framebuffer (encargado de las animaciones) y su obvia incapacidad para la renderización de modelos y objetos 3D. Los videojuegos modernos utilizan DirectX u OpenGL que resuelven estos problemas.

Los objetos más importantes son:

HDC: (handle to device context; "contexto de dispositivo"). Que viene a ser el lienzo donde se pinta.

HBRUSH: (handle to brush; "brocha"). Permite rellenar zonas con colores.

HPEN: (handle to pen; "pluma"). Que permite el dibujado de líneas o bordes de otros elementos gráficos.

La forma de trabajar es heredada del lenguaje de programación C. Hay que crear y destruir explícitamente los elementos de dibujo, por lo que es muy propenso a fugas de memoria. Microsoft está en vías de reemplazar dicho sistema de dibujo por otro completamente distinto y basado en C++ denominado GDI+. (Petzold, 2007)

**c) Direct3D:** Direct3D es parte de DirectX (conjunto de bibliotecas para multimedia), propiedad de Microsoft. Consiste en una API para la programación de

gráficos 3D. Está disponible tanto en los sistemas Windows de 32 y 64 bits, como para sus consolas Xbox y Xbox 360. (Zink, 2011)

El objetivo de esta API es facilitar el manejo y trazado de entidades gráficas elementales, como líneas, polígonos y texturas, en cualquier aplicación que despliegue gráfico en 3D, así como efectuar de forma transparente transformaciones geométricas sobre dichas entidades. Direct3D provee también una interfaz transparente con el hardware de aceleración gráfica. (Zink, 2011)

Se usa principalmente en aplicaciones donde el rendimiento es fundamental, como los videojuegos, aprovechando el hardware de aceleración gráfica disponible en la tarjeta gráfica.

Direct3D es uno de los múltiples componentes que contiene la API DirectX de Windows. Se le podría situar al nivel del GDI de Windows, presentando un nivel de abstracción entre una aplicación de gráficos 3D y los drivers de la tarjeta gráfica. (Zink, 2011)

La mayor ventaja que presenta Direct3D frente al GDI es que Direct3D se comunica directamente con los drivers de pantalla, consiguiendo mejores resultados en la representación de los gráficos por pantalla que aquel.

Direct3D está compuesto por dos grandes API. El modo retenido y el modo inmediato. El modo inmediato da soporte a todas las primitivas de procesamiento 3D que permiten las tarjetas gráficas (luces, materiales, transformaciones, control de profundidad, etc.). El modo retenido, construido sobre el anterior, presenta una abstracción de nivel superior ofreciendo funcionalidades pre construidas de gráficos como jerarquías o animaciones. El modo retenido ofrece muy poca libertad a los desarrolladores, siendo el modo inmediato el que más se usa. (Zink, 2011)

El modo inmediato de Direct3D trabaja fundamentalmente con los llamados dispositivos (devices). Son los encargados de realizar la renderización de la escena. El dispositivo ofrece una interfaz que permite diferentes opciones de



renderización. Por ejemplo un dispositivo mono permite la renderización en blanco y negro mientras que un dispositivo RGB permite el uso de colores (Zink, 2011). Podemos clasificar los dispositivos en tres clases principales:

**Dispositivo HAL (hardware abstract layer):** Permite aceleración hardware. Se trata del dispositivo más rápido.

**Dispositivo de referencia:** Es necesaria la instalación previa del SDK de Direct3D para poder usar este tipo de dispositivo. Permite la simulación software de un tipo de renderización y resulta muy útil cuando el hardware todavía no tiene incorporadas nuevas características de renderización. Un caso muy concreto del dispositivo de referencia es el Null reference device cuya función es presentar la pantalla en negro. Se usa por defecto cuando se intenta usar un dispositivo de referencia y no se encuentra el SDK.

**Dispositivos de conexión software (pluggable software device):** Permite opciones de rasterización software. Previamente se ha tenido que obtener el dispositivo mediante el método RegisterSoftwareDevice. Este tipo de dispositivos no fueron usados hasta DirectX 9.0.1

Cada dispositivo tiene asociada una o más cadenas de intercambio (swap chains). Dichas cadenas están compuestas por varios buffers de superficies, considerando a una superficie como un conjunto de píxeles más todos los atributos asociados a cada uno de ellos como la profundidad, el color, la transparencia (canal alfa), etc.

Además, los dispositivos tienen asociados también una colección de recursos (resources) o datos concretos necesarios para realizar la renderización. Cada uno de estos recursos tiene los siguientes atributos:

**Tipo (type):** Define el tipo de recurso del que se trata: superficie, volumen, textura, textura de cubo, textura de volumen, textura de superficie, buffer de vértices o buffer de índices.

**Almacén (pool):** Describe dónde se almacena el recurso durante la ejecución. Default indica que se almacena junto con el dispositivo, managed que se guarda en la memoria del sistema y se copia en el dispositivo cuando éste lo necesita, system memory que se encuentra exclusivamente en la memoria del sistema, al igual que scratch, ignorando este último las restricciones de la tarjeta gráfica.

**Formato (format):** describe el formato en que se almacena el recurso en memoria. La información más importante es respecto al almacenamiento de los píxeles en memoria. Un ejemplo de valor de format es D3DFMT\_R8G8B8 que indica que una profundidad de color de 24 bits (los 8 de más peso para el rojo, los 8 de en medio para el verde y los 8 de menos peso para el azul).

**Uso (usage):** Mediante una lista de flags, indica cómo se va a usar el recurso. También permite distinguir los recursos estáticos, aquellos que una vez cargados solo interesa su valor, de los recursos dinámicos, cuyo valor se modifica repetidamente.

#### **Etapas del proceso de renderización en direct3D:**

**Input Assembler:** Aporta los datos de entrada (líneas, puntos y triángulos).

**Vertex Shader:** Se encarga de las operaciones de vértices (iluminación, texturas, transformaciones). Trata los vértices individualmente.

**Geometry Shader:** Realiza operaciones con entidades primitivas (líneas, triángulos o vértices). A partir de una primitiva, el geometry shader puede descartarla, o devolver una o más primitivas nuevas.

**Stream Output:** Almacena la salida de la etapa anterior en memoria. Resulta útil para realimentar la pipeline con datos ya calculados.

**Rasterizer:** Convierte la imagen 3D en píxeles.

**Pixel Shader:** Operaciones con los píxeles.

**Output Merger:** Se encarga de combinar la salida del pixel shader con otros tipos de datos, como los patrones de profundidad, para construir el resultado final.

Direct3D permite la reconfiguración de todas las etapas, aumentando considerablemente la flexibilidad de esta pipeline.

#### **Modos de presentación en pantalla:**

**Pantalla completa o exclusive mode:** gracias a que Direct3D tiene conexión directa con el driver de la pantalla, el dispositivo Direct3D hace uso exclusivo de la pantalla. Ninguna otra aplicación podrá hacer uso de la pantalla mientras se encuentre en este modo. (Zink, 2011)

**En ventana o windowed mode:** El resultado se muestra dentro de una ventana de Windows. Direct3D tiene que colaborar con el GDI para finalizar dicha representación. Aunque este modo resulta más lento que el anterior, al no bloquear la pantalla y permitir el uso de otras aplicaciones, resulta muy cómodo en la depuración de errores. (Zink, 2011)

#### **4.2.3.4 Animación**

La animación es un proceso utilizado para dar la sensación de movimiento a imágenes o dibujos o a otro tipo de objetos inanimados (figuras de plastilina, por ejemplo). Se considera normalmente una ilusión óptica. Existen numerosas técnicas para realizar animación que van más allá de los familiares dibujos animados. Los cuadros se pueden generar dibujando, pintando o fotografiando los minúsculos cambios hechos repetidamente a un modelo de la realidad o a un modelo tridimensional virtual; también es posible animar objetos de la realidad y actores. Entre los formatos de animación (o que soportan animación) se encuentran el GIF, el SWF (animación flash), etc. Las animaciones en GIF son guardadas imagen por imagen, pero existen animaciones que no se logran así, sino que son interpretadas y "armadas" en tiempo real al ejecutarse (como las de formato SWF). (Kerlow, 2009)

La animación es una ilusión óptica en la que una serie de objetos o de imágenes estáticos, al ser visionados unos detrás de otros a una cierta velocidad, crean la ilusión del movimiento natural.

Esta ilusión óptica se basa en el principio de persistencia de la visión (también llamado principio de persistencia de la retina), establecido en 1829 por el físico belga Joséph Plateau. Aunque Plateau estableció que según el principio de persistencia de la visión hacía falta un mínimo de 10 visionados por segundo para que la realidad apareciese a nuestros ojos con una fluidez natural (y no como una sucesión abrupta de imágenes fijas) en el cine y en otras artes que usan tecnología visual o audiovisual se ha establecido un estándar de captura y proyección de imágenes de 24 fotogramas por segundo. (Kerlow, 2009)

Hace falta una cantidad mínima de imágenes por segundo para que una animación parezca fluida y natural y no una sucesión abrupta de imágenes claramente diferenciadas unas de otras. Sea cual sea esa cantidad mínima se habla de animación limitada cuando la animación está constituida por una frecuencia de imágenes inferior al mínimo. Se habla en cambio de animación completa cuando se cumple o se sobrepasa el mínimo de imágenes por segundo.

#### **8.1.4.4.1 Animación por Computadora**

La animación por computadora (también llamada animación digital, animación informática o animación por ordenador) es la técnica que consiste en crear imágenes en movimiento mediante el uso de ordenadores o computadoras. (Kerlow, 2009)

Algunas veces el objetivo de la animación es la computación en sí misma, otras puede ser otro medio, como una película. Los diseños se elaboran con la ayuda de programas de diseño, modelado y por último renderizado. En la animación, sin embargo, las imágenes no se toman sino que se producen individualmente, y por ello no tienen que cumplir necesariamente con el estándar del cine. Una película de animación tiene siempre 24 fotogramas por segundo, pero no necesariamente todos esos fotogramas muestran imágenes diferentes.

Se puede considerar el aporte de la tecnología informática en dos campos: como herramienta de creación y como medio de representación.

Para las animaciones dibujadas o pintadas a mano hay programas que asisten a la creación de los cuadros intermedios. Cabe recordar que se necesita una gran cantidad de éstos para dar la sensación de movimiento. En las animaciones hechas con gráficos vectoriales y con modelos tridimensionales el programa mismo calcula la transformación (interpola) de una pose a otra.

Diversos formatos de archivo permiten representar animación en una computadora, y a través de Internet. Entre los más conocidos están SWF, GIF, MNG y SVG. El archivo puede contener una secuencia de cuadros, como gráficos rasterizados (o la diferencia entre un cuadro y el anterior), o puede contener la definición de trazos y sus deformaciones en el tiempo, en un formato vectorial. Hay formatos de archivo específicos para animaciones, y también se utilizan formatos genéricos que pueden contener diversos tipos de multimedia. (Kerlow, 2009)

Para crear la ilusión del movimiento, una imagen se muestra en pantalla sustituyéndose rápidamente por una nueva imagen en un fotograma diferente. Esta técnica es idéntica a la manera en que se logra la ilusión de movimiento en las películas y en la televisión.

Para las animaciones 3D, los objetos se modelan en la computadora (modelado) y las figuras 3D se unen con un esqueleto virtual (huesos). Para crear una cara en 3D se modela el cuerpo, ojos, boca, etc. del personaje y posteriormente se animan con controladores de animación. Finalmente, se renderiza la animación.

En la mayor parte de los métodos de animación por ordenador, un animador crea una representación simplificada de la anatomía de un personaje, pues tiene menos dificultad para ser animada. En personajes bípedos o cuadrúpedos, muchas partes del esqueleto del personaje corresponden a los huesos reales. La animación con huesos también se utiliza para animar otras muchas cosas, tales

como expresiones faciales, un coche u otro objeto que se quiera dotar de movimiento. (Kerlow, 2009)

En contraste, otro tipo de animación más realista sería la captura de movimiento, que requiere que un actor vista un traje especial provisto de sensores, siendo sus movimientos capturados por una computadora y posteriormente incorporados en el personaje.

Para animaciones 3D, los fotogramas deben ser renderizados después de que el modelo es completado. Para animaciones vectoriales 2D, el proceso de renderizado es clave para el resultado. Para grabaciones grabadas anticipadamente, los fotogramas son convertidos a un formato diferente o a un medio como una película o video digital. Los fotogramas pueden ser renderizados en tiempo real, mientras estos son presentados al usuario final. Las animaciones para transmitir vía Internet en anchos de banda limitados (ejem. 2D Flash, X3D) utilizan programas en el ordenador del usuario para renderizar en tiempo real la animación como una alternativa para la transmisión y para animaciones pre-cargadas para enlaces de alta velocidad. (Kerlow, 2009)

Las imágenes por segundo (fotogramas por segundo o cuadros por segundo, en inglés frames per second o FPS) es la medida de la frecuencia a la cual un reproductor de imágenes genera distintos fotogramas (frames). En informática estos fotogramas están constituidos por un número determinado de píxeles que se distribuyen a lo largo de una red de texturas. La frecuencia de los fotogramas es proporcional al número de píxeles que deben generarse, incidiendo en el rendimiento del ordenador que los reproduce.

Para engañar al ojo y al cerebro para que alguien piense que está viendo un objeto en movimiento, las imágenes deben ser mostradas a alrededor de 12 imágenes o marcos por segundo o más rápido. Con velocidades superiores a los 70 frames/segundo, no se notará una mejoría en el realismo o suavidad en el movimiento de la imagen debido a la manera en que el ojo y cerebro procesan las imágenes. A velocidades menores a 12 frames/segundo la mayoría de las

personas podrán detectar un parpadeo en el momento en que se muestre la secuencia de imágenes y disminuirá la ilusión de un movimiento realista. Animaciones convencionales realizadas a mano, normalmente utilizan 15 frames/segundo con el objetivo de disminuir la cantidad de dibujo que se requiere, pero esto es normalmente aceptado debido a la naturaleza de los dibujos animados. Por esto, para crear una animación por ordenador realista, se requiere una cantidad superior de frames/segundo. (Kerlow, 2009)

El motivo de que a altas velocidades no sea perceptible el parpadeo de la imagen, es por la «persistencia de la visión». De momento a momento, el ojo y cerebro trabajando juntos almacenan cualquier cosa que se esté mirando por una fracción de segundos, y automáticamente realiza «saltos» pequeños y suaves. Las películas que se exhiben en los cines, corren a 24 frames/segundo, que es suficiente para crear esta ilusión de movimiento continuo.

### **Métodos para animar personajes virtuales**

En la mayor parte de los sistemas de animación 3D, un animador crea una representación simplificada del cuerpo del personaje, análogo a un esqueleto o stick figure. La posición de cada segmento del modelo del esqueleto es definida por variables de la animación o Avars. (Kerlow, 2009)

En personajes humanos y animales, muchas partes del modelo de esqueleto corresponden a la ubicación real de los huesos, pero la animación del modelo de esqueleto skeletal animation es también utilizada para animar otras cosas, como expresiones faciales (a pesar de que existen otros métodos de animación facial). Woody, el personaje de Toy Story, por ejemplo, usa 700 Avars, incluyendo 100 Avars en la cara. La computadora no renderiza el modelo de esqueleto directamente de forma habitual el esqueleto es invisible, pero usa el modelo de esqueleto para calcular la posición exacta y la orientación del personaje, que es eventualmente renderizado en una imagen. Cambiando los valores de los Avars sobre la línea de tiempo, el animador crea movimiento realizando el personaje cuadro por cuadro. (Kerlow, 2009)

Existen varios métodos para generar los valores de Avars para obtener un movimiento realista. Tradicionalmente, los animadores manipulan los Avars directamente. En lugar de crear Avars para cada cuadro, usualmente colocan los Avars en puntos estratégicos de los cuadros y permiten a la computadora realizar la transición entre ellos, este proceso es llamado keyframing.

Keyframing pone el control en las manos del animador, y está basado en la animación manual (traditional animation.)

En contraste, un método nuevo llamado captura de movimiento (motion capture), utiliza acción real (live action). Cuando la animación por ordenador es realizada por esta técnica, un actor real realiza la escena como si fuera el personaje a ser animado. Su movimiento es grabado en una computadora utilizando cámaras de vídeo y marcadores, y ese movimiento es aplicado al personaje animado.

Cada método tiene sus ventajas y hasta el 2007, juegos y películas utilizaban alguno o ambos de ellos en sus producciones. La animación por cuadros puede producir movimientos que serían imposibles de realizar para un actor, mientras que la captura de movimiento puede reproducir las características de un actor en particular. Por ejemplo, en la película del 2006 *Pirates of the Caribbean: Dead Man's Chest*, el actor Bill Nighy realizó el personaje Davy Jones. Incluso aunque él prácticamente no aparece en la película, la producción se benefició grabando las características de su lenguaje corporal, posturas, expresiones faciales, etc. Esta «captura de movimiento» es apropiada en situaciones en las que se requiere un comportamiento realista, pero que las características de un personaje excede lo que se puede realizar con maquillaje y vestuario convencional. (Kerlow, 2009)

Ha habido casos de animaciones excelentes en diversas películas, la de *Final Fantasy The Spirits Within*, el cabello de Violeta en *Los Increíbles*, la animación de escenarios en *Toy Story*, el movimiento del cabello de Tifa al viento en *Final Fantasy Advent Children*, la animación del agua en la *Danza del Agua* del videojuego *Final Fantasy X* y la creación de monstruos en *Monsters Inc.*



### **4.3 Game Engines-Motores de videojuegos**

Un “Game Engine” es un sistema compuesto por varias herramientas para la creación y desarrollo de videojuegos. Dicho sistema incluye funcionalidad para la compilación, renderización de gráficos 2D y 3D, sonido, animación, inteligencia artificial, sistema distribuido, paralelización, asignación de memoria y gráficos. También se puede considerar como una serie de rutinas de programación que permiten el diseño, la creación y la representación de un videojuego. Una ventaja de estos motores es el poder de utilizar todas las herramientas para crear diferentes juegos, logrando que el desarrollo de un juego sea mucho más rápido y económico. (De la Dueña, 2011)

Los “Game Engine” proporcionan un conjunto de herramientas de desarrollo visual, además de componentes de software reutilizables. Estas herramientas se proporcionan generalmente en un entorno de desarrollo integrado para permitir el desarrollo de juegos simplificado y rápido.

La mayoría de las suites Game Engine proporcionan instalaciones que facilitan el desarrollo, tales como gráficos, sonido, física y AI funciones. Estos Game Engine a veces son llamados "middleware", ya que proporcionan una plataforma de software flexible y reutilizable que proporciona todas las funcionalidades básicas necesarias para desarrollar una aplicación de juego. Es una herramienta creada por programadores y para programadores. (De la Dueña, 2011)

Al igual que otras soluciones middleware, los Game Engine suelen proporcionar abstracción de la plataforma, permitiendo que el mismo juego se pueda ejecutar en varias plataformas, como consolas de videojuegos y ordenadores personales con pocos o ningún cambio realizado en el código fuente del juego. A menudo, los Game Engine están diseñados con una arquitectura basada en componentes que permite a ciertos sistemas específicos ser sustituidos o ampliados por otros más especializados (y, a menudo más

caros), como Havok para la física, Miles Sound System para el sonido, o Bink para el vídeo.

Algunos motores, como RenderWare están diseñados como una serie de componentes independientes que pueden ser combinados para crear un motor personalizado, en lugar del enfoque más común de extender o personalizar una solución integrada flexible. Sin embargo, la extensibilidad sigue siendo una alta prioridad en los "Game Engine" debido a la gran variedad de usos para los que se aplican. (De la Dueña, 2011)

A pesar de la especificidad del nombre, los Game Engine a menudo se utilizan para otros tipos de aplicaciones con gráficos en tiempo real, como demostraciones de marketing, visualizaciones arquitectónicas, simulaciones y entornos de modelado.

Otros sólo proporcionan capacidades de tiempo real en 3D en lugar de la amplia gama de funcionalidad que necesitan los juegos. Estos motores se basan en el desarrollador para crear el resto de la funcionalidad o conseguirla mediante otros componentes de middleware. Estos tipos de motores son llamados generalmente "motor de gráficos", "motor de renderizado", o "motor 3D" en lugar del término más general "motor de juegos". Algunos ejemplos de los motores gráficos son: Crystal Space, Genesis3D, Irrlicht, motor JMonkey, OGRE, RealmForge, Truevision3D y el Vision Engine. (De la Dueña, 2011)

#### **4.3.1 Historias de los motores de videojuegos**

Antes de los Game Engine, los juegos eran desarrollados generalmente como entidades singulares: un juego para el Atari 2600, por ejemplo, tuvo que ser diseñado desde muy bajo nivel para hacer un uso óptimo del hardware de los gráficos.

Otras plataformas tenían más libertad de acción, pero incluso cuando los gráficos no eran una preocupación, las limitaciones de la memoria por lo general sabotaban los intentos de crear el diseño que un motor necesitaba. Incluso

en plataformas con más capacidad, muy poco se podía reutilizar entre los juegos. Así la mayoría de los diseños de juegos en la década de 1980 fueron diseñados con un fuerte conjunto de reglas con una pequeña cantidad de datos y gráficos. (De la Dueña, 2011)

La primera generación de motores de gráficos o renderizadores (y precursor de lo que hoy conocemos como motores) fue dominada por tres entidades: Brender de Argonaut Software, Renderware de Criterion Software Limited y RenderMorphics de Reality Lab.

Reality Lab fue el más rápido de los tres y fue el primero en ser adquirido por parte de Microsoft. El equipo de RenderMorphics: Servan Keondjian, Seekings Kate y Rabson Doug se sumaron a los proyectos de Microsoft que hizo cambiar la empresa Reality Lab en Direct3D. Renderware fue comprado finalmente por EA (Electronic Arts), pero fue marginado por el gigante de los juegos. (De la Dueña, 2011)

El término "Game Engine" surgió a mediados de la década de 1990, especialmente en relación con los juegos 3D como los "shooters" en primera persona (FPS). Tal fue la popularidad de los juegos "Doom" y "Quake" que en lugar de trabajar a partir de cero, muchos desarrolladores compraron la licencia de las partes fundamentales del software y a partir de ellas diseñaron sus propios gráficos, personajes, armas y niveles. La separación de las normas específicas del juego y los datos básicos, como la detección de colisiones y la entidad del juego, produjo que los equipos pudiesen crecer y especializarse.

Más tarde, los juegos, como "Quake III Arena" y "Unreal" de "Epic Games" de 1998, fueron diseñados con este enfoque en mente, con el motor y el contenido desarrollado de forma separada. La práctica de la concesión de licencias tecnológicas, como se ha demostrado, es una corriente útil de ingresos auxiliares para algunos desarrolladores de videojuegos, ya que una licencia para un motor de juego de alta calidad puede variar desde 10,000 a millones de dólares y el número de licenciatarios pueden alcanzar varias docenas de empresas, como se

ha visto con el Unreal Engine. Los motores permiten el desarrollo de secuelas de juegos de una manera más rápida y fácil, que es una ventaja valiosa en la industria del video juego. (De la Dueña, 2011)

Los motores modernos de juegos son algunas de las aplicaciones más complejas creadas, a menudo con docenas de sistemas finamente sincronizados que interactúan entre ellos controlados con precisión. La continua evolución de los motores de juegos ha creado una fuerte separación entre la renderización, guión y diseño de los niveles.

Aunque el término de Game Engine fue utilizado por primera vez en la década de 1990, hay algunos sistemas a principios de la década de 1980 que también son considerados como motores de juegos, tales como Adventure Game Interpreter (AGI) de Sierra, el sistema SCUMM de LucasArts y el motor Freescape de Incentive Software. A diferencia de la mayoría de los motores de juego moderno, estos motores de juego nunca fueron utilizados en los productos de terceros. (De la Dueña, 2011)

#### **4.3.2 Game Middleware**

En el sentido más amplio del término, los motores de juegos se pueden describir como middleware. En el contexto de los videojuegos, sin embargo, a menudo es usado para referirse a los subsistemas de la funcionalidad dentro de un motor de juegos. Algunos middleware realizan sólo una cosa, pero hace que sea más convincente y más eficiente que el middleware sea de propósito general. Por ejemplo, SpeedTree se utiliza para representar los árboles y la vegetación en un videojuego. (De la Dueña, 2011)

Los cuatro paquetes de middleware más utilizados que proporcionan la funcionalidad de los subsistemas incluyen: Bink RAD Game Tools, Firelight FMOD, Havok y GFX Scaleform.

RAD Game Tools Bink se desarrolla para la prestación básica de vídeo, junto con el audio Miles y Granny 3D. Firelight FMOD es una herramienta y librería de audio. Havok ofrece un sistema de simulación física robusta, junto con un conjunto de soluciones de animación y comportamiento. Scaleform ofrece GFX de alto rendimiento Flash UI, junto con una solución de vídeo de alta calidad de reproducción. (De la Dueña, 2011)

Algunos middleware contienen el código fuente completo mientras que otros sólo ofrecen una API que hace referencia a una biblioteca. Algunos programas de middleware pueden necesitar un cierto tipo de licencia, generalmente para un acceso más amplio para el código fuente completo.

### **4.3.3 Tendencias actuales**

La tecnología del Game Engine va madurando volviéndose cada vez más fácil de utilizar. La aplicación de los motores se ha ampliado en su alcance. Ahora están siendo utilizados para visualización, formación, medicina y aplicaciones militares de simulación. Para facilitar esta accesibilidad, las nuevas plataformas de hardware están siendo dirigidas por los motores de juego, incluidos los teléfonos móviles (por ejemplo, los teléfonos Android e iPhone) y navegadores web (por ejemplo, Shockwave, Flash, Webvision Trinigy es, Silverlight, Unity Web Player, O3D y dhtml puro). (De la Dueña, 2011)

Además, están surgiendo motores de juegos que se basan en lenguajes de alto nivel tales como Java y C#.NET (por ejemplo, TorqueX y Visual3D.NET) o Python (Panda3D). La mayoría de los juegos 3D están ahora limitados por GPU (es decir, limitada por la potencia de la tarjeta gráfica), la ralentización del potencial de lenguajes de alto nivel es insignificante, mientras que el aumento de la productividad que ofrecen estos idiomas trabaja en beneficio de los desarrolladores del motor de juegos. Estas tendencias actuales están siendo impulsadas por empresas como Microsoft para apoyar el desarrollo

independiente (Indie) de juegos. Microsoft desarrolló XNA como el SDK de elección para todos los juegos lanzados para Xbox y productos relacionados.

#### **4.3.4 Tipos de Motores**

Existen multitud de motores con diferente nivel de conocimientos en programación. Para ver lo diferentes que pueden llegar a ser unos de otros se van a mostrar tres tipos de motores: roll-your-own version, mostly-ready version y el motor point-and-click. (De la Dueña, 2011)

##### **4.3.4.1 Motores “roll your own - créatelo tú mismo” (nivel más bajo)**

A pesar del costo, muchas empresas (así como los responsables de juegos Indie) tratan de desarrollar sus propios motores. Esto significa que utilizan aplicaciones disponibles al público, como las API de DirectX, OpenGL, para crear sus propios motores. Además, recurren a otras bibliotecas, tanto comerciales o de código abierto, para su implementación. Estas bibliotecas pueden incluir funcionalidades para la física, funcionalidades para los gráficos y funcionalidades GUI.

En general, estos sistemas proporcionan a los programadores la mayor flexibilidad, permitiéndoles escoger y elegir los componentes que quieren e integrarlos exactamente como ellos quieren. Pero también se pierde mucho tiempo en su construcción. Además, los programadores con frecuencia se tienen que construir todas las herramientas a partir de cero, ya que rara vez se puede confiar en que todas las bibliotecas utilizadas trabajen juntas a la perfección. Esto hace al motor muy poco atractivo para la mayoría de los desarrolladores de juegos. (De la Dueña, 2011)

##### **4.3.4.2 Motores “Mostly ready - todo listo” (nivel intermedio)**

Estos motores están listos desde el primer momento para trabajar con ellos, proporcionando renderización, interfaz gráfica de usuario, física, etc. Muchos de ellos incluso poseen un conjunto de herramientas las cuales no hay que configurar para su funcionamiento conjunto (roll-your-own) (De la Dueña, 2011).

Todos estos motores requieren todavía un poco de programación para lograr poner en marcha un juego completo. Se puede llegar a necesitar código de bajo nivel para hacer trabajar un juego. Muchos de estos motores son el producto de decenas de personas trabajando a lo largo de cientos de horas, y proporcionarán un mejor rendimiento con menos esfuerzo que la mayoría de los motores del primer tipo, incluso si no llegan a realizar exactamente lo que se quiere.

#### **4.3.4.3 Motores “Point-and-click - apuntar y pulsar” (nivel más alto)**

Motores de este tipo son cada vez más comunes hoy en día. Incluyen una herramienta completa que permite la creación de un juego propio. Estos motores, como GameMaker, Torque Game Builder y Unity3D, están diseñados para ser lo más amigable posible, y para utilizar tan poco código como sea posible. Eso no quiere decir que teniendo conocimientos de programación no ayuda, pero no es una necesidad.

El problema con muchos motores de este tipo es que pueden llegar a ser extremadamente limitados. Muchos proporcionan uno o dos tipos de géneros, o uno o dos tipos de modos gráficos. Incluso frente a las restricciones de estas herramientas, es posible hacer juegos muy creativos o incluso encontrar formas creativas en torno a esas restricciones. Lo mejor de estos motores es que permiten trabajar de forma rápida sin necesidad de mucho trabajo.

#### **4.3.5 Lista de motores**

En las siguientes tablas se muestra el listado de los motores más importantes con licencia gratuita y código abierto con libre distribución del código y desarrollo libre.

Tabla 6. Motores Free y Open Source

Nombre	Descripción
Aleph One	Motor desarrollado en el lenguaje C y utilizado para implementar el juego "Marathon". Licencia tipo GPL
Axiom Engine	Motor desarrollado en lenguaje C# con licencia GPL.
Blender	Motor desarrollado en lenguaje C++ con licencia GPL.
Box2D	Motor desarrollado en lenguaje C++ con licencia MIT. Un juego desarrollado con este motor es "Angry Birds".
Build engine	Motor desarrollado en lenguaje C y licencia Custom. Un ejemplo de juego es "Duke Nuken 3D".
Crystal Space	Motor desarrollado en lenguaje C++ con licencia LGPL
Cube	Motor desarrollado en lenguaje C++ y su licencia es zlib.
Delta3d	Motor desarrollado en lenguaje C++ con licencia LGPL.
Flexible Isometric Free Engine	Motor desarrollado en lenguaje C++ y su licencia es LGPL. Un ejemplo de juego desarrollado con este motor es "Unknown Horizons".
Genesis3D	Motor desarrollado en lenguaje C.
id Tech	Motor desarrollado en lenguaje C con licencia GPL. Ejemplos de juegos creados con este motor son "Doom", "Doom 2", "HeXen", "Quake II" y "Quake III Arena".
OpenSceneGraph	Motor desarrollado en lenguaje C++.
Panda3D	Motor desarrollado en lenguaje C++ y licencia BSD. El juego "Pirates of the Caribbean" ha sido creado con este motor
PixelLight	Motor desarrollado en lenguaje C++ con licencia LGPL.
Quake engine	Motor desarrollado en lenguaje C con licencia GPL.
Second Life	Motor desarrollado en lenguaje C++.



**Tabla 7. Motores Freeware**

<b>Nombre</b>	<b>Descripción</b>
Adventure Game Studio	Este motor es uno de los más populares a la hora de desarrollar juegos de aventuras Amateur.
DX Studio	Motor freeware con un conjunto muy completo de herramientas para el desarrollo de videojuegos en 3D. Posee una licencia de pago para tener acceso a ciertas funcionalidades.
Unity	Motor muy completo en el que se puede desarrollar para web, Windows y Mac. Obteniendo una licencia se puede desarrollar para iPhone, Android, Nintendo Wii, Playstation 3 y la Xbox 360.
Unreal Engine	Completo motor para desarrollar juegos para PC. Obteniendo una licencia se puede desarrollar juegos para Xbox 360 y PS3.

**Tabla 8. Motores Propietarios**

Nombre	Descripción
Bork3D Game Engine	Motor para desarrollar juegos de plataformas para iPhone
BigWorld	Servidor, cliente y herramientas de desarrollo de MMOG
BRender	Motor gráfico 3D en tiempo real para simuladores y herramientas gráficas
DX Studio	Motor que permite la creación de juegos en tiempo real y simuladores.
Enigma Engine	Motor de juegos de táctica en tiempo real, utilizado para Blitzrieg.
Freescape (1986)	Uno de los primeros motores 3D propietarios, utilizado.
Gamestudio	Sencillo motor para principiantes.
Infinity Engine	Permite la creación de videojuegos de rol.
M.U.G.E.N	Creado por "Elechbyte" para juegos de lucha e 2D.
Q (game engine)	Completo framework desarrollado por el equipo de Direct3D que permite el desarrollo para PC, Wii, PS2, PS3, Xbox, Xbox 360, PSP, iPhone etc.
RAGE	Creado por "Rockstar Games" para mejorar sus juegos para Xbox 360 y PlayStation 3. Utilizado para el juego Grand Theft Auto 4.
Scaleform	Motor gráfico utilizado para visualizar elementos Adobe Flash, HUDs y texturas animadas para juegos en PC, Mac, Linux, Xbox 360, PlayStation 2, PlayStation Portable, PlayStation 3 y Wii.
Source engine	Desarrollado por "Valve Software" para el juego "Half Life 2".
Zillions of Games	Utilizado para desarrollar juegos de tablero, como ajedrez.

#### 4.3.6 Unreal Engine

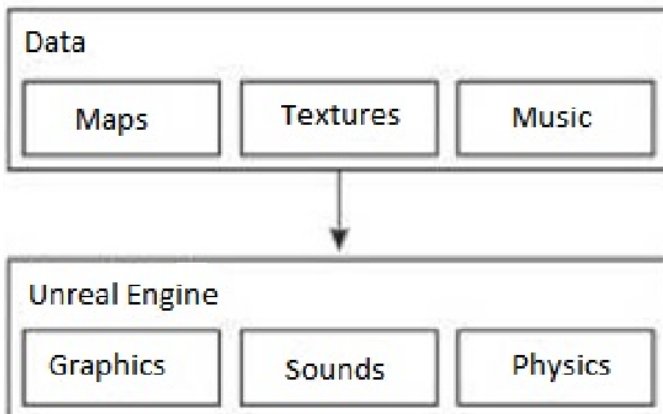
"Unreal Engine" es un motor de juegos desarrollado por Epic Games. El primer desarrollo con este motor fue "Unreal" en 1998, shooter en primera persona, que ha sido la base de muchos juegos desde entonces, incluyendo Unreal Tournament, Deus Ex, Turok, Tom Clancy's Rainbow Six 3: Raven Shield, Tom Clancy's Rainbow Six: Vegas, America's Army, Red Steel, Gears of War, Bioshock, Bioshock 2, Star Wars Republic Commando, Operaciones Tácticas: Asalto contra el terrorismo, Borderlands, Destroy All Humans! Path of the Furon, Batman: Arkham Asylum. (De la Dueña, 2011)

Unreal Engine, en pocas palabras, es un sistema que organiza tus propios elementos, como personajes, diseños, armas, efectos de sonido, música, etc, dentro de un gran entorno interactivo visual.

Aunque primeramente se utilizó para shooters en primera persona, se ha utilizado con éxito en una gran variedad de géneros, incluyendo el sigilo (Tom Clancy's Splinter Cell), MMORPG (Vanguard: Saga of Heroes), así como en juegos de rol como Mass Effect, The Last Remnant, Matar Piso y Harry Potter y la piedra filosofal. (De la Dueña, 2011)

Con su núcleo escrito en C++, Unreal Engine cuenta con un alto grado de portabilidad siendo una herramienta utilizada por los desarrolladores de juegos soportando a una gran multitud de plataformas, incluyendo Microsoft Windows, Linux, IOS, Mac OS y Mac OS X en ordenadores personales y muchas consolas de videojuegos como Dreamcast, Xbox, Xbox 360, PlayStation 2 y PlayStation 3, aunque la última versión del Unreal Engine, UE3, no funciona en plataformas de generación anterior. Una gran parte del código de un juego se puede escribir con UnrealScript, un lenguaje de programación propietario, pero la mayoría del juego se puede modificar sin ahondar profundamente en el motor interno. Además, al igual que los paquetes de middleware, Unreal Engine ofrece diversas herramientas para la creación de contenidos, tanto para los diseñadores como los artistas.

### Imagen 3. Entorno Unreal Engine



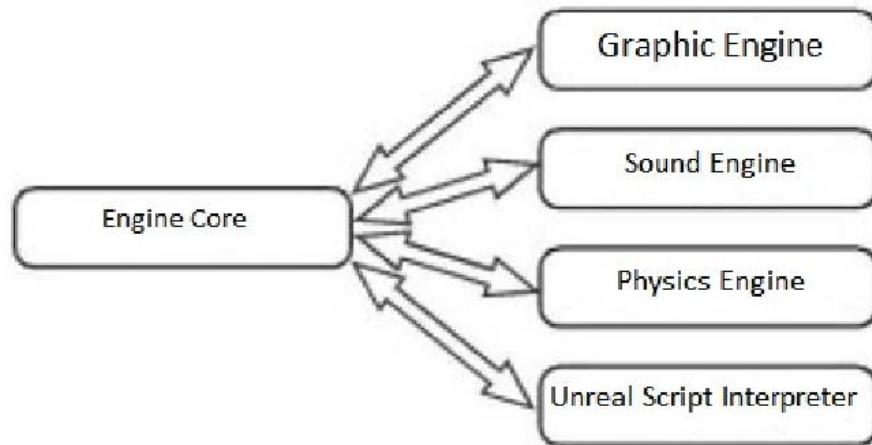
#### 4.3.6.1 Componentes

Cada uno de los componentes de este motor pueden trabajar independientemente, pero todo el poder de desarrollo de este motor viene de hacer trabajar a todos conjuntamente en armonía gracias a un núcleo central. (De la Dueña, 2011)

El Nucleo del Engine o Engine Core, contiene los siguientes componentes los cuales funcionan en conjunto:

- Graphics Engine
- Sound Engine
- Physics Engine
- UnrealScript Interpreter

**Imagen 4. Componentes de Unreal Engine**



##### 4.3.6.1.1 Motor Gráfico (Graphics Engine)

El motor gráfico controla todo lo que se muestra por pantalla. Sin este motor, sólo se vería una gran lista de valores como la localización del jugador, vida, armas que posee el jugador, entre otras. El motor recoge toda esta información y presenta el entorno mostrado durante el juego.

Es responsable de realizar todos los cálculos necesarios denominados “behind - the-scenes” (detrás de las escenas), como qué objetos se muestran delante de otros, y cuáles ocultan la visualización de otros.

Cuando algunos objetos son ocultados por otros, éstos no son renderizados por completo. Esta manera de trabajar permite liberar ciclos de trabajo para optimizar la eficiencia.

Una parte muy importante en la renderización es la iluminación. El motor proporciona un sistema robusto de iluminación. Permite una iluminación y sombreado dinámico, con lo que aumenta el realismo. (De la Dueña, 2011)

#### **4.3.6.1.2 Motor de sonido (Sound Engine)**

El 90 por ciento de lo que se percibe de un entorno es por medio del oído. El sonido es una parte importantísima en los juegos, aunque gráficamente sea de lo mejor. El sonido en un juego permite mejorar el realismo: el sonido producido por el jugador al pasar por distintos terrenos, la rotura de un cristal, recarga del arma, etc.

El motor de sonido permite obtener todos estos sonidos en el juego. Los sonidos a utilizar en un juego, el motor los acopla a los eventos que se produzcan, para reproducirlos simultáneamente, utilizando el hardware de sonido que posee el equipo. (De la Dueña, 2011)

#### **4.3.6.1.3 Motor de Físicas (Physics Engine)**

Todos los efectos que se producen en el mundo real pueden ser simulados en un juego mediante simulación e implementación. Estos efectos son simulados gracias al motor de físicas PhysX.

PhysX soporta casi la totalidad de simulaciones físicas, como cuerpos estáticos e incluso cuerpos dinámicos. Cada uno de ellos puede ser interactivo con los eventos del juego y pueden ser manipulados por el jugador.

Este motor puede trabajar en conjunto con el motor de sonido para obtener efectos de sonido dinámicos y realistas. (De la Dueña, 2011)

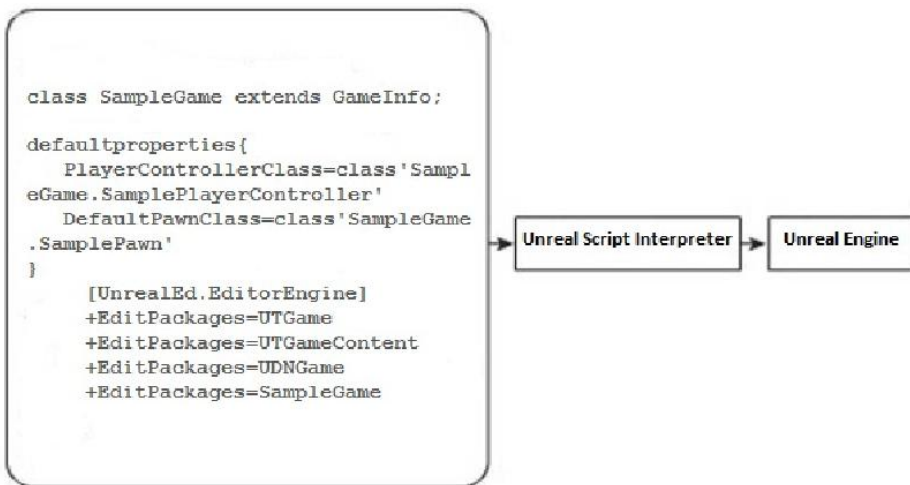
#### 4.3.6.1.4 Interprete UnrealScript

UnrealScript es uno de los aspectos más revolucionarios del motor Unreal Engine. Éste es un lenguaje de programación propio que permite realizar cambios en el motor, sin tener que modificar el código fuente.

Este lenguaje es muy similar a los lenguajes Java y C++, pero intentando que fuese algo más sencillo de aprender y utilizar.

Un intérprete de este lenguaje es el componente del motor responsable de transformar el código escrito en UnrealScript en código que el motor puede procesar. (De la Dueña, 2011)

#### Imagen 5. Interprete UnrealScript



#### 4.3.6.2 Manejador de entradas

Sin poder manejar al personaje principal de un juego, no se tiene juego, se tendrá una animación de éste.

Las entradas de un juego pueden ser: botones presionados, movimientos del ratón y la forma más moderna, mandatos vocales.

Cada tipo de entrada debe ser reconocido y convertido a mandatos que produzcan una reacción en el juego. Éste es el trabajo a realizar por el manejador de entradas.

Cada vez que se produce una entrada en el juego, el manejador crea una serie de propiedades para realizar los cambios oportunos en dicho juego. Por ejemplo, cuando el jugador presiona la tecla de “salto”, el manejador envía la señal al núcleo del motor para reproducir la animación de salto del personaje principal y el envío al motor gráfico la información necesaria para renderizar la acción en pantalla, a la vez que se envía la señal al motor de sonido para reproducir el correspondiente sonido. Claro está, que toda esta secuencia de acciones se produce de una manera muy rápida sin producirse un retardo entre la entrada y la reacción producida. (De la Dueña, 2011)

#### **4.3.6.3 Infraestructura en red**

Una opción actual a la hora de desarrollar un juego es la de multijugador (multiplayer). Durante un juego multijugador, los equipos de todos los jugadores deben intercambiar información a través de la red.

La forma más eficiente de realizar todo esto es limitar la información que es enviada a través de la red a la más importante del juego: posición de cada jugador y la interacción inmediata con el entorno. Enviando sólo la información más relevante se consigue un juego rápido aún con conexiones lentas. (De la Dueña, 2011)

#### **4.3.6.4 Historia del Motor UnrealEngine**

En el verano del año 1998 se produjo el nacimiento del motor Unreal Engine. Epic Games no estaba creando un juego, buscaba un sistema de herramientas y programas para poder crear cualquier juego.

Unreal Engine proporcionó a Epic Games, y a las compañías con licencia, un buen punto de partida de desarrollo de videojuegos. De repente, no era necesario recrear todo desde el principio y se podía combinar el motor con la implementación.

#### **4.3.6.4.1 Unreal Engine 1.3**

Dicha versión del motor Unreal Engine ya integraba renderización, detección de colisiones, IA, física, distribución del sistema y sistema de gestión de archivos en un motor completo. Era necesario realizar algunas concesiones para mantener los niveles de rendimiento con el hardware que se disponía en ese momento. Por ejemplo, Epic decidió utilizar "la detección de colisiones cilíndrica" sobre la "detección de colisiones IK" en un esfuerzo por mantener la tasa de fotogramas en equipos que eran comunes en el momento de su lanzamiento. Internamente, Epic desarrolló los juegos Unreal y Unreal Tournament mediante este motor.

Unreal Engine 1 utilizó la API Glide, desarrollada específicamente para 3dfx GPU y se ha actualizado para OpenGL y Direct3D. Probablemente la razón principal de su popularidad fue que la arquitectura del motor y la inclusión de un lenguaje de scripting lo hizo muy fácil de utilizar. Otra de las mejoras de Unreal en comparación con los otros motores era su tecnología de red, que en gran medida mejoraba la escalabilidad del motor en modo multijugador. Unreal fue también el primero en utilizar un verdadero modelo cliente-servidor en la arquitectura del motor. (De la Dueña, 2011)

Entre Marzo y Julio de 2001, Epic Games lanzó "The Unreal Developer Network", también llamado UDN. Mediante esta web, los usuarios de Unreal Engine pueden encontrar documentación y tutoriales actualizados sobre este motor. A día de hoy, "udn.epicgames.com" es el sitio de referencia online para el desarrollo con Unreal Engine.

#### **4.3.6.4.2 Unreal Engine 2**

En esta versión el núcleo y el motor de renderizado fueron completamente re-escritos y con el nuevo UnrealEd 2 integrado. También integró el Karma SDK para la física, lo que potencia las físicas de los personajes en Unreal Tournament 2003 y Unreal Championship (2003), enfocados totalmente al tipo de juego multiplayer (multijugador). (De la Dueña, 2011)



Otro elemento que integra esta versión es la herramienta Matinee, un sistema integrado para la creación de animaciones de los objetos de un videojuego.

Muchos elementos del motor también se actualizaron, con mejoras y añadiendo soporte para las consolas PlayStation 2, GameCube y Xbox. Con UE2.5, una actualización, se mejoró el rendimiento de la representación física y añadió físicas de vehículos y el editor de las partículas para UnrealEd y soporte de 64 bits en Unreal Tournament 2004. (De la Dueña, 2011)

Una versión especializada de UE2.5 llamada UE2X, que fue utilizada para Unreal Championship 2, desarrolla optimizaciones específicas de la primera generación de Xbox. EAX 3.0 y también compatible con sonido.

El 24 de marzo de 2011, Ubisoft Montreal reveló que habían hecho funcionar con éxito el Unreal Engine 2.5 en la nueva plataforma de Nintendo, Nintendo 3DS.

#### **4.3.6.4.3 Unreal Engine 3**

La tercera generación de Unreal Engine (UE3) fue diseñada para DirectX 9/10/11 en PC, Xbox 360 y PlayStation 3. Su renderización soporta muchas técnicas avanzadas, iluminación per-pixel, sombras dinámicas y basada en las herramientas disponibles en las versiones anteriores del motor. (De la Dueña, 2011)

Epic ha utilizado esta versión del motor para sus juegos Unreal Tournament 3, Gears of War, y una versión mejorada de Gears of War 2. Epic Games demostró con el juego Gears of War toda la capacidad que poseía esta versión del Unreal Engine. Un estilo especial de cámara en tercera persona, un rico y extremadamente detallado mundo y uno de los más viscerales estilos de combate. Con todas estas propiedades, "Gears of War" llegó a ser uno de los grandes éxitos en los videojuegos. (De la Dueña, 2011)

Debido a la concesión agresiva de licencias, esta versión actual se ha ganado un gran apoyo por parte de grandes desarrolladores, como Atari, Activision, Capcom,

Disney, Konami, Koei, 2K Games, Midway Games, THQ, Ubisoft, Sega, Sony, Electronic Arts, Square Enix, 3D Realms y muchos más.

Epic Games ha anunciado en la GDC 2009 una serie de mejoras al motor de Unreal 3. Entre ellos:

- Iluminación estática de alta calidad con efectos de última generación, como sombras suaves, penumbras y la reflexión especular.
- La posibilidad de agregar efectos de fractura de mallas estáticas para simular entornos destructibles.
- Suavizado de los cuerpos físicos.
- Simulación de grandes multitudes.

Además de la industria del juego, UE3 ha sido adoptado por muchos proyectos de diseño y simulación de construcción, simulación de conducción, centros comerciales virtuales y guiones de cine. (De la Dueña, 2011)

#### **4.3.6.4.4 Unreal Development Kit (UDK)**

Mientras que el Unreal Engine 3 ha sido bastante abierto para desarrollar proyectos con él, la capacidad de publicar y vender juegos hechos con UE3 está restringida bajo licencia. Sin embargo, en noviembre de 2009, Epic lanzó una versión gratuita de su motor, llamada Unreal Development Kit (UDK), que está disponible para el público en general. De acuerdo con la actual EULA, los desarrolladores de juegos pueden vender sus juegos mediante el pago a Epic de \$99 en un principio, y el 25% de todos los ingresos si son superiores a 50.000\$. Además, en diciembre de 2010, UDK viene con soporte para la creación de juegos iOS. (De la Dueña, 2011)

#### **4.3.6.4.5 Unreal Engine 4**

Mark Rein, vicepresidente de Epic Games, reveló el 18 de agosto de 2005 que el Unreal Engine 4 había estado en desarrollo desde 2003. La única persona que trabajaba en el sistema de Unreal Engine 4 hasta ese momento era Tim Sweeney, director técnico y fundador de Epic Games. Sin embargo, en la 2006 Game Developers Conferencia, Tim Sweeney dijo que el desarrollo no se iniciaría en serio hasta algún momento de 2008. (De la Dueña, 2011)

Predijo que la próxima generación de juegos de consolas llegaría en 2009, en la que los diseñadores deberían trabajar con CPUs con 20 o más núcleos, 80 o más subprocesos de hardware, y más de un teraflop de potencia de cálculo.

Michael Capps, presidente de Epic Games, indicó que el Unreal Engine 4 debería estar listo alrededor de 2012.

#### **4.4 Ingeniería del Software**

La ingeniería del software es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema hasta el mantenimiento de este después que se utiliza. (Sommerville, 2005)

En esta definición, existen dos frases clave:

- **Disciplina de la ingeniería:** Los ingenieros hacen que las cosas funcionen. Aplican teorías, métodos y herramientas donde sean convenientes, pero la utilizan de forma selectiva y siempre tratando de descubrir soluciones.
- **Todos los aspectos de la producción del Software:** La ingeniería del Software no solo comprende los procesos técnicos de desarrollo de software si no también actividades tales como la gestión de proyectos de software y el desarrollo de herramientas, métodos y teorías de apoyo a la producción del software

#### **4.4.1 Procesos para el desarrollo de Software**

Los modelos prescriptivos de procesos definen un conjunto distinto de actividades, acciones, tareas, fundamentos y productos de trabajo que se requieren para desarrollar software de alta calidad. Estos modelos de software no son perfectos, pero proporcionan una guía útil para el trabajo de la ingeniería del software (Pressman, 2006).

A estos modelos se les llama “prescriptivos” por que prescriben un conjunto de elementos del proceso tales como: actividades del marco de trabajo, acciones de ingeniería del software, tareas, productos del software, aseguramiento de la calidad del software y mecanismos de control del cambio para cada proyecto. (Pressman, 2006)

Dentro de los modelos prescriptivos tenemos los siguientes:

- Modelo en cascada ( SDLC)
- Modelos de procesos incrementales
  - ✓ Modelo incremental
  - ✓ Modelo DRA
  - ✓ Scrum Framework
- Modelos de procesos evolutivos
  - ✓ Construcción de prototipos
  - ✓ Modelo en espiral
  - ✓ Modelo de desarrollo concurrente
- Modelos especializados de procesos
  - ✓ Desarrollo basado en componentes
  - ✓ Modelo de métodos formales
  - ✓ Desarrollo del software orientado a aspectos
- Proceso unificado
- Programación Extrema (XP)

A continuación se define el modelo de proceso con el que se desarrollará el Video juego.

## 4.4.2 Un proceso para el desarrollo de videojuegos: “Huddle”

### 4.4.2.1 Descripción del proceso Huddle

Se llama Huddle a la reunión que se realiza en el juego antes de cada jugada en el fútbol americano; la filosofía es que mediante breves reuniones de planeación a corto plazo, se planee cada “jugada” que se inicie; con esto se da un seguimiento más estrecho al avance del proyecto y es posible hacer correcciones tempranas a posibles desviaciones. El proceso de desarrollo Huddle está diseñado para ser utilizado en proyectos de desarrollo de videojuegos y está basado en dos procesos de desarrollo de software: Modelo de desarrollo en Cascada y el Scrum Framework. (Morales Gerardo, 2010)

Todo el proceso se divide en 3 fases: Preproducción, Producción, Postmortem.

### Imagen 6. Las tres fases del proceso Huddle



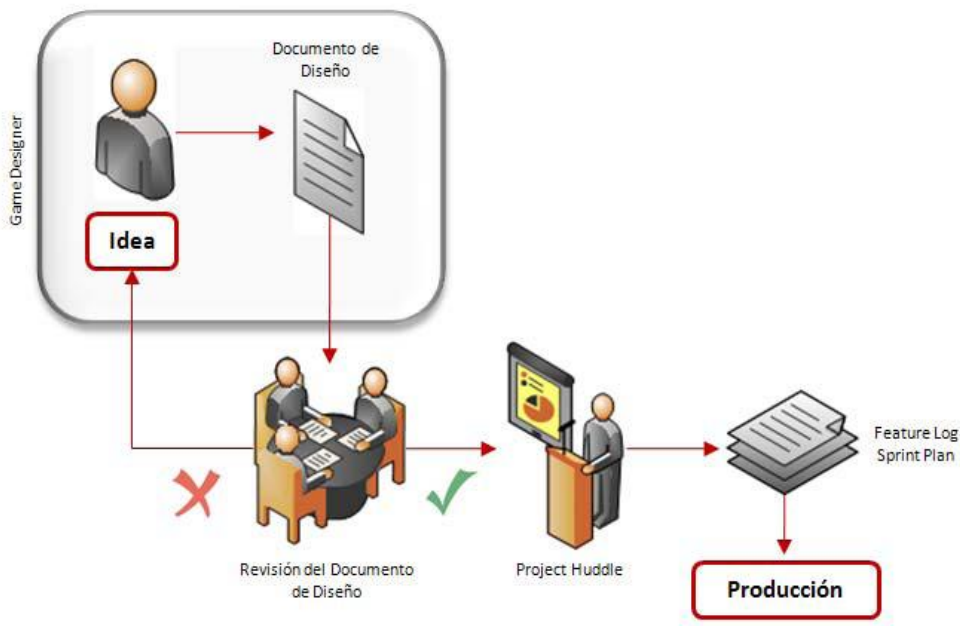
#### 4.4.2.2 Fase de Preproducción

La planeación de un proyecto es la clave para obtener un producto de calidad y que sea desarrollado dentro del tiempo y costos estimados. La fase de Preproducción tiene como objetivo migrar la idea del diseñador a dos artefactos llamados Sprint Plan y Feature Log; estos son documentos de planeación que serán la guía en la de la siguiente fase: producción del videojuego. (Morales Gerardo, 2010)

El Sprint Plan es un documento de planeación, cada sprint (evento) está compuesto por una meta a alcanzar, un número de identificador del evento, fecha de inicio y fin, el número de días que tomara alcanzar la meta.

El Feature Log es un documento que contiene cada una de las tareas a realizar para cumplir con la meta de cada uno de los sprints, está compuesto por un número de identificador, el identificador del Sprint al que pertenece, el número de días que tomara llevar a cabo la actividad, y la descripción de la actividad.

**Imagen 7. Modelo de fase Preproducción**



Uno de los propósitos principales de esta fase es el análisis del proyecto, fase en la que se revisará y se aceptará, en su caso, la idea inicial y se realizará la planeación completa de la fase de producción, en esta fase es necesaria la participación de todos los miembros involucrados en el proyecto.

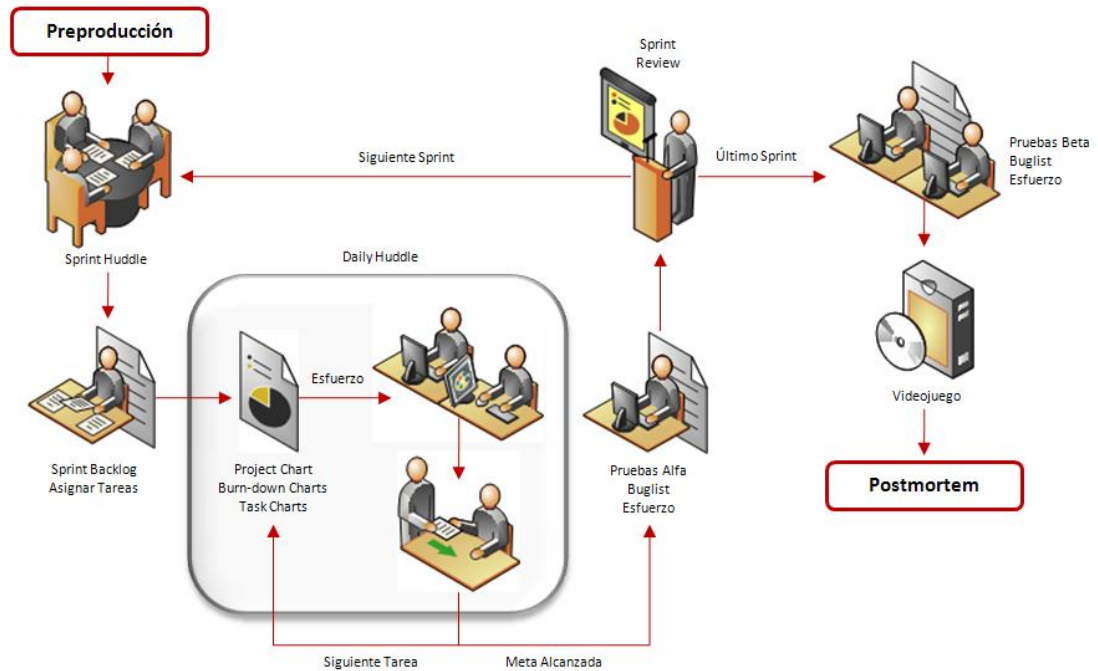
Inicialmente se parte de un documento de diseño que expresa formalmente la idea principal y detalles de la propuesta de videojuego. Este documento es revisado con la finalidad de saber si es factible, en caso contrario se modificará el documento de diseño hasta que sea aprobado o en su caso rechazado definitivamente. Si el proyecto es aprobado se pasará al Project Huddle esta es una etapa de preproducción en el cual se hará la planeación completa del proyecto para poder pasar a la fase de producción, una vez terminada esta actividad se deberá contar con los documentos de planeación del proyecto: Feature Log y con un Sprint Plan. (Morales Gerardo, 2010)

En la etapa Project Huddle es necesaria la participación de todos los integrantes del equipo: El Project Manager (Jefe de proyecto), Game Designer (Diseñador del juego) y Game Developers (Desarrolladores del videojuego); aquí se decidirá el rumbo que tomará el proyecto durante la etapa de producción, desde las características del videojuego hasta los tiempos estimados de cada sprint y del proyecto en general.

### 4.4.2.3 Fase de Producción

La segunda etapa, la más importante y la más larga es la de Producción. Ésta, se apoya en herramientas preparadas en preproducción: Sprint Backlog y Feature Log. (Morales Gerardo, 2010)

**Imagen 8. Modelo de fase Producción Huddle**



El rol de Project Manager es asegurarse que el proceso se siga al pie de la letra y de que el equipo trabaje de manera eficiente.

El Game Designer juega un papel muy importante en esta fase, ya que estará pendiente de cómo se van generando los diferentes requerimientos especificados en la etapa de preproducción y de las características nuevas que se puedan agregar durante los sprints.



Esta etapa comienza con el primer sprint entrando de lleno al desarrollo del videojuego. Una vez que se ha comenzado el sprint, se lleva a cabo el Sprint Huddle, en la cual se reúne todo el equipo y se analizan los requerimientos que se definieron anteriormente en el Feature Log, generando entonces un Sprint Backlog este es un documento que contiene las tareas a realizar, a quien serán asignadas y el tiempo que tomara su realización para poder lograr la meta del sprint. (Morales Gerardo, 2010)

Las tareas, son seleccionadas por los miembros del equipo, ellos se administran, deciden el tiempo necesario para su desarrollo. En esta parte del proceso se hace evidente el trabajo en equipo ya que será necesario que todos los integrantes obtengan y/o proporcionen la ayuda necesaria para lograr la meta del sprint.

Una vez asignadas las tareas los miembros se reúnen y discuten su progreso y/u obstáculos presentados. Al alcanzar la meta del sprint, se procede a la etapa de pruebas Alfa. Durante esta actividad, los miembros dedicados a pruebas analizan cada característica que se implementó durante el sprint, se aseguran que efectivamente la meta del sprint se haya alcanzado y que no haya errores en la codificación e integración de recursos. (Morales Gerardo, 2010)

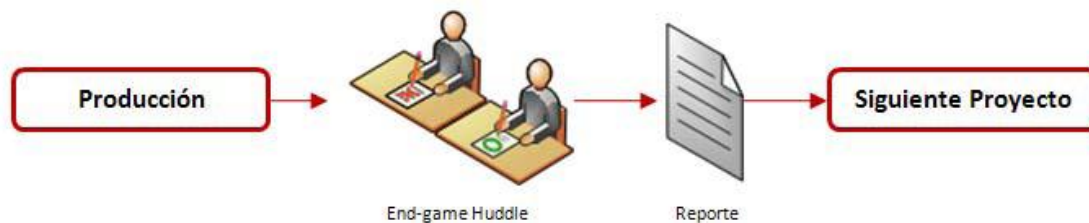
Al término de todos los sprints, se alcanza el hito de una versión beta del videojuego, el cual será probado por personas que no sean miembros del equipo de desarrolladores.

Al finalizar las pruebas Beta, se obtiene el producto final o Gold Master. Una vez obtenido, se deberá pasar a la etapa final que recibe el nombre de Postmortem. (Morales Gerardo, 2010)

#### 4.4.2.4 Fase postmortem

La etapa del postmortem consiste en generar un reporte cuyo propósito es describir a detalle las actividades específicas que fueron más efectivas para el proyecto desde el inicio del proceso hasta la entrega del producto; de igual manera, describe las actividades que llegaron a perjudicar el desarrollo junto con sugerencias para corregir dichos problemas con la finalidad de no acarrearlos al siguiente proyecto.

#### Imagen 9. Modelo de fase Postmortem



Para realizar esta fase, el equipo debe realizar la última actividad llamada End-game Huddle en la cual se analizarán los aspectos positivos y negativos del proyecto. Una manera de llevar estas reuniones es respondiendo entre todos, las siguientes preguntas:

- ✓ ¿Qué salió bien?
- ✓ ¿Qué salió mal?
- ✓ ¿Qué obstáculos se presentaron?

Del End-game Huddle saldrán sugerencias que deberán ser analizadas y filtradas con la intención de generar un reporte que incluya todas aquellas propuestas de mejora al proceso con el objetivo de que sean incorporadas en el próximo proyecto. Esta fase es de importancia debido a que resulta más sencillo comenzar a un proyecto basándose en los resultados obtenidos de otros; es de esta manera como se tiene un desarrollo más confiable y eficiente. (Morales Gerardo, 2010)

#### **4.4.2.5 Plantillas del proceso Huddle**

Una de las características de cualquier proceso de desarrollo de software es que debe de ser soportable (Sommerville, 2005) es decir, las actividades deben de poder desarrollarse en herramientas que ayuden a la aplicación del proceso. Huddle cumple con esto, proporcionando plantillas para el Documento de Diseño, Sprint Plan, Feature Log, Sprint Backlog y Reporte Postmortem .

##### **4.4.2.5.1 Documento de preproducción**

El Documento de Diseño es el artefacto más importante del proceso, se acarrea a lo largo del proyecto y deberá estar dispuesto a sufrir varios cambios desde la etapa de revisión. Contiene todas las especificaciones necesarias para comenzar el proyecto, éstas van desde el tema principal del videojuego hasta el número de niveles que tendrá.

El diseñador del juego asienta la idea a este documento con información detallada del proyecto, desde el título del juego, el género, una visión general y mecánica, aspectos de jugabilidad, modos de juego, plataforma, software que se utilizará, etc. Los videojuegos comúnmente tienen conceptos muy originales, y estandarizar este documento sería como estandarizar la creatividad, hay secciones y campos que pueden o no aplicar al proyecto, sin embargo, hay cierta información que es esencial en todos los documentos de diseño de videojuegos. (Morales Gerardo, 2010)

A lo largo del proyecto, este documento sufrirá varios cambios, es importante que la primera versión de este documento no contenga información tan específica del videojuego, no se puede saber todo desde el principio, tal vez el juego requiera más niveles o menos, la mecánica puede cambiar a algo más entretenido o se pueden agregar otros modos de juego, conocimiento y experiencia.

El Documento de Diseño es la guía del proceso Huddle, es el que define si el proyecto merece entrar a una etapa de Producción y dicta las características que se registran en el Feature Log. El documento será sometido a revisión con

escrutinio por parte de desarrolladores independientes que conformarían parte del equipo o por medio de un representante o ejecutivos de una empresa que publica videojuegos. Este paso dicta el futuro del proyecto, es probable que los ejecutivos decidan obstaculizar el proyecto, después de todo, ellos lo publicarán, puede que la idea nunca llegue a la etapa de Producción y deberá cambiarse de manera mínima, severa o incluso abandonarla. (Morales Gerardo, 2010)

En un grupo con desarrolladores independientes cambia el caso, comúnmente es parte del rol de Game Designer escoger su equipo de trabajo a través de las habilidades que el proyecto necesita, el equipo entonces opina que partes del documento y del videojuego exceden capacidades o que otras características se podrían agregar, es sin embargo el diseñador es el que tiene la última palabra.

**Tabla 9. Plantilla del Documento de Diseño**

CAMPO DESCRIPCIÓN	CONCEPTO
Título	El título del juego, debe ser un nombre que capte la atención del jugador y del lector del documento. A grandes rasgos, debe de incluir el concepto del juego. El título debe ser algo memorable.
Estudio/Diseñadores	El nombre del estudio y/o del diseñador o diseñadores del documento.
Género	El género abarca que tipo de juego será. Simulación, FPS, Rol, etc.
Plataforma	Qué hardware se requiere para jugarlo. Computadora Personal, Xbox 360, PS3, etc.
Sinopsis de Jugabilidad y Contenido	En uno o dos párrafos, describir la esencia de jugar el juego. Incluir un poco del contenido que tendrá, historia, personajes, objetivo, etc.
Mecánica	Describir la jugabilidad y el control del juego. ¿Qué hace el jugador? ¿Qué usa para lograr sus objetivos?
Tecnología	Enlistar que hardware y software se requiere para producir el juego. Desde lenguaje de programación hasta editor de sonidos.
Público	veterano.
<b>MECÁNICA DEL JUEGO</b>	
Esta sección esencialmente describe lo que el jugador puede hacer y cómo puede hacerlo. Describir las acciones del jugador, de preferencia en secuencia a cómo será en el juego.	
Cámara	Describir el tipo de cámara que se utilizará. Es decir, qué perspectiva tiene el jugador ante lo que está viendo en el juego, si es 3D o 2D, vista isométrica, en primera persona, etc
Periféricos	¿Qué periféricos utilizará el jugador para lograr los objetivos mencionados? Incluir todos los que apliquen: teclado, mouse, <i>gamepad</i> , micrófono, etc.
Controles	Describir los botones y teclas que invoquen las acciones mencionadas en la sección de Mecánica del Juego.
Puntaje	Explicar de qué manera el juego se mantiene al tanto de los logros del jugador. Incluir también si existe una tabla de puntajes que compare los mismos entre los jugadores, ya sea de manera local o en línea.
<b>NIVELES</b>	
Los juegos comúnmente se dividen en niveles o en mapas secuenciales dentro de los cuales se debe cumplir con ciertos objetivos para progresar en el juego. Existen juegos en los cuales los niveles solo cambian a razón de la dificultad y los objetivos siguen siendo los mismos, de igual manera se deben describir esos cambios en esta sección.	

Título del Nivel	El nombre del nivel.
Encuentro	Describir si es el primer nivel, un tutorial o un bonus, en otras palabras, ¿Cuándo es que el jugador llega a este nivel?
Descripción	Una descripción detallada del nivel.
Objetivos	Qué debe de hacer el jugador para terminar el nivel? Este campo también debe incluir si el jugador tiene que resolver ciertos acertijos o derrotar a cierto enemigo para progresar.
Progreso	Describir que ocurre cuando el jugador termina el nivel
Enemigos	Si el nivel tiene enemigos que el jugador debe enfrentar, éstos se enlistan en este campo, de lo contrario este campo puede ser omitido.
Items	Enlistar los objetos que el jugador o los enemigos pueden usar y que aparecen en este nivel, este campo se puede omitir si no existen dichos objetos
Música y Efectos de Sonido	Describir la música de este nivel al igual que los efectos de sonido de ambiente que contiene.
<b>PERSONAJES</b>	
Los personajes principales y secundarios que aparecerán en el juego. Esta sección se puede omitir si el juego no tiene personajes.	
Nombre del Personaje	
Descripción	Describir detalladamente el físico del personaje, si es humano o extraterrestre, su vestimenta, etc.
Concepto	Describir la conducta y comportamiento, al igual que los motivos del personaje. Mencionar también si es el enemigo principal o el protagonista. El concepto también puede relatarse como una historia del personaje, detallando en las relaciones con otros personajes del juego.
Encuentro	¿Cuándo aparece este personaje en el juego?
Habilidades	Enlistar las habilidades del personaje.
Armas	Enlistar las armas del personaje.
<b>ENEMIGOS</b>	
Los enemigos obstaculizan el progreso del jugador, pueden ser máquinas, otros personajes, monstruos, etc.	
Nombre	El nombre del enemigo.
Descripción	Describir detalladamente el físico del enemigo así como también su comportamiento.
Encuentro	¿Cuándo aparece este enemigo en el juego?
Imagen	Fotografía o dibujo conceptual del enemigo.
Habilidades	Enlistar las habilidades del enemigo.
Armas	Enlistar las armas del enemigo.
<b>GUIÓN</b>	
En esta sección se incluyen todos los diálogos del juego. Estos pueden ser muy variantes o inexistentes dependiendo de la naturaleza del juego. El guión debe de incluir encabezados, nombres, diálogo, acción y transiciones.	
<b>MÚSICA Y SONIDOS</b>	
La música y/o sonidos que se usarán en el juego, nombre, descripción junto con un número de referencia. Si es música de fondo, la referencia debe de empezar con una 'M' seguida de un número en secuencia. Si es un efecto de sonido, empezar con 'S'.	
<b>MIEMBROS DEL EQUIPO</b>	
Información de las personas que trabajarán en el proyecto, incluye su nombre, el rol o roles que desempeñan y medios por los cuales se les puede contactar	
<b>DETALLES DE PRODUCCIÓN</b>	
Antes de entrar a la etapa de Producción, se definen en el documento algunos detalles del proyecto.	
Fecha de Inicio	¿Cuándo empieza la etapa de Producción del proyecto?
Fecha de Terminación	¿Cuándo termina la etapa de Producción del proyecto?
Presupuesto	Una estimación aproximada del presupuesto del juego.

#### 4.4.2.5.2 Documentos de Producción

Huddle proporciona diferentes plantillas que se utilizan durante la fase de Producción que incluye: Feature Log, Sprint Plan, Sprint Backlog

**Tabla 10. Plantillas de los Documentos de Producción**

CAMPO	DESCRIPCIÓN
<b>SPRINT PLAN</b>	
SprintID	Identificador numérico del sprint.
Start	Fecha de inicio del sprint.
Days	Número de días que tomará el sprint.
Ends	Fecha de término del sprint.
Status	Estado en el que se encuentra la tarea Planeada / En desarrollo/ Terminada
Goal	Meta a alcanzar en el sprint.
%	Porcentaje total del sprint dentro del proyecto.
<b>FEATURE LOG</b>	
FeatureID	Identificador numérico del feature.
Status	Estado en el que se encuentra el feature, puede ser: Planeado, Sin planear, En desarrollo, Terminado, Feature Creep o Eliminado.
Days	Días que tomará desarrollar el feature.
SprintID	Número del sprint al que pertenece el feature.
Comments	Comentarios sobre el feature.
<b>SPRINT BACKLOG</b>	
Sprint # Backlog	Título que incluye el número de sprint del cual se está creando el backlog.
# FeatureID	Identificador numérico del feature al que pertenece la tarea.
Number of tasks	Número de tareas que contiene el Sprint Backlog.
Task	Nombre que recibe la tarea.
Member	Miembro del equipo designado para realizar dicha tarea.
Status	Estado en el que se encuentra la tarea, puede ser: Planeado, En desarrollo o Terminado.

#### 4.4.2.5.3 Documento de la fase Postmortem

El propósito del Reporte Postmortem es describir a detalle las actividades específicas que fueron más efectivas para el proyecto desde el inicio del proceso hasta la entrega del producto; de igual manera describe las actividades que llegaron a perjudicar el desarrollo junto con sugerencias para corregir dichos problemas para no acarrearlos al siguiente proyecto.

La meta del documento es entonces informar a los desarrolladores de proyectos futuros sobre los obstáculos a los que se enfrentó el equipo durante este proyecto. (Morales Gerardo, 2010)

**Tabla 11. Plantilla del Reporte Postmortem**

CAMPO	DESCRIPCIÓN
<b>ANTECEDENTE</b>	
Título	El título del juego especificado en el Documento de Diseño.
Estudio/Diseñadores	El nombre del estudio y/o del diseñador o diseñadores del videojuego.
Género	El género del juego.
Plataforma	El hardware que se requiere para jugarlo.
Antecedente	Una descripción breve pero detallada del proyecto que incluya los aspectos más importantes y característicos del videojuego.
<b>MIEMBROS DEL EQUIPO</b>	
Información de las personas que trabajaron en el proyecto, incluyendo su nombre, el rol o roles que desempeñaron y los medios para contactarlos.	
<b>EFECTO POSITIVO</b>	
Es una lista de aquellas actividades que generaron un efecto positivo durante el proceso de desarrollo.	
Categoría	La actividad que se llevó a cabo.
Declaraciones	Descripción de cómo es que dicha actividad generó un efecto positivo.
¿Cómo continuar así?	En este campo se enlistan por referencia las actividades adjuntando sugerencias de cómo seguir generando un efecto positivo.
<b>EFECTO NEGATIVO</b>	
Es una lista de aquellas actividades que generaron un efecto negativo durante el proceso de desarrollo.	
Categoría	La actividad que se llevó a cabo.
Declaraciones	Descripción de cómo es que dicha actividad generó un efecto negativo.
¿Cómo corregir?	En este campo se enlistan por referencia las actividades adjuntando sugerencias de cómo corregir estos problemas.
<b>CONCLUSIÓN</b>	
Esta sección sintetiza lo positivo y lo negativo del proyecto, a su vez, menciona las lecciones que se aprendieron a lo largo del desarrollo del mismo.	

## **V. Diseño metodológico**

### **5.1 Tipo de estudio**

Este proyecto está orientado a una investigación exploratoria, cuyo objetivo es examinar un tema poco conocido.

El objetivo de la investigación posee características muy propias, que no se encuentran en la misma proporción en comparación con otras temáticas relacionadas con las aplicaciones de software.

Se tomó base para la realización de este proyecto las herramientas, conocimientos teóricos y roles para el desarrollo de un videojuego propuestos en el proceso para el desarrollo de videojuegos Huddle.

### **5.2 Estudio de factibilidad**

Al iniciar el desarrollo de este proyecto este se enfocara en el aspecto técnico (equipo a utilizar) y económico (gastos de equipo).

#### **5.2.1 Factibilidad técnica**

El equipo para desarrollo del proyecto debe cumplir con las siguientes características:

#### **Recursos de Hardware**

Procesador Intel core i5 3.2 GHz

Memoria RAM 4 GB DDR3

Tarjeta Madre Asrock

Disco Duro 500 GB

Tarjeta de Video 3 GB Geforce 8400 DDR3

Monitor 17" entrada HDMI



### Recursos de Software

Sistema operativo Microsoft Windows 7  
Unreal Developmet Kitment (UDK)  
Autodesk 3DS Max Studio 2012

### 5.2.2 Factibilidad Económica

Costos de los recursos

#### Software

Sistema operativo Microsoft Windows 7	\$ 60
Unreal Developmet Kitment (UDK)	Sin Costo
Autodesk 3DS Max Studio 2012	Versión Gratuita

#### Hardware

Procesador Intel core i5 3.2 GHz	\$ 253
Memoria RAM 4 GB DDR3	\$ 44
Tarjeta Madre Asrock	\$ 55
Disco Duro 500 GB	\$ 67
Tarjeta de Video 3 GB Geforce 8400 DDR3	\$ 41
Monitor 17" entrada HDMI	\$ 100
<b>Total</b>	<b>\$620</b>

### 5.3 Proceso metodológico

Basados en la alta demanda del género de Disparos en tercera persona (third person Shooter) mostrada en los estudios de mercados de la industria de los videojuegos y en la gran variedad de características que posee dicho género optamos por desarrollar un guión cuya historia y mecánicas de juego se implementan perfectamente utilizando este género.

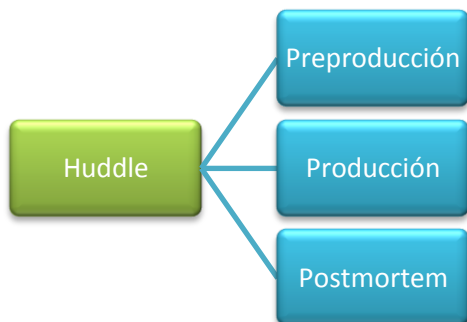
Pese a la gran variedad de procesos que podrían ser susceptibles de adecuarse para el desarrollo de videojuegos, se optó por utilizar un proceso caracterizado desde el principio a este tipo de software. Seleccionamos el Proceso Huddle desarrollado y diseñado por una comunidad de informáticos de México, está basado en dos procesos de desarrollo de software: Proceso de desarrollo en cascada y Scrum Framework.

Huddle es un proceso específico para el desarrollo de videojuegos con las siguientes características:

- ✓ Ágil
- ✓ Óptimo para equipos multidisciplinarios de 3 a 10 personas
- ✓ Iterativo
- ✓ Incremental
- ✓ Evolutivo

Huddle se divide en las siguientes etapas:

**Imagen 10. Etapas del proceso metodológico utilizado**



### **5.3.1 Fase de preproducción del videojuego**

Como primer paso en la etapa de preproducción se creó el primer boceto de guión del videojuego cuya idea inicial fue la historia de un enfrentamiento futurístico entre una potencia mundial y una unión de países; donde la causa del enfrentamiento es el ansia de poder y conquista de parte de la potencia en contra de los países más pequeños.

Este guión en cada uno de los sprints en la etapa de preproducción fue sometido a diversas modificaciones basadas en las sugerencias y aportes de todos los miembros del grupo, dando como resultado final un guión con un argumento bien definido y satisfactorio, que fue aprobado por todos los integrantes del equipo de trabajo.

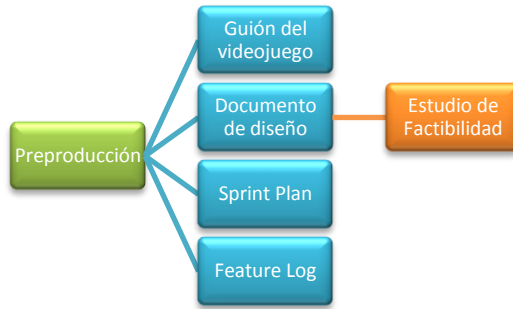
Se analizaron diferentes motores para creación de videojuegos entre los cuales sobresalieron por presentar las características más apegadas al género seleccionado: Motor Unity y Motor UDK (Unreal Engine Development Kit)

Finalmente el motor seleccionado fue UDK, producido por la compañía de juegos Epic Games. Esta decisión está basada en el hecho de que UDK posee una licencia gratuita de uso académico y los recursos que ofrece se adaptan al guión del video juego el cual está basado en combate de armas y UDK es indicado para crear juegos a fines. **(Ver Resultado 1 y Resultado 2)**

#### **Project huddle**

Una vez que se aprobó por todo el grupo el documento de diseño del video juego, se pasó a la fase Project huddle o planeación general del proyecto, en el cual se elaboró el sprint plan y el Feature log que se muestran a continuación estos contienen el cronograma de reuniones y las tareas que se ejecutarán en la fase de producción. **(Ver Resultado 3 y Resultado 4)**

**Imagen 11. Resultados en la fase de preproducción**



### 5.3.2 Fase de Producción del videojuego

En esta etapa se llevaron a cabo los trece sprints definidos en preproducción, cada uno inicio con una reunión o Sprint Huddle donde se analizó la meta correspondiente al sprint, se designaron las tareas definidas en el feature log dividiéndolas entre los tres miembros del equipo iniciando el desarrollo del videojuego y dejando esta etapa documentada en el sprint backlog. **(Ver Resultado 5)**

Al terminar todos los sprints se obtuvo la versión beta del videojuego la cual fue sometida a las pruebas beta estas consistieron en invitar personas externas al equipo de desarrollo para que probaran el videojuego, como resultado no hubieron errores de ejecución. Al finalizar las pruebas beta se obtuvo el producto final del video juego y la conclusión de la etapa de producción.

**Imagen 12. Resultados en preproducción**



### 5.3.3 Fase Postmortem

En esta etapa se llevó a cabo la última actividad llamada End game huddle, consistió en una reunión donde todo el equipo de trabajo expuso que actividades salieron bien, que actividades salieron mal y que obstáculos se presentaron, quedando documentada en el reporte mostrado postmortem. **(Ver Resultado 21)**

#### Imagen 13. Resultado de la fase Postmortem



## 5.4 Desarrollo del Videojuego

### 5.4.1 Diseño de niveles

Siguiendo la estructura narrativa establecida el videojuego se desarrolla en una base militar, se establecieron 3 niveles en el videojuego, con una presentación en forma de mapa de secuencia, con la ruta que debe seguir el jugador y marcas visuales para la localización de elementos claves teniendo como contexto en el primer nivel con una cárcel en forma laberinto llena de obstáculos. **(Ver Resultado 6)**

El segundo piso es el segundo nivel del videojuego es un laboratorio que contiene equipo para experimentos, material radiactivo y aloja todo el armamento tecnológico de la base militar. **(Ver Resultado 7)**

El tercer nivel es el campo abierto de la base militar, rodeado de aviones militares, vehículos de guerra y un campo de entrenamiento. **(Ver Resultado 8)**

### 5.4.2 Creación de materiales

Una vez diseñado los, se les asignó un material que adopta las propiedades de color, reflexión de luz, emisión de colores como fuente lumínica Para esto, el UDK

utiliza el Editor de Materiales (Material Editor), haciendo uso de archivos de imagen TGA a los diferentes atributos del material.

El material puede ser escalado y rotado en cualquier dirección, lo que brinda un gran control en la forma como se pueden graficar las texturas sobre una superficie. **(Ver Resultado 9)**

### **5.4.3 Personaje y enemigos**

El personaje utilizado que corresponde al protagonista del juego, posee patrones de movimiento físicos estándar proporcionados por el motor, la apariencia está compuesta por un static mesh personalizado en el content Browser, es una estructura de robot humanoide siendo congruente con el argumento. **(Ver Resultado 10)**

Los enemigos utilizados pertenecen al grupo de recursos spawn proporcionados por el motor, los cuales poseen una inteligencia artificial básica la cual consiste en reconocer su objetivo y atacar

### **5.4.4 Programación de eventos**

Para la creación de eventos se utilizó la herramienta principal de desarrollo de UDK UnrealKismet. En este se pueden visualizar actores, elementos y comportamientos para programar secuencias y lograr los eventos deseados.

Dicha herramienta permite modificar el flujo de ejecución del mapa. La interfaz que se nos presenta es una pantalla vacía en la cual se podrán ir añadiendo distintos elementos para realizar modificaciones en el flujo del nivel.

#### **5.4.4.1 Programación de cambio de personaje a tercera persona**

Esta se realizó a través del evento playerSpawned (Generar jugador) el personaje principal es lanzado al cargar el nivel, seguidamente este evento es conectado a la función ChangePlayerMesh (Cambiar jugador) la cual se encarga de cambiar el personaje que el motor trae por defecto por nuestro propio personaje con sus propios movimientos, seguidamente es conectado a la función Console Command

(consola de comando) que se conectada a la entrada Target (objetivo) esta se encarga de cambiar el personaje a perspectiva de tercera persona. **(Ver Resultado 11)**

#### **5.4.4.2 Creación de enemigos y sus eventos**

Para la creación de enemigos utilizamos la función Actor Factory mediante esta función agregamos un enemigo, configuramos sus características, como tipo de enemigos utilizamos personajes bípedos, como tipo de armas rayos laser y cañón de disparos. Esta función se activa al cargar el nivel mediante el evento Level Load. **(Ver Resultado 12)**

#### **5.4.4.3 Programación de puerta con llave**

En esta sección el jugador al pasar por un punto definido en el juego en donde se ubicó un evento Triger Tocuh (interruptor) activa la función compare bool (comparar verdad) la cual devuelve un valor falso si el jugador no ha tomado la llave o verdadero si el jugador tiene la llave.

Si el valor es falso se coloca un mensaje a través del evento playerannouncement (Anuncio) el cual se encarga de enviar un mensaje indicando que necesita obtener una llave para poder abrir la puerta.

El jugador obtiene la llave ubicada en un punto específico mediante otro evento trigger touch, una vez que el jugador toma la llave esta desaparece, para lograr este efecto se utilizó el evento destroy (destruir) que se encarga de destruir el punto físico de la llave, luego se pasa al evento bool que asigna valor verdadero, luego es enviado nuevamente al evento compare bool al detectar que el valor de la función es verdadero lo envía al evento matinee encargado de abrir la puerta para dar pase de entrada al jugador. **(Ver Resultado 13)**

#### **5.4.4.4 Programación del evento ascensor**

La construcción del ascensor está compuesta por tres fases, la primera consiste en una función Trigger Touch que al ser activada por el jugador, esta carga un evento Matinee el cual se encarga de generar una animación que abre la puerta del ascensor. La segunda fase se ejecuta cuando el personaje sube al ascensor se

activa un segundo trigger touch que carga un evento matiné el cual está encargado de subir el ascensor, la tercera fase se da cuando este evento termina y se inicia un nuevo matiné que abre la puerta cuando el ascensor llega a sus destino. **(Ver Resultado 14)**

#### **5.4.4.5 Animaciones**

En esta sección se utilizó el evento Level Loaded (Cargar) el cual es una de los principales del UnrealKismet, se utilizó para generar dos animaciones una para hacer despegar un avión y la otra para activar un lanza misiles, se usó el evento matiné, para generar una trayectoria de arrastre hacia donde queremos desplazar la animación. **(Ver Resultado 15)**

### **5.5 Diagramas del Lenguaje unificado de modelado**

#### **5.5.1 Diagramas de casos de usos**

##### **5.5.1.1 Diagrama de caso de uso jugador ejecuta el videojuego**

Este diagrama representa las etapas que contiene el proceso de interacción entre la interfaz de usuario y el jugador en el cual, este interactúa con las diferentes opciones entre las cuales están iniciar partida, recorrer el nivel, interactuar con los objetos del escenario, así como también salir de la aplicación. **(Ver Resultado 16)**

#### **5.5.2 Diagramas de secuencia**

##### **5.5.2.1 Diagrama de secuencia jugador ejecuta el videojuego**

Este diagrama representa las secuencias de interacción entre la interfaz de usuario y el jugador, entre las cuales están: iniciar, pausar, reanudar y salir de la partida, así como también de la aplicación. **(Ver Resultado 17)**

##### **5.5.2.2 Diagrama de secuencia jugador enfrenta enemigo**

Este diagrama representa la secuencia de interacción entre el jugador y los enemigos durante un combate, en el cual se puede presentar los distintos



escenarios, los cuales están en dependencia de las armas que tenga cada uno.  
**(Ver Resultado 18)**

### **5.5.2.3 Diagrama de secuencia de la interacción entre jugador y puerta con llave**

Este diagrama representa las secuencias en el proceso de interacción entre el jugador y las puertas con llave, en las cuales interactúa el trigger el cual es el elemento detonante de todos los eventos de la secuencia. **(Ver Resultado 19)**

### **5.5.3 Diagrama de clases del videojuego**

Este diagrama representa las diferentes clases contenidas en el videojuego, en él se detallará las clases, variables y los métodos utilizados **(Ver Resultado 20)**

## **VI. Resultados**

### **Resultado 1: Guión del videojuego**

#### **Soul of Wars (Alma de guerras)**

Año 3023

El imperio norte americano, luego de la escases de petróleo y recursos, en medio de la desesperación, opta por invadir los países aledaños tomando ventajas de su avanzado desarrollo tecnológico y militar, teniendo como principal objetivo los recursos de Centro y Latinoamérica.

El objetivo del imperio del norte al invadir el territorio del sur, además de expandir sus dominios y aumentar sus recursos económicos y bélicos, es también hacerse poseedor de un raro recurso natural descubierto muy recientemente, solo encontrado en territorio Latinoamericano, llamado Mineral X. Este mineral combinado con tecnología de fusión nuclear podría crear una fuente de energía de proporciones atómicas nunca antes poseída.

La invasión inicio por México, la reacción del resto de países latinoamericanos temiendo una pronta invasión y viéndose en desventaja en el caso de contraatacar individualmente, deciden establecer una alianza con el objetivo de unir fuerzas y hacerle frente a la invasión, funcionando todos como uno, aportando todos sus recursos para desarrollar una tecnología militar que les permita defenderse del avanzado armamento norteamericano. Esta Unión se le denomino Unión Latinoamericana.

Mientras se constituye esta unión, la invasión del imperio continua avanzando en su ataque contra el territorio de Latinoamérica.

La unión Latinoamericana, trata de desarrollar sus tecnologías en secreto, teniendo como objetivo, que con pocos recursos puedan hacer frente a la masiva cantidad de unidades militares que posee el imperio del norte, sabiendo que no cuentan con igual ventaja numérica que el enemigo, tienden a enfocarse más en el poder del arma que en la cantidad de unidades que desarrollaran.

Después de mucho tiempo de experimentación e implementación de modelos fallidos, logran crear un arma biónica, un androide de guerra equipado de potentes armas, las cuales son utilizadas en el campo de batalla para hacer frente al armamento bélico del imperio del norte.

Pero incluso, estos soldados biónicos no logran ser lo suficientemente potentes para poder superar la ventaja numérica y potencial del imperio del norte, dejando a la Unión Latinoamericana en una posición vulnerable ya que todas sus posibles defensas hasta ahora habían sido abatidas, dejando un gran saldo de soldados tanto humanos como maquinas abatidos en el campo de batalla quedando expuestos a una invasión total y a su conquista inminente.

Estas máquinas tenían un problema, carecían de lógica y del espíritu de lucha que caracteriza a la raza humana, por lo que no tuvieron un buen desempeño en el campo de batalla, derivando estos hechos en la perdida de los países de la Unión Latinoamericana desde México hasta Guatemala.

Ante esto la Unión latinoamericana empieza en secreto el proyecto de tecnología de guerra más complejo y grande que jamás se haya creado, utilizando como uno de los recursos principales el precioso mineral X sabiendo que en su territorio cuentan con los últimos yacimientos de este mineral, iniciando así el desarrollo de un androide que posea una fuente de energía ilimitada, utilizando fusión nuclear , y las propiedades del mineral X que permiten controlar y canalizar la potencia de esta energía atómica.

Sin embargo, hasta el momento semejante poder no había podido ser controlado por ningún tipo de computadora o inteligencia artificial, dando como resultado un androide inestable que terminaba autodestruyéndose y por lo tanto inutilizable en la guerra.

Debido a estos inconvenientes y a la prisa por sobrevivir, la unión latinoamericana decide llevar a cabo el desarrollo de esta arma a un nivel que nunca imaginaron alcanzar.

Con el fin de hacer frente a la fuerza del imperio del norte, debían crear el arma perfecta, ya poseían un androide con la potencia requerida pero aún no habían logrado tener el control sobre ella.

El proyecto para perfeccionar el arma fue bautizado bajo el nombre de “Digitalización Mental” cuyo objetivo era mejorar las habilidades de lógica y perseverancia. Para poder controlar a la maquina más potente diseñada por el hombre, debían de utilizar la maquina más compleja....La mente humana.

El proyecto consistía en la implantación de una mente humana en el androide, solo la mente humana, la cual es más compleja que cualquier máquina, podría controlar la potencia del androide dentro de su cuerpo y canalizarla, evitando que la energía lo domine y se autodestruya.

El proyecto de digitalización mental radicaba en tomar los impulsos eléctricos de las neuronas del cerebro, en las cuales se encuentran la memoria, la personalidad, las emociones y pensamientos y transferir estos impulsos eléctricos a un cerebro digital en forma de tarjeta electrónica en cuyo centro se encuentra un cristal del mineral X y que es colocada en el androide, migrando toda la esencia de la mente humana a una máquina.

Sin embargo, hasta ese momento los intentos de digitalizar una mente humana habían sido fallidos, debido a los sentimientos de miedo a la muerte que habían experimentado los voluntarios a la hora de llevar a cabo el proceso, no sobreviviendo a este.

Mientras sucedían estos hechos un joven llamado Khris Fenix de origen sudamericano, que siempre estuvo en contra de la opresión, habiendo sido inculcado por su padre y su abuelo en los ideales Revolucionarios del pasado, como la libertad y la independencia. Al ser Khris testigo de la invasión del imperio inmediatamente se enlista en el ejército de la Unión Latinoamericana, llevando a cabo el entrenamiento y alistándose para ir a combatir al campo de batalla.

Pero en los últimos días de su entrenamiento sus fuerzas vitales comenzaron a decaer y su cuerpo a debilitarse de manera inexplicable, después de muchos exámenes médicos le es detectado un tipo muy raro de cáncer y se encuentra etapa terminal, no quedándole mucho tiempo de vida, debido a esto queda fuera de lucha al lado del ejército Latinoamericano.

Luego de estos hechos Khris se sumerge en total depresión viendo como sus amigos luchan por la libertad dando la vida en batalla, mientras él no tiene el cuerpo apto ni para caminar, su desesperación crece cuando ve que las fuerzas de la Unión son dominadas casi en su totalidad, siendo obligadas a retroceder, perdiendo grandes territorios.

Khris lloraba amargamente todos los días ante el sentimiento de impotencia que experimentaba por su enfermedad, esto sucedió hasta que uno de los científicos desarrolladores del proyecto de digitalización mental lo observa y le hace las siguientes preguntas:

- ¿El deseo que sientes dentro de tu alma de luchar por la libertad de tu gente, trasciende más allá de los límites físicos de tu cuerpo?
- ¿Tu deseo es lo suficientemente fuerte como para trascender más allá de tu muerte inminente?
- ¿De qué serías capaz con tal de tener una oportunidad de poder llevar a tu gente a la libertad?

Las preguntas que el científico hizo al joven lo inundaron de un espíritu de lucha que lo llenó como fuego por dentro, este poniéndose de pie a duras penas, exclama con todas sus fuerzas un fuerte grito con lágrimas en sus ojos: “ SI LO HARIAAAAA!!!!!!!!!! “

Al ver el científico que en este joven, su mente era mucho más fuerte que su débil cuerpo, le propone la idea de ofrecerse como voluntario para el proyecto, a lo que él responde afirmativamente, sin pensarlo dos veces.

Es entonces cuando Khris es sometido al proceso de digitalización, abandonando todo miedo a la muerte y apegado únicamente al deseo de liberar a su gente.

El proceso de digitalización mental, dio inicio.

La mente humana, recuerdos, personalidad, valores y sentimientos, en fin toda la esencia de un ser humano, estaba siendo trasladada en forma de impulsos eléctricos, a un cerebro electrónico; Proceso al cual hasta ahora nadie ha sobrevivido, ya que en experimentos anteriores, los impulsos eléctricos causados por el miedo a la muerte, habían saboteado el proceso, matando a los voluntarios.

El proceso parecía estar llevándose, en los parámetros esperados, y figuraba ser todo un éxito, pero de pronto....

La invasión había llegado hasta ellos, debido a que cierta información del proyecto, había sido filtrada al imperio, siendo finalmente atacados en ese preciso momento.

Todas las personas, que se encontraban en el lugar fueron asesinadas, toda la maquinaria fue destruida, y el proceso interrumpido abruptamente, no completándose. Los dos científicos fueron secuestrados, para su interrogatorio y tortura.

El cuerpo de Khris se quedó sin mente, inerte y vacío. Sin embargo, su mente, no pudo ser aparentemente digitalizada al cien por ciento al interrumpirse el proceso, y como resultado el androide no mostraba signos de vida. El proceso había fracasado una vez más.

Los soldados invasores, robaron el prototipo del androide, para estudiarlo, y reproducirlo en masa.

El prototipo fue llevado a la base militar ubicada en los límites de los territorios hasta ahora conquistados por el imperio, la zona que alguna vez correspondió al país de honduras. Acá fue, internado en una prisión de máxima seguridad, donde se iba a ser estudiarlo.

Pero algo, paso....

El proceso de digitalización, no fue un fracaso del todo.

Ya que estando en la prisión, el último suspiro de vida de Khris, se conservaba aun, debido a que se rehusaba a morir, y esta fuerza de voluntad provoco un estallido de energía, luego el cuerpo del androide recobro su vida, resultando el proceso de digitalización que había sido aparentemente un rotundo fracaso, todo un éxito.

Fue así como se dio el origen de un ser biónico, capaz de pensar con toda la lógica e inteligencia de un ser humano, pero a su vez con toda la potencia y velocidad de una máquina, permitiéndole también un multiprocesamiento de miles de ideas al mismo tiempo, originando un ser supremamente inteligente.

De igual manera al ser un androide con energía nuclear ilimitada, este nuevo ser podía correr a altas velocidades y trabajar por una cantidad de horas ilimitadas, dando origen al soldado y al ser perfecto.

Khris, despierta en su nuevo cuerpo, y se da cuenta que ya no está en el mismo lugar, que es prisionero y rápidamente deduce que lo están estudiando, Procede de inmediato a buscar su libertad, eliminando todo aparato en el laboratorio que pudiera contener algún registro de él.

Mientras se adapta a su nuevo cuerpo sus energías y poder no son comprendidos por el mismo y no los puede explotar a su máxima expresión.

A cómo puede intenta abrirse paso a través de la prisión, enfrentándose a soldados, androides y mutaciones biológicas que el imperio tiene prisioneros como sujetos de pruebas.

Explora cada uno de los pisos de la fortaleza en busca de la salida, y es atacado por todas las defensas, enfrentándose a algunos oponentes bastantes fuertes en comparación al dominio que tenía de su poder en ese momento.

Después de explorar y luchar intensamente, no consigue encontrar a los científicos que fueron tomados como rehenes. Cuando estaba muy cerca de la salida se encuentra frente a frente con el guardián superior de la fortaleza, una verdadera máquina destructiva, con una magnitud de poder desastrosa y un tamaño descomunal, quedando encerrado en una lucha uno a uno con este enemigo.

La lucha se lleva a cabo siendo la primera donde Khris se siente verdaderamente presionado, siendo su oponente muy difícil de superar. Tras un intenso esfuerzo después de insistir constantemente logra derrotarlo, descubriendo una vía de escape de la prisión. Ahí encuentra un vehículo reservado para emergencias y lo aborda, siguiendo un camino directo al exterior.

Sin embargo esta libertad no le iba a salir fácil, ya que es inmediatamente perseguido por las fuerzas guardianas de la prisión, que abordo de naves, le disparan durante todo el camino de huida; después de luchar y luchar para esquivar sus ataques, logra escapar de sus perseguidores liberándose al fin.

Inmediatamente se dirige a la zona donde fue capturado, temiendo que las fuerzas invasoras hayan avanzado y conquistado el país completo. Al llegar efectivamente descubre que las fuerzas de la Unión han caído en batalla y que la mayoría de combatientes han sido ejecutados.

Arribando en el momento preciso en que las fuerzas del imperio ya estaban dando por tomada toda el área correspondiente al territorio de Nicaragua, sin demora se lanza el solo contra el ejército completo, abriéndose paso a través de los soldados y robots, adentrándose en una encarnada lucha, contra los más fuertes.



Enfrentando todo tipo de ataques como Bombas muy potentes, rayos, tanques de guerra con poder masivo, naves, robots, logra finalmente avanzar al corazón de la trinchera enemiga, después de batallar solo por largas horas.

El corazón de la trinchera era una torre, en la frontera de Honduras-Nicaragua, donde estaba siendo descargada una gran cantidad de robots y armamento por una inmensa nave nodriza en la punta de la torre. Al llegar Khris a la cima de la torre, la nave nodriza toma posición de batalla contra él, elevándose por encima de la torre, demostrando su imponente tamaño.

La nave Nodriza, empieza un ataque con armas de inmensa magnitud destructiva, que Khris apenas logra esquivar y de recibirlas lo dañarían severamente.

Khris Dispara con todo su armamento a la coraza de la nave, pero no logra hacerle daño notorio, intentando una y otra vez pero sin ningún efecto. La nave derriba a Khris desde la punta de la torre hacia las afueras de esta, la nave lanza un implacable ataque lleno de potencia, destruyendo todas las trincheras que se encontraban en el lugar, no dejándole un sitio seguro para protegerse, sin embargo él logra volver a la torre y subir a la cima, de esta a duras penas esquivando los destructivos ataques.

Pero esta vez Khris nota un detalle particular en la parte inferior de la nave, cierto punto débil en el mecanismo de apertura de la compuerta, al que dispara con toda su potencia abriéndola, a la cual salta para introducirse dentro de ella.

Una vez dentro acaba con toda la tripulación, pero antes tiene que enfrentar una dura batalla con un robot de gran potencia llamado destroyer, que le da muy dura lucha y al final logra derrotarlo, acabando con los pilotos de la nave y tomando el completo control de esta.

Utilizando la nave destruye a todo enemigo en ese territorio de guerra, siendo la primera vez que la Unión Latinoamericana logra su primer victoria contra el imperio deteniendo el avance de las tropas conquistadoras.

Pero la misión no estaba completa ya que Khris necesitaba avanzar hacia el corazón del imperio establecido en el norte, donde debía poner fin a esta nación demente y traer finalmente la paz al mundo.

Por lo que se dirige hacia al norte, utilizando la nave nodriza que tomó, es interceptado en el camino por un sin número de enemigos aéreos, a los cuales derrota haciendo uso de las habilidades combinadas de la nave y las suyas.

Hasta que de pronto es atacado por un robot de habilidades superiores llamado Air Slayer, el cual posee la capacidad de volar por su propia cuenta a velocidad supersónica, y rápidamente descubre el mismo punto débil de la puerta, a la cual ataca y logra abrir también.

Entrando en la nave tomando sorpresivamente a Khris por el cuello y sacándolo junto con el de un gran impulso por la ventana, dejando a la nave sin control la cual se derriba.

El Air Slayer lo eleva a una gran altura, mientras Khris intenta moverse para atacar su enemigo lo neutraliza atándolo con unas fajas de acero que envuelven todo el cuerpo de Khris.

Es ahí que el instinto de supervivencia que diferencia a Khris de una máquina, entra en acción ya que descubre una nueva habilidad donde puede canalizar su energía no solo a través de su arma, sino a través de todo su cuerpo quemando completamente al air slayer.

Luego de esto Khris se encuentra en caída libre junto con el cuerpo quemado de air slayer, sin embargo logra ver que las partes de propulsión siguen intactas, y las incorpora en su propio cuerpo ganando así la habilidad para volar de manera supersónica.

Inmediatamente se dirige a la capital del imperio, recorriendo miles de kilómetros en pocos minutos.

Al llegar a la capital del imperio, desciende para no ser detectado por los radares y poder acercarse sin ser detectado.

Entrando por un desagüe a las afueras de la ciudad, se abre paso por el alcantarillado para entrar desapercibido. Sin embargo repentinamente siente un acecho, una criatura parecida a un reptil de enormes patas y brazos largos como de unos tres metros de largo lo acecha, no dejándose ver hasta el ataque. Ahí se libra una dura batalla, donde Khris enfrenta a un enemigo que lo supera enormemente en velocidad.

Una vez habiendo derrotado a su enemigo, Khris se da cuenta que todo ha sido una trampa y que todo el ducto está dinamitado con el objetivo de tenderle una emboscada, así que corre a alta velocidad antes que el lugar sea detonado buscando la salida más, al salir se da cuenta que está en el centro de la ciudad, justamente enfrente del castillo sede principal del imperio.

Ahí le esperaba la armada más grande y temida del imperio, la responsable de haber conquistado la mitad de América, ahí mismo, en todo su esplendor, en contra de un solo soldado.

Pero Khris no era un soldado normal, era alguien que confiaba en sí mismo, que jamás se daba por vencido, alguien comprometido con la libertad, sin dar un paso atrás se enfrenta valientemente a la armada completa, dando inicio a la batalla más encarnizada de la historia.

El castillo se encontraba protegido con un campo de fuerza extremadamente resistente, debido a esto las detonaciones del enfrentamiento no le afectaban.

Mutaciones, tanques de guerra, naves de ataque, naves Nodrizas, Destroyers, Air Slayer, Robots de combate, etc. Atacan masivamente a Khris, pero este no cede, se enfrenta a todos y los va destruyendo uno a uno, nave tras nave va siendo derribada, robot tras robot siendo destruido, tanque tras tanque siendo detonado. A pesar de sentir que su cuerpo está ya deteriorado por las batallas, la fuerza del corazón de Khris lo impulsa a seguir peleando más y más sin importar el número.

La lucha se extendió durante tres días completos, un solo soldado contra todo lo inimaginable, sin embargo este prevaleció y al amanecer del cuarto día, era el único en pie en el campo de batalla.

De inmediato se dirige al castillo, donde misteriosamente parece recibir una invitación a entrar, ya que el escudo de energía que lo protegía se desactivo y las puertas se abrieron.

Con mucho valor se adentra sintiéndose ya muy cerca de cumplir con su misión de frenar por siempre al imperio.

Hasta que llega a la sala del trono. Ahí sale a su encuentro nada más y nada menos que el monarca absoluto del imperio, el hombre responsable de tanta destrucción y muerte, el emperador Charles.

Un Hombre muy alto, robusto de cabellera blanca, con una fuerte presencia, el cual con una mirada asesina observa fijamente a Khris y con una risa demencial exclama:

-¡Cuanto poder! ¡Cuánta Perfección!, con esta arma, no solo Latinoamérica, también el mundo entero será dominado por el sagrado imperio.

-Es hora, de la prueba final.

Khris lo observa confundido y se pregunta a que prueba podría estarse refiriendo, ya que toda su armada había sido completamente destruida en la guerra de los tres días.

De pronto del suelo emerge una capsula metálica, la cual rápidamente se abre y deja ver que dentro de ella se encuentra un androide de color negro, que para sorpresa de Khris es una réplica exacta de sí mismo, pero con la implantación de una mente sicópata y destructiva.

El enfrentamiento comienza y las fuerzas de ambos son bastante igualadas, cada golpe, cada arma, cada manera de moverse, es nivelada por ambos.

Entonces Charles exclama:

-¡Creíste tu cautiverio, fue en vano!

-¡Creíste que toda la información que extrajimos de ti había sido destruida!  
Jajajajaja!

-¡Pobre diablo miserable! ¡Caerás ante la máxima expresión de Poder!

La lucha entre Khris y su réplica prosigue hasta que se resume a un impacto de energía de sus más potentes armas. Sin embargo, al llegar a este punto donde ya lo habían dado todo solo algo podría prevalecer, la fuerza de voluntad, esto fue lo que permitió a Khris duplicar su flujo de energía nuclear y emanando poder de todo su cuerpo logra un ataque definitivo con el cual acaba con su doble.

Durante la pelea Charles había desaparecido, terminando esta Khris sube a la Azotea más alta del castillo buscándolo.

De pronto escucha una voz Estrepitosa que viene de todas las direcciones diciendo:

-¡Tu arrogancia, ha llegado muy lejos!

-¡La arrogancia del hombre de Oponerse a los dioses!

Luego Khris mira hacia atrás y ve a un ser resplandeciente y rebosante de poder, una especie de androide muy superior de gran altura, que bajaba del cielo.

Su poder era tan grande que todos los metales en la zona se doblaron y todos los cristales se rompieron.

Tanto era su poder que no volaba por propulsión, sino por la presión ejercida por el poder que emanaba. Y le dice a Khris:

-¿Confundido? Jajajaja esta es mi verdadera forma, el verdadero y único ser perfecto.

Resulta que después de haber logrado construir una réplica de Khris e implantarle una mente perversa, Charles no se detuvo, sino que mejoro el cuerpo androide usando el metal más indestructible de la tierra. Este cuerpo mejoraba el efecto de amplificación y control del poder de la fusión nuclear, dándole un poder destructivo muy superior.

Empeorando la situación la mente que ocupaba este cuerpo fue nada más y nada menos que la mente del Mismísimo conquistador supremo, el emperador Charles.

El emperador no se había arriesgado antes a migrar su mente al prototipo anterior por miedo a la muerte, así que lo hizo hasta que este defecto fuera eliminado en el prototipo que estaba desarrollando. Durante la batalla de Khris con el prototipo anterior, Charles tomo finalmente la decisión de dar inicio al proceso de digitalización del mismo.

El poder del nuevo modelo era exageradamente destructivo y su resistencia parecía infinita. Khris ataco con todas sus fuerzas pero parecía no afectarle en lo absoluto y cada ataque de Charles devastaba terriblemente el cuerpo de Khris.

Siendo Khris brutalmente masacrado y destruido en un ochenta por ciento no se daba por vencido, aún mantenía la confianza de que podría hacer algo aunque le costase la vida.

Así que recuerda que el origen de su poder es la fusión nuclear, la misma que puede dar origen a la explosión atómica y que la función de su cuerpo androide es tomar un control de esta potencia, nivelarla y usarla como fuente de energía y ataque.

Así que corre hacia Charles lo agarra con ambos brazos con toda la fuerza que tiene y destruye las barreras de su propio cuerpo que regulaban la potencia nuclear, desencadenando a consecuencia una gran explosión atómica.

En ese momento Charles intenta usar su gran poder para liberarse pero, la potencia de la explosión atómica no le permitió reaccionar, así que la única posibilidad que le queda Charles es liberar también toda su energía atómica, pero

al hacer esto se arriesga a ser destruido por su propia explosión, así que por su miedo a desaparecer por completo no hace nada y solo confía que su cuerpo pueda resistir la explosión atómica causada por Khris.

Resulta una explosión más fuerte que cualquier explosión atómica dada alguna vez en la historia.

Toda la capital del imperio queda completamente reducida a cenizas y todo el país del norte queda por completo en ruinas.

Charles y Khris fueron consumidos por la explosión, no quedando ningún rastro aparente de ambos ya que fueron el corazón de la destrucción.

La guerra ha terminado, el imperio del norte una amenaza mundial de casi mil años de existencia ha sido finalmente eliminado. La paz ha vuelto al mundo y Latinoamérica crece prospera y libre.

Pasan cientos de años, los escombros del imperio del norte ahora yacen enterrados miles de kilómetros bajo el agua.

Una expedición de investigadores está realizando una exploración en esa zona, tratando de rescatar ciertas ruinas del fondo del mar, con el fin de documentar la historia.

Al remover una piedra descubren dos pequeñas tarjetas electrónicas aparentemente dañadas, con un pequeño foco de cristal en el centro, las llevan a la superficie y al recibir la luz del sol inesperadamente brillan y el foco de cristal se vuelve a activar en ambas, mostrándose dentro de este impulsos eléctricos continuos, hecho ante el cual los investigadores observan asombrados.....

## **Resultado 2: Documento de Diseño**

### **Título del videojuego**

Soul of Wars (Alma de guerras)

### **Diseñadores**

Meyling Antonieta Lara Narváez

Jaedddson Jeannick Sánchez Arana

José Ramón Duran Ramírez

### **Género**

Acción / Disparos en tercera persona

### **Plataforma**

Computadora personal (PC)

### **Sinopsis de Jugabilidad y Contenido**

En este juego se vive la experiencia de la supervivencia ya que el personaje se abrirá paso en territorio enemigo por si solo luchando contra gran cantidad de oponentes tratando de obtener la libertad, el género tercera persona permite explorar en gran detalle áreas muy variadas, que afectan de manera directa cómo se desempeña el personaje en combate presentándose diferentes niveles de desafío a medida que las características del área vayan cambiando.

En el juego el usuario encarna a Khris Fénix un prototipo de androide con mente humana y de grandes habilidades el cual es la pieza clave para la victoria en la guerra del Imperio del Norte y la Unión latinoamericana. El objetivo del personaje es primeramente escapar del corazón de la fortaleza enemiga en la cual se encuentra cautivo y poder incorporarse en el campo de batalla. A medida que el personaje avance irá descubriendo una variedad de habilidades en su cuerpo las cuales le permitirá llevar finalmente a las fuerzas de la unión latino americana a la victoria sobre el imperio.



## **Mecánica**

Se presenta un entorno abierto no lineal donde el jugador podrá abrirse camino por diferentes vías la cuales podrá escoger libremente , enfrentándose a sus oponentes los cuales lo podrán atacar individualmente o en grupo y el jugador deberá colocarse en las posiciones precisas que le permitan evitar los ataques y contraatacar, todo esto lo hará utilizando su armamento el cual tendrá también un límite y que podrá recargar en diversos puntos del escenario sin antes enfrentar ciertas dificultades en el proceso.

## **Tecnología**

### **Software**

Sistema operativo Microsoft Windows 7

Unreal Developmet Kitment (UDK)

Autodesk 3DS Max Studio 2012

### **Hardware**

Procesador Intel core i5 3.2 GHz

Memoria RAM 4 GB DDR3

Disco Duro 500 GB

Tarjeta de Video 3 GB Geforce 8400 DDR3

Monitor 17" entrada HDMI

### **Público**

El video juego está dirigido a una audiencia con edad mayor de 17 años.

### **Mecánica del juego**

Cámara: perspectiva 3D en tercera persona, adaptable a la posición del personaje automáticamente, no controlable.

**Periféricos:** teclado, mouse, parlantes

## **Controles**

**Teclas direccionales:** ir a la izquierda, ala derecha, adelante y atrás.

**Tecla A:** ir a la izquierda

**Tecla D:** ir a la derecha

**Tecla W:** ir adelante

**Tecla S:** ir atrás

**Barra espaciadora:** Saltar

**Tecla c:** Agacharse

**Tecla G:** Cambiar de arma

**Tecla Enter:** Tomar Vehículo

**Tecla Escape:** Pausar

## **Niveles**

### **Nivel I**

**Título:** Pesadilla en la prisión subterránea.

**Encuentro:** El primer nivel inicia cuando el personaje despierta en una prisión y escapa de su celda.

**Descripción:** Una extensa prisión en forma de laberinto que consta de dos pisos donde se encuentra un laboratorio y diversos sujetos de prueba que son sometidos a experimentos.

**Objetivos:** Encontrar la salida del laberinto y encontrar la llave que abre esa salida, derrotar a los enemigos que encuentra en su camino y derrotar al jefe final.

**Enemigos:** Droides esparcidos en todo el laberinto y jefe final, el cual es un droide de mayor tamaño y resistencia.

**Items:** Armas, vida, llaves, vehiculo, armadura, vida extra.

**Música y Efectos de Sonido:** Música de suspenso.

## **Nivel II**

**Título:** Batalla en la fortaleza.

**Encuentro:** Inicia luego que el personaje toma el elevador que lo lleva desde el subterráneo a la superficie dentro de la misma fortaleza.

**Descripción:** Una fortaleza de dos pisos de forma cuadrada con una sola salida disponible.

**Objetivos:** encontrar la llave que abre la puerta de salida de la fortaleza derrotar a los enemigos que obstaculizan la salida y escapar.

**Enemigos:** Droides esparcidos en toda la fortaleza y jefe final, el cual es un droide de mayor tamaño y resistencia que protege la salida.

**Items:** Armas, vida, llaves, vehículo, armadura, vida extra.

**Música y Efectos de Sonido:** Música de acción y batalla.

## **Nivel III**

**Título:** Escape a la libertad

**Encuentro:** Inicia una vez que el jugador atraviesa la puerta principal de la fortaleza que lo lleva al exterior.

**Descripción:** Campo abierto de batalla.

**Objetivos:** Derrotar a los enemigos ubicados en el campo abierto y abrirse camino hacia la libertad.

**Enemigos:** Vehículos enemigos, droides y tanques de guerra.

**Items:** Armas, vida, llaves, vehículo, armadura, vida extra.

**Música y Efectos de Sonido:** Música de acción y batalla épica.

## **Personajes**

**Nombre del Personaje:** Khris Fénix



**Descripción:** androide con cuerpo robótico, de gran altura y textura metálica.

### **Concepto:**

Khris Fénix de 20 años es un joven de origen sudamericano que siempre estuvo en contra de la opresión, se enlista en el ejército de la Unión Latinoamericana, llevando a cabo el entrenamiento, pero le es detectado un tipo muy raro de cáncer y se encuentra etapa terminal. Conoce a un científico que lo anima a ofrecerse como voluntario para someterse al proyecto de digitalización mental.

Luego de esto, se dio el origen de un ser biónico, capaz de pensar con toda la lógica e inteligencia de un ser humano, pero a su vez con toda la potencia y velocidad de una máquina, permitiéndole también un multiprocesamiento de miles de ideas al mismo tiempo, originando un ser supremamente inteligente.

De igual manera al ser un androide con energía nuclear ilimitada, este nuevo ser podía correr a altas velocidades y trabajar por una cantidad de horas ilimitadas, dando origen al soldado y al ser perfecto.

**Encuentro:** El personaje aparece en el inicio del juego cuando esta despierta en una celda descubriendo que es prisionero en una gran fortaleza.

**Habilidades:** Correr a alta velocidad, luchar por tiempo indefinido sin agotarse, manejo de distintos tipos de armas, resistencia a los ataques.

**Armas:** Rifle de rayos laser, cañón

## **ENEMIGOS**

**Nombre:** Droides.

**Descripción:** Robots blindados que portan un arma de de ataque.

**Encuentro:** aparecer al iniciar el juego en la prisión subterránea.

**Habilidades:** Perseguir y disparar, robar vida, robar armas.

**Armas:** Cañón de rayos de energía atómica.

## **Guión**

### **Soul of Wars (Alma de guerras)**

Año 3023

El imperio norte americano, luego de la escases de petróleo y recursos, en medio de la desesperación, opta por invadir los países aledaños tomando ventajas de su avanzado desarrollo tecnológico y militar, teniendo como principal objetivo los recursos de Centro y Latinoamérica.

El objetivo del imperio del norte al invadir el territorio del sur, además de expandir sus dominios y aumentar sus recursos económicos y bélicos, es también hacerse poseedor de un raro recurso natural descubierto muy recientemente, solo encontrado en territorio Latinoamericano, llamado Mineral X. Este mineral combinado con tecnología de fusión nuclear podría crear una fuente de energía de proporciones atómicas nunca antes poseída.

La invasión inicio por México, la reacción del resto de países latinoamericanos temiendo una pronta invasión y viéndose en desventaja en el caso de contraatacar individualmente, deciden establecer una alianza con el objetivo de unir fuerzas y hacerle frente a la invasión, funcionando todos como uno, aportando todos sus recursos para desarrollar una tecnología militar que les permita defenderse del avanzado armamento norteamericano. Esta Unión se le denomino Unión Latinoamericana.

Mientras se constituye esta unión, la invasión del imperio continua avanzando en su ataque contra el territorio de Latinoamérica.

La unión Latinoamericana, trata de desarrollar sus tecnologías en secreto, teniendo como objetivo, que con pocos recursos puedan hacer frente a la masiva cantidad de unidades militares que posee el imperio del norte, sabiendo que no cuentan con igual ventaja numérica que el enemigo, tienden a enfocarse más en el poder del arma que en la cantidad de unidades que desarrollaran.

Después de mucho tiempo de experimentación e implementación de modelos fallidos, logran crear un arma biónica, un androide de guerra equipado de potentes armas, las cuales son utilizadas en el campo de batalla para hacer frente al armamento bélico del imperio del norte.

Pero incluso, estos soldados biónicos no logran ser lo suficientemente potentes para poder superar la ventaja numérica y potencial del imperio del norte, dejando a la Unión Latinoamericana en una posición vulnerable ya que todas sus posibles defensas hasta ahora habían sido abatidas, dejando un gran saldo de soldados tanto humanos como maquinas abatidos en el campo de batalla quedando expuestos a una invasión total y a su conquista inminente.

Estas máquinas tenían un problema, carecían de lógica y del espíritu de lucha que caracteriza a la raza humana, por lo que no tuvieron un buen desempeño en el campo de batalla, derivando estos hechos en la perdida de los países de la Unión Latinoamericana desde México hasta Guatemala.

Ante esto la Unión latinoamericana empieza en secreto el proyecto de tecnología de guerra más complejo y grande que jamás se haya creado, utilizando como uno de los recursos principales el precioso mineral X sabiendo que en su territorio cuentan con los últimos yacimientos de este mineral, iniciando así el desarrollo de un androide que posea una fuente de energía ilimitada, utilizando fusión nuclear , y las propiedades del mineral X que permiten controlar y canalizar la potencia de esta energía atómica.

Sin embargo, hasta el momento semejante poder no había podido ser controlado por ningún tipo de computadora o inteligencia artificial, dando como resultado un androide inestable que terminaba autodestruyéndose y por lo tanto inutilizable en la guerra.

Debido a estos inconvenientes y a la prisa por sobrevivir, la unión latinoamericana decide llevar a cabo el desarrollo de esta arma a un nivel que nunca imaginaron alcanzar. Con el fin de hacer frente a la fuerza del imperio del norte, debían crear el arma perfecta, ya poseían un androide con la potencia requerida pero aún no habían logrado tener el control sobre ella.

El proyecto para perfeccionar el arma fue bautizado bajo el nombre de “Digitalización Mental” cuyo objetivo era mejorar las habilidades de lógica y perseverancia. Para poder controlar a la maquina más potente diseñada por el hombre, debían de utilizar la maquina más compleja...La mente humana.

El proyecto consistía en la implantación de una mente humana en el androide, solo la mente humana, la cual es más compleja que cualquier máquina, podría controlar la potencia del androide dentro de su cuerpo y canalizarla, evitando que la energía lo domine y se autodestruya.

El proyecto de digitalización mental radicaba en tomar los impulsos eléctricos de las neuronas del cerebro, en las cuales se encuentran la memoria, la personalidad, las emociones y pensamientos y transferir estos impulsos eléctricos a un cerebro digital en forma de tarjeta electrónica en cuyo centro se encuentra un cristal del mineral X y que es colocada en el androide, migrando toda la esencia de la mente humana a una máquina.

Sin embargo, hasta ese momento los intentos de digitalizar una mente humana habían sido fallidos, debido a los sentimientos de miedo a la muerte que habían experimentado los voluntarios a la hora de llevar a cabo el proceso, no sobreviviendo a este.

Mientras sucedían estos hechos un joven llamado Khris Fénix de origen sudamericano, que siempre estuvo en contra de la opresión, habiendo sido inculcado por su padre y su abuelo en los ideales Revolucionarios del pasado, como la libertad y la independencia. Al ser Khris testigo de la invasión del imperio inmediatamente se enlista en el ejército de la Unión Latinoamericana, llevando a cabo el entrenamiento y alistándose para ir a combatir al campo de batalla.

Pero en los últimos días de su entrenamiento sus fuerzas vitales comenzaron a decaer y su cuerpo a debilitarse de manera inexplicable, después de muchos exámenes médicos le es detectado un tipo muy raro de cáncer y se encuentra etapa terminal, no quedándole mucho tiempo de vida, debido a esto queda fuera de lucha al lado del ejército Latinoamericano.

Luego de estos hechos Khris se sumerge en total depresión viendo como sus amigos luchan por la libertad dando la vida en batalla, mientras él no tiene el cuerpo apto ni para caminar, su desesperación crece cuando ve que las fuerzas de la Unión son dominadas casi en su totalidad, siendo obligadas a retroceder, perdiendo grandes territorios.

Khris lloraba amargamente todos los días ante el sentimiento de impotencia que experimentaba por su enfermedad, esto sucedió hasta que uno de los científicos desarrolladores del proyecto de digitalización mental lo observa y le hace las siguientes preguntas:

- ¿El deseo que sientes dentro de tu alma de luchar por la libertad de tu gente, trasciende más allá de los límites físicos de tu cuerpo?
- ¿Tu deseo es lo suficientemente fuerte como para trascender más allá de tu muerte inminente?
- ¿De qué serías capaz con tal de tener una oportunidad de poder llevar a tu gente a la libertad?



Las preguntas que el científico hizo al joven lo inundaron de un espíritu de lucha que lo llenó como fuego por dentro, este poniéndose de pie a duras penas, exclama con todas sus fuerzas un fuerte grito con lágrimas en sus ojos: “ SI LO HARIAAAAA!!!!!!! “

Al ver el científico que en este joven, su mente era mucho más fuerte que su débil cuerpo, le propone la idea de ofrecerse como voluntario para el proyecto, a lo que él responde afirmativamente, sin pensarlo dos veces.

Es entonces cuando Khris es sometido al proceso de digitalización, abandonando todo miedo a la muerte y apegado únicamente al deseo de liberar a su gente.

El proceso de digitalización mental, dio inicio.

La mente humana, recuerdos, personalidad, valores y sentimientos, en fin toda la esencia de un ser humano, estaba siendo trasladada en forma de impulsos eléctricos, a un cerebro electrónico; Proceso al cual hasta ahora nadie ha sobrevivido, ya que en experimentos anteriores, los impulsos eléctricos causados por el miedo a la muerte, habían saboteado el proceso, matando a los voluntarios.

El proceso parecía estar llevándose, en los parámetros esperados, y figuraba ser todo un éxito, pero de pronto....

La invasión había llegado hasta ellos, debido a que cierta información del proyecto, había sido filtrada al imperio, siendo finalmente atacados en ese preciso momento.

Todas las personas, que se encontraban en el lugar fueron asesinadas, toda la maquinaria fue destruida, y el proceso interrumpido abruptamente, no completándose. Los dos científicos fueron secuestrados, para su interrogatorio y tortura.

El cuerpo de Khris se quedó sin mente, inerte y vacío. Sin embargo, su mente, no pudo ser aparentemente digitalizada al cien por ciento al interrumpirse el proceso, y como resultado el androide no mostraba signos de vida. El proceso había fracasado una vez más.B

Los soldados invasores, robaron el prototipo del androide, para estudiarlo, y reproducirlo en masa. El prototipo fue llevado a la base militar ubicada en los límites de los territorios hasta ahora conquistados por el imperio, la zona que alguna vez correspondió al país de honduras. Acá fue, internado en una prisión de máxima seguridad, donde se iba a ser estudiarlo.

Pero algo, paso....

El proceso de digitalización, no fue un fracaso del todo.

Ya que estando en la prisión, el último suspiro de vida de Khris, se conservaba aun, debido a que se rehusaba a morir, y esta fuerza de voluntad provoco un estallido de energía, luego el cuerpo del androide recobro su vida, resultando el proceso de digitalización que había sido aparentemente un rotundo fracaso, todo un éxito.

Fue así como se dio el origen de un ser biónico, capaz de pensar con toda la lógica e inteligencia de un ser humano, pero a su vez con toda la potencia y velocidad de una máquina, permitiéndole también un multiprocesamiento de miles de ideas al mismo tiempo, originando un ser supremamente inteligente.

De igual manera al ser un androide con energía nuclear ilimitada, este nuevo ser podía correr a altas velocidades y trabajar por una cantidad de horas ilimitadas, dando origen al soldado y al ser perfecto.

Khris, despierta en su nuevo cuerpo, y se da cuenta que ya no está en el mismo lugar, que es prisionero y rápidamente deduce que lo están estudiando, Procede de inmediato a buscar su libertad, eliminando todo aparato en el laboratorio que pudiera contener algún registro de él.

Mientras se adapta a su nuevo cuerpo sus energías y poder no son comprendidos por el mismo y no los puede explotar a su máxima expresión.

A cómo puede intentar abrirse paso a través de la prisión, enfrentándose a soldados, androides y mutaciones biológicas que el imperio tiene prisioneros como sujetos de pruebas.

Explora cada uno de los pisos de la fortaleza en busca de la salida, y es atacado por todas las defensas, enfrentándose a algunos oponentes bastantes fuertes en comparación al dominio que tenía de su poder en ese momento.

Después de explorar y luchar intensamente, no consigue encontrar a los científicos que fueron tomados como rehenes. Cuando estaba muy cerca de la salida se encuentra frente a frente con el guardián superior de la fortaleza, una verdadera máquina destructiva, con una magnitud de poder desastrosa y un tamaño descomunal, quedando encerrado en una lucha uno a uno con este enemigo.

La lucha se lleva a cabo siendo la primera donde Khris se siente verdaderamente presionado, siendo su oponente muy difícil de superar. Tras un intenso esfuerzo después de insistir constantemente logra derrotarlo, descubriendo una vía de escape de la prisión. Ahí encuentra un vehículo reservado para emergencias y lo aborda, siguiendo un camino directo al exterior.

Sin embargo esta libertad no le iba a salir fácil, ya que es inmediatamente perseguido por las fuerzas guardianas de la prisión, que abordo de naves, le disparan durante todo el camino de huida; después de luchar y luchar para esquivar sus ataques, logra escapar de sus perseguidores liberándose al fin.

Inmediatamente se dirige a la zona donde fue capturado, temiendo que las fuerzas invasoras hayan avanzado y conquistado el país completo. Al llegar efectivamente descubre que las fuerzas de la Unión han caído en batalla y que la mayoría de combatientes han sido ejecutados.

Arribando en el momento preciso en que las fuerzas del imperio ya estaban dando por tomada toda el área correspondiente al territorio de Nicaragua, sin demora se lanza el solo contra el ejército completo, abriéndose paso a través de los soldados y robots, adentrándose en una encarnada lucha, contra los más fuertes.

Enfrentando todo tipo de ataques como Bombas muy potentes, rayos, tanques de guerra con poder masivo, naves, robots, logra finalmente avanzar al corazón de la trinchera enemiga, después de batallar solo por largas horas.

El corazón de la trinchera era una torre, en la frontera de Honduras-Nicaragua, donde estaba siendo descargada una gran cantidad de robots y armamento por una inmensa nave nodriza en la punta de la torre. Al llegar Khris a la cima de la torre, la nave nodriza toma posición de batalla contra él, elevándose por encima de la torre, demostrando su imponente tamaño.

La nave Nodriza, empieza un ataque con armas de inmensa magnitud destructiva, que Khris apenas logra esquivar y de recibirlas lo dañarían severamente.

Khris Dispara con todo su armamento a la coraza de la nave, pero no logra hacerle daño notorio, intentando una y otra vez pero sin ningún efecto. La nave derriba a Khris desde la punta de la torre hacia las afueras de esta, la nave lanza un implacable ataque lleno de potencia, destruyendo todas las trincheras que se encontraban en el lugar, no dejándole un sitio seguro para protegerse, sin embargo él logra volver a la torre y subir a la cima, de esta a duras penas esquivando los destructivos ataques.

Pero esta vez Khris nota un detalle particular en la parte inferior de la nave, cierto punto débil en el mecanismo de apertura de la compuerta, al que dispara con toda su potencia abriendola, a la cual salta para introducirse dentro de ella.

Una vez dentro acaba con toda la tripulación, pero antes tiene que enfrentar una dura batalla con un robot de gran potencia llamado destroyer, que le da muy dura lucha y al final logra derrotarlo, acabando con los pilotos de la nave y tomando el completo control de esta.

Utilizando la nave destruye a todo enemigo en ese territorio de guerra, siendo la primera vez que la Unión Latinoamericana logra su primera victoria contra el imperio deteniendo el avance de las tropas conquistadoras.

Pero la misión no estaba completa ya que Khris necesitaba avanzar hacia el corazón del imperio establecido en el norte, donde debía poner fin a esta nación demente y traer finalmente la paz al mundo.

Por lo que se dirige hacia al norte, utilizando la nave nodriza que tomó, es interceptado en el camino por un sin número de enemigos aéreos, a los cuales derrota haciendo uso de las habilidades combinadas de la nave y las suyas.

Hasta que de pronto es atacado por un robot de habilidades superiores llamado Air Slayer, el cual posee la capacidad de volar por su propia cuenta a velocidad supersónica, y rápidamente descubre el mismo punto débil de la puerta, a la cual ataca y logra abrir también.

Entrando en la nave tomando sorpresivamente a Khris por el cuello y sacándolo junto con el de un gran impulso por la ventana, dejando a la nave sin control la cual se derriba.

El Air Slayer lo eleva a una gran altura, mientras Khris intenta moverse para atacar su enemigo lo neutraliza atándolo con unas fajas de acero que envuelven todo el cuerpo de Khris.

Es ahí que el instinto de supervivencia que diferencia a Khris de una máquina, entra en acción ya que descubre una nueva habilidad donde puede canalizar su energía no solo a través de su arma, sino a través de todo su cuerpo quemando completamente al air slayer.

Luego de esto Khris se encuentra en caída libre junto con el cuerpo quemado de air slayer, sin embargo logra ver que las partes de propulsión siguen intactas, y las incorpora en su propio cuerpo ganando así la habilidad para volar de manera supersónica.

Inmediatamente se dirige a la capital del imperio, recorriendo miles de kilómetros en pocos minutos.

Al llegar a la capital del imperio, desciende para no ser detectado por los radares y poder acercarse sin ser detectado.

Entrando por un desagüe a las afueras de la ciudad, se abre paso por el alcantarillado para entrar desapercibido. Sin embargo repentinamente siente un acecho, una criatura parecida a un reptil de enormes patas y brazos largos como de unos tres metros de largo lo acecha, no dejándose ver hasta el ataque. Ahí se libra una dura batalla, donde Khris enfrenta a un enemigo que lo supera enormemente en velocidad.

Una vez habiendo derrotado a su enemigo, Khris se da cuenta que todo ha sido una trampa y que todo el ducto está dinamitado con el objetivo de tenderle una emboscada, así que corre a alta velocidad antes que el lugar sea detonado buscando la salida más, al salir se da cuenta que está en el centro de la ciudad, justamente enfrente del castillo sede principal del imperio.

Ahí le esperaba la armada más grande y temida del imperio, la responsable de haber conquistado la mitad de América, ahí mismo, en todo su esplendor, en contra de un solo soldado.

Pero Khris no era un soldado normal, era alguien que confiaba en sí mismo, que jamás se daba por vencido, alguien comprometido con la libertad, sin dar un paso atrás se enfrenta valientemente a la armada completa, dando inicio a la batalla más encarnizada de la historia.

El castillo se encontraba protegido con un campo de fuerza extremadamente resistente, debido a esto las detonaciones del enfrentamiento no le afectaban.

Mutaciones, tanques de guerra, naves de ataque, naves Nodrizas, Destroyers, Air Slayer, Robots de combate, etc. Atacan masivamente a Khris, pero este no cede, se enfrenta a todos y los va destruyendo uno a uno, nave tras nave va siendo derribada, robot tras robot siendo destruido, tanque tras tanque siendo detonado.

A pesar de sentir que su cuerpo está ya deteriorado por las batallas, la fuerza del corazón de Khris lo impulsa a seguir peleando más y más sin importar el número.

La lucha se extendió durante tres días completos, un solo soldado contra todo lo inimaginable, sin embargo este prevaleció y al amanecer del cuarto día, era el único en pie en el campo de batalla.

De inmediato se dirige al castillo, donde misteriosamente parece recibir una invitación a entrar, ya que el escudo de energía que lo protegía se desactivo y las puertas se abrieron.

Con mucho valor se adentra sintiéndose ya muy cerca de cumplir con su misión de frenar por siempre al imperio.

Hasta que llega a la sala del trono. Ahí sale a su encuentro nada más y nada menos que el monarca absoluto del imperio, el hombre responsable de tanta destrucción y muerte, el emperador Charles.

Un Hombre muy alto, robusto de cabellera blanca, con una fuerte presencia, el cual con una mirada asesina observa fijamente a Khris y con una risa demencial exclama:

-¡Cuanto poder! ¡Cuánta Perfección!, con esta arma, no solo Latinoamérica, también el mundo entero será dominado por el sagrado imperio.

-Es hora, de la prueba final.

Khris lo observa confundido y se pregunta a que prueba podría estarse refiriendo, ya que toda su armada había sido completamente destruida en la guerra de los tres días.

De pronto del suelo emerge una capsula metálica, la cual rápidamente se abre y deja ver que dentro de ella se encuentra un androide de color negro, que para sorpresa de Khris es una réplica exacta de sí mismo, pero con la implantación de una mente sicópata y destructiva.

El enfrentamiento comienza y las fuerzas de ambos son bastante igualadas, cada golpe, cada arma, cada manera de moverse, es nivelada por ambos.

Entonces Charles exclama:

-¡Creíste tu cautiverio, fue en vano!

-¡Creíste que toda la información que extrajimos de ti había sido destruida!  
Jajajajajaja!

-¡Pobre diablo miserable! ¡Caerás ante la máxima expresión de Poder!

La lucha entre Khris y su réplica prosigue hasta que se resume a un impacto de energía de sus más potentes armas. Sin embargo, al llegar a este punto donde ya lo habían dado todo solo algo podría prevalecer, la fuerza de voluntad, esto fue lo que permitió a Khris duplicar su flujo de energía nuclear y emanando poder de todo su cuerpo logra un ataque definitivo con el cual acaba con su doble.

Durante la pelea Charles había desaparecido, terminando esta Khris sube a la Azotea más alta del castillo buscándolo.

De pronto escucha una voz Estrepitosa que viene de todas las direcciones diciendo:

-¡Tu arrogancia, ha llegado muy lejos!

-¡La arrogancia del hombre de Oponerse a los dioses!

Luego Khris mira hacia atrás y ve a un ser resplandeciente y rebosante de poder, una especie de androide muy superior de gran altura, que bajaba del cielo.

Su poder era tan grande que todos los metales en la zona se doblaron y todos los cristales se rompieron.

Tanto era su poder que no volaba por propulsión, sino por la presión ejercida por el poder que emanaba. Y le dice a Khris:



-¿Confundido? Jajajaja esta es mi verdadera forma, el verdadero y único ser perfecto.

Resulta que después de haber logrado construir una réplica de Khris e implantarle una mente perversa, Charles no se detuvo, sino que mejoro el cuerpo androide usando el metal más indestructible de la tierra. Este cuerpo mejoraba el efecto de amplificación y control del poder de la fusión nuclear, dándole un poder destructivo muy superior.

Empeorando la situación la mente que ocupaba este cuerpo fue nada más y nada menos que la mente del Mismísimo conquistador supremo, el emperador Charles.

El emperador no se había arriesgado antes a migrar su mente al prototipo anterior por miedo a la muerte, así que lo hizo hasta que este defecto fuera eliminado en el prototipo que estaba desarrollando. Durante la batalla de Khris con el prototipo anterior, Charles tomo finalmente la decisión de dar inicio al proceso de digitalización del mismo.

El poder del nuevo modelo era exageradamente destructivo y su resistencia parecía infinita. Khris ataco con todas sus fuerzas pero parecía no afectarle en lo absoluto y cada ataque de Charles devastaba terriblemente el cuerpo de Khris.

Siendo Khris brutalmente masacrado y destruido en un ochenta por ciento no se daba por vencido, aún mantenía la confianza de que podría hacer algo aunque le costase la vida.

Así que recuerda que el origen de su poder es la fusión nuclear, la misma que puede dar origen a la explosión atómica y que la función de su cuerpo androide es tomar un control de esta potencia, nivelarla y usarla como fuente de energía y ataque.

Así que corre hacia Charles lo agarra con ambos brazos con toda la fuerza que tiene y destruye las barreras de su propio cuerpo que regulaban la potencia nuclear, desencadenando a consecuencia una gran explosión atómica.

En ese momento Charles intenta usar su gran poder para liberarse pero, la potencia de la explosión atómica no le permitió reaccionar, así que la única posibilidad que le queda Charles es liberar también toda su energía atómica, pero al hacer esto se arriesga a ser destruido por su propia explosión, así que por su miedo a desaparecer por completo no hace nada y solo confía que su cuerpo pueda resistir la explosión atómica causada por Khris.

Resulta una explosión más fuerte que cualquier explosión atómica dada alguna vez en la historia.

Toda la capital del imperio queda completamente reducida a cenizas y todo el país del norte queda por completo en ruinas.

Charles y Khris fueron consumidos por la explosión, no quedando ningún rastro aparente de ambos ya que fueron el corazón de la destrucción.

La guerra ha terminado, el imperio del norte una amenaza mundial de casi mil años de existencia ha sido finalmente eliminado. La paz ha vuelto al mundo y Latinoamérica crece prospera y libre.

Pasan cientos de años, los escombros del imperio del norte ahora yacen enterrados miles de kilómetros bajo el agua.

Una expedición de investigadores está realizando una exploración en esa zona, tratando de rescatar ciertas ruinas del fondo del mar, con el fin de documentar la historia.

Al remover una piedra descubren dos pequeñas tarjetas electrónicas aparentemente dañadas, con un pequeño foco de cristal en el centro, las llevan a la superficie y al recibir la luz del sol inesperadamente brillan y el foco de cristal se vuelve a activar en ambas, mostrándose dentro de este impulsos eléctricos continuos, hecho ante el cual los investigadores observan asombrados.....

**Miembros del equipo**

**Project Manager:**

Meyling Lara

**Game Designer:**

Jaeddson Sánchez

**Game Developer**

José Duran

**Equipo de trabajo:**

Meyling Lara

Jaeddson Sánchez

José Duran

**Detalles de producción**

**Fecha de Inicio**

15 de Mayo 2013

**Fecha de Terminación**

15 de Octubre

## Presupuesto

El equipo para desarrollo del proyecto debe cumplir con las siguientes características:

### Hardware

Procesador Intel core i5 3.2 GHz	\$ 253
Memoria RAM 4 GB DDR3	\$ 44
Tarjeta Madre Asrock	\$ 55
Disco Duro 500 GB	\$ 67
Tarjeta de Video 3 GB Geforce 8400 DDR3	\$ 41
Monitor 17" entrada HDMI	\$ 100

### Software

Sistema operativo Microsoft Windows 7	\$ 60
Unreal Developmet Kitment (UDK)	Sin Costo
Autodesk 3DS Max Studio 2012	Versión Gratuita
<b>Total</b>	<b>\$620</b>

**Resultado 3: Tabla 12. Sprint Plan**

<b>Sprint ID</b>	<b>Start</b>	<b>Days</b>	<b>Ends</b>	<b>Status</b>	<b>Goal</b>	<b>%</b>
1	04/05/2013	15	04/05/2013	Terminada	Presentar el primer diseño del primer nivel del videojuego y la primera propuesta del documento de tesis.	5%
2	18/05/2013	15	18/05/2013	Terminada	Desarrollo del primer nivel y continuar desarrollo del documento de tesis	8%
3	01/06/2013	15	01/06/2013	Terminada	Continuar desarrollo del primer nivel y continuar desarrollo del documento de tesis	8%
4	15/06/2013	15	15/06/2013	Terminada	Finalizar el primer nivel y continuar desarrollo del documento de tesis	9%
5	29/06/2013	15	29/06/2013	Terminada	Presentar el primer diseño del segundo nivel del video juego y continuar desarrollo del documento de tesis	5%
6	13/07/2013	15	13/07/2013	Terminada	Continuar desarrollo del segundo nivel y desarrollo del documento de tesis	8%
7	27/07/2013	15	27/07/2013	Terminada	Continuar desarrollo del segundo nivel, revisión de la tesis y obtener la valoración y recomendaciones del equipo	10%
8	10/08/2013	15	10/08/2013	Terminada	Finalizar el segundo nivel, obtener el documento corregido por todo el equipo, entregarlo al tutor para revisión y corrección	9%
9	24/08/2013	15	24/08/2013	Terminada	Presentar el primer diseño del tercer nivel, obtener corrección del documento realizada por el tutor	5%
10	07/09/2013	15	07/09/2013	Terminada	Continuar desarrollo del tercer nivel y obtener una versión del documento de tesis corregida en base a recomendaciones de tutor.	9%
11	21/09/2013	15	21/09/2013	Terminada	Continuar desarrollo del tercer nivel y obtener la versión final de documento de tesis	10%
12	05/10/2013	15	05/10/2013	Terminada	Finalizar el desarrollo del tercer nivel	9%
13	14/11/2013	15	14/11/2013	Terminada	Obtener el producto final del Videojuego	5%

## Resultado 4: Tabla 13. Feature Log

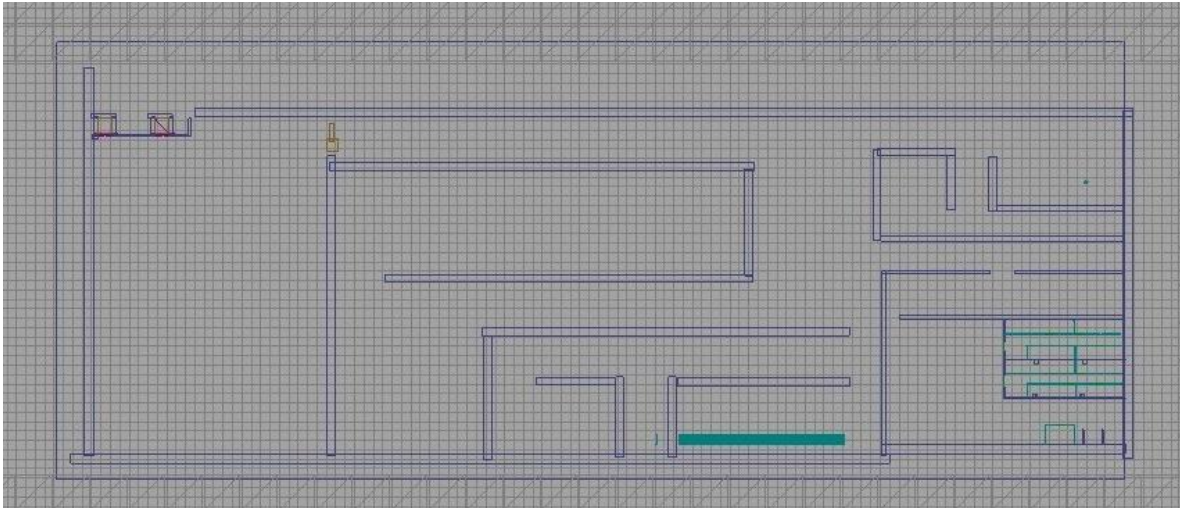
Feature ID	Status	Days	Sprint ID	Comments
1	terminada	15	1	Realizar el primer diseño del primer nivel del videojuego y la primera propuesta del documento de tesis.
2	terminada	15	2	Desarrollar primer nivel y continuar desarrollo del documento de tesis
3	terminada	15	3	desarrollar primer nivel, llevar a cabo pruebas alfa y desarrollar documento de tesis
4	terminada	15	4	Realizar correcciones necesarias, finalizar el desarrollo de primer nivel y continuar desarrollo del documento de tesis
5	terminada	15	5	Realizar el primer diseño del segundo nivel y continuar desarrollo del documento de tesis
6	terminada	15	6	Desarrollar segundo nivel y continuar desarrollo del documento de tesis
7	terminada	15	7	Continuar desarrollo del segundo nivel llevar a cabo pruebas alfa, realizar una revisión del documento de tesis por todo el grupo de trabajo, hacer observaciones y recomendaciones acerca del documento
8	terminada	15	8	Realizar correcciones necesarias y finalizar el desarrollo del segundo nivel, realizar correcciones del documento basadas en las recomendaciones del equipo
9	terminada	15	9	Realizar el primer diseño del tercer nivel, obtener una versión del documento aprobada por todo el equipo de trabajo y someterla a revisión de parte del tutor de la tesis
10	terminada	15	10	Continuar desarrollo del tercer nivel y realizar correcciones al documento basadas en las recomendaciones del tutor de la tesis
11	terminada	15	11	continuar desarrollo del tercer nivel y llevar a cabo pruebas alfa, corregir los últimos detalles de la tesis
12	terminada	15	12	Realizar correcciones necesarias al videojuego y finalizar el desarrollo del tercer nivel
13	terminada	15	13	Llevar a cabo pruebas finales de los tres niveles y obtener la versión final del video juego.

**Resultado 5: Tabla 14. Sprint Backlog**

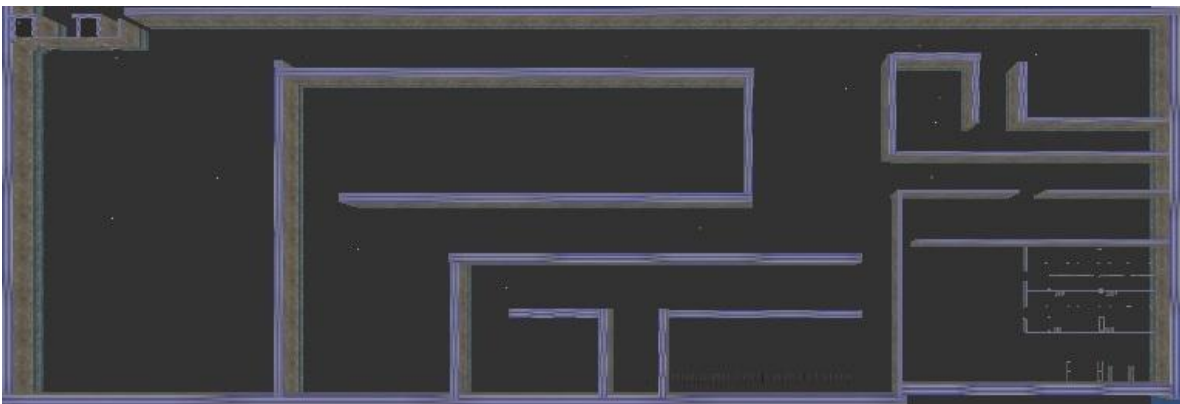
Sprint # Backlog	# Feature ID	Number of tasks	Task	Miembro del equipo	Status
1	1	2	Elaborar primer diseño de primer nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Elaborar la primera propuesta del documento de tesis	Meyling Lara / Jaedson	Terminada
2	2	2	Continuar desarrollo del primer nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Continuar desarrollo del documentos de tesis	Meyling Lara / Jaedson	Terminada
3	3	3	Continuar desarrollo del primer nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Ejecutar pruebas alfa del primer nivel del videojuego	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
			Continuar desarrollo del documentos de tesis	Meyling Lara / Jaedson	Terminada
4	4	2	Corregir los errores detectados en las pruebas alfa	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
			Continuar desarrollo del documentos de tesis	Meyling Lara / Jaedson	Terminada
5	5	2	Elaborar primer diseño del segundo nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Continuar desarrollo del documentos de tesis	Meyling Lara / Jaedson	Terminada
6	6	2	Continuar desarrollo del segundo nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Continuar desarrollo del documentos de tesis	Meyling Lara / Jaedson	Terminada
7	7	3	Continuar desarrollo del segundo nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Ejecutar pruebas alfa del segundo nivel del videojuego	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
			Revisar el documento de tesis, hacer observaciones y recomendaciones	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
8	8	2	Corregir los errores detectados en las pruebas alfa	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
			Corregir el documento de tesis basado en las recomendaciones del equipo	Meyling Lara / Jaedson	Terminada
9	9	3	Elaborar primer diseño del tercer nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Realizar revision final del documento y aprobar documento	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
			Entregar documento de tesis al tutor	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
10	10	2	Continuar desarrollo del tercer nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Corregir documento basado en las recomendaciones del tutor	Meyling Lara / Jaedson	Terminada
11	11	3	Continuar desarrollo del tercer nivel del videojuego	Jaedson Sanchez / Jose Duran	Terminada
			Ejecutar pruebas alfa del tercer nivel del videojuego	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
			corregir los últimos detalles de la tesis basados en las recomendaciones de tutor	Meyling Lara	Terminada
12	12	1	Corregir los errores detectados en las pruebas alfa	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
13	13	2	Ejecutar pruebas alfa de los tres niveles integrados	Jaedson Sanchez / Jose Duran / Meyling Lara	Terminada
			Corregir los errores detectados en las pruebas alfa	Jaedson Sanchez / Jose Duran	Terminada

**Resultado 6: Diseño del mapa de primer nivel**

**Imagen 14. Mapa de nivel 1: Cárcel vista plano**



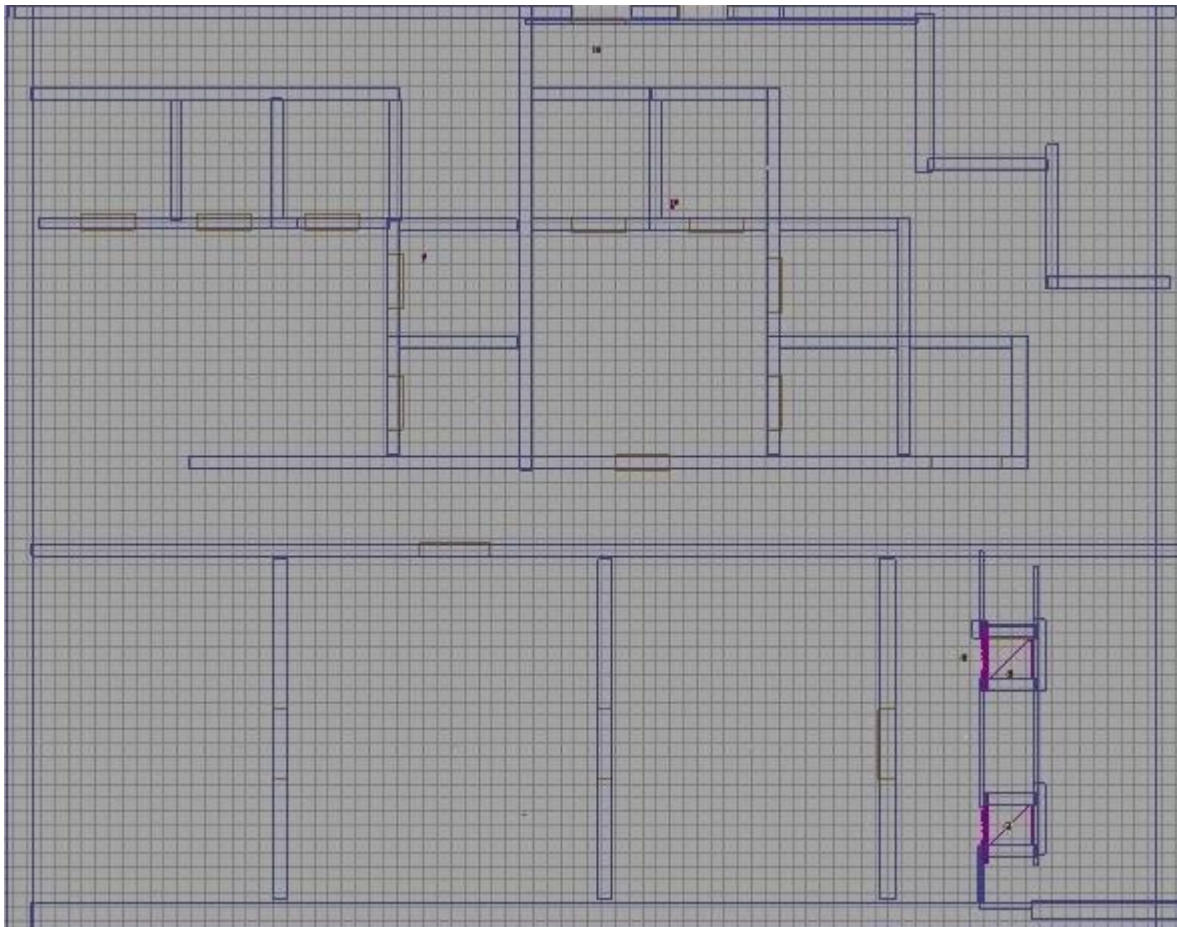
**Imagen 15. Mapa de nivel 1: Cárcel vista modo juego**



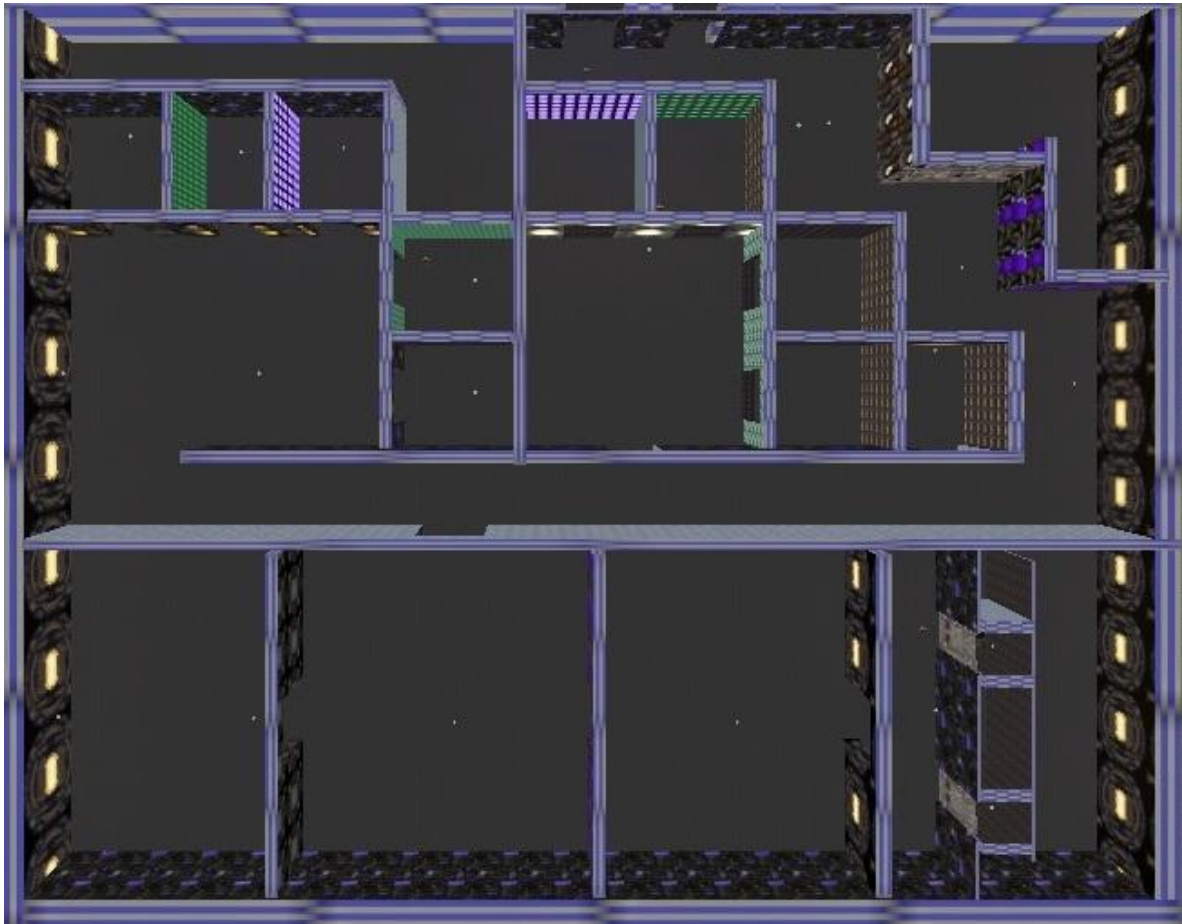


**Resultado 7: Diseño del mapa de segundo nivel**

**Imagen 16. Mapa del Nivel 2: Laboratorio vista plano**



**Imagen 17. Mapa del Nivel 2: Laboratorio vista modo juego**



Resultado 8: Diseño del mapa de tercer nivel

Imagen 18. Mapa del Nivel 3: Campo militar vista plano

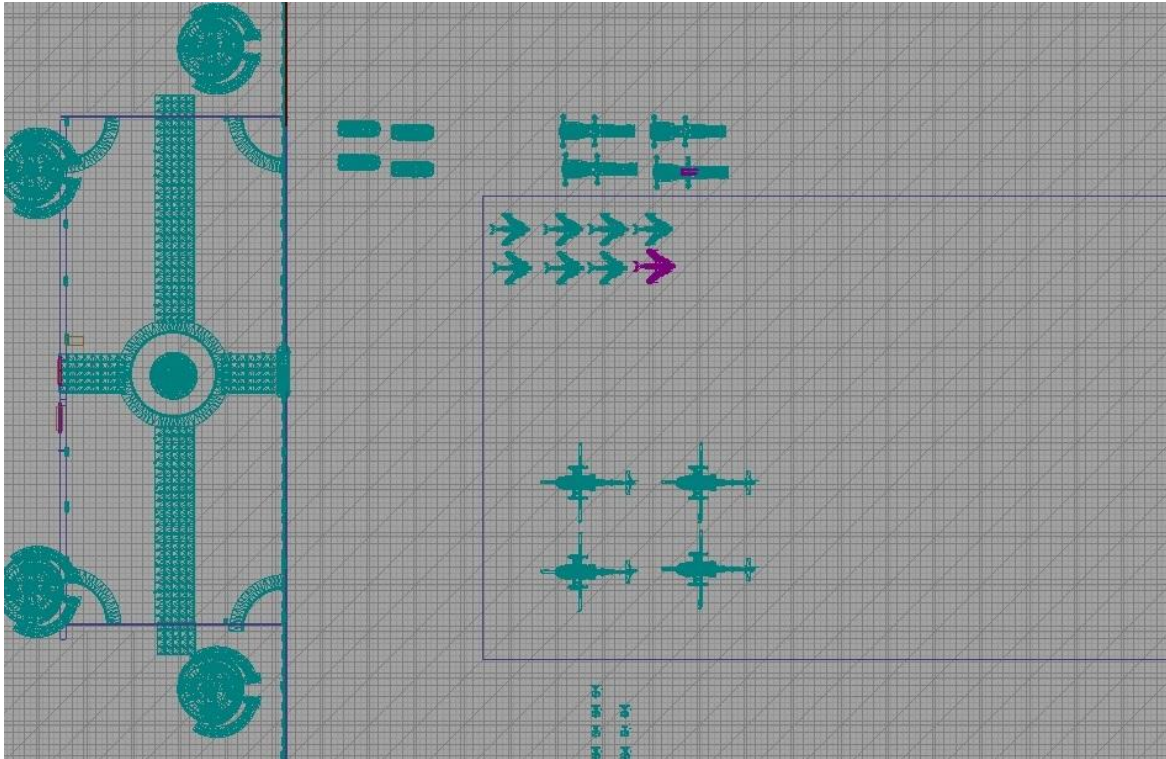




Imagen 19. Mapa del Nivel 3: Campo militar vista modo juego



**Resultado 9: Creación de Materiales**

**Imagen 20: Materiales creados**

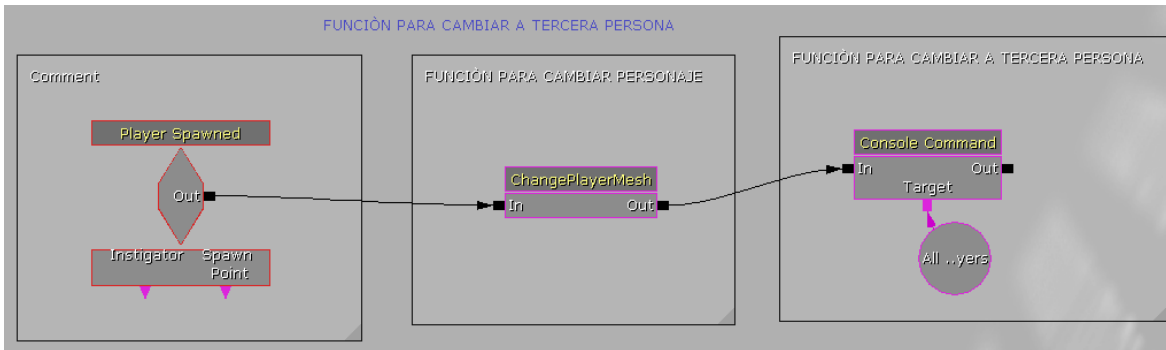


**Resultado 10: Imagen 21. Personaje**



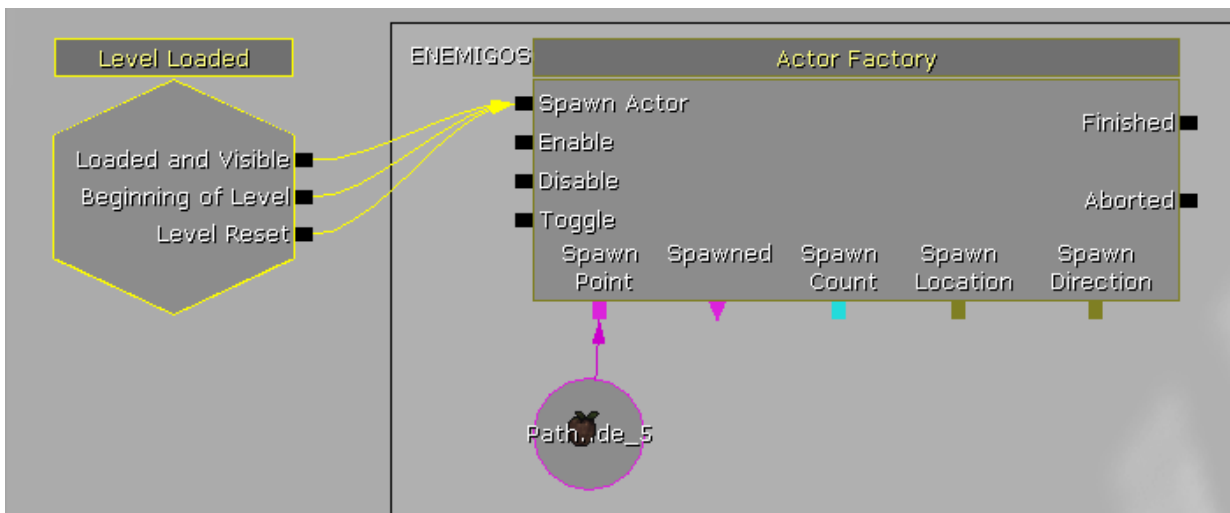
### Resultado 11: Programación de cambio de personaje a tercera persona

Imagen 22. Función tercera persona



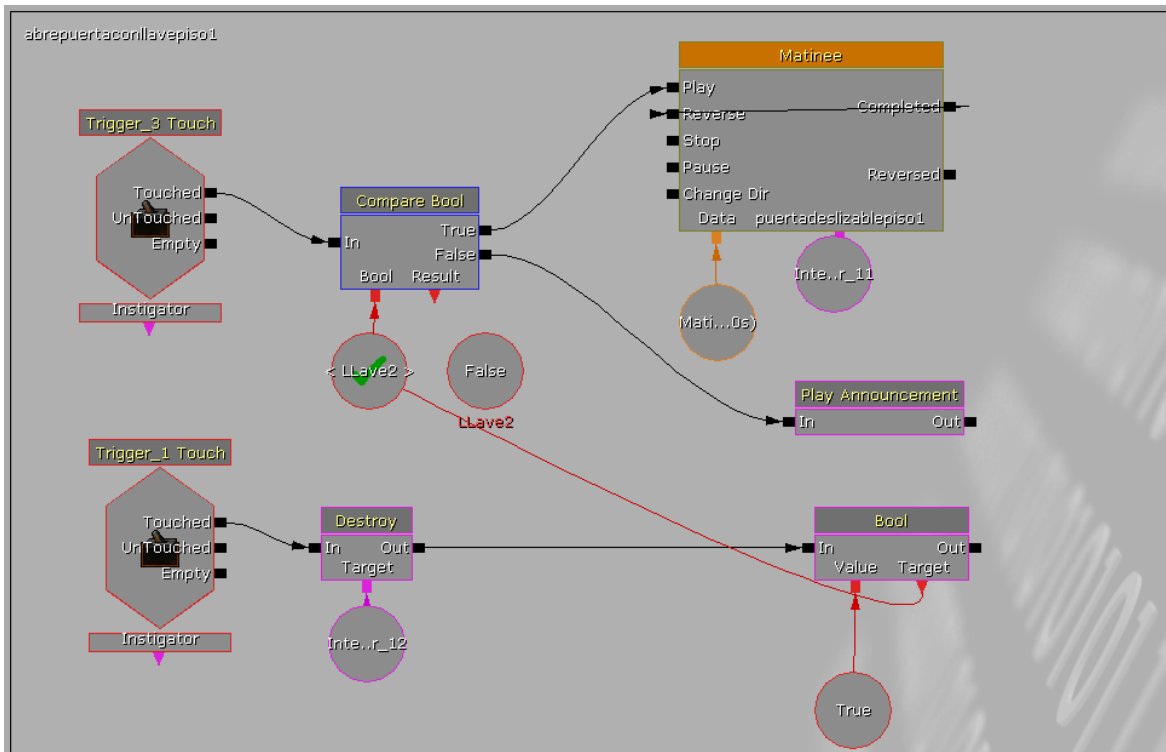
### Resultado 12: Creación de enemigos y sus eventos

Imagen 23. Creación de enemigos



### Resultado 13: Programación de puerta con llave

Imagen 24. Función abrir puerta con llave



### Resultado 14: Programación del evento ascensor

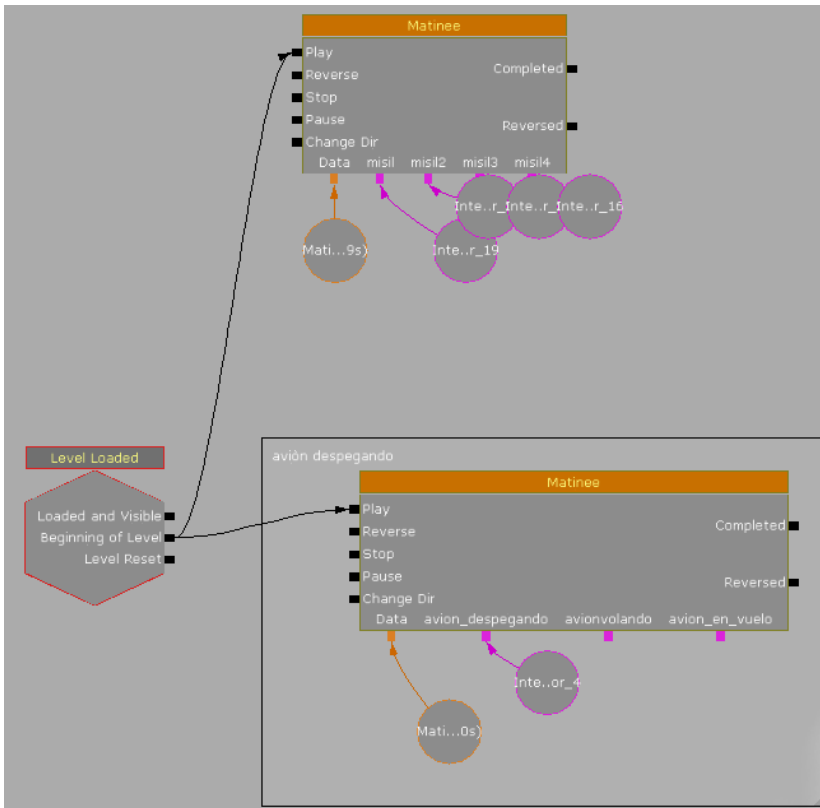
Imagen 25. Función ascensor



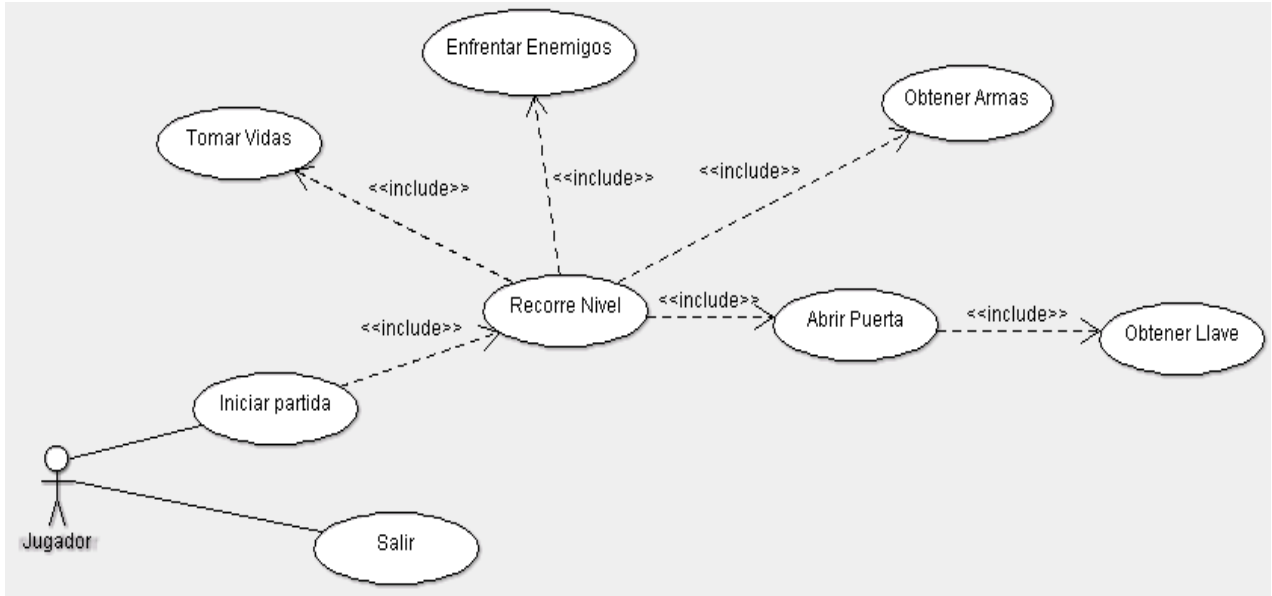


## Resultado 15: Animaciones

Imagen 26. Animaciones



**Resultado 16: Imagen 27. Caso de uso jugador ejecuta el videojuego**



**Descripción del caso de uso**

<b>Caso de Uso</b>	Iniciar partida	1
<b>Actores</b>	Jugador	
<b>Tipo</b>	Principal	
<b>Referencias</b>	Recorrer nivel (Include)	
<b>Precondición</b>	El juego debe estar instalado	
<b>Pos condición</b>	Salir del juego	
<b>Autores</b>	Meyling Lara, Jaeddson Sánchez, José Duran	<b>Fecha</b> 01/11/13 <b>Versión</b> 1

<b>Propósito</b>
Permite al jugador interactuar con el nivel 1 del videojuego

<b>Curso Normal</b>			
#	Acción Usuario	#	Acción Juego
1	Usuario da clic en el botón inicio	2	Cargar Menú
		3	Espera selección del usuario

<b>Caso de Uso</b>	Salir				2
<b>Actores</b>	Jugador				
<b>Tipo</b>	Principal				
<b>Referencias</b>	Ninguna				
<b>Precondición</b>	Haber iniciado el juego				
<b>Pos condición</b>	Aplicación finalizada				
<b>Autores</b>	Meyling Lara, Jaedddson Sánchez, José Duran	<b>Fecha</b>	01/11/13	<b>Versión</b>	1

<b>Propósito</b>
Permite al jugador salir del juego

Curso Normal			
#	Acción Usuario	#	Acción Juego
1	Usuario da clic en el botón salir		
		2	Finaliza aplicación
3	El usuario pierde		
		4	Presenta pantalla de finalizar juego

<b>Caso de Uso</b>	Recorrer nivel				3
<b>Actores</b>	Jugador				
<b>Tipo</b>	Principal				
<b>Referencias</b>	Tomar vidas, enfrentar enemigos, obtener armas, abrir puertas				
<b>Precondición</b>	Iniciar partida				
<b>Pos condición</b>	Pasar al siguiente nivel				
<b>Autores</b>	Meyling Lara, Jaedddson Sánchez, José Duran	<b>Fecha</b>	01/11/13	<b>Versión</b>	1

<b>Propósito</b>
Permite al jugador recorrer el video juego

Curso Normal			
#	Acción Usuario	#	Acción Juego
1	El usuario se moviliza a través del escenario		
		2	Muestra escenario
		3	Muestra vidas
		4	Muestra armas
		5	Muestra llaves

<b>Caso de Uso</b>	Tomar vidas				4
<b>Actores</b>	Jugador				
<b>Tipo</b>	Opcional				
<b>Referencias</b>	Ninguna				
<b>Precondición</b>	Recorrer Nivel				
<b>Pos condición</b>	Incrementa variable vida				
<b>Autores</b>	Meyling Lara, Jaedddson Sánchez, José Duran	<b>Fecha</b>	01/11/13	<b>Versión</b>	1

<b>Propósito</b>
Permite al jugador incrementar variable vida

<b>Curso Normal</b>			
<b>#</b>	<b>Acción Usuario</b>	<b>#</b>	<b>Acción Juego</b>
1	El usuario toma vidas		
		2	Incrementa la variable vida

<b>Caso de Uso</b>	Enfrentar Enemigos				5
<b>Actores</b>	Jugador				
<b>Tipo</b>	Obligatorio				
<b>Referencias</b>	Ninguna				
<b>Precondición</b>	Recorrer nivel				
<b>Pos condición</b>	Disminuye variable vida				
<b>Autores</b>	Meyling Lara, Jaedddson Sánchez, José Duran	<b>Fecha</b>	01/11/13	<b>Versión</b>	1

<b>Propósito</b>
El jugador enfrenta enemigos

<b>Curso Normal</b>			
<b>#</b>	<b>Acción Usuario</b>	<b>#</b>	<b>Acción Juego</b>
1	Dispara armas		
		2	Si dispara entonces disminuye variable municiones
		3	Si enemigo dispara entonces disminuye variable vida

<b>Caso de Uso</b>	Obtener Armas				6
<b>Actores</b>	Jugador				
<b>Tipo</b>	Opcional				
<b>Referencias</b>	Ninguna				
<b>Precondición</b>	Recorrer nivel				
<b>Pos condición</b>	Aumenta variable armas				
<b>Autores</b>	Meyling Lara, Jaeddson Sánchez, José Duran	<b>Fecha</b>	01/11/13	<b>Versión</b>	1

<b>Propósito</b>
Permite al jugador obtener armas

<b>Curso Normal</b>			
<b>#</b>	<b>Acción Usuario</b>	<b>#</b>	<b>Acción Juego</b>
1	Usuario toma armas		
		2	Incrementa variable municiones

<b>Caso de Uso</b>	Abrir puerta				7
<b>Actores</b>	Jugador				
<b>Tipo</b>	Principal				
<b>Referencias</b>	Obtener llave				
<b>Precondición</b>	Recorrer nivel				
<b>Pos condición</b>	Pasar al siguiente nivel				
<b>Autores</b>	Meyling Lara, Jaeddson Sánchez, José Duran	<b>Fecha</b>	01/11/13	<b>Versión</b>	1

<b>Propósito</b>
Permite al jugador pasar al siguiente nivel

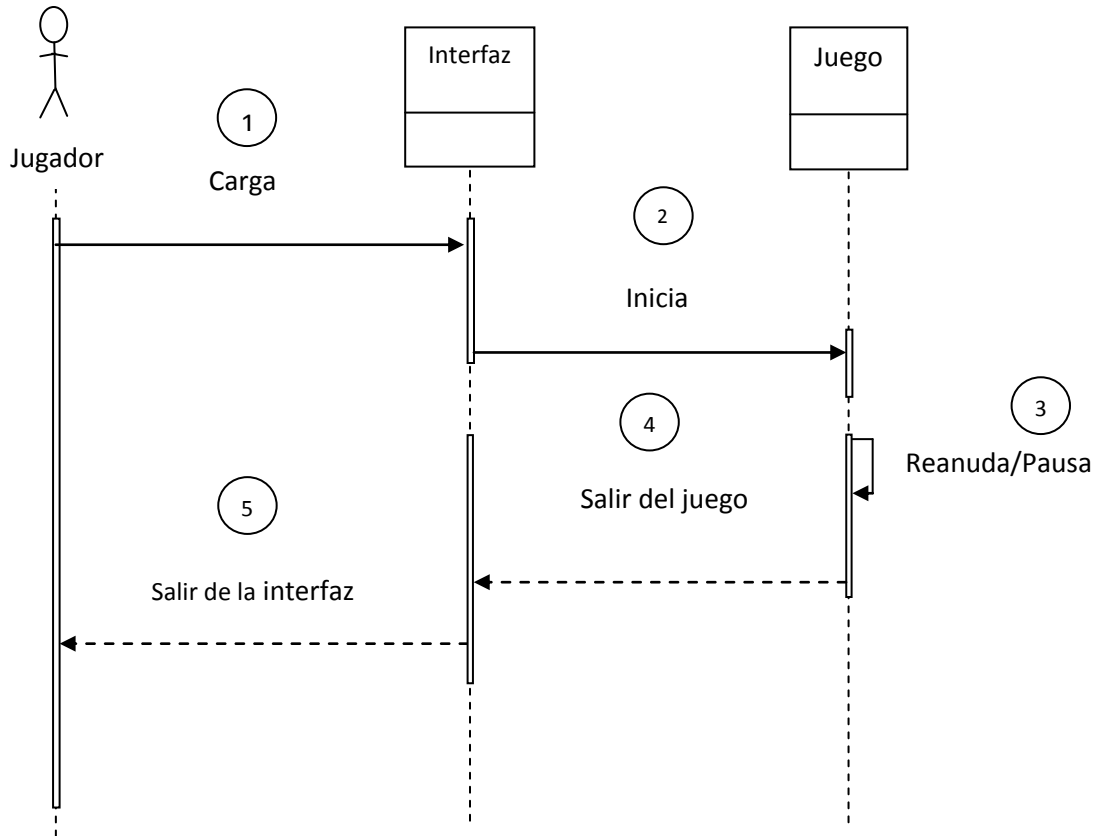
<b>Curso Normal</b>			
<b>#</b>	<b>Accion Usuario</b>	<b>#</b>	<b>Accion Juego</b>
1	Usuario utiliza llave		
		2	El sistema abre puerta

<b>Caso de Uso</b>	Obtener llave	8			
<b>Actores</b>	Jugador				
<b>Tipo</b>	Principal				
<b>Referencias</b>	Ninguna				
<b>Precondición</b>	Recorrer nivel				
<b>Pos condición</b>	Abrir puerta				
<b>Autores</b>	Meyling Lara, Jaedddson Sánchez, José Duran	<b>Fecha</b>	01/11/13	<b>Versión</b>	1

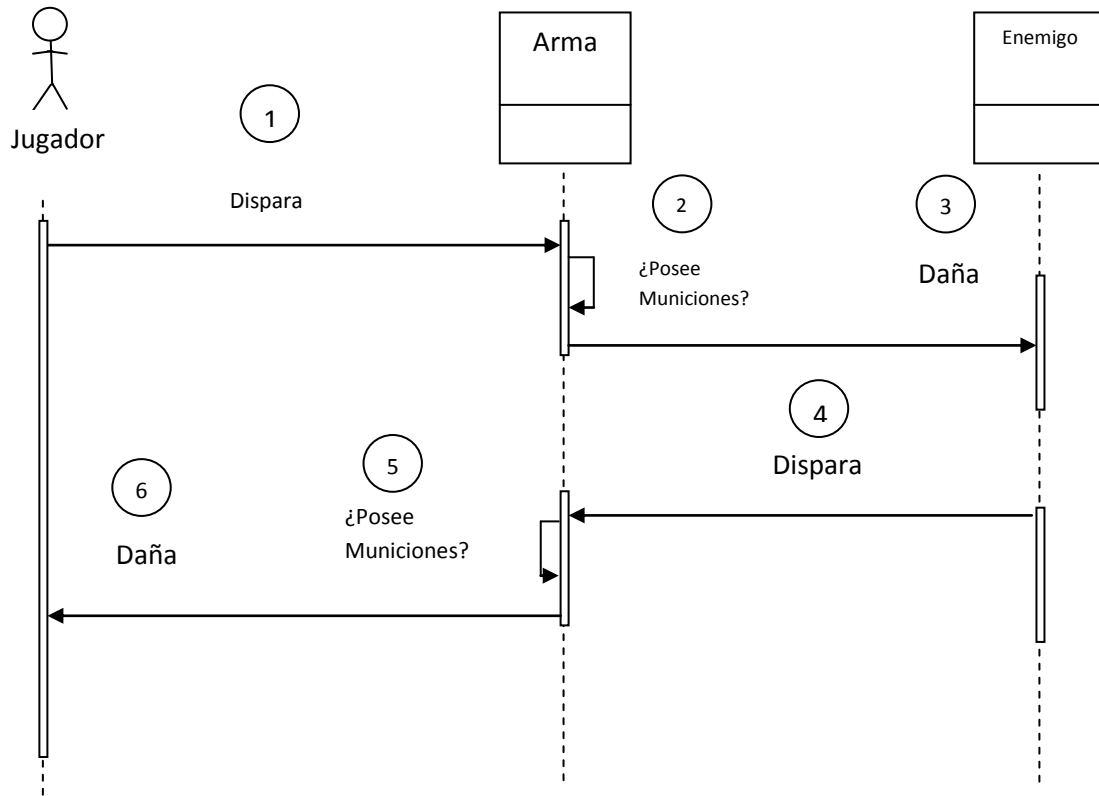
<b>Propósito</b>
Permite al jugador abrir la puerta

<b>Curso Normal</b>			
#	Acción Usuario	#	Acción Juego
1	Usuario obtiene llave		
		2	Sistema asigna valor false a objeto llave
		3	Quitar llave del escenario

**Resultado 17: Imagen 28. Diagrama de secuencia jugador ejecuta el videojuego**

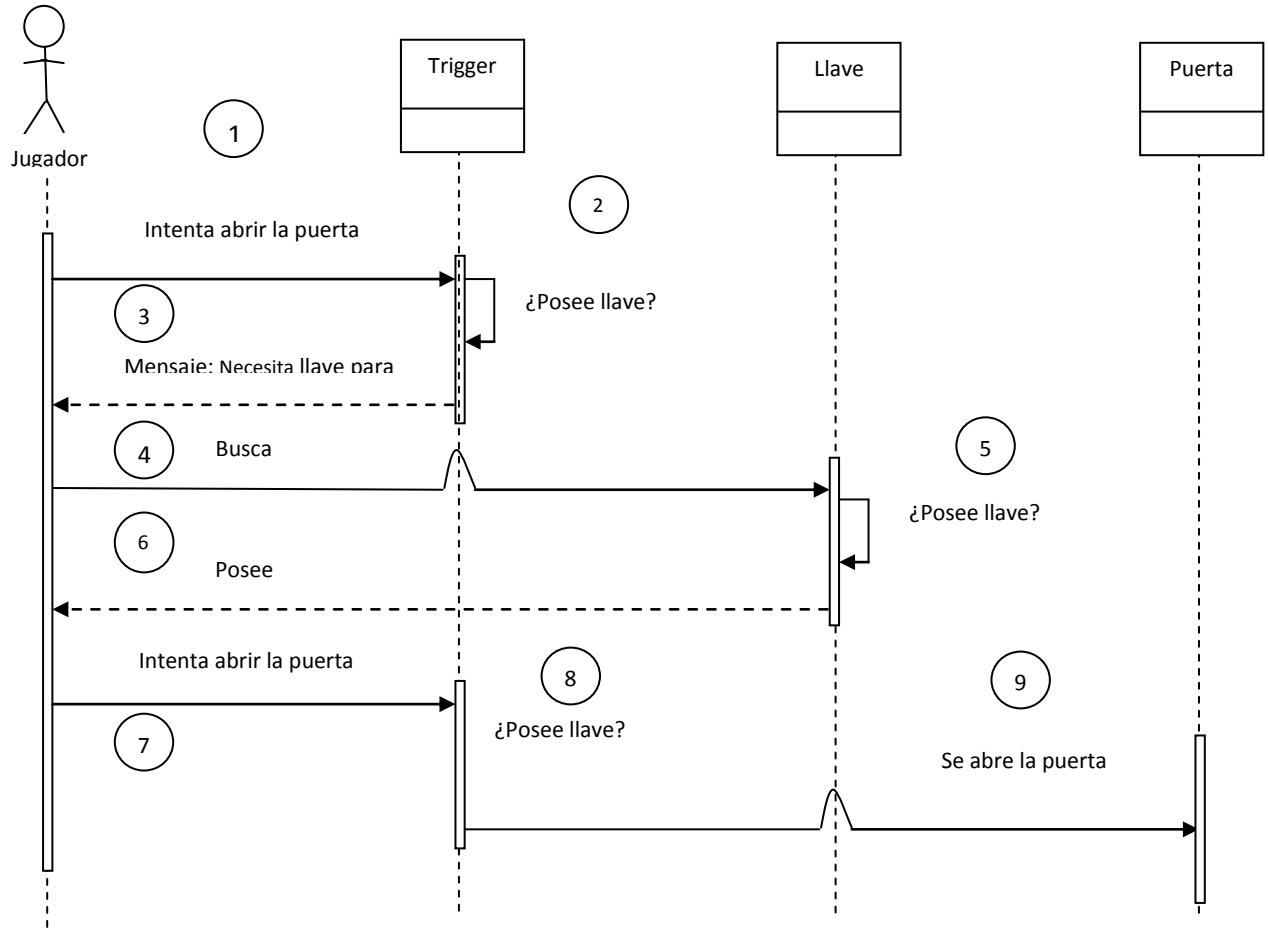


Resultado 18: Imagen 29. Diagrama de secuencia jugador enfrenta enemigo

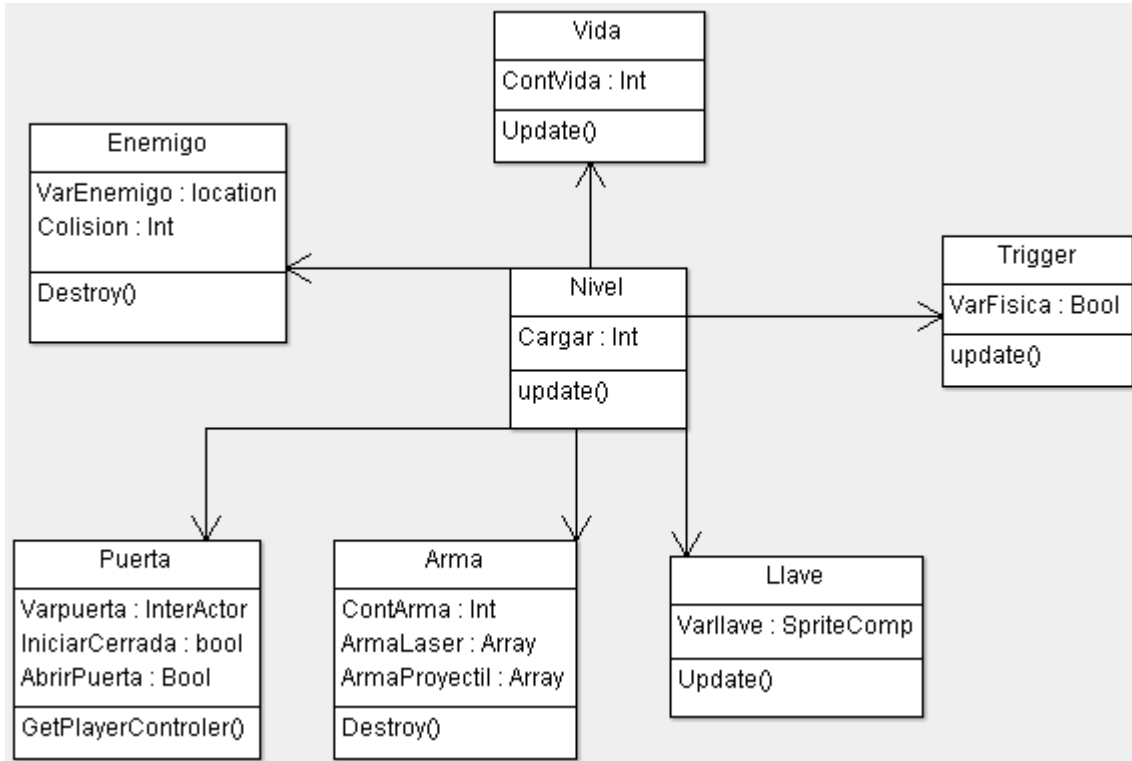




Resultado 19: Imagen 30. Diagrama de secuencia jugador y puerta con llave



Resultado 20: Imagen 31. Diagrama de clases del Videojuego



## **Resultado 21: Reporte Postmortem**

### **Título del videojuego**

Soul of Wars (Alma de guerras)

### **Diseñadores**

Jaeddson Jeannick Sánchez Arana

José Ramón Duran Ramírez

Meyling Antonieta Lara Narváez

### **Género**

Accion / Disparos en tercera persona

### **Plataforma**

Computadora personal (PC)

### **Antecedente**

Este proyecto consiste en el desarrollo de un videojuego del género tercera persona para plataforma PC utilizando Unreal Development Kitment.

Se presenta un entorno abierto donde el jugador podrá abrirse camino por diferentes vías, enfrentándose a sus oponentes colocándose en las posiciones precisas que le permitan evitar los ataques y atacar utilizando su armamento el cual podrá recargar en diversos puntos del escenario.

### **Miembros del equipo**

#### **Projerct Manager:**

Meyling Lara

#### **Game Designer:**

Jaeddson Sánchez

#### **Game Developer**

José Duran

## **Efecto positivo**

### **Actividad 1**

Elegir un motor para crear videojuegos con licencia gratuita.

### **Como seguir generando un efecto positivo**

Investigar acerca de Software de licencia gratuita que pueda ser una herramienta útil y factible en el proyecto.

### **Actividad 2**

Utilizar un programa de diseño de alto atractivo gráfico.

### **Como seguir generando un efecto positivo**

Estudiar que software es el que más se adapta al proyecto y que este permita sacar mayor provecho al hardware y obtener la mejor calidad gráfica.

### **Actividad 3**

Invertir en Hardware de alta calidad

### **Como seguir generando un efecto positivo**

Realizar cotizaciones en diferentes establecimientos, elegir el hardware con él más factible económicamente sin sacrificar los requerimientos.

## **Efecto negativo**

### **Actividad 1**

Disponer de tiempo limitado para el desarrollo del proyecto.

### **¿Cómo corregir este efecto negativo?**

Dedicarse de tiempo completo a trabajar en el proyecto en caso que este tenga fines comerciales.

### **Actividad**

Disponer de un número de miembros limitados para integrar el equipo de trabajo.

### **¿Cómo corregir este efecto negativo?**

Elegir el número óptimo de personas que integren el equipo de trabajo en caso que el proyecto tenga fines comerciales.

### **Conclusión del reporte postmortem**

Este proyecto fue llevado a cabo como un desarrollo independiente, se aprovechó al máximo el tiempo y los recursos disponibles y a pesar de las limitaciones técnicas, económicas y de tiempo, a través del trabajo en equipo, de la disposición de colaborar de cada uno de los miembros y de resolver en conjunto cada una de los obstáculos que se presentaron permitieron que este proyecto culminara con éxito.

Al culminar este trabajo se sentó un precedente que permite retomar un proyecto de la misma envergadura basado en la experiencia y el aprendizaje de las diferentes técnicas en el desarrollo de videojuegos para lograr alcanzar un producto final de calidad.

## **VII. Conclusiones**

- Se elaboró un guión que comprende una historia completa para crear un juego basado en ciertos segmentos de esta historia.
- Se diseñaron múltiples ambientes correspondientes a cada nivel, ambiente de cárcel, zona amplia para combate, escenario de laboratorio y zona de campo abierto usando las herramientas del motor UDK.
- Se programaron mecánicas de juego basadas en el guión, a través del uso de programación orientada a objetos.

## **VIII. Recomendaciones**

- Se recomienda una ampliación del videojuego basada en nuevas ideas que enriquezcan la versión realizada.
- Se propone diseñar nuevos ambientes y escenarios.
- Se sugiere continuar profundizando en el estudio de la programación de videojuegos para ampliar las mecánicas de juego e incluirlas en futuras versiones.

## **IX. Bibliografía**

- Boreskov, A. (2013). *Computer Graphics: From Pixels to Programmable Graphics Hardware* (Computer Graphics, Geometric Modeling, and Animation Series). Oklahoma, USA: Chapman and Hall/CRC.
- Consalvo, M. (2006). *Console video games and global corporations: Creating a hybrid culture. Consalvo New Media Society.*
- De la Dueña, J. M. (2011). *Manual para la creación de videojuegos mediante el motor Unreal Development Kit.* Madrid, España: Universidad Carlos III de Madrid.
- Despain, W. (2009). *Writing for Video Game Genres: From FPS to RPG.* Minnesota, USA: AK Peters/CRC Press.
- Dominguez Marin, L. (2007). *Desarrollo de un prototipo de videojuego basado en el cuento "Dalia y Zazir".* Bogotá, Colombia.
- Donovan, T. (2010). *Replay: The History of Video Games.* United Kingdom: Yellow Ant Media Limited.
- Entertainment Software Rating Board. (01 de Enero de 2013). *About ESRB.* Obtenido de <http://www.esrb.org/about/index.jsp>
- Entertainment Software Rating Board. (2013). *ESRB Ratings Guide.* Obtenido de [http://www.esrb.org/ratings/ratings\\_guide.jsp](http://www.esrb.org/ratings/ratings_guide.jsp)
- Facultad de Informatica de Barcelona. (23 de Agosto de 2010). *Historias de los video juegos los inicios.* Obtenido de <http://www.fib.upc.edu/retro-informatica/historia/videojocs.html>
- Forster, W. (2011). *Game Machines 1972-2012 - The Encyclopedia of Consoles, Handhelds and Home Computers.* Nueva Orleans, USA: Enati Media; 2nd edition.
- Garcia, J. (27 de Septiembre de 2012). *Historia de los videojuegos.* Obtenido de <http://videojuegosjp.blogspot.com/>
- González, D. (2011). *Diseño de Videojuegos Da Forma a tus Sueños.* Mexico: Alfaomega.
- Gortler, S. J. (2012). *Foundations of 3D Computer Graphics.* Massachussets, USA: The MIT Press.
- Grimsdale, R. L. (2011). *Advances in Computer Graphics Hardware IV (Focus on Computer Graphics).* Delaware, USA: Springer.



- Herrera, D. R. (30 de Junio de 2010). *El éxito tiene muchos padres, y los videojuegos tienen mucho éxito*. Obtenido de <http://www.libertaddigital.com/opinion/historia/el-exito-tiene-muchos-padres-y-los-videojuegos-tienen-mucho-exito-1276237958.html>
- Johnson, B. (2006). *Xbox 360 For Dummies*. Florida, USA: For Dummies.
- Kent, S. (2001). *The Ultimate History of Video Games: From Pong to Pokemon--The Story Behind the Craze That Touched Our Lives and Changed the World*. Chicago, United States: Three Rivers Press.
- Kerlow, I. (2009). *The Art of 3D Computer Animation and Effects*. California, USA: Wiley.
- Landa, R. (2006). *2D: Visual Basics For Designers*. Texas, USA: Cengage Learning.
- Lueptow, R. M. (2007). *Graphics Concepts for Computer-Aided Design (2nd Edition)*. New Jersey, USA: Prentice Hall.
- Lutznzer8. (05 de Junio de 2013). *Historia de los Videojuegos (Mega Post)*. Obtenido de <http://www.3djuegos.com/comunidad-foros/tema/22030389/0/historia-de-los-videojuegos-mega-post/>
- Melissinos, C. (2012). *The Art of Video Games: From Pac-Man to Mass Effect*. Atlanta, USA: Welcome Books.
- Morales Gerardo, C. N. (2010). *Procesos para el desarrollo de videojuegos*. Mexico: CULCYT.
- O'Connor, J. (2010). *Mastering mental ray: Rendering Techniques for 3D and CAD Professionals*. Colorado, USA: Sybex.
- PC Gaming Alliance, s. (10 de Noviembre de 2012). *PC Gaming Alliance Releases To Members-Exclusive Reports Covering All Aspects of the Still-Dominant PC Gaming Industry*. Obtenido de <http://pcgamingalliance.org/press/details/pc-gaming-alliance-releases-two-member-exclusive-reports>
- PC World. (24 de Junio de 2013). *Una breve historia del ajedrez computarizado* . Obtenido de <http://www.pcworld.com.mx/Articulos/29444.htm>
- Petzold, C. (2007). *3D Programming for Windows: Three-Dimensional Graphics Programming for the Windows Presentation Foundation (Pro - Developer)*. Virginia, USA: Microsoft Press.
- Pressman, R. (2006). *Ingeniería del software: un enfoque práctico*. España: McGraw Hill.
- Sellers, J. (2001). *Arcade Fever The Fan's Guide To The Golden Age Of Video Games*. Philadelphia, USA: Running Press.
- Shreiner, D. (2013). *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.3 (8th Edition)*. Illinois, USA: Addison-Wesley Professional.

- Sommerville, I. (2005). *Ingeniería del Software*. Madrid, España: Pearson Educacion S.A.
- Velasco, J. J. (20 de Julio de 2011). *Historia de la tecnología: Nimrod, un videojuego de 1951*. Obtenido de <http://alt1040.com/2011/07/nimrod-un-videojuego-de-1951>
- Velasco, J. J. (29 de Julio de 2011). *Historia de la tecnología: Spacewar! el videojuego que nació en el MIT*. Obtenido de <http://alt1040.com/2011/07/spacewar-el-videojuego-que-nacio-en-el-mit>
- Winter, D. (5 de Junio de 2013). *Pong Story*. Obtenido de <http://www.pong-story.com>
- Wolf, M. J. (2007). *The Video Game Explosion: A History from PONG to PlayStation and Beyond*. Arizona, USA: Greenwood.
- Wong, W. (1972). *Principles of Two-Dimensional Design*. Oklahoma, USA: Wiley.
- Wright, A. (2010). *The Ultimate PS3(tm) Repair Guide*. Texas, USA: Longitudes, LLC.
- Zackariasson, P. (2012). *The Video Game Industry: Formation, Present State, and Future* (Routledge Studies in Innovation, Organization and Technology). Alabama, USA: Routledge.
- Zink, J. (2011). *Practical Rendering and Computation with Direct3D 11*. Oregon, USA: A K Peters/CRC Press.

## X. Anexos

### 10.1 Cotización del equipo



Calle Principal Altamira D'este No.589 Ferreteria Sinsa 25vrs arriba  
 Telefono:PBX (505) 2264-8800 - Email : ventas@comtech.com.ni  
 RUC No. J0310000000603 - www.comtech.com.ni

**No. Prof.** 90431

**Cliente:** JAEDDSON SANCHEZ

Atención:

Teléfono: 89559498

Email: jeannick2007@hotmail.com

Fecha 12/04/2013

Asesor Venta Karla Largaespada

Teléfono: 2264-8800 Ext. 7706

Celular 8999-6325

E-Mail klargaespada@comtech.com.ni

#	Código	Descripción	Cant	Precio	Total
1	00101-065	TARJETA MADRE INTEL DH67BL - MICRO ATX - LGA1155 - IH67 - CORE I3 - I5 - I7 - BOXDH67BLB3	1	US 110.00	US 110.00
2	00101-066	TARJETA MADRE INTEL DH61WW - MICRO ATX - LGA1155 SOCKET - IH61 - CORE I3 - I5 - I7 - BOXDH61WWB3	1	US 72.00	US 72.00
3	00101-122	TARJETA MADRE ASROCK H61M-DGS - S1155 - DDR3 - FSB1333 - MB-ASROCK-H61M-DGS	1	US 52.50	US 52.50
4	00201-191	MEMORIA KINGSTON 8GB 1333MHZ DDR3/KVR1333D3N9/8G	1	US 58.50	US 58.50
5	00301-062	PROCESADOR INTEL CORE I3 3220 3.30 GHZ LGA1155 / BX80637I33220	1	US 144.80	US 144.80
6	00301-047	PROCESADOR INTEL PENTIUM G860 3.0 GHZ - LGA1155 BX80623G860	1	US 77.78	US 77.78
7	03201-003	TARJETA DE VIDEO 1GB / GEFORCE 8400 GS 64B DDR3 PCIE / 01G-P3-1302-LR/KL	1	US 38.50	US 38.50
8	03201-068	TARJETA DE VIDEO MSI 4GB - DDR3 - GEFORCE - PCIE 2.0 / N630GT-MD4GD3	1	US 99.50	US 99.50
9	03201-069	TARJETA DE VIDEO MSI 2GB - DDR3 - GEFORCE - PCIE 2.0 / N610GT-MD2GD3/LP	1	US 66.50	US 66.50

Subtotal	US 720.08
Impuesto	US 108.01
<b>Total</b>	<b>US 828.09</b>

T/Cambio	24.6900
Condiciones	CONTADO
T/Entrega	INMEDIATA
Oferta Valida	8 dias
Garantía	
Comentarios:	

**COMTECH recomienda instalar licencia originales en sus Equipos.**

**Es valida solamente con el sello de la empresa**

## 10.2 Manual de usuario de videojuego Soul of wars

### Menú de inicio



En este menú de inicio se muestra el logotipo del videojuego soul of wars y a su vez se muestran dos opciones:

Iniciar juego: al dar clic en esta opción iniciara una nueva partida.

Salir: esta opción permite salir del juego y cerrar la aplicación.

### Menú de fin de juego



Este menú se carga al finalizar el juego y haber cumplido el objetivo de escapar de la prisión, nos presenta dos opciones:

Iniciar nueva partida: al dar clic en esta opción iniciara una nueva partida.

Salir: esta opción permite salir del juego y cerrar la aplicación.

## **Controles**

### **Controles de jugador:**

Tecla direccional arriba / W: avanzar en línea recta.

Tecla direccional abajo / S: retroceder en línea recta.

Tecla direccional izquierda / A: girar vista hacia el lado izquierdo.

Tecla direccional derecha / D: girar vista hacia el lado derecho.

Tecla direccional arriba / W + Tecla direccional izquierda / A: avanza en diagonal hacia el lado izquierdo.

Tecla direccional arriba / W + Tecla direccional derecha / D: avanza en diagonal hacia el lado derecho.

Tecla direccional abajo / S + Tecla direccional izquierda / A: retrocede en diagonal hacia el lado izquierdo.

Tecla direccional abajo / S + Tecla direccional derecha / D: retrocede en diagonal hacia el lado derecho.

Teclear dos veces continuas tecla direccional arriba: saltar hacia adelante.

Teclear dos veces continuas tecla direccional abajo: saltar hacia atrás.

Teclear dos veces continuas tecla direccional izquierda: saltar hacia la izquierda.

Teclear dos veces continuas tecla direccional derecha: saltar hacia la derecha.

Tecla espaciadora: saltar en el lugar.

Teclear dos veces continuas barra espaciadora: utilizando el objeto botas de salto, ejecuta un salto con gran altura.

Tecla C: agacharse

Tecla G: Seleccionar la mejor arma que el jugador porte en el momento.

Tecla Esc: pausar juego y mostrar menú de pausa.

Movimiento del mouse: enfoca la vista del jugador.

Clic izquierdo: disparo básico.

Clic derecho: disparo avanzado.

Botón scroll del mouse: cambiar arma.

## **Controles de vehículos**

### **Controles de vehículo Scorpion**



Tecla enter: entrar y salir de vehículo

Tecla direccional arriba / W: vehículo avanza en dirección al giro de llantas

Tecla direccional abajo / S: vehículo retrocede en línea recta.

Tecla A: girar llantas hacia el lado izquierdo.

Tecla D: girar llantas hacia el lado derecho.

Tecla direccional arriba / W + tecla A: vehículo avanza en diagonal hacia el lado izquierdo.

Tecla direccional arriba / W + Tecla D: vehículo avanza en diagonal hacia el lado derecho.

Tecla direccional abajo / S + Tecla A: retrocede en diagonal hacia el lado izquierdo.

Tecla direccional abajo / S + Tecla D: retrocede en diagonal hacia el lado derecho.

Teclear doblemente direccional arriba / W: Activará el turbo.

Movimiento del mouse: enfoca la vista del jugador.

Clic izquierdo: disparo de bomba.

Clic derecho: activar pinzas frontales de destrucción.

## **Controles de Nave manta**



Tecla direccional arriba / W: avanzar en línea recta.

Tecla direccional abajo / S: retroceder en línea recta.

Tecla direccional izquierda / A: girar vista hacia el lado izquierdo.

Tecla direccional derecha / D: girar vista hacia el lado derecho.

Tecla direccional arriba / W + Tecla direccional izquierda / A: avanza en diagonal hacia el lado izquierdo.

Tecla direccional arriba / W + Tecla direccional derecha / D: avanza en diagonal hacia el lado derecho.

Tecla direccional abajo / S + Tecla direccional izquierda / A: retrocede en diagonal hacia el lado izquierdo.

Tecla direccional abajo / S + Tecla direccional derecha / D: retrocede en diagonal hacia el lado derecho.

Tecla espaciadora: saltar en el lugar.

Clic izquierdo: disparo básico.

## **Controles de Nave Cicada**



Tecla direccional arriba / W: avanzar en línea recta.

Tecla direccional abajo / S: retroceder en línea recta.

Tecla direccional izquierda / A: girar vista hacia el lado izquierdo.

Tecla direccional derecha / D: girar vista hacia el lado derecho.

Tecla direccional arriba / W + Tecla direccional izquierda / A: avanza en diagonal hacia el lado izquierdo.

Tecla direccional arriba / W + Tecla direccional derecha / D: avanza en diagonal hacia el lado derecho.

Tecla direccional abajo / S + Tecla direccional izquierda / A: retrocede en diagonal hacia el lado izquierdo.

Tecla direccional abajo / S + Tecla direccional derecha / D: retrocede en diagonal hacia el lado derecho.

Tecla espaciadora: elevar nave.

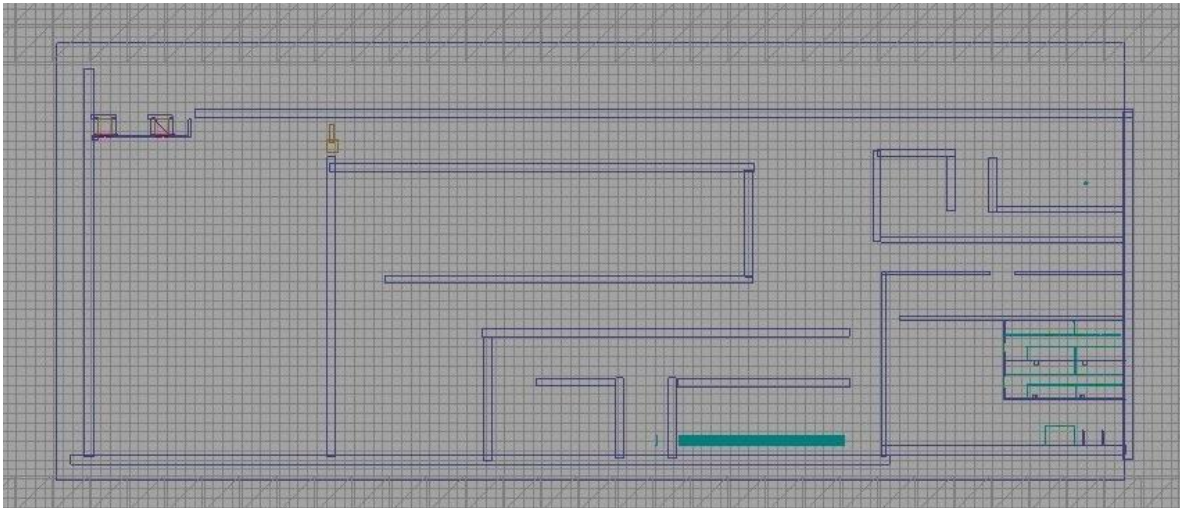
Clic izquierdo: disparo básico.

Clic derecho sostenido por varios segundos: carga y dispara múltiples proyectiles simultáneamente

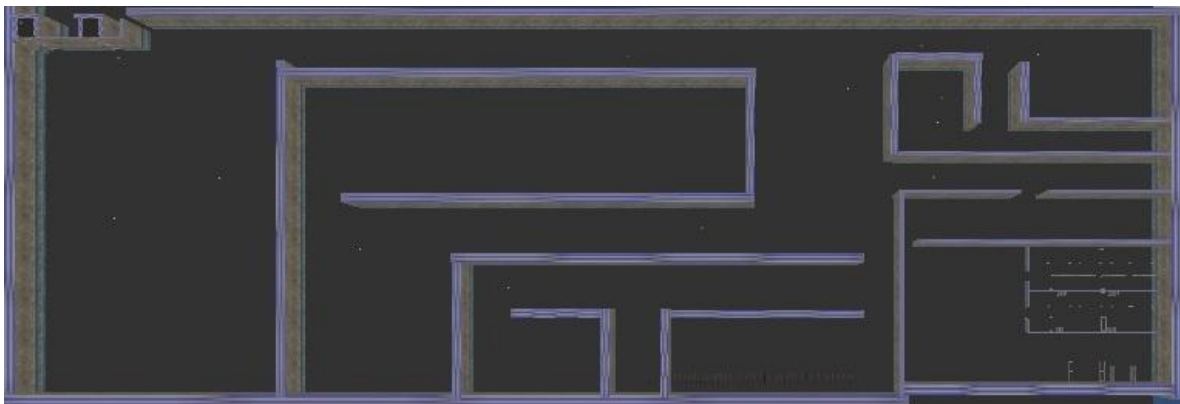


## Mapas

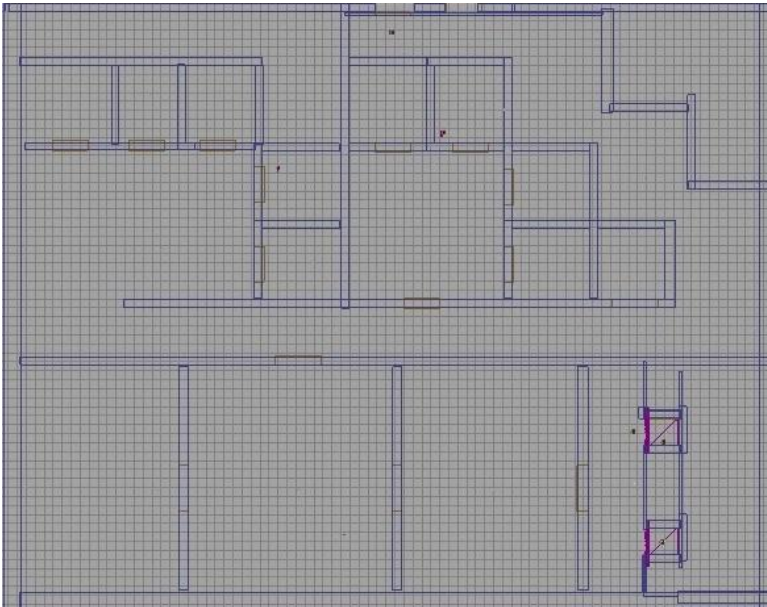
### Mapa de nivel 1: Cárcel vista plano



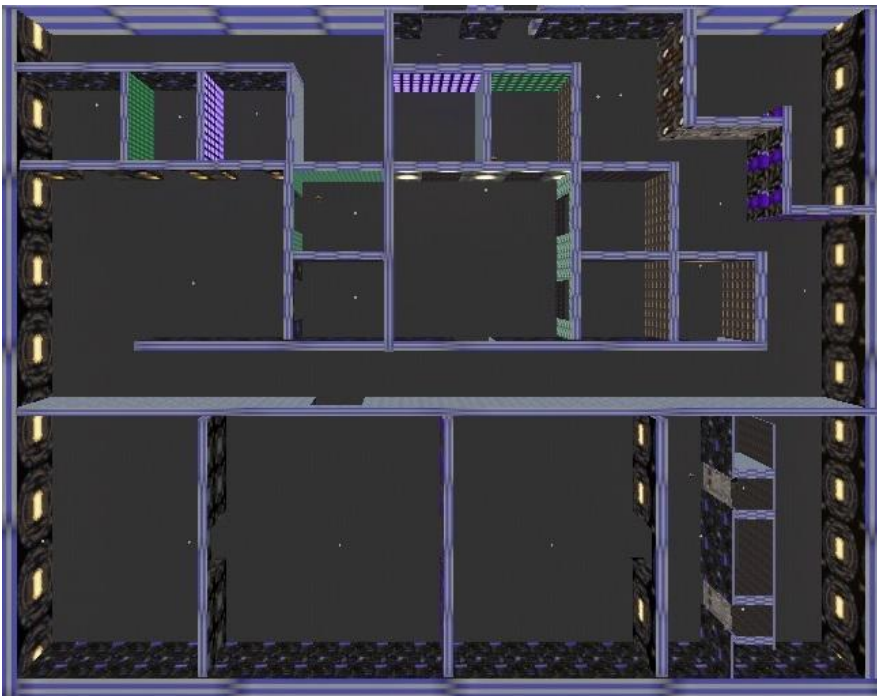
### Mapa de nivel 1: Cárcel vista modo juego



**Mapa del Nivel 2: Laboratorio vista plano**

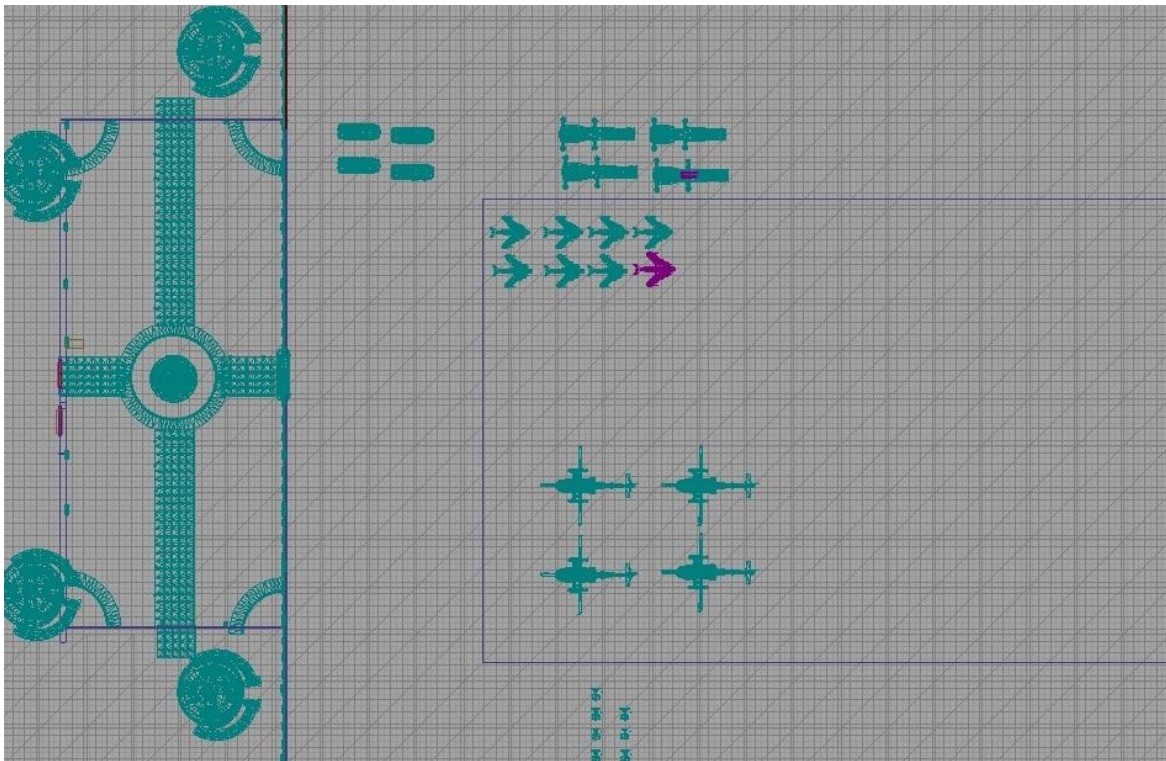


**Mapa del Nivel 2: Laboratorio vista modo juego**





Mapa del Nivel 3: Campo militar vista plano



Mapa del Nivel 3: Campo militar vista modo juego



## **Objetos**

### **Armas**

#### **Pistola Láser**



Arma básica utilizada por el jugador al inicio de la partida, es el arma de menor potencia en el juego, con el clic izquierdo dispara pequeñas emisiones laser que dañan muy parcialmente al enemigo y sosteniendo el clic derecho libera una ráfaga laser, la cual si es sostenida matará a un enemigo en unos instantes. Sus municiones se restauran automáticamente a medida que el tiempo avanza.

#### **Rifle de rayos plasma**

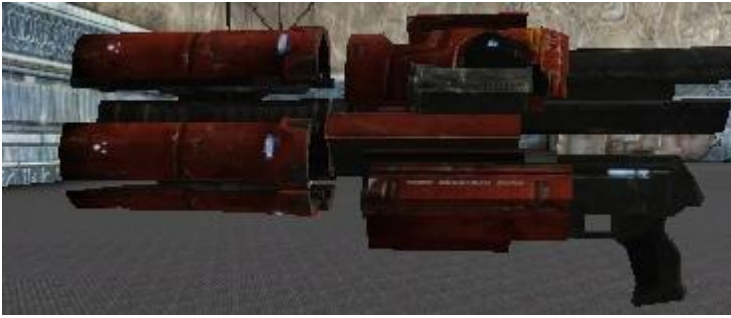


Arma de color purpura, de alta potencia, velocidad y precisión. Con el clic izquierdo del mouse libera un rápido, fino y potente disparo láser, el cual bastaría con tres disparos para aniquilar a un enemigo común.

Con el clic derecho, liberara una bola de energía de muy alta potencia pero de menor velocidad y mayor rango de impacto, bastarían dos de ellas para aniquilar a un enemigo común.

Si se arroja una bola de energía de plasma con el clic derecho y acto seguido a esta bola de plasma se le impacta con un disparo del clic izquierdo, producirá una gran detonación capaz de dañar a los enemigos alrededor, bastaría un impacto de esta detonación para aniquilar a un enemigo común.

## **Triple Cañón Lanzacohetes**



Es el arma más potente de todo el juego, pero tiene como desventaja que la velocidad de su disparo es bastante lenta.

Con el clic izquierdo liberará un potente disparo que aniquilaría a un enemigo común al instante, con el clic derecho sostenido cargara el triple cañón, cargando un proyectil por segundo, si se suelta el clic antes que se cargue el tercer proyectil, disparará el numero de proyectiles que se le haiga permitido cargar hasta ese momento, en caso de sostener el clic hasta que se carguen los tres proyectiles automáticamente los disparará de manera simultánea, aumentando la probabilidad de impactar múltiples enemigos.

## **Municiones**

### **Munición Láser**



Paquete de municiones para la pistola láser, aumentara 50 unidades a las municiones que se posean en ese momento

### **Munición Plasma**



Paquete de municiones para el rifle de rayos plasma, aumentara 10 unidades a las municiones que se posean en ese momento.

### **Munición Lanza Cohetes**



Paquete de municiones para el Triple Cañón Lanzacohetes, aumentara 10 unidades a las municiones que se posean en ese momento.

### **Objetos que recuperan y aumentan el porcentaje de Salud**

#### **Paquete básico de Salud**



Aumentará la salud en un 5%, reaparece unos momentos después de ser consumido.



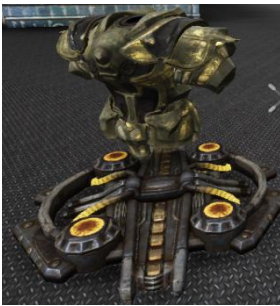
### **Súper Paquete de Salud**



Al tocarlo aumentará 100% el porcentaje de salud que se porte en ese instante, tardará mucho más tiempo en reaparecer que el paquete básico de vida.

### **Armaduras**

#### **Armadura para Dorso**



Aumentará el porcentaje de resistencia de la armadura en un 50%, tardara algunos minutos en reaparecer nuevamente.

### **Armadura para piernas**



Aumentará el porcentaje de resistencia de la armadura en un 30%, tardará algunos minutos en reaparecer nuevamente.

### **Cinturón de Poder**



Aumentará el porcentaje de resistencia de la armadura en un 100%, tardará mucho más tiempo en reaparecer nuevamente que el resto de armaduras.



## Vehículos

### Vehículo de terreno Scorpion



Vehículo de cuatro ruedas muy potente que a medida que avanza arrolla todo a su paso, ya sean enemigos u objetos.

Presionando rápidamente la tecla direccional arriba o la tecla w, activara el turbo por un periodo breve de tiempo, luego necesitara un tiempo más para recuperarse y volver a utilizar el turbo.

Utilizando el botón izquierdo del mouse, disparará una potente bomba que daña seriamente a un enemigo común.

Manteniendo presionado el botón derecho del mouse, liberara unas pinzas frontales que serán capaces de arrastrar a los enemigos que arrolle a su paso, si estas pinzas se golpean en alguna pared u objeto muy sólido se romperán.

### Nave de combate Manta



Ligera y veloz nave de combate, la cual se suspende a unos cuantos metros del suelo, dispara proyectiles de mediana potencia a muy alta velocidad.

### **Nave de combate Cicada**



Grande, potente y pesada nave de combate, la cual puede elevarse a gran altura y disparar muy potentes proyectiles.

Con el clic izquierdo liberará un potente disparo, con el clic derecho sostenido cargará el cañón múltiple que cuenta con dieciséis proyectiles, cargando un proyectil por segundo, si se suelta el clic antes que se carguen todos los proyectiles, se disparará el número de proyectiles que se le haiga permitido cargar hasta ese momento, en caso de sostener el clic hasta que se carguen los dieciséis proyectiles automáticamente los disparará de manera simultánea.