



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

Posgrado en Ciencia e Ingeniería de la Computación

Propuesta de un modelo de procesos para el desarrollo de videojuegos independientes

Tesis

Que para optar por el grado de:
Maestro en Ingeniería (Computación)

Presenta:

Ing. Edgar Ignacio Peña Flores

Directora de tesis:
M. en C. María Guadalupe Elena Ibarquengoitia González
Facultad de Ciencias

Ciudad Universitaria, Cd.Mx. julio 2016



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

DEDICATORIA

A mis padres:

Por su gran apoyo, cariño y por ser un ejemplo constante de esfuerzo y superación.

AGRADECIMIENTOS

A la *M. en C. María Guadalupe Elena Ibargüengoitia González* por su paciencia, confianza, guía y apoyo para realizar esta tesis.

A la *Dra. Hanna Oktaba*, por sus consejos, paciencia y apoyo durante toda la maestría.

A los miembros del jurado: *M. en TI. Elsa Ramírez Hernández*, *Dra. Sylviane Francoise Levy Amselle* y *M. en C. Gustavo Arturo Márquez Flores*, por enriquecer este trabajo con sus comentarios y correcciones.

A *Jorge Garrido* y *César Arias* por su apoyo y colaboración en la validación de la propuesta realizada en el presente trabajo.

Al personal académico y administrativo del *Posgrado en Ciencia e Ingeniería de la Computación* de la *UNAM* por su apoyo durante mis estudios en esta gran institución.

A *Germán Strempler* por su amistad y apoyo en momentos difíciles.

Finalmente, al *Consejo Nacional de Ciencia y Tecnología (CONACYT)* por otorgarme una beca para realizar mis estudios de posgrado.

A todos... muchas gracias.

ÍNDICE

INTRODUCCIÓN	1
1. CAPÍTULO 1. VIDEOJUEGOS INDEPENDIENTES	5
1.1. ¿Qué es un videojuego?	5
1.2. Videojuegos Independientes.....	6
1.3. Estado de la industria de los videojuegos en México	7
1.4. Resultados del sondeo aplicado a los estudios mexicanos.....	8
1.5. Análisis de resultados y conclusiones	13
2. CAPÍTULO 2. EL ESTÁNDAR ISO/IEC 29110 PERFIL BÁSICO	15
2.1. Proceso de software.....	15
2.2. Modelo de proceso de software	16
2.3. Visión general del estándar ISO/IEC 29110 Perfil Básico	18
2.4. Descripción del estándar ISO/IEC 29110 Perfil Básico	20
2.5. Proceso de Administración del Proyecto en el Perfil Básico.....	21
2.6. Proceso de Implementación del Software en el Perfil Básico.....	24
3. CAPÍTULO 3. CARACTERÍSTICAS BÁSICAS EN EL DESARROLLO DE VIDEOJUEGOS	27
3.1. Diferencias en el proceso de desarrollo de videojuegos	27
3.2. Principales dificultades en el desarrollo de videojuegos	29
3.3. Características básicas en el desarrollo de videojuegos	30
3.4. Trabajos relacionados y procesos de desarrollo de videojuegos.....	32
3.4.1. Procesos de desarrollo de videojuegos generales	32
3.4.2. Procesos ágiles de desarrollo de videojuegos.....	36
3.4.3. Procesos de desarrollo de videojuegos centrados en el jugador.....	38
3.4.4. Modelo de referencia.....	39
3.4.5. Proceso de desarrollo de videojuegos basado en ISO/IEC 29110.....	40
3.5. Fases, actividades y productos del modelo de procesos propuesto.....	42

4. CAPÍTULO 4. INTEGRACIÓN DEL MODELO DE PROCESOS PARA EL DESARROLLO DE VIDEOJUEGOS INDEPENDIENTES	44
4.1. Introducción	44
4.2. Metodología para la integración del modelo de procesos	44
4.3. Definición de características básicas, fases, actividades y dificultades que deberá cubrir el modelo de procesos.	45
4.4. Adaptación del Perfil Básico al desarrollo de videojuegos independientes	47
4.4.1. Diagrama general del modelo de procesos propuesto	47
4.5. Proceso de Administración del Proyecto	49
4.5.1. Actividad Inicio	52
4.5.2. Actividad Planeación del Proyecto	53
4.5.3. Actividad Ejecución del Plan del Proyecto	55
4.5.4. Actividad Evaluación y Control del Proyecto.....	56
4.5.5. Actividad Cierre del Proyecto.....	57
4.6. Proceso de Implementación del Videojuego	59
4.6.1. Fase de Concepto	61
4.6.2. Actividad Concepto del Videojuego	61
4.6.3. Actividad Análisis de Mercado y Financiamiento.....	65
4.6.4. Fase de Preproducción	69
4.6.5. Actividad Requerimientos y Diseño del Videojuego	69
4.6.6. Actividad Diseño Técnico del Videojuego	80
4.6.7. Fase de Producción	86
4.6.8. Actividad Construcción de Software	86
4.6.9. Actividad Integración y Pruebas de Software	88
4.6.10. Fase de Posproducción.....	92
4.6.11. Actividad Entrega del Producto.....	92
4.6.12. Actividad Mantenimiento del Videojuego	95
4.7. Roles involucrados	97

5. CAPÍTULO 5. VALIDACIÓN DEL MODELO DE PROCESOS	98
5.1. Revisión y evaluación por experto práctico	98
5.1.1. Objetivo	98
5.1.2. Metodología	98
5.1.3. Cuestionarios de validación aplicados a expertos prácticos	99
5.1.4. Validación del primer experto	99
5.1.5. Validación del segundo experto	100
5.2. Retroalimentación	103
5.2.1. Análisis de los cuestionarios de validación	103
5.2.2. Propuestas y ajustes del modelo de procesos	105
CONCLUSIONES Y TRABAJO A FUTURO	107
ANEXOS	109
ANEXO A: SONDEO REALIZADO A LOS ESTUDIOS DE DESARROLLO DE VIDEOJUEGOS INDEPENDIENTES EN MÉXICO	109
ANEXO B: GUÍA DE IMPLEMENTACIÓN DEL MODELO DE PROCESOS PARA EL DESARROLLO DE VIDEOJUEGOS INDEPENDIENTES	118
Introducción	118
Visión General	118
Roles involucrados	120
Organización del equipo	124
PROCESO DE ADMINISTRACIÓN DEL PROYECTO (AP)	125
AP.1 - Inicio	128
AP.2 - Planeación del Proyecto	129
AP.3 - Ejecución del Plan del Proyecto	131
AP.4 - Evaluación y Control del Proyecto	133
AP.5 - Cierre del Proyecto	134

PROCESO DE IMPLEMENTACIÓN DEL VIDEOJUEGO (IV).....	136
IV. Fase de Concepto.....	138
IV.1 - Concepto del Videojuego.....	138
IV.2 - Análisis de Mercado y Financiamiento	143
IV. Fase de Preproducción.....	145
IV.3 - Requerimientos y Diseño del Videojuego.....	145
IV.4 - Diseño Técnico del Videojuego.....	154
IV. Fase de Producción	158
IV.5 - Construcción de Software.....	158
IV.6 - Integración y Pruebas de Software.....	160
IV. Fase de Posproducción	164
IV.7 - Entrega del Producto	164
IV.8 - Mantenimiento del Videojuego.....	165
Descripción de productos	167
REFERENCIAS.....	179

INTRODUCCIÓN

Antecedentes

La industria de los videojuegos alcanza ya una dimensión de mercado superior a los 74 mil millones de dólares a nivel global [1]. Desde hace años superó a la industria cinematográfica y con tendencias a desmarcarse con mucha facilidad y rapidez.

El surgimiento de nuevos esquemas de mercado y dispositivos que facilitan el acceso para una mayor cantidad de consumidores han dado pie a la formación de un mercado maduro.

El rápido crecimiento de este mercado se debe (entre otras razones) al surgimiento de nuevos modelos de comercialización por medio de la descarga de juegos a través de Internet en todo tipo de consolas, y el incremento en el uso de aplicaciones en dispositivos móviles [2].

El desarrollo de software para la industria de los videojuegos está evolucionando rápidamente y se está volviendo cada vez más compleja.

Desarrollar videojuegos es una tarea complicada que involucra la experiencia de profesionales de varias disciplinas incluyendo ciencias de la computación, arte, diseño y negocios [3]. La presión que existe en el desarrollo de videojuegos para llegar al mercado tan pronto como sea posible complica la planeación de los tiempos de desarrollo. Cuestiones de ingeniería de software clásicas asociadas con el desarrollo de videojuegos puede incluir administración de requerimientos, administración de la configuración, verificación y validación.

Aunque se han incorporado prácticas del desarrollo de software por los desarrolladores de videojuegos, una diferencia fundamental es que un videojuego tiene como objetivo proporcionar una experiencia lúdica, esto es, la industria de los videojuegos tiene énfasis en la experiencia de usuario y la usabilidad. Esto puede causar una divergencia en las prácticas típicas de la ingeniería de software.

Situación actual de la industria de los videojuegos en México

Año con año, el mercado de los videojuegos en México ha cautivado a más jugadores, lo que ha propiciado que cada vez más empresas, desarrolladores y proveedores miren hacia esta industria.

México es un gran consumidor de videojuegos, sin embargo no ocupa un lugar relevante como productor de videojuegos [4].

La industria de desarrollo de videojuegos en México se encuentra actualmente minimizada debido a la falta de incentivos para realizar este tipo de actividad en un esquema empresarial [5]. Existe una alta proporción de empleados dedicados a la creación de videojuegos bajo un esquema *independiente*, es decir sin apoyo financiero de distribuidores.

En México, una tercera parte de los desarrolladores de videojuegos laboran de modo *independiente* (principalmente generando contenido gratuito) y solamente el 2% de los desarrolladores emprendieron su propio negocio. Esto demuestra que una gran proporción de esta mano de obra se encuentra deslindada de los grandes corporativos [5].

Es importante mencionar que 4 de cada 10 desarrolladores cuentan con un trabajo adicional para solventar sus necesidades debido a que no cuentan con una remuneración. La falta de incentivos da cabida a una escasez de mano de obra calificada para participar en el desarrollo de videojuegos.

En esta joven industria mexicana hay una relación entre la falta de madurez y la formación que se ofrece para quienes desean incursionar en ella. La formación académica es un elemento básico para promover la competitividad y la productividad, sobretodo en una industria tan diversa y dinámica como la de los videojuegos.

Sin embargo, las oportunidades de crecimiento para el mercado de los videojuegos en México son altas, gracias a que las aplicaciones para dispositivos móviles y los juegos en línea permiten tener un mayor acceso a programadores o pequeños estudios.

De este modo, se presenta en México un entorno óptimo para impulsar e innovar la labor del desarrollo de videojuegos, así como la incorporación en distintas plataformas [2].

Objetivo

El objetivo de esta investigación es desarrollar una propuesta de un modelo de procesos para el desarrollo de videojuegos independientes, que proporcione guía para planificar, gestionar y ejecutar la construcción de este tipo de videojuegos tomando como base el estándar *ISO/IEC 29110*.

Metodología

Para realizar la propuesta del modelo de procesos para el desarrollo de videojuegos independientes se llevarán a cabo las siguientes actividades:

- 1. Investigar el estado del arte de la industria de los videojuegos*, en particular la Ingeniería de Software aplicada a su desarrollo. Esta actividad se realiza mediante una Revisión Sistemática de Literatura y se documentará la investigación sobre el estado del arte.
- 2. Investigar el estado del arte de la industria de los videojuegos independientes en México*. Esta actividad se realiza mediante un sondeo que se aplicará a los estudios independientes mexicanos y se asistirá a eventos especializados sobre el desarrollo de videojuegos en México.
- 3. Investigar trabajos relacionados sobre el desarrollo de videojuegos independientes*. Esta actividad se realiza mediante una Revisión Sistemática de Literatura y se documentará la investigación.

4. *Construir un modelo de procesos para el desarrollo de videojuegos independientes.* Se realizará con base en las necesidades y características de la industria mexicana, considerando el estado del arte y utilizando como marco el ISO/IEC 29110 Perfil Básico.

5. *Validar el modelo de procesos propuesto.* Obtener la retroalimentación a través de la validación por expertos prácticos dedicados al desarrollo de videojuegos independientes.

6. *Analizar y ajustar el modelo de procesos considerando los resultados de validación.* Se documentará el modelo de procesos considerando los resultados de validación.

Relevancia y contribución

La industria de los videojuegos en México aún es joven y hay pocos estudios sobre el desarrollo de videojuegos.

Contar con un modelo de procesos que se enfoque a las particularidades del desarrollo de videojuegos independientes, contribuiría a minimizar las principales dificultades que se distinguen en este tipo de desarrollo de software y mejorar la calidad en los videojuegos desarrollados.

Además este modelo de procesos aceleraría el conocimiento técnico para poder comenzar a crear más estudios de desarrollo de videojuegos en México.

Gran parte de los estudios de desarrollo de videojuegos son pequeñas y medianas empresas que buscan diversos modelos de procesos que se puedan adoptar a sus necesidades en el desarrollo de videojuegos.

Con la publicación del estándar ISO/IEC 29110 Perfil Básico se tiene un marco de trabajo que va dirigido a pequeñas entidades para el desarrollo de software [6] y con la integración de un modelo de procesos enfocado en las particularidades del desarrollo de videojuegos independientes se podría obtener el beneficio de una adopción adecuada y sin tantas dificultades para los estudios independientes.

Estructura de la tesis

El presente trabajo de investigación se encuentra estructurado por los siguientes capítulos:

El capítulo 1 *Videojuegos Independientes* introduce el concepto de videojuegos independientes y se analiza el estado de la industria de los videojuegos en México.

En el capítulo 2 se hace una breve descripción sobre el estándar ISO/IEC 29110 Perfil Básico, que se toma como referencia para la integración del modelo de procesos para el desarrollo de videojuegos independientes.

En el capítulo 3 se identifican las principales diferencias, dificultades y características en el proceso de desarrollo de videojuegos, como resultado de la evaluación de los trabajos relacionados y propuestas enfocadas en el desarrollo de videojuegos. Finalmente se concluye el capítulo con la definición de las fases, actividades y productos clave que tendrá la propuesta del modelo de procesos para el desarrollo de videojuegos independientes.

La integración de la propuesta del modelo de procesos se describe en el capítulo 4 *Integración del modelo de procesos para el desarrollo de videojuegos independientes*.

El capítulo 5 *Validación del modelo de procesos* presenta los resultados y retroalimentación obtenida tras la validación del modelo propuesto por parte de expertos prácticos.

Finalmente se presentan las conclusiones obtenidas a lo largo de la investigación y el trabajo a futuro que puede generar.

1. CAPÍTULO 1. VIDEOJUEGOS INDEPENDIENTES

1.1. ¿Qué es un videojuego?

Un videojuego es un artefacto en un medio digital audiovisual, se propone como un objeto de entretenimiento, e intenta proporcionar tal entretenimiento a través del empleo de reglas y objetivos de jugabilidad (*gameplay*) o ficción interactiva [7].

Los videojuegos son sistemas complejos e interactivos que requieren una investigación significativa en gráficas en tiempo real, inteligencia artificial y sistemas distribuidos. Además, los videojuegos son software de entretenimiento que aplican múltiples modos (gráficas, audio, interacción, historia y juego colaborativo) con el objetivo de sumergir al usuario en experiencias interactivas específicamente combinando estos modos [8].

Los videojuegos combinan partes más pequeñas en un sistema complejo. La historia del juego dirige al juego a través de episodios individuales, efectos visuales y de sonido incluyendo mecanismos complejos de interacción, motores de juego para propósitos de control de juego y componentes de hardware. Los elementos principales de un videojuego son los siguientes [31]:

1. **Mecánicas.** Son los procedimientos y reglas del juego. Las mecánicas describen las metas del juego, cómo los jugadores pueden y no pueden intentar lograrlas, y lo que pasa cuando lo intentan. Las mecánicas son cruciales para la jugabilidad (*gameplay*).
2. **Historia.** Esta es la secuencia de eventos que ocurren en el videojuego. Podría ser lineal, pre-guiada, o podría ser derivada y emergente. Cuando se tiene una historia que contar a través del juego, se deben elegir las mecánicas que refuercen esa historia y dejen que emerja.
3. **Estética (Aesthetics).** Esto es cómo se ve el juego, suena y se siente. La estética del juego es muy importante ya que tiene la relación más directa con la experiencia del jugador. Se deberán elegir las mecánicas que hagan sentir al jugador inmerso en el mundo que la estética ha definido. Así como la tecnología que pueda soportarla.
4. **Tecnología.** La tecnología que se elija para el juego permite que se hagan ciertas cosas y prohíbe hacer otras. La tecnología es esencialmente el medio en el cual la estética tiene lugar, en la cual las mecánicas ocurrirán, y a través de la cual será contada la historia.

Ninguno de los cuatro elementos es más importante que el otro. Todos son esenciales, no importa el juego que se esté diseñando. Cada elemento se considera de forma separada, y después todos juntos como un todo [31].

1.2. Videojuegos Independientes

La definición de videojuegos independientes (*indie games*) no es exacta, y tanto desarrolladores como jugadores discuten periódicamente sobre lo que ellos creen que la palabra "independiente" significa. Si bien los videojuegos independientes son aquellos que han sido creados sin el respaldo de un *publisher*, el término "independiente" ha evolucionado aún más y ahora generalmente se refiere a un videojuego que ha sido desarrollado con un cierto "espíritu independiente" por un equipo pequeño o un individuo [9].

Los videojuegos independientes pueden ser experimentales, emotivos, nostálgicos, y ocasionalmente simplemente extraños [9]. Al no haber restricciones creativas generalmente se enfocan en la innovación, creatividad, y experimentación artística (FIGURA 1.1).

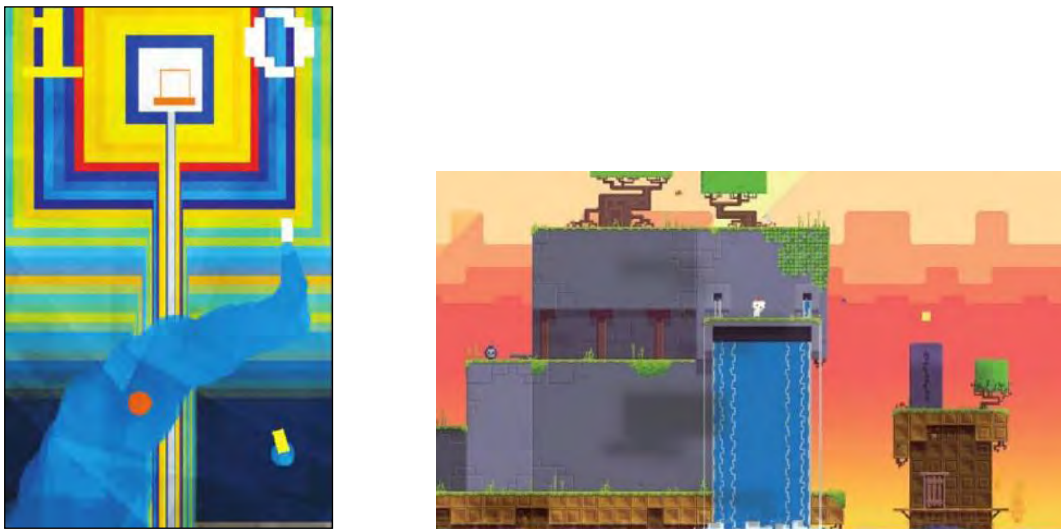


FIGURA 1.1 - Ejemplos de videojuegos independientes (Jetpack Basketball y Fez)

Este tipo de videojuegos comparten ciertas características: son desarrollados por individuos o pequeños grupos, son de corto alcance (si se compara con los videojuegos desarrollados por grandes compañías con grandes presupuestos), los estudios o desarrolladores independientes a veces no cuentan con apoyo financiero de *publishers* y usualmente tienen un presupuesto limitado o nulo para crearlos, sin embargo el dinero y la mano de obra reducida no representa un límite para la imaginación [10].

Los juegos independientes han ido cobrando mucha importancia en los catálogos de las tiendas de aplicaciones. La *App Store* de Apple abre sus puertas a más de 500 videojuegos cada día. Por supuesto, su calidad varía ampliamente, pero una gran parte de esos videojuegos provienen de desarrolladores o estudios independientes [10].

Al final de la primera década del siglo XXI se vio un *boom* en el número de desarrolladores independientes y en el recibimiento de la escena independiente. El número de herramientas y plataformas ha crecido, adicionalmente, las herramientas han mejorado y son más fáciles de utilizar, permitiendo que más gente haga videojuegos [10].

Los videojuegos independientes traen de regreso ideas frescas a la industria de los videojuegos que se últimamente se ha enfocado enteramente en las ventas [10].

1.3. Estado de la industria de los videojuegos en México

Año con año, el mercado de los videojuegos en México ha cautivado a más jugadores, lo que ha propiciado que cada vez más empresas, desarrolladores y proveedores miren hacia esta industria. México es un gran consumidor de videojuegos, sin embargo no ocupa un lugar relevante como productor de videojuegos [5].

Para conocer el estado de la industria de los videojuegos en México, se realizó un sondeo a diferentes estudios de desarrollo de videojuegos, con el objetivo de conocer su nivel de madurez, herramientas de desarrollo y prácticas que llevan a cabo para desarrollar videojuegos.

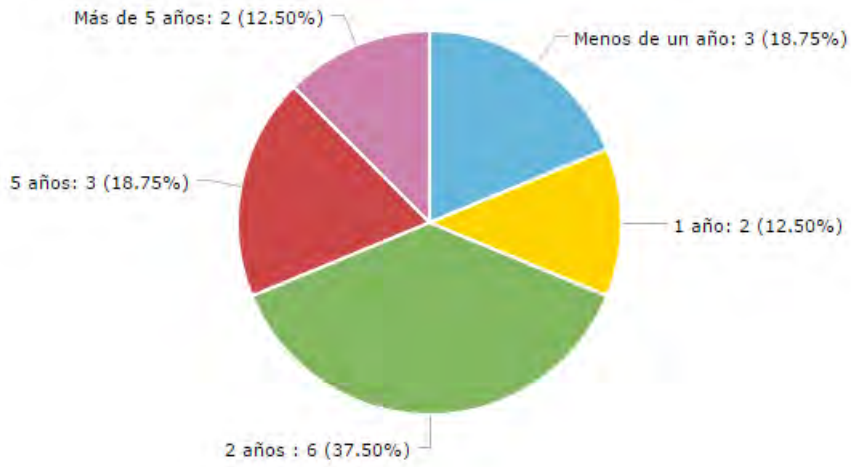
El sondeo se realizó a través de un formulario en línea (*Google Forms*). Se realizaron 19 preguntas agrupadas en las siguientes categorías:

1. Información general del estudio
2. Desarrollo (programación)
3. Administración de proyectos
4. Procesos de software
5. Pruebas
6. Marketing
7. Ventas
8. Dificultades y tipos de apoyo que requieren

En el sondeo participaron 16 estudios activos. A continuación se muestran los resultados más importantes.

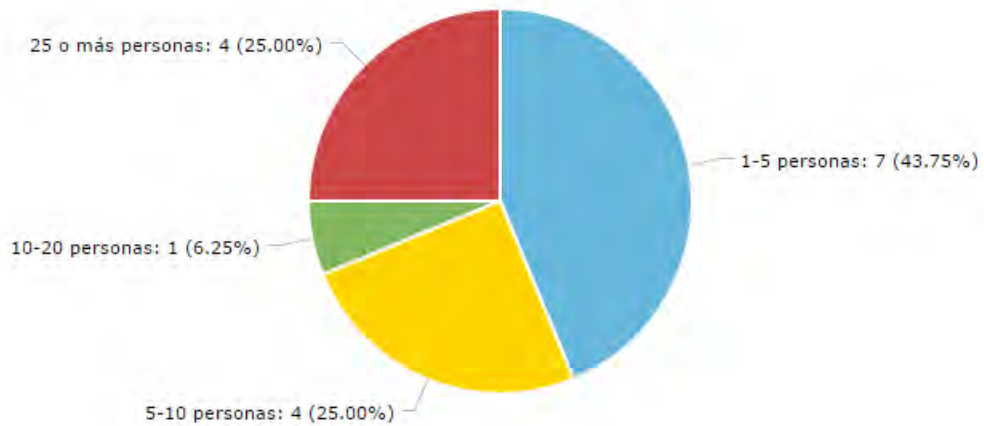
1.4. Resultados del sondeo aplicado a los estudios mexicanos

1. Tiempo desarrollando videojuegos



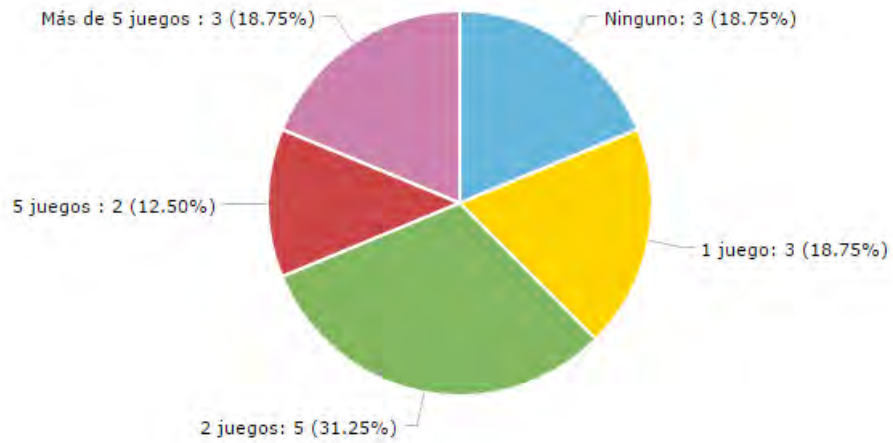
GRAFICA 1.1 - Tiempo desarrollando videojuegos

2. Número de personas que trabajan en el estudio



GRAFICA 1.2 - Número de personas que trabajan en el estudio

3. Número de videojuegos desarrollados



GRAFICA 1.3 - Número de videojuegos que han desarrollado

4. Estructura jerárquica común en los estudios de desarrollo de videojuegos

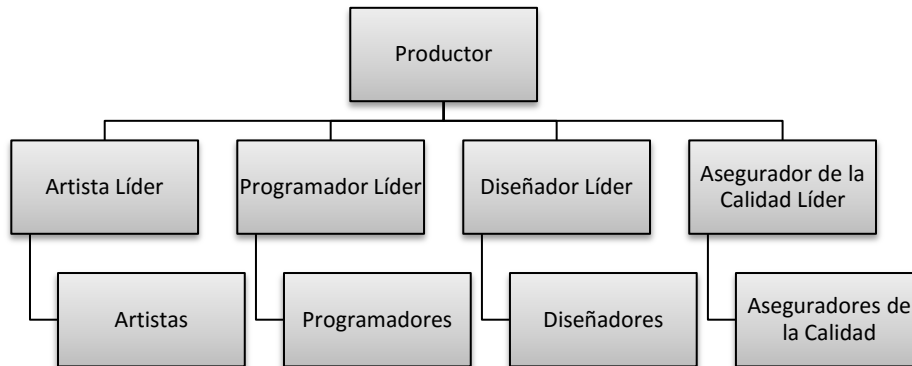
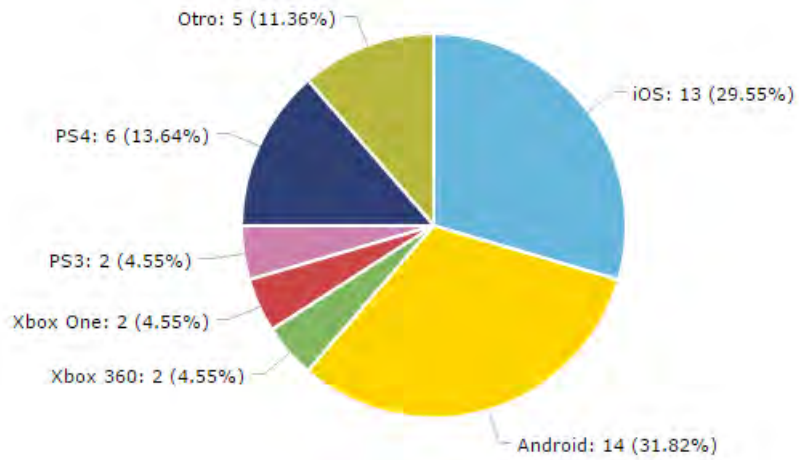


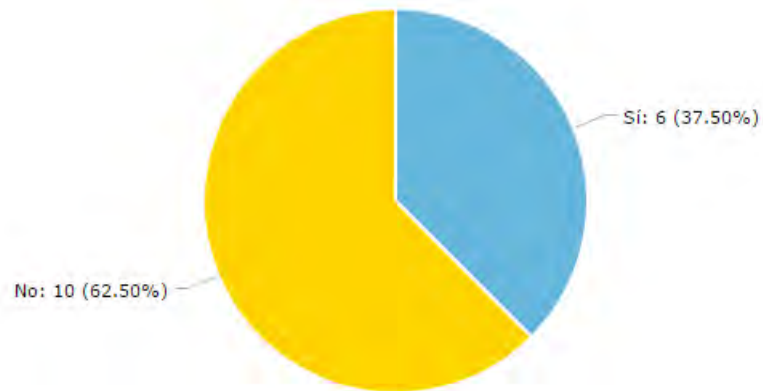
FIGURA 1.2 - Estructura jerárquica común en los estudios mexicanos

5. Principales plataformas de desarrollo



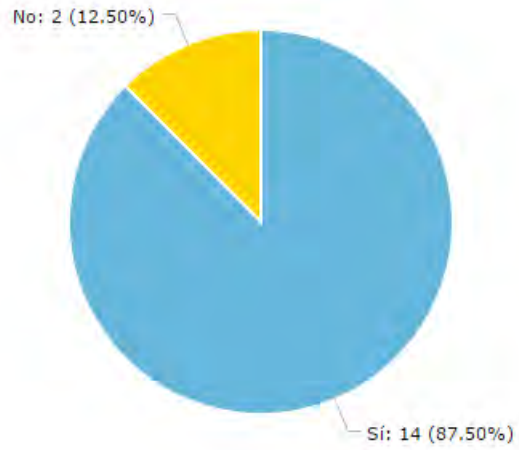
GRAFICA 1.4 - Principales plataformas de desarrollo

6. ¿Tienen implementado algún modelo de procesos para el desarrollo de videojuegos?



GRAFICA 1.5 - Implementación de modelos de procesos de software

7. ¿Llevan la administración de los proyectos que realizan?



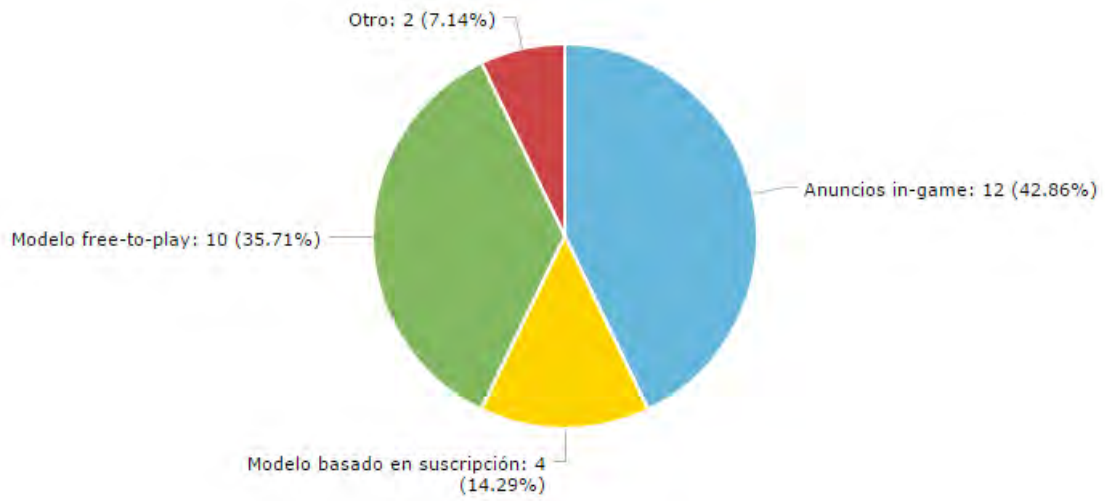
GRAFICA 1.6 - Administración de proyectos

8. ¿Realizan algún estudio de mercado antes de cada desarrollo?



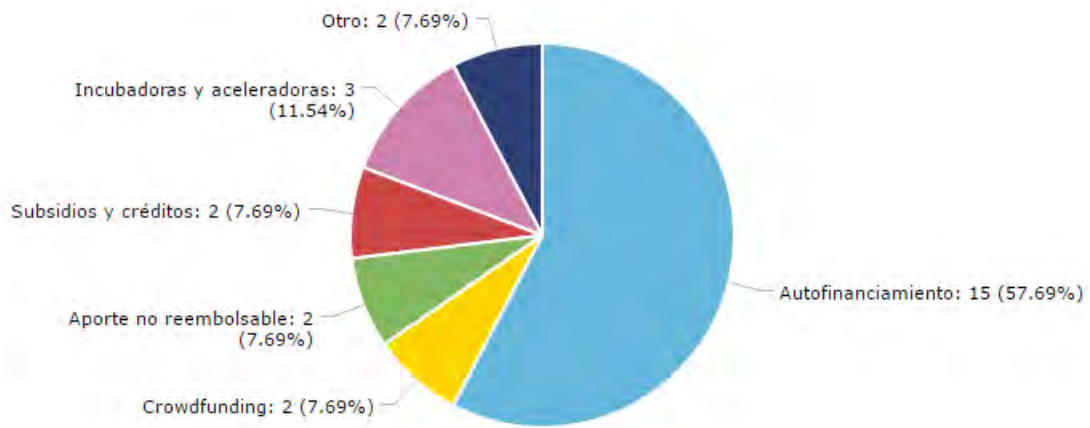
GRAFICA 1.7 - Estudio de mercado

9. Principales modelos de monetización



GRAFICA 1.8 - Principales modelos de monetización utilizados

10. Principales formas de financiamiento



GRAFICA 1.9 - Principales modelos de financiamiento utilizados

11. Principales dificultades que tienen en el desarrollo de videojuegos

TABLA 1.1 - Principales dificultades que tienen los estudios durante el desarrollo de videojuegos

PRINCIPALES DIFICULTADES
Financiamiento de proyectos (8)
Capacitación técnica (7)
Tiempo para realizar los proyectos (4)
Organización del equipo (4)
Poca experiencia en la industria (4)
Adquisición de equipo, software y licencias (2)
Estrategias de marketing (1)
Pipeline de producción (1)
Desarrollo de herramientas (1)

12. ¿Qué tipo de apoyos facilitarían el desarrollo de sus videojuegos?

TABLA 1.2 - Principales apoyos que facilitarían el desarrollo de videojuegos

PRINCIPALES APOYOS QUE FACILITARÍAN EL DESARROLLO DE VIDEOJUEGOS
Financiamiento (12)
Capacitación (7)
Eventos dedicados al desarrollo de videojuegos (4)
Apoyo para adquirir licencias de software y equipo especializado (2)
Incentivos fiscales (1)
Estrategias de mercadotecnia (1)
Apoyo para participar en eventos internacionales (1)

1.5. Análisis de resultados y conclusiones

En el estudio se puede observar que la industria de desarrollo de videojuegos en México es joven, la mayoría de los estudios encuestados tienen menos de 5 años en la industria y son estudios pequeños (menos de 10 personas). La estructura jerárquica común de los estudios se puede observar en la FIGURA 1.2.

La mayoría de los estudios ha publicado menos de 5 videojuegos, siendo las plataformas móviles (Android, iOS) las preferidas para publicar sus videojuegos y utilizando un modelo de monetización basado en anuncios dentro del videojuego y *free-to-play*.

Se puede observar que la gran mayoría de estudios no tienen implementado un modelo de procesos de desarrollo de videojuegos y no tienden a seguir ningún método de desarrollo sistemático, trabajando principalmente *ad-hoc*.

Entre las principales dificultades que tienen los estudios se encuentra la falta de financiamiento para concluir sus proyectos y la dificultad para adquirir equipo especializado (software y hardware), así como la falta de capacitación, organización y estrategias de mercadotecnia.

La industria de desarrollo de videojuegos en México se encuentra actualmente minimizada debido a la falta de incentivos para realizar este tipo de actividad en un esquema empresarial.

En esta joven industria mexicana hay una relación entre la falta de madurez y la formación que se ofrece para quienes desean incursionar en ella. La formación académica es un elemento básico para promover la competitividad y la productividad, sobretodo en una industria tan diversa y dinámica como la de los videojuegos.

Sin embargo, las oportunidades de crecimiento para el mercado de los videojuegos en México son altas, gracias a que las aplicaciones para dispositivos móviles y los juegos en línea permiten tener un mayor acceso a programadores o pequeños estudios.

De este modo, se presenta en México un entorno óptimo para impulsar e innovar la labor del desarrollo de videojuegos, así como la incorporación en distintas plataformas.

Los resultados completos del sondeo se incluyen en el ANEXO A (SONDEO REALIZADO A LOS ESTUDIOS DE DESARROLLO DE VIDEOJUEGOS INDEPENDIENTES EN MÉXICO).

2. CAPÍTULO 2. EL ESTÁNDAR ISO/IEC 29110 PERFIL BÁSICO

Antes de comenzar con la descripción del estándar ISO/IEC 29110 Perfil Básico, se introducen los conceptos de *Proceso de Software* y *Modelo de Proceso de Software*.

2.1. Proceso de software

Un *Proceso de Software* es una serie de actividades relacionadas que conduce a la elaboración de un producto de software [11]. Estas actividades pueden incluir el desarrollo de software desde cero en un lenguaje de programación estándar como Java o C.

El nuevo software empresarial con frecuencia se desarrolla extendiendo y modificando los sistemas existentes, o configurando e integrando el software comercial o componentes del sistema. Existen muchos procesos de software, pero todos deben incluir cuatro actividades que son fundamentales para la ingeniería de software [11]:

1. **Especificación del Software.** Tienen que definirse tanto la funcionalidad del software como las restricciones de su operación.
2. **Diseño e implementación del Software.** Debe desarrollarse el software para cumplir con las especificaciones.
3. **Validación del Software.** Hay que validar el software para asegurarse de que cumple lo que el cliente quiere.
4. **Evolución del Software.** El software tiene que evolucionar para satisfacer las necesidades cambiantes del cliente.

En la práctica estas actividades son complejas en sí mismas e incluyen tareas tales como la validación de requerimientos, diseño arquitectónico, pruebas unitarias, etc. También existen actividades de soporte al proceso, como la documentación y el manejo de la configuración del software.

Cuando los procesos se discuten y describen, por lo general se habla de actividades como especificar un modelo de datos, diseñar una interfaz de usuario, etc., así como del orden de dichas actividades. Sin embargo, al igual que las actividades, también las descripciones de los procesos deben incluir:

1. **Productos**, que son los resultados de una actividad del proceso. Por ejemplo, el resultado de la actividad del diseño arquitectónico es un modelo de la arquitectura de software.
2. **Roles**, que reflejan las responsabilidades de la gente que interviene en el proceso. Ejemplos de roles: administrador del proyecto, programador, etc.
3. **Precondiciones y postcondiciones**, que son declaraciones válidas antes y después de que se realice una actividad del proceso o se cree un producto. Por ejemplo antes de comenzar el diseño arquitectónico, una precondición es que el cliente haya aprobado todos los

requerimientos; después de terminar esta actividad, una postcondición podría ser que se revisen aquellos modelos UML que describen la arquitectura.

Los procesos de software son complejos y, como todos los procesos intelectuales creativos, se apoyan en personas con capacidad de juzgar y tomar decisiones. No hay un proceso ideal, además, las organizaciones pueden crear sus propios procesos de desarrollo de software.

En el caso de los videojuegos con requerimientos rápidamente cambiantes, es probable que sea más efectivo un proceso menos formal y más flexible [8].

En ocasiones los procesos de software se clasifican como dirigidos por un plan (*plan-driven*) o como procesos ágiles. Los procesos dirigidos por un plan son aquellos donde todas las actividades del proceso se planean por anticipado y el avance se mide contra dicho plan. En los procesos ágiles la planeación es incremental y es más fácil modificar el proceso para reflejar los requerimientos cambiantes del cliente. Cada enfoque es adecuado para diferentes tipos de software. Se necesita encontrar un equilibrio entre procesos dirigidos por un plan y procesos ágiles.

2.2. Modelo de proceso de software

Un modelo de procesos representa a otro desde una particular perspectiva y, por lo tanto, ofrece solo información acerca de dicho proceso. Por ejemplo, un modelo de actividad del proceso muestra las actividades y su secuencia, pero quizá sin presentar los roles de las personas que intervienen en esas actividades [11].

Algunos modelos de proceso generales son abstracciones del proceso que se utilizan para explicar los diferentes enfoques del desarrollo de software. Se pueden considerar marcos del proceso que se extienden y se adaptan para crear procesos más específicos de ingeniería de software [11]. Por ejemplo:

El modelo de cascada (*waterfall*). Toma las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución y, luego, los representa como fases separadas del proceso, tal como especificación de requerimientos, diseño de software, implementación, pruebas, etc. [11].

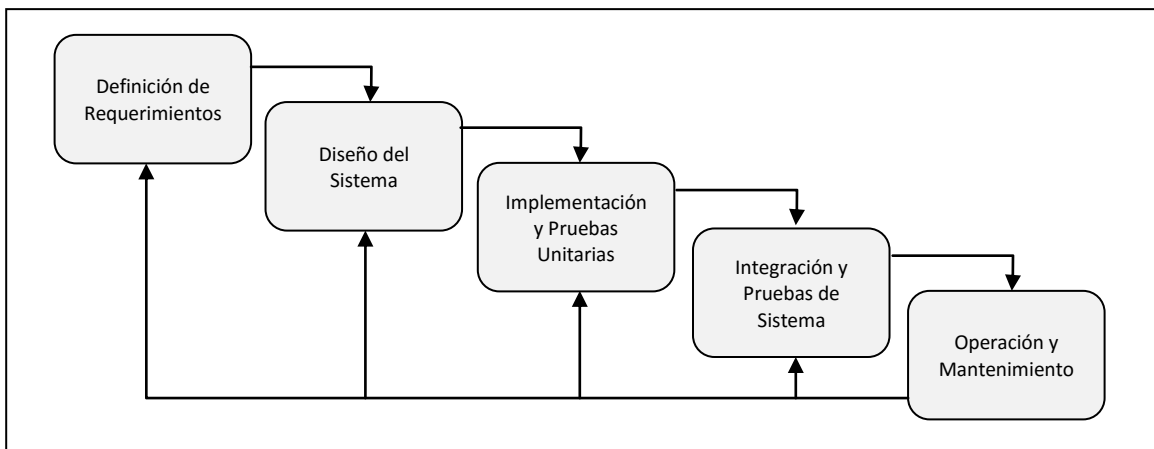


FIGURA 2.1 - El modelo en cascada [11]

Desarrollo incremental. Este enfoque vincula las actividades de especificación, desarrollo y validación. El sistema desarrolla como una serie de versiones (incrementos), y cada versión añade funcionalidad a la versión anterior [11].

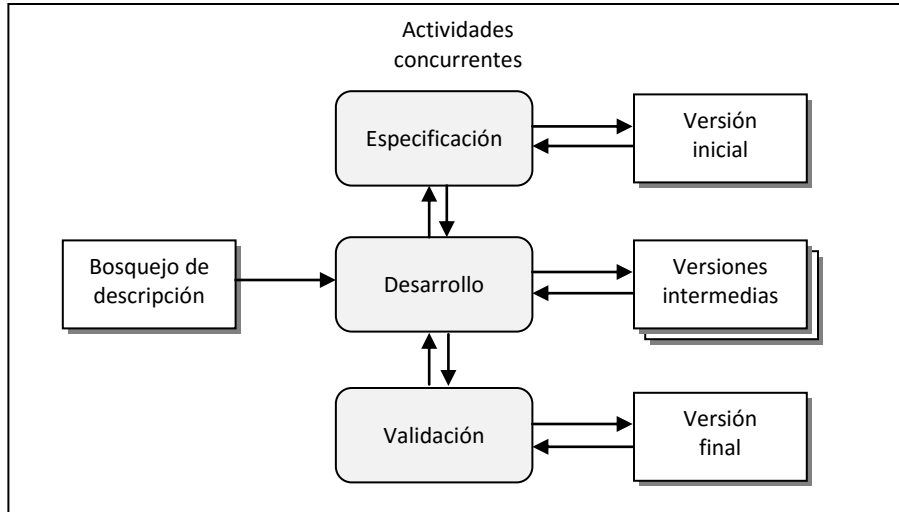


FIGURA 2.2 - Desarrollo incremental [11]

Ingeniería de software orientada a la reutilización. Este enfoque se basa en la existencia de un número significativo de componentes reutilizables. El proceso de desarrollo del sistema se enfoca en la integración de estos componentes en un sistema, en vez de desarrollarlo desde cero [11].

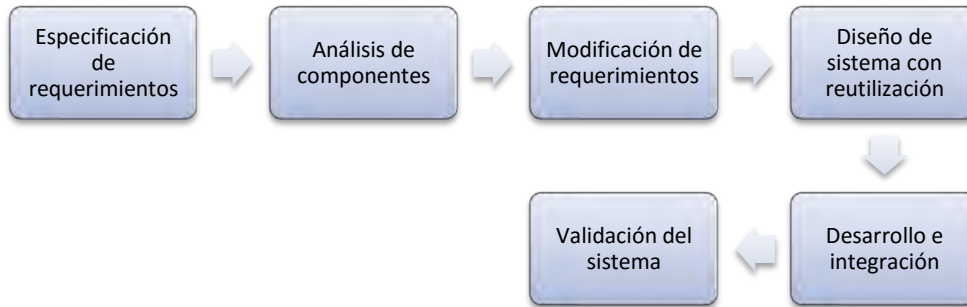


FIGURA 2.3 - Ingeniería de software orientada a la reutilización [11]

Dichos modelos no son mutuamente excluyentes y con frecuencia se usan en conjunto, sobre todo para el desarrollo de grandes sistemas. Para este tipo de sistemas, tiene sentido combinar algunas de las mejores características de los modelos de desarrollo en cascada e incremental. Se necesita contar con información sobre los requerimientos esenciales del sistema para diseñar la arquitectura de software que apoye dichos requerimientos. Los subsistemas dentro de un sistema más grande se desarrollan usando diferentes enfoques. Partes del sistema que son bien comprendidas pueden especificarse y desarrollarse al utilizar un proceso basado en cascada.

Partes del sistema que por adelantado son difíciles de especificar, como la interfaz de usuario, siempre deben desarrollarse con un enfoque incremental.

2.3. Visión general del estándar ISO/IEC 29110 Perfil Básico

El desarrollo de software con un equipo pequeño de personas es un gran reto debido a que deben llevar a cabo diferentes tipos de responsabilidades simultáneas.

Para ayudar a organizar las actividades de pequeños equipos de desarrollo existe el estándar ISO/IEC 29110 Perfiles del ciclo de vida para las pequeñas organizaciones.

El estándar ISO/IEC 29110 ofrece un modelo general para el desarrollo de software, el cual promueve las mejores prácticas de otros procesos estándar existentes, y es lo suficientemente ligero para aplicarse en pequeñas organizaciones de desarrollo de software. Proporciona varios perfiles: *De entrada*, *Básico*, *Intermedio* y *Avanzado*. Estos perfiles brindan un enfoque progresivo para satisfacer una mayoría de las VSEs ("Very Small Entity").

El estándar se crea para atender a las necesidades de las pequeñas organizaciones que desarrollan software, que de acuerdo a un estudio de la OECD representan más del 95% de las empresas en todo el mundo.

Define procesos del ciclo de vida, un marco de trabajo, taxonomía y una guía de evaluación para las pequeñas organizaciones para comprender y desarrollar sus prácticas.

El estándar permitirá a las pequeñas organizaciones evolucionar hacia modelos de procesos de software más maduros tales como ISO/IEC 12207 independientemente del enfoque actual en la organización.

La industria del software mundial, en su enorme mayoría (más del 90%) está constituida por entidades muy pequeñas (VSE). Una VSE es una empresa, organización, departamento o proyecto que tiene hasta 25 personas. Para este tipo de entidades es muy difícil adaptar estándares, en particular a los procesos de ciclo de vida del software, que fueron creados pensados en grandes organizaciones. No cuentan con los recursos, ni ven claro los posibles beneficios de su adopción.

Para facilitar el acceso a los estándares a este tipo de entidades, el subcomité SC7 del JTC1 ISO/IEC decidió crear perfiles de los estándares existentes, que tomen en cuenta a las características de las VSEs. Un *Perfil* es un subconjunto, seleccionado con cierto propósito, de los elementos de los estándares existentes, por ejemplo, subconjunto de procesos y sus salidas (*outcomes*) de ISO/IEC 12207 y de productos de trabajo de ISO/IEC 15289. Los perfiles están acompañados de guías que ayudan a implementar estos subconjuntos de estándares.

El conjunto de documentos ISO/IEC 29110 público objetivo, ha sido desarrollado para mejorar la calidad de productos y/o calidad de los servicios, y el desempeño de los procesos [6].

TABLA 2.1 - ISO/IEC 29110 Público Objetivo.

ISO/IEC 29110	TÍTULO	PÚBLICO OBJETIVO
PARTE 1	VISIÓN GENERAL	VSEs, evaluadores, productores de estándares, proveedores de herramientas y metodologías
PARTE 2	MARCO DE TRABAJO Y TAXONOMÍA	Productores de estándares, proveedores de herramientas y metodologías. No dirigido para VSEs.
PARTE 3	GUÍA DE EVALUACIÓN	Evaluadores y VSEs.
PARTE 4	ESPECIFICACIÓN DE PERFILES	Productores de estándares, proveedores de herramientas y metodologías. No dirigido para VSEs.
PARTE 5	GUÍAS DE ADMINISTRACIÓN E INGENIERÍA	VSEs.

Si un nuevo perfil es necesario, ISO/IEC 29110 Partes 4 y 5 pueden ser desarrolladas sin impactar en los documentos existentes y se denominarían ISO/IEC 29110 Parte 4-m y Parte 5-m-n respectivamente, a través del proceso ISO/IEC [6].

ISO/IEC 29110-1: Define los términos de negocio comunes al Conjunto de Documentos de Perfiles de la VSEs. Introduce los conceptos de procesos, ciclo de vida y estandarización, y la serie ISO/IEC 29110. Asimismo, es una introducción a las características y requerimientos de una VSEs, y clarifica los fundamentos para los perfiles específicos de las VSEs, documentos, estándares y guías [6].

ISO/IEC 29110-2: Introduce los conceptos para el perfil estandarizado de Ingeniería de Software para las VSEs, y define reglas comunes para el Conjunto de Documentos de Perfiles de las VSEs. Establece la lógica detrás de la definición y aplicación de perfiles estandarizados. Especifica los elementos comunes para todos los perfiles estandarizados (estructura, cumplimiento, evaluación) e introduce la taxonomía (catálogo) de perfiles ISO/IEC 29110 [6].

ISO/IEC 29110-3: determina las guías de evaluación del proceso y conforma los requerimientos necesarios para cumplir el propósito de los perfiles de la VSEs definidos. ISO/IEC 29110-3 también contiene información que puede ser útil para desarrolladores de métodos y herramientas de evaluación. ISO/IEC 29110-3 está dirigido a personas que tienen relación directa con el proceso de evaluación, por ejemplo, el asesor y el responsable de la evaluación, quienes necesitan ser guiados en asegurar que los requerimientos para realizar una evaluación sean alcanzados.

ISO/IEC 29110-4-1: provee la especificación para todos los perfiles del Grupo de Perfiles Genérico. El Grupo de Perfiles Genérico es aplicable a las VSEs que no desarrollan productos de software crítico. Los perfiles están basados en subconjuntos de elementos de estándares adecuados. Los perfiles de VSEs se aplican y están dirigidos a autores/proveedores de guías y de herramientas, y otro material de apoyo [6].

ISO/IEC 29110-5-1-2: provee una guía de implementación sobre Administración e Ingeniería para el Perfil Básico del Grupo de Perfiles Genérico descrito en ISO/IEC IS 29110 Parte 4-1. El Perfil Básico describe el desarrollo de software de una sola aplicación por un solo equipo de proyecto sin riesgos o factores situacionales especiales.

En la FIGURA 2.4 se describe el Conjunto de Documentos ISO/IEC 29110 y los posiciona dentro del marco de referencia. La Visión General y las Guías están publicadas como Reportes Técnicos (RT) y los Perfiles están publicados como Estándares Internacionales (EI).

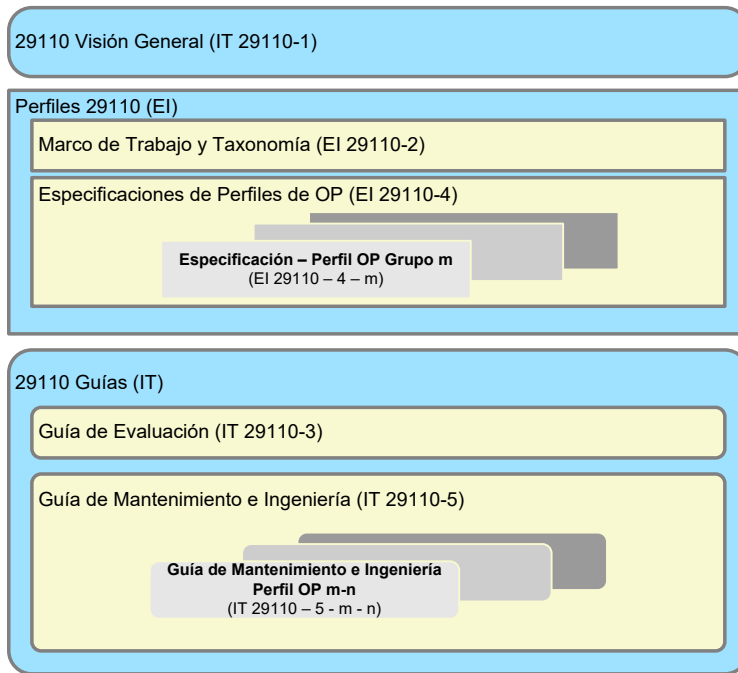


FIGURA 2.4 — Conjunto de Documentos ISO/IEC 29110 [6].

2.4. Descripción del estándar ISO/IEC 29110 Perfil Básico

El modelo propuesto en este trabajo toma como referencia la guía de implementación del proceso de *Administración del Proyecto* y proceso de *Implementación de Software* del Perfil Básico definidos en [ISO/IEC TR 29110-5-1-2, 2011]. Estos procesos se encuentran interrelacionados como se muestra en la FIGURA 2.5 :

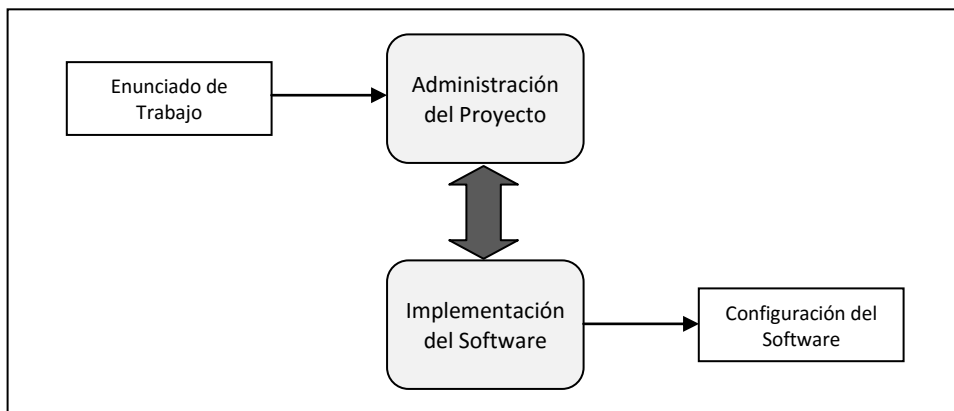


FIGURA 2.5 - Procesos interrelacionados

Las actividades definidas por el estándar en el Perfil Básico se dividen en los procesos de *Administración del Proyecto e Implementación de Software*. Estos procesos están interconectados, la administración dirige a la implementación y la implementación proporciona realimentación a la administración, ambas con sus propias actividades. El modelo se muestra en la siguiente FIGURA :

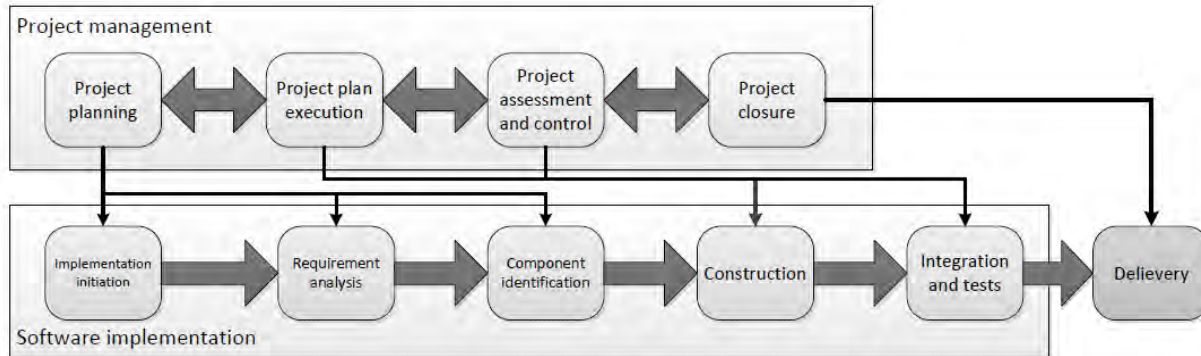


FIGURA 2.6 - Procesos de ISO/IEC 29110 Perfil Básico

El proceso de *Administración del Proyecto* tiene cuatro principales actividades: *Planeación del Proyecto*, *Ejecución del Plan del Proyecto*, *Evaluación y Control del Proyecto* y *Cierre del Proyecto*. Estas actividades se llevan a cabo conforme avanza el proyecto, con la ejecución del plan del proyecto y evaluación y control del proyecto como actividades paralelas.

El proceso de *Implementación de Software* tiene 6 actividades. Esas actividades son: *Inicio de la Implementación de Software*, *Análisis de Requerimientos*, *Arquitectura y Diseño Detallado de Software*, *Construcción de Software*, *Integración y Pruebas de Software* y *Entrega del Producto*. A diferencia del proceso de *Administración*, las fases avanzan conforme el producto madura.

En los siguientes puntos se describe el propósito y objetivos de cada uno de estos procesos.

2.5. Proceso de Administración del Proyecto en el Perfil Básico

El propósito del proceso de *Administración del Proyecto* es establecer y realizar de manera sistemática las tareas de un proyecto de implementación de software, que permitan cumplir con los objetivos del proyecto en calidad, tiempo y costos esperados.

Los objetivos del proceso de *Administración del Proyecto* son los siguientes:

- O1.- El *Plan del Proyecto* para la ejecución del proyecto es desarrollado de acuerdo al *Enunciado del Trabajo* y revisado y aceptado por el cliente. Las tareas y recursos necesarios para el trabajo son dimensionados y estimados.
- O2.- El progreso del proyecto es monitoreado contra el *Plan del Proyecto* y registrado en el *Reporte de Avance*. Correcciones para remediar problemas y desviaciones del plan son

tomadas cuando los objetivos del proyecto no se han logrado. El cierre del proyecto se realiza para obtener la aceptación del cliente en el *Documento de Aceptación*.

- O3.- Las *Solicitudes de Cambio* son tratadas mediante su recepción y análisis. Los cambios a los requerimientos de software son evaluados por su costo, tiempo e impacto técnico.
- O4.- Se llevan a cabo reuniones de revisión entre el *Equipo de Trabajo* y el *Cliente*. Los acuerdos son registrados y se les da seguimiento.
- O5.- Los riesgos son identificados en el desarrollo y durante la realización del proyecto.
- O6.- Se desarrolla una *Estrategia de Control de Versiones* de software. Los elementos de *Configuración del Software* son identificados, definidos y considerados en una línea base. Las modificaciones y liberaciones de los elementos son controladas y puestas a disposición del *Cliente* y del *Equipo de Trabajo*. El almacenamiento, manejo y entrega de los productos es controlada.
- O7.- El *Aseguramiento de la Calidad del Software* se realiza para dar garantías de que los productos de trabajo y procesos cumplen con el *Plan del Proyecto* y *Especificación de Requerimientos*.

La FIGURA 2.7 muestra el diagrama del flujo de información entre las actividades del proceso de *Administración del Proyecto* y sus productos de trabajo más relevantes.

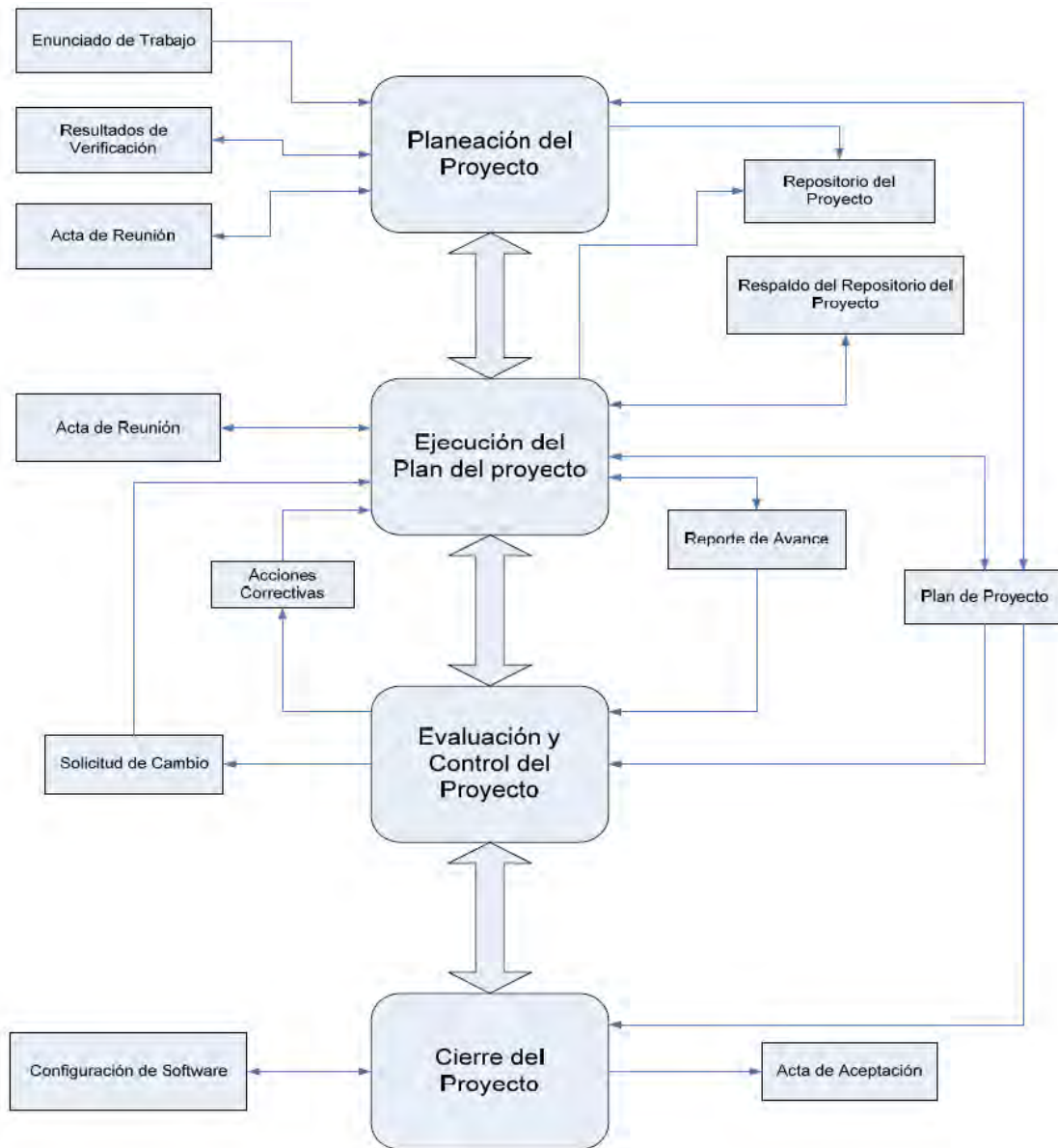


FIGURA 2.7 - Diagrama del proceso de Administración del Proyecto

2.6. Proceso de Implementación del Software en el Perfil Básico

El propósito del proceso de *Implementación de Software* es la realización sistemática de las actividades de análisis, diseño, construcción, integración, pruebas y entrega para productos de software, nuevos o modificados, de acuerdo a los requerimientos especificados [6].

Los objetivos del proceso de *Implementación del Software* son los siguientes:

- O1.- Las tareas de las actividades se realizan a través del cumplimiento del Plan del Proyecto actual.
- O2.- Los requerimientos del software se definen, analizan para su corrección y pruebas, son agregados a la línea base, comunicados y aprobados por el cliente.
- O3.- La *Arquitectura y Diseño Detallado del Software* es desarrollada y puesta en línea base. Aquí se describen los componentes de software y las interfaces internas y externas de los mismos. Se establece la consistencia y trazabilidad de los requerimientos de software.
- O4.- Se producen los componentes de software definidos por el diseño. Se definen y realizan las pruebas unitarias para verificar la consistencia con los requerimientos y el diseño. Se establece la trazabilidad de los requerimientos y el diseño.
- O5.- El software es producido realizando la integración de componentes de software y es verificado usando los *Casos de Prueba y Procedimientos de Prueba*. Los resultados son registrados en el *Reporte de Pruebas*. Los defectos son corregidos y se establece la consistencia y trazabilidad del *Diseño de Software*.
- O6.- La *Configuración del Software* es integrada, puesta en la línea base y almacenada en el *Repositorio del Proyecto*, asegurando que cumpla con la *Especificación de Requerimientos* acordada con el cliente. Se incluyen los manuales de usuario, operación y mantenimiento. Se detectan necesidades de cambios en la *Configuración de Software* y se propone la *Solicitud de Cambio*.
- O7.- Las tareas de *Verificación y Validación* de todos los productos de trabajo requeridos se realizan utilizando los criterios definidos para lograr la consistencia entre productos de entrada y salida en cada actividad. Los defectos son identificados y corregidos; los registros se almacenan en los *Resultados de Verificación/Validación*.

La FIGURA 2.8 muestra el diagrama del proceso de *Implementación de Software*, y sus productos de trabajo más relevantes.

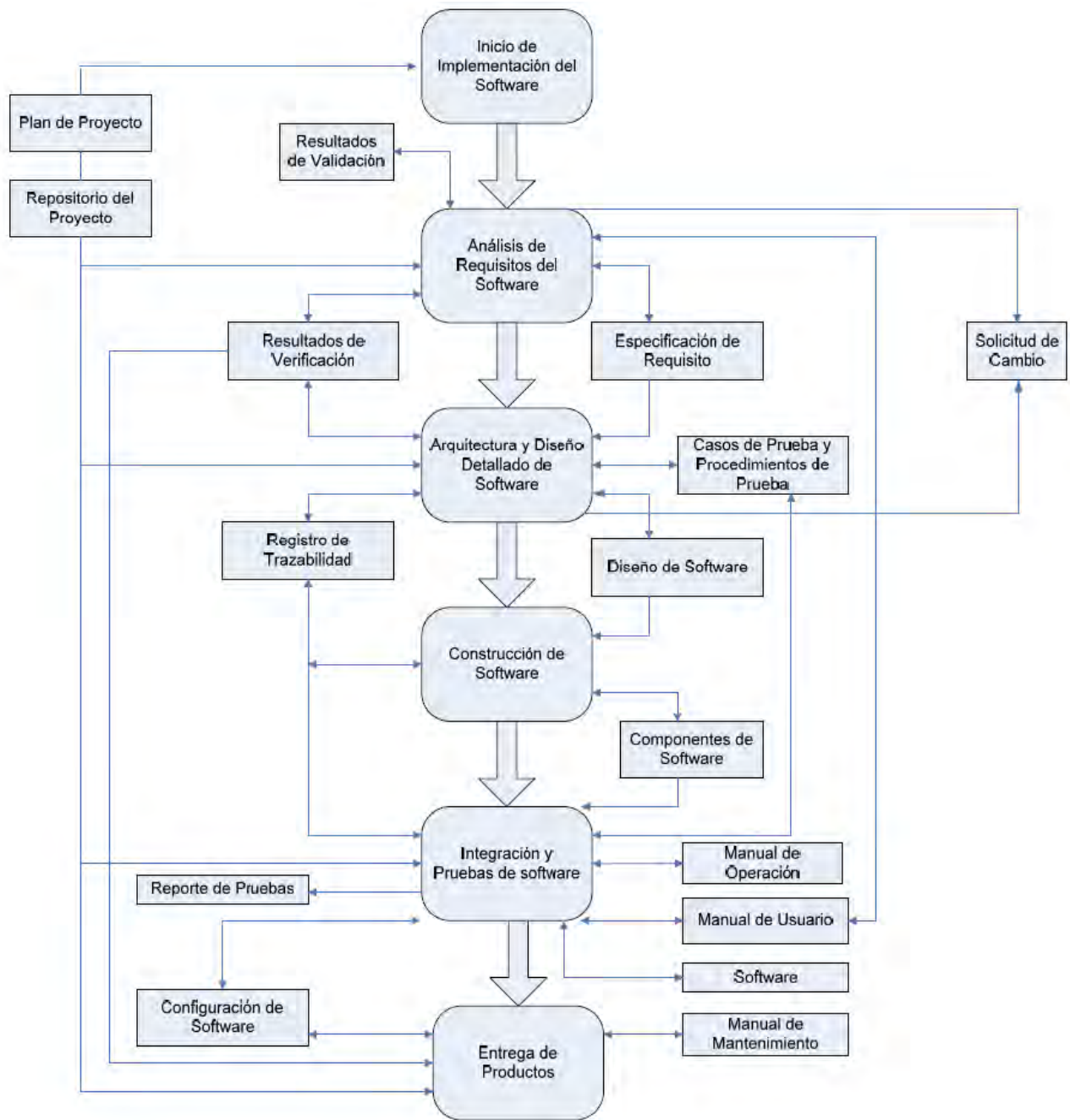


FIGURA 2.8 - Diagrama del proceso de Implementación de Software

La industria de los videojuegos podría beneficiarse de las prácticas de administración de proyectos desarrolladas por la industria del software en general.

El estándar *ISO/IEC 29110 Perfil Básico* no va dirigido hacia algún tipo de software en particular como el desarrollo de videojuegos. De acuerdo al estudio realizado en [15], este estándar podría utilizarse en la industria de los videojuegos, sin embargo deberán realizarse ciertos cambios para poder adaptarse a las necesidades de esta industria.

Es por eso que *ISO/IEC 29110 Perfil Básico* podría ser un modelo útil para el desarrollo de videojuegos y se tomará como base para la propuesta del modelo de procesos para el desarrollo de videojuegos independientes.

3. CAPÍTULO 3. CARACTERÍSTICAS BÁSICAS EN EL DESARROLLO DE VIDEOJUEGOS

3.1. Diferencias en el proceso de desarrollo de videojuegos

Desarrollar videojuegos es una tarea complicada que involucra la experiencia de profesionales de varias disciplinas incluyendo ciencias de la computación, arte, diseño y negocios. La presión que existe en el desarrollo de videojuegos para llegar al mercado tan pronto como sea posible complica la planeación de los tiempos de desarrollo.

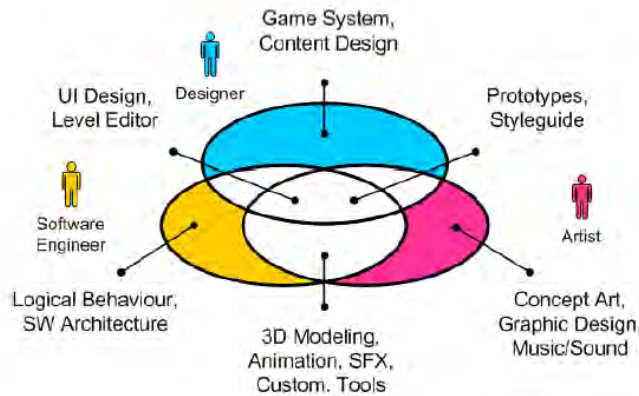


FIGURA 3.1 - Desarrollo de videojuegos como una actividad colaborativa entre varias disciplinas [8]

Aunque se han incorporado prácticas del desarrollo de software por los desarrolladores de videojuegos, una diferencia fundamental es que un videojuego tiene como objetivo proporcionar una experiencia lúdica. Además, la industria de los videojuegos tiene énfasis en la experiencia de usuario y la usabilidad. Esto puede causar una divergencia en las prácticas.

En general, cuando se compara a la industria del software y la industria de los videojuegos hay diferencias ocasionadas por los diferentes públicos a los que van dirigidos, diferentes fuentes de información y usos previstos para los productos. Hay incluso diferencias en aspectos como la creatividad, y los requerimientos técnicos para implementarlos. Las diferencias y similitudes entre la industria del software y la industria de los videojuegos pueden resumirse en lo siguiente [12]:

TABLA 3.1 - Diferencias observadas entre la industria de los videojuegos y la industria del software

INDUSTRIA DE LOS VIDEOJUEGOS	INDUSTRIA DEL SOFTWARE
Los productos son creados para entretener, y no necesariamente tienen un fin más allá del entretenimiento.	Los productos son construidos a propósito, para llenar alguna necesidad y proporcionar funcionalidad.
Los productos tienen como objetivo maximizar el tiempo utilizado con el producto.	Los productos tienen como objetivo minimizar el tiempo necesario para alcanzar la funcionalidad.
Alta prioridad en la calidad de la experiencia del usuario, calidad definida por los usuarios.	La calidad difiere entre productos y proyectos, la calidad está basada en requerimientos bien definidos.

La solución técnica solo es una parte del producto, se requiere creatividad en la historia, narrativa, jugabilidad, audio y arte.	Es importante la solución técnica y que funcione correctamente. Creatividad principalmente en el diseño de la interfaz de usuario.
Es común que se agregue contenido descargable (DLC) después del lanzamiento del producto.	Se corrigen errores y se le da mantenimiento.
Los requerimientos y el diseño están basados en conceptos abstractos.	Los requerimientos y el diseño están basados en conceptos del mundo real.

El desarrollo de software para la industria de los videojuegos está evolucionando rápidamente y se está volviendo cada vez más complejo. Incluso también hay diferencias en las actividades de los procesos de desarrollo en ambas industrias [12]. Un resumen de estas diferencias se muestra en la siguiente TABLA:

TABLA 3.2 - Diferencias en las actividades de los procesos de desarrollo de videojuegos y de software

ACTIVIDAD	PRINCIPALES DIFERENCIAS
1. PLANEACIÓN DEL PROYECTO	Generalmente las actividades de administración del proyecto son similares al desarrollo de productos de software. Sin embargo, los retos en el alcance del proyecto son provocados por una falta de conocimiento sobre los requerimientos afectivos y la falta de métodos de validación que podrían abordar eficazmente dichos requisitos [8]. También hay periodos de presión y calendarios muy marcados.
2. PRUEBAS	La industria de los videojuegos hace énfasis en la experiencia del usuario y la funcionalidad, aunque los métodos de prueba son similares a los que se aplican en la industria del software.
3. DISEÑO	La actividad de diseño del videojuego es diferente a la de otros productos de software, debido a la creatividad e inventiva requerida en la historia, personajes, niveles, jugabilidad, música, animaciones, cinemáticas, etc. El aspecto creativo del desarrollo no es modelado utilizando prácticas de <i>Ingeniería de Software</i> .
4. ADMINISTRACIÓN DE LOS CAMBIOS	El diseño del videojuego cambia constantemente, los productos pueden sufrir cambios importantes en etapas avanzadas del desarrollo si cualquier aspecto del diseño no es atractivo para los usuarios. Algunas veces las mejores ideas surgen durante el desarrollo dándole otra dirección al proyecto. Esto podría significar una sobrecarga de trabajo en periodos cortos o medianos [8].
5. DISEÑO TÉCNICO	Algunas herramientas orientadas al desarrollo de videojuegos como los motores de juego son lo suficientemente sofisticados para requerir conocimientos específicos en el desarrollo del juego. También se deben integrar componentes externos y muchas veces se desarrolla el videojuego simultáneamente para múltiples plataformas.
6. ADMINISTRACIÓN DE LA CALIDAD	La industria de los videojuegos pone más énfasis en la experiencia de usuario y en los requerimientos no funcionales que la industria del software, y la calidad requerida no se puede satisfacer con definiciones funcionales o técnicas.
7. STAKEHOLDERS	Los equipos son multidisciplinarios (diseñadores, programadores, artistas, músicos, mercadólogo), y en estudios pequeños los miembros del equipo de trabajo deberán tener múltiples responsabilidades.

<p>8. ANÁLISIS DE REQUERIMIENTOS</p>	<p>No solo los requerimientos pueden cambiar en etapas avanzadas del proyecto, también existen requerimientos como la "diversión" que no tienen métricas que puedan ser aplicadas, lo que es divertido para una audiencia podría no serlo para otra. Además, la diversión debe de ser validada en cada etapa del proceso de desarrollo.</p> <p>La creatividad en el desarrollo de videojuegos juega un papel importante, definir un concepto de juego con algunos requerimientos deseados no es suficiente.</p>
--------------------------------------	---

3.2. Principales dificultades en el desarrollo de videojuegos

A continuación se describen las principales dificultades que existen en el proceso de desarrollo de videojuegos.

TABLA 3.3 - Principales dificultades en el desarrollo de videojuegos

DIFICULTAD	DESCRIPCIÓN
<p>1. CREAR EL CONCEPTO DEL VIDEOJUEGO CON CAMBIOS CONSTANTES</p>	<p>La fase de concepto en el proceso de desarrollo de videojuegos es diferente y más compleja que en otro tipo de productos de software, debido a que la atmósfera del juego tiene que ser definida en un concepto inicial que podría ser muy diferente de la versión final del juego.</p> <p>Además, es probable que el concepto inicial cambie constantemente durante esta fase y mucho trabajo artístico y prototipos de jugabilidad se tengan que crear incluso durante esta fase.</p>
<p>2. ANTICIPAR EL TIPO DE VIDEOJUEGOS QUE TENDRÁN ÉXITO</p>	<p>Se debe identificar un público al que estará dirigido el videojuego. También se tiene que considerar la existencia de políticas de publicación en diferentes mercados, las cuales definen la manera, restricciones, condiciones, tiempo de publicación y tipo de videojuegos que pueden distribuirse en cada una de ellas.</p> <p>Es difícil anticipar qué tipo de videojuegos tendrán éxito en un mercado que cambia rápida y constantemente.</p>
<p>3. OBTENER FINANCIAMIENTO PARA CONCRETAR LOS PROYECTOS</p>	<p>La falta de presupuesto para concretar los proyectos es un problema constante sobre todo en los estudios independientes. Aunque ya existen algunas plataformas para poder obtener los recursos y poder financiar el proyecto, es difícil obtenerlos.</p>
<p>4. PLANEAR EL PROYECTO Y DETERMINAR SU ALCANCE</p>	<p>Hay que considerar aspectos como la planeación, ejecución, evaluación y control y cierre del proyecto. Sin embargo, el entorno cambiante en el desarrollo de videojuegos y las políticas de publicación en mercados de aplicaciones obligan a que la administración de proyectos de desarrollo de videojuegos tenga que ser ágil y altamente adaptable a cambios.</p> <p>Además, durante la fase de diseño hay muchos momentos decisivos que podrían cambiar drásticamente el proyecto, por ejemplo, pueden surgir ideas creativas que terminen dándole otra dirección, comenzar con una idea completamente nueva o cancelar el proyecto.</p> <p>Debido a las altas expectativas que se tienen del juego, no se dimensiona la cantidad de trabajo necesario para construir el tipo de juego que se quiere.</p>

<p>5. ANALIZAR Y VALIDAR REQUERIMIENTOS EN CONSTANTE CAMBIO</p>	<p>Cambio constante de los requerimientos incluso en etapas avanzadas del proyecto, si este no se considera entretenido o lo suficientemente bueno para tener éxito.</p> <p>Existen requerimientos como la "diversión" que no tienen métricas que puedan ser aplicadas, lo que es divertido para una audiencia podría no serlo para otra. Además, la diversión debe de ser validada en cada etapa del proceso de desarrollo.</p> <p>No es suficiente definir un concepto de juego con algunos requerimientos deseados, ya que la creatividad juega un papel muy importante en el desarrollo de videojuegos.</p>
<p>6. DISEÑAR EL VIDEOJUEGO</p>	<p>Se requiere mucha creatividad e inventiva para diseñar la historia, personajes, niveles, jugabilidad, música, animaciones, cinemáticas, etc. El aspecto creativo del desarrollo no es modelado utilizando prácticas de <i>Ingeniería de Software</i>.</p>
<p>7. DISEÑAR TÉCNICAMENTE EL VIDEOJUEGO</p>	<p>Pueden existir dificultades técnicas como la elección de un motor de juego y cambios de plataforma durante el desarrollo del videojuego. Niveles del juego que técnicamente sea complejo implementar.</p> <p>Regularmente durante el proceso de desarrollo se necesitan construir herramientas que permitan agilizar los procesos de producción de <i>assets</i> del juego.</p>
<p>8. INNOVAR O CREAR VIDEOJUEGOS COMPLETAMENTE ORIGINALES</p>	<p>No es fácil crear videojuegos completamente originales. El proceso de desarrollo de videojuegos es particularmente difícil en ese aspecto. Los videojuegos independientes se enfocan en la innovación, creatividad, y experimentación artística.</p>
<p>9. ORGANIZAR UN EQUIPO MULTIDISCIPLINARIO</p>	<p>Se puede organizar a un equipo de diferentes formas, dependiendo del tamaño del estudio y los roles. Sin embargo, en estudios independientes esto es un problema ya que es común que una sola persona tenga asignado múltiples roles en un proyecto, por ejemplo <i>Productor-Programador-Diseñador Líder</i>.</p>

3.3. Características básicas en el desarrollo de videojuegos

Considerando las diferencias generales entre la industria de los videojuegos y la industria del software, así como las principales dificultades durante el proceso de desarrollo de videojuegos, se busca identificar aquellos elementos que permitan minimizar estas dificultades.

Por lo anterior, se propone integrar un conjunto de características básicas que deben de ser consideradas al desarrollar videojuegos independientes. En la siguiente TABLA se identifican y describen estas características básicas.

TABLA 3.4 - Características básicas en el desarrollo de videojuegos

CARACTERÍSTICA	DESCRIPCIÓN
1. DESARROLLO DEL VIDEOJUEGO EN FASES	Son muchos expertos los que coinciden en que el ciclo de vida del desarrollo de videojuegos se debe aproximar al del desarrollo de una película de cine, estableciendo tres fases claramente diferenciadas que son <i>Preproducción</i> , <i>Producción</i> y <i>Posproducción</i> (por lo menos). A su vez en cada una de estas fases se identifican diversas etapas significativas y el equipo de producción se distribuye para colaborar en cada una de ellas [16].
2. CONCEPTO DE JUEGO Y PROTOTIPOS	La fase de concepto en el proceso de desarrollo de videojuegos es diferente y más compleja que en otro tipo de productos de software, debido a que la atmósfera del juego tiene que ser definida en un concepto inicial que podría ser muy diferente de la versión final del juego. Además, es probable que el concepto inicial cambie constantemente durante esta fase y mucho trabajo artístico y prototipos de jugabilidad se tengan que crear incluso durante esta fase.
3. CONCIENCIA DEL MERCADO	Se debe identificar un público al que estará dirigido el videojuego. También se tiene que considerar la existencia de políticas de publicación en diferentes mercados, las cuales definen la manera, restricciones, condiciones, tiempo de publicación y tipo de videojuegos que pueden distribuirse en cada una de ellas. Es difícil anticipar qué tipo de videojuegos tendrán éxito en un mercado que cambia rápida y constantemente. Por lo tanto, es fundamental investigar y evaluar videojuegos similares que estén publicados y probar que hay un mercado para el videojuego que se quiere construir.
4. CONSIDERACIÓN DE FUENTES DE FINANCIAMIENTO	Una de los principales problemas sobre todo para los estudios independientes es la falta de recursos. Por lo tanto en cada proyecto debe considerarse un modelo de financiamiento. Alguien tiene que aprobar o invertir en el concepto desarrollado, antes de poder pensar acerca del proyecto y plan de contratación [13].
5. AGILIDAD	Los métodos ágiles aumentan la flexibilidad y producción de videojuegos proporcionando los medios de adaptación a cambios a requerimientos y entornos sumado al aprendizaje de experiencias de desarrollo. Muchos estudios están utilizando metodologías ágiles como Scrum [14].
6. DESARROLLO ITERATIVO	Un proceso iterativo estándar para desarrollar tareas sería lo más útil para la industria de los videojuegos, orientado a las pequeñas y medianas organizaciones [15].
7. SOPORTE PARA CAMBIOS CONSTANTES EN EL VIDEOJUEGO	Los cambios de los requerimientos son constantes, incluso en etapas avanzadas del proyecto, si este no se considera entretenido o lo suficientemente bueno para tener éxito. Además, durante la fase de diseño hay muchos momentos decisivos que podrían cambiar drásticamente el proyecto, por ejemplo, pueden surgir ideas creativas que terminen dándole otra dirección, comenzar con una idea completamente nueva o cancelar el proyecto.

8. VALORACIÓN DE LA EXPERIENCIA DEL JUGADOR	La idea fundamental es la de involucrar al jugador y hacerlo al principio y durante el proceso de desarrollo, ya que muchos de los problemas del software se deben a una carencia en las fases iniciales del desarrollo (requerimientos). Esto ya ha sido contemplado en diversos estándares que plantean ciclos de vida del proceso que incluyen modelos de madurez para la usabilidad como pilar fundamental que garantiza el éxito del producto en cuanto a la Experiencia del Usuario [16].
9. SOPORTE PARA MANTENIMIENTO	Es común que después de la publicación de un videojuego se realicen algunas actualizaciones. Pueden incluir correcciones de errores, balanceo de mecánicas de juego, nuevo contenido, actualizaciones, etc.
10. INTEGRACIÓN DE UN EQUIPO MULTIDISCIPLINARIO	Desarrollar videojuegos es una tarea complicada que involucra la experiencia de profesionales de varias disciplinas incluyendo ciencias de la computación, arte, diseño y negocios. En estudios independientes es común que una sola persona tenga asignado múltiples roles en un proyecto.

Este conjunto de características básicas es utilizado en la integración del modelo de procesos para el desarrollo de videojuegos independientes, el cuál es descrito en el capítulo 4.

A continuación se describen propuestas de trabajos relacionados y metodologías enfocadas al desarrollo de videojuegos, como resultado de la revisión sistemática de literatura.

3.4. Trabajos relacionados y procesos de desarrollo de videojuegos

Una de las finalidades primordiales del presente trabajo es la de minimizar las principales dificultades que se presentan durante el desarrollo de videojuegos y proporcionar una guía para planificar, gestionar y ejecutar la construcción de videojuegos. Sin embargo no es la primera propuesta en su tipo y existen otros trabajos relacionados para cubrir estas dificultades.

Durante la investigación se encontraron una gran variedad de procesos y modelos de desarrollo de videojuegos como resultado de una revisión sistemática de literatura, que podemos agruparlos de la siguiente forma:

3.4.1. Procesos de desarrollo de videojuegos generales

- En [17] se describe un proceso de desarrollo de videojuegos básico. Las fases propuestas en este proceso son las siguientes: *Preproducción*, *Producción*, *Pruebas (Testing)* y *Posproducción (Wrap-Up)*.

El siguiente diagrama define las actividades generales de cada fase de un ciclo *básico* de producción y muestra cómo el éxito de cada fase se basa en la finalización de la fase anterior (como en un modelo en *cascada*).

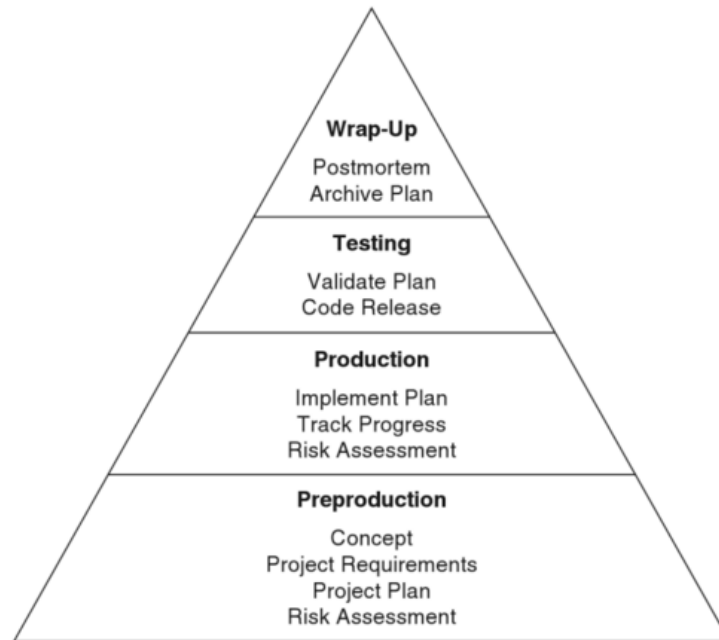


FIGURA 3.2 - Ciclo básico de producción de videojuegos [17]

La fase de *Preproducción* es crítica para definir lo que será el juego, cuánto tiempo tomará construirlo, cuánta gente se necesitará y cuánto costará. Esta fase podría durar desde una semana hasta más de un año dependiendo del alcance del proyecto (*Preproducción* requiere entre 10-25% del tiempo total de desarrollo [17]).

La meta principal de la fase de *Preproducción* es crear el plan del proyecto ya que proporciona una base sólida para la construcción del videojuego. El plan debe incluir información sobre el concepto de juego, características y restricciones que afectan este concepto, y la base técnica y documentación del diseño. Los componentes principales de esta fase son los siguientes: *Concepto*, *Requerimientos del Proyecto*, *Plan del Proyecto* y *Evaluación de riesgos* (ejecutado en cada etapa).

La fase de *Producción* es cuando el equipo comienza la producción de *assets* y código para el videojuego. En muchos casos, la línea entre las fases de *Preproducción* y *Producción* es confusa porque se puede comenzar la *Producción* de algunas características mientras otras están en *Preproducción*. Es muy probable que durante esta fase existan cambios en algunas características del juego o que algunos *assets* sean agregados, cambiados, o removidos.

Esta fase se centra en el contenido y la creación de código, evaluación del progreso y completar tareas. Además se mantiene la evaluación de riesgos. Las actividades definidas para esta fase son las siguientes: *Hacer el Plan de Implementación*, *Evaluación del Progreso* y realización de tareas.

En la fase de *Pruebas (Testing)* se asegura que todo funcione correctamente y que no haya errores en el código. Las pruebas se realizan durante la fase de *Producción*, ya que el departamento de *Aseguramiento de la Calidad* revisará una versión jugable (*game build*) en cada hito del proyecto. Después de las pruebas *beta*, cuando todos los *assets* y características del juego están totalmente

implementadas, el objetivo principal del equipo de desarrollo será corregir los errores encontrados y crear nuevas versiones del juego (*game builds*) para probarse. Es una fase crítica en el desarrollo de videojuegos y plantea dos actividades: *Plan de Validación* y *Liberación de código*.

En la fase de *Posproducción (Wrap-up)*, después de que el código se libera y se aprueba para manufactura, el proceso de desarrollo de videojuegos necesita ser "envuelto" antes de que se complete oficialmente. Es una oportunidad para reflexionar y evaluar la reciente experiencia de desarrollo. La fase de *Posproducción* considera dos actividades *Lecciones Aprendidas* y *Archivar el Plan*.

- Un modelo similar, también en cascada se describe en [16]. Se mantienen las mismas fases, aunque varían las actividades de cada fase (la actividad *Diseño de Juego* se realiza en *Producción*).

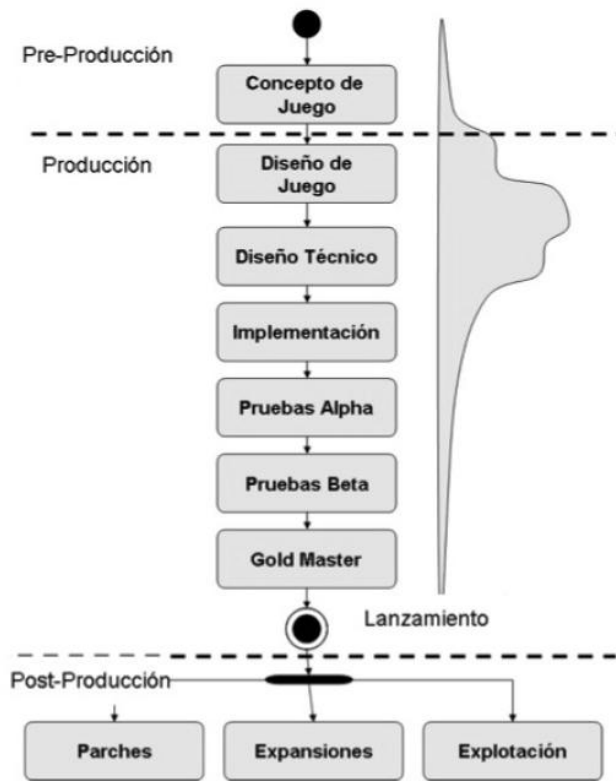


FIGURA 3.3 - Fases y actividades generales en la producción de un videojuego [16]

- Finalmente, en [17] también se propone el siguiente modelo si se planea construir el videojuego utilizando prototipos funcionales que añadan nuevas características gradualmente.

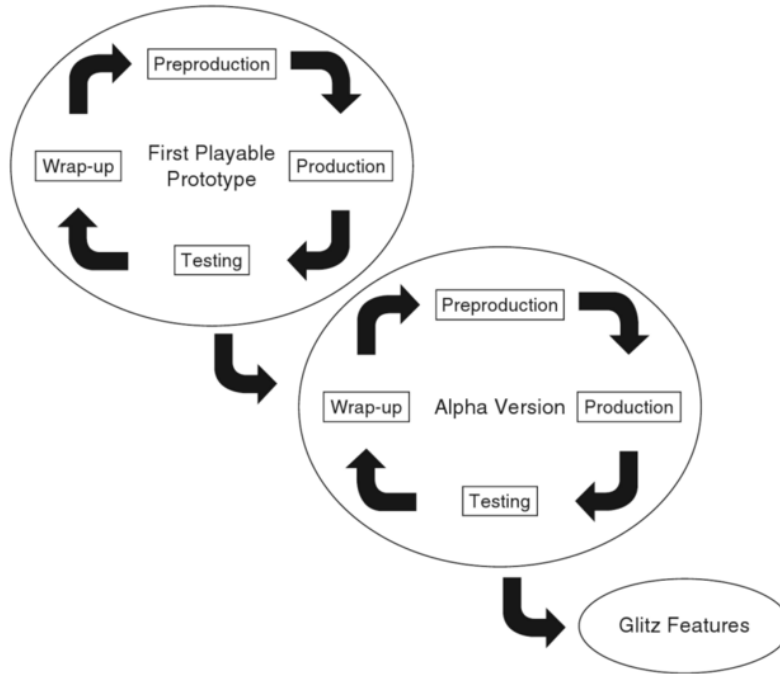


FIGURA 3.4 - Múltiples ciclos de producción para un solo proyecto

El primer ciclo consiste en *Preproducción*, *Producción* y *Pruebas* del prototipo; el segundo se enfoca en las características esenciales y *assets* del juego; y en el tercer ciclo se crean y agregan las características detalladas del juego y *assets*.

- Es un modelo similar al *Modelo Pipeline* planteado en [18], en el cual el prototipo se desarrolla por pruebas de concepto y se desarrolla gradualmente en varias iteraciones.

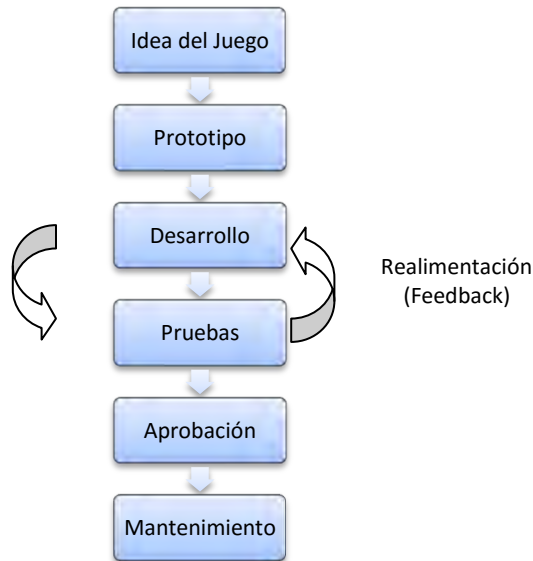


FIGURA 3.5 - Modelo Pipeline

3.4.2. Procesos ágiles de desarrollo de videojuegos

En [8] se propone un proceso de desarrollo de videojuegos basado en *Scrum*¹. El proceso es separado en las siguientes fases: *Preproducción*, *Producción* y *Cierre del proyecto*. La fase de *Preproducción* se enfoca en la invención y es orientada a corto plazo, mientras que la fase de *Producción* es adecuada para el desarrollo a largo plazo.

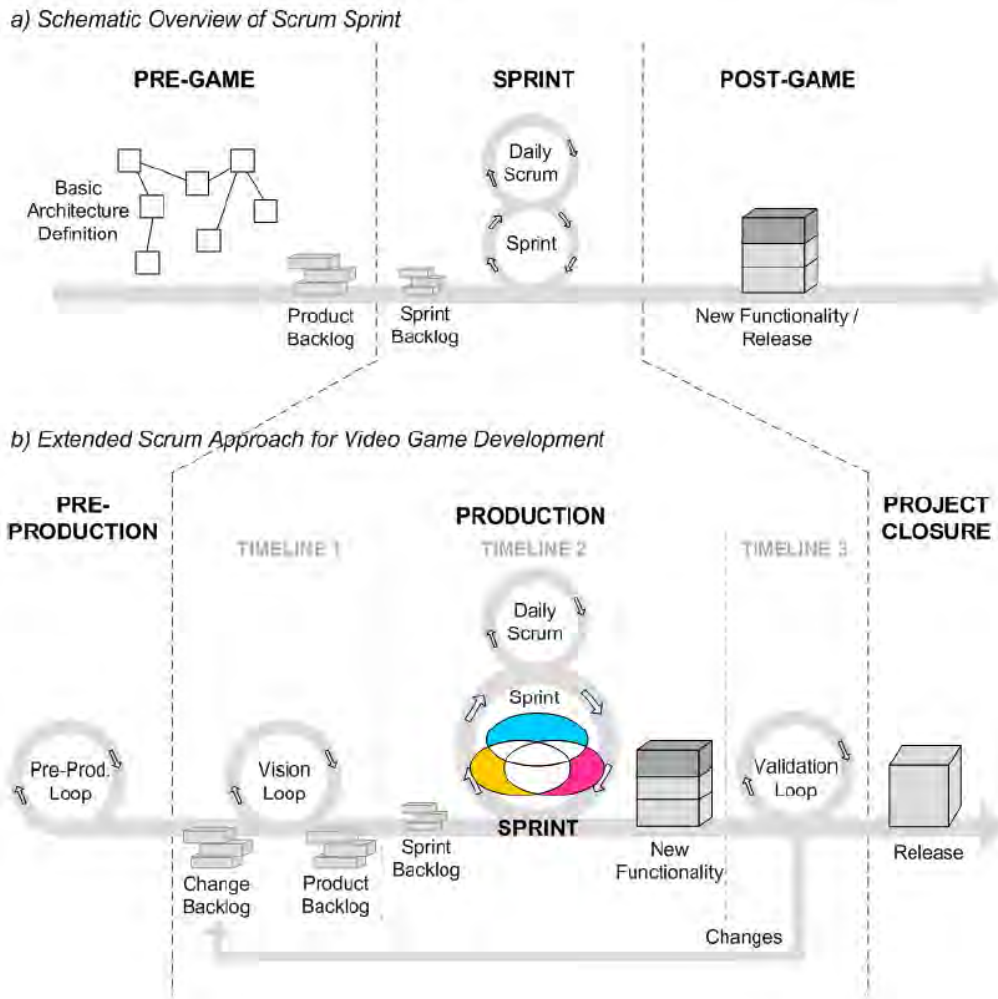


FIGURA 3.6 - Proceso de desarrollo de videojuegos propuesto basado en Scrum

La principal tarea de la *Preproducción* es identificar posibles candidatos de proyectos de software, análisis de requerimientos, evaluación de riesgos y requerimientos generales del proyecto, por ejemplo financiamiento. La identificación de proyectos de software candidatos se logra principalmente realizando prototipos orientados al usuario final.

¹ *Scrum* es un método de desarrollo ágil, que se basa en *Sprints*: ciclos de desarrollo cortos. *Scrum* puede usarse como base para administración de proyectos ágiles [11]. El desarrollo ágil es un conjunto de metodologías que se enfocan en la realización de productos de valor a través de iteraciones y retroalimentación [17].

La fase de *Producción* recibe el proyecto completo de *Preproducción* y crea un producto vendible dentro del tiempo, presupuesto y calidad esperado. El flujo general de trabajo de producción está basado en *Scrum*, por lo cual es separado en tres procesos paralelos:

- **Vision Loop.** Es comparable con el *Product Owner*² en un *Scrum* normal e involucra a autoridades técnicas y creativas. Mantiene la visión del producto durante el ciclo de vida del proyecto y administra el *Product Backlog*³. Además, *Visión Loop* recibe los resultados de las pruebas de *Validation Loop* y las mantiene en una bitácora de cambios (*change backlog*).
- **Sprint.** Ejecuta un *Sprint* clásico de *Scrum* y crea un incremento completo del producto al final de un *Sprint*.
- **Validation Loop.** Cubre la validación del producto, verificación y es responsable del control de calidad general del producto.

La fase *Cierre del Proyecto* cubre la distribución del juego final así como el análisis de la retrospectiva (postmortems) y lecciones aprendidas.

De esta propuesta se puede concluir lo siguiente:

El proceso propuesto basado en *Scrum* puede ser utilizado para mejorar la integración y colaboración del flujo de trabajo de equipos de desarrollo heterogéneos. La integración de disciplinas heterogéneas se logra ejecutando múltiples flujos de trabajo específicos durante un *Sprint* de producción que son ajustados por reuniones diarias de *Scrum*.

La fase de *Preproducción* proporciona suficiente espacio a los equipos para elaborar ideas potenciales y toma en cuenta enfoques como el diseño de prototipos de conceptos, que conducen a una buena aproximación de los requerimientos del usuario final en fases tempranas del proyecto, pero requiere la creación de prototipos rápidamente.

El diseño del videojuego no es solo un paso dentro de un *sprint* sino un proceso ejecutado concurrentemente. Este enfoque del proceso permite cambios frecuentes, incorporación de cambios continuos y se asemeja a estructuras de procesos de industrias que tienen dominada la estética en la ingeniería.

La calidad general del producto creado, se valida en un proceso separado (*Validation Loop*), optimizado para la evaluación cuantitativa y cualitativa de software. Los resultados se registran y son evaluados y probablemente lleven a cambios artísticos, técnicos, de diseño, etc.

Las observaciones de las tendencias en la industria internacional de los videojuegos sugieren que el proceso propuesto podría ser prometedor para el trabajo futuro. Sin embargo, se requiere de una versión más detallada del proceso propuesto y una evaluación del proceso en un ambiente controlado para verificar los beneficios esperados de forma sistemática.

² Product owner establece y comunica la visión del juego y prioriza sus características.

³ Product backlog es una lista priorizada de los requerimientos o características de un juego, un conjunto de herramientas, o la línea de producción para hacer el juego [14].

3.4.3. Procesos de desarrollo de videojuegos centrados en el jugador

Está inspirado directamente en los principios fundamentales del *Diseño Centrado en el Usuario*⁴ y de los procesos de desarrollo de software que se han derivado de los mismos.

La idea fundamental del *Diseño Centrado en el Usuario* es la de involucrar al usuario y hacerlo al principio de cualquier proceso de desarrollo, ya que muchos de los problemas del software se deben a una carencia en las fases iniciales del desarrollo (requerimientos). Esto ya ha sido contemplado en diversos estándares que plantean ciclos de vida del proceso que incluyen modelos de madurez para la usabilidad como pilar fundamental que garantiza el éxito del producto en cuanto a la Experiencia del Usuario [16].

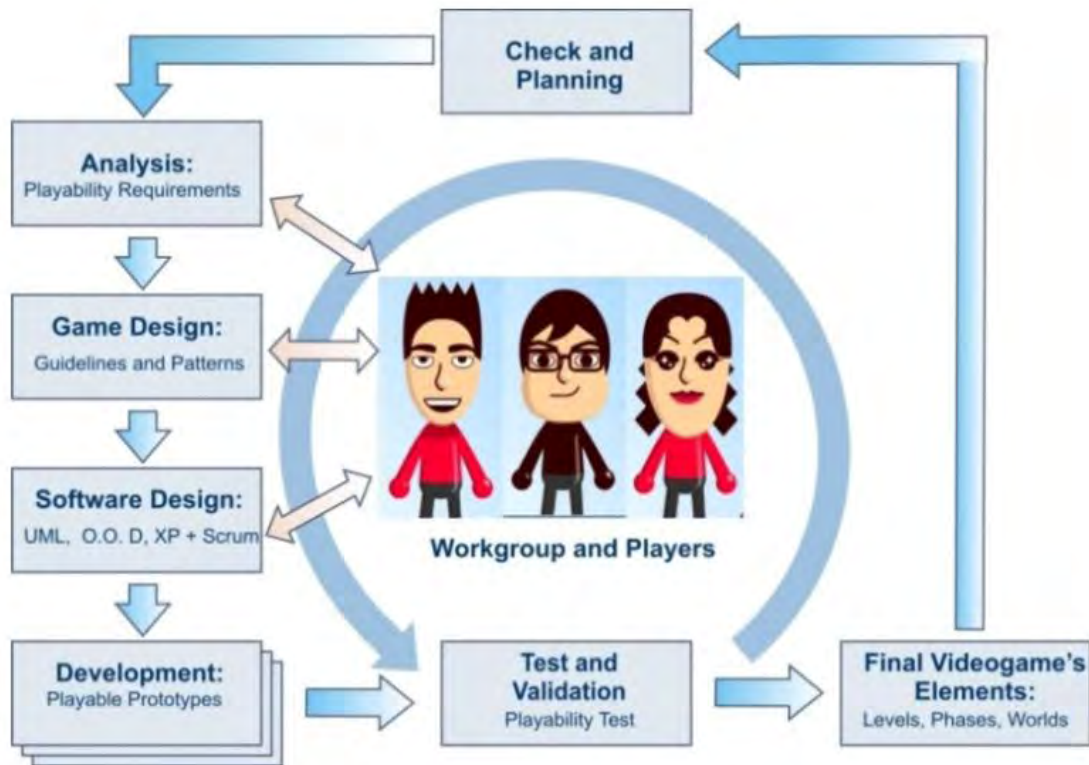


FIGURA 3.7 - Proceso de Desarrollo Centrado en el Jugador

De la misma forma que el *Diseño Centrado en el Usuario* es necesario para el desarrollo de aplicaciones que cubran los requisitos del usuario de forma adecuada, el *Diseño Centrado en el Jugador* es especialmente importante para considerar la diversidad y subjetividad de los perfiles de jugadores existentes. Además, esto contribuye directamente a la reducción de la proliferación de productos que requieren numerosos “parches” incluso desde los primeros meses de vida en el mercado.

⁴ El Diseño Centrado en el Usuario (DCU), o User Centered Design (UCD), es definido por la User Experience Professionals Association (UXPA) como un enfoque de diseño cuyo proceso está dirigido por información sobre las personas que van a hacer uso del producto.

3.4.4. Modelo de referencia

Finalmente, en [13] se propone un modelo de referencia para producción de videojuegos. Se analizan cuatro métodos de producción: tres métodos documentados (*Game Development and Production* [19], *Introduction to Game Development* [20], y *The Game Production Handbook* [17]) y un método obtenido a través de un caso de estudio en una compañía de desarrollo de videojuegos. En este estudio se comparan todas las actividades y conceptos de los cuatro métodos analizados.

El método de referencia final contiene cuatro fases: *Concepto*, *Preproducción*, *Producción* y *Posproducción*, y trece actividades principales.

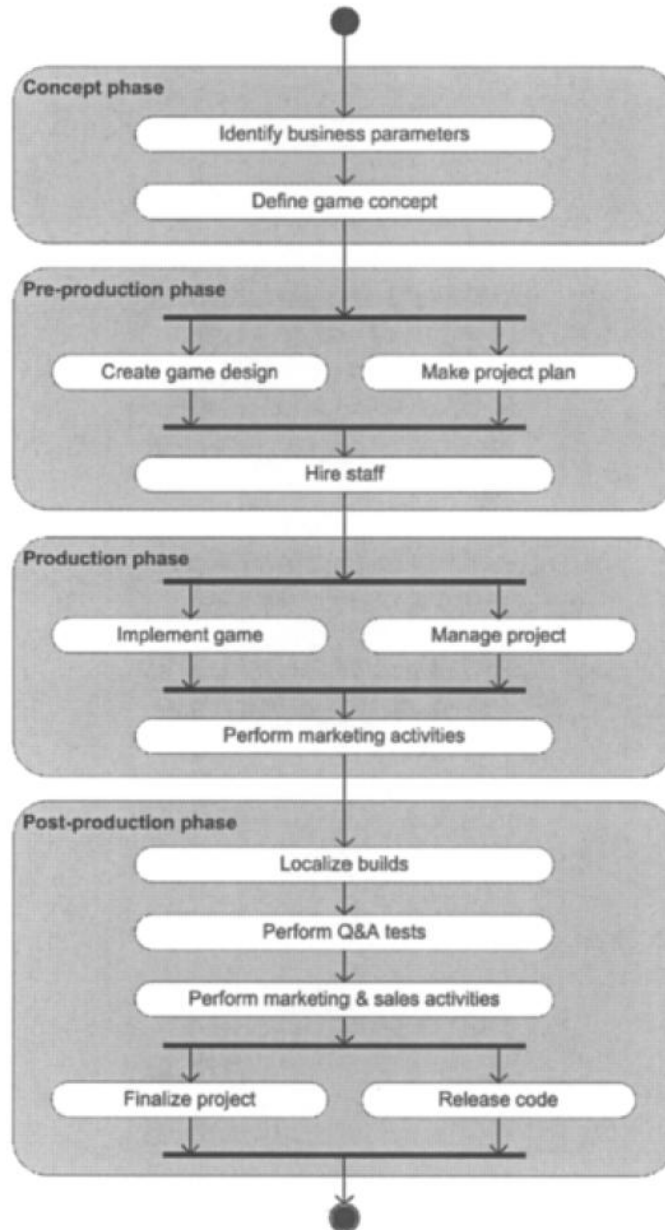


FIGURA 3.8 - Método de referencia para producción de videojuegos [13]

En la fase de *Concepto*, se identifican los primeros parámetros de negocio (presupuesto, recursos y análisis competitivo). Se define un concepto de juego y se presenta por medio de prototipos. Después en la fase de *Preproducción*, se desarrolla el documento de diseño de juego, el cual describe la historia, jugabilidad y requerimientos. Además, se crean el plan del proyecto y el plan de contratación. Alguien tiene que aprobar o invertir en la idea que se desarrolla en esta fase, antes de poder pensar acerca del proyecto y plan de contratación.

Después, en la fase de *Producción*, el juego se implementa y se crea la versión final del videojuego. Durante la implementación se administra el alcance de los cambios. Al final de esta fase, se desarrolla un demo con fines de promoción. Finalmente, la cuarta fase contiene las actividades típicas de *Posproducción*. Se necesita crear una versión *local* del juego de tal forma que pueda lanzarse en diferentes países, se realizan pruebas de aseguramiento de calidad y se crea material para promocionar el juego. El proyecto concluye con la liberación y entrega del juego.

De esta investigación bibliográfica se puede concluir lo siguiente:

Se analizaron tres métodos de producción de videojuegos documentados (*Game Development and Production* [19], *Introduction to Game Development* [20], y *The Game Production Handbook* [17]) y un método obtenido a través de un caso de estudio en una compañía de desarrollo de videojuegos, desarrollando meta-modelos con tablas de conceptos y actividades.

Se propone un modelo de referencia para la producción de videojuegos utilizando un enfoque formal para comparar los métodos de producción de videojuegos utilizando tablas de actividades y conceptos.

El método de referencia integra las diferentes perspectivas de los métodos evaluados y ***puede ser utilizado como una referencia completa sobre cómo podría estar organizado un proceso.***

3.4.5. Proceso de desarrollo de videojuegos basado en ISO/IEC 29110

El modelo de procesos de ISO/IEC 29110 Perfil Básico está dirigido a las pequeñas organizaciones de desarrollo de software. Está dividido en dos procesos, *Administración* e *Implementación*, con cuatro actividades de *Administración* y seis actividades de *Implementación*. En el capítulo 2 se ve el estándar ISO/IEC 29110 a detalle.

En [15] se discute la aplicabilidad del modelo de procesos ISO/IEC 29110 Perfil Básico en el contexto del desarrollo de videojuegos. Se analizó como funcionó el estándar en siete estudios de desarrollo de videojuegos y se identificaron un conjunto de cambios que podrían ayudar a esas organizaciones a aplicar el estándar en la práctica.

De acuerdo a este estudio, un proceso iterativo estándar para desarrollar tareas sería lo más útil para la industria de los videojuegos, orientado a las pequeñas y medianas organizaciones. Las organizaciones muy pequeñas operan con métodos altamente flexibles e informales, y no se benefician de los estándares tanto como las pequeñas o medianas organizaciones lo harían.

Propone algunos cambios para la aplicación de un modelo adecuado para las pequeñas organizaciones:

- El diseño no debe finalizar antes de la implementación.
- Los detalles variables deberían ser permitidos en el diseño inicial.
- Los cambios en el producto deben ser permitidos incluso cerca del final del proyecto.
- El estándar debe soportar enfoques iterativos.
- El estándar debe soportar actividades de entrega de fase, tales como pruebas de aceptación y actividades de planeación de fase como prototipos de prueba de conceptos.

En la siguiente FIGURA se muestra la propuesta del modelo con los cambios sugeridos:

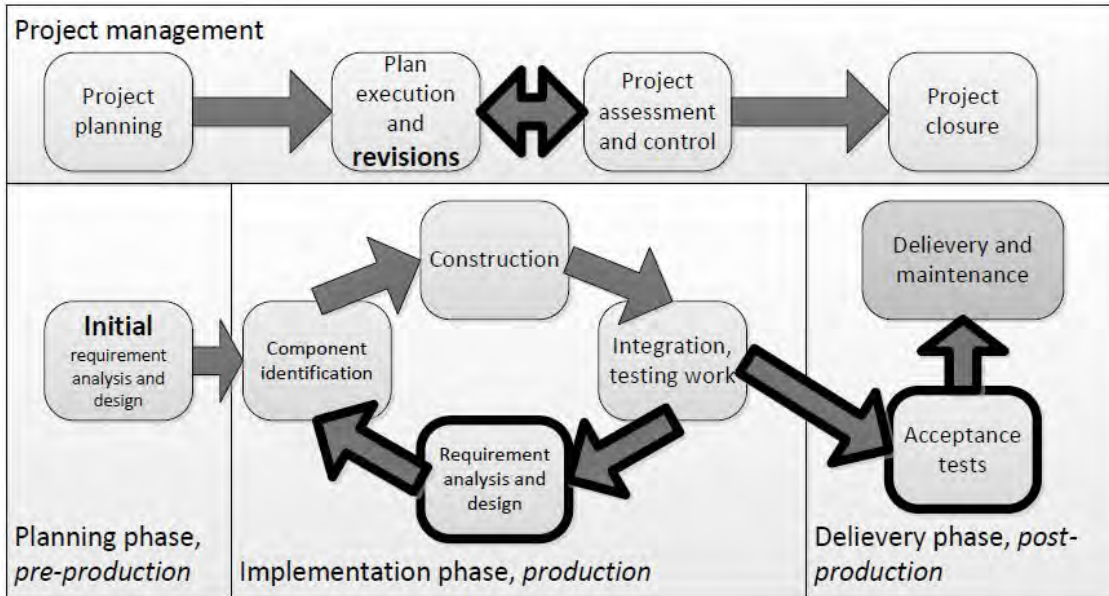


FIGURA 3.9 - El modelo propuesto

La principal diferencia entre el modelo de procesos ISO/IEC 29110 Perfil Básico y el modelo propuesto en este trabajo, se observa en el proceso de *Implementación de Software*. Para este proceso se proponen tres fases principales: *Preproducción (Planeación)*, *Producción (Implementación)* y *Posproducción (Entrega)*, y se hace énfasis en el desarrollo iterativo de la implementación.

La actividad *Análisis de Requerimientos y Diseño Iniciales*, incluye la experimentación de nuevas ideas con prototipos funcionales (en la fase de *Preproducción*), mientras que la fase de *Producción (Implementación)* tiene su propia actividad de *Análisis de Requerimientos y Diseño* para permitir revisiones al plan durante el desarrollo. Las actividades de finalización tales como *Pruebas de Aceptación* o identificación de *Assets* reusables están separadas en la fase de *Posproducción (Entrega)*.

El proceso de *Administración del Proyecto* se mantiene relativamente intacto, ya que se considera que es la parte más exacta del estándar. La planeación del proyecto se hace antes de que comience el desarrollo y en la actividad *Ejecución del Plan y Revisiones* se pueden hacer cambios al plan del proyecto para atender las necesidades de cambio durante el desarrollo. El cierre del

proyecto ocurre en la fase de *Posproducción*, cuando se hacen los ajustes finales y las actividades de aseguramiento de calidad para validar los requerimientos.

De este estudio se puede concluir lo siguiente:

Los resultados indican que los estudios pequeños no tienden a seguir ningún método de desarrollo sistemático, trabajando principalmente *ad-hoc*.

De acuerdo a este estudio la aplicación del estándar ISO/IEC 29110 podría ser posible, pero el modelo debería soportar el desarrollo iterativo, ya que en la industria de los videojuegos el diseño del producto es principalmente una guía, la cual cambia a través del proceso de desarrollo.

Sin embargo, el estándar ISO/IEC 29110 indica claramente que está destinado para ser utilizado por las pequeñas organizaciones (VSEs) para **establecer procesos para implementar cualquier enfoque o metodología de desarrollo**, incluyendo por ejemplo, ágil, evolutivo, incremental, desarrollo dirigido por pruebas, etc., basado en las necesidades del estudio o del proyecto.

Se considera que el proceso de *Administración del Proyecto* es la parte más exacta del estándar y sus actividades se mantienen prácticamente intactas. Se hace énfasis en que el estándar ISO/IEC 29110 Perfil Básico debería promover enfoques más ágiles.

Finalmente, los resultados de ese estudio podrían ayudar a las pequeñas organizaciones a entender cómo podrían aplicar el estándar ISO/IEC 29110 Perfil Básico, aunque el modelo propuesto debería ser evaluado por los estudios de desarrollo de videojuegos en proyectos reales.

3.5. Fases, actividades y productos del modelo de procesos propuesto

No existe un solo modelo de procesos de desarrollo para videojuegos, todos los procesos están escritos desde diferentes perspectivas, lo cual provoca una diferencia entre las actividades y conceptos que contienen.

Como resultado de la evaluación de los diferentes procesos de desarrollo de videojuegos presentados en este capítulo, en la siguiente TABLA se identifican las fases, actividades y productos **clave** que tendrá el modelo de procesos para desarrollo de videojuegos independientes propuesto en este trabajo.

TABLA 3.5 - Principales fases, actividades y productos del modelo de procesos propuesto

FASE	ACTIVIDADES	PRODUCTOS
CONCEPTO	<ul style="list-style-type: none"> - Identificar parámetros de negocio iniciales (presupuesto, recursos, análisis competitivo). - Definir concepto de juego (Identificar género, características clave, resumir historia, ambientación, plataforma y jugabilidad). - Construir prototipo basado en el concepto. - Obtener financiamiento para realizar el proyecto. 	<ul style="list-style-type: none"> - Documento de concepto - Prototipo - Pitch
PREPRODUCCIÓN	<ul style="list-style-type: none"> - Crear plan del proyecto (incluyendo plan de contratación y evaluación de riesgos). - Analizar los requerimientos del proyecto. - Crear diseño detallado del videojuego. - Crear diseño técnico del videojuego. 	<ul style="list-style-type: none"> - Plan del Proyecto - Plan de Contratación - Especificación de requerimientos - Documento de Diseño de Juego - Documento de Diseño Técnico
PRODUCCIÓN	<ul style="list-style-type: none"> - Administrar el proyecto (evaluar y controlar el proyecto). - Crear assets del videojuego (arte, música, animaciones, etc.). - Crear componentes de software. - Integrar el videojuego (producir versiones <i>alfa</i>, <i>beta</i> y <i>gold</i>). - Probar el videojuego (usabilidad, jugabilidad, rendimiento, etc.) - Realizar actividades de mercadotecnia. 	<ul style="list-style-type: none"> - Assets - Componentes de software - Game Builds - Plan de pruebas - Prototipos
POSPRODUCCIÓN	<ul style="list-style-type: none"> - Concluir el proyecto. - Entregar el producto (incluir versiones locales del juego). - Realizar autoevaluación (lecciones aprendidas). - Realizar actividades de mercadotecnia y ventas. - Realizar mantenimiento del videojuego (nuevo contenido, actualizaciones, etc.). 	<ul style="list-style-type: none"> - Producto entregado - Mantenimiento

4. CAPÍTULO 4. INTEGRACIÓN DEL MODELO DE PROCESOS PARA EL DESARROLLO DE VIDEOJUEGOS INDEPENDIENTES

4.1. Introducción

En los primeros tres capítulos se presentaron los fundamentos teóricos del presente trabajo, de los cuales se han identificado y descrito los siguientes puntos clave:

- Estado de la industria de los videojuegos en México.
- Marcos de trabajo para el modelo de procesos para el desarrollo de videojuegos.
- Principales diferencias entre la industria de los videojuegos y la industria del software.
- Principales dificultades en el desarrollo de videojuegos independientes.
- Características básicas en el desarrollo de videojuegos independientes.
- Trabajos relacionados y procesos de desarrollo de videojuegos.
- Principales fases, actividades y productos clave que tendrá el modelo propuesto, como resultado de la evaluación de trabajos relacionados y procesos de desarrollo de videojuegos.

Estos puntos clave sirven como base para la propuesta del modelo de procesos para el desarrollo de videojuegos independientes.

4.2. Metodología para la integración del modelo de procesos

Para integrar el modelo de procesos y adaptar la *ISO/IEC 29110 Perfil Básico* al desarrollo de videojuegos independientes, se estableció la siguiente metodología:

- Se definieron las dificultades, características básicas, fases y actividades que deberá cubrir el modelo de procesos para el desarrollo de videojuegos independientes.
- Se definieron los procesos y actividades principales que integran el modelo de procesos propuesto y su diagrama general.
- Se analizaron las actividades del proceso de *Administración del Proyecto* del estándar *ISO/IEC 29110 Perfil Básico* y se complementaron con las actividades identificadas del proceso de desarrollo de videojuegos.
- Se creó el diagrama del proceso de *Administración del Proyecto*. Este diagrama muestra el flujo de información entre las actividades de este proceso incluyendo los productos de trabajo más relevantes y la relación entre ellos.
- Se propusieron las tareas de cada actividad del proceso de *Administración del Proyecto* con el objetivo de cumplir con las características básicas y minimizar las principales dificultades en el desarrollo de videojuegos independientes, así como sus productos de entrada y salida.

- Se definieron las fases y actividades del proceso de *Implementación del Videojuego* del modelo de procesos propuesto. Se analizaron las actividades del proceso de *Implementación del Software* del estándar *ISO/IEC 29110 Perfil Básico*, y se complementaron con las fases y actividades clave identificadas del proceso de desarrollo de videojuegos.
- Se creó el diagrama del proceso de *Implementación del Videojuego*. Este diagrama muestra el flujo de información entre las actividades de este proceso incluyendo los productos de trabajo más relevantes y la relación entre ellos.
- Se propusieron las tareas de cada actividad del proceso de *Implementación del Videojuego* con el objetivo de cumplir con las características básicas y minimizar las principales dificultades en el desarrollo de videojuegos independientes, así como sus productos de entrada y salida. Los productos del proceso de *Implementación del Videojuego* se propusieron comparando y homologando los productos generados en diferentes procesos de desarrollo de videojuegos documentados. Se respetaron los nombres de los productos más importantes que se utilizan en la industria de los videojuegos.
- Se propusieron los roles del modelo de procesos, comparando y homologando los roles propuestos en diferentes procesos de desarrollo de videojuegos documentados y considerando el estudio sobre la industria de los videojuegos en México (capítulo 1).

4.3. Definición de características básicas, fases, actividades y dificultades que deberá cubrir el modelo de procesos.

Para las dificultades y características básicas en el desarrollo de videojuegos independientes, se establecieron los siguientes identificadores:

TABLA 4.1 - Dificultades en el proceso de desarrollo de videojuegos

ID	DIFICULTAD
D1	CREAR EL CONCEPTO DEL VIDEOJUEGO CON CAMBIOS CONSTANTES
D2	ANTICIPAR EL TIPO DE VIDEOJUEGOS QUE TENDRÁN ÉXITO
D3	OBTENER FINANCIAMIENTO PARA CONCRETAR LOS PROYECTOS
D4	PLANEAR EL PROYECTO Y DETERMINAR SU ALCANCE
D5	ANALIZAR Y VALIDAR REQUERIMIENTOS EN CONSTANTE CAMBIO
D6	DISEÑAR EL VIDEOJUEGO
D7	DISEÑAR TÉCNICAMENTE EL VIDEOJUEGO
D8	INNOVAR O CREAR VIDEOJUEGOS COMPLETAMENTE ORIGINALES
D9	ORGANIZAR UN EQUIPO MULTIDISCIPLINARIO

TABLA 4.2 - Características básicas en el desarrollo de videojuegos independientes

ID	CARACTERÍSTICA BÁSICA
C1	DESARROLLO DEL VIDEOJUEGO EN FASES
C2	CONCEPTO DE JUEGO Y PROTOTIPOS
C3	CONCIENCIA DEL MERCADO
C4	CONSIDERACIÓN DE FUENTES DE FINANCIAMIENTO
C5	AGILIDAD
C6	DESARROLLO ITERATIVO
C7	SOPORTE PARA CAMBIOS CONSTANTES EN EL VIDEOJUEGO
C8	VALORACIÓN DE LA EXPERIENCIA DEL JUGADOR
C9	SOPORTE PARA MANTENIMIENTO
C10	INTEGRACIÓN DE UN EQUIPO MULTIDISCIPLINARIO

Para las fases y actividades clave en el desarrollo de videojuegos independientes, se establecieron los siguientes identificadores:

TABLA 4.3 - Principales fases y actividades en el desarrollo de videojuegos

ID	FASE	ID	ACTIVIDAD GENERAL
F1	CONCEPTO	F1.1	Identificar parámetros de negocio iniciales (presupuesto, recursos, análisis competitivo).
		F1.2	Definir concepto de juego (Identificar género, características clave, resumir historia, ambientación, plataforma y jugabilidad).
		F1.3	Construir prototipo basado en el concepto.
		F1.4	Obtener financiamiento para realizar el proyecto.
F2	PREPRODUCCIÓN	F2.1	Crear plan del proyecto (incluyendo plan de contratación y evaluación de riesgos).
		F2.2	Analizar los requerimientos del proyecto.
		F2.3	Crear diseño detallado del videojuego.
		F2.4	Crear diseño técnico del videojuego.
F3	PRODUCCIÓN	F3.1	Administrar el proyecto (evaluar y controlar el proyecto).
		F3.2	Crear <i>assets</i> del videojuego (arte, música, animaciones, etc.).
		F3.3	Crear componentes de software.
		F3.4	Integrar el videojuego (producir versiones <i>alfa</i> , <i>beta</i> y <i>gold</i>).
		F3.5	Probar el videojuego (usabilidad, jugabilidad, rendimiento, etc.)
		F3.6	Realizar actividades de mercadotecnia.
F4	POSPRODUCCIÓN	F4.1	Concluir el proyecto.
		F4.2	Entregar el producto (incluir versiones locales del juego).
		F4.3	Realizar autoevaluación (lecciones aprendidas).
		F4.4	Realizar actividades de mercadotecnia y ventas.
		F4.5	Realizar mantenimiento del videojuego (nuevo contenido, actualizaciones, etc.).

Los identificadores se utilizan para denotar de forma simple que dificultades, características, fases y actividades se cubren al añadir una tarea, rol, producto al modelo de procesos propuesto.

Después de integrar el modelo de procesos, se definió su guía de implementación (anexo A) en la cual se detallan las actividades, tareas y productos a realizar.

4.4. Adaptación del Perfil Básico al desarrollo de videojuegos independientes

Para adaptar la *ISO/IEC 29110 Perfil Básico* al desarrollo de videojuegos independientes se analizó cada una de las actividades definidas en sus procesos y posteriormente fueron complementadas con actividades, productos, roles y procesos adicionales, con el principal propósito de cumplir con las características básicas y minimizar las dificultades en el desarrollo de videojuegos independientes.

4.4.1. Diagrama general del modelo de procesos propuesto

El modelo de procesos para el desarrollo de videojuegos independientes tiene dos procesos principales: *Administración del Proyecto (AP)* e *Implementación del Videojuego (IV)*.

El propósito del proceso de *Administración del Proyecto* es establecer y llevar a cabo de manera sistemática las tareas de un proyecto de *Implementación del Videojuego*, que permitan cumplir con los objetivos del proyecto en calidad, tiempo y costos esperados. Para este proceso se proponen las siguientes actividades:

TABLA 4.4 - Actividades del proceso de Administración del Proyecto

ID	ACTIVIDAD
AP.1	INICIO
AP.2	PLANEACIÓN DEL PROYECTO
AP.3	EJECUCIÓN DEL PLAN DEL PROYECTO
AP.4	EVALUACIÓN Y CONTROL DEL PROYECTO
AP.5	CIERRE DEL PROYECTO

El propósito del proceso de *Implementación del Videojuego* es la realización sistemática de las actividades de concepto, diseño, construcción, integración y pruebas para productos de software, nuevos o modificados, de acuerdo a los requerimientos especificados. Para este proceso se proponen las siguientes fases y actividades:

TABLA 4.5 - Fases y actividades del proceso de Implementación del Videojuego

FASE	ID	ACTIVIDAD
1. CONCEPTO	IV.1	Concepto del Videojuego
	IV.2	Análisis de Mercado y Financiamiento
2. PREPRODUCCIÓN	IV.3	Requerimientos y Diseño del Videojuego
	IV.4	Diseño Técnico del Videojuego
3. PRODUCCIÓN	IV.5	Construcción de Software
	IV.6	Integración y Pruebas de Software
4. POSPRODUCCIÓN	IV.7	Entrega del Producto
	IV.8	Mantenimiento del Videojuego

Ambos procesos están interrelacionados, como se ve en la FIGURA 4.1:

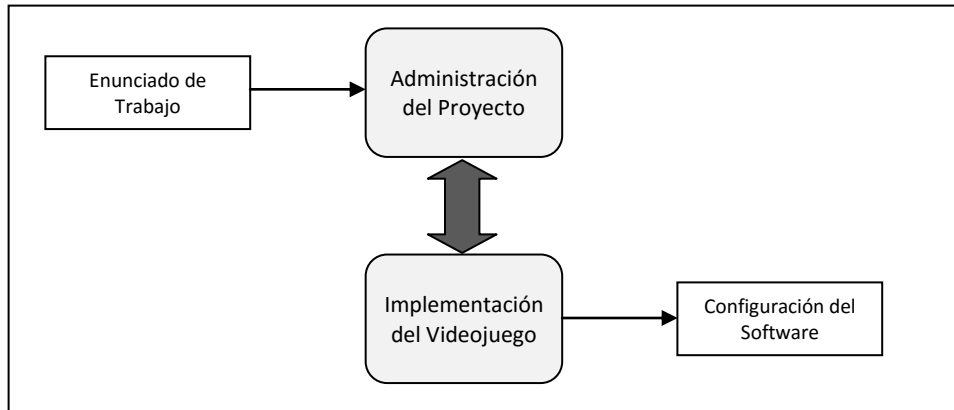


FIGURA 4.1 - Procesos principales del modelo de procesos para el desarrollo de videojuegos independientes

El proceso de *Administración del Proyecto* utiliza el producto de *Enunciado de Trabajo* del *Cliente* para elaborar el *Plan del Proyecto*. Las tareas de *Evaluación y Control del Proyecto* comparan el progreso del proyecto contra el *Plan del Proyecto* y se toman las acciones necesarias para eliminar desviaciones o incorporar cambios al *Plan del Proyecto*. La actividad de *Cierre del Proyecto* consiste en la entrega de la *Configuración del Software*, producida por la *Implementación del Videojuego*, y se obtiene la aceptación por parte del *Cliente* para formalizar el fin del proyecto. Un *Repositorio del Proyecto* se establece para guardar los productos de trabajo y controlar sus versiones durante el proyecto.

La ejecución del proceso de *Implementación del Videojuego* es dirigida por el *Plan del Proyecto*. El *Plan del Proyecto* guiará la ejecución de las actividades de *Concepto del Videojuego*, *Requerimientos y Diseño del Videojuego*, *Diseño Técnico del Videojuego*, *Construcción del Software*, *Integración y Pruebas de Software*, y *Entrega del Producto*.

Para eliminar los defectos del producto las tareas de verificación, validación y pruebas son incluidas en las actividades del flujo de trabajo.

La siguiente FIGURA muestra el diagrama general del modelo de procesos propuesto:

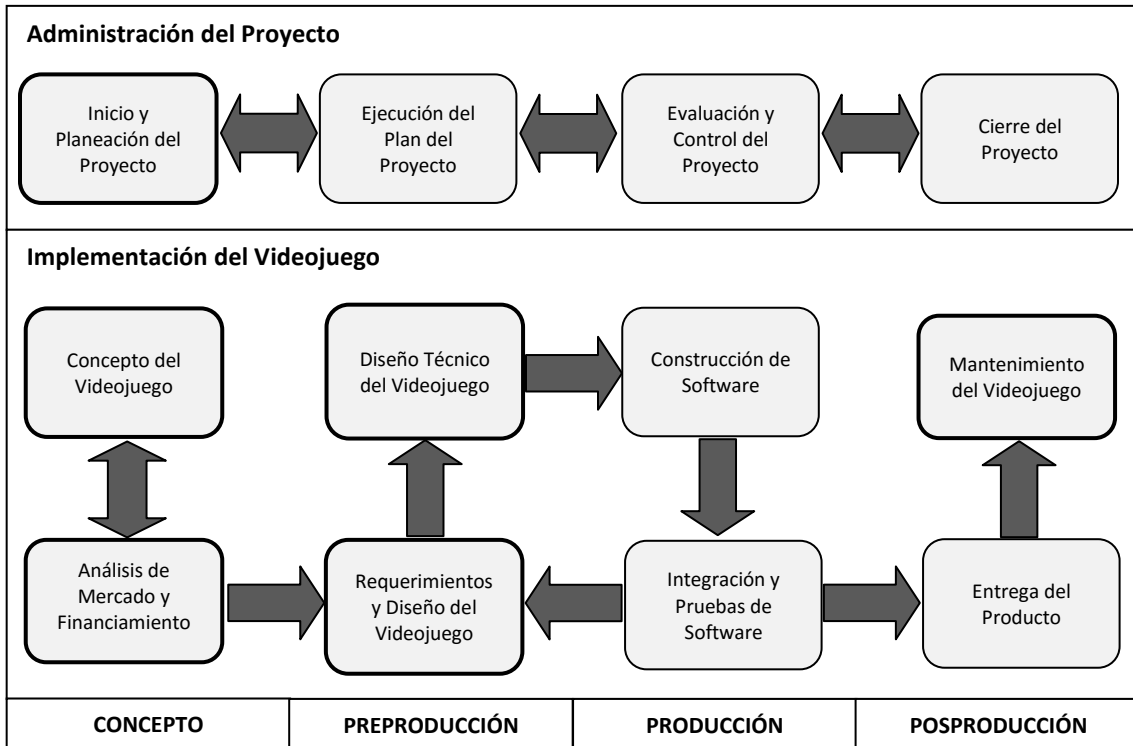


FIGURA 4.2 - Procesos y actividades del modelo de procesos

4.5. Proceso de Administración del Proyecto

El propósito del proceso de *Administración del Proyecto* es establecer y llevar a cabo de manera sistemática las tareas de un proyecto de *Implementación del Videojuego*, que permitan cumplir con los objetivos del proyecto en calidad, tiempo y costos esperados. Las actividades de este proceso se muestran en la siguiente TABLA.

TABLA 4.6 - Actividades del proceso de Administración del Proyecto

ID	ACTIVIDAD
AP.1	INICIO
AP.2	PLANEACIÓN DEL PROYECTO
AP.3	EJECUCIÓN DEL PLAN DEL PROYECTO
AP.4	EVALUACIÓN Y CONTROL DEL PROYECTO
AP.5	CIERRE DEL PROYECTO

Como en todas las industrias, la administración es importante. Una pobre administración puede afectar negativamente incluso a los mejores equipos, mientras que una excelente administración puede mejorar a cualquier equipo. A medida que los videojuegos aumentan en complejidad, el número de integrantes del equipo de trabajo aumenta y eso complica la administración. La buena

comunicación en un estudio es necesaria para el éxito. Algunas veces la falta de información administrativa impide identificar las áreas problemáticas [22].

Un error importante en la industria de los videojuegos es la mala definición del alcance del proyecto, que se agrava aún más debido a que se agrega nueva funcionalidad durante el desarrollo del videojuego, lo cual incrementa el tamaño del proyecto [22].

La administración de un proyecto de desarrollo de videojuegos consiste en la supervisión de equipos multidisciplinarios, no solamente programadores. Estos equipos producen *assets* de arte (modelos 3D, animaciones, efectos de audio, etc.) que deben ser integrados en la línea de producción del juego. Estos *assets* son muy importantes para el éxito de un videojuego. Y los programadores deben crear las herramientas de software para integrarlos [22].

En [23] se realiza un análisis de *postmortems* de diferentes estudios de desarrollo de videojuegos. Como resultado de este análisis, un gran porcentaje de problemas internos se relacionaron con asuntos de administración de proyectos clásica.

Así como la capacitación es importante para el equipo de desarrollo, la capacitación en temas de administración de proyectos también es vital [23].

En [15] se discute la aplicabilidad del modelo de procesos *ISO/IEC 29110 Perfil Básico* en el contexto del desarrollo de videojuego, el proceso de *Administración del Proyecto* se mantiene relativamente intacto, ya que se considera que es la parte más exacta del estándar.

La FIGURA 4.3 corresponde al diagrama general del proceso de *Administración del Proyecto*. En este diagrama se muestra el flujo de información entre las actividades propuestas para este proceso, incluyendo los productos de trabajo más relevantes y la relación entre ellos.

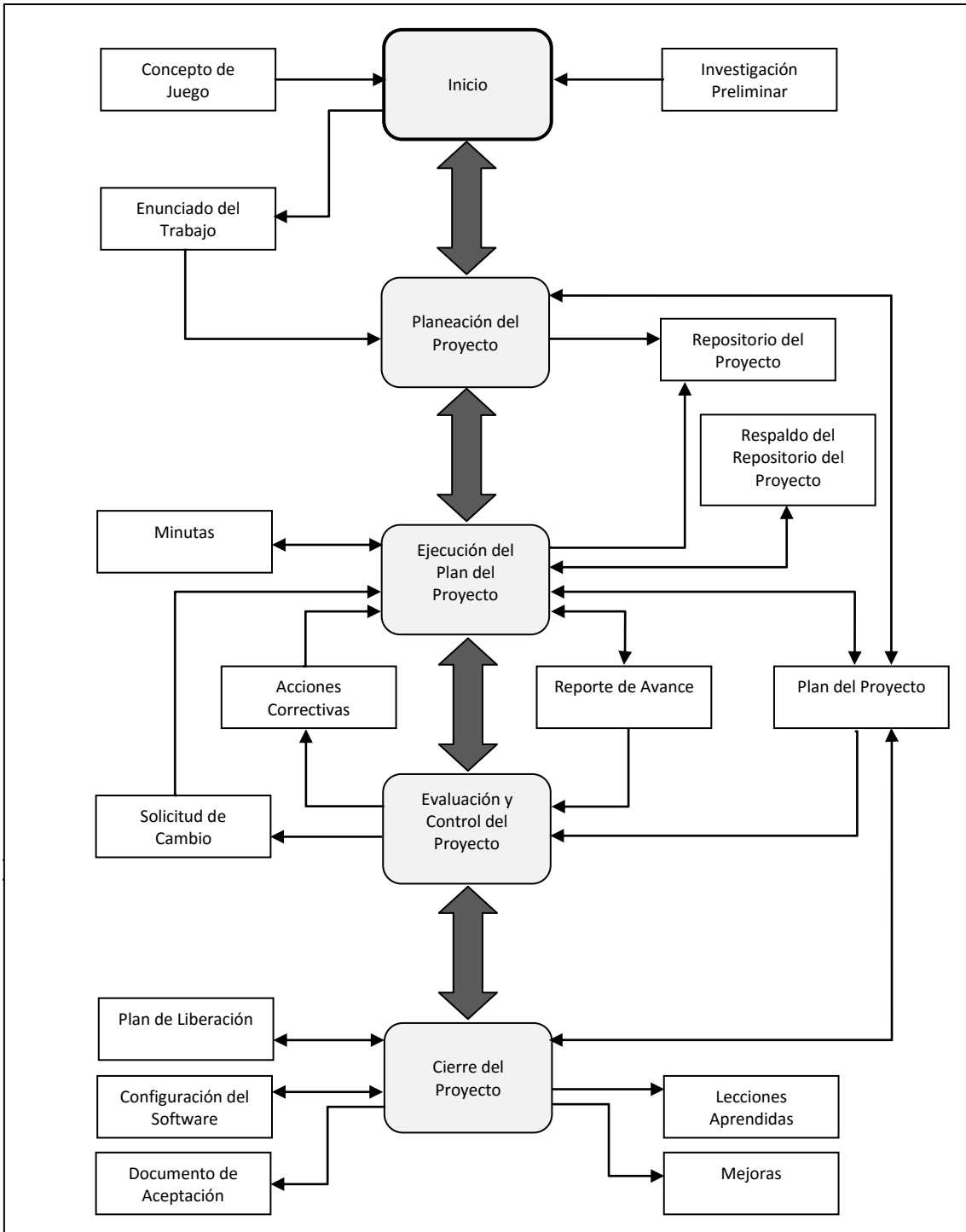


FIGURA 4.3 - Diagrama del proceso de Administración de Proyecto

En los siguientes puntos se definen, analizan y complementan cada una de las actividades del proceso de *Administración del Proyecto*.

4.5.1. Actividad Inicio

En el proceso de *Administración del Proyecto* de la *ISO/IEC 29110 Perfil Básico* no se define directamente la actividad *Inicio* del proyecto. Sin embargo, en [6] se establecen las siguientes precondiciones:

- Haber documentado el *Enunciado de Trabajo* del proyecto.
- Analizar la viabilidad del proyecto de manera previa.
- Equipo de trabajo, incluido el administrador del proyecto, debe haber sido asignado y entrenado.
- Contar con bienes, servicios e infraestructura disponible para iniciar el proyecto.

En la actividad *Inicio* propuesta se realiza el *Enunciado del Trabajo* del proyecto, con base en la *Investigación Preliminar*, el *Documento de Concepto de Juego* y el *Análisis Competitivo*. Finalmente es revisado por el *Equipo de Trabajo* y se obtiene la aprobación por parte del *Cliente*.

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Inicio*.

TABLA 4.7 - Tareas principales para la actividad Inicio

TAREA	DESCRIPCIÓN
1. REALIZAR ENUNCIADO DEL TRABAJO	Realizar el Enunciado del Trabajo con base en lo documentado en la investigación preliminar, el documento de concepto de juego y análisis competitivo. Se deberán definir los objetivos del videojuego, determinar el alcance inicial que tendrá el videojuego, identificar los stakeholders que intervendrán en el desarrollo del videojuego y finalmente analizar si el desarrollo del videojuego es factible.
2. APROBAR ENUNCIADO DE TRABAJO	Revisar Enunciado del Trabajo y obtener la aprobación del equipo de trabajo y finalmente por parte del cliente.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de F1.1. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.8 - Productos de trabajo de la actividad Inicio

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Documento de Concepto de Juego	Concepto del Videojuego
Investigación Preliminar	Análisis de Mercado y Financiamiento
Análisis Competitivo	Análisis de Mercado y Financiamiento
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Enunciado del Trabajo [aceptado]	Planeación del Proyecto
PRODUCTOS INTERNOS	
NOMBRE	
Enunciado del Trabajo [aprobado]	

4.5.2. Actividad Planeación del Proyecto

La *Planeación del Proyecto* es vital para obtener un producto de calidad diseñado dentro de los costos y los tiempos previstos.

Es importante identificar las tareas generales del proyecto, así como el esfuerzo y costo necesario para realizarlas. Las tareas a realizar deben ser priorizadas con base en el riesgo, valor y necesidad.

Una vez priorizadas las tareas se puede crear el *Plan del Proyecto* definiendo un calendario de tareas. Se pueden considerar los tiempos de publicación en las tiendas de videojuegos.

En [6] ya se establece el tener continua cercanía con el cliente para asegurar la calidad a través de validaciones de los entregables en las distintas actividades. También se establecen tareas de verificación dentro del equipo de trabajo de varios de los productos que son creados.

Se deben definir claramente los protocolos de entrega de los avances y productos finales. Se debe incluir la dinámica de las reuniones y revisiones, listas de validación, solicitud de cambios, minutas y responsables de las validaciones.

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Planeación del Proyecto*.

TABLA 4.9 - Tareas principales para la actividad Planeación del Proyecto

TAREA	DESCRIPCIÓN
1. DEFINIR LOS PROTOCOLOS DE ENTREGA	Definir los protocolos de entrega, dinámica de reuniones y revisiones. Se deberán establecer listas de validación, solicitudes de cambio y responsables de validación.
2. DEFINIR LOS PROTOCOLOS DE VERIFICACIÓN	Definir los protocolos de verificación que se emplearán dentro del equipo de trabajo e incorporarlo al plan del proyecto. Se deberá definir la dinámica de reuniones y verificaciones, establecer listas de verificación y solicitudes de cambio.
3. IDENTIFICAR TAREAS Y ESTIMAR SU DURACIÓN	Identificar las tareas específicas a realizar para producir elementos entregables del videojuego (documentos, assets, <i>game builds</i>). Así como establecer la duración estimada para realizar cada tarea.

4. IDENTIFICAR RECURSOS	Identificar y documentar los recursos: humanos, materiales, equipo y herramientas, estándares, incluyendo la capacitación requerida por el equipo de trabajo. Se deberán incluir las fechas en el calendario cuando sean requeridos los recursos y las fechas de capacitación (si es requerido).
5. ESTABLECER COMPOSICIÓN DEL EQUIPO DE TRABAJO	Establecer la composición del equipo de trabajo, asignando los roles y responsabilidades acordes a los Recursos.
6. GENERAR CALENDARIO DE TAREAS	Asignar las fechas de inicio y fin estimadas para cada tarea para generar el calendario de tareas considerando los recursos asignados, la prioridad, secuencia y dependencia de las tareas.
7. CALCULAR ESFUERZO Y COSTO	Calcular y documentar el esfuerzo y costo, así como el calendario de flujo de fondos estimado del proyecto.
8. IDENTIFICAR RIESGOS	Identificar los riesgos que pueden afectar el proyecto.
9. DEFINIR ESTRATEGIA DE CONTROL DE VERSIONES	Definir y documentar la estrategia de control de versiones. Se deberá definir control de versiones para la documentación y establecer el repositorio.

También se debe definir claramente dentro del grupo de trabajo los protocolos de verificación para los avances y productos finales.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *D4*, *C5* y *F2.1*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.10 - Productos de trabajo de la actividad Planeación del Proyecto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Enunciado del Trabajo	Inicio
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Plan del Proyecto	Administración del Proyecto (AP) Implementación del Videojuego (IV)
PRODUCTOS INTERNOS	
NOMBRE	
Protocolos de Entrega	
Protocolos de Verificación	
Tareas	
Duración Estimada	
Recursos	
Composición del Equipo de Trabajo	
Calendario de Tareas	
Esfuerzo y Costo estimado	
Calendario de Flujo de Fondos	
Plan de Contratación	
Identificación de Riesgos	
Estrategia de Control de Versiones	
Descripción del producto	
Alcance	
Objetivos	

Entregables
Repositorio del Proyecto

4.5.3. Actividad Ejecución del Plan del Proyecto

En esta actividad se definen tareas para implementar el plan documentado en el proyecto. En [6] ya se establece el monitoreo de la ejecución del plan del proyecto para realizar un reporte de avance. Además, se establece el realizar reuniones con el cliente y atender solicitudes de cambio hechas por el cliente o dentro del mismo equipo de trabajo.

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Ejecución del Plan del Proyecto*.

TABLA 4.11 - Tareas principales para la actividad Ejecución del Plan del Proyecto

TAREA	DESCRIPCIÓN
1. MONITOREAR TAREAS	Llevar a cabo el plan del proyecto según las tareas establecidas y priorizadas. Se deberán monitorear las tareas según lo establecido en el calendario de tareas del proyecto.
2. ANALIZAR Y EVALUAR EL IMPACTO DE LOS CAMBIOS	Analizar y evaluar el impacto en costo, tiempo y tecnológico de la solicitud de cambio que es propuesta por el equipo de trabajo o cliente. Si se afectan acuerdos previos necesita ser negociada por ambas partes.
3. DIRIGIR REUNIONES CON EL EQUIPO DE TRABAJO	Dirigir reuniones de revisión con el equipo de trabajo para identificar problemas, revisar riesgos, registrar acuerdos, nuevas ideas para el videojuego y darles seguimiento hasta su conclusión.
4. REALIZAR REUNIONES CON EL CLIENTE	Realizar reuniones frecuentes con el cliente para obtener validaciones, atender solicitud de cambio y registrar nuevos acuerdos.

Para esta actividad primero se propone dirigir reuniones dentro del equipo de trabajo para llevar un monitoreo y registro de avances con el objetivo de detectar los posibles problemas y riesgos que se puedan presentar y poder brindar una solución y seguimiento a estos.

Posteriormente se propone tener reuniones frecuentes con el cliente para obtener la validación de elementos entregables y atender las solicitudes de cambio. En estas reuniones se presentan reportes de avances.

En esta actividad se propone añadir elementos para cubrir principalmente las características C5, C6, C7 y F3.1. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.12 - Productos de trabajo de la actividad Ejecución del Plan del Proyecto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [aceptado]	Planeación del Proyecto
Solicitud de Cambio [propuesta]	Evaluación y Control del Proyecto
Acciones Correctivas	Evaluación y Control del Proyecto
Estrategia de Control de Versiones	Planeación del Proyecto
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Reporte de Avance [actualizado]	Evaluación y Control del Proyecto
Solicitud de Cambio [evaluada]	Evaluación y Control del Proyecto
Plan del Proyecto [actualizado]	Administración del Proyecto (IV) Implementación del Videojuego (IV)
Retroalimentación del Hito	Evaluación y Control del Proyecto
PRODUCTOS INTERNOS	
NOMBRE	
Minutas	
Respaldo del Repositorio del Proyecto	

4.5.4. Actividad Evaluación y Control del Proyecto

En esta actividad se evalúa el desempeño del plan contra los compromisos documentados. En [6] ya se establece evaluar el progreso del proyecto respecto al plan del proyecto con el fin de establecer acciones para corregir desviaciones o problemas que amenacen el cumplimiento del plan del proyecto.

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Evaluación y Control del Proyecto*.

TABLA 4.13 - Tareas principales para la actividad Evaluación y Control del Proyecto

TAREA	DESCRIPCIÓN
1. EVALUAR EL PROGRESO DEL PROYECTO	Evaluar el progreso del proyecto con respecto al Plan del Proyecto comparando tareas reales contra tareas planeadas, resultados reales contra objetivos establecidos en el proyecto, costos reales contra el presupuesto estimado, tiempo utilizado contra el tiempo programado, riesgo real contra el identificado previamente.
2. ESTABLECER ACCIONES CORRECTIVAS	Establecer acciones correctivas e identificar nuevos riesgos.
3. IDENTIFICAR CAMBIOS A REQUERIMIENTOS	Identificar cambios a requerimientos al Plan del Proyecto para hacer frente a desviaciones importantes, potenciales riesgos o problemas relativos al cumplimiento del plan. Proponer solicitud de cambio y dar seguimiento hasta su conclusión.

Se deberán establecer acciones correctivas e identificar cambios en los requerimientos o en el diseño del videojuego para hacer frente a desviaciones importantes.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *D4*, *D5*, *C7* y *C8*, *F3.1*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.14 - Productos de trabajo de la actividad Evaluación y Control del Proyecto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [aceptado]	Planeación del Proyecto
Reporte de Avance [actualizado]	Ejecución del Plan del Proyecto
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Acciones Correctivas	Ejecución del Plan del Proyecto
Solicitud de Cambio [propuesta]	Ejecución del Plan del Proyecto
PRODUCTOS INTERNOS	
NOMBRE	
Reporte de Avance [evaluado]	
Documento de Seguimiento de Assets	

4.5.5. Actividad Cierre del Proyecto

En esta actividad se proporciona la documentación y productos del proyecto de acuerdo con los protocolos de entrega.

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Cierre del Proyecto*.

TABLA 4.15 - Tareas principales para la actividad Cierre del Proyecto

TAREA	DESCRIPCIÓN
1. REALIZAR PLAN DE LIBERACIÓN	Realizar el plan de liberación de la versión final del software (gold master build). Se deberá definir publicación del videojuego, elementos publicitarios, costo final del videojuego, definir los medios de distribución y fechas aproximadas de publicación
2. PUBLICAR VIDEOJUEGO	Llevar a cabo la publicación del videojuego de acuerdo al Plan de Liberación. Se deberán realizar actividades de mercadotecnia y promoción del videojuego, así como publicar el videojuego de acuerdo a los requerimientos de las políticas de publicación en mercados.
3. FORMALIZAR CONCLUSIÓN DEL PROYECTO	Formalizar conclusión del proyecto de acuerdo a los protocolos de entrega.
4. REALIZAR AUTOEVALUACIÓN EN EL GRUPO DE TRABAJO	Realizar una autoevaluación en el grupo de trabajo para obtener retroalimentación y documentar las lecciones aprendidas durante el desarrollo del proyecto. Se deberán establecer posibles procesos, tareas, actividades o mecanismos de mejora para futuros proyectos.

Se propone como primera tarea para esta actividad incluir un Plan de Liberación del videojuego, en el cual se definen los medios de difusión, elementos publicitarios, costo del videojuego, medios de distribución y fechas aproximadas de publicación.

Adicionalmente se propone al finalizar el proyecto realizar la tarea de autoevaluación dentro del equipo de trabajo con el fin de obtener retroalimentación y registrar las lecciones aprendidas. Se pueden proponer procesos, actividades, tareas o mecanismos de mejora para futuros proyectos.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *F4.1*, *F4.2*, *F4.3*, *F4.4*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.16 - Productos de trabajo de la actividad Cierre del Proyecto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [aceptado]	Planeación del Proyecto
Protocolos de Entrega	Planeación del Proyecto
Software (Gold Master Build) [validado]	Integración y Pruebas de Software
Políticas de Publicación en Mercados	Análisis de Mercado
Configuración del Software [entregada]	Entrega del Producto
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Software (Gold Master Build) [disponible]	Tienda de Aplicaciones
Configuración del Software [aceptada]	Cliente
Lecciones Aprendidas	Equipo de Trabajo
Mejoras	Equipo de Trabajo
PRODUCTOS INTERNOS	
NOMBRE	
Plan de Liberación	

4.6. Proceso de Implementación del Videojuego

El propósito del proceso *de Implementación del Videojuego* es la realización sistemática de las actividades de concepto, diseño, construcción, integración y pruebas para productos de software, nuevos o modificados, de acuerdo a los requerimientos especificados. Para este proceso se proponen las siguientes fases y actividades:

TABLA 4.17 - Fases y actividades del proceso de Implementación del Videojuego

FASE	ID	ACTIVIDAD
1. CONCEPTO	IV.1	Concepto del Videojuego
	IV.2	Análisis de Mercado y Financiamiento
2. PREPRODUCCIÓN	IV.3	Requerimientos y Diseño del Videojuego
	IV.4	Diseño Técnico del Videojuego
3. PRODUCCIÓN	IV.5	Construcción de Software
	IV.6	Integración y Pruebas de Software
4. POSPRODUCCIÓN	IV.7	Entrega del Producto
	IV.8	Mantenimiento del Videojuego

En los siguientes puntos se definen, analizan y complementan cada una de las actividades del proceso de *Implementación del Videojuego*.

La FIGURA 4.4 corresponde al diagrama general del proceso de *Implementación del Videojuego*. En este diagrama se muestra el flujo de información entre las actividades propuestas para este proceso, incluyendo los productos de trabajo más relevantes y la relación entre ellos.

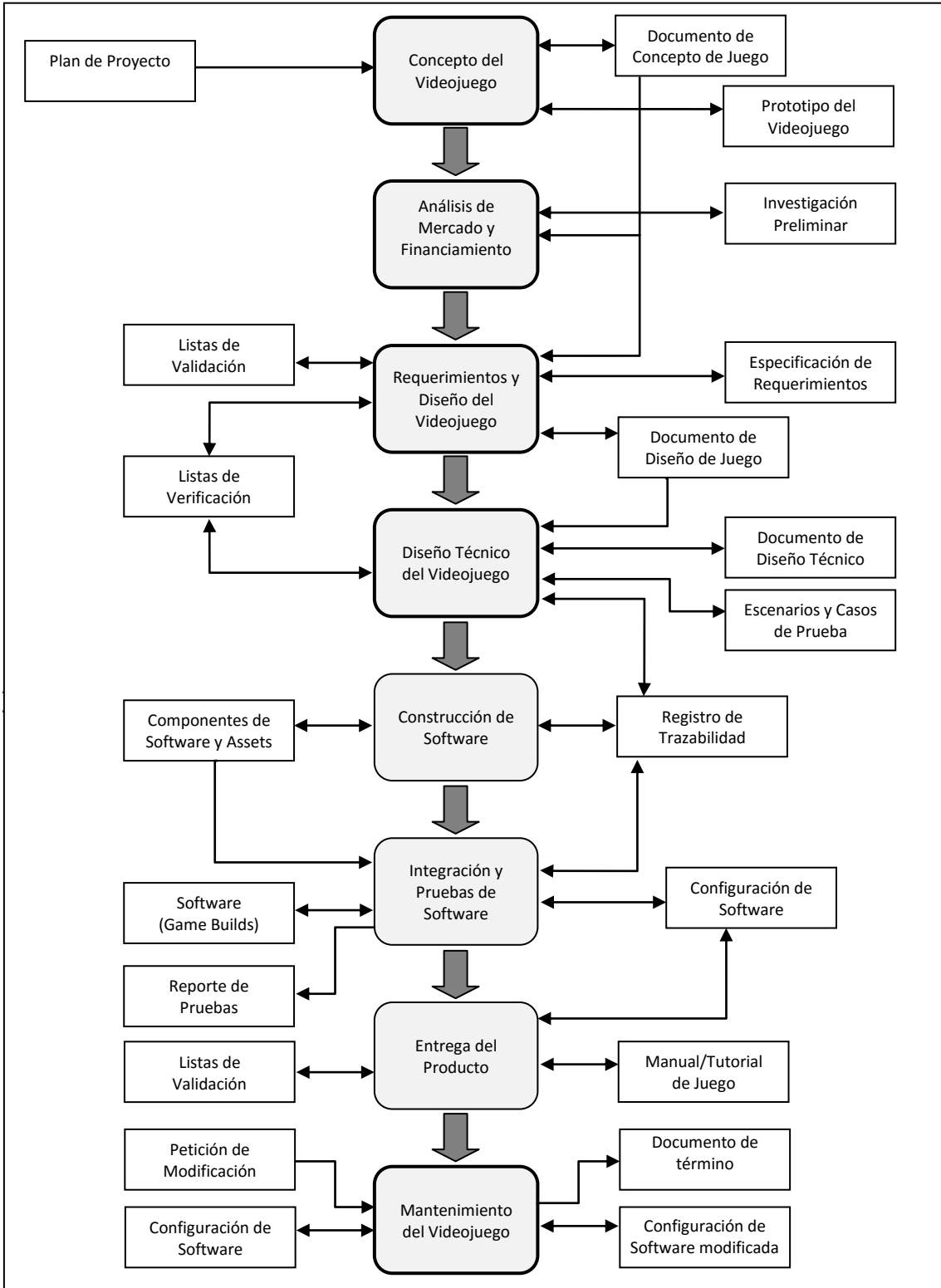


FIGURA 4.4 - Diagrama del proceso de Implementación del Videojuego

4.6.1. Fase de Concepto

En la fase de *Concepto* se llevará a cabo la concepción de la idea general del videojuego, identificando los elementos fundamentales que lo caracterizarán, finalizando en un diseño conceptual del mismo y en un prototipo. También en esta fase se propone llevar a cabo un análisis de mercado para evaluar los videojuegos similares que estén publicados y probar que hay un mercado potencial para el videojuego que se quiere construir.

Finalmente, para que el proyecto pueda llevarse a cabo, se considerará un modelo de financiamiento con el objetivo de que alguien apruebe e invierta en la idea que será desarrollada en esta fase.

La fase de concepto en el proceso de desarrollo de videojuegos es diferente y más compleja que en otro tipo de productos de software, debido a que la atmósfera del juego tiene que ser definida en un concepto inicial que podría ser muy diferente a la versión final del juego.

Además, es probable que el concepto inicial cambie constantemente durante esta fase y mucho trabajo artístico y prototipos de jugabilidad se tengan que crear incluso durante esta fase.

4.6.2. Actividad Concepto del Videojuego

En esta actividad se crea el diseño conceptual del videojuego identificando los elementos fundamentales que lo caracterizarán (género, plataforma, resumen de la historia, mecánicas principales, jugabilidad (gameplay), ambientación, arte conceptual y elementos de audio), finalizando en la construcción de un prototipo con base en el diseño conceptual.

Lo primero que debe crearse en esta actividad es el *Concepto Inicial*. Los elementos principales son los siguientes:

- **Idea.** Identificar la idea de un juego es el primer paso en el proceso de desarrollo de videojuego. Un buen juego comienza con una fuerte idea, la cual viene de entender lo que *motiva* a la gente a jugar un videojuego [21].
- **Motivación.** Identificar las razones por las cuáles el jugador jugará el videojuego. Jugar juegos satisface tres necesidades psicológicas de la gente: *realización (achievement)*, *reconocimiento* y *satisfacción*. Además, los juegos satisfacen una de las más importantes necesidades emocionales de la gente: la necesidad de conectarse e *interactuar con otros*. Actualmente los videojuegos permiten la interacción social con personas de todo el mundo mediante el juego en línea [21].
- **Público objetivo.** Para desarrollar un videojuego efectivo para una determinada audiencia, es esencial familiarizarse con las cosas que le gustan y no le gustan a este público. Los jugadores se pueden clasificar según su edad (preescolares, niños, adolescentes, adultos), género (hombres, mujeres), basado en las necesidades y niveles motivacionales (casuales, *hardcore*, intermedio, profesional), basados en factores demográficos (localización geográfica, nacionalidad, cultura, religión) [21].

- **Género del videojuego.** Los videojuegos están clasificados en varios géneros basados en el tipo de interacción del jugador con el juego (*gameplay*). Se decide el género del juego considerando la motivación y el público objetivo identificado para el juego. Algunos géneros importantes son: *acción, aventura, deportes, simulación, juegos de rol, fantasía, cartas, de mesa* [21].
- **Plataforma.** Los videojuegos pueden ser jugados en una gran variedad de plataformas, en línea, de modo local, por un solo jugador o por muchos jugadores simultáneamente. Algunas de las plataformas más comunes que se pueden seleccionar son: *arcade⁵, consola doméstica, consola portátil, PC, dispositivos móviles y otras (televisor, smartwatch, realidad aumentada, realidad virtual)*.
- **Estatus del juego.** Basado en el estatus, los juegos pueden ser de dos tipos: *En línea (online)*, requieren de una conexión a Internet para jugarlos a través de un servicio y *local (offline)*, no requieren de una conexión a Internet, pueden ser jugados de manera local [21].
- **Número de jugadores.** Basado en el número de jugadores, los juegos pueden ser de dos tipos: *Un solo jugador (single player)* y *multijugador (multiplayer)*, en donde muchas personas pueden jugar en el mismo ambiente simultáneamente [21].

Este *Concepto Inicial* deberá ser evaluado y aprobado para comenzar a crear el *Concepto Detallado (High Concept)*. Los elementos principales del *Concepto Detallado* son los siguientes:

- **Declaración de misión.** La *declaración de misión* del juego describe el objetivo principal del juego, los retos involucrados en la consecución de ese objetivo y los recursos disponibles para el jugador que ayudarán a conseguirlo [21].
- **Ambientación del videojuego.** Definir los escenarios y ambientación (ciencia ficción, mundo real, fantasía, histórico). La ambientación influye en la apariencia del videojuego (*look and feel*), tema visual, objetos, personajes, interfaz, tema de audio y otros elementos que son parte del juego.
- **Línea argumental.** La línea argumental es el conflicto matriz de la historia explicado de forma breve. Generalmente no excede los cinco o seis renglones, donde se incluye presentación, desarrollo y solución del conflicto. No es necesario mayor detalle en esta fase [21].
- **Modelo de monetización.** Es la forma en la que se va a obtener dinero a través del videojuego. Al momento de elegir un modelo de monetización (*anuncios in-game, ad-servers, economía virtual, etc.*) se deberá tomar en cuenta el alcance, la plataforma, así como los objetivos del proyecto. El modelo de monetización impacta el diseño del videojuego [24].

⁵ *Arcade* está vinculado a salones recreativos, atracciones o parques temáticos. Se crean para usarse con volantes, vehículos, naves, pistolas, etc. Pensados para partidas cortas e intensas y tienen un mercado más reducido.

- **Jugabilidad (gameplay).** La *jugabilidad (gameplay)* es la forma de interactuar con el videojuego, es decir, aquello que experimenta el jugador durante el transcurso del juego. Por eso, cada género tiene una jugabilidad específica, que dependerá también de otros factores como los aspectos gráficos o visuales. También se puede definir como "la dinámica del juego resultante de la interacción entre reglas y la geografía del videojuego" [24].
- **Mecánicas de juego principales.** Las *mecánicas* son los procedimientos y reglas del juego. Las mecánicas describen las metas del juego, cómo los jugadores pueden y no pueden intentar lograrlas, y lo que pasa cuando lo intentan. Las mecánicas son cruciales para la jugabilidad (gameplay) [31].
- **Arte conceptual.** El arte conceptual muestra cuál será la apariencia de los elementos visuales del videojuego antes de que sean producidos para el juego. En la FIGURA 4.5 se muestra un ejemplo del arte conceptual de un personaje.



FIGURA 4.5 - Arte conceptual del personaje Jago del juego Killer Instinct (2013) [40]

- **Elementos de Audio.** En esta fase conceptual se deberán considerar los tipos de efectos de sonido que se requieren, cuáles funcionan mejor, qué personajes tendrán voz, cómo funcionan los diálogos de los personajes, qué parte del juego deberá incluir música y qué tipo de música funciona mejor con el juego [24].
- **Modos de distribución del videojuego.** Se refiere a las formas en las que se puede distribuir el videojuego. Los videojuegos independientes a menudo se centran en la innovación y se basan en la *distribución digital*. Estos videojuegos han visto un aumento en los últimos años, principalmente debido a los nuevos métodos de distribución en línea y herramientas de desarrollo.

Después de crear el *Concepto Detallado*, se deberá construir un *Prototipo* con base en el *Concepto Detallado* del videojuego.

Las tareas propuestas para esta actividad se obtienen como resultado de la comparación y homologación de las tareas descritas para crear el *Concepto del videojuego* en tres procesos documentados [13], [17] y [21].

En la siguiente TABLA se muestran las tareas propuestas a realizar para definir el *Concepto del Videojuego*.

TABLA 4.18 - Tareas principales para la actividad Concepto del Videojuego

TAREA	DESCRIPCIÓN
1. IDEAR EL JUEGO	Identificar la idea principal del videojuego que se quiere construir.
2. CREAR CONCEPTO INICIAL	Identificar motivación, público objetivo, género del videojuego, plataforma, estatus del juego (online, offline) y número de jugadores.
3. REALIZAR ANÁLISIS COMPETITIVO	Identificar fortalezas y debilidades de la competencia del videojuego, oportunidades de mercado y amenazas que podrían impactar el éxito del juego en el mercado. Considerando el concepto inicial del videojuego
4. APROBAR CONCEPTO INICIAL	Aprobar el concepto inicial para comenzar a crear el concepto detallado del juego.
5. CREAR CONCEPTO DETALLADO (HIGH CONCEPT)	Definir declaración de misión, ambientación del videojuego, resumen de la historia (línea argumental), modelo de monetización, jugabilidad (gameplay), mecánicas de juego principales, arte conceptual, elementos de audio y modos de distribución del videojuego.
6. CONSTRUIR PROTOTIPO	Construir un prototipo del videojuego con base en el concepto detallado del juego. Este prototipo servirá como base o estándar para el juego completo en fases posteriores.

En el desarrollo de un videojuego es común que el diseño conceptual no se acerque a la versión final del videojuego. Los prototipos se utilizan como parte del proceso de diseño para permitirle a los ingenieros y diseñadores explorar alternativas en el diseño, probar teorías y confirmar el rendimiento antes de comenzar con la producción del videojuego.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *D1 D2, C2, C3, F1.2, F1.3*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.19 - Productos de trabajo de la actividad Concepto del Videojuego

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Planteamiento del Videojuego	Estudio, Publisher
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Documento de Concepto de Juego	Análisis de Mercado y Financiamiento Requerimientos y Diseño del Videojuego Inicio y Planeación del Proyecto
Archivos de Referencia	Requerimientos y Diseño del Videojuego
Prototipo del Videojuego	PREPRODUCCIÓN PRODUCCIÓN
Estándares de Plataforma	Requerimientos
PRODUCTOS INTERNOS	
NOMBRE	
Generación de Idea	
Concepto Inicial	
Análisis Competitivo	

4.6.3. Actividad Análisis de Mercado y Financiamiento

Análisis de Mercado. En la actividad *Análisis de Mercado* se proponen tareas para investigar y evaluar los videojuegos similares que estén publicados, con el objetivo de probar que hay un mercado potencial para el videojuego que se quiere construir. Esta actividad se realiza en paralelo con la actividad *Concepto del Videojuego*.

Como primera tarea se propone investigar las *Políticas de Publicación de Mercados*, con el objetivo de minimizar los problemas a la hora de distribuir el videojuego. Las políticas de publicación podrían impactar el diseño del videojuego debido a que no todas las tiendas permiten publicar cualquier videojuego. Por lo tanto, es importante tomarlas en cuenta en esta fase de *Concepto*.

A partir de la generación del diseño conceptual del videojuego, se deberá realizar de manera preliminar una evaluación del mercado potencial (tipo de videojuegos más jugados, plataformas más comunes, modelos de monetización, etc.), evaluación técnica y evaluación del negocio. Con esta investigación preliminar se deben tener elementos para detectar las necesidades de los jugadores, establecer el modelo de negocio, evaluar viabilidad técnica y definir el alcance del videojuego a desarrollar.

También se propone realizar un *Análisis FODA (Fortalezas, Oportunidades, Debilidades y Amenazas)* del concepto propuesto (identificar las *Fortalezas* y *Debilidades* del concepto del videojuego, *Oportunidades* de mercado y *Amenazas* que podrían impactar el éxito del videojuego). Además, se deberá realizar un análisis competitivo. En la siguiente FIGURA se muestra un ejemplo de un *Análisis FODA* de la estrategia de mercadotecnia de Nintendo.



FIGURA 4.6 - Análisis FODA (SWOT) de la estrategia de mercadotecnia de Nintendo [39]

En la siguiente TABLA se muestran las tareas propuestas a realizar en la actividad *Análisis de Mercado*.

TABLA 4.20 - Tareas principales para la actividad Análisis de Mercado

TAREA	DESCRIPCIÓN
1. INVESTIGAR POLÍTICAS DE PUBLICACIÓN	Investigar políticas de publicación de mercados.
2. REALIZAR Y DOCUMENTAR INVESTIGACIÓN PREELIMINAR	Investigar y evaluar el mercado de videojuegos. Investigar y evaluar videojuegos similares que estén publicados y probar que hay un mercado para el videojuego que se quiere construir. También se deberán investigar y evaluar aspectos legales relacionados con derecho de autor, licencias, políticas de publicación de mercados potenciales, etc.
3. REALIZAR ANÁLISIS FODA	Identificar las fortalezas y debilidades del concepto del videojuego, oportunidades de mercado y amenazas que podrían impactar el éxito del videojuego.
4. REALIZAR ANÁLISIS COMPETITIVO	Identificar las fortalezas y debilidades de la competencia del videojuego.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *F1.1*, *D2* y *C3*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.21 - Productos de trabajo de la actividad Análisis de Mercado

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Documento de Concepto de Juego	Concepto del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Políticas de Publicación de Mercados	Requerimientos y Diseño del Videojuego Cierre del Proyecto
Investigación Preliminar	Financiamiento Inicio y Planeación del Proyecto
Análisis Competitivo	Financiamiento Inicio y Planeación del Proyecto
PRODUCTOS INTERNOS	
NOMBRE	
Análisis FODA	

Financiamiento. Uno de los problemas principales para los estudios independientes, es la falta de financiamiento para concluir los proyectos. Es por eso que en esta actividad se propone definir un modelo de financiamiento con el objetivo de que alguien apruebe e invierta los recursos necesarios para desarrollar el proyecto.

No solo en los emprendimientos tecnológicos, sino en todas las industrias hay un momento clave: ¿cómo conseguir el dinero? Actualmente existen muchas personas dispuestas a invertir a través de mecanismos innovadores que no existían años atrás, y también organizaciones dispuestas a apostar en negocios y acelerar su puesta en marcha [24].

Como primera tarea se propone determinar el modelo de financiamiento. Una opción de financiamiento interesante para los estudios independientes es *crowdfunding*. El *crowdfunding* consiste en la obtención de un aporte colectivo a través de Internet de personas conocidas o desconocidas para que el proyecto salga adelante. La forma de funcionamiento es simple: una persona que busca inversores establece un perfil en sitios de *crowdfunding* como *Kickstarter* o *Indiegogo*, describiendo en detalle el proyecto por el que está solicitando fondos. Todo interesado se puede convertir en un posible socio, inversor o simplemente donante para lograr el monto objetivo que se haya fijado [24].

Si se alcanza el monto objetivo fijado, entonces el sistema entregará el dinero recaudado y no será necesario devolverlo ni entregar participaciones de la empresa. Si no se logra llegar al monto fijado se le devolverá el dinero a todas las personas contribuyentes. Lo que caracteriza a este sistema es el fuerte componente social, de cooperación colectiva [24].

Entre otros modelos de financiamiento se encuentran: aporte no reembolsable, subsidios y créditos, e incubadoras y aceleradoras [24].

Cuando se presenta la idea del videojuego a un inversor, por lo general no habrá mucho tiempo para convencerlo. Se deberá ser concreto y el discurso deberá ser corto y atractivo, que cautive rápidamente al interlocutor.

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Financiamiento*.

TABLA 4.22 - Tareas principales para la actividad Financiamiento

TAREA	DESCRIPCIÓN
1. DETERMINAR EL MODELO DE FINANCIAMIENTO	Determinar la estrategia para conseguir el financiamiento para desarrollar el videojuego. Entre los principales modelos de financiamiento se encuentran: <i>crowdfunding</i> , aporte no reembolsable, subsidios y créditos, e incubadoras y aceleradoras.
2. CREAR DOCUMENTO DE VENTA (PITCH)	Crear el documento de venta (pitch) considerando los elementos descritos en el documento de concepto de juego.
3. PRESENTAR LA IDEA DEL VIDEOJUEGO A UNA FUENTE DE FINANCIAMIENTO	Presentar concepto de videojuego a cliente, publisher o alguna otra fuente de financiamiento. La presentación incluye el documento de venta (pitch) y el prototipo del videojuego.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *F1.4*, *D3* y *C4*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.23 - Productos de trabajo de la actividad Financiamiento

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Investigación Preliminar	Análisis de Mercado
Análisis Competitivo	Análisis de Mercado
Prototipo del Videojuego	Concepto del Videojuego
Documento de Concepto de Juego	Concepto del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Documento de Venta (Pitch)	Fuente de Financiamiento
Prototipo del Videojuego	Fuente de Financiamiento
Presentación	Fuente de Financiamiento
PRODUCTOS INTERNOS	
NOMBRE	
Modelo de Financiamiento	

4.6.4. Fase de Preproducción

En la fase de *Preproducción* se analizarán los diferentes tipos de requerimientos que tendrá el videojuego para cumplir con los objetivos y expectativas, y se definirá la línea de producción de los diferentes *assets* que requerirá el videojuego. También se realizará el diseño detallado del videojuego con base en la especificación de requerimientos y finalizará con el diseño detallado de software.

4.6.5. Actividad Requerimientos y Diseño del Videojuego

Requerimientos. En la actividad *Requerimientos* se analizan los diferentes tipos de requerimientos (tecnología, funcionales, interfaz de usuario, fiabilidad, desempeño, etc.) que el videojuego tendrá que cubrir para cumplir con los objetivos y expectativas del juego deseados. También se define la línea de producción de los diferentes *assets* que requerirá el videojuego.

El desarrollo de un videojuego no es el resultado directo de las necesidades del usuario o un negocio. Los videojuegos no se enfocan en resolver problemas de usuario, sino en entretenerlos. Por otra parte, los requerimientos como la inmersión, sorpresa, placer y nostalgia están presentes en los videojuegos. De esta forma la ingeniería de requerimientos tradicional no puede ser aplicada a los videojuegos tal como es [26].

En [23] se estudió la ingeniería de requerimientos en la industria de los videojuegos y se concluye que el éxito de los videojuegos depende de la resolución de los problemas de comunicación en un equipo multidisciplinario (con formación técnica y artística), el impacto de los videojuegos desarrollados anteriormente, integración entre arte y tecnología, y el impacto de los requerimientos no funcionales. Además, el rol de los requerimientos no funcionales como diversión, estética, apariencia (*look and feel*) es muy importante en los videojuegos, pero estos requerimientos son difíciles de administrar y trazar.

La diversión es el principal objetivo de los videojuegos. Esto se relaciona con la intrínseca motivación para jugar juegos, sin embargo, pueden existir otros motivadores como el aprendizaje [27]. Con el fin de ser atractivo y jugado repetidamente, los videojuegos deben tratar con las emociones del jugador, como consecuencia, los videojuegos incluyen requerimientos emocionales [28]. En [29], se propone que los requerimientos emocionales pueden ser administrados utilizando *mapas de terreno (terrain maps)*, *mapas de intensidad emocional (emotional intensity maps)* y *línea del tiempo emocional (emotion timeline)*.

Otro intento para aplicar ingeniería de requerimientos en el diseño de la experiencia de usuario, llamada requerimientos de experiencia, se realizó con el fin de proporcionar un mecanismo para los desarrolladores de videojuegos para predecir las impresiones y experiencia de un jugador. La idea era permitir que las compañías aplicaran técnicas de ingeniería de requerimientos al principio del desarrollo del videojuego, pero la complejidad de utilizar esta técnica fue mayor de lo previsto [30].

La creación de videojuegos también está estrechamente ligada a iteraciones y creación de prototipos. Es usual crear muchos prototipos con el objetivo de cumplir con requerimientos como

la *diversión* [30]. Los prototipos se realizan en la fase de *Preproducción* con el propósito de ayudar a los diseñadores a encontrar el tipo de videojuego que les gustaría crear.

El videojuego puede cambiar significativamente con base en la retroalimentación del área de mercadotecnia. En muchos casos la actividad de *Pruebas de Software* afecta el diseño del producto y cambia algunas de sus características.

En [28] se concluye que los estudios de desarrollo de videojuegos no difieren significativamente de otras organizaciones que desarrollan productos de software. Sin embargo, el énfasis en los requerimientos no funcionales y el aseguramiento de la calidad con pruebas de usuario tiene un impacto en cómo operan estos estudios. Principalmente se utilizan procesos informales con enfoques tradicionales como el modelo en cascada y desarrollo incremental guiado por prototipos. Además, los estudios necesitan administración de requerimientos debido a que los productos podrían tener diferencias significativas entre los primeros prototipos y el producto final, basado en la retroalimentación de los jugadores y el área de mercadotecnia.

Aunque las prácticas de ingeniería de requerimientos no son ampliamente adaptadas por los estudios de desarrollo de videojuegos, si se aplican algunos conceptos como identificación y administración de requerimientos, y gestión de riesgos provocada por los requerimientos no funcionales (generados por constantes prototipos y pruebas de usuarios).

En [22] se recomienda que el administrador del proyecto aplique ingeniería de requerimientos para identificar todos los requerimientos de diseño y determinar un alcance factible del proyecto considerando a los equipos involucrados. Además, debe utilizar administración de riesgos para manejar el constante cambio en los requerimientos y en el diseño del videojuego.

En esta actividad también se deberá definir la línea de producción de los diferentes *assets* del videojuego y el software (*game builds*). Durante el desarrollo de videojuegos se generan una gran cantidad de *assets*, por lo que es fundamental que los desarrolladores optimicen sus herramientas y la línea de producción para administrar los *assets* [22]. En la FIGURA 4.7 se muestra un ejemplo de la línea de producción de arte 3D.

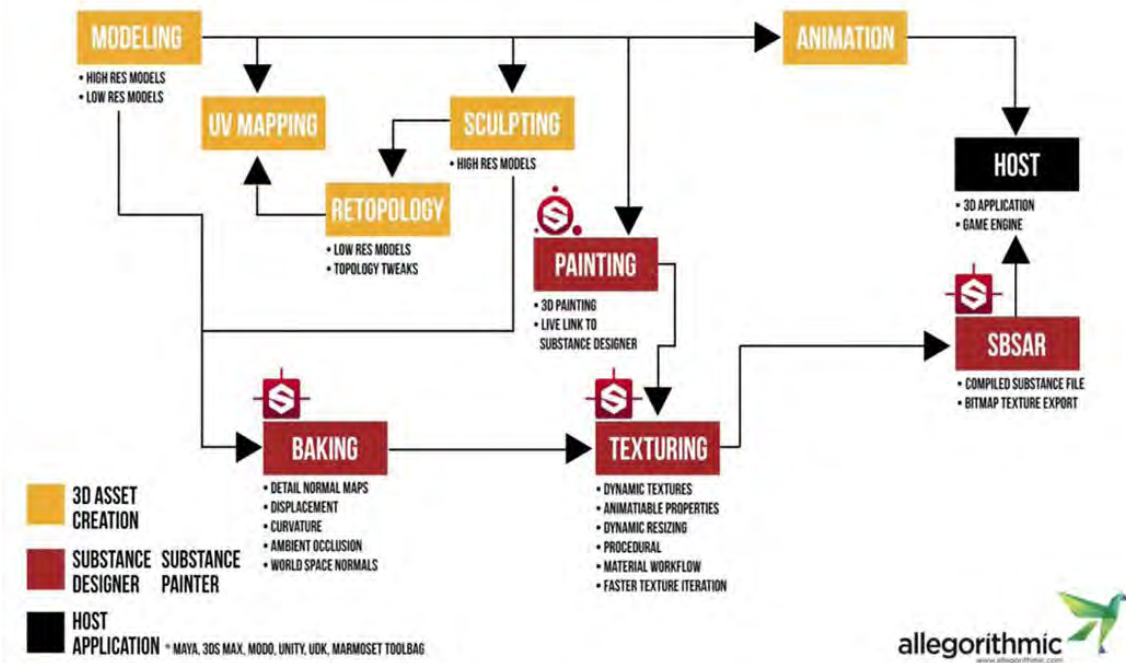


FIGURA 4.7 - Línea de producción de arte 3D (Allegorithmic) [41]

Para definir las tareas que se realizarán en esta actividad, se consideraron las tareas definidas para la actividad *Análisis de Requerimientos* del estándar *ISO/IEC 29110 Perfil Básico*, se complementan con las tareas definidas en la sección *Game requirements* de [17] y se incluyeron los requerimientos emocionales y de jugabilidad.

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Requerimientos*.

TABLA 4.24 - Tareas principales para la actividad Requerimientos

TAREA	DESCRIPCIÓN
1. DEFINIR CARACTERÍSTICAS DEL VIDEOJUEGO (GAME FEATURES)	Identificar y priorizar las características principales que tendrá el videojuego (multijugador, mecánicas de juego, jugabilidad (gameplay), entre otros).
2. EVALUAR TECNOLOGÍA NECESARIA PARA EL PROYECTO	Evaluar la tecnología necesaria en el proyecto. Se deberá evaluar el motor de juego (game engine), motor de IA, motor de física, motor de rendering, herramientas (arte, scripting) y otros elementos técnicos necesarios para proporcionar la funcionalidad deseada del videojuego. Además, se deberá decidir cómo se va a obtener la tecnología necesaria.
3. DEFINIR TECNOLOGÍA	Definir tecnología necesaria para desarrollar los elementos técnicos necesarios para proporcionar la funcionalidad deseada del videojuego.
4. IDENTIFICAR REQUERIMIENTOS DEL VIDEOJUEGO	Identificar los diferentes tipos de requerimientos (funcionales, interfaz de usuario, interfaces externas, fiabilidad, desempeño, legales y regulativos, emocionales, experiencia, jugabilidad) que el videojuego tendrá que cubrir para cumplir con los objetivos y expectativas deseados.

5. IDENTIFICAR LIMITACIONES Y RESTRICCIONES DE DISEÑO	Identificar limitaciones o restricciones del diseño del videojuego y construcción, necesidades impuestas por el <i>cliente</i> .
6. IDENTIFICAR LIMITACIONES DEL SISTEMA	Identificar las limitaciones del sistema. Se deberán aclarar ciertos límites que no deberían ser pasados (número de polígonos, número de mensajes enviados por segundo, número simultaneo de explosiones en pantalla, etc.).
7. DEFINIR LÍNEA DE PRODUCCIÓN DE ASSETS Y SOFTWARE (GAME BUILDS)	Definir la línea de producción de los diferentes <i>assets</i> del videojuego y el software (game builds). Se deberán definir los pasos necesarios para que los componentes de software y los <i>assets</i> del juego funcionen correctamente.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *D5*, *C7* y *F2.2*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.25 - Productos de trabajo de la actividad Requerimientos

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [revisado]	Planeación del Proyecto
Documento de Concepto de Juego [validado]	Concepto del Videojuego
Estándares de Plataforma	Análisis de Mercado y Financiamiento
Políticas de Publicación en Mercados	Análisis de Mercado
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Especificación de Requerimientos [validado]	Diseño del Videojuego
PRODUCTOS INTERNOS	
NOMBRE	
Características del Videojuego (Game Features)	
Evaluación de Tecnología	
Tecnología	
Requerimientos Funcionales	
Requerimientos de Interfaz de Usuario	
Requerimientos de Interfaces Externas	
Requerimientos de Fiabilidad	
Requerimientos de Desempeño	
Limitaciones y Restricciones	
Requerimientos Legales y Regulatorios	
Requerimientos Emocionales	
Requerimientos de Experiencia	
Requerimientos de Jugabilidad	
Limitación del Sistema	
Línea de Producción	
Listas de Verificación	
Configuración del Software	
Solicitud de Cambio [propuesta]	

Diseño del Videojuego. En la actividad *Diseño del Videojuego* se proponen tareas para diseñar el videojuego de forma detallada con base en la especificación de requerimientos considerando el diseño conceptual inicial y se crea el documento de diseño de juego (*Game Design Document*).

El *Diseño del Videojuego* es el proceso a través del cual un diseñador crea un contexto, que luego experimentará el jugador, y del cual emerge algún significado [25]. El contexto en el juego tendría que ver con los espacios, los objetos, la historia, las narrativas y las conductas; mientras que el significado es el resultado de las acciones de los jugadores durante el juego.

El diseño se trata de entender las limitaciones como el dinero, tiempo, tecnología y experiencia (habilidades del equipo), y convertirlas en ventajas a través del conocimiento y creatividad.

Hay varias formas de clasificar a los diferentes elementos que forman un juego. En [31] se definen cuatro categorías elementales:

1. **Mecánicas.** Son los procedimientos y reglas del juego. Las mecánicas describen las metas del juego, cómo los jugadores pueden y no pueden intentar lograrlas, y lo que pasa cuando lo intentan. Las mecánicas son cruciales para la jugabilidad (gameplay).
2. **Historia.** Esta es la secuencia de eventos que ocurren en el videojuego. Podría ser lineal, pre-guiada, o podría ser derivada y emergente. Cuando se tiene una historia que contar a través del juego, se deben elegir las mecánicas que refuercen esa historia y dejen que emerja.
3. **Estética (Aesthetics).** Esto es cómo se ve el juego, suena y se siente. La estética del juego es muy importante ya que tiene la relación más directa con la experiencia del jugador. Se deberán elegir las mecánicas que hagan sentir al jugador inmerso en el mundo que la estética ha definido. Así como la tecnología que pueda soportarla.
4. **Tecnología.** La tecnología que se elija para el juego permite que se hagan ciertas cosas y prohíbe hacer otras. La tecnología es esencialmente el medio en el cual la estética tiene lugar, en la cual las mecánicas ocurrirán, y a través de la cual será contada la historia.

Ninguno de los cuatro elementos es más importante que el otro. Todos son esenciales, no importa el juego que se esté diseñando. Cada elemento se considera de forma separada, y después todos juntos como un todo [31].

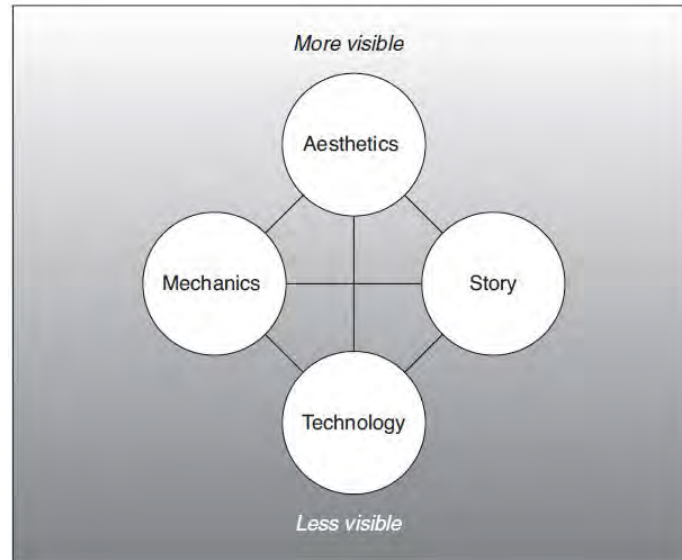


FIGURA 4.8 - Los cuatro elementos básicos de un videojuego [31]

En [31] se definen elementos importantes que se deberán considerar cuando se diseña un videojuego:

- **Experiencia.** El diseño del videojuego deberá considerar la experiencia que tendrá el jugador, lo esencial de esa experiencia y cómo se podrá capturar en el juego. Si hay una gran diferencia entre la experiencia que se quiere crear y la que se está creando, entonces el juego necesitará cambios. Se deberá establecer claramente la experiencia deseada, y encontrar tantas formas como sea posible para crear esa experiencia en el juego.
- **Sorpresa.** La sorpresa es una parte crucial del entretenimiento. Se deberán considerar los elementos del juego que sorprenderán al jugador.
- **Diversión.** Para maximizar la diversión del videojuego se deberán analizar las partes del videojuego que son divertidas y las que deben serlo más.
- **Curiosidad.** Es un elemento que sirve como motivación para el jugador, no solo se deben definir las metas u objetivos del videojuego, sino la razón por cual el jugador quiere alcanzar esas metas u objetivos.
- **Resolución de Problemas.** Se deberán considerar los problemas que los jugadores deberán resolver para tener éxito en el juego.
- **Diseño holográfico.** Se refiere a la forma en la que se relacionan los cuatro elementos fundamentales de un videojuego (mecánicas, historia, estética y tecnología) y la experiencia del jugador. Se deberán analizar los elementos del juego que hacen la experiencia disfrutable y los que no, así como los cambios necesarios para mejorar la experiencia.

- **Temas.** El tema es sobre lo que trata el videojuego. Es la idea que todos los elementos deben soportar.
- **Resonancia.** Se deberán identificar los elementos que hacen al videojuego especial, las mejores características del juego que lo vuelven atractivo para los jugadores.
- **Planteamiento del Problema.** Definir las restricciones y metas del videojuego como un planteamiento del problema para ayudar a aclarar el diseño del videojuego de forma más rápida.
- **Mitigación de Riesgos.** Se deberán considerar las cosas que podrían salir mal en el juego. Enfocarse en las partes del juego que están en peligro y no solo en las que funcionan bien.
- **Prototipos.** Los prototipos permiten validar o probar jugabilidad y mecánicas de juego o cualquier cosa que no esté bien definida (línea de producción). Se deberá tener claro el objetivo de construir un prototipo, de lo contrario será una pérdida de tiempo.
- **El Jugador.** Siempre se deberá diseñar el videojuego pensando en el jugador, la experiencia de juego y las mecánicas, todo al mismo tiempo. También será muy importante observar a los jugadores cuando prueben el juego. Mientras más se les observe jugando, será más sencillo predecir lo que van a disfrutar del juego.
- **Flujo (Flow).** Se deberá analizar si el juego tiene metas claras y si existen partes dentro del juego que distraen al jugador al grado de hacerlo olvidar los objetivos. Además, se deberá analizar si el juego proporciona un flujo constante de retos no tan fáciles y no tan difíciles, considerando el hecho de que las habilidades del jugador podrían mejorar gradualmente.
- **Juicio.** Se refiere al juicio que el videojuego hará sobre los jugadores. Se deberá identificar lo que juzgará el videojuego de los jugadores, cómo lo comunicará y si los jugadores sentirán que es justo y si les importará tanto como para mejorar.
- **Mecánicas (espacio funcional).** Se refiere al espacio donde se llevará a cabo el juego. Se deberá definir si el espacio será continuo o delimitado, el número de dimensiones que tendrá, los sub-espacios y cómo estarán conectados.
- **Mecánicas (objetos, atributos y estados).** Se deberá analizar los objetos en el espacio del juego (personajes, accesorios, marcadores, cualquier cosa que pueda ser vista o manipulada en el juego), sus atributos, los posibles estados de cada atributo y lo que hace cambiar estos estados. Además, se deberá analizar qué estados serán conocidos solamente por el juego y qué estados serán conocidos por el jugador. Decidir los diferentes atributos, sus estados y quien los conoce, es el núcleo de las mecánicas del juego.
- **Mecánicas (acciones).** Las acciones son los "verbos" de las mecánicas. Se deberá considerar cuántas acciones podrán realizar los jugadores, sobre qué objetos actuarán, de cuántas formas los jugadores podrán alcanzar sus metas y cuántos elementos controlarán.

- **Mecánicas (reglas).** Las reglas son las mecánicas más importantes. Definen el espacio, los objetos, las acciones, las consecuencias de las acciones, las restricciones sobre las acciones y las metas. Se deberán definir las reglas fundacionales del juego, los diferentes modos de juego y asegurarse de que las reglas cubran cada circunstancia y sean claras.
- **Mecánicas (habilidad).** Se refiere a las habilidades que requerirá el juego de los jugadores (físicas, mentales, sociales). Se deberán definir las habilidades que requerirán los jugadores, las habilidades dominantes y las habilidades que crean las experiencias deseadas. Además, se deberá analizar si los jugadores podrían mejorar las habilidades requeridas con práctica y el nivel de demanda.
- **Mecánicas (oportunidad).** Se refiere a las interacciones entre las otras mecánicas (espacio, objetos, acciones, reglas y habilidad). La oportunidad es una parte esencial de un juego divertido porque significa incertidumbre e incertidumbre significa sorpresa. Se deberán analizar que partes del juego serán aleatorias y si estas darán a los jugadores sensaciones positivas (emoción y reto) o negativas (desesperanza y falta de control). Además, se deberá controlar la relación entre oportunidad y habilidad.
- **Balance de Mecánicas.** Se refiere a ajustar los elementos del juego hasta que entreguen la experiencia deseada. Se deberán considerar elementos como *justicia, reto, opciones significativas, habilidad vs oportunidad, competencia, cooperación, tiempo, recompensa, castigo, simplicidad/complejidad*.
- **Interfaz.** El objetivo de la interfaz es que los jugadores se sientan en control de su experiencia. Se deberá cuidar el diseño de la interfaz del juego, ya que un control significativo es esencial para lograr la experiencia de juego deseada.
- **Interfaz física.** La interfaz física son los dispositivos de entrada (mouse, teclado, joystick, control, etc.) que permiten que el jugador interactúe de alguna forma con el juego. Se deberá asegurarse de que la interfaz física este bien ajustada al juego que se quiere construir.
- **Interfaz virtual.** Para diseñar la interfaz virtual se deberá considerar la información necesaria que el jugador debe recibir y cómo puede ser presentada de tal forma que no interfiera con la interacción del jugador dentro del juego.
- **Curva de interés.** Las curvas de interés pueden ser una herramienta útil cuando se crea una experiencia de entretenimiento. Al graficar el nivel de interés esperado sobre el curso de una experiencia, los puntos conflictivos se vuelven claros y pueden ser corregidos. Se deberá observar a los jugadores teniendo la experiencia de juego y comparar el nivel de interés observado con el nivel esperado.
- **Historia.** Se deberá describir lo que pasa en el juego y a quien le pasa, la acción y las motivaciones de los personajes. Además, se deberán analizar las diferentes opciones que tendrá el jugador para conseguir los objetivos y las diferentes historias que puedan generar. Las buenas historias tienen buenas curvas de interés.

- **Mundo.** El mundo del juego es una cosa que existe aparte, el videojuego es una entrada a un lugar que existe en la imaginación de los jugadores. Se deberán establecer las reglas de consistencia para los escenarios del juego, así como la arquitectura y organización del espacio.
- **Diseño de niveles.** Se deberán diseñar los niveles que tendrá el videojuego. Se deberá planear lo que el jugador hará en cada nivel y como progresará a través del juego, el diseño de niveles contemplará: mecánicas relativas al escenario, mecánicas de los enemigos, objetivos, resolución de acertijos, estrategias, eventos, jefes finales, puntos de guardado, recolección de premios, rutas a través de un escenario, arte conceptual.
- **Personajes.** Se deberán diseñar los personajes que aparecen en el juego, así como el lugar y el momento de aparición, sus habilidades y equipo, y todos los riesgos asociados con la creación de cada personaje. Además, se deberán definir los roles de cada personaje.
- **Multijugador.** Se deberá decidir cuidadosamente si el juego tendrá la opción de multijugador, ya que los juegos con esta opción son mucho más complicados técnicamente y sus mecánicas son más difíciles de balancear.
- **Musicalización.** La música puede canalizar el diseño del videojuego, apoyar las distintas situaciones de los videojuegos, relajar o tensar al jugador en distintos momentos del juego, comunicar eventos, ayudar a narrar la historia y facilitar la creación de un cierto estado emocional en el jugador. Se deberá elegir la música y los efectos de sonido apropiados de acuerdo al diseño del videojuego.
- **Modelo de negocios.** Al momento de elegir un sistema de monetización (anuncios *in-game*, *ad-servers*, economía virtual, etc.) se deberá tomar en cuenta el alcance, plataforma, así como los objetivos del proyecto.

Las tareas de la actividad *Diseño del Videojuego* se obtienen como resultado de la comparación y homologación de las tareas descritas en [31], [32] y [17] para diseñar un videojuego. En la siguiente TABLA se muestran las tareas propuestas a realizar en esta actividad.

TABLA 4.26 - Tareas principales para la actividad Diseño del Videojuego

TAREA	DESCRIPCIÓN
1. CREAR HISTORIA DEL VIDEOJUEGO	Crear la historia del juego y el resumen narrativo. Se deberá describir lo que pasa en el juego y a quien le pasa, la acción y las motivaciones de los personajes.
2. CREAR LISTA DE EVENTOS CLAVE EN LA HISTORIA	Crear la lista de eventos clave en la historia. Estos son los puntos clave en la historia que indican la necesidad de incluir algún elemento de jugabilidad, diálogos, cinemáticas o introducen un nuevo nivel en el juego.
3. CREAR GUIÓN LITERARIO	Crear el guión literario del juego (si es que lo requiere). Crear los diálogos para todas las partes del juego, incluyendo cinemáticas y jugabilidad.

4. CREAR BIBLIA DEL MUNDO	Crear la biblia del mundo del videojuego. En la biblia del mundo se establecen las reglas de consistencia para los escenarios del juego. Estas reglas ayudan a los artistas, animadores, diseñadores de niveles y programadores a entender cómo crear <i>assets</i> del juego y el tipo de física permitida en el juego.
5. CREAR LISTA DE PERSONAJES	Crear la lista de personajes que aparecen en el juego. Esta lista incluye dónde y cuándo aparece cada personaje en el juego, qué habilidades y equipo tiene, y todos los riesgos asociados con la creación del personaje.
6. DISEÑAR JUGABILIDAD (GAMEPLAY)	Diseñar los elementos de jugabilidad (gameplay). Entre los elementos de jugabilidad, se encuentran: retos para el jugador, recompensas del jugador, curva de aprendizaje, esquema del control, acciones del jugador, elementos del multijugador.
7. DISEÑAR MECÁNICAS DEL VIDEOJUEGO	Diseñar las mecánicas de juego. También se deberán balancear o equilibrar las mecánicas que rompan la experiencia de juego deseada (objetos o elementos, comportamientos, reglas).
8. DISEÑAR CONTROL DEL JUEGO	Diseñar la configuración del control de acuerdo a la plataforma de juego, diseñar los comandos de las mecánicas que se ejecuten mediante el control.
9. DISEÑAR NIVELES Y PLANOS DE NIVELES	Diseñar los niveles y sus mapas o planos que tendrá el videojuego. Se deberá planear lo que el jugador hará en cada nivel y como progresará a través del juego, el diseño de niveles contemplará: mecánicas relativas al escenario, mecánicas de los enemigos, objetivos, resolución de acertijos, estrategias y dinámicas, eventos, jefes finales, puntos de guardado, recolección de premios, rutas a través de un escenario, arte conceptual.
10. DEFINIR BIBLIA DE ARTE	Definir la biblia de arte del juego. Se deberán definir las guías para ayudar a mantener la apariencia del videojuego consistente, el estilo de personajes, objetos y ambientes.
11. CREAR ARTE CONCEPTUAL	Crear el arte conceptual requerido por el videojuego (objetos, personajes y escenarios donde se llevará a cabo la acción del juego).
12. DISEÑAR INTERFAZ DE USUARIO	Diseñar la Interfaz de usuario del juego. Se deberá definir la forma en la que el jugador interactúa con el videojuego y con qué mecanismos contará para ello.
13. CREAR DIAGRAMA DE FLUJO DE INTERFAZ DE USUARIO	Crear el diagrama de flujo de interfaz de usuario. Este diagrama se centra en los menús vistos en pantalla por el jugador durante el juego. También describe que pantallas se necesitan y en qué orden deben aparecer.
14. CREAR DIAGRAMA DE FLUJO DEL JUEGO	Crear el diagrama de flujo del juego. Se deberá establecer la progresión a través del juego desde el punto de vista de la interacción del jugador con el juego, los diferentes niveles que encontrará y en el orden en el que aparecerán, los objetivos del jugador en cada nivel, poderes y habilidades del jugador en cada escenario, narrativas clave y cualquier cinemática que sea necesaria para explicar la historia.
15. DISEÑAR INTELIGENCIA ARTIFICIAL DEL VIDEOJUEGO	Diseñar la inteligencia artificial (IA) del videojuego incluye métodos para producir la ilusión de inteligencia en la conducta de personajes no jugables.
16. DEFINIR LOS LOGROS (ACHIEVEMENTS)	Identificar los logros (achievements) que tendrá el videojuego. Se deberá definir el número de logros que tendrá el videojuego, en qué consiste cada uno de ellos y cómo se obtendrán.

17. CONSTRUIR PROTOTIPOS	Construir prototipos que permitan validar o probar jugabilidad y mecánicas de juego o cualquier cosa que no esté bien definida (línea de producción).
18. CREAR ELEMENTOS DE AUDIO	Crear los elementos de audio que requerirá el videojuego (narración, efectos de sonido, música, voces, diálogos, etc.) de acuerdo a la ambientación, historia, jugabilidad.

Es importante mencionar que no todas las tareas propuestas se podrán realizar en todos los desarrollos. La realización y la complejidad de estas tareas está en función del tipo de videojuego a desarrollar. Por ejemplo, un videojuego como *Tetris* (FIGURA 4.9) no requirió de la creación de la historia, personajes, guión literario o diálogos. En este tipo de videojuegos (*puzzles*) las mecánicas y la jugabilidad son lo más importante.



FIGURA 4.9 - Tetris Ultimate (2014) [42]

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *D6*, *C7*, *C8* y *F2.3*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.27 - Productos de trabajo de la actividad Diseño del Videojuego

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [revisado]	Planeación del Proyecto
Especificación de Requerimientos [validado]	Requerimientos
Documento de Concepto de Juego [validado]	Concepto del Videojuego
Archivos de Referencia	Concepto del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Documento de Diseño de Juego [validado]	Requerimientos Diseño Técnico del Videojuego Construcción de Software
Prototipos	Requerimientos Diseño Técnico del Videojuego Construcción de Software
PRODUCTOS INTERNOS	
NOMBRE	
Información General del Juego	
Historia del Juego	
Resumen Narrativo	
Lista de Eventos Clave en la Historia	
Guión Literario	
Biblia del Mundo	
Lista de Personajes	
Jugabilidad (Gameplay)	
Mecánicas de Juego	
Control	
Diseño de Niveles	
Planos de Niveles	
Biblia de Arte (Guía de Estilo de Arte.)	
Arte Conceptual	
Interfaz de Usuario	
Diagrama de Flujo de Interfaz de Usuario	
Diagrama de Flujo del Juego	
Inteligencia Artificial	
Logros (Achievements)	
Elementos de Audio	
Configuración del Software	

4.6.6. Actividad Diseño Técnico del Videojuego

El *Diseño Técnico del Videojuego*, comprende principalmente dos actividades: *Arquitectura de Software* y *Diseño Detallado del Software*.

Arquitectura de Software. En la actividad *Arquitectura de Software* se diseña la arquitectura a un alto nivel de abstracción para describir la estructura global del software. El diseño arquitectónico se interesa por entender cómo debe organizarse un sistema y cómo tiene que diseñarse la estructura global de ese sistema [11].

La *Arquitectura de Software* de un programa o sistema computacional es la estructura o estructuras del sistema, las cuales constan de elementos de software, las propiedades visibles externamente de esos elementos, y la relación entre ellos [33].

Los videojuegos comenzaron como piezas bastante pequeñas y limitadas de software. Sin embargo, como se observa en la FIGURA 4.10 se tuvieron que tomar decisiones arquitectónicas para identificar y desarrollar sub-módulos, los cuales tenían su propia responsabilidad.

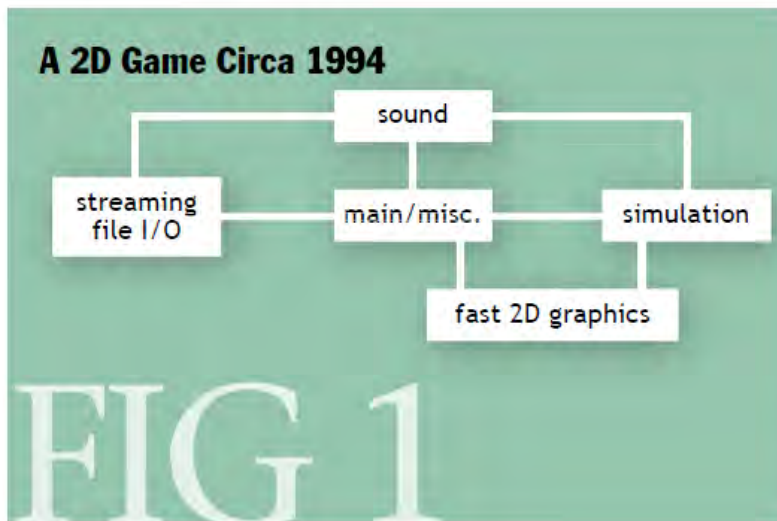


FIGURA 4.10 - Estructura típica de un juego 2D por el año 1994 [34]

Un par de años después, los gráficos en 3D se convirtieron en una parte muy importante de los videojuegos y aumentó la complejidad del código fuente.

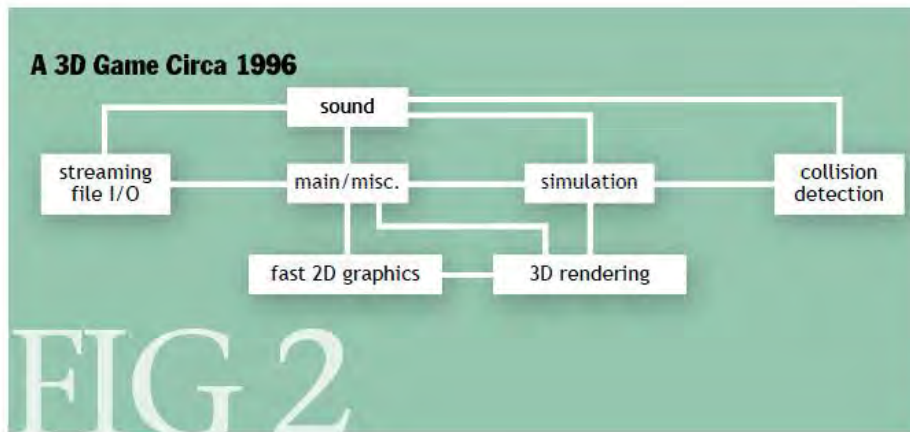


FIGURA 4.11 - Estructura típica de un juego 3D por el año 1996 [34]

Cuando se observa la estructura típica de un juego 3D del año 2004 (FIGURA 4.12), se concluye que no hay forma de que una compañía pueda producir tal pieza de software sin considerar la

arquitectura de software. Por lo tanto, una arquitectura de software adecuada y bien planeada es parte integral de un juego exitoso [33].

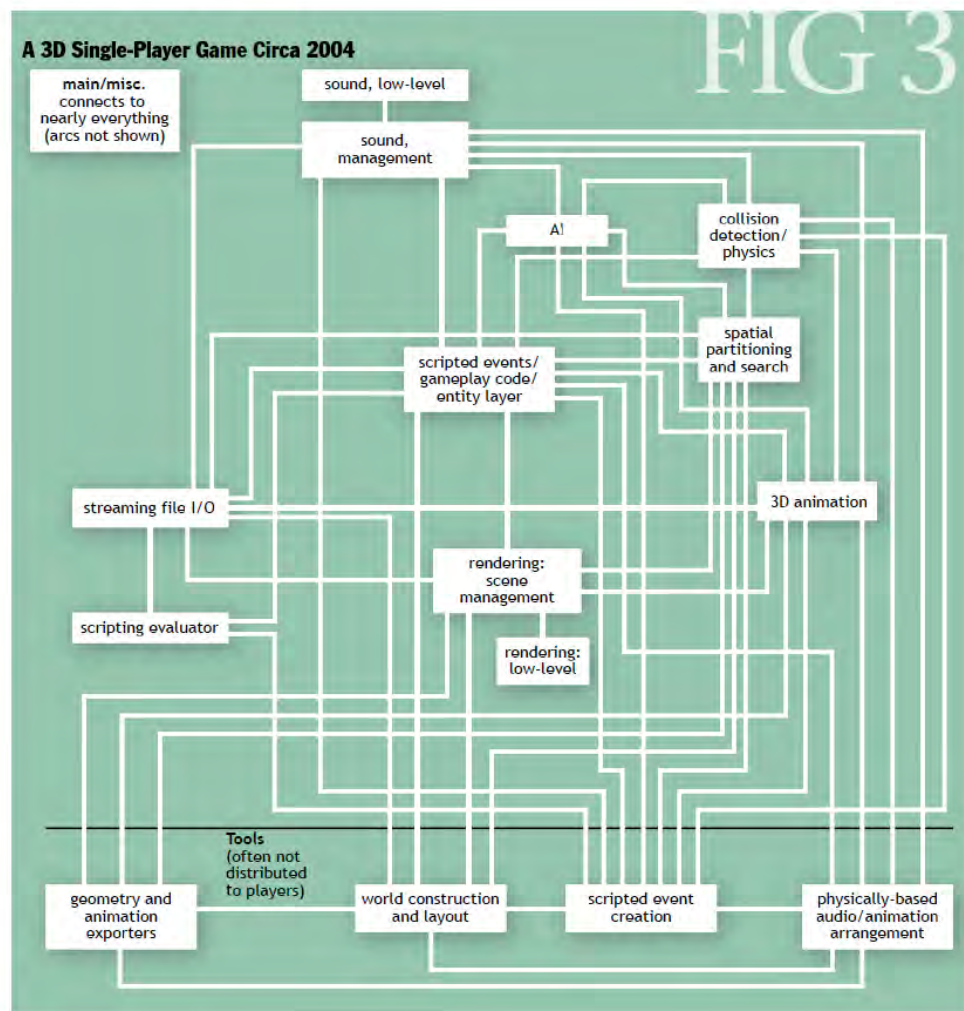


FIGURA 4.12 - Estructura típica de un juego 3D por el año 2004 [34]

Una posible forma de diferenciar un *Motor de Juego (Game Engine)* y el software que representa a un juego está asociada al concepto de *Arquitectura Dirigida por Datos (Data-Driven Architecture)*. Básicamente, cuando un juego contiene parte de su lógica o funcionamiento en el propio código (*hard-coded logic*), entonces no resulta práctico reutilizarla para otro juego, ya que implicaría modificar el código fuente sustancialmente. Sin embargo, si dicha lógica o comportamiento no está definido a nivel de código, sino por ejemplo mediante una serie de reglas definidas a través de un lenguaje de script, entonces la reutilización sí es posible y beneficiosa, ya que optimiza el tiempo de desarrollo. Este planteamiento facilita la reutilización de software [16].

El concepto de *Motor de Juego* se hizo más popular a medida que otros desarrolladores comenzaron a utilizar diversos módulos o juegos previamente licenciados para generar sus propios juegos. En otras palabras, era posible diseñar un juego del mismo tipo sin apenas modificar el núcleo o *Motor de Juego*, sino que el esfuerzo se podía dirigir directamente a la parte artística y a las mecánicas del mismo [16].

En [35], se identificaron dos clases de módulos que están presentes en prácticamente todos los motores de juego: *módulos principales (core modules)* y *módulos de jugabilidad (gameplay modules)*. Los *módulos principales* se caracterizan por realizar tareas de propósito general como *rendering*, física, detección de colisiones, sistema de partículas, etc., y son módulos que realizan tareas complejas y deben estar optimizados tanto como sea posible [33]. Los *módulos de jugabilidad* se centran en las partes que los desarrolladores necesitan modificar para desarrollar un juego específico. Esto consiste en el manejo de entidades, *scripting*, paso de mensajes, etc., y generalmente no son los módulos más intensos en cuanto a desempeño se refiere.

En la FIGURA 4.13 se muestra la visión conceptual de la arquitectura general de un *Motor de Juego* [16].

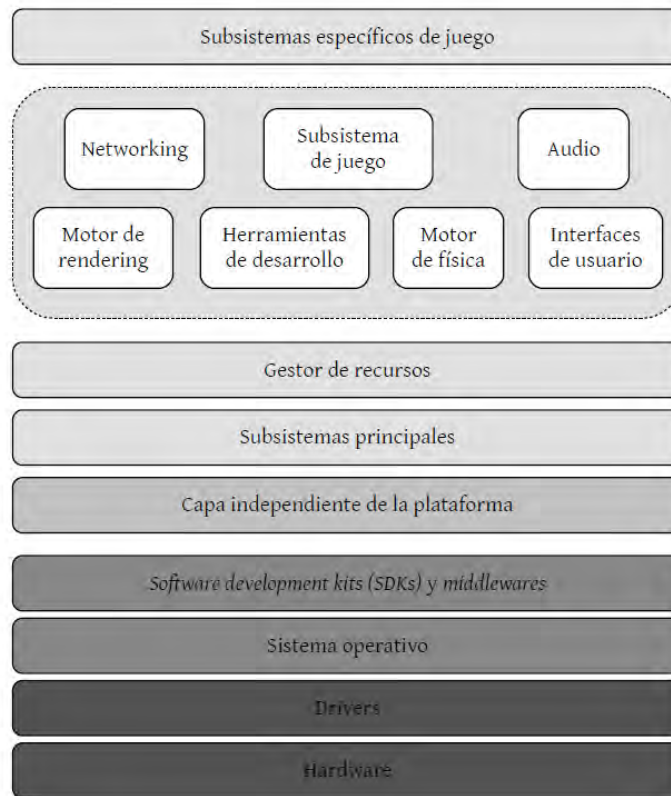


FIGURA 4.13 - Visión conceptual de la arquitectura general de un Motor de Juego [16].

Actualmente existen una gran variedad de motores, los cuales se pueden adquirir de forma gratuita o pagando algún tipo de licencia (*Unreal Engine, Unity, CryENGINE*). La ventaja de utilizar un *Motor de Juego* es que el estudio se puede enfocar solamente en el contenido del juego en lugar de desarrollar el motor desde cero. Una desventaja es que el estudio está limitado a lo que el motor es capaz de hacer. Por supuesto, el estudio de desarrollo puede modificar, adaptar y extender el motor de juego. En cada desarrollo el estudio deberá evaluar si se va a construir un motor de juego desde cero, si se adaptará alguno existente o se utilizará alguno que este en el mercado.

En la siguiente TABLA se muestran las tareas propuestas a realizar en la actividad *Arquitectura de Software*.

TABLA 4.28 - Tareas principales para la actividad Arquitectura de Software

TAREA	DESCRIPCIÓN
1. CREAR ESPECIFICACIÓN TÉCNICA DE DESARROLLO	Crear la especificación técnica de desarrollo (herramientas, lenguajes de programación, estándares de codificación).
2. CREAR DESCRIPCIÓN DE ARQUITECTURA	Crear la descripción de arquitectura de software con elementos identificados en la especificación de requerimientos. Se deberán identificar elementos que influyen en el desarrollo del videojuego y conforman su estructura como: personas, software, hardware, otros sistemas, conectividad, equipo de trabajo, medios de distribución.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de D7, y F2.4 Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.29 - Productos de trabajo de la actividad Arquitectura de Software

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [aceptado]	Planeación del Proyecto
Especificación de Requerimientos [validada]	Requerimientos
Documento de Diseño de Juego [validado]	Diseño del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Especificación Técnica de Desarrollo	Diseño Detallado de Software
Descripción de Arquitectura de Software [validada]	Diseño Detallado de Software
Registro de Trazabilidad	Diseño Detallado de Software
PRODUCTOS INTERNOS	
NOMBRE	
Listas de Verificación	
Listas de Validación	

Diseño Detallado del Software. En la actividad *Diseño Detallado de Software*, las tareas a realizar en esta actividad se enfocan en los siguientes elementos:

- **Diseño Detallado del Software.** A un bajo nivel de abstracción incluye detalle de los *Componentes de Software* para facilitar su construcción y prueba dentro del ambiente de desarrollo (*Motor de Juego*).
- **Escenarios y Casos de Prueba.** Establecen los elementos necesarios para probar el código, integrar los *Componentes de Software* y probar el videojuego.

En la siguiente TABLA se muestran las tareas propuestas a realizar en la actividad *Diseño Detallado de Software*.

TABLA 4.30 - Tareas principales para la actividad Diseño Detallado de Software

TAREA	DESCRIPCIÓN
1. CREAR DISEÑO DETALLADO DE SOFTWARE	Crear la especificación técnica de desarrollo (herramientas, lenguajes de programación, estándares de codificación).
2. CREAR CASOS DE PRUEBAS DE INTEGRACIÓN	Crear los casos de pruebas de integración con el objetivo de verificar el correcto funcionamiento de los componentes de software al ser integrados.
3. CREAR ESCENARIOS Y CASOS DE PRUEBAS DE USABILIDAD	Crear grupos de escenarios y casos de pruebas de usabilidad para la plataforma en la que se desarrollará el videojuego. Se deberá definir la lista de tareas a realizar y los cuestionarios que se aplicarán a los usuarios participantes en las pruebas de usabilidad. También se deberá determinar el periodo en el que se realizarán estas pruebas.
4. CREAR ESCENARIOS Y CASOS DE PRUEBAS DE JUGABILIDAD	Crear grupos de escenarios y casos de pruebas de jugabilidad para la plataforma en la que se desarrollará el videojuego. Se deberá definir la lista de tareas a realizar y los cuestionarios que se aplicarán a los usuarios participantes en las pruebas de jugabilidad. También se deberá determinar el periodo en el que se realizarán estas pruebas.
5. CREAR PRUEBAS DE PUBLICACIÓN EN MERCADOS	Crear grupos de pruebas de publicación en mercado por plataforma o tienda de aplicaciones para cumplir sus las políticas de publicación.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *D7* y *F2.4*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.31 - Productos de trabajo de la actividad Diseño Detallado de Software

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Especificación de Requerimientos [validada]	Requerimientos
Documento de Diseño de Juego [validado]	Diseño del Videojuego
Descripción de Arquitectura de Software [validada]	Arquitectura
Políticas de Publicación en Mercados	Análisis de Mercado
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Documento de Diseño Técnico	Construcción de Software Integración y Pruebas de Software
PRODUCTOS INTERNOS	
NOMBRE	
Diseño Detallado de Software	
Casos de Pruebas de Integración	
Escenarios y Casos de Pruebas de Usabilidad	
Escenarios y Casos de Pruebas de Jugabilidad	
Pruebas de Publicación en Mercados	
Listas de tareas a realizar	
Cuestionarios para aplicar	
Registro de Trazabilidad	
Configuración del Software	

4.6.7. Fase de Producción

En la fase de *Producción* se construirán los diferentes *assets* que requerirá el videojuego (texturas, modelos, animaciones, cinemáticas, escenarios, etc.) de acuerdo a la línea de producción, así como los componentes de software específicos con base en la especificación técnica de desarrollo y diseño detallado de software. Finalmente se integrarán los componentes de software que se han desarrollado y se aplicarán los escenarios o casos de prueba para verificar el correcto funcionamiento del software.

4.6.8. Actividad Construcción de Software

En la actividad *Construcción de Software* se establecen tareas para la construcción de *assets* y componentes de software de acuerdo a la línea de producción y el diseño de pruebas unitarias que permiten detectar y corregir defectos encontrados en los componentes de software desarrollados.

El *Motor de Juego (Game Engine)* representa la estructura interna del videojuego a la que después se le añadirá el contenido. Por lo tanto, como primera tarea se propone construir o adaptar el *Motor de Juego* que se va a utilizar en el desarrollo del videojuego.

Si se considera la opción de utilizar un *Motor de Juego* existente en el mercado, se deberá generar el contenido y después incorporarlo al motor. Para integrar el contenido al *Motor de Juego*, se deberá considerar lo siguiente:

- **Escritura de scripts:** La escritura de *scripts* se realiza con un editor de texto externo a las herramientas del motor gráfico, pero se considera contenido íntimamente relacionado del *Motor de Juego*.
- **Escritura de shaders:** Los *shaders* son programas que son utilizados comúnmente para producir iluminación y sombreado en modelos 3D. Los *shaders* también se escriben con un editor externo.
- **Importación de assets:** Uno de los pasos iniciales dentro del entorno integrado del *Motor de Juego* es añadir al proyecto todo el material generado anteriormente (*assets*) y ajustar sus atributos; como formatos, tamaños de textura, ajuste de propiedades, cálculo de normales, etc.
- **Creación de escenas:** Regularmente se crea una escena por cada nivel o conjunto de menús del videojuego. En la escena se establecen relaciones entre objetos y se crean instancias de ellos.
- **Creación de prefabs:** Los *prefabs* son agrupaciones de objetos que se guardan como un objeto con entidad propia.
- **Optimización de la escena:** La optimización es uno de los pasos fundamentales que se lleva a cabo al final del desarrollo de la escena.

En esta actividad también se deberán construir las herramientas para facilitar la construcción y administración de *assets*, geometría y modelos 3D, *web tools* (para crear bitácoras, información de depuración), herramientas para creación de prototipos rápidos, herramientas para construcción automatizada del software (*game builds*), etc. Todas las herramientas necesarias que pudieran facilitar la construcción del videojuego [36].

Además, se deberán construir los componentes de software necesarios (para jugabilidad, gestión de recursos, iluminación, control, cámara, enemigos, eventos, personajes, interfaz de usuario, inteligencia artificial, física, multijugador, logros, tutorial de juego, etc.) con base en la especificación técnica de desarrollo y el diseño detallado de software.

Finalmente se deberán diseñar los casos de pruebas unitarias y aplicarlos para verificar que los componentes de software cumplan con la funcionalidad establecida.

En la siguiente TABLA se muestran las tareas propuestas a realizar en la actividad *Construcción de Software*.

TABLA 4.32 - Tareas principales para la actividad Construcción de Software

TAREA	DESCRIPCIÓN
1. CONSTRUIR O ADAPTAR MOTOR DE JUEGO	Construir o adaptar el motor de juego (si se requiere). Se deberá crear la estructura interna del videojuego, en la que después se añadirá el contenido.
2. CONSTRUIR HERRAMIENTAS (TOOLS)	Construir herramientas (tools) para facilitar la implementación e integración del contenido.
3. CONSTRUIR ASSETS DEL VIDEOJUEGO	Construir assets de arte de acuerdo a la línea de producción. Los assets de arte podrían incluir: textos, modelos 3D, animaciones, cinemáticas, interfaz de usuario, mercadotecnia, entre otros. También se deberán construir los assets de audio que requiera el videojuego (música, efectos de sonido, voces, diálogos, etc.).
4. CONSTRUIR COMPONENTES DE SOFTWARE	Construir los componentes de software necesarios con base en la especificación técnica de desarrollo y el diseño detallado de software. Se deberán construir los componentes de software para los siguientes elementos: jugabilidad (gameplay), control, cámara, enemigos, eventos, personajes, interfaz de usuario (IU), inteligencia artificial (IA), física, multijugador, logros (achievements), tutorial de juego, entre otros.
5. DISEÑAR CASOS DE PRUEBAS UNITARIAS	Diseñar los casos de pruebas unitarias y aplicarlos para verificar que los componentes de software cumplan con la funcionalidad establecida.
6. CORREGIR ERRORES EN LOS COMPONENTES DE SOFTWARE	Corregir defectos encontrados en los componentes de software hasta lograr pruebas exitosas.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de C5, C6, C7, F3.2 y F3.3. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.33 - Productos de trabajo de la actividad Construcción de Software

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto	Planeación del Proyecto
Documento de Diseño Técnico	Diseño Detallado de Software
Especificación de Requerimientos	Requerimientos
Documento de Diseño de Juego	Diseño de Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Motor de Juego (Game Engine)	Integración y Pruebas de Software
Componentes de Software	Integración y Pruebas de Software
Assets	Integración y Pruebas de Software
PRODUCTOS INTERNOS	
NOMBRE	
Herramientas (Tools)	
Casos de Pruebas Unitarias	

4.6.9. Actividad Integración y Pruebas de Software

Integración. En la actividad *Integración* se establecen tareas para integrar los assets y componentes de software que se han desarrollado y posteriormente aplicar los escenarios y casos de pruebas para verificar el correcto funcionamiento del software.

Es importante tener un proceso para crear el software (*game builds*) de forma regular, de modo que las características principales (*game features*) y los *assets* puedan ser revisados dentro del juego. Si no se construye el software de forma regular, el equipo de desarrollo no puede hacer revisiones adecuadas de la funcionalidad del juego o asegurarse de que los *assets* se visualicen correctamente [17].

Si hay alguna dificultad para crear el software, esto puede indicar que existen errores (*bugs*) en el juego que impiden la compilación del juego. Los desarrolladores podrían no darse cuenta de esos errores hasta que intenten compilarlo. Si pasa mucho tiempo sin crear el software (*game builds*), los errores críticos permanecerán y será más complicado lidiar con el código conforme el desarrollo avanza [17].

La línea de producción de software se define en la fase de *Preproducción*, y se implementa tan pronto como sea posible en la fase de *Producción* (usualmente cuando los *assets* y el código están disponibles y pueden ser utilizados para crear una versión del juego). Mientras más pronto se compile el juego, más pronto se podrá comenzar a corregir errores y hacer mejoras en el juego [17].

Un proceso flexible para crear software permite que se puedan crear versiones especiales bajo pedido. Por ejemplo, el área de mercadotecnia podría requerir un demo con solo ciertos niveles y personajes disponibles para presentarlo en alguna una conferencia o evento especializado. O el área de aseguramiento de la calidad podría requerir una versión especial que solo incluya el multijugador para poder realizar pruebas.

Se puede crear un proceso automatizado para construir el software, es usualmente simple de configurar y ahorra mucho tiempo. La creación de software usualmente va acompañado de notas incluidas sobre la construcción del juego, estas notas indican que elementos fueron incluidos y los requerimientos técnicos para poder correr el juego en la plataforma elegida [17].

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Integración*.

TABLA 4.34 - Tareas principales para la actividad Integración

TAREA	DESCRIPCIÓN
1. INTEGRAR SOFTWARE (GAME BUILDS)	Integrar y construir el software del videojuego (game builds) utilizando los componentes de software y assets de acuerdo a la línea de producción. Si es necesario actualizar los casos de pruebas de integración. Se deberán construir versiones jugables gradualmente (alfa, beta, gold).
2. REALIZAR PRUEBAS DE INTEGRACIÓN	Realizar pruebas de software (game builds) usando los casos de pruebas de integración.
3. CORREGIR DEFECTOS ENCONTRADOS	Corregir los defectos encontrados hasta satisfacer criterio de término.
4. VERIFICAR SOFTWARE	Verificar y obtener aprobación del software (game builds) respecto al diseño detallado del software y descripción de arquitectura.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de F3.4. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.35 - Productos de trabajo de la actividad Integración

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto	Planeación del Proyecto
Componentes de Software [corregidos]	Construcción de Software
Especificación de Requerimientos	Requerimientos
Documento de Diseño Técnico	Diseño Técnico del Videojuego
Registro de Trazabilidad [en línea base]	Diseño Técnico del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Software (Game Build) [verificado]	Pruebas de Software
Casos de Pruebas de Integración [actualizados]	Pruebas de Software
Reporte de Pruebas [defectos eliminados]	Pruebas de Software
Base de Datos de Errores [actualizada]	Pruebas de Software
PRODUCTOS INTERNOS	
NOMBRE	
Listas de Verificación	
Solicitud de Cambio	
Configuración del Software	

Pruebas de Software. En la actividad *Pruebas de Software* se aplican los escenarios y casos de pruebas para verificar el correcto funcionamiento del software (game builds).

En este punto del desarrollo, muchos elementos en el juego necesitarán ser probados con el objetivo de encontrar defectos y errores críticos. Las *Pruebas de Software* son una parte integral del desarrollo de videojuegos, así que es importante no reducir los tiempos de pruebas planeados [17].

Para evaluar la calidad de un videojuego, se podrán aplicar diferentes tipos de pruebas [31]:

- **Grupo Focal (Focus Group):** Se refiere a las sesiones donde jugadores potenciales son entrevistados sobre lo que les gusta y no les gusta, en un intento por determinar si les gustaría una idea de un juego que un estudio está considerando.
- **Pruebas de Aseguramiento de la Calidad.** El objetivo principal de estas pruebas es encontrar defectos y errores críticos en el juego.
- **Pruebas de Usabilidad.** El objetivo de estas pruebas es determinar si la interfaz (control, hud, menús, etc.) es intuitiva y fácil de utilizar.
- **Pruebas de Jugabilidad.** El objetivo principal de estas pruebas es evaluar que tan divertido es el videojuego.

Antes de que las pruebas de aseguramiento de la calidad comiencen, se deberá determinar la forma de rastrear y reportar errores. En general se sigue el siguiente proceso [17]:

- **Encontrar.** Un asegurador de la calidad encuentra un error y lo registra en la base de datos de errores.
- **Corregir.** Alguien asigna el error a una persona para que lo corrija. Esta persona corrige el error y lo marca como *corregido* en la base de datos.
- **Verificar.** El asegurador de la calidad obtendrá una nueva versión del juego (*game build*), y verifica que el error se haya corregido. Después lo registra como *verificado* en la base de datos.
- **Cerrar.** El administrador confirma que todos los errores se hayan verificado. El error no aparecerá cuando se hagan búsquedas en la base de datos.

Las pruebas de usabilidad consisten principalmente en:

- Seleccionar un grupo de usuarios finales con base en el perfil de usuario identificado inicialmente.
- Recrear y ejecutar los escenarios o casos de prueba definidos en la actividad *Diseño Detallado de Software*.

- Observar al usuario para llevar el registro de las tareas que puede cumplir con facilidad, las tareas que no puede cumplir y las diferentes reacciones expresadas al realizar cada una de ellas.
- Posteriormente, aplicar un cuestionario para recopilar el punto de vista de los usuarios, comentarios, opiniones o quejas sobre la interfaz de usuario del videojuego.
- Finalmente analizar los resultados y obtener una retroalimentación para tomar medidas que beneficien la mejora en la experiencia de usuario.

Para asegurar que las pruebas de jugabilidad sean buenas, se deberán contestar las siguientes preguntas [31]:

- ¿Por qué se está realizando la prueba de jugabilidad?
- ¿Quiénes deberían realizar estas pruebas?
- ¿En dónde se deberían de realizar?
- ¿Qué es lo que se busca?
- ¿Cómo se obtendrá la información que se necesita?

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Pruebas de Software*.

TABLA 4.36 - Tareas principales para la actividad Pruebas de Software

TAREA	DESCRIPCIÓN
1. APLICAR ESCENARIOS Y CASOS DE PRUEBAS	Aplicar los escenarios y casos de pruebas de usabilidad y jugabilidad. También se deberán aplicar las pruebas de publicación en mercado para verificar que el videojuego cumple con las políticas de publicación (si es necesario, o para videojuego final).
2. CORREGIR DEFECTOS	Corregir posibles defectos o dificultades encontrados en las pruebas de usabilidad, jugabilidad o de publicación hasta satisfacer criterios de cumplimiento.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de C8 y F3.5. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.37 - Productos de trabajo de la actividad Pruebas de Software

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto	Planeación del Proyecto
Documento de Diseño Técnico	Diseño Detallado de Software
Software (Game Build) [verificado]	Integración
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Software (Game Builds) [corregido]	Integración
Base de Datos de Errores [actualizada]	Integración
Reporte de Pruebas [actualizado]	Integración
Videos del Grupo Focal	Integración
Gráfica de Trabajo Pendiente	Integración
PRODUCTOS INTERNOS	
NOMBRE	
Escenarios y Casos de Pruebas de Usabilidad [aplicados]	
Escenarios y Casos de Pruebas de Jugabilidad [aplicados]	
Pruebas de Publicación en Mercado [aplicados]	

4.6.10. Fase de Posproducción

En la fase de *Posproducción* se llevará a cabo la entrega del producto de acuerdo a los protocolos de entrega y se propone una actividad de mantenimiento del videojuego en donde se realizará de manera sistemática la modificación del software, de acuerdo a las peticiones de mantenimiento de clientes o usuarios finales (jugadores).

4.6.11. Actividad Entrega del Producto

En esta actividad se lleva a cabo la *Entrega del Producto* de acuerdo a los protocolos de entrega establecidos con el cliente.

La primera tarea que se propone para esta actividad es la creación del manual del juego y los tutoriales de juego necesarios.

Antes de realizar la entrega, una tarea importante es organizar todos los archivos fuente de los *assets* y el código en un *kit de cierre (closing kit)*. Los archivos fuente son necesarios si se requiere en algún momento *remasterizar* el juego, crear un parte o actualizar contenido, portar el juego a otra plataforma, crear una versión *local* en otro idioma, desarrollar una secuela, o cualquier otro tipo de proceso que requiera los archivos fuente de *assets* y códigos originales del juego [17].

Los *kits de cierre* se crean cuando el código del juego es liberado de modo que los desarrolladores puedan incluir los *assets* finales y el código fuente. Si alguna actualización o parches se crean después del *kit de cierre*, se podrían agregar después. El contenido de un *kit de cierre* incluye [17]:

- **Assets.** Todos los *assets* de texto, audio y arte del juego en el *kit de cierre*. También, todos los archivos fuente originales son necesarios, de tal forma que puedan hacerse cambios a los *assets in-game*. Los *asset in-game* finales también son necesarios.

- **Assets de Texto.** Incluye *assets de texto (in-game)* en todos los lenguajes, archivos de ayuda (README), mensajes de error.
- **Assets de Narración (Voiceover).** Incluye archivos de audio sin comprimir en todos los lenguajes, narración y guiones de cinemáticas, especificación técnica de las narraciones.
- **Assets de Arte.** Incluye *assets* finales de arte, archivos fuente para todos los *assets*.
- **Assets de Cinemáticas.** Incluye cinemáticas finales (*in-game*), *video codecs* y reproductores de video, cinemáticas sin comprimir, archivos fuente de cinemáticas. Las pistas de audio, música, narración y efectos de sonido utilizadas en las cinemáticas se deberán incluir por separado y sin comprimir.
- **Assets de Localización.** Incluye texto traducido, archivos de narración traducidos (versiones comprimidas y sin comprimir), glosarios de localización.
- **Assets de Empaquetado.** Incluye los *assets* para las cajas, manuales, capturas de pantalla utilizados para el empaquetado.
- **Herramientas (Tools).** Se refiere a todas las herramientas que se necesitaron para crear los *assets in-game* finales. Por ejemplo, si un *plug-in* se utilizó para comprimir las texturas para los modelos de personajes se debería incluir. Si software comercial se necesitó para crear cualquier *asset* del juego, la información y la versión correcta se debe anotar.
- **Código Fuente.** El código fuente es un componente muy importante del *kit de cierre*. Incluye versión *gold* del juego, código fuente (incluyendo documentación sobre cómo compilar el juego) y código fuente de las herramientas utilizadas en el juego.
- **Documentación del Juego.** La documentación del juego es importante incluirla para cualquiera que quiera crear nuevo contenido o *ports*. Incluye todos los documentos de diseño y documentos técnicos generados durante el desarrollo del videojuego.
- **Guías Técnicas.** Incluye la línea de producción, instrucciones para crear el software (*game builds*), lista de requerimientos de software para la línea de producción, requerimientos de hardware, instrucciones para instalar y utilizar las herramientas que fueron desarrolladas.
- **Información General del Producto.** La información general del producto proporciona una visión general del juego.

Se deben organizar los *assets* del juego, código, documentación y herramientas en carpetas separadas. La FIGURA 4.14 muestra la estructura básica de directorios de un *kit de cierre*.

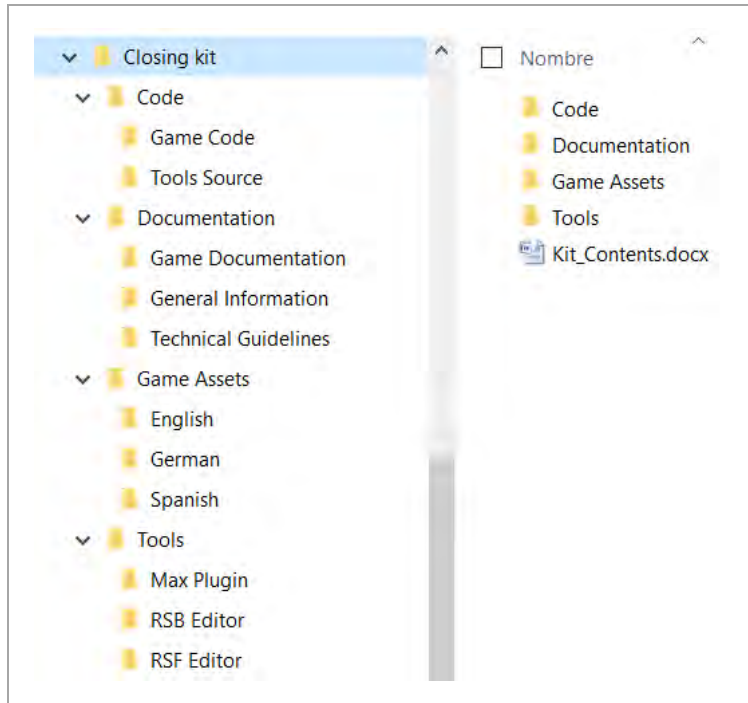


FIGURA 4.14 - Estructura básica de directorios de un kit de cierre

Cuando el *kit de cierre* está completo, se crean copias y se almacenan en diferentes sitios (una copia con el desarrollador, otra copia con el *publisher*). Las etiquetas de cada disco, llevan la siguiente información:

- Nombre del juego
- Versión del juego
- Plataforma objetivo del juego
- Fecha de creación
- Número de disco, escrito como "x" de "x" discos
- Descriptor de contenido ("documentación", "cinemáticas", "código fuente", etc.)

La última tarea propuesta para esta actividad es llevar a cabo la entrega de acuerdo a los protocolos de entrega establecidos.

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Entrega del Producto*.

TABLA 4.38 - Tareas principales para la actividad Entrega del Producto

TAREA	DESCRIPCIÓN
1. REALIZAR MANUAL Y TUTORIALES DEL VIDEOJUEGO	Realizar el manual y tutoriales del videojuego.
2. CREAR KIT DE CIERRE	Realizar el <i>kit de cierre</i> del videojuego.
3. LLEVAR A CABO LA ENTREGA	Llevar a cabo la entrega de acuerdo al protocolo de entrega.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de *F4.1*, *F4.2* y *F4.4*. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.39 - Productos de trabajo de la actividad Entrega del Producto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto	Planeación del Proyecto
Protocolos de Entrega	Planeación del Proyecto
Software (Gold Master Build)	Pruebas de Software
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Kit de Cierre [entregado]	Equipo de Trabajo Cliente
Configuración del Software [entregada]	Cliente
PRODUCTOS INTERNOS	
NOMBRE	
Manual	
Configuración del Software	

4.6.12. Actividad Mantenimiento del Videojuego

En la actividad *Mantenimiento del Videojuego* se realiza de manera sistemática la modificación del software, de acuerdo a las peticiones de mantenimiento de clientes o usuarios finales (jugadores).

En la siguiente TABLA se muestran las tareas propuestas a realizar para la actividad *Mantenimiento del Videojuego*.

TABLA 4.40 - Tareas principales para la actividad Mantenimiento del Videojuego

TAREA	DESCRIPCIÓN
1. REALIZAR ACUERDO DE MANTENIMIENTO	Realizar el acuerdo de mantenimiento del videojuego. Se deberá describir el tipo de mantenimiento que se hará y el alcance que tendrá en cuanto a plataformas, dispositivos, servicios y tiempo. También se deberá definir el protocolo de petición de modificación con base en el acuerdo de mantenimiento.
2. ANALIZAR Y EVALUAR LA PETICIÓN DE MODIFICACIÓN	Analizar, evaluar y priorizar la petición de modificación. Se deberá analizar y evaluar el impacto tecnológico, en costo y tiempo para determinar si la petición es aceptada o rechazada. Además se deberá determinar el tipo de modificación que debe aplicarse (mantenimiento, error de funcionamiento, balanceo del juego, actualización, expansión, contenido descargable, etc.)
3. DETERMINAR LAS ACCIONES DE MODIFICACIÓN A SEGUIR	Analizar la petición y determinar las acciones de modificación a seguir.
4. EJECUTAR LAS MODIFICACIONES	Ejecutar las modificaciones a la configuración de software con base en las acciones de modificación establecidas. Identificar los <i>assets</i> y componentes de software que fueron afectados.

5. REALIZAR PRUEBAS UNITARIAS	Ejecutar pruebas unitarias a los componentes de software que fueron afectados. Se deberán crear casos de pruebas unitarias y aplicarlos para verificar que los componentes de software cumplan con la nueva funcionalidad establecida.
6. CORREGIR DEFECTOS ENCONTRADOS	Corregir defectos encontrados hasta lograr pruebas exitosas.
7. INTEGRAR COMPONENTES DE SOFTWARE	Integrar nuevamente los componentes de software.
8. REALIZAR PRUEBAS DE SOFTWARE	Realizar pruebas de software (integración, jugabilidad, usabilidad, etc.) usando los escenarios y casos de pruebas.
9. VERIFICAR SOFTWARE Y OBTENER APROBACIÓN	Verificar y obtener la aprobación del Software respecto a la Petición de Modificación. Además, se deberá validar y obtener aprobación del software por parte del cliente.
10. REEMPLAZAR EL SOFTWARE	Llevar a cabo el reemplazo del software (game build) y realizar su distribución.
11. FORMALIZAR CONCLUSIÓN	Formalizar conclusión del mantenimiento del videojuego con base en el acuerdo de mantenimiento.

En esta actividad se propone añadir elementos para cubrir principalmente propiedades de C9 y F4.5. Los productos de trabajo que se utilizan y generan en esta actividad se muestran en la siguiente TABLA.

TABLA 4.41 - Productos de trabajo de la actividad Mantenimiento del Videojuego

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Enunciado del Trabajo	Cliente
Software (Gold Master Build)	Cliente
Petición de Modificación [solicitada]	Cliente
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Acuerdo de Mantenimiento	Cliente
Protocolo de Petición de Modificación	Cliente
Software (Gold Master Build) [disponible, modificado]	Cliente
Documento de Término de Mantenimiento [aceptado]	Cliente
PRODUCTOS INTERNOS	
NOMBRE	
Petición de Modificación [priorizada]	
Acciones de Modificación	
Elementos a modificar en la Configuración de Software	
Componentes de Software [modificados]	
Casos de Pruebas Unitarias	
Escenarios y Casos de Pruebas	
Reporte de Pruebas	
Listas de Verificación	
Listas de Validación	
Software (Game Build) [probado]	
Estrategia de Control de Versiones	
Repositorio del Proyecto	
Configuración del Software	

4.7. Roles involucrados

Los roles propuestos para este modelo de procesos, se definieron comparando y homologando los roles definidos en diferentes procesos de desarrollo de videojuegos documentados ([17], [31]) y considerando el estudio sobre la industria de los videojuegos en México (capítulo 1). La descripción y competencias sugeridas para cada rol se describen en el ANEXO B (GUÍA DE IMPLEMENTACIÓN DEL MODELO DE PROCESOS).

TABLA 4.42 - Roles involucrados

ROLES	
PRODUCCIÓN	
PRODUCTOR (ADMINISTRADOR DEL PROYECTO)	
ARTE	
ANIMADOR	ARTISTA DE PERSONAJES (MODELADOR)
ARTISTA CONCEPTUAL	ARTISTA DE TEXTURAS
ARTISTA DE ENTORNOS 3D	ARTISTA LÍDER
ARTISTA DE ILUMINACIÓN	ARTISTA TÉCNICO
ARTISTA DE INTERFAZ GRÁFICA	RIGGER
ARTISTA DE MERCADOTECNIA	
PROGRAMACIÓN	
PROGRAMADOR DE ASEGURAMIENTO DE LA CALIDAD	PROGRAMADOR DE INTERFAZ DE USUARIO
PROGRAMADOR DE AUDIO	PROGRAMADOR DE JUEGO EN LÍNEA
PROGRAMADOR DE HERRAMIENTAS	PROGRAMADOR DE MOTOR DE JUEGO
PROGRAMADOR DE INTELIGENCIA ARTIFICIAL	PROGRAMADOR LÍDER
DISEÑO	
CREADOR DEL MANUAL Y TUTORIAL DE JUEGO	DISEÑADOR LÍDER
DISEÑADOR DE INTERFAZ DE USUARIO	ESCRITOR
DISEÑADOR DE NIVEL (WORLD BUILDER)	GUIONISTA
DISEÑADOR DE SISTEMAS	
AUDIO	
COMPOSITOR	DISEÑADOR DE AUDIO
ASEGURAMIENTO DE LA CALIDAD	
ASEGURADOR DE LA CALIDAD (TESTER)	ASEGURADOR DE LA CALIDAD LÍDER
MERCADOTECNIA	
MERCADÓLOGO	
FINANCIAMIENTO	
CLIENTE	INVERSIONISTAS
PUBLISHER	
EQUIPOS	
EQUIPO DE TRABAJO	PROGRAMADORES
LÍDERES DE ÁREA	ARTISTAS
DISEÑADORES	

5. CAPÍTULO 5. VALIDACIÓN DEL MODELO DE PROCESOS

5.1. Revisión y evaluación por experto práctico

El modelo de procesos, junto con su guía de implementación cuentan con sustento teórico resultado de una revisión sistemática de literatura. Sin embargo, se requiere de experiencia práctica en el desarrollo de videojuegos independientes que proporcione una retroalimentación y validación con el objetivo de dar mayor robustez a la propuesta.

La validación del modelo propuesto se realizó a través de expertos prácticos, los cuales revisaron y evaluaron el modelo junto con su guía de implementación.

5.1.1. Objetivo

Validar y obtener una retroalimentación del modelo de procesos para el desarrollo de videojuegos independientes y su guía de implementación, a través de la revisión y evaluación por parte de expertos prácticos.

5.1.2. Metodología

Para la validación del modelo de procesos y su guía de implementación se aplicaron cuestionarios de validación que permitieron obtener información que sirvió de retroalimentación. Con el objetivo de conocer el valor y utilidad que aporta, así como la facilidad de su comprensión e implementación en proyectos reales.

Se tuvo contacto con dos expertos prácticos con 6 y 4 años de experiencia en la industria de los videojuegos (trabajando principalmente en estudios de desarrollo de videojuegos internacionales). Se expuso un resumen sobre el trabajo de tesis y la forma en la que fue integrado el modelo de procesos y su guía de implementación.

A cada experto se le entregó la guía de implementación del modelo de procesos para el desarrollo de videojuegos independientes, la cual tuvo que revisar y evaluar con base en su experiencia para responder a las preguntas definidas en el cuestionario de validación.

Finalmente, se analizó el cuestionario de validación y sus conclusiones con el fin de obtener una retroalimentación y documentar los ajustes necesarios al modelo de procesos propuesto y su guía de implementación.

5.1.3. Cuestionarios de validación aplicados a expertos prácticos

5.1.4. Validación del primer experto

El primer experto práctico tiene 6 años de experiencia en la industria y ha trabajado principalmente en estudios de desarrollo de videojuegos internacionales (USC Interactive Media Department, Disney Interactive Media Group y The Odd Gentlemen). Actualmente trabaja como *Programador Líder (Lead Engineer)* y se encarga de la programación de videojuegos, incluyendo el diseño y programación de *herramientas (tools)* para integración y administración de *assets*. Es responsable de las decisiones técnicas durante el desarrollo de videojuegos.

A continuación, se presentan las respuestas al cuestionario de validación:

1. ¿Consideras que la guía para el desarrollo de videojuegos independientes es de fácil comprensión?

R. Sí. La guía es concisa y breve.

2. ¿Consideras que la guía para el desarrollo de videojuegos independientes es fácil implementarla en algún proyecto real?

R. Asumiendo que existe un capital suficiente para llevar el proyecto a cabo, sí.

3. ¿Consideras de valor y utilidad los artefactos propuestos en la guía para el desarrollo de videojuegos independientes?

R. Sí, la delineación clara de roles y necesidades del proyecto son esenciales.

4. ¿Cuál es tu opinión acerca de las actividades de Análisis de Mercado y Financiamiento propuestas en el modelo de procesos?

R. El modelo presentado es bastante detallado. En mi experiencia, el análisis de mercado y financiamiento ocurre simultáneamente con la fase de concepto del juego.

5. ¿Consideras que la guía para el desarrollo de videojuegos independientes minimiza las dificultades presentadas en este tipo de desarrollos?

R. Considero que minimiza el potencial de problemas imprevistos en materia de producción.

6. ¿Cuáles son los aportes más importantes que tiene la guía para el desarrollo de videojuegos independientes?

R. La delineación de procesos de proyecto, de roles, y de actividades es indispensable para tener un equipo de producción ágil y útil para el proyecto. También facilita tener una flexibilidad cuando el proyecto necesite adaptarse a cambios en su financiamiento, duración, etc.

7. Comentarios y observaciones

R. El modelo presentado es preciso, y fácilmente equiparable con documentos y prácticas usadas profesionalmente.

Conclusiones

En mi experiencia profesional, el modelo presentado es el ideal para un estudio independiente de videojuegos. Dependiendo del capital con el que cuente el proyecto, muchos de los roles serán asumidos por una persona, particularmente en un estudio independiente. Sin embargo, es muy clara la utilidad, y el objetivo de este modelo, y es uno que es activamente utilizado en la industria.

5.1.5. Validación del segundo experto

El segundo experto práctico tiene 4 años de experiencia en la industria. Ha trabajado en estudios independientes (Gameloft y CardanFX) como *Programador Líder* y *Líder de Proyecto*, desarrollando principalmente videojuegos para dispositivos móviles. Tiene experiencia y conocimientos en Administración de Proyectos, tanto en proyectos de desarrollo de videojuegos como en software empresarial.

A continuación, se presentan las respuestas al cuestionario de validación:

1. ¿Consideras que la guía para el desarrollo de videojuegos independientes es de fácil comprensión?

R. Esta guía toma como base el proceso de Administración de proyectos establecido por el Project Management Institute. Para cualquier persona que tenga experiencia en la administración de proyectos tendrá facilidad para entender y llevar a cabo la guía en sus 5 grupos de procesos, que para el caso de la guía propuesta de Implementación de Videojuegos resume el primer y segundo grupo de procesos en uno solo, es decir el inicio y la planeación del proyecto.

Sin embargo, tengo que mencionar que la guía específica de manera sencilla y detallada el proceso de Implementación de Videojuego que puede ser entendido por cualquier profesional del área de la Ingeniería de Software.

2. ¿Consideras que la guía para el desarrollo de videojuegos independientes es fácil implementarla en algún proyecto real?

R. La guía de Implementación de Videojuegos propuesta no sólo es de fácil comprensión, también es de fácil implementación.

La razón de mi respuesta es que esta guía toma todos los procesos que en la actualidad se llevan a cabo para no únicamente desarrollar videojuegos sino cualquier tipo de software.

Aunado a estas dos razones la guía toma las mejores prácticas de la Administración de Proyectos de PMI para la consecución del proyecto en tiempo y forma, satisfaciendo las necesidades y expectativas de los involucrados, los accionistas y los clientes finales del producto.

3. ¿Consideras de valor y utilidad los artefactos propuestos en la guía para el desarrollo de videojuegos independientes?

R. Todos los artefactos propuestos son los que actualmente se manejan en la industria de los videojuegos.

Gracias a la lista de artefactos propuesta es como se puede lograr desarrollar un videojuego de calidad que satisfaga las expectativas de los video jugadores y con ello los accionistas reciban las utilidades propuestas.

Uno de los puntos que hay que tener especial cuidado cuando se desarrolla un videojuego, es la Administración de Riesgos ya que en la actualidad en la industria el 80% de los proyectos de desarrollo de videojuegos presentan retrasos con respecto al calendario y/o exceden el presupuesto estimado durante la etapa de planeación.

4. ¿Cuál es tu opinión acerca de las actividades de Análisis de Mercado y Financiamiento propuestas en el modelo de procesos?

R. Las actividades propuestas son el "core" del Enunciado de Trabajo ya que para que los involucrados, específicamente los accionistas, den luz verde al inicio del proyecto deberán estar convencidos que al finalizar el videojuego tendrá un Retorno de Inversión positivo.

Por otra parte en la industria independiente de desarrollo de videojuegos se debe de buscar maneras de financiar los proyectos entre los que destacan los procesos de crowdfunding, donde un grupo de personas aporta recursos económicos en caso de que la propuesta de videojuego a desarrollar les parezca adecuada, convirtiéndose así en los principales Involucrados del proyecto.

5. ¿Consideras que la guía para el desarrollo de videojuegos independientes minimiza las dificultades presentadas en este tipo de desarrollos?

R. La guía propuesta para la implementación de videojuegos efectivamente minimiza las dificultades presentadas a lo largo de un proyecto de desarrollo ya que aplica el ciclo de Deming⁶ adecuadamente con lo cual se tiene un control adecuado a lo largo de todas las fases propuestas por esta guía.

⁶ El ciclo Planificar-Hacer-Verificar-Actuar, también conocido como PDCA (del inglés *Plan-Do-Check-Act*) es la base para la mejora de la calidad, según la definición de Shewhart, modificada por Deming [38].

6. ¿Cuáles son los aportes más importantes que tiene la guía para el desarrollo de videojuegos independientes?

R. La parte más importante de la Administración de Proyectos es el control de este, los objetivos de administrar la implementación de videojuegos son:

- *Terminar el proyecto en tiempo y forma.*
- *Asegurar la calidad del producto terminado*
- *No exceder el presupuesto establecido*
- *Maximizar la rentabilidad y el retorno de inversión para los accionistas.*

Sin embargo, puedo comentar que la mayoría de las empresas independientes de desarrollo de videojuegos no llevan una metodología para el desarrollo de éstos, es debido a eso que muchas fracasan. He ahí la importancia de esta guía para la Implementación de Videojuegos.

7. Comentarios y observaciones

Por último, esta guía se puede acompañar de un Plan de Calidad, que no debe ser confundido con el Aseguramiento de la Calidad que es propia de la fase de Ejecución del proyecto. Este Plan de Calidad tiene como objetivo buscar la calidad en los procesos y encaminar hacia la mejora.

Por otra parte, tengo que comentar que para el desarrollo de videojuegos pueden aplicar otro tipo de metodologías, un ejemplo de estas puede ser SCRUM que es una metodología ágil, XP (Extreme Programming) que es una metodología iterativa, pero la que debe de imperar es Rapid Prototyping.

Conclusiones

Después de haber trabajado 4 años en la industria de los videojuegos en donde la mayoría de las empresas donde trabajé eran independientes y una de ellas era una empresa multinacional sumamente exitosa, únicamente en una de ellas se seguía una metodología de desarrollo de software, y para mi sorpresa no fue la empresa multimillonaria.

En la empresa multinacional no se llevaba administración alguna sobre los desarrollos y en la pequeña empresa independiente por el contrario se utilizaba SCRUM.

La guía propuesta para el desarrollo de videojuegos puede y debería ser utilizada por todas las empresas independientes de desarrollo de software interactivo, aunque cada una podrá elegir otra metodología como las propuestas en mis comentarios.

5.2. Retroalimentación

Después de obtener las respuestas y conclusiones de los expertos prácticos, se procedió a realizar el análisis de cada una de las respuestas con el objetivo de encontrar los posibles ajustes y mejoras para el modelo de procesos.

5.2.1. Análisis de los cuestionarios de validación

A continuación, se muestra el análisis de cada pregunta de los cuestionarios de validación en la siguiente TABLA.

TABLA 5.1 - Análisis de cuestionarios de validación

No	ANÁLISIS DE RESPUESTA
1	<p><i>¿Consideras que la guía para el desarrollo de videojuegos independientes es de fácil comprensión?</i></p> <p>En general, se considera que la guía de implementación del modelo de procesos para el desarrollo de videojuegos independientes es concisa y de fácil comprensión. Sin embargo, uno de los expertos considera que para llevar a cabo la administración del proyecto se requerirá de alguna persona con experiencia en administración de proyectos.</p>
2	<p><i>¿Consideras que la guía para el desarrollo de videojuegos independientes es fácil implementarla en algún proyecto real?</i></p> <p>Ambos expertos coinciden en que la guía podría implementarse en algún proyecto real sin dificultades. Debido a que el modelo propuesto considera los procesos y actividades que en la actualidad se llevan a cabo para desarrollar videojuegos y otros productos de software. Aunque, como considera uno de los expertos, deberá existir el capital suficiente para llevar a cabo el proyecto (la falta de presupuesto es una de las principales dificultades para los estudios independientes, de acuerdo a la investigación sobre el estado de la industria de los videojuegos en México).</p>
3	<p><i>¿Consideras de valor y utilidad los artefactos propuestos en la guía para el desarrollo de videojuegos independientes?</i></p> <p>El modelo de procesos propuesto, considera los productos o artefactos de otros modelos evaluados y respeta los nombres de los productos utilizados actualmente en la industria de los videojuegos. Ambos expertos coinciden en que los artefactos propuestos en el modelo de procesos son de gran utilidad, y son los utilizados actualmente en la industria de los videojuegos.</p> <p>Uno de los expertos considera que se debe hacer énfasis en la <i>Administración de Riesgos</i> ya que, actualmente el 80% de los proyectos de desarrollo de videojuegos presentan retrasos con respecto al calendario y/o exceden el presupuesto estimado.</p>

<p>4</p>	<p><i>¿Cuál es tu opinión acerca de las actividades de Análisis de Mercado y Financiamiento propuestas en el modelo de procesos?</i></p> <p>Uno de los expertos analiza que en el diagrama general del modelo de procesos, en la fase de <i>Concepto</i>, la actividad <i>Análisis de Mercado y Financiamiento</i> está inmediatamente después de la actividad <i>Concepto del Videojuego</i>. Sin embargo, de acuerdo a su experiencia, el análisis de mercado ocurre simultáneamente con la actividad <i>Concepto del Videojuego</i>.</p> <p>Por otra parte, la actividad <i>Financiamiento</i> se considera fundamental en el caso de los estudios y desarrolladores independientes, siendo el modelo de financiamiento <i>crowdfunding</i> una de las mejores opciones para estos estudios.</p>
<p>5</p>	<p><i>¿Considera que la guía para el desarrollo de videojuegos independientes minimiza las dificultades presentadas en este tipo de desarrollos?</i></p> <p>Los dos expertos coinciden en que la guía propuesta minimiza las principales dificultades presentadas durante el desarrollo de videojuegos. Debido a que el proceso de <i>Administración del Proyecto</i> permite llevar un control adecuado a lo largo de todas las fases del proceso de <i>Implementación del Videojuego</i>. Además, la guía de implementación está destinada para ser utilizada por los estudios independientes para establecer procesos para implementar cualquier enfoque o metodología de desarrollo, incluyendo por ejemplo, ágil, evolutivo, incremental, desarrollo dirigido por pruebas, etc. basada en las necesidades del estudio o del proyecto.</p>
<p>6</p>	<p><i>¿Cuáles son los aportes más importantes que tiene la guía para el desarrollo de videojuegos independientes?</i></p> <p>Uno de los expertos considera que los aportes más importantes del modelo de procesos propuesto y su guía de implementación, son la delineación de actividades de los procesos de <i>Administración del Proyecto e Implementación de Videojuego</i>, los roles propuestos y las tareas específicas para cada actividad propuestas.</p> <p>Debido a que la gran mayoría de los estudios independientes no tienden a seguir ningún método de desarrollo sistemático y trabajan principalmente <i>ad-hoc</i>. Esta guía podría ser útil y podría ser un modelo de referencia básico para el desarrollo de videojuegos independientes. Además, proporciona flexibilidad cuando el proyecto requiera adaptarse a cambios en alcance, requerimientos, diseño, financiamiento, etc.</p>
<p>7</p>	<p><i>Comentarios y observaciones</i></p> <p>Para uno de los expertos el modelo presentado es preciso, y fácilmente equiparable con productos o artefactos, y prácticas utilizadas actualmente.</p> <p>El segundo experto validador, propone complementar la guía con un <i>Plan de Calidad</i> con el objetivo de buscar la calidad en los procesos y encaminar hacia la mejora. Además, considera que se deberían utilizar metodologías ágiles como <i>SCRUM</i> y <i>XP (Extreme Programming)</i>, así como <i>Rapid Prototyping</i> la cual considera ideal para el desarrollo de videojuegos independientes</p>

8	<p><i>Conclusiones</i></p> <ul style="list-style-type: none"> ▪ El modelo presentado podría ser implementado sin dificultad en un proyecto real por los estudios independientes. ▪ La guía propuesta minimiza las principales dificultades presentadas durante el desarrollo de videojuegos. Debido a que el proceso de <i>Administración del Proyecto</i> permite llevar un control adecuado a lo largo de todas las fases del proceso de <i>Implementación del Videojuego</i>. ▪ Dependiendo del capital con el que cuente el proyecto, muchos de los roles serán asumidos por una persona, particularmente en estudios pequeños. Las actividades, tareas y productos propuestos en el modelo de procesos y su guía de implementación se utilizan en la industria actualmente. ▪ Cada estudio podrá elegir cualquier enfoque o metodología de desarrollo, incluyendo por ejemplo, ágil, evolutivo, incremental, desarrollo dirigido por pruebas, etc. basada en sus necesidades. ▪ Se considera fundamental la <i>Administración de Riesgos</i> debido a que actualmente los proyectos de desarrollo de videojuegos presentan retrasos con respecto al calendario y/o exceden el presupuesto estimado.
---	---

5.2.2. Propuestas y ajustes del modelo de procesos

El primer ajuste a considerar es en el diagrama general del modelo de procesos, en la fase de *Concepto*, la actividad *Análisis de Mercado y Financiamiento* está inmediatamente después de la actividad *Concepto del Videojuego*. Sin embargo, el *Análisis de Mercado* ocurre simultáneamente con la actividad *Concepto del Videojuego*. Por lo tanto, se ajustará el modelo para indicar que estas actividades se podrán realizar en paralelo.

La implementación del modelo de procesos propuesto, se podría complementar con un *Plan de Calidad*⁷ como lo sugiere uno de los expertos. Una vez que el estudio de desarrollo de videojuegos ha decidido desarrollar un *Plan de Calidad*, deberá identificar las entradas y el alcance del *Plan de Calidad* [37]. Los aspectos a tener en cuenta en la preparación del plan son los siguientes:

- Identificar a la persona responsable de la preparación del *Plan de Calidad*.
- Documentación del *Plan de Calidad*.
- Responsabilidades.
- Coherencia y compatibilidad del contenido y formato con el alcance, los elementos de entrada y las necesidades de los usuarios previstos.
- Presentación y estructura.

⁷ Según la norma UNE-EN ISO 9000:2005, sistemas de gestión de la calidad, fundamentos y vocabulario, se define Plan de calidad como **documento que especifica qué procedimientos y recursos asociados deben aplicarse, quién debe aplicarlos y cuándo deben aplicarse a un proyecto, producto, proceso o contrato específico.**

Hay varias situaciones en que los planes de la calidad pueden ser útiles o necesarios, por ejemplo:

- Mostrar cómo el sistema de gestión de la calidad de la organización se aplica a un caso específico.
- Cumplir con los requisitos legales, reglamentarios o del cliente.
- En el desarrollo y validación de nuevos productos o procesos.
- Demostrar, interna y/o externamente, cómo se cumplirá con los requisitos de calidad.
- Organizar y gestionar actividades para cumplir los requisitos de calidad y objetivos de la calidad.
- Optimizar el uso de recursos para el cumplimiento de los objetivos de la calidad.
- Minimizar el riesgo de no cumplir los requisitos de calidad.
- Utilizarlos como base para dar seguimiento y evaluar el cumplimiento de los requisitos para la calidad.
- En ausencia de un sistema de la gestión de calidad documentado.

Otra propuesta que surge es poner énfasis en la *Administración de Riesgos* ya que en la actualidad, en la industria de los videojuegos, el 80% de los proyectos presentan retrasos con respecto al calendario y/o exceden el presupuesto estimado durante la etapa de planeación. Principalmente se debe a un error frecuente en esta industria, provocado por una mala definición del alcance del proyecto, que se agrava aún más por la funcionalidad agregada constantemente durante el desarrollo del videojuego (en la fase de *Producción*), lo cual incrementa el tamaño del proyecto.

La guía de implementación está destinada para ser utilizada por los estudios independientes para establecer procesos para implementar **cualquier enfoque o metodología de desarrollo**, incluyendo por ejemplo, ágil, evolutivo, incremental, desarrollo dirigido por pruebas, etc. basada en las necesidades del estudio o del proyecto. Por lo tanto, considera las metodologías ágiles sugeridas por uno de los expertos (*SCRUM, XP*), así como *Rapid Prototyping*.

Particularmente en la actividad *Diseño del Videojuego* en la fase de *Preproducción*, no todas las tareas propuestas se podrán realizar en todos los desarrollos. Se deberá indicar que la realización y la complejidad de estas tareas está en función del tipo de videojuego a desarrollar. Por ejemplo, un videojuego como *Tetris* no requirió de la creación de la historia, personajes o diálogos. En este tipo de videojuegos las mecánicas y la jugabilidad son lo más importante.

CONCLUSIONES Y TRABAJO A FUTURO

Conclusiones

El principal objetivo de este trabajo de investigación, fue integrar un modelo de procesos para el desarrollo de videojuegos independientes basado en propuestas actuales y experiencia de profesionales con el fin de minimizar las principales dificultades que se presentan en este tipo de desarrollos.

Para cumplir con este objetivo, se identificaron las principales diferencias de la industria de los videojuegos y la industria del software, las principales dificultades en el desarrollo de videojuegos independientes y el estado de la industria de los videojuegos en México (mediante un sondeo en línea). Con base en estas dificultades y características, se definieron diez características básicas para el desarrollo de videojuegos independientes. Se realizó una revisión sistemática de literatura para conocer el estado de arte sobre propuestas de procesos o metodologías enfocadas en el desarrollo de videojuegos en general e independientes. Cada propuesta fue analizada con el objetivo de obtener las principales fases, actividades y productos clave que integrarían el modelo de procesos propuesto (mediante comparación y homologación).

Posteriormente, se integró el modelo de procesos para el desarrollo de videojuegos independientes, adaptando el estándar *ISO/IEC 29110 Perfil Básico* para cubrir las características básicas definidas. Se complementaron y adaptaron los procesos, actividades, tareas, productos y roles existentes al desarrollo de videojuegos independientes.

El modelo de procesos propuesto fue evaluado por expertos prácticos que revisaron el modelo y su guía de implementación, con el objetivo de obtener una retroalimentación para identificar los posibles ajustes y mejoras.

Como resultado final del presente trabajo de investigación se obtiene el modelo de procesos para el desarrollo de videojuegos independientes con su guía de implementación, enfocado al desarrollo de videojuegos independientes y dirigido a estudios y desarrolladores independientes.

El modelo de procesos integra las diferentes perspectivas de todos los modelos y procesos evaluados, y podría ser utilizado como una referencia completa sobre cómo podría organizarse un modelo de procesos para el desarrollo de videojuegos.

Finalmente, los resultados de este trabajo podrían ayudar a los estudios y desarrolladores independientes a entender cómo podrían aplicar el estándar *ISO/IEC 29110 Perfil Básico* en el contexto del desarrollo de videojuegos independientes. Sin embargo, el modelo de procesos propuesto no pretende ser una solución universal y deberá ser evaluado en proyectos reales por los estudios y desarrolladores independientes para seguir mejorándolo y ajustándolo a sus necesidades.

Trabajo a futuro

Para el trabajo a futuro se propone lo siguiente:

- Aplicar el modelo de procesos para el desarrollo de videojuegos independientes en proyectos reales que permitan seguir obteniendo retroalimentación.
- Establecer paquetes de puesta en operación. Proponer plantillas para los diferentes productos generados durante el desarrollo de videojuegos.
- Publicar y poner a disposición de manera gratuita el modelo propuesto junto con su guía de implementación en sitios especializados.
- Preparar y enviar artículos a foros o revistas especializadas en el desarrollo de videojuegos.
- Desarrollar una infraestructura de conocimiento para el desarrollo de videojuegos independientes que contenga procesos de desarrollo, plantillas y mejores prácticas.

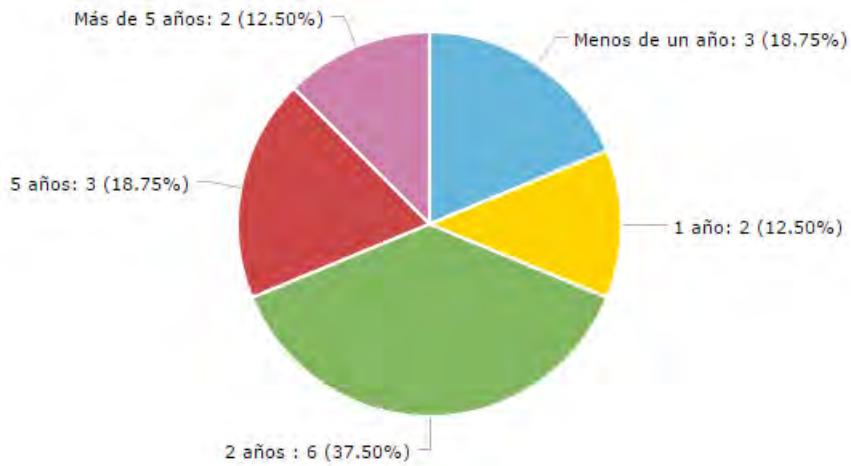
ANEXOS

ANEXO A: SONDEO REALIZADO A LOS ESTUDIOS DE DESARROLLO DE VIDEOJUEGOS INDEPENDIENTES EN MÉXICO

El objetivo de este sondeo es conocer el nivel de madurez de los estudios de desarrollo de videojuegos mexicanos, así como las herramientas de desarrollo y prácticas que llevan a cabo para desarrollar videojuegos.

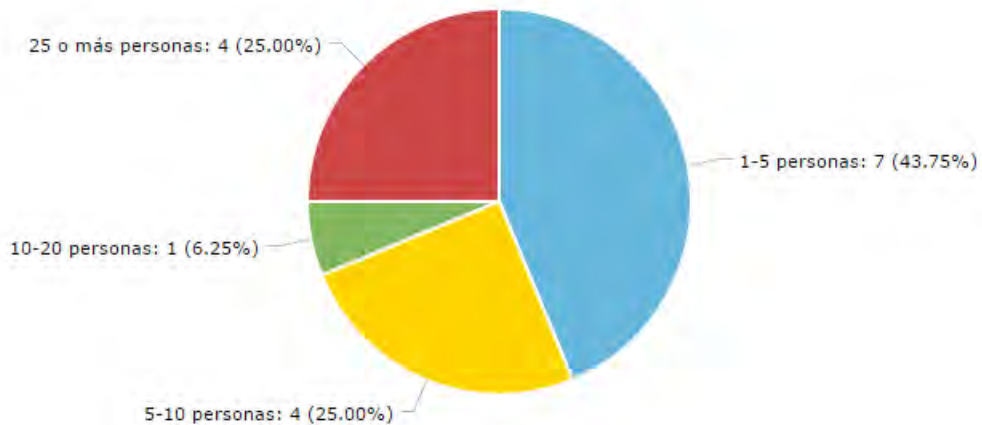
Este sondeo se realizó a través de un cuestionario en línea. En el sondeo participaron 16 estudios. A continuación se muestran las preguntas realizadas junto con sus respuestas.

1. ¿Cuánto tiempo llevan desarrollando videojuegos?



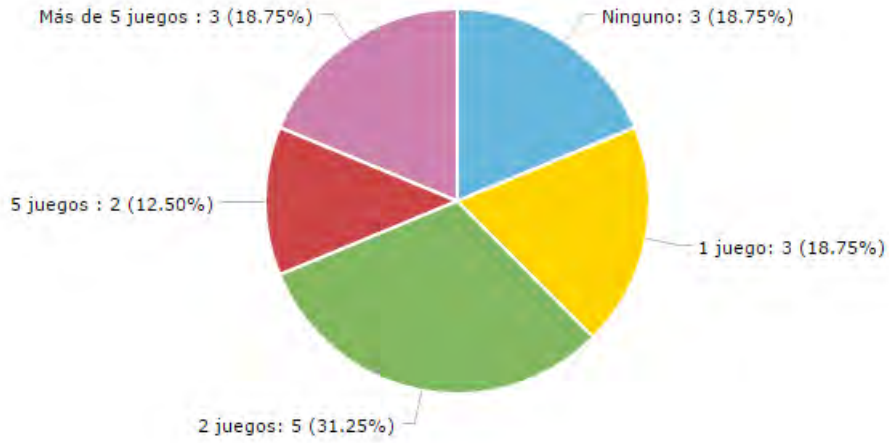
GRAFICA A.1 - Tiempo desarrollando videojuegos

2. ¿Cuántas personas trabajan en su estudio?



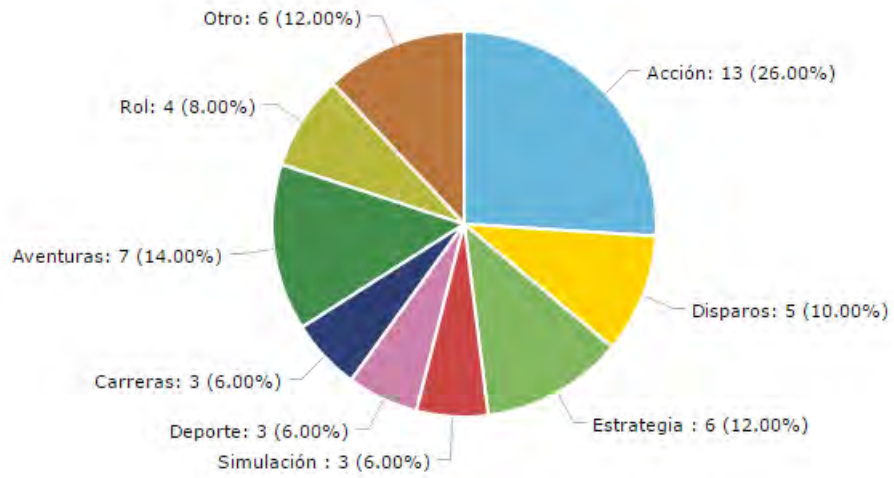
GRAFICA A.2 - Número de personas que trabajan en el estudio

3. ¿Cuántos juegos han desarrollado?



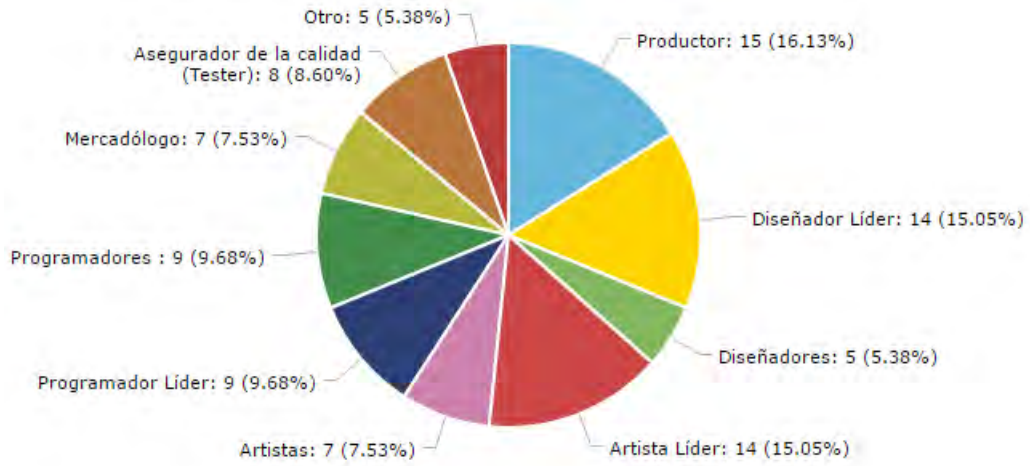
GRAFICA A.3 - Número de juegos que han desarrollado

4. ¿Qué tipo de videojuegos desarrollan?



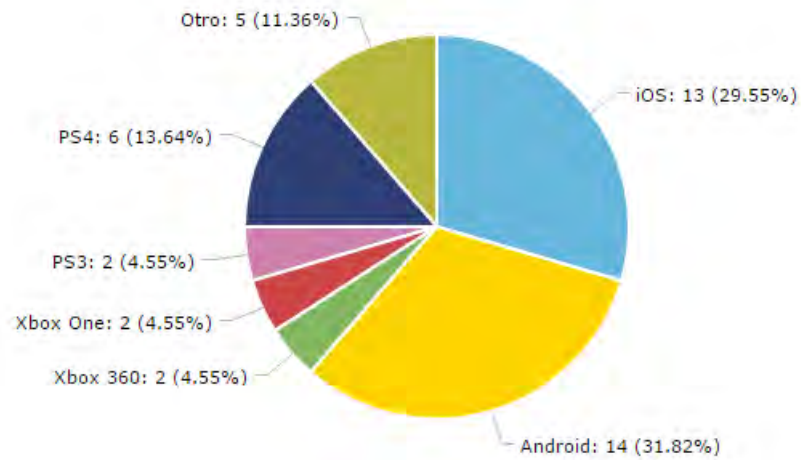
GRAFICA A.4 - Tipos de videojuegos que desarrollan

5. ¿Cuáles son los roles que desempeñan los integrantes de su estudio?



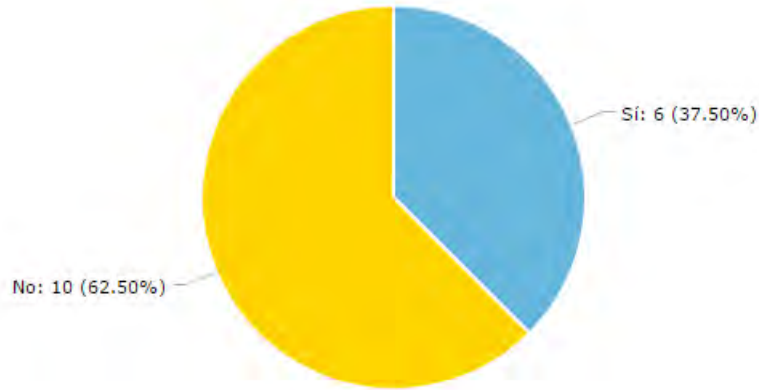
GRAFICA A.5 - Principales roles que desempeñan los integrantes

6. ¿Para qué plataforma desarrollan sus videojuegos?



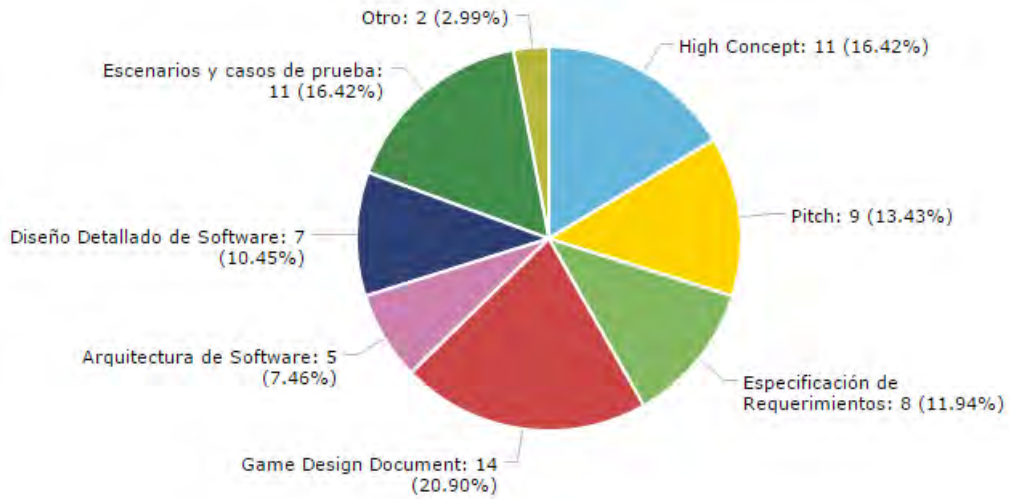
GRAFICA A.6 - Principales plataformas de desarrollo

7. ¿Tienen implementado algún modelo de proceso de software para el desarrollo de videojuegos?



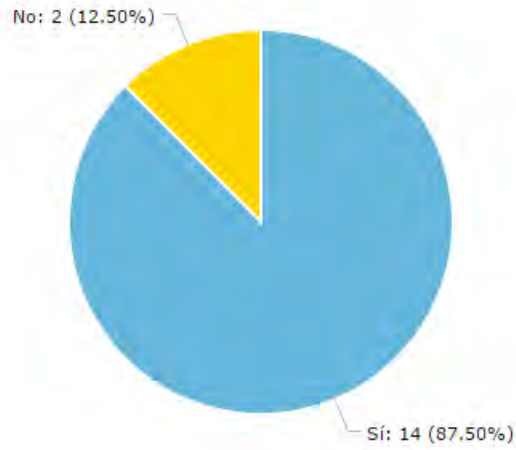
GRAFICA A.7 - Implementación de modelos de procesos de software

8. ¿Cuáles son los principales artefactos que generan durante el desarrollo?



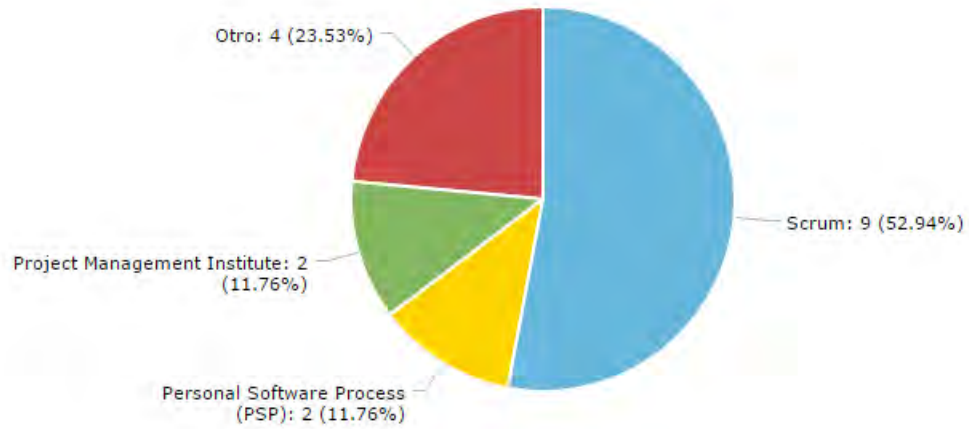
GRAFICA A.8 - Principales artefactos generados durante el desarrollo de videojuegos

9. ¿Llevan la administración de los proyectos que realizan?



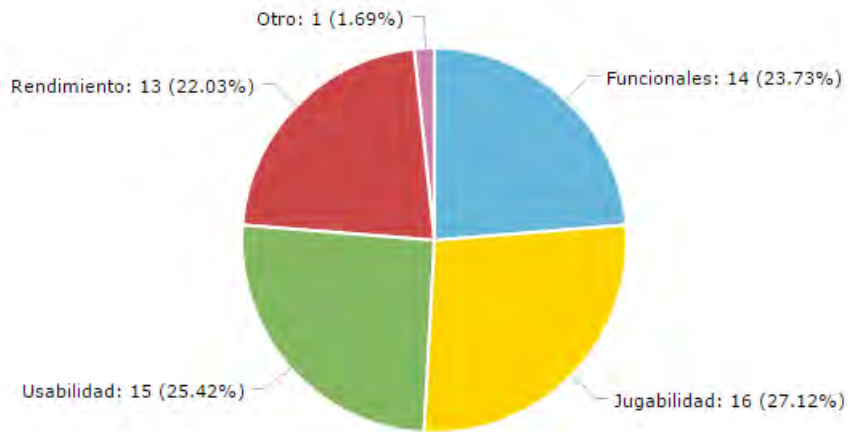
GRAFICA A.9 - Administración de proyectos

10. ¿Qué métodos de administración de proyectos utilizan?



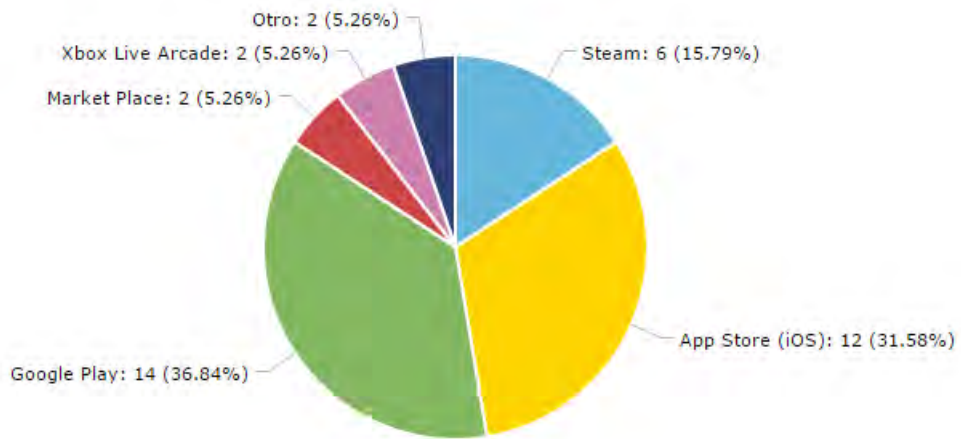
GRAFICA A.10 - Métodos de administración de proyectos utilizados

11. ¿Qué tipo de pruebas realizan a los videojuegos?



GRAFICA A.11 - Tipos de pruebas realizadas durante el desarrollo de videojuegos

12. ¿En qué tienda de aplicaciones o plataformas publican sus videojuegos?



GRAFICA A.12 - Tiendas principales de publicación de videojuegos

13. ¿Realizan algún estudio de mercado antes de cada desarrollo?



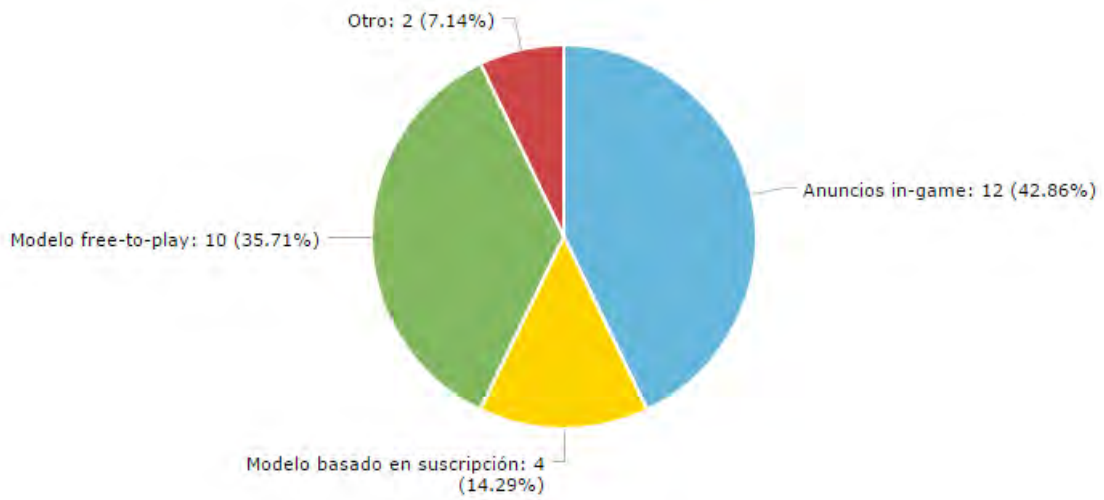
GRAFICA A.13 - Estudios de mercado

14. ¿Tienen estrategias de mercadotecnia?



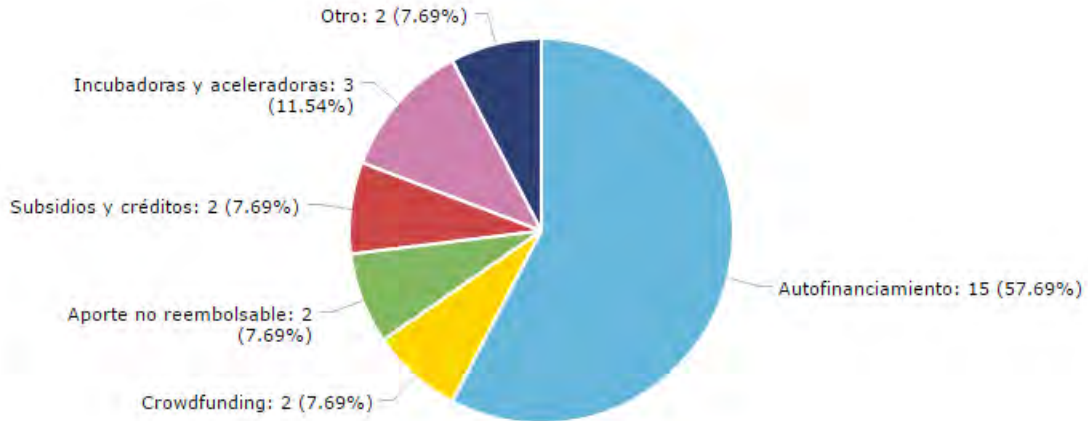
GRAFICA A.14 - Estrategias de mercadotecnia

15. ¿Cuál es su modelo de monetización?



GRAFICA A.15 - Principales modelos de monetización utilizados

16. ¿Cómo financian sus proyectos?



GRAFICA A.16 - Principales modelos de financiamiento utilizados

17. ¿Cuáles son las principales dificultades que tienen en el desarrollo de videojuegos?

TABLA A.1 - Principales dificultades que tienen los estudios durante el desarrollo de videojuegos

PRINCIPALES DIFICULTADES
1. Capacitación, falta de tiempo y recursos para desarrollar el videojuego.
2. Capacitación técnica y financiamiento.
3. Falta de financiamiento.
4. Pipeline de producción.
5. Tiempo y organización.
6. Equipo y licencias.
7. Tiempo disponible de los integrantes, dificultades económicas.
8. El tiempo y el dinero.
9. Capacitación técnica, financiamiento del proyecto, organización para poder terminar los proyectos a tiempo, software y licencias, poca experiencia en la industria de los videojuegos.
10. Lograr un buen promedio de tiempos de desarrollo que compita con las empresas asiáticas y estadounidenses.
11. El financiamiento y el marketing.
12. Tener gente mexicana con las habilidades de AAA.
13. Experiencia faltante.
14. Las integración de los <i>backend services</i> .
15. Pocos talentos.
16. El comienzo es difícil por la falta de organización, capacitación técnica y falta de experiencia. También se requiere de apoyo para financiar los proyectos.

17. ¿Qué tipo de apoyos facilitarían el desarrollo de sus videojuegos?

TABLA A.2 - Apoyos que facilitarían el desarrollo de videojuegos

APOYOS QUE FACILITARÍAN EL DESARROLLO DE VIDEOJUEGOS
1. Capacitación Financiamiento.
2. Apoyos financieros.
3. Mayor financiamiento para el desarrollo de videojuegos.
4. Incentivos fiscales.
5. Dejando de lado lo económico, eventos en donde se nos dé la oportunidad de participar para compartir experiencias y mostrar nuestros proyectos.
6. Apoyo en equipo y apoyo en viajes a convenciones internacionales.
7. <i>Crowdfunding</i> .
8. Apoyos económicos.
9. Capacitación, financiamiento, estrategias de mercadotecnia, eventos dedicados al desarrollo de videojuegos.
10. Apoyo para adquisición de licencias de software especializado o compra de tecnología innovadora (ej. equipo de fotogrametría, equipo de audio holográfico o equipo de <i>motion capture</i>).
11. Apoyos económicos enfocados a proyectos creativos o artísticos, capacitaciones para marketing.
12. Apoyos gubernamentales para dar asesorías a las universidades.
13. De conocimiento.
14. Capacitación y <i>mentorías</i> en temas técnicos de desarrollo, capacitación y <i>mentorías</i> en temas relacionados a monetización.
15. Inversión en contenidos e industrias creativas.
16. La capacitación y el financiamiento son los puntos más importantes. También sería bueno que existieran más eventos sobre desarrollo de videojuegos en México. Apoyar a los estudios que comienzan.

ANEXO B: GUÍA DE IMPLEMENTACIÓN DEL MODELO DE PROCESOS PARA EL DESARROLLO DE VIDEOJUEGOS INDEPENDIENTES

Introducción

La industria de los videojuegos alcanza ya una dimensión de mercado superior a los 74 mil millones de dólares a nivel global. Desde hace años superó a la industria cinematográfica y con tendencias a desmarcarse con facilidad y rapidez.

El rápido crecimiento de este mercado se debe (entre otras razones) al surgimiento de nuevos modelos de distribución por medio de la descarga digital de videojuegos en tiendas de aplicaciones y el incremento en el consumo de todo tipo de videojuegos en dispositivos móviles.

Sin duda esto ha beneficiado a los pequeños *Estudios Independientes* ya que ahora es más fácil publicar un videojuego para diferentes plataformas.

Desarrollar videojuegos es una tarea complicada que involucra la experiencia de profesionales de varias disciplinas incluyendo ciencias de la computación, arte, diseño y negocios. La presión que existe en el desarrollo de videojuegos para llegar al mercado tan pronto como sea posible complica la planeación de los tiempos de desarrollo. Cuestiones de ingeniería de software clásica asociados con el desarrollo de videojuegos puede incluir administración de requerimientos, administración de la configuración, verificación y validación.

Los *Estudios Independientes* deben enfrentar los diversos retos que se presentan durante el desarrollo de videojuegos el cual difiere en varios aspectos al desarrollo tradicional de software. Por lo que desarrollar videojuegos de calidad no es una tarea sencilla.

Con la publicación del estándar ISO/IEC 29110 Perfil Básico se tiene un marco de trabajo dirigido a *Pequeñas Organizaciones (PO)* de hasta 25 personas para el desarrollo de software. El *Modelo de Procesos* propuesto en este trabajo y su guía de implementación están basadas en el estándar ISO/IEC 29110 Perfil Básico (ISO/IEC TR 29110-5-1-2, 2011) y se enfoca en el desarrollo de videojuegos independientes lo cual ofrece los beneficios de minimizar las principales dificultades presentadas en este tipo de desarrollos, crear videojuegos que cumplan sus objetivos en tiempo y costos establecidos y la posibilidad de ser adoptada sin grandes dificultades por *Estudios Independientes*.

Visión General

El objetivo de la guía es facilitar la implementación del *Modelo de Procesos* propuesto que está enfocado en el desarrollo de videojuegos independientes y minimizar las dificultades que se puedan presentar en el desarrollo y mantenimiento de videojuegos, permitiendo que cumplan con las especificaciones requeridas en tiempo y con los costos esperados.

El propósito del proceso de *Administración del Proyecto* es establecer y llevar a cabo de manera sistemática las tareas de un proyecto de *Implementación del Videojuego*, que permitan cumplir con los objetivos del proyecto en calidad, tiempo y costos esperados.

El propósito del proceso de *Implementación del Videojuego (IV)* es la realización sistemática de las actividades de concepto, diseño, construcción, integración y pruebas para productos de software, nuevos o modificados, de acuerdo a los requerimientos especificados.

Ambos procesos están interrelacionados, como se ve en la siguiente FIGURA:

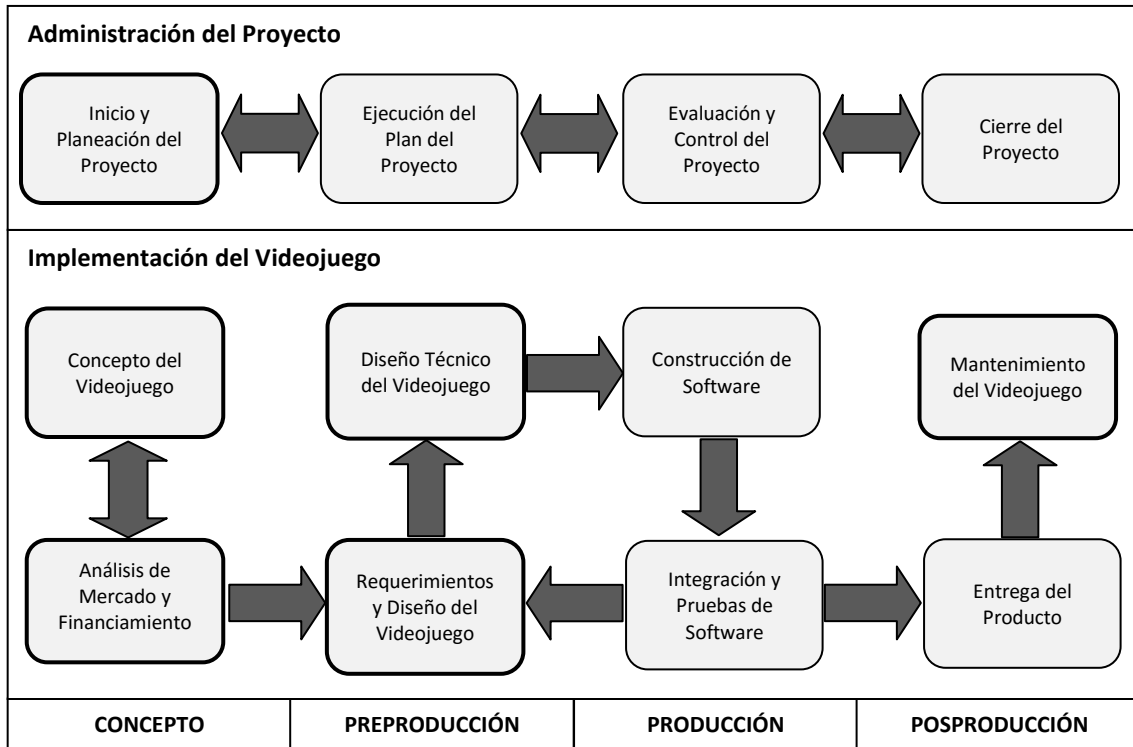


FIGURA B.1 - Procesos y actividades del modelo de procesos

El proceso de *Administración del Proyecto* utiliza el producto de *Enunciado de Trabajo del Cliente* para elaborar el *Plan del Proyecto*. Las tareas de *Evaluación y Control del Proyecto* comparan el progreso del proyecto contra el *Plan del Proyecto* y se toman las acciones necesarias para eliminar desviaciones o incorporar cambios al *Plan del Proyecto*. La actividad de *Cierre del Proyecto* consiste en la entrega de la *Configuración del Software*, producida por la *Implementación del Videojuego*, y se obtiene la aceptación por parte del *Cliente* para formalizar el fin del proyecto. Un *Repositorio del Proyecto* se establece para guardar los productos de trabajo y controlar sus versiones durante el proyecto.

La ejecución del proceso de *Implementación del Videojuego* es dirigida por el *Plan del Proyecto*. El *Plan del Proyecto* guiará la ejecución de las actividades de *Concepto del Videojuego*, *Requerimientos y Diseño del Videojuego*, *Diseño Técnico del Videojuego*, *Construcción del Software*, *Integración y Pruebas de Software*, y *Entrega del Producto*.

Para eliminar los defectos del producto las tareas de verificación, validación y pruebas son incluidas en las actividades del flujo de trabajo.

Roles involucrados

Esta es una lista de los *Roles* involucrados en orden alfabético por área, con sus identificadores y descripción de las competencias sugeridas.

TABLA B.1 - Roles involucrados

ROL	ID	DESCRIPCIÓN
PRODUCCIÓN		
PRODUCTOR ADMINISTRADOR DEL PROYECTO	PRO	<p>Su principal responsabilidad es asegurarse de que el videojuego sea entregado a tiempo, dentro del presupuesto, con todas las características esperadas, en la más alta calidad posible.</p> <p>Debe de tener las siguientes habilidades:</p> <ul style="list-style-type: none"> - Liderazgo - Comunicación efectiva - Organización - Trabajo en equipo - Conocimiento de la industria - Conocimientos de administración de proyectos - Conocimientos de manejo de personal
ARTE		
ANIMADOR	ANM	Se encarga de crear las animaciones y cinemáticas del videojuego. Incluye la animación de objetos, personajes, cámaras, captura de movimiento. Deberá tener conocimientos en animación tradicional y animación digital.
ARTISTA CONCEPTUAL	ACN	<p>Es responsable de crear los conceptos de todos los <i>Assets</i> de arte antes de que sean producidos.</p> <p>Debe de tener una formación o habilidad para ilustrar y realizar bocetos, es recomendable que tenga conocimientos en modelado 3D y en muchos casos se demanda saber crear conceptos en programas de escultura digital. Trabaja directamente con el <i>Artista Líder</i> en la creación y documentación de la visión artística del juego.</p>
ARTISTA DE ENTORNOS 3D	A3D	<p>Se ocupa de la creación de los entornos y sus elementos. Crea los modelos, mapeado, texturas, paisajes, fondos, terrenos, edificios, interiores, mobiliario, armas, vehículos y demás objetos.</p> <p>Debe manejar programas de escultura digital y tener una formación en diseño digital o bellas artes, es recomendable que tenga nociones de ilustración y arquitectura.</p>
ARTISTA DE ILUMINACIÓN	AIL	Se encarga de iluminar los elementos del videojuego tanto estáticos como dinámicos. Debe tener conocimientos de programas de modelado 3D, motores de iluminación y <i>rendering</i> , fotografía y cinemática.
ARTISTA DE INTERFAZ GRÁFICA	AIG	Se encarga de la creación de los elementos de la <i>Interfaz de Usuario</i> . Debe tener conocimientos sobre diseño y evaluación de interfaces, interacción y usabilidad.
ARTISTA DE	AMR	Es responsable de la creación de todos los <i>Assets</i> de mercadotecnia para

MERCADOTECNIA		el juego. Por ejemplo, capturas de pantalla, videos de jugabilidad, arte en alta resolución, y todo lo que necesario para promocionar el juego.
ARTISTA DE PERSONAJES MODELADOR	APR	Se encarga de la creación de los personajes del videojuego. Debe tener dominio de programas de creación 3D y 2D para modelado o ilustración y conocimientos de anatomía.
ARTISTA DE TEXTURAS	ATX	Se encarga casi exclusivamente de la creación de texturas para usar en entornos, objetos o personajes.
ARTISTA LÍDER	ALD	Define la visión artística del juego y se asegura de que se mantenga a través del proceso de desarrollo. Se encarga de la gestión y supervisión del flujo de trabajo, y de garantizar el nivel de calidad de todos los elementos de arte creados. Deberá tener una formación amplia en todos los campos artísticos, conocimientos de otras disciplinas, experiencia en desarrollo y producción de arte.
ARTISTA TÉCNICO	ATC	Se encarga de gestionar los aspectos técnicos relacionados con la producción del arte. Por ejemplo, las áreas de colisión, la exportación correcta de los objetos al <i>Motor de Juego</i> , aplicar atributos físicos a un objeto. Es un vínculo entre el departamento de programación y el de arte, por lo tanto necesita tener conocimiento técnico para comunicarse con programadores.
RIGGER	RIG	Se encarga de preparar a los modelos de personajes y objetos que deben ser animados, de la creación de sistemas de huesos y asignación de pesos para mover a los <i>Modelos</i> . Debe de tener conocimientos de animación tradicional y digital, así como conocimientos de anatomía.
PROGRAMACIÓN		
PROGRAMADOR DE ASEGURAMIENTO DE LA CALIDAD	PAC	Se encarga de rastrear y corregir los errores del videojuego.
PROGRAMADOR DE AUDIO	PAU	Se enfoca en la creación del <i>Motor de Audio</i> para el videojuego. Trabaja cercanamente con el <i>Diseñador de Audio</i> para asegurarse de que el <i>Motor de Audio</i> pueda soportar las características de audio deseadas para el juego.
PROGRAMADOR DE HERRAMIENTAS	PHR	Es responsable de crear las <i>Herramientas (Tools)</i> necesarias durante el desarrollo del videojuego. Estas herramientas incluyen <i>scripts</i> , iluminación, exportadores, herramientas de localización y otras herramientas que puedan ser programadas para coordinar la <i>Línea de Producción (Production Pipeline)</i> del videojuego.
PROGRAMADOR DE INTELIGENCIA ARTIFICIAL	PIA	Crea el código para las rutinas de <i>Inteligencia Artificial</i> . Trabaja cercanamente con el <i>Equipo de Diseño</i> para identificar que conductas y funcionalidad se necesitan para los personajes en el videojuego.
PROGRAMADOR DE INTERFAZ DE USUARIO	PIU	Se encarga de crear el código que permite el funcionamiento de la <i>Interfaz de Usuario y Menús</i> .
PROGRAMADOR DE JUEGO EN LÍNEA	PJL	Se encarga de crear el código que permite el juego en línea (multijugador). Trabaja cercanamente con el <i>Diseñador Líder</i> para asegurarse de que este considerada toda la funcionalidad necesaria del juego en línea.
PROGRAMADOR DE MOTOR DE JUEGO	PMJ	Crea el código del <i>Motor de Juego</i> desde cero o adapta un motor existente.

PROGRAMADOR LÍDER	PLD	<p>Es responsable de administrar las tareas diarias de los <i>Programadores</i>, establecer estándares de codificación y determinar todas las tecnologías que se utilizarán en el desarrollo del videojuego. Trabaja cercanamente con el <i>Artista Líder</i> para determinar qué tecnologías se necesitan para desarrollar el videojuego.</p> <p>Debe tener conocimiento sobre las últimas tecnologías de desarrollo de videojuegos y ser capaz de determinar cómo pueden ser utilizadas en el código del juego. También deberá tener conocimiento general de todas las áreas de tecnología de videojuegos y buenas habilidades de comunicación.</p>
DISEÑO		
CREADOR DEL MANUAL Y TUTORIAL DE JUEGO	CMN	Crea el <i>Manual</i> y los diferentes <i>Tutoriales de Juego</i> que tendrá el videojuego.
DISEÑADOR DE INTERFAZ DE USUARIO	DIU	Diseña la <i>Interfaz de Usuario</i> del videojuego incluyendo como funcionarán las pantallas y cómo se ajustan al videojuego.
DISEÑADOR DE NIVEL WORLD BUILDER	DNV	Es responsable de la creación y diseño de los niveles que tendrá el videojuego, la colocación de los objetos interactivos y enemigos en los niveles. Esencialmente, decide a cuantos enemigos se enfrentará el jugador, donde aparecerán los retos en cada nivel, y con cuántos personajes (no jugables) interactuará el jugador.
DISEÑADOR DE SISTEMAS	DST	Diseña el sistema de componentes dentro de la jugabilidad. Por ejemplo, el sistema de puntuación, el modelo de combate, el esquema de control, el sistema de creación de personajes, el sistema de combos en un juego de peleas, etc.
DISEÑADOR LÍDER	DLD	Es responsable de administrar las tareas diarias de los <i>Diseñadores</i> , dirigir al equipo en la documentación del diseño, prototipos de jugabilidad, implementación de características de diseño, balanceo de la jugabilidad y rediseñar algunas características cuando sea necesario.
ESCRITOR	ESC	<p>Se encarga de crear el argumento principal del videojuego y la historia. Puede enfocarse en los diálogos, creación de personajes y el desarrollo del mundo del juego.</p> <p>Durante la fase conceptual se encarga de realizar una investigación que ayude a crear la historia del videojuego (datos históricos, personajes, lugares, edificios, leyendas, etc.), cómo debe contarse y qué tan profunda debe de ser.</p>
GUIONISTA	GUI	Se encarga de crear el <i>Guión Literario</i> , los diálogos para todas las partes del juego, incluyendo cinemáticas y jugabilidad. Las voces de los personajes son grabadas después utilizando el <i>Guión Literario</i> .
AUDIO		
COMPOSITOR	COM	Se encarga de crear la música de acuerdo a las especificaciones del videojuego. La música puede ser creada con software o grabada por músicos en algún estudio. Trabaja cercanamente con el <i>Productor</i> del videojuego y los <i>Diseñadores de Audio</i> .
DISEÑADOR DE AUDIO	DAU	Contribuye en todos los aspectos del contenido de audio del videojuego. Esto puede incluir el diseño de los efectos de sonido, la grabación y procesamiento de diálogos y edición de música. También puede contribuir en la implementación de <i>Assets</i> de audio.

ASEGURAMIENTO DE LA CALIDAD		
ASEGURADOR DE LA CALIDAD TESTER	ACD	Es responsable de revisar la funcionalidad del videojuego de acuerdo a los <i>Escenarios y Casos de Pruebas</i> , probar nuevas características y <i>Prototipos</i> , y encontrar defectos en el software del videojuego. Además, se encarga de que el videojuego cumpla con los requerimientos técnicos de la plataforma en la que será publicado.
ASEGURADOR DE LA CALIDAD LÍDER	ACL	Es responsable de escribir los diferentes <i>Escenarios y Casos de Pruebas</i> del videojuego. Deberá conocer exactamente cada parte de las funciones del juego, de tal forma que esos detalles sean incluidos en los <i>Escenarios y Casos de Pruebas</i> . También se encarga de administrar las tareas del equipo y asignarles áreas específicas para su revisión. Trabaja en conjunto con el <i>Productor</i> y otros líderes para evaluar las características del videojuego y determinar cuándo el juego estará listo para su liberación.
MERCADOTECNIA		
MERCADÓLOGO	MER	Se encarga de crear una estrategia adecuada para promocionar el videojuego durante el proceso de producción. Deberá tener conocimiento de estudios y tendencias de la industria de los videojuegos, y capacidad para manejo de elementos publicitarios en el desarrollo de videojuegos.
FINANCIAMIENTO		
CLIENTE	CLI	Conocimiento y experiencia en el dominio del negocio. Capacidad para expresar los requerimientos de los usuarios.
PUBLISHER	PUB	Son grandes inversionistas y distribuidores de videojuegos. Se encargan de rentabilizar y hacer llegar al mercado los productos de los desarrolladores. Funcionan de manera similar a un sello discográfico. Por lo general los <i>Estudios Independientes</i> contactan con un <i>Publisher</i> para mostrarle su videojuego casi terminado o al menos un <i>Prototipo del Videojuego</i> , si logran captar su atención entraran en una fase de negociación donde el <i>Publisher</i> invierte en el proyecto para su finalización y comercialización. Plantean un <i>Modelo de Monetización</i> y una estrategia de mercadotecnia adecuada y distribuyen el juego en los países que a su criterio tendrán mejor acogida.
EQUIPOS		
EQUIPO DE TRABAJO	EQT	Conocimiento y experiencia de acuerdo a sus roles dentro del proyecto.
LÍDERES DE ÁREA	LAR	Líderes de cada área de acuerdo a la <i>Organización del Equipo</i> .
DISEÑADORES	EDS	Equipo de diseñadores que tiene el estudio.
PROGRAMADORES	EPR	Equipo de programadores que tiene el estudio.
ARTISTAS	EAR	Equipo de artistas que tiene el estudio.

Organización del equipo

Se puede organizar a un equipo de diferentes formas, dependiendo del tamaño del *Estudio Independiente* y los *Roles*. En estudios pequeños es común que una sola persona tenga asignado múltiples *Roles* en un proyecto, por ejemplo, *Productor-Diseñador Líder*, en estudios más grandes se podría tener a una persona con un rol único y bien definido como *Programador de Inteligencia Artificial* o *Artista Conceptual*.

No importa cuál sea el tamaño del equipo, debe establecerse una organización clara de tal forma que la gente sepa con quien tiene que comunicarse.

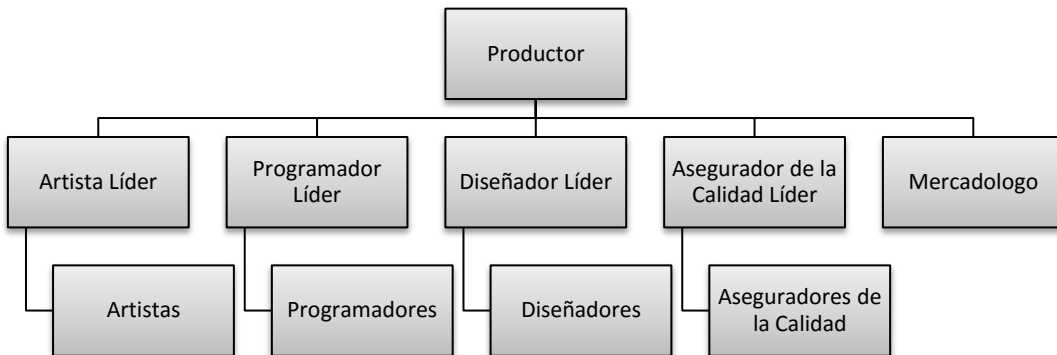


FIGURA B.2 - Organización jerárquica de un estudio independiente pequeño

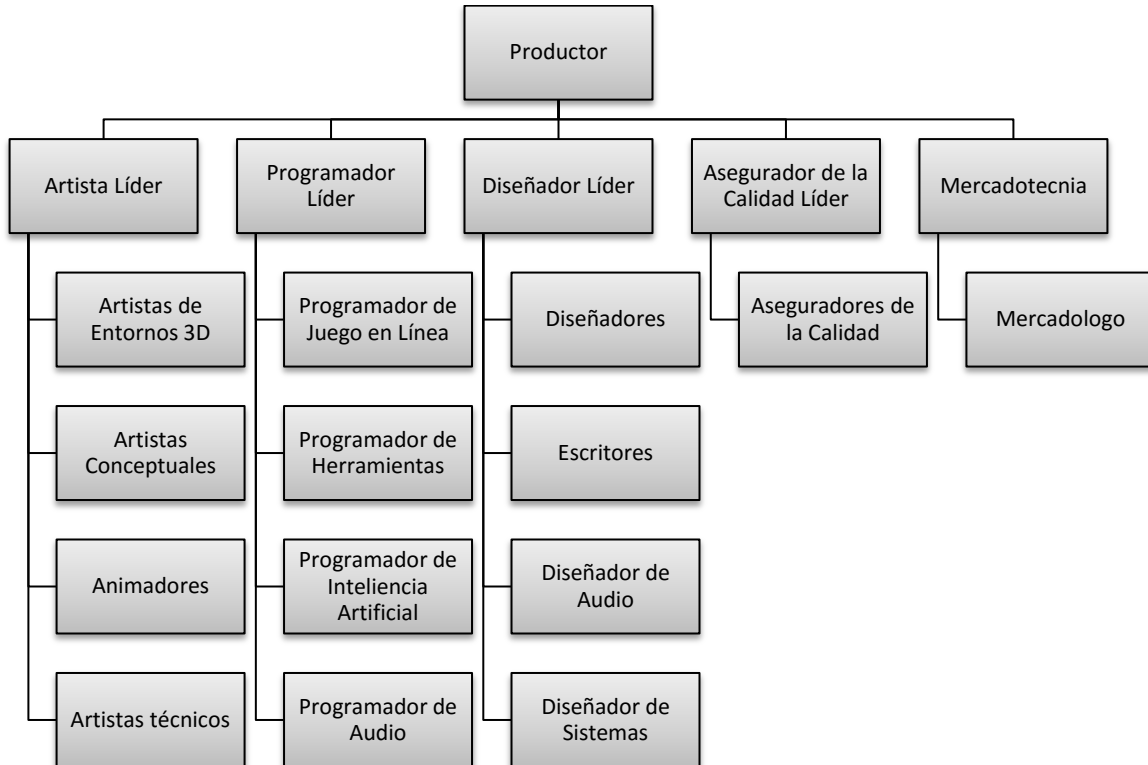


FIGURA B.3 - Organización jerárquica de un estudio independiente mediano

PROCESO DE ADMINISTRACIÓN DEL PROYECTO (AP)

AP - Propósito

El propósito del proceso de *Administración del Proyecto (AP)* es establecer y llevar a cabo de manera sistemática las *Tareas* de un proyecto de *Implementación de Videojuego (IV)*, que permitan cumplir con los objetivos del proyecto en la calidad, tiempo y costos esperados.

Este documento está destinado para ser utilizado por los *Estudios Independientes* para establecer procesos para implementar cualquier enfoque o metodología de desarrollo, incluyendo por ejemplo, ágil, evolutivo, incremental, desarrollo dirigido por pruebas, etc. basada en las necesidades del estudio o del proyecto.

AP - Actividades

El proceso de *Administración del Proyecto* consiste en las siguientes actividades:

TABLA B.2 - Actividades del Proceso de Administración del Proyecto

ID	ACTIVIDAD
AP.1	INICIO
AP.2	PLANEACIÓN DEL PROYECTO
AP.3	EJECUCIÓN DEL PLAN DEL PROYECTO
AP.4	EVALUACIÓN Y CONTROL DEL PROYECTO
AP.5	CIERRE DEL PROYECTO

AP - Objetivos

- El *Plan del Proyecto* para la ejecución del proyecto es desarrollado de acuerdo al *Enunciado de Trabajo* y revisado por el *Cliente*. Las *Tareas* y los *Recursos* necesarios para completar el trabajo son dimensionados y estimados.
- El avance del proyecto es monitoreado contra el *Plan del Proyecto* y registrados en el *Reporte de Avance*. Las correcciones para resolver los problemas y desviaciones respecto del plan son realizadas cuando los objetivos del proyecto no son logrados. El cierre del proyecto es ejecutado para conseguir la aceptación documentada del *Cliente* en el *Documento de Aceptación*.
- Las *Solicitudes de Cambio* son atendidas mediante su recepción y análisis. Los cambios a los requerimientos son evaluados por su impacto técnico, en costo y calendario.
- Reuniones de revisión con el *Equipo de Trabajo* y el *Cliente* son realizadas. Los acuerdos que surgen de estas reuniones son documentadas y se les da seguimiento.

- Los riesgos son identificados en el desarrollo y durante la realización del proyecto.
- Una *Estrategia de Control de Versiones* es desarrollada. Los elementos de *Configuración del Software* son identificados, definidos e incorporados a la línea base.
- El *Aseguramiento de la Calidad del Software* es realizado para proporcionar garantía de que los productos y procesos de trabajo cumplan con el *Plan del Proyecto* y *Especificación de Requerimientos*.

La FIGURA B.4 muestra el diagrama del flujo de información entre las actividades del proceso de *Administración del Proyecto* y sus productos de trabajo más relevantes.

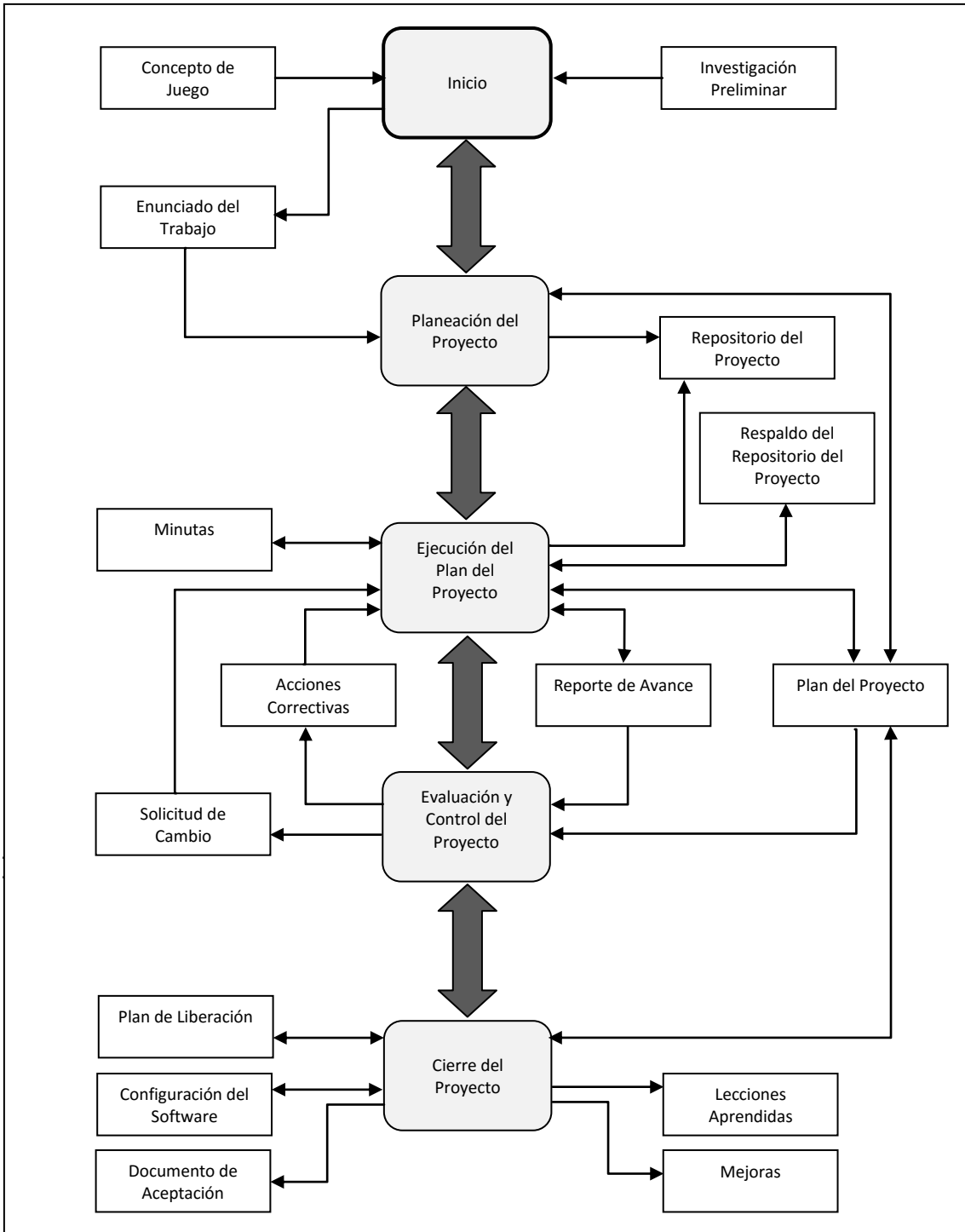


FIGURA B.4 - Diagrama del proceso de Administración de Proyecto

AP.1 - Inicio

En la actividad *Inicio* se realiza el *Enunciado del Trabajo* del proyecto, con base en la *Investigación Preliminar*, el *Documento de Concepto de Juego* y el *Análisis Competitivo*. Finalmente es revisado por el *Equipo de Trabajo* y se obtiene la aprobación por parte del *Cliente*.

TABLA B.3 - Tareas para la actividad Inicio y Planeación del Proyecto

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
INICIO			
PRO LAR MER	<p>Realizar el <i>Enunciado del Trabajo</i> con base en lo documentado en la <i>Investigación Preliminar</i>, el <i>Documento de Concepto de Juego</i> y <i>Análisis Competitivo</i>.</p> <ul style="list-style-type: none"> - Definir los objetivos del videojuego. - Determinar el alcance que tendrá el videojuego. - Identificar los <i>Stakeholders</i> que intervendrán el desarrollo del videojuego. - Analizar si el desarrollo del videojuego es factible. 	<p><i>Documento de Concepto de Juego</i></p> <p><i>Investigación Preliminar</i></p> <p><i>Análisis Competitivo</i></p>	<i>Enunciado de Trabajo</i>
EQT	Revisar dentro del <i>Equipo de Trabajo</i> y obtener la aprobación del <i>Enunciado del Trabajo</i> .	<i>Enunciado del Trabajo</i>	<i>Enunciado del Trabajo [aprobado]</i>
CLI PRO LAR MER	<p>Revisar y obtener la aprobación del <i>Enunciado del Trabajo</i>.</p> <p>El <i>Cliente</i> revisa y valida (acepta) el <i>Enunciado del Trabajo</i>.</p>	<i>Enunciado del Trabajo [aprobado]</i>	<i>Enunciado del Trabajo [aceptado]</i>

TABLA B.4 - Productos de Trabajo de la actividad Inicio

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Documento de Concepto de Juego	Concepto del Videojuego
Investigación Preliminar	Análisis de Mercado y Financiamiento
Análisis Competitivo	Análisis de Mercado y Financiamiento
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Enunciado del Trabajo [aceptado]	Planeación del Proyecto
PRODUCTOS INTERNOS	
NOMBRE	
Enunciado del Trabajo [aprobado]	

AP.2 - Planeación del Proyecto

La actividad de *Planeación del Proyecto* documenta los detalles de la planeación necesarios para administrar el proyecto.

TABLA B.5 - Tareas para la actividad Planeación del Proyecto

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
PLANEACIÓN DEL PROYECTO			
PRO CLI	<p>Definir los <i>Protocolos de Entrega</i> e incorporarlo al <i>Plan del Proyecto</i>.</p> <ul style="list-style-type: none"> - Definir dinámica de reuniones y revisiones. - Establecer listas de validación. - Establecer solicitudes de cambio. - Establecer responsables de validación. 	<i>Enunciado del Trabajo [aceptado]</i>	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Protocolos de Entrega</i>
PRO LAR	<p>Definir los <i>Protocolos de Verificación</i> que se emplearán dentro del <i>Equipo de Trabajo</i> e incorporarlo al <i>Plan del Proyecto</i>.</p> <ul style="list-style-type: none"> - Definir dinámica de reuniones y verificaciones. - Establecer listas de verificación. - Establecer solicitudes de cambio 	<i>Enunciado del Trabajo [aceptado]</i>	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Protocolos de Verificación</i>
PRO LAR	Identificar las <i>Tareas</i> específicas a realizar para producir elementos <i>Entregables</i> del videojuego (documentos, <i>Assets</i> , <i>Game Builds</i>).	<i>Enunciado del Trabajo [aceptado]</i>	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Tareas</i>
PRO LAR	Establecer la <i>Duración Estimada</i> para realizar cada <i>Tarea</i> .	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Tareas</i> 	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Duración Estimada</i>
PRO LAR	<p>Identificar y documentar los <i>Recursos</i>: humanos, materiales, equipo y herramientas, estándares, incluyendo la capacitación requerida por el <i>Equipo de Trabajo</i>.</p> <p>Incluir las fechas en el calendario cuando sean requeridos los <i>Recursos</i>. Incluir las fechas de capacitación (si es requerido).</p>	<i>Enunciado del Trabajo [aceptado]</i>	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Recursos</i> - <i>Plan de Contratación</i>
PRO LAR	Establecer la composición del <i>Equipo de Trabajo</i> , asignando los roles y responsabilidades acordes a los <i>Recursos</i> .	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Recursos</i> 	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Composición del Equipo de Trabajo</i>

PRO LAR	Asignar las fechas de inicio y fin estimadas para cada <i>Tarea</i> para generar el <i>Calendario de Tareas</i> considerando los <i>Recursos</i> asignados, la prioridad, secuencia y dependencia de las <i>Tareas</i> .	<i>Plan del Proyecto</i> - <i>Tareas</i> - <i>Duración Estimada</i> - <i>Composición del Equipo de Trabajo</i>	<i>Plan del Proyecto</i> - <i>Calendario de Tareas</i>
PRO LAR	Calcular y documentar el <i>Esfuerzo</i> y <i>Costo</i> y <i>Calendario de Flujo de Fondos</i> estimado del proyecto.	<i>Plan del Proyecto</i> - <i>Calendario de Tareas</i> - <i>Recursos</i>	<i>Plan del Proyecto</i> - <i>Esfuerzo y Costo</i> estimado - <i>Calendario de Flujo de Fondos</i>
PRO LAR	Identificar y documentar los <i>Riesgos</i> que pueden afectar el proyecto.	<i>Plan del Proyecto</i> - <i>Todos los elementos previamente definidos</i>	<i>Plan del Proyecto</i> - <i>Identificación de Riesgos</i>
PRO LAR	Definir y documentar la <i>Estrategia de Control de Versiones</i> . Definir <i>Control de Versiones</i> para la documentación y establecer el repositorio.	<i>Plan del Proyecto</i> - <i>Todos los elementos previamente definidos</i>	<i>Plan del Proyecto</i> - <i>Estrategia de Control de Versiones</i>
PRO LAR	Incluir la descripción del producto, el alcance, los objetivos y los entregables en el <i>Plan del Proyecto</i> .	<i>Enunciado de Trabajo</i> - <i>Descripción del producto</i> - <i>Alcance</i> - <i>Objetivos</i> - <i>Entregable</i>	<i>Plan del Proyecto</i> - <i>Descripción del producto</i> - <i>Alcance</i> - <i>Objetivos</i> - <i>Entregables</i>
PRO LAR	Generar el <i>Plan del Proyecto</i> integrando los elementos previamente identificados y documentados.	<i>Plan del Proyecto</i> - <i>Todos los elementos previamente definidos</i>	<i>Plan del Proyecto</i> - <i>Todos los elementos previamente definidos</i>
PRO LAR CLI	Verificar y obtener la aprobación del <i>Plan del Proyecto</i> . Verificar la viabilidad y consistencia de todos los elementos.	<i>Plan del Proyecto</i>	<i>Plan del Proyecto</i> [verificado]
PRO CLI	Revisar y obtener la aprobación del <i>Plan del Proyecto</i> .	<i>Plan del Proyecto</i> [verificado]	<i>Plan del Proyecto</i> [aceptado]
PRO PLD	Establecer el <i>Repositorio del Proyecto</i> usando la <i>Estrategia de Control de Versiones</i>	<i>Estrategia de Control de Versiones</i>	<i>Repositorio del Proyecto</i>

TABLA B.6 - Productos de Trabajo de la actividad Planeación del Proyecto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Enunciado del Trabajo	Inicio
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Plan del Proyecto	Administración del Proyecto (AP) Implementación del Videojuego (IV)
PRODUCTOS INTERNOS	
NOMBRE	
Protocolos de Entrega	
Protocolos de Verificación	
Tareas	
Duración Estimada	
Recursos	
Composición del Equipo de Trabajo	
Calendario de Tareas	
Esfuerzo y Costo estimado	
Calendario de Flujo de Fondos	
Plan de Contratación	
Identificación de Riesgos	
Estrategia de Control de Versiones	
Descripción del producto	
Alcance	
Objetivos	
Entregables	
Repositorio del Proyecto	

AP.3 - Ejecución del Plan del Proyecto

En la actividad *Ejecución del Plan del Proyecto* se definen tareas para implementar el plan documentado en el Proyecto.

TABLA B.7 - Tareas para la actividad Ejecución del Plan del Proyecto

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
EJECUCIÓN DEL PLAN DEL PROYECTO			
PRO	Llevar a cabo el <i>Plan del Proyecto</i> según las <i>Tareas</i> establecidas y priorizadas. Monitorear <i>Tareas</i> según lo establecido en el <i>Calendario de Tareas del Proyecto</i> .	<i>Plan del Proyecto [aceptado]</i>	<i>Reporte de Avance [actualizado]</i>

PRO LAR	Analizar y evaluar el impacto en costo, tiempo y tecnológico de la <i>Solicitud de Cambio</i> que es propuesta por el <i>Equipo de Trabajo</i> o <i>Cliente</i> . Si se afectan acuerdos previos necesita ser negociada por ambas partes.	<i>Solicitud de Cambio</i> [propuesta] <i>Plan del Proyecto</i> [aceptado]	<i>Solicitud de Cambio</i> [evaluada] <i>Plan del Proyecto</i> [actualizado]
PRO LAR	Dirigir reuniones de revisión con el <i>Equipo de Trabajo</i> para identificar problemas, revisar riesgos, registrar acuerdos, nuevas ideas para el videojuego y darles seguimiento hasta su conclusión.	<i>Plan del Proyecto</i> [aceptado] <i>Reporte de Avance</i> [actualizado] <i>Acciones Correctivas</i> <i>Minutas</i>	<i>Minuta</i> [actualizada]
PRO CLI PLD	Realizar reuniones frecuentes con el <i>Cliente</i> para obtener validaciones, atender <i>Solicitud de Cambio</i> y registrar nuevos acuerdos.	<i>Plan del Proyecto</i> [aceptado] <i>Reporte de Avance</i> [actualizado] <i>Solicitud de Cambio</i> [evaluada] <i>Minuta</i>	<i>Minuta</i> [actualizada] <i>Solicitud de Cambio</i> [aceptada] <i>Plan del Proyecto</i> [actualizado] <i>Retroalimentación del Hito</i>
PRO	Realizar el <i>Respaldo del Repositorio del Proyecto</i> de acuerdo a la <i>Estrategia de Control de Versiones</i>	<i>Estrategia de Control de Versiones</i>	<i>Respaldo del Repositorio del Proyecto</i>

TABLA B.8 - Productos de Trabajo de la actividad Ejecución del Plan del Proyecto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [aceptado]	Planeación del Proyecto
Solicitud de Cambio [propuesta]	Evaluación y Control del Proyecto
Acciones Correctivas	Evaluación y Control del Proyecto
Estrategia de Control de Versiones	Planeación del Proyecto
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Reporte de Avance [actualizado]	Evaluación y Control del Proyecto
Solicitud de Cambio [evaluada]	Evaluación y Control del Proyecto
Plan del Proyecto [actualizado]	Administración del Proyecto (IV) Implementación del Videojuego (IV)
Retroalimentación del Hito	Evaluación y Control del Proyecto
PRODUCTOS INTERNOS	
NOMBRE	
Minutas	
Respaldo del Repositorio del Proyecto	

AP.4 - Evaluación y Control del Proyecto

La actividad *Evaluación y Control del Proyecto* evalúa el desempeño del plan contra los compromisos documentados.

TABLA B.9 - Tareas para la actividad Evaluación y Control del Proyecto

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
EVALUACIÓN Y CONTROL DEL PROYECTO			
PRO	Evaluar el progreso del proyecto con respecto al <i>Plan del Proyecto</i> comparando: <ul style="list-style-type: none"> - <i>Tareas</i> reales contra tareas planeadas - Resultados reales contra objetivos establecidos en el proyecto - <i>Costos</i> reales contra el presupuesto estimado - <i>Tiempo</i> utilizado contra el tiempo programado - <i>Riesgo</i> real contra el identificado previamente 	<i>Plan del Proyecto [aceptado]</i> <i>Reporte de Avance [actualizado]</i>	<i>Reporte de Avance [evaluado]</i> <i>Documento de Seguimiento de Assets</i>
PRO LAR	Establecer <i>Acciones Correctivas</i> e identificar nuevos riesgos.	<i>Reporte de Avance [evaluado]</i>	<i>Acciones Correctivas</i>
PRO	Identificar cambios a requerimientos al <i>Plan del Proyecto</i> para hacer frente a desviaciones importantes, potenciales riesgos o problemas relativos al cumplimiento del plan. Proponer <i>Solicitud de Cambio</i> y dar seguimiento hasta su conclusión.	<i>Reporte de Avance [evaluado]</i>	<i>Solicitud de Cambio [propuesta]</i>

TABLA B.10 - Productos de Trabajo de la actividad Evaluación y Control del Proyecto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [aceptado]	Planeación del Proyecto
Reporte de Avance [actualizado]	Ejecución del Plan del Proyecto
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Acciones Correctivas	Ejecución del Plan del Proyecto
Solicitud de Cambio [propuesta]	Ejecución del Plan del Proyecto
PRODUCTOS INTERNOS	
NOMBRE	
Reporte de Avance [evaluado]	
Documento de Seguimiento de Assets	

AP.5 - Cierre del Proyecto

La Actividad de *Cierre del Proyecto* proporciona documentación y productos del proyecto de acuerdo con los *Protocolos de Entrega*.

TABLA B.11 - Tareas para la actividad Cierre del Proyecto

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
CIERRE DEL PROYECTO			
PRO MER EQT CLI	<p>Realizar el <i>Plan de Liberación</i> de la versión final del <i>Software (Gold Master Build)</i>.</p> <ul style="list-style-type: none"> - Definir publicación del videojuego - Definir elementos publicitarios - Costo final del videojuego - Definir los medios de distribución - Definir fechas aproximadas de publicación 	<p><i>Plan del Proyecto</i></p> <p><i>Software (Gold Master Build) [validado]</i></p>	<i>Plan de Liberación</i>
MER LAR	<p>Llevar a cabo la publicación del videojuego de acuerdo al <i>Plan de Liberación</i>.</p> <p>Realizar actividades de mercadotecnia y promoción del videojuego.</p> <p>Publicar el videojuego de acuerdo a los requerimientos de las <i>Políticas de Publicación</i> en Mercados.</p>	<p><i>Plan de Liberación</i></p> <p><i>Políticas de Publicación en Mercados</i></p> <p><i>Software (Gold Master Build) [validado]</i></p>	<i>Software (Gold Master Build) [disponible]</i>
PRO CLI	<p>Formalizar conclusión del proyecto de acuerdo a los <i>Protocolos de Entrega</i>. Incluir en el <i>Repositorio del Proyecto</i>.</p>	<p><i>Plan del Proyecto</i></p> <ul style="list-style-type: none"> - <i>Protocolos de Entrega</i> <p><i>Configuración del Software [entregada]</i></p>	<p><i>Documento de Aceptación</i></p> <p><i>Configuración del Software [aceptada]</i></p>
EQT	<p>Realizar una autoevaluación en el <i>Grupo de Trabajo</i> para obtener retroalimentación y documentar las <i>Lecciones Aprendidas</i> durante el desarrollo del proyecto.</p>		<i>Lecciones Aprendidas</i>
EQT	<p>Establecer posibles procesos, tareas, actividades o mecanismos de mejora para futuros proyectos.</p>	<i>Lecciones Aprendidas</i>	<i>Mejoras</i>

TABLA B.12 - Productos de Trabajo de la actividad Cierre del Proyecto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [aceptado]	Planeación del Proyecto
Protocolos de Entrega	Planeación del Proyecto
Software (Gold Master Build) [validado]	Integración y Pruebas de Software
Políticas de Publicación en Mercados	Análisis de Mercado
Configuración del Software [entregada]	Entrega del Producto
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Software (Gold Master Build) [disponible]	Tienda de Aplicaciones
Configuración del Software [aceptada]	Cliente
Lecciones Aprendidas	Equipo de Trabajo
Mejoras	Equipo de Trabajo
PRODUCTOS INTERNOS	
NOMBRE	
Plan de Liberación	

PROCESO DE IMPLEMENTACIÓN DEL VIDEOJUEGO (IV)

IV - Propósito

El propósito del proceso *de Implementación del Videojuego* es la realización sistemática de las actividades de concepto, diseño, construcción, integración y pruebas para productos de software, nuevos o modificados, de acuerdo a los requerimientos especificados.

IV - Fases y Actividades

Para el proceso *de Implementación del Videojuego* se proponen las siguientes fases y actividades:

TABLA B.13 - Fases y Actividades del Proceso de Implementación del Videojuego

FASE	ID	ACTIVIDAD
1. CONCEPTO	IV.1	Concepto del Videojuego
	IV.2	Análisis de Mercado y Financiamiento
2. PREPRODUCCIÓN	IV.3	Requerimientos y Diseño del Videojuego
	IV.4	Diseño Técnico del Videojuego
3. PRODUCCIÓN	IV.5	Construcción de Software
	IV.6	Integración y Pruebas de Software
4. POSPRODUCCIÓN	IV.7	Entrega del Producto
	IV.8	Mantenimiento del Videojuego

La FIGURA B.5 muestra el diagrama del flujo de información entre las actividades del proceso de *Implementación del Videojuego* y sus productos de trabajo más relevantes.

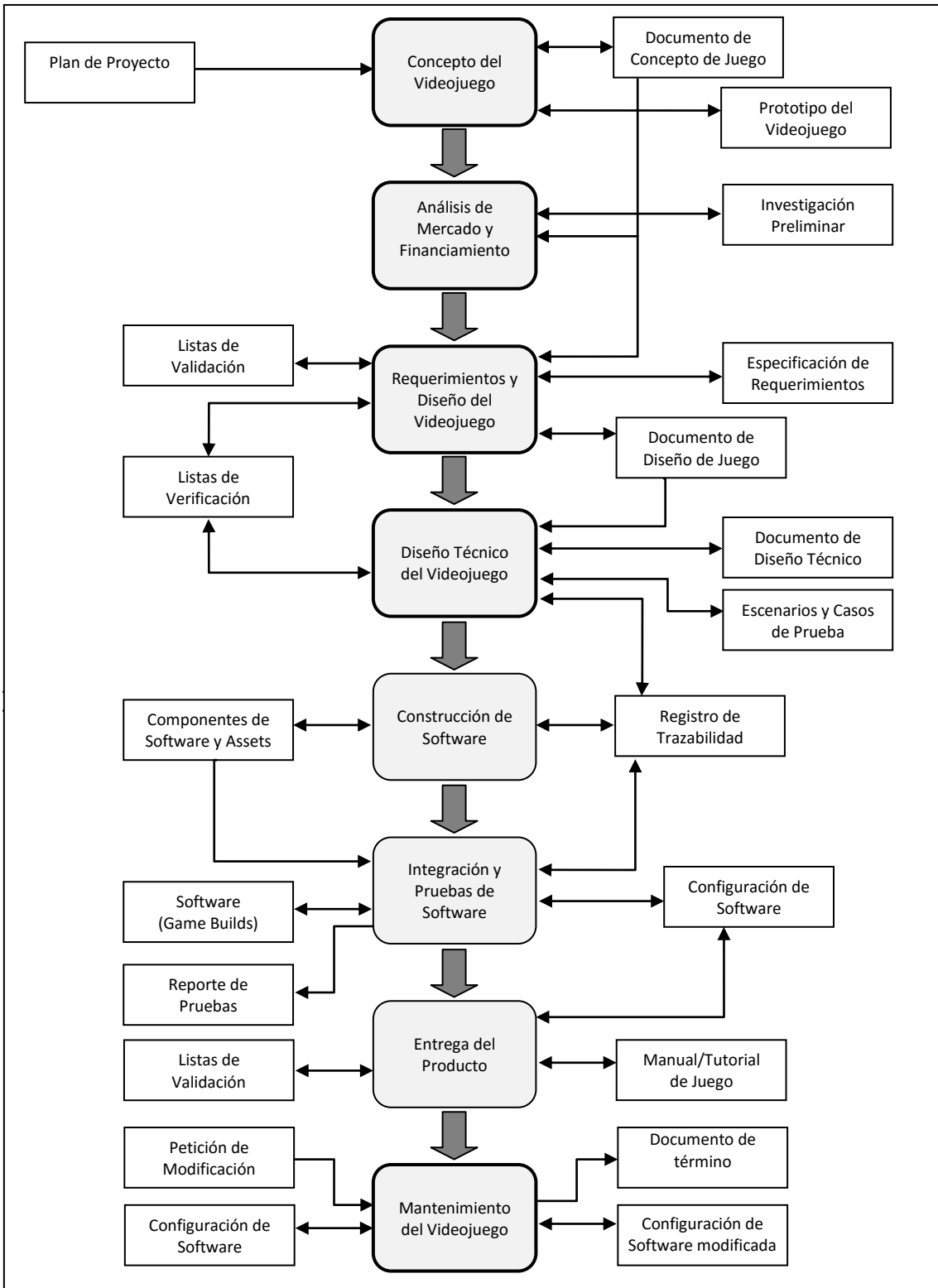


FIGURA B.5 - Diagrama del proceso de Implementación del Videojuego

IV. Fase de Concepto

En la fase de *Concepto* se llevará a cabo la concepción de la idea general del videojuego, identificando los elementos fundamentales que lo caracterizarán, finalizando en un diseño conceptual del mismo y en un *Prototipo*. También en esta fase se propone llevar a cabo un *Análisis de Mercado* para evaluar los videojuegos similares que estén publicados y probar que hay un mercado potencial para el videojuego que se quiere construir. Finalmente, para que el proyecto pueda llevarse a cabo, se considerará un *Modelo Financiamiento* con el objetivo de que alguien apruebe e invierta en la idea que será desarrollada en esta fase.

IV.1 - Concepto del Videojuego

En la actividad *Concepto del Videojuego* se crea el diseño conceptual del videojuego identificando los elementos fundamentales que lo caracterizarán (*Género, Plataforma, Resumen de la Historia, Mecánicas Principales, Jugabilidad, Ambientación, Arte Conceptual y Elementos de Audio*), finalizando en la construcción de un *Prototipo* con base en el diseño conceptual.

TABLA B.14 - Lista de tareas para la actividad Concepto del Videojuego

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
CONCEPTO DEL VIDEOJUEGO			
PRO DLD MER	Realizar <i>Propuesta de Ideas</i> para el desarrollo de un nuevo videojuego.	<i>Planteamiento del Videojuego</i>	<i>Generación de Idea</i>
PRO DLD MER	Evaluar <i>Generación de Idea</i> desarrolladas para el desarrollo del videojuego.	<i>Generación de Idea</i>	<i>Generación de Idea [evaluadas]</i>
PRO DLD MER	Crear <i>Concepto Inicial</i> . Un concepto no detallado del juego que presenta la <i>Idea general</i> y <i>Objetivo Principal</i> del videojuego. Proporciona las bases para todas las decisiones del juego.	<i>Generación de Ideas [evaluadas]</i>	<i>Concepto Inicial</i> - <i>Idea General</i> - <i>Objetivo Principal</i>
PRO DLD MER	Identificar la <i>Motivación</i> . Identificar las razones por las cuales la gente jugará el videojuego. - Búsqueda - Aprendizaje - Administración de Tareas - Reto - Competencia - Emoción - Otra	<i>Concepto Inicial</i> - <i>Idea General</i> - <i>Objetivo Principal</i>	<i>Concepto Inicial</i> - <i>Motivación</i>
PRO DLD MER	Identificar el <i>Público Objetivo</i> . Identificar el grupo de personas para los cuales el videojuego está destinado.	<i>Concepto Inicial</i> - <i>Idea General</i> - <i>Objetivo Principal</i> - <i>Motivación</i>	<i>Concepto Inicial</i> - <i>Público Objetivo</i>

PRO DLD MER	Identificar el <i>Género</i> del videojuego basado en la <i>Motivación</i> y en el <i>Público Objetivo</i> del juego. <ul style="list-style-type: none"> - Acción - Aventura - Deportes - Simulación - Juegos de Rol - Cartas - Fantasía - Otro 	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Motivación</i> - <i>Público Objetivo</i> 	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Género</i>
PRO DLD MER	Elegir <i>Plataforma</i> de juego. La <i>Plataforma</i> se refiere al hardware utilizado para jugar el videojuego. <ul style="list-style-type: none"> - Consola - PC - Arcade - Móviles - Otra <p>Investigar <i>Estándares de Plataforma</i>.</p>	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Idea General</i> - <i>Objetivo Principal</i> - <i>Motivación</i> - <i>Público Objetivo</i> - <i>Género</i> 	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Plataforma</i> <p><i>Estándares de Plataforma</i></p>
DLD	Determinar el <i>Estatus del Juego</i> . Determinar si el juego se podrá jugar de modo local y/o en línea. <ul style="list-style-type: none"> - Online - Offline 	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Todos los elementos anteriormente descritos</i> 	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Estatus del Juego</i>
DLD	Determinar <i>Número de Jugadores</i> . <ul style="list-style-type: none"> - Un solo jugador - Multijugador 	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Todos los elementos anteriormente descritos</i> 	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Número de jugadores</i>
PRO DLD	Generar el <i>Concepto Inicial</i> integrando los elementos previamente identificados y documentados.	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Todos los elementos anteriormente descritos</i> 	<i>Concepto Inicial</i>
PRO MER DLD	Realizar <i>Análisis Competitivo</i> . Identificar las fortalezas y debilidades de la competencia del videojuego.	<i>Concepto Inicial</i> <ul style="list-style-type: none"> - <i>Todos los elementos anteriormente descritos</i> 	<i>Análisis Competitivo</i>
PRO MER DLD	Aprobar <i>Concepto Inicial</i> .	<i>Concepto Inicial</i> <i>Análisis Competitivo</i>	<i>Concepto Inicial [aprobado]</i>

PRO	<p>Crear <i>Documento de Concepto de Juego</i>. Escribir la <i>Declaración de Misión</i> del juego, considerando:</p> <ul style="list-style-type: none"> - <i>Objetivo Principal</i> del juego. - <i>Retos Involucrados</i> en la consecución del <i>Objetivo Principal</i>. - <i>Recursos Disponibles</i> que le ayudaran al jugador a conseguir ese objetivo. 	<p><i>Concepto Inicial [aprobado]</i></p>	<p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Declaración de Misión</i>
DLD ALD	<p>Crear o actualizar los <i>Archivos de Referencia</i> que servirán para crear los diferentes elementos del videojuego (fotografías, videos, música, etc).</p>	<p><i>Concepto Inicial [aprobado]</i></p>	<p><i>Archivos de Referencia</i></p>
DLD ESC ALD	<p>Definir la <i>Ambientación de Juego</i>. Definir los escenarios y ambientación de acuerdo al <i>Concepto Inicial</i>.</p> <ul style="list-style-type: none"> - Ciencia ficción - Mundo real - Fantasía - Histórico - Otro 	<p><i>Concepto Inicial [aprobado]</i></p> <p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Declaración de Misión</i> <p><i>Archivos de Referencia</i></p>	<p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Ambientación del Juego</i>
DLD ESC	<p>Crear el <i>Resumen de la Historia</i>. Crear el argumento fundamental que define el flujo del juego de inicio a fin. El resumen integra la ambientación, jugabilidad, mecánicas y los personajes.</p>	<p><i>Concepto Inicial [aprobado]</i></p> <p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Ambientación del Juego</i> <p><i>Archivos de Referencia</i></p>	<p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Resumen de la Historia</i>
PRO MER DLD	<p>Proponer <i>Modelo de Monetización</i> del videojuego. Se refiere a la forma en la que el videojuego va a generar dinero.</p> <ul style="list-style-type: none"> - <i>Anuncios in-game</i> - <i>Modelo basado en suscripción</i> - <i>Modelo Free-to-play</i> - <i>Otro</i> 	<p><i>Concepto Inicial [aprobado]</i></p> <p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Todos los elementos anteriormente descritos</i> 	<p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Modelo de Monetización</i>

DLD ALD ESC ACN COM	<p>Conceptualizar la <i>Jugabilidad (Gameplay)</i>. Describir la forma de interactuar con el juego, aquello que experimentará el jugador durante el transcurso del juego.</p> <ul style="list-style-type: none"> - Retos para el jugador - Recompensas del jugador - Curva de aprendizaje - Esquema del control - Acciones del jugador - Elementos del multijugador - Otros 	<p><i>Concepto Inicial [aprobado]</i></p> <p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Todos los elementos previamente descritos</i> 	<p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Jugabilidad (Gameplay)</i>
DLD	<p>Conceptualizar las <i>Mecánicas de Juego</i>.</p> <ul style="list-style-type: none"> - Objetos o elementos - Comportamientos - Reglas 	<p><i>Concepto Inicial [aprobado]</i></p> <p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Jugabilidad (Gameplay)</i> 	<p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Mecánicas de Juego</i>
DLD ALD ACN	<p>Crear el <i>Arte Conceptual</i> del juego. El arte conceptual muestra cuál será la apariencia de los elementos visuales del videojuego antes de que sean producidos para el juego.</p> <p>Crear bocetos de los objetos, personajes y escenarios donde se llevará a cabo la acción del juego.</p>	<p><i>Concepto Inicial [aprobado]</i></p> <p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Resumen de la Historia</i> - <i>Ambientación del Juego</i> <p><i>Archivos de Referencia</i></p>	<p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Arte Conceptual</i>
DLD DAU COM PAU	<p>Crear el plan inicial de <i>Elementos de Audio</i> del videojuego. Considerar narración, efectos de sonido, música de acuerdo a la ambientación, historia, jugabilidad que podría incluir.</p>	<p><i>Concepto Inicial</i></p> <p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Ambientación del Juego</i> - <i>Resumen de la Historia</i> - <i>Arte Conceptual</i> - <i>Jugabilidad</i> - <i>Mecánicas de Juego Principales</i> <p><i>Archivos de Referencia</i></p>	<p><i>Documento de Concepto de Juego</i></p> <ul style="list-style-type: none"> - <i>Elementos de Audio</i>

PRO MER DLD	Proponer los <i>Modos de Distribución</i> del videojuego e <ul style="list-style-type: none"> - <i>Mercado de aplicaciones</i> - <i>Portal</i> - <i>Servicio</i> - <i>Otro</i> 	<i>Concepto Inicial [aprobado]</i> <i>Documento de Concepto de Juego</i> - <i>Todos los elementos previamente descritos</i>	<i>Documento de Concepto de Juego</i> - <i>Modos de Distribución</i>
PRO DLD	Generar el <i>Documento de Concepto de Juego</i> integrando los elementos previamente identificados y documentados.	<i>Documento de Concepto de Juego</i> - <i>Todos los elementos anteriormente descritos</i>	<i>Documento de Concepto de Juego</i> - <i>Todos los elementos anteriormente descritos</i>
PRO	Validar <i>Documento de Concepto de Juego</i> .	<i>Documento de Concepto de Juego</i> - <i>Todos los elementos anteriormente descritos</i>	<i>Documento de Concepto de Juego</i> [validado]
EQT	Construir un <i>Prototipo del Videojuego</i> con base en el concepto detallado del juego. Este prototipo servirá como base o estándar para el juego completo en fases posteriores.	<i>Documento de Concepto de Juego</i> [validado]	<i>Prototipo del Videojuego</i>
PRO DLD	Evaluar <i>Prototipo del Videojuego</i> .	<i>Prototipo del Videojuego</i>	<i>Prototipo del Videojuego</i> [evaluado]

TABLA B.15 - Productos de Trabajo de la actividad Concepto del Videojuego

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Planteamiento del Videojuego	Estudio, Publisher
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Documento de Concepto de Juego	Análisis de Mercado y Financiamiento Requerimientos y Diseño del Videojuego Inicio y Planeación del Proyecto
Archivos de Referencia	Requerimientos y Diseño del Videojuego
Prototipo del Videojuego	PREPRODUCCIÓN PRODUCCIÓN
Estándares de Plataforma	Requerimientos
PRODUCTOS INTERNOS	
NOMBRE	
Generación de Idea	
Concepto Inicial	
Análisis Competitivo	

IV.2 - Análisis de Mercado y Financiamiento

IV.2.1 - Análisis de Mercado

En la actividad *Análisis de Mercado* se proponen tareas para investigar y evaluar los videojuegos similares que estén publicados, con el objetivo de probar que hay un mercado potencial para el videojuego que se quiere construir.

TABLA B.16 - Tareas para la actividad Análisis de Mercado

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
ANÁLISIS DE MERCADO			
MER	Investigar <i>Políticas de Publicación de Mercados</i> .	<i>Documento de Concepto de Juego [validado]</i>	<i>Políticas de Publicación de Mercados</i>
MER	Realizar y documentar <i>Investigación Preliminar</i> . Investigar y evaluar el <i>Mercado de Videojuegos</i> . Investigar y evaluar videojuegos similares que estén publicados y probar que hay un mercado para el videojuego que se quiere construir. Investigar y evaluar <i>Aspectos Legales</i> relacionados con derecho de autor, licencias, etc. Evaluar <i>Políticas de Publicación de Mercados</i> potenciales.	<i>Documento de Concepto de Juego [validado]</i> <i>Políticas de Publicación de Mercados</i>	<i>Investigación Preliminar</i> - <i>Evaluación del Mercado de Videojuegos.</i> - <i>Evaluación de Aspectos Legales.</i> - <i>Evaluación de Políticas de Publicación de Mercados</i>
PRO MER DLD	Realizar <i>Análisis FODA</i> . Identificar las fortalezas y debilidades del concepto del videojuego, oportunidades de mercado y amenazas que podrían impactar el éxito del videojuego.	<i>Documento de Concepto de Juego [validado]</i>	<i>Análisis FODA</i>
PRO MER DLD	Realizar <i>Análisis Competitivo</i> . Identificar las fortalezas y debilidades de la competencia del videojuego.	<i>Documento de Concepto de Juego [validado]</i> <i>Análisis FODA</i>	<i>Análisis Competitivo</i>

TABLA B.17 - Productos de Trabajo de la actividad Análisis de Mercado

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Documento de Concepto de Juego	Concepto del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Políticas de Publicación de Mercados	Requerimientos y Diseño del Videojuego Cierre del Proyecto
Investigación Preliminar	Financiamiento Inicio y Planeación del Proyecto
Análisis Competitivo	Financiamiento Inicio y Planeación del Proyecto
PRODUCTOS INTERNOS	
NOMBRE	
Análisis FODA	

IV.2.2 - Financiamiento

En la actividad *Financiamiento* se define un *Modelo de Financiamiento* con el objetivo de que alguien apruebe e invierta los recursos necesarios para desarrollar el proyecto.

TABLA B.18 - Tareas para la actividad Financiamiento

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
FINANCIAMIENTO			
PRO MER DLD	Determinar el <i>Modelo de Financiamiento</i> . Determinar la estrategia para conseguir el financiamiento para desarrollar el videojuego. <ul style="list-style-type: none"> - <i>Crowdfunding</i> - <i>Aporte no Reembolsable</i> - <i>Subsidios y créditos</i> - <i>Incubadoras y aceleradoras</i> - <i>Otro</i> 	<i>Investigación Preliminar</i> <i>Análisis Competitivo</i>	<i>Modelo de Financiamiento</i>
PRO DLD ALD	Crear <i>Documento de Venta (Pitch)</i> considerando los elementos descritos en el <i>Documento de Concepto de Juego</i> .	<i>Investigación Preliminar</i> <i>Modelo de Financiamiento</i> <i>Documento de Concepto de Juego [validado]</i>	<i>Documento de Venta (Pitch)</i>
PRO	Validar <i>Documento de Venta (Pitch)</i> .	<i>Documento de Venta (Pitch)</i>	<i>Documento de Venta (Pitch) [validado]</i>
PRO DLD ALD	Presentar concepto de videojuego a <i>Cliente, Publisher</i> o alguna otra fuente de financiamiento.	<i>Documento de Venta (Pitch) [validado]</i> <i>Prototipo del Videojuego</i>	<i>Presentación</i>

TABLA B.19 - Productos de Trabajo de la actividad Financiamiento

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Investigación Preliminar	Análisis de Mercado
Análisis Competitivo	Análisis de Mercado
Prototipo del Videojuego	Concepto del Videojuego
Documento de Concepto de Juego	Concepto del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Documento de Venta (Pitch)	Fuente de financiamiento
Prototipo del Videojuego	Fuente de financiamiento
Presentación	Fuente de financiamiento
PRODUCTOS INTERNOS	
NOMBRE	
Modelo de Financiamiento	

IV. Fase de Preproducción

En la *Fase de Preproducción* se analizarán los diferentes tipos de *Requerimientos* que tendrá el videojuego para cumplir con los objetivos y expectativas, y se definirá la *Línea de Producción* de los diferentes *Assets* que requerirá el videojuego. También se realizará el *Diseño Detallado del Videojuego* con base en la *Especificación de Requerimientos* y finalizará con el *Diseño Detallado de Software*.

IV.3 - Requerimientos y Diseño del Videojuego

IV.3.1 - Requerimientos

En esta actividad se analizan los diferentes tipos de *Requerimientos* (*Tecnología, Funcionales, IU, Fiabilidad, Desempeño, etc.*) que el videojuego tendrá que cubrir para cumplir con los objetivos y expectativas del juego deseados. También se define la *Línea de Producción* de los diferentes *Assets* que requerirá el videojuego.

TABLA B.20 - Tareas para la actividad Requerimientos

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
REQUERIMIENTOS			
DLD EQT	Asignar <i>Tareas</i> a los miembros del <i>Equipo de Trabajo</i> de acuerdo a cada rol, basado en el <i>Plan del Proyecto</i> actual.	<i>Plan del Proyecto</i> [revisado] - <i>Tareas</i>	
DLD	Identificar y priorizar las características principales que tendrá el videojuego (multijugador, mecánicas de juego, jugabilidad (gameplay), entre otros).	<i>Documento de Concepto de Juego</i> [validado]	<i>Características del Videojuego (Game Features)</i>

PLD	<p>Evaluar la tecnología necesaria en el proyecto. Se deberá evaluar el <i>Motor de Juego (Game Engine)</i>, <i>Motor de IA</i>, <i>Motor de Física</i>, <i>Motor de Rendering</i>, herramientas (arte, <i>scripting</i>) y otros elementos técnicos necesarios para proporcionar la funcionalidad deseada del videojuego.</p> <p>Decidir cómo se va a obtener la tecnología necesaria.</p>	<i>Documento de Concepto de Juego [validado]</i>	<i>Evaluación de Tecnología</i>
PLD	Definir <i>Tecnología</i> necesaria para desarrollar los elementos técnicos necesarios para proporcionar la funcionalidad deseada del videojuego.	<i>Evaluación de Tecnología</i>	<i>Especificación de Requerimientos</i> - <i>Tecnología</i>
DLD	Identificar o actualizar <i>Requerimientos Funcionales</i> . Identificar las necesidades establecidas que debe satisfacer el software cuando se utiliza en condiciones específicas.	<i>Documento de Concepto de Juego [validado]</i>	<i>Especificación de Requerimientos</i> - <i>Requerimientos Funcionales</i>
DLD DIU AIG PIU	Identificar o actualizar <i>Requerimientos de Interfaz de Usuario (IU)</i> . Definir las características de la IU que permitan comprender y aprender el uso de <i>Software</i> fácilmente.	<i>Documento de Concepto de Juego [validado]</i>	<i>Especificación de Requerimientos</i> - <i>Requerimientos de Interfaz de Usuario</i>
DLD PLD	Identificar o actualizar <i>Requerimientos de Interfaces Externas</i> . Definir las interfaces con otro <i>Software</i> o <i>Hardware</i> . <ul style="list-style-type: none"> - Dispositivo de entrada (control, joystick, ratón, teclado, otros). - Dispositivo de salida (monitor, televisión, dispositivos táctiles, sonido). 	<i>Documento de Concepto de Juego [validado]</i>	<i>Especificación de Requerimientos</i> - <i>Requerimientos de Interfaces Externas</i>
DLD PLD	Identificar o actualizar <i>Requerimientos de Fiabilidad</i> . Especificar el nivel de ejecución de <i>Software</i> referente a su madurez, tolerancia a fallas y su capacidad de recuperación.	<i>Documento de Concepto de Juego</i>	<i>Especificación de Requerimientos</i> - <i>Requerimientos de Fiabilidad</i>
DLD PLD	Identificar o actualizar <i>Requerimientos de Desempeño</i> . Identificar el hardware, memoria y requerimientos de red para un óptimo desempeño del videojuego. Considerar el <i>Impacto del Desempeño Gráfico</i> . <ul style="list-style-type: none"> - Requerimientos de memoria específicos para cada plataforma - Requerimientos de desempeño gráfico - Requerimientos de red para 	<i>Documento de Concepto de Juego [validado]</i> <i>Estándares de Plataforma</i>	<i>Especificación de Requerimientos</i> - <i>Requerimientos de Desempeño</i>

	multijugador		
DLD PLD	Identificar o actualizar <i>Limitaciones y Restricciones</i> . Identificar limitaciones o restricciones del diseño del videojuego y construcción, necesidades impuestas por el <i>Cliente</i> .	<i>Documento de Concepto de Juego [validado]</i>	<i>Especificación de Requerimientos</i> - <i>Limitaciones y Restricciones</i>
DLD PLD MER	Identificar <i>Requerimientos Legales y Regulatorios</i> . Necesidades impuestas por mercados, regulaciones, etc.	<i>Documento de Concepto de Juego</i> <i>Políticas de Publicación en Mercados</i>	<i>Especificación de Requerimientos</i> - <i>Requerimientos Legales y Regulatorios</i>
DLD ALD	Identificar o actualizar <i>Requerimientos Emocionales, Requerimientos de Experiencia y Requerimientos de Jugabilidad</i> . Entender los requerimientos emocionales que el diseñador quiere capturar.	<i>Documento de Concepto de Juego</i>	<i>Especificación de Requerimientos</i> - <i>Requerimientos Emocionales</i> - <i>Requerimientos de Experiencia</i> - <i>Requerimientos de Jugabilidad</i>
PLD	Identificar <i>Limitaciones del Sistema</i> . Aclarar ciertos límites que no deberían ser pasados (número de polígonos, número de mensajes enviados por segundo, número simultaneo de explosiones en pantalla, etc.).	<i>Especificación de Requerimientos</i> - <i>Tecnología</i>	<i>Especificación de Requerimientos</i> - <i>Limitación del Sistema</i>
DLD ALD PLD	Definir o actualizar <i>Línea de Producción</i> de los diferentes <i>Assets</i> del videojuego y el <i>Software (Game Build)</i> . Definir los pasos necesarios para que los <i>Componentes de Software</i> y los <i>Assets</i> del juego funcionen correctamente.	<i>Especificación de Requerimientos</i> - <i>Tecnología</i> - <i>Limitación del Sistema</i>	<i>Especificación de Requerimientos</i> - <i>Línea de Producción (Production Pipeline)</i>
PRO DLD PLD ALD	Generar o actualizar la <i>Especificación de Requerimientos</i> .	<i>Especificación de Requerimientos</i> - <i>Todos los elementos anteriormente descritos</i>	<i>Especificación de Requerimientos</i> - <i>Todos los elementos anteriormente descritos</i>
PRO LAR	Verificar y obtener la aprobación de la <i>Especificación de Requerimientos</i> . Para cambios se propone <i>Solicitud de Cambio</i> .	<i>Especificación de Requerimientos</i>	<i>Listas de Verificación</i> <i>Especificación de Requerimientos [verificado]</i> <i>Solicitud de Cambio [propuesta]</i>

PRO LAR	Validar y obtener la aprobación de la <i>Especificación de Requerimientos</i> .	<i>Especificación de Requerimientos [verificado]</i>	<i>Listas de Validación [validado]</i> <i>Especificación de Requerimientos [validado]</i>
PLD	Incorporar la <i>Especificación de Requerimientos</i> a la <i>Configuración del Software</i> en la línea base.	<i>Especificación de Requerimientos [validado]</i>	<i>Configuración del Software</i> - <i>Especificación de Requerimientos [validado, en línea base]</i>

TABLA B.21 - Productos de Trabajo de la actividad Requerimientos

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [revisado]	Planeación del Proyecto
Documento de Concepto de Juego [validado]	Concepto del Videojuego
Estándares de Plataforma	Análisis de Mercado y Financiamiento
Políticas de Publicación en Mercados	Análisis de Mercado
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Especificación de Requerimientos [validado]	Diseño del Videojuego
PRODUCTOS INTERNOS	
NOMBRE	
Características del Videojuego (Game Features)	
Evaluación de Tecnología	
Tecnología	
Requerimientos Funcionales	
Requerimientos de Interfaz de Usuario	
Requerimientos de Interfaces Externas	
Requerimientos de Fiabilidad	
Requerimientos de Desempeño	
Limitaciones y Restricciones	
Requerimientos Legales y Regulatorios	
Requerimientos Emocionales	
Requerimientos de Experiencia	
Requerimientos de Jugabilidad	
Limitación del Sistema	
Línea de Producción	
Listas de Verificación	
Configuración del Software	
Solicitud de Cambio [propuesta]	

IV.3.2 - Diseño del videojuego

En esta actividad se realiza el *Diseño del Videojuego* de forma detallada con base en la *Especificación de Requerimientos* considerando el diseño conceptual inicial y se crea el *Documento de Diseño de Juego (Game Design Document)*.

TABLA B.22 - Tareas para la actividad Diseño del Videojuego

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
DISEÑO DEL VIDEOJUEGO			
DLD EDI	Asignar <i>Tareas</i> a los miembros del <i>Equipo de Trabajo</i> de acuerdo a cada rol, basado en el <i>Plan del Proyecto</i> actual.	<i>Plan del Proyecto</i> [revisado] - <i>Tareas</i>	
DLD	Generar o actualizar el <i>Documento de Diseño de Juego</i> . Crear o actualizar el diseño detallado del videojuego con base en la <i>Especificación de Requerimientos</i> y el <i>Documento de Concepto de Juego</i> .	<i>Especificación de Requerimientos</i> [validado] <i>Documento de Concepto de Juego</i> [validado]	<i>Documento de Diseño de Juego</i> [creado]
DLD	Definir la <i>Información General del Juego</i> basado en el <i>Documento de Concepto de Juego</i> .	<i>Especificación de Requerimientos</i> [validado] <i>Documento de Concepto de Juego</i> [validado]	<i>Documento de Diseño de Juego</i> - <i>Información General del Juego</i>
DLD ESC GUI	Crear o actualizar <i>Historia del Juego</i> y <i>Resumen Narrativo</i> . Describen lo que pasa en el juego y a quien le pasa. La acción y las motivaciones de los personajes.	<i>Especificación de Requerimientos</i> [validado] <i>Documento de Concepto de Juego</i> [validado]	<i>Documento de Diseño de Juego</i> - <i>Historia del Juego</i> - <i>Resumen Narrativo</i>
DLD ESC	Crear o actualizar <i>Lista de Eventos Clave en la Historia</i> . Estos son los puntos clave en la historia que indican la necesidad de incluir algún elemento de jugabilidad, diálogos, cinemáticas o introducen un nuevo nivel en el juego.	<i>Documento de Diseño de Juego</i> - <i>Historia del Juego</i> - <i>Resumen Narrativo</i>	<i>Documento de Diseño de Juego</i> - <i>Lista de Eventos Clave en la Historia</i>
DLD ESC GUI	Crear o actualizar <i>Guión Literario</i> del juego. Crear los diálogos para todas las partes del juego, incluyendo cinemáticas y jugabilidad.	<i>Documento de Diseño de Juego</i> - <i>Historia del Juego</i> - <i>Resumen Narrativo</i> - <i>Lista de Eventos Clave en la Historia</i>	<i>Documento de Diseño de Juego</i> - <i>Guión Literario</i>

DLD ESC	Crear o actualizar <i>Biblia del Mundo</i> . Establecer las reglas de consistencia para los escenarios del juego. Estas reglas ayudan a los artistas, animadores, diseñadores de niveles y programadores a entender cómo crear <i>Assets</i> del juego y el tipo de física permitida en el juego.	<i>Documento de Diseño de Juego</i> - <i>Historia del Juego</i> - <i>Resumen Narrativo</i>	<i>Documento de Diseño de Juego</i> - <i>Biblia del Mundo</i>
DLD ALD	Definir o actualizar <i>Lista de Personajes</i> que aparecen en el juego. Esta lista incluye dónde y cuándo aparece cada personaje en el juego, qué habilidades y equipo tiene y todos los riesgos asociados con la creación del personaje.	<i>Documento de Diseño de Juego</i> - <i>Historia del Juego</i> - <i>Resumen Narrativo</i>	<i>Documento de Diseño de Juego</i> - <i>Lista de Personajes</i>
DLD ALD ESC ACN COM	Diseñar o actualizar <i>Jugabilidad (Gameplay)</i> . - Retos para el jugador - Recompensas del jugador - Curva de aprendizaje - Esquema del control - Acciones del jugador - Elementos del multijugador - Otros	<i>Documento de Diseño de Juego</i> - <i>Biblia del Mundo</i>	<i>Documento de Diseño de Juego</i> - <i>Jugabilidad (Gameplay)</i>
DLD	Diseñar o actualizar las <i>Mecánicas de Juego</i> . Balancear o equilibrar las mecánicas que rompan la experiencia de juego deseada. - Objetos o elementos - Comportamientos - Reglas	<i>Documento de Concepto de Juego</i> - <i>Jugabilidad (Gameplay)</i>	<i>Documento de Concepto de Juego</i> - <i>Mecánicas de Juego</i>
DLD	Diseñar o actualizar <i>Control</i> del Juego. Diseñar o actualizar comandos, mecánicas que se ejecuten mediante el control y la configuración que tendrá el control.	<i>Documento de Diseño de Juego</i> - <i>Mecánicas</i> - <i>Jugabilidad</i> - <i>Interfaz de Usuario</i>	<i>Documento de Diseño de Juego</i> - <i>Control</i>

DLD DNV ACN	<p>Crear o actualizar el <i>Diseño de Niveles</i> y los <i>Planos de Niveles</i> que tendrá el juego. Planear lo que el jugador hará en cada nivel y como progresará a través del juego, el diseño de niveles contemplará:</p> <ul style="list-style-type: none"> - Mecánicas relativas al escenario - Mecánicas de los enemigos - Objetivos - Resolución de acertijos - Estrategias y dinámicas - Eventos - Jefes finales - Puntos de guardado - Recolección de premios - Rutas a través de un escenario - Arte conceptual - Otros 	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Mundo del Juego</i> - <i>Historia del Juego</i> - <i>Resumen Narrativo</i> - <i>Lista de Personajes</i> 	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Diseño de Niveles</i> - <i>Planos de Niveles</i>
ALD	<p>Definir o actualizar <i>Biblia de Arte</i>. Definir las guías para ayudar a mantener la apariencia del videojuego consistente. Definir el estilo de personajes, objetos y ambientes.</p>	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Mundo del Juego</i> - <i>Historia del Juego</i> - <i>Resumen Narrativo</i> - <i>Lista de Personajes</i> - <i>Diseño de Niveles</i> - <i>Planos de Nivel</i> <p><i>Archivos de Referencia</i></p>	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Biblia de Arte (Guía de Estilo de Arte.)</i>
ACN	<p>Crear o actualizar el <i>Arte Conceptual</i> requerido por el videojuego (objetos, personajes y escenarios donde se llevará a cabo la acción del juego).</p>	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Biblia de Arte (Guía de Estilo de Arte)</i> <p><i>Archivos de Referencia</i></p>	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Arte Conceptual</i>
DLD DIU ACN AIG	<p>Diseñar o actualizar la <i>Interfaz de Usuario</i> del juego. Definir la forma en la que el jugador interactuará con el videojuego y con qué mecanismos contará para ello.</p>	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Biblia de Arte (Guía de Estilo de Arte)</i> <p><i>Archivos de Referencia</i></p>	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Interfaz de Usuario</i>
DLD DIU	<p>Crear o actualizar <i>Diagrama de Flujo de Interfaz de Usuario</i>. Este diagrama se centra en los menús vistos en pantalla por el jugador durante el juego. Describe que pantallas se necesitan y en qué orden deben aparecer.</p>	<p><i>Documento de Diseño de Juego:</i></p> <ul style="list-style-type: none"> - <i>Interfaz de Usuario</i> 	<p><i>Documento de Diseño de Juego</i></p> <ul style="list-style-type: none"> - <i>Diagrama de Flujo de Interfaz de Usuario</i>

DLD DNV	Crear o actualizar <i>Diagrama de Flujo del Juego</i> . Establecer la progresión a través del juego desde el punto de vista de la interacción del jugador con el juego, los diferentes niveles que encontrará y en el orden en el que aparecerán, los objetivos del jugador en cada nivel, poderes y habilidades del jugador en cada escenario, narrativas clave y cualquier cinemática que sea necesaria para explicar la historia.	<i>Documento de Diseño de Juego:</i> - <i>Mundo del Juego</i> - <i>Historia del Juego</i> - <i>Resumen Narrativo</i> - <i>Lista de Personajes</i> - <i>Diseño de Niveles</i> - <i>Planos de Nivel</i> - <i>Arte Conceptual</i>	<i>Documento de Diseño de Juego</i> - <i>Diagrama de Flujo del Juego</i>
DLD PLD PIA	Diseñar o actualizar <i>Inteligencia Artificial (IA)</i> del videojuego. Diseñar la IA del videojuego incluye métodos para producir la ilusión de inteligencia en la conducta de personajes no jugables.	<i>Documento de Diseño de Juego:</i> - <i>Lista de Personajes</i> - <i>Diseño de Niveles</i> - <i>Planos de Nivel</i> <i>Archivos de Referencia</i>	<i>Documento de Diseño de Juego</i> - <i>Inteligencia Artificial</i>
DLD	Identificar o actualizar los <i>Logros (Achievements)</i> que tendrá el videojuego. Definir el número de Logros que tendrá el videojuego y describir en qué consiste cada uno y como se obtendrán.	<i>Documento de Diseño de Juego</i> - <i>Mecánicas</i> - <i>Jugabilidad</i> - <i>Lista de Personajes</i> - <i>Diseño de Niveles</i>	<i>Documento de Diseño de Juego</i> - <i>Logros</i>
DLD EQT	Construir Prototipos que permitan validar o probar <i>Jugabilidad</i> y <i>Mecánicas de Juego</i> o cualquier cosa que no esté bien definida (<i>Línea de Producción</i>).	<i>Documento de Diseño de Juego</i> - <i>Mecánicas</i> - <i>Jugabilidad</i> - <i>Control</i>	<i>Prototipos</i>
DLD DAU COM PAU	Crear o actualizar <i>Elementos de Audio</i> del videojuego (narración, efectos de sonido, música, voces, diálogos, etc.) de acuerdo a la ambientación, historia, jugabilidad.	<i>Documento de Diseño de Juego</i> - <i>Ambientación del Juego</i> - <i>Resumen de la Historia</i> - <i>Arte Conceptual</i> - <i>Jugabilidad</i> - <i>Mecánicas</i> <i>Archivos de Referencia</i>	<i>Documento de Diseño de Juego</i> - <i>Elementos de Audio</i>
PRO DLD	Generar el <i>Documento de Diseño de Juego</i> integrando los elementos previamente identificados y documentados.	<i>Documento de Diseño de Juego</i> - <i>Todos los elementos anteriormente descritos</i>	<i>Documento de Diseño de Juego</i> - <i>Todos los elementos anteriormente descritos</i>

PRO DLD	Validar <i>Documento de Diseño de Juego</i> .	<i>Documento de Diseño de Juego</i> - <i>Todos los elementos anteriormente descritos</i>	<i>Documento de Diseño de Juego [validado]</i>
PLD	Incorporar el <i>Documento de Diseño de Juego</i> a la <i>Configuración del Software</i> en la línea base.	<i>Documento de Diseño de Juego [validado]</i>	<i>Configuración del Software</i> - <i>Documento de Diseño de Juego [validado, en línea base]</i>

TABLA B.23 - Productos de Trabajo de la actividad Diseño del Videojuego

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [revisado]	Planeación del Proyecto
Especificación de Requerimientos [validado]	Requerimientos
Documento de Concepto de Juego [validado]	Concepto del Videojuego
Archivos de Referencia	Concepto del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Documento de Diseño de Juego [validado]	Diseño Técnico del Videojuego Construcción de Software
Prototipos	Diseño Técnico del Videojuego Construcción de Software
PRODUCTOS INTERNOS	
NOMBRE	
Información General del Juego	
Historia del Juego Resumen Narrativo	
Lista de Eventos Clave en la Historia	
Guión Literario	
Biblia del Mundo	
Lista de Personajes	
Jugabilidad (Gameplay)	
Mecánicas de Juego	
Control	
Diseño de Niveles	
Planos de Niveles	
Biblia de Arte (Guía de Estilo de Arte.)	
Arte Conceptual	
Interfaz de Usuario	
Diagrama de Flujo de Interfaz de Usuario	
Diagrama de Flujo del Juego	
Inteligencia Artificial	
Logros (Achievements)	
Elementos de Audio	
Configuración del Software	

IV.4 - Diseño Técnico del Videojuego

El diseño técnico del videojuego, comprende principalmente dos actividades: *Arquitectura de Software* y *Diseño Detallado del Software*.

IV.4.1 - Arquitectura de Software

En la actividad *Arquitectura de Software* se diseña la arquitectura a un alto nivel de abstracción para describir la estructura global del *Software*.

TABLA B.24 - Tareas para la actividad Arquitectura

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
ARQUITECTURA			
PLD	Asignar <i>Tareas</i> a los miembros del <i>Equipo de Trabajo</i> de acuerdo a cada rol, basado en el <i>Plan del Proyecto</i> actual.	<i>Plan del Proyecto [aceptado]</i> - <i>Tareas</i>	
PLD	Crear o actualizar <i>Especificación Técnica de Desarrollo</i> (herramientas, lenguajes de programación, estándares de codificación).	<i>Especificación de Requerimientos [validado]</i>	<i>Especificación Técnica de Desarrollo</i>
PLD	Crear o actualizar <i>Descripción de Arquitectura de Software</i> con elementos identificados en la <i>Especificación de Requerimientos</i> . Identificar elementos que influyen en el desarrollo del videojuego y conforman su estructura como: personas, software, hardware, otros sistemas, conectividad, equipo de trabajo, medios de distribución.	<i>Especificación de Requerimientos [validado]</i> <i>Especificación Técnica de Desarrollo</i> <i>Documento de Diseño de Juego [validado]</i>	<i>Descripción de Arquitectura de Software</i> <i>Registro de Trazabilidad</i>
PLD	Verificar y obtener la aprobación de la <i>Descripción de Arquitectura de Software</i> .	<i>Descripción de Arquitectura de Software</i>	<i>Descripción de Arquitectura de Software [verificada]</i> <i>Listas de Verificación</i>
PLD	Validar y obtener la aprobación de la <i>Descripción de Arquitectura de Software</i> .	<i>Descripción de Arquitectura de Software</i>	<i>Descripción de Arquitectura de Software [validada]</i> <i>Listas de Validación</i>

TABLA B.25 - Productos de Trabajo de la actividad Arquitectura

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto [aceptado]	Planeación del Proyecto
Especificación de Requerimientos [validada]	Requerimientos
Documento de Diseño de Juego [validado]	Diseño del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Especificación Técnica de Desarrollo	Diseño Detallado de Software
Descripción de Arquitectura de Software [validada]	Diseño Detallado de Software
Registro de Trazabilidad	Diseño Detallado de Software
PRODUCTOS INTERNOS	
NOMBRE	
Listas de Verificación	
Listas de Validación	

IV.4.2 - Diseño Detallado de Software

En la actividad *Diseño Detallado de Software*, las tareas a realizar en esta actividad se enfocan en los siguientes elementos:

- ***Diseño Detallado de Software***. A un bajo nivel de abstracción incluye detalle de los *Componentes de Software* para facilitar su construcción y prueba dentro del ambiente de desarrollo (*Motor de Juego*).
- ***Escenarios y Casos de Prueba***. Establecen los elementos necesarios para probar el código, integrar los *Componentes de Software* y probar el videojuego.

TABLA B.26 - Tareas para la actividad Diseño Detallado de Software

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
DISEÑO DETALLADO DE SOFTWARE			
PLD EPR	<p>Crear o actualizar el <i>Diseño Detallado de Software</i> con base en la <i>Arquitectura de Software</i>, <i>Especificación de Requerimientos</i> y <i>Documento de Diseño de Juego</i>. Se deberá contemplar el diseño de <i>Componentes de Software</i> específicos de juego, por ejemplo:</p> <ul style="list-style-type: none"> - Motor de Juego (Game Engine) - Jugabilidad (Gameplay) - Control - Cámara - Enemigos - Niveles - Eventos - Personajes 	<p><i>Especificación de Requerimientos</i> [validada]</p> <p><i>Documento de Diseño de Juego</i> [validado]</p> <p><i>Descripción de Arquitectura de Software</i> [validada]</p>	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Arquitectura de Software</i> - <i>Diseño Detallado de Software</i>

	<ul style="list-style-type: none"> - Interfaz de Usuario (IU) - Inteligencia Artificial (IA) - Física - Multijugador - Herramientas (Tools) - Logros (Achievements) - Otros <p>Definir patrones de diseño a utilizar, diagramas de clases, diagramas de flujo generales, casos de uso, etc.</p> <p>Incorporar <i>Arquitectura de Software</i> y el <i>Diseño Detallado de Software</i> al <i>Documento de Diseño Técnico</i></p>		
PLD	<p>Crear o actualizar <i>Casos de Pruebas de Integración</i> con el objetivo de verificar el correcto funcionamiento de los <i>Componentes de Software</i> al ser integrados.</p>	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Diseño Detallado de Software</i> 	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Casos de Pruebas de Integración</i>
PLD	<p>Crear o actualizar grupos de <i>Escenarios y Casos de Pruebas de Usabilidad</i> para la plataforma en la que se desarrollará el videojuego.</p> <p>Definir lista de tareas a realizar por los usuarios participantes en las pruebas de usabilidad.</p> <p>Definir cuestionarios para aplicar a usuarios participantes en las pruebas.</p> <p>Determinar el periodo en el que se realizarán las pruebas de usabilidad.</p>	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Diseño Detallado de Software</i> 	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Escenarios y Casos de Pruebas de Usabilidad</i> - <i>Listas de tareas a realizar</i> - <i>Cuestionarios para aplicar</i>
PLD	<p>Crear o actualizar grupos de <i>Escenarios y Casos de Pruebas de Jugabilidad</i> para la plataforma en la que se desarrollará el videojuego.</p> <p>Definir lista de tareas a realizar por los usuarios participantes en las pruebas de jugabilidad.</p> <p>Definir cuestionarios para aplicar a usuarios participantes en las pruebas.</p> <p>Determinar el periodo en el que se realizarán las pruebas de jugabilidad.</p>	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Diseño Detallado de Software</i> - 	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Escenarios y Casos de Pruebas de Jugabilidad</i> - <i>Listas de tareas a realizar</i> - <i>Cuestionarios para aplicar</i>

PLD	Crear grupos de <i>Pruebas de Publicación en Mercado</i> por plataforma o tienda de aplicaciones para cumplir sus las políticas de publicación.	<i>Políticas de Publicación en Mercados</i>	<i>Documento de Diseño Técnico</i> - <i>Pruebas de Publicación en Mercados</i>
PLD	Generar el <i>Documento de Diseño Técnico</i> integrando los elementos previamente identificados y documentados.	<i>Documento de Diseño Técnico</i> - <i>Todos los elementos anteriormente descritos</i>	<i>Documento de Diseño Técnico</i> - <i>Todos los elementos anteriormente descritos</i>
PLD	Actualizar el <i>Registro de Trazabilidad</i> incorporando el <i>Diseño Detallado de Software, Casos de Pruebas de Integración, Escenarios y Casos de Pruebas de Usabilidad, Escenarios y Casos de Pruebas de Jugabilidad, y Pruebas de Publicación en Mercados.</i>	<i>Documento de Diseño Técnico</i> - <i>Todos los elementos previamente definidos</i>	<i>Registro de Trazabilidad [actualizado]</i>
PLD	Incorporar <i>Documento de Diseño Técnico y Registro de Trazabilidad</i> a la <i>Configuración de Software</i> y establecer en línea base.	<i>Documento de Diseño Técnico</i> <i>Registro de Trazabilidad</i>	<i>Configuración del Software</i> - <i>Documento de Diseño Técnico [en línea base]</i> - <i>Registro de Trazabilidad [en línea base]</i>

TABLA B.27 - Productos de Trabajo de la actividad Diseño Detallado de Software

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Especificación de Requerimientos [validada]	Requerimientos
Documento de Diseño de Juego [validado]	Diseño del Videojuego
Descripción de Arquitectura de Software [validada]	Arquitectura
Políticas de Publicación en Mercados	Análisis de Mercado
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Documento de Diseño Técnico	Construcción de Software Integración y Pruebas de Software
PRODUCTOS INTERNOS	
NOMBRE	
Diseño Detallado de Software	
Casos de Pruebas de Integración	
Escenarios y Casos de Pruebas de Usabilidad	
Escenarios y Casos de Pruebas de Jugabilidad	

Pruebas de Publicación en Mercados
Listas de tareas a realizar
Cuestionarios para aplicar
Registro de Trazabilidad
Configuración del Software

IV. Fase de Producción

En la *Fase de Producción* se construirán los diferentes *Assets* que requerirá el videojuego (texturas, modelos, animaciones, cinemáticas, escenarios, etc.) de acuerdo a la *Línea de Producción*, así como los *Componentes de Software* específicos con base en la *Especificación Técnica de Desarrollo* y *Diseño Detallado de Software*. Finalmente se integrarán los *Componentes de Software* que se han desarrollado y se aplicarán los escenarios o casos de prueba para verificar el correcto funcionamiento del software.

IV.5 - Construcción de Software

En la actividad *Construcción de Software* se establecen tareas para la construcción de *Assets* y *Componentes de Software* (de acuerdo a la *Línea de Producción*) y el diseño de *Pruebas Unitarias* que permiten detectar y corregir defectos encontrados en los *Componentes de Software* desarrollados.

TABLA B.28 - Tareas para la actividad de Construcción de Software

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
CONSTRUCCIÓN DE SOFTWARE			
PLD ALD	Asignar <i>Tareas</i> a los miembros del <i>Equipo de Trabajo</i> de acuerdo al rol y <i>Plan del Proyecto</i> actual.	<i>Plan del Proyecto</i> [aceptado] - <i>Tareas</i>	
PMJ	Construir o actualizar <i>Motor de Juego</i> . Crear la estructura interna del videojuego, en la que después se añadirá el contenido.	<i>Documento de Diseño Técnico</i> - <i>Diseño Detallado de Software</i>	<i>Motor de Juego (Game Engine)</i>
PHR	Construir o actualizar <i>Herramientas (Tools)</i> para facilitar la implementación del contenido.	<i>Documento de Diseño Técnico</i> - <i>Diseño Detallado de Software</i>	<i>Herramientas (Tools)</i>
EAR	Construir o actualizar <i>Assets</i> de arte de acuerdo a la <i>Línea de Producción</i> . - Textos - Modelos - Animaciones - Cinemáticas - Interfaz de Usuario - Mercadotecnia	<i>Especificación de Requerimientos</i> - <i>Línea de Producción</i> <i>Documento de Diseño de Juego</i>	<i>Assets</i> - <i>Arte</i>

DAU COM PAU	<p>Crear o actualizar <i>Assets de Audio</i> para el videojuego.</p> <ul style="list-style-type: none"> - <i>Música</i> - <i>Efectos de Sonido</i> - <i>Voces</i> - <i>Diálogos</i> 	<p><i>Especificación de Requerimientos</i></p> <ul style="list-style-type: none"> - <i>Línea de Producción</i> <p><i>Documento de Diseño de Juego</i></p>	<p><i>Assets</i></p> <ul style="list-style-type: none"> - <i>Audio</i>
EPR	<p>Construir o actualizar los <i>Componentes de Software</i> necesarios con base en la <i>Especificación Técnica de Desarrollo y Diseño Detallado de Software</i> de:</p> <ul style="list-style-type: none"> - Motor de Juego (Game Engine) - Jugabilidad (Gameplay) - Control - Cámara - Enemigos - Eventos - Personajes - Interfaz de Usuario (IU) - Inteligencia Artificial (IA) - Física - Multijugador - Herramientas (Tools) - Logros (Achievements) - Tutorial de Juego - Otros 	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Diseño Detallado de Software</i> <p><i>Especificación Técnica de Desarrollo</i></p> <p><i>Assets</i></p> <p><i>Documento de Diseño del Juego</i></p> <p><i>Especificación de Requerimientos</i></p>	<p><i>Componentes de Software</i></p>
ACL PLD	<p>Diseñar o actualizar los <i>Casos de Pruebas Unitarias</i> y aplicarlos para verificar que los <i>Componentes de Software</i> cumplen con la funcionalidad establecida.</p>	<p><i>Componentes de Software</i></p>	<p><i>Componentes de Software [probados]</i></p> <p><i>Casos de Pruebas Unitarias [aplicados]</i></p>
PAC	<p>Corregir defectos encontrados hasta lograr pruebas exitosas.</p>	<p><i>Componentes de Software [probados]</i></p>	<p><i>Componentes de Software [corregidos]</i></p>
PLD	<p>Actualizar <i>Registro de Trazabilidad</i> incorporando <i>Componentes de Software</i> construidos.</p>	<p><i>Componentes de Software [corregidos]</i></p> <p><i>Registro de Trazabilidad [verificado, en línea base]</i></p>	<p><i>Registro de Trazabilidad [actualizado]</i></p>
PLD	<p>Incorporar <i>Componentes de Software</i> y <i>Registro de Trazabilidad</i> a la <i>Configuración del Software</i> y establecer en línea base.</p>	<p><i>Componentes de Software [corregidos]</i></p> <p><i>Registro de Trazabilidad [actualizado]</i></p>	<p><i>Configuración del Software</i></p> <ul style="list-style-type: none"> - <i>Componentes de Software [corregidos]</i> - <i>Registro de Trazabilidad [actualizado]</i>

TABLA B.29 - Productos de Trabajo de la actividad Construcción de Software

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto	Planeación del Proyecto
Documento de Diseño Técnico	Diseño Detallado de Software
Especificación de Requerimientos	Requerimientos
Documento de Diseño de Juego	Diseño de Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Motor de Juego (Game Engine)	Integración y Pruebas de Software
Componentes de Software	Integración y Pruebas de Software
Assets	Integración y Pruebas de Software
PRODUCTOS INTERNOS	
NOMBRE	
Herramientas (Tools)	
Casos de Pruebas Unitarias	

IV.6 - Integración y Pruebas de Software

IV.6.1 - Integración

En la actividad *Integración* se establecen tareas para integrar los *Assets* y *Componentes de Software* que se han desarrollado y posteriormente aplicar los *Escenarios* y *Casos de Pruebas* para verificar el correcto funcionamiento del *Software*.

TABLA B.30 - Tareas para la actividad de Integración

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
INTEGRACIÓN			
PLD	Asignar <i>Tareas</i> a los miembros del <i>Equipo de Trabajo</i> de acuerdo al rol y <i>Plan del Proyecto</i> actual.	<i>Plan del Proyecto</i> [<i>aceptado</i>] - <i>Tareas</i>	
PLD EPR	Integrar y construir el <i>Software (Game Build)</i> utilizando los <i>Componentes de Software</i> y <i>Assets</i> del videojuego de acuerdo a la <i>Línea de Producción</i> . Si es necesario actualizar los <i>Casos de Pruebas de Integración</i> . Construir versiones jugables del videojuego (alfa, beta, gold).	<i>Componentes de Software</i> [<i>corregidos</i>] <i>Especificación de Requerimientos</i> - <i>Línea de Producción</i> <i>Documento de Diseño Técnico</i> - <i>Casos de Pruebas de Integración</i> <i>Registro de Trazabilidad</i> [<i>en línea base</i>]	<i>Software (Game Build)</i> <i>Documento de Diseño Técnico</i> - <i>Casos de Pruebas de Integración</i> [<i>actualizado</i>]

ACL	<p>Realizar pruebas de <i>Software (Game Build)</i> usando los <i>Casos de Pruebas de Integración</i>.</p> <p>Generar <i>Reporte de Pruebas y Base de Datos de Errores</i>.</p>	<p><i>Software (Game Builds)</i></p> <p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Casos de Pruebas de Integración [actualizado]</i> 	<p><i>Software (Game Build) [probado]</i></p> <p><i>Reporte de Pruebas [actualizado]</i></p> <p><i>Base de Datos de Errores [actualizada]</i></p>
PAC	<p>Corregir los defectos encontrados hasta satisfacer criterio de término.</p>	<p><i>Software (Game Build) [probado]</i></p> <p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Casos de Pruebas de Integración [actualizado]</i> <p><i>Reporte de Pruebas</i></p> <p><i>Base de Datos de Errores</i></p>	<p><i>Software (Game Build) [corregido]</i></p> <p><i>Reporte de Pruebas [defectos eliminados]</i></p> <p><i>Base de Datos de Errores [actualizada]</i></p>
ACD	<p>Verificar y obtener aprobación del <i>Software (Game Build)</i> respecto al <i>Diseño Detallado del Software y Descripción de Arquitectura</i>. Para cambios se propone <i>Solicitud de Cambio</i>.</p>	<p><i>Software (Game Build) [corregido]</i></p> <p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Descripción de Arquitectura de Software</i> - <i>Diseño Detallado de Software</i> 	<p><i>Listas de Verificación</i></p> <p><i>Software (Game Build) [verificado]</i></p> <p><i>Solicitud de Cambio [propuesta]</i></p>
PLD	<p>Incorporar a <i>Configuración del Software, los Casos de Pruebas de Integración, Software (Game Build), Reporte de Pruebas y Base de Datos de Errores</i>.</p>	<p><i>Documento de Diseño Técnico</i></p> <ul style="list-style-type: none"> - <i>Casos de Pruebas de Integración [actualizado]</i> <p><i>Software (Game Build) [verificado]</i></p> <p><i>Reporte de Pruebas [defectos eliminados]</i></p> <p><i>Base de Datos de Errores [actualizada]</i></p>	<p><i>Configuración del Software</i></p> <ul style="list-style-type: none"> - <i>Documento de Diseño Técnico</i> - <i>Casos de Pruebas de Integración [actualizado]</i> - <i>Software (Game Build) [verificado]</i> - <i>Reporte de Pruebas [defectos eliminados]</i> - <i>Base de Datos de Errores [actualizada]</i>

TABLA B.31 - Productos de Trabajo de la actividad Integración

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto	Planeación del Proyecto
Componentes de Software [corregidos]	Construcción de Software
Especificación de Requerimientos	Requerimientos
Documento de Diseño Técnico	Diseño Técnico del Videojuego
Registro de Trazabilidad [en línea base]	Diseño Técnico del Videojuego
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Software (Game Build) [verificado]	Pruebas de Software
Casos de Pruebas de Integración [actualizados]	Pruebas de Software
Reporte de Pruebas [defectos eliminados]	Pruebas de Software
Base de Datos de Errores [actualizada]	Pruebas de Software
PRODUCTOS INTERNOS	
NOMBRE	
Listas de Verificación	
Solicitud de Cambio	
Configuración del Software	

IV.6.1 - Pruebas de Software

En la actividad *Pruebas de Software* se aplican los *Escenarios y Casos de Pruebas* para verificar el correcto funcionamiento del *Software (Game Builds)*.

TABLA B.32 - Tareas para la actividad Pruebas de Software

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
PRUEBAS DE SOFTWARE			
PLD	Asignar <i>Tareas</i> a los miembros del <i>Equipo de Trabajo</i> de acuerdo al rol y <i>Plan del Proyecto</i> actual.	<i>Plan de Proyecto [aceptado]</i> - <i>Tareas</i>	
ACD	Aplicar los <i>Escenarios y Casos de Pruebas de Usabilidad</i> Aplicar los <i>Escenarios y Casos de Pruebas de Jugabilidad</i> Aplicar <i>Pruebas de Publicación en Mercado</i> para verificar que el videojuego cumple con las políticas de publicación (si es necesario, o para videojuego final)	<i>Documento de Diseño Técnico</i> - <i>Escenarios y Casos de Pruebas de Usabilidad</i> - <i>Escenarios y Casos de Pruebas de Jugabilidad</i> - <i>Pruebas de Publicación en Mercado</i> <i>Software (Game Build) [verificado]</i>	<i>Documento de Diseño Técnico</i> - <i>Escenarios y Casos de Pruebas de Usabilidad [aplicados]</i> - <i>Escenarios y Casos de Pruebas de Jugabilidad [aplicados]</i> - <i>Pruebas de Publicación en Mercado [aplicados]</i> <i>Videos del Grupo Focal</i>

			<i>Reporte de Pruebas [actualizado]</i> <i>Base de Datos de Errores [actualizada]</i> <i>Gráfica de Trabajo Pendiente</i>
PAC	Corregir posibles defectos o dificultades encontrados en las pruebas de usabilidad, jugabilidad o de publicación hasta satisfacer criterios de cumplimiento.	<i>Reporte de Pruebas</i> <i>Base de Datos de Errores</i> <i>Gráfica de Trabajo Pendiente</i>	<i>Base de Datos de Errores [actualizada]</i> <i>Gráfica de Trabajo Pendiente [actualizada]</i> <i>Software (Game Builds) [corregido]</i>
PLD	Establecer <i>Configuración del Software</i> en línea base e incorporar al repositorio.	<i>Estrategia de Control de Versiones</i> <i>Configuración del Software</i>	<i>Repositorio del Proyecto</i>

TABLA B.33 - Productos de Trabajo de la actividad Pruebas de Software

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto	Planeación del Proyecto
Documento de Diseño Técnico	Diseño Detallado de Software
Software (Game Build) [verificado]	Integración
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Software (Game Builds) [corregido]	Integración
Base de Datos de Errores [actualizada]	Integración
Reporte de Pruebas [actualizado]	Integración
Videos del Grupo Focal	Integración
Gráfica de Trabajo Pendiente	Integración
PRODUCTOS INTERNOS	
NOMBRE	
Escenarios y Casos de Pruebas de Usabilidad [aplicados]	
Escenarios y Casos de Pruebas de Jugabilidad [aplicados]	
Pruebas de Publicación en Mercado [aplicados]	

IV. Fase de Posproducción

En la *Fase de Posproducción* se llevará a cabo la *Entrega de Producto* de acuerdo a los *Protocolos de Entrega* y se propone una actividad de *Mantenimiento del Videojuego* en donde se realizará de manera sistemática la modificación del software, de acuerdo a las peticiones de mantenimiento de *Cientes* o usuarios finales (*Jugadores*).

IV.7 - Entrega del Producto

En esta actividad se lleva a cabo la *Entrega del Producto* de acuerdo a los *Protocolos de Entrega* establecidos con el *Cliente*.

TABLA B.34 - Tareas para la actividad Entrega del Producto

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
ENTREGA DEL PRODUCTO			
PRO EQT	Asignar <i>Tareas</i> a los miembros del <i>Equipo de Trabajo</i> de acuerdo a cada rol, basado en el <i>Plan del Proyecto</i> actual.	<i>Plan del Proyecto</i> [revisado] - <i>Tareas</i>	
CMN	Realizar o actualizar el <i>Manual</i> del videojuego.	<i>Software (Gold Master Build)</i>	<i>Manual</i>
PRO CLI	Validar <i>Manual</i> del videojuego.	<i>Manual</i>	<i>Manual [validado]</i>
PLD	Crear el <i>Kit de Cierre</i> del juego.	<i>Configuración del Software</i>	<i>Kit de Cierre</i>
PRO	Llevar a cabo la entrega de acuerdo al <i>Protocolo de Entrega</i> .	<i>Configuración del Software</i> <i>Software (Gold Master Build)</i> . <i>Protocolos de Entrega</i>	<i>Kit de Cierre [entregado]</i> <i>Configuración del Software [entregada]</i>

TABLA B.35 - Productos de Trabajo de la actividad Entrega del Producto

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Plan del Proyecto	Planeación del Proyecto
Protocolos de Entrega	Planeación del Proyecto
Software (Gold Master Build)	Pruebas de Software
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Kit de Cierre [entregado]	Equipo de Trabajo Cliente
Configuración del Software [entregada]	Cliente
PRODUCTOS INTERNOS	
NOMBRE	
Manual	
Configuración del Software	

IV.8 - Mantenimiento del Videojuego

En la actividad *Mantenimiento del Videojuego* se realiza de manera sistemática la modificación del *Software*, de acuerdo a las peticiones de mantenimiento de *Clientes* o usuarios finales (jugadores).

TABLA B.36 - Tareas para la actividad Mantenimiento del Videojuego

ROL	TAREAS	PRODUCTOS DE ENTRADA	PRODUCTOS DE SALIDA
MANTENIMIENTO DEL VIDEOJUEGO			
CLI	<p>Realizar <i>Acuerdo de Mantenimiento</i> del videojuego.</p> <p>Describir el tipo de mantenimiento que se hará y el alcance que tendrá en cuanto a Plataformas/Dispositivos, servicios y tiempo.</p>	<p><i>Enunciado del Trabajo</i></p> <p><i>Software (Gold Master Build)</i></p>	<p><i>Acuerdo de Mantenimiento</i></p>
CLI	<p>Definir <i>Protocolo de Petición de Modificación</i> con base en el <i>Acuerdo de Mantenimiento</i>.</p>	<p><i>Acuerdo de Mantenimiento</i></p>	<p><i>Protocolo de Petición de Modificación</i></p>
PRO LAR	<p>Analizar, evaluar y priorizar la <i>Petición de Modificación</i>.</p> <p>Analizar y evaluar el impacto tecnológico, en costo y tiempo para determinar si la petición es aceptada o rechazada.</p> <p>Determinar el tipo de modificación que debe aplicarse, ya sea mantenimiento por error de funcionamiento, balanceo del juego, actualización, expansión, contenido descargable (DLC), etc.</p>	<p><i>Petición de Modificación [solicitada]</i></p>	<p><i>Petición de Modificación [priorizada]</i></p>
PRO LAR	<p>Analizar la petición y determinar las <i>Acciones de Modificación</i> a seguir.</p>	<p><i>Petición de Modificación [priorizada]</i></p>	<p><i>Acciones de Modificación</i></p> <p><i>Elementos a modificar en la Configuración de Software</i></p>
EQT	<p>Ejecutar las modificaciones a la <i>Configuración de Software</i> con base en las <i>Acciones de Modificación</i> establecidas.</p> <p>Identificar los <i>Componentes de Software</i> que fueron afectados.</p>	<p><i>Elementos a modificar en la Configuración de Software</i></p>	<p><i>Componentes de Software [modificados]</i></p>
ACL ACD	<p>Ejecutar pruebas unitarias a los <i>Assets</i> y <i>Componentes de Software</i> que fueron afectados. Crear <i>Casos de Pruebas Unitarias</i> y aplicarlos para verificar que los <i>Componentes de Software</i> cumplen con la nueva funcionalidad establecida.</p>	<p><i>Componentes de Software [modificados]</i></p> <p><i>Casos de Pruebas Unitarias</i></p>	<p><i>Componentes de Software [probados]</i></p>

PAC	Corregir defectos encontrados hasta lograr pruebas exitosas.	<i>Componentes de Software [probados]</i>	<i>Componentes de Software [corregidos]</i>
EPR	Integrar nuevamente los <i>Componentes de Software</i> .	<i>Componentes de Software [corregidos]</i>	<i>Software (Game Build)</i>
ACL ACD	Realizar <i>Pruebas de Software</i> usando los <i>Escenarios y Casos de Pruebas</i> .	<i>Software (Game Build)</i> <i>Escenarios y Casos de Pruebas</i>	<i>Software [probado]</i> <i>Reporte de Pruebas</i>
PRO CLI	Verificar y obtener la aprobación del <i>Software</i> respecto a la <i>Petición de Modificación</i> .	<i>Software [corregido]</i> <i>Petición de Modificación</i>	<i>Listas de Verificación</i> <i>Software [verificado]</i>
PRO CLI	Validar y obtener aprobación del <i>Software</i> por parte del <i>Cliente</i> .	<i>Software [verificado]</i> <i>Petición de Modificación</i>	<i>Listas de Validación</i> <i>Software [validado]</i>
PLD	Establecer nueva <i>Configuración del Software</i> en línea base e incorporar al repositorio la nueva versión.	<i>Estrategia de Control de Versiones</i> <i>Configuración del Software</i>	<i>Repositorio del Proyecto</i>
PLD	Llevar a cabo el reemplazo del <i>Software (Game Build)</i> y realizar su distribución.	<i>Software [validado]</i> <i>Políticas de Publicación en Mercados (si es necesario).</i>	<i>Software [disponible, modificado]</i>
PRO CLI	Formalizar conclusión del mantenimiento del videojuego con base en el <i>Acuerdo de Mantenimiento</i> .	<i>Acuerdo de Mantenimiento</i>	<i>Documento de Terminación de Mantenimiento [aceptado]</i>

TABLA B.37 - Productos de Trabajo de la actividad Mantenimiento del Videojuego

PRODUCTOS DE ENTRADA	
NOMBRE	ORIGEN
Enunciado del Trabajo	Cliente
Software (Gold Master Build)	Cliente
Petición de Modificación [solicitada]	Cliente
PRODUCTOS DE SALIDA	
NOMBRE	DESTINO
Acuerdo de Mantenimiento	Cliente
Protocolo de Petición de Modificación	Cliente
Software (Gold Master Build) [disponible, modificado]	Cliente
Documento de Término de Mantenimiento [aceptado]	Cliente
PRODUCTOS INTERNOS	
NOMBRE	
Petición de Modificación [priorizada]	
Acciones de Modificación	
Elementos a modificar en la Configuración de Software	
Componentes de Software [modificados]	
Casos de Pruebas Unitarias	
Escenarios y Casos de Pruebas	

Reporte de Pruebas
Listas de Verificación
Listas de Validación
Software (Game Build) [probado]
Estrategia de Control de Versiones
Repositorio del Proyecto
Configuración del Software

Descripción de productos

Esta es una lista alfabética de los productos de entrada, salida e internos de las actividades y tareas de los procesos, sus descripciones y estados posibles.

TABLA B.38 - Descripción de productos

NOMBRE	DESCRIPCIÓN
ACCIONES CORRECTIVAS	Descripción de las acciones que se llevarán a cabo para corregir desvíos, incidencias o problemas dentro del proyecto de desarrollo.
ACCIONES DE MODIFICACIÓN	Descripción de las acciones que se llevarán a cabo para dar solución a las modificaciones solicitadas.
ACUERDO DE MANTENIMIENTO	<p>Descripción del mantenimiento a realizar para el videojuego. Puede incluir:</p> <ul style="list-style-type: none"> - Alcances y tipo de mantenimiento (corrección de errores, balanceo de mecánicas, contenido descargable, actualizaciones, etc.) - Plataformas/Dispositivos en los que se dará mantenimiento al videojuego. - Tiempo en el que se dará mantenimiento. <p>El estado aplicable es: revisado.</p>
ANÁLISIS COMPETITIVO	Este documento contiene las fortalezas y debilidades de la competencia del videojuego que se quiere construir.
ANÁLISIS FODA	Contiene las fortalezas y debilidades del videojuego, oportunidades de mercado y amenazas que podrían impactar el éxito del videojuego.
ARCHIVO FINAL	Es el archivo que se utilizará para manufacturar el juego. Incluye todos los archivos utilizados incluyendo la compilación (<i>Build</i>) final del juego. El <i>Publisher</i> requiere este archivo en caso de que haya un problema con el juego, o si una secuela es hecha por otro estudio.
ARCHIVOS DE REFERENCIA	<p>Este documento contiene cualquier tipo de referencia que pueda ser utilizado para crear los diferentes <i>Assets</i> del videojuego. La referencia más común son las fotografías e imágenes utilizadas por los artistas. Los videos también son utilizados para ilustrar movimientos y acciones de personajes y objetos.</p> <p>Otro tipo de referencias comunes son: libros, objetos físicos, sonidos, efectos, música, voces, modelos y ambientes 3D creados en juegos anteriores.</p>
ARTE CONCEPTUAL	<p>Es una representación visual en 2D de un elemento del juego que puede ser ajustado y perfeccionado antes de ser construido en el juego. Generalmente se ve más como una ilustración que una imagen computarizada. Estos documentos son utilizados también en la publicidad del juego.</p> <p>Los artistas conceptuales crean estos archivos basados en las ideas del diseñador del juego utilizando algún editor de imágenes.</p>

ASSETS	Los <i>Assets</i> son objetos o elementos audiovisuales que integran un videojuego y deben estar en un formato que pueda soportar el <i>Motor de Juego</i> . Existen diferentes tipos de <i>Assets</i> , por ejemplo: <i>Assets de Arte</i> (texturas, modelos 3D, animaciones), <i>Assets de Texto</i> , <i>Assets de Audio</i> (Efectos de sonido, música, diálogos, voces) Cinemáticas.
BASE DE DATOS DE ERRORES	Base de datos donde se almacenan los errores encontrados en el juego. Los errores podrían clasificarse para identificar la prioridad en la corrección de los mismos.
BIBLIA DE ARTE GUÍA DE ESTILO DE ARTE ART BIBLE	Contiene las guías para ayudar a mantener la apariencia del videojuego consistente. Es utilizado para comunicar a los artistas y animadores como se debería ver el juego y como lograrlo. El estilo de personajes, objetos y ambientes también son incluidos en este documento. Imágenes de referencia son incluidas para ayudar a comunicar elementos de estilo específicos. También podría incluir instrucciones sobre cómo utilizar el software de diseño para crear varios estilos, desde la creación de texturas, iluminación, número de polígonos, etc.
BIBLIA DEL MUNDO MUNDO DEL JUEGO WORLD BIBLE	En este documento se establecen las reglas de consistencia para los escenarios del juego. Estas reglas ayudan a los artistas, animadores, diseñadores de niveles y programadores a entender cómo crear <i>Assets</i> del juego y el tipo de física permitida en el juego. Este documento se crea con mayor frecuencia en juegos de largo alcance, como los juegos multijugador en línea masivos.
BUILD NOTES	Son las notas incluidas sobre la construcción del juego (<i>Game Builds</i>), indican que elementos fueron incluidos y los requerimientos técnicos para poder correr el juego en la <i>Plataforma</i> elegida.
CALENDARIO DE FLUJO DE FONDOS	Este documento es incluido con el <i>Documento de Venta (Pitch)</i> . Esboza cuanto gente se necesitará para trabajar en un proyecto potencial, cuando se necesitarán y los costos asociados. Incluye otros costos como licencias y <i>outsourcing</i> .
CALENDARIO DE TAREAS	Estimación de tiempo en calendario para realizar las <i>Tareas</i> del proyecto. El estado aplicable es: actualizado.
CAPTURAS DE PANTALLA DEL JUEGO	Estas imágenes muestran el progreso del juego y cómo luce actualmente, así como los componentes específicos que se han creado. El video también puede ser utilizado para mostrar algún movimiento particular o algún elemento de la jugabilidad que no pudiera demostrarse mediante imágenes.
CASOS DE PRUEBAS DE INTEGRACIÓN	Elementos necesarios para probar el funcionamiento del código después de integrar uno o más componentes de software. El estado aplicable es: aplicados.
CASOS DE PRUEBAS UNITARIAS	Elementos necesarios para probar el funcionamiento del código de un componente de software. El estado aplicable es: aplicados.
COMPONENTES DE SOFTWARE	Es una <i>Unidad de Software</i> independiente que puede organizarse con otros componentes para crear un sistema de <i>Software</i> . Se consideran la unidad fundamental de composición en un sistema.
COMPOSICIÓN DEL EQUIPO DE TRABAJO	Organización del <i>Equipo de Trabajo</i> , dependiendo del tamaño del estudio y los roles.
CONCEPTO INICIAL	Presenta la idea general del videojuego, motivación, objetivo principal, género y plataforma de juego. Proporciona las bases para todas las decisiones del videojuego.

CONFIGURACIÓN DE SOFTWARE	<p>Un conjunto de productos de software identificados de forma única y consistente. Los elementos de <i>Configuración de Software</i> incluyen:</p> <ul style="list-style-type: none"> - Especificación de Requerimientos. - Código Fuente - Ejecutables - Especificación de Desarrollo. - Escenarios y Casos de Pruebas Globales. - Reporte de Pruebas.
CONTRATO Y ENMIENDAS	<p>El <i>Contrato</i> establece las responsabilidades del <i>Publisher</i> y desarrollador, y los términos en los que ambos están dispuestos a trabajar juntos.</p>
DESCRIPCIÓN DE ARQUITECTURA DE SOFTWARE	<p>Describe la estructura global del Software y puede incluir:</p> <ul style="list-style-type: none"> - <i>Componentes de Software</i> requeridos - Relaciones entre <i>Componentes de Software</i> - Descripción general de la <i>Interfaz de Usuario</i> - Riesgos de la <i>Arquitectura</i> - Consideraciones requeridas - Elementos diferentes a <i>Componentes de Software</i> que la pueden integrar - Otros
DIAGRAMA DE FLUJO DE INTERFAZ DE USUARIO	<p>Un <i>Diagrama de Flujo de Interfaz de Usuario</i> se centra en los menús vistos en pantalla por el jugador durante el juego. Describe que pantallas se necesitan y en qué orden deben aparecer.</p>
DIAGRAMA DE FLUJO DEL JUEGO	<p>Muestra la progresión a través del videojuego desde el punto de vista de la interacción del jugador. El diagrama se mueve de izquierda a derecha, documentando los diferentes niveles que un jugador encontrará y en el orden en el que aparecerán, los objetivos del jugador en cada nivel, poderes y habilidades del jugador en cada escenario, narrativas clave; y cualquier cinemática que sea necesaria para explicar la historia.</p> <p>También se incluyen estimaciones del tiempo que le tomará a un jugador jugar a través del nivel y estimaciones de los niveles de emoción y dificultad en cualquier punto del juego. Si el juego requiere granularidad, cada diagrama de flujo deberá realizarse por nivel.</p>
DISEÑO DE NIVELES	<p>Incluye el plan de lo que el jugador hará en cada nivel y como progresará a través del juego. Puede incluir:</p> <ul style="list-style-type: none"> - Mecánicas relativas al escenario - Mecánicas de los enemigos - Objetivos - Resolución de acertijos - Estrategias y dinámicas - Eventos - Jefes finales - Puntos de guardado - Recolección de premios - Rutas a través de un escenario - Arte conceptual <p>Mientras que la <i>Biblia de Arte</i> y la <i>Biblia del Mundo</i> se enfocan en cómo se verá y se moverá un enemigo, este documento describe cómo y dónde el jugador enfrentará al enemigo.</p>

DISEÑO DETALLADO DEL SOFTWARE	<p>Incluye detalle de los <i>Componentes de Software</i> para facilitar su construcción puede incluir:</p> <ul style="list-style-type: none"> - Patrones de diseño a emplear - Estructuras de datos a utilizar - Descripción de funcionamientos - Protocolos de conexión - Otros
DOCUMENTO DE ACEPTACIÓN	<p>Documentación de la aceptación de los <i>Entregables</i> por parte del cliente. Puede tener las siguientes características:</p> <ul style="list-style-type: none"> - Registro de la recepción de la entrega - Fecha de recepción - Registro de los criterios de aceptación del cliente - Registro de asuntos pendientes - Firma de recibido y conformidad del cliente
DOCUMENTO DE CONCEPTO DE JUEGO HIGH CONCEPT	<p>Expresa la idea central del videojuego. Es un documento de una o dos páginas que es necesariamente breve y sencillo con el objetivo de fomentar el flujo de ideas. Puede contener:</p> <ul style="list-style-type: none"> - Género - Plataforma - Estatus del Juego - Número de jugadores - Público Objetivo - Motivación - Declaración de Misión - Ambientación del Juego - Resumen de la Historia - Modelo de Monetización - Jugabilidad (Gameplay) - Mecánicas de Juego - Arte Conceptual - Elementos de Audio - Modos de Distribución
DOCUMENTO DE DISEÑO DE JUEGO GAME DESIGN DOCUMENT	<p>Establece las características generales de estilo, historia y jugabilidad. También servirá como guía para que todas las áreas involucradas tengan claridad y certeza de lo que deben hacer tanto ellos como las demás áreas. Puede incluir:</p> <ul style="list-style-type: none"> - Información General del Juego - Historia del Juego - Resumen Narrativo - Lista de Eventos Clave en la Historia - Guión Literario - Biblia del Mundo - Lista de Personajes - Jugabilidad (Gameplay) - Mecánicas de Juego - Control - Diseño de Niveles - Planos de Niveles - Biblia de Arte (Guía de Estilo de Arte.) - Arte Conceptual

	<ul style="list-style-type: none"> - Interfaz de Usuario - Diagrama de Flujo de Interfaz de Usuario - Diagrama de Flujo del Juego - Inteligencia Artificial - Logros - Plan de Elementos de Audio <p>El <i>Documento de Diseño de Juego</i>, es uno de los documentos esenciales en el proceso de producción del videojuego. Dependiendo del alcance y tipo de videojuego, el contenido de este documento podría separarse en otros documentos. Por ejemplo, <i>Biblia del Mundo</i> o <i>Historia del Juego</i>.</p>
DOCUMENTO DE DISEÑO TÉCNICO TECHNICAL DESIGN DOCUMENT	<p>Es una descripción de los aspectos técnicos de la producción del juego. Puede incluir:</p> <ul style="list-style-type: none"> - Especificación Técnica de Desarrollo - Descripción de Arquitectura de Software - Diseño Detallado de Software - Casos de Pruebas de Integración - Escenarios y Casos de Pruebas de Usabilidad - Escenarios y Casos de Pruebas de Jugabilidad - Pruebas de Publicación en Mercados - Línea de Producción de Assets <p>El documento es actualizado en diferentes etapas de la producción del juego.</p>
DOCUMENTO DE EVALUACIÓN DE DESARROLLADORES	<p>Contiene una serie de preguntas para evaluar a los desarrolladores e información del estudio sobre su organización y procesos de desarrollo. Es requerido por los <i>Publishers</i> antes de financiar un proyecto.</p>
DOCUMENTO DE SEGUIMIENTO DE ASSETS	<p>Contiene una lista con todos los <i>Assets</i> y las fechas cuando deben de ser enviados al <i>Publisher</i>. Así como el registro de los assets que ya fueron enviados y aprobados.</p> <p>Este documento se incluye en cada hito y es enviado de ida y vuelta entre el estudio y el <i>Publisher</i> conforme nuevos assets son aprobados. Algunas veces son regresados con comentarios sobre los ajustes que necesitan realizarse.</p>
DOCUMENTO DE VENTA PITCH	<p>Este documento es utilizado para vender la idea del juego a un <i>Publisher</i> u otra fuente de financiamiento, con el objetivo de ser contratados u obtener los recursos necesarios para desarrollar el videojuego.</p>
DURACIÓN ESTIMADA	<p>Duración estimada en tiempo total en la que se realizarán las <i>Tareas</i> del proyecto.</p>
ENTREGABLES	<p>Lista de productos a entregar al <i>Cliente</i>.</p>
ENUNCIADO DEL TRABAJO	<p>El <i>Enunciado del Trabajo</i> es una descripción narrativa de los productos, servicios o resultados que debe entregar el <i>Proyecto</i>. En el caso de proyectos internos, el iniciador del proyecto o patrocinador proporciona el <i>Enunciado del Trabajo</i> sobre la base de las necesidades del estudio o de los requisitos del producto o servicio.</p> <p>En el caso de proyectos externos, el <i>Enunciado del Trabajo</i> puede ser proporcionado por el <i>Cliente</i> como parte de un documento de licitación (p.ej., una solicitud de propuesta, una solicitud de información, o una solicitud de oferta), o como parte de un contrato.</p>

ESCENARIOS Y CASOS DE PRUEBAS DE JUGABILIDAD	<p>Escenarios de prueba creados para realizar las pruebas de jugabilidad con usuarios reales. Describe tareas a realizar bajo circunstancias específicas.</p> <p>Contiene cuestionarios aplicables a los usuarios para detectar los problemas con la jugabilidad.</p>
ESCENARIOS Y CASOS DE PRUEBAS DE USABILIDAD	<p>Escenarios de prueba creados para realizar las pruebas de usabilidad con usuarios reales. Describe tareas a realizar bajo circunstancias específicas.</p> <p>Contiene cuestionarios aplicables a los usuarios para detectar los problemas con la usabilidad.</p>
ESFUERZO Y COSTO	Estimación del <i>Esfuerzo y Costo</i> que tendrá el realizar las <i>Tareas</i> del Proyecto.
ESPECIFICACIÓN DE REQUERIMIENTOS	<p>Identifica los diferentes tipos de <i>Requerimientos</i> del videojuego. Este puede tener las siguientes características:</p> <p>Introducción (descripción general del <i>Software</i>) Descripción de requerimientos:</p> <ul style="list-style-type: none"> - Funcionalidad - Interfaz de Usuario - Interfaces externas - Fiabilidad - Eficiencia - Mantenimiento - Portabilidad - Limitaciones/restricciones del diseño y construcción - Interoperabilidad - Reutilización - Legales y regulativos <p>Cada requerimiento es identificado, único y es verificable o susceptible de evaluación. Los estados aplicables son: verificado, validado e incorporado en línea base.</p>
ESPECIFICACIÓN TÉCNICA DE DESARROLLO	Describe el ambiente de desarrollo a utilizar para una plataforma o dispositivo en particular.
ESTÁNDARES DE PLATAFORMA CONSOLE VENDOR STANDARDS	Contiene los estándares en el diseño del juego que se requieren implementar para una plataforma específica. Estos estándares incluyen como deben de funcionar los controles y qué títulos deben aparecer al principio del juego.
ESTRATEGIA DE CONTROL DE VERSIONES	Describe la manera en que se llevara a cabo el <i>Control de Versiones</i> tanto de la documentación como del <i>Software</i> .
GAME BUILD	Es una versión jugable del videojuego en la <i>Plataforma</i> elegida. Son requeridas usualmente en cada hito, de tal forma que se pueda ver el progreso realizado.
GENERACIÓN DE IDEA	Descripción borrador sobre la idea inicial que se tiene para desarrollar un videojuego.
GOLD MASTER CANDIDATE BUILD	Es la versión final (<i>Game Build</i>) del videojuego que es enviada para su aprobación. Contiene el archivo del juego completo. Una vez que el juego es aprobado para su venta, las copias del juego se producirán y distribuirán.
GRÁFICA DE TRABAJO PENDIENTE	El propósito de la Gráfica de Trabajo Pendiente es monitorear en cuantos errores está trabajando cada programador, y estimar la fecha cuando todos los errores serán corregidos.

GUIÓN LITERARIO	Este documento contiene los diálogos para todas las partes del juego, incluyendo cinemáticas y jugabilidad. Las voces de los personajes son grabadas después utilizando este <i>Guión Literario</i> .
HEAD-UP DISPLAY (HUD)	El <i>Head-up Display (HUD)</i> es la información que aparece visible en todo momento en pantalla como: puntos, vidas, nivel de energía actual, tiempo, mapa de nivel, radares, accesorios, etc. Los elementos que se muestran en el <i>HUD</i> varían según el tipo de videojuego. Es un elemento esencial de la <i>Interfaz de Usuario</i> del videojuego.
HISTORIA DEL JUEGO	Contiene la historia principal del juego, guión y perfiles de los personajes. Contiene una descripción en prosa de héroes y villanos (si los hubiera), del punto desencadenante de la historia y de los objetivos a lograr para resolver la historia.
IDENTIFICACIÓN DE RIESGOS	Identifica y describe los posibles <i>Riesgos</i> así como su prevención, mitigación y postergado.
INTERFAZ DE USUARIO IU	La <i>Interfaz de Usuario</i> se compone de: pantallas, <i>HUD</i> , botones, pantallas de alerta, personajes que emergen en pantalla para darnos pistas, pantallas de información, la tipografía que se utilizará, la señalización, la ubicación de los elementos, los colores y las formas entre otros.
INVESTIGACIÓN PRELIMINAR	Contiene una evaluación de videojuegos similares que estén publicados, aspectos legales relacionados con derecho de autor, licencias, etc. así como una evaluación de las políticas de publicación en mercados potenciales. El objetivo es probar que hay un mercado potencial para el videojuego que se quiere construir.
KIT DE CIERRE	El <i>Kit de Cierre</i> se crea cuando el código del juego es liberado de modo que los desarrolladores puedan incluir los <i>Assets</i> finales y el código fuente. Si alguna actualización o parches se crean después del <i>Kit de Cierre</i> , se podrían agregar después. El contenido de un <i>Kit de Cierre</i> incluye: <ul style="list-style-type: none"> - <i>Assets</i> - <i>Herramientas (Tools)</i> - <i>Código del juego</i> - Documentación generada durante el desarrollo del videojuego - Guía técnica (para integrar <i>Assets</i>, convertir <i>Assets</i> a los formatos específicos del juego, especificaciones de hardware y software) - Información general del producto.
LECCIONES APRENDIDAS	Es lo que se asimila durante un proyecto y estas enseñanzas pueden ser identificadas en cualquier momento del proyecto. Para que éstas queden aprendidas han de registrarse como una base de conocimiento para que pueda ser revisada y estudiada en ocasiones futuras.
LIMITACIÓN DEL SISTEMA	Aclara ciertos límites que no deberían ser pasados para cumplir con los requisitos de desempeño del videojuego, por ejemplo el número de polígonos, número de mensajes enviados por segundo, número simultáneo de explosiones en pantalla, etc.
LÍNEA DE PRODUCCIÓN PRODUCTION PIPELINE	La <i>Línea de Producción</i> se refiere a los pasos necesarios para que los <i>Assets</i> y <i>Componentes de Software</i> se integren y funcionen correctamente para crear una <i>Versión Jugable</i> del videojuego. Es generada por los <i>Programadores</i> explícitamente para los <i>Artistas</i> . Especifica lo que los <i>Artistas</i> deben y no deben hacer para que los <i>Assets</i> de arte aparezcan correctamente en el <i>Motor de Juego</i> .

LISTA DE EVENTOS CLAVE EN LA HISTORIA	<p>Contiene la lista de eventos clave en la historia que indican la necesidad de incluir algún elemento de <i>Jugabilidad</i>, diálogos, cinemáticas o introducen un nuevo nivel en el juego.</p> <p>Este documento le permite al desarrollador comenzar a visualizar el flujo del juego. También es utilizado para crear el <i>Guión Literario</i>.</p>
LISTA DE PERSONAJES	<p>Contiene la lista de todos los personajes que aparecen en el juego. Este documento incluye la descripción de cada personaje, habilidades, accesorios, etc. Describe dónde y cuándo aparece cada personaje en el juego, y todos los riesgos asociados con la creación del personaje.</p>
LISTAS DE VALIDACIÓN	<p>Documenta la ejecución de la validación y puede incluir:</p> <ul style="list-style-type: none"> - Participantes - Fecha - Elementos aprobados - Elementos no aprobados - Elementos pendientes de validar - Defectos identificados durante la validación - Comentarios o Cambios
LISTAS DE VERIFICACIÓN	<p>Documenta la ejecución de la verificación y puede incluir:</p> <ul style="list-style-type: none"> - Participantes - Fecha - Elementos aprobados - Elementos no aprobados - Elementos pendientes de verificar - Defectos identificados durante la verificación - Comentarios o Cambios
MANUAL	<p>Contiene información sobre la historia, personajes, jugabilidad, control, configuración, modos de juego, multijugador, etc. El diseño final del manual a veces es determinado por el Publisher.</p>
MEJORAS	<p>Documento que presenta las propuestas de mejora para el <i>Equipo de Trabajo</i>.</p>
MINUTA	<p>Registra los acuerdos establecidos con el <i>Publisher</i> y/o <i>Equipo de Trabajo</i>. Puede tener las siguientes características:</p> <ul style="list-style-type: none"> - Propósito de la reunión - Asistentes - Fecha y lugar - Referencias a minutas de reuniones previas - Qué fue logrado - Identifica cuestiones planteadas - Acuerdos - Próxima reunión
MODELO DE FINANCIAMIENTO	<p>Describe la forma en la que se van a obtener el dinero para producir el videojuego. Algunos modelos de financiamiento para videojuegos independientes pueden ser:</p> <ul style="list-style-type: none"> - <i>Crowdfunding</i> - Aporte No Reembolsable - Subsidios y Créditos - Incubadoras y aceleradoras

MODELOS	<p>Son los modelos tridimensionales de objetos o personajes creados por <i>Artistas</i> utilizando herramientas de modelado.</p> <p>Antes de que un <i>Asset</i> de arte pueda ser incluido en el juego, debe ser creado fuera del <i>Motor de Juego</i>, una vez que los <i>Modelos</i> son terminados, deberán ser exportados al <i>Motor de Juego</i>.</p>
MOTOR DE JUEGO GAME ENGINE	<p>Es un sistema diseñado para la creación de videojuegos que aglutina un conjunto de aplicaciones necesarias para su desarrollo. Su función principal es dotar al juego de un <i>Motor Gráfico</i> para el <i>rendering</i> de los <i>Modelos</i> y <i>Animaciones</i> que forman el videojuego, aunque a menudo los motores incorporan un entorno de desarrollo formado por varias herramientas para facilitar a los desarrolladores el trabajo, como un <i>Motor de Física</i> o un <i>Motor de Colisiones</i>.</p> <p>Desarrollar un motor eficiente necesita de una gran dedicación y capacidad de desarrollo, por ello a menudo los <i>Estudios Independientes</i> adquieren licencias para usar en sus videojuegos los motores creados por otros, y sólo los estudios de desarrollo más grandes pueden contar con su propio motor.</p>
PETICIÓN DE MODIFICACIÓN	<p>Petición realizada para llevar a cabo una modificación de funcionamiento o evolución en la aplicación móvil. Puede incluir:</p> <ul style="list-style-type: none"> - Origen de la Petición de Modificación - Descripción del problema de modificación a realizar
PLAN DE CONTRATACIÓN	<p>El Plan de Contratación describe que áreas necesitan más contrataciones y en qué etapa de la producción del videojuego esas personas serán contratadas.</p>
PLAN DE LIBERACIÓN	<p>Presenta como se va a distribuir y difundir el videojuego. Puede incluir:</p> <ul style="list-style-type: none"> - Medios publicitarios a utilizar - Posibles fechas de publicación - Costo de lanzamiento de la aplicación
PLAN DEL PROYECTO	<p>Presenta como serán ejecutados los procesos y actividades del proyecto para asegurar su conclusión exitosa, así como la calidad de los productos entregables. Puede incluir lo siguiente:</p> <ul style="list-style-type: none"> - Descripción de producto. - <i>Alcance</i> (descripción respecto a lo que está incluido). - <i>Objetivos</i> del proyecto. - <i>Entregables</i> (lista de productos a ser entregados). - <i>Tareas</i> a realizar. - Duración estimada de las <i>Tareas</i>. - <i>Recursos</i> (humanos, materiales, estándares, equipos, herramientas). - Estructura del <i>Equipo de Trabajo</i>. - <i>Calendario de las Tareas</i> del proyecto. - <i>Esfuerzo</i> y el <i>Costo Estimado</i>. - Identificación de los <i>Riesgos</i> del proyecto. - Estrategia para el <i>Control de Versiones</i>. - <i>Protocolos de Entrega</i> y verificación. <p>Los estados aplicables son: verificado, aceptado, actualizado y revisado.</p>

PLANOS DE NIVELES	Los <i>Planos de Niveles</i> , van de la mano con el <i>Diseño de Niveles</i> . Estas son las descripciones gráficas de los niveles realizados por los diseñadores. Los diseños sirven como mapas de los ambientes, ilustrando ítems y áreas que un jugador encontrará.
POLÍTICAS DE PUBLICACIÓN DE MERCADOS	Políticas en la que un Mercado de Aplicaciones describe los requerimientos y procesos necesarios para publicar un videojuego. Algunas tiendas de aplicaciones en las que se pueden distribuir videojuegos independientes son las siguientes: <ul style="list-style-type: none"> - Appstore (iOS) - Google Play (Android) - MarketPlace (Windows Phone) - Xbox Live Arcade (XBOX) - Steam (PC)
POST MORTEM	Es una oportunidad para que el estudio reflexione acerca de la reciente producción del videojuego, las cosas que salieron mal y las que salieron bien. También incluye algunas posibles soluciones para las cosas que salieron mal.
PROTOCOLO DE PETICIÓN DE MODIFICACIÓN	Requisitos para solicitar las peticiones de modificación por parte del <i>Cliente</i> y <i>Equipo de Trabajo</i> . Puede incluir: <ul style="list-style-type: none"> - Responsables de enviar/recibir las peticiones de modificación - Forma de comunicar las peticiones de modificación
PROTOCOLOS DE ENTREGA	Conjunto de instrucciones y procesos de entrega de documentos o <i>Software</i> para validación por parte del <i>Cliente</i> o <i>Publisher</i> .
PROTOCOLOS DE VERIFICACIÓN	Conjunto de instrucciones y procesos de verificación de documentos o software dentro del <i>Equipo de Trabajo</i> .
PROTOTIPO	Un <i>Prototipo</i> es una versión simplificada del videojuego en la plataforma especificada que es utilizada para explorar una cuestión particular sobre las <i>Mecánicas del Juego</i> , movimientos de vehículos, ángulos de cámaras, sistemas de combate, etc. Lo más importante del prototipo es que permita probar rápidamente los elementos que quieren explorarse, por lo que no se requiere mucho trabajo en el aspecto visual. Podría tener interfaces monocromáticas y versiones simples de los objetos y ambientes que eventualmente llenarán los espacios.
PROTOTIPO DEL VIDEOJUEGO	Es una <i>Versión Jugable</i> del videojuego que el <i>Estudio Independiente</i> podría enviar junto con el <i>Documento de Venta (Pitch)</i> . También es utilizado como una forma de mostrar las habilidades de los desarrolladores. El diseño es creado e implementado por los <i>Diseñadores</i> , <i>Artistas</i> y <i>Programadores</i> , y probado por los <i>Aseguradores de la calidad</i> . El <i>Prototipo del Videojuego</i> será jugable en la plataforma en la cual el juego será eventualmente lanzado.
PRUEBAS DE PUBLICACIÓN EN MERCADO	Conjunto de pruebas diseñadas con base en las restricciones en las <i>Políticas de Publicación</i> en las diversas tiendas de videojuegos. Tienen el objetivo de evitar lo más posible que el videojuego sea rechazado al ser enviada a la tienda de videojuegos. Puede Incluir: <ul style="list-style-type: none"> - Plataforma/Dispositivo en el que se realiza la prueba - Tienda de aplicaciones al que será enviado - Pruebas realizadas - Reporte de resultados

RECURSOS	<p>Descripción de recursos necesarios para llevar a cabo el proyecto. Puede incluir:</p> <ul style="list-style-type: none"> - Recursos humanos - Recursos materiales y tecnológicos - Recursos sobre servicios y elementos proporcionados por terceros
REGISTRO DE TRAZABILIDAD	<p>Documenta la relación entre los <i>Requerimientos</i> incluidos en la <i>Especificación de Requerimientos</i>, los elementos de la <i>Descripción de la Arquitectura de Software</i>, <i>Diseño Detallado de Software</i>, los <i>Componentes de Software</i>, <i>Escenarios</i> y <i>Casos de Prueba</i>. Puede incluir:</p> <ul style="list-style-type: none"> - Especificación de los <i>Requerimientos</i> por rastrear - Proporciona el mapeo (hacia adelante y hacia atrás)
REPORTE DE AVANCE	<p>Registra el estado del proyecto contra el <i>Plan del Proyecto</i>. Describe el estado y avance de las tareas generales concluidas hasta el momento respecto al <i>Plan del Proyecto</i>.</p>
REPORTE DE PRUEBAS	<p>Documenta la ejecución de las pruebas particulares. Puede incluir:</p> <ul style="list-style-type: none"> - Caso/Escenario de prueba en cuestión - Quién aplica el caso de prueba - Defectos encontrados - Resumen de defectos - Fecha de realización de prueba - Quién y cómo soluciona defecto
REPOSITORIO DEL PROYECTO	<p>Contenedor electrónico para almacenar los <i>Productos de Trabajo</i> y <i>Entregables</i> del proyecto (documentos o software). Puede tener las siguientes características:</p> <ul style="list-style-type: none"> - Almacena los productos de trabajo del proyecto - Almacena los productos de trabajo ya liberados - Capacidades de almacenamiento y recuperación - Facilidad para navegar su contenido - Enlista los contenidos y la descripción de los atributos - Comparte y transfiere productos de trabajo entre los grupos involucrados - Recuperación de las versiones anteriores de los productos de trabajo <p>Los estados aplicables son: recuperado y actualizado</p>
RESPALDO DEL REPOSITORIO DEL PROYECTO	<p>Contiene un respaldo del <i>Repositorio del Proyecto</i>, realizado de acuerdo a la <i>Estrategia de Control de Versiones</i>.</p>
RESUMEN NARRATIVO	<p>Describe lo que pasa en el juego y a quien le pasa. La acción y las motivaciones de los personajes. Está escrito en formato narrativo y se lee como una historia corta. Los diálogos no son incluidos en este documento.</p> <p>Es un documento corto que describe los escenarios importantes, personajes y acciones que se llevarán a cabo en el videojuego.</p>
RETROALIMENTACIÓN DEL HITO	<p>Contiene la retroalimentación por parte del <i>Publisher</i> de los <i>Entregables</i>, incluso podría incluir instrucciones para mejorar diferentes <i>Assets</i>.</p>
SCRIPTS	<p>Los <i>Scripts</i> son escritos en el lenguaje de programación específico del <i>Motor del Juego</i>. Son incorporados directamente en el motor, y son incluidos junto con los demás <i>Assets</i> cuando el juego es construido desde el <i>Motor de Juego</i>.</p>

SOFTWARE	<p>Elemento de Software (código fuente y código ejecutable) entregable y visible para un <i>Cliente</i>, integrado por un conjunto de <i>Componentes de Software</i>.</p> <p>El Software es una solución computacional específica, que implementa cierta <i>Arquitectura</i> y logra un conjunto de comportamientos esperados en una plataforma, para una aplicación requerida.</p>
SOLICITUD DE CAMBIO	<p>Requisitos para incluir una modificación o actualización en el <i>Software</i> o documentación. Puede incluir lo siguiente:</p> <ul style="list-style-type: none"> - Propósito del cambio - Estado del cambio - Nivel de impacto en la documentación o componentes de software asociados
TAREAS	<p>Conjunto de actividades a realizar para generar un <i>Entregable</i> (documento o software).</p>
TEXTURAS	<p>Es una imagen utilizada para cubrir la superficie de un objeto virtual, ya sea tridimensional o bidimensional. Las <i>Texturas</i> para los <i>Modelos</i> son creadas con algún editor de imágenes y después son importadas al software de modelado.</p>
TUTORIAL DE JUEGO	<p>El <i>Tutorial de Juego</i> incluye información sobre cómo debe jugarse el juego. Por ejemplo, cómo realizar movimientos básicos, cómo utilizar armas, el sistema de combos de un juego de peleas, etc.</p>
VIDEOS DEL GRUPO FOCAL	<p>Videos grabados de las sesiones de pruebas (usabilidad, jugabilidad), realizada a generalmente a un pequeño grupo de voluntarios quienes prueban el juego y discuten acerca de lo que les gustó y no les gustó.</p> <p>Esta información es útil para el desarrollador para cambiar algunas cosas antes de que el juego sea finalizado.</p>

REFERENCIAS

- [1] El Financiero (2014), *México: ¿potencia en el desarrollo de videojuegos?*
<http://www.elfinanciero.com.mx/opinion/mexico-potencia-en-la-industria-del-videojuego.html> (última consulta 11/12/2014).
- [2] The Competitive Intelligence Unit (2015). *Potencial de Desarrollo en el Mercado de Videojuegos*
http://the-ciu.net/nwsltr/298_1Distro.html (última consulta 01/12/2015).
- [3] Ann Osborne O'Hagan, Gerry Coleman y Rory V. O'Connor (2014). *Software Development Processes for Games: A Systematic Literature Review*
Springer-Verlag Berlin Heidelberg 2014.
- [4] Excelsior (2014), *Videojuegos, industria que crece en México*
<http://www.excelsior.com.mx/2012/02/20/dinero/811802> (última consulta 11/12/2014).
- [5] The Competitive Intelligence Unit (2015), *Producción de videojuegos móviles: un mercado en proceso de maduración*
http://the-ciu.net/nwsltr/331_1Distro.html (última consulta 02/01/2015).
- [6] ISO/IEC: *ISO/IEC 29110 Software Engineering – Lifecycle Profiles for Very Small Entities (VSEs)*, Part 5-1-2: Management and engineering guide: Generic profile group: Basic profile ISO/IEC JTC1/SC7, Technical Report dated 2011-05-15. (2011).
- [7] Grant Travinor (2009). *The Art of Videogames*: Wiley-Blackwell John Wiley & Sons.
- [8] Juergen Musil, Angelika Schweda, Dietmar Winkler y Stefan Biffel (2010). *Improving Video Game Development: Facilitating Heterogeneous Team Collaboration Through Flexible Software Processes*: Springer.
- [9] Mike Rose (2011). *250 Indie Games You Must Play*: CRC Press.
- [10] Mike Diver (2016). *Indie Games, The Complete Introduction To Indie Gaming*.
- [11] Ian Sommerville (2011). *Ingeniería de Software Novena Edición*: Addison Wesley.
- [12] Kasurinen J., Laine R. (2014). *Games from the viewpoint of software engineering*: Lappeenranta University of Technology.
- [13] Inge van de Weerd, Stefan de Weerd y Sjaak Brinkkemper (2007). *Developing a Reference Method for Game Production by Method Comparison*: Springer.
- [14] Clinton Keith (2010). *Agile Game Development with Scrum*: Addison-Wesley.

- [15] Jussi Kasurinen, Risto Laine y Kari Smolander (2013). *How Applicable is ISO/IEC 29110 in Game Software Development*: Springer.
- [16] D. Vallejo, C. González, D. Villa, F. Jurado et al. (2014). *Desarrollo de Videojuegos: Un Enfoque Práctico*: Escuela Superior de Informática Universidad de Castilla-La Mancha.
- [17] Heather Maxwell Chandler (2014). *The Game Production Handbook Third Edition*: Jones & Bartlett Learning.
- [18] Sanam Narula. *RE in Game Development: A necessity or an overhead*: Presentación.
- [19] E. Bethke (2003). *Game Development and Production*: Wordware Publishing, Inc., Plano, Texas.
- [20] S. Rabin (2005). *Introduction to Game Development*: Charles River Media, Boston.
- [21] Microsoft Official Academic Course (2014). *Gaming Development Fundamentals*: Wiley.
- [22] Christopher M. Kanode y Hisham M. Haddad (2009). *Software Engineering Challenges in Game Development*: IEEE Computer Society.
- [23] D. Callele, E. Neufeld, K. Schneider (2005). *Requirements Engineering and the Creative Process in the Video Game Industry*: Requirements Engineering, IEEE.
- [24] Mara Ares (2015). *Videojuegos Construye tu empresa en 10 pasos*: Alfaomega.
- [25] Katie Salen y Eric Zimmerman (2006). *The Game Design Reader. A Rules of Play Anthology*: Cambridge, MIT Press.
- [26] David Callele, Eric Neufeld y Kevin Schneider (2011). *A Report on Select Research Opportunities in Requirements Engineering for Videogame Development*: Proceedings of the 2011 Fourth International Workshop on Multimedia and Enjoyable Requirements Engineering (MERE'11)
- [27] Draper, S.W. (1999). *Analyzing fun as a candidate software requirement*: Personal Technologies.
- [28] Jussi Kasurinen, Andrey Maglyas, and Kari Smolander (2014). *Is Requirements Engineering Useless in Game Development?*: Springer International Publishing Switzerland.
- [29] Callele, D., Neufeld, E., Schneider, K. (2006): *Emotional Requirements in Video Games*: 14th IEEE International Conference on Requirements Engineering Conference (RE'06).

- [30] Callele, D., Neufeld, E., Schneider, K. (2010). *An Introduction to Experience Requirements*: 18th IEEE International Conference on Requirements Engineering
- [31] Jesse Schell (2008). *The Art of Game Design A Book of Lenses*: ELSEVIER
- [32] Tynan Sylvester (2013). *Designing Games A Guide To Engineering Experiences*: O'REILLY
- [33] Njal Nordmark (2012). *Software Architecture and the Creative Process in Game Development*: NTNU - Trondheim Norwegian University of Science and Technology.
- [34] Jonathan Blow (2004). *Game Development: Harder Than You Think*. Queue - Game Development.
- [35] Kjetil Guldbrandsen y Kjell Ivar Storstein (2009). *Apocalypse Engine: A Study of Software Architecture and Conventions in Modern Game Engines*: Technical report, Norwegian University of Science and Technology.
- [36] Alessandro Ardolino, Remi Arnaud, Paula Berinstein, et al. (2014). *Game Development Tool Essentials*: Apress
- [37] Asociación Española para la Calidad (2016). *Plan de Calidad*
<http://www.aec.es/web/guest/centro-conocimiento/plan-de-calidad>
(última consulta 2/5/2016).
- [38] Project Management Institute (2013). *Guía de los Fundamentos para la Dirección de Proyectos (Guía del PMBOK) - Quinta Edición*: Project Management Institute, Inc.
- [39] Juan Mejía (2013), *Nintendo Marketing Strategy analysis and proposal*
<http://www.slideshare.net/juangmejia/nintendo-marketing-strategy-analysis-and-proposal>
(última consulta 10/5/2016).
- [40] Fighters Generation (2013), *Killer Instinct - Concept Artwork*
<http://www.fightersgeneration.com/games/killerinstinct-2013-p3.html>
(última consulta 10/5/2016).
- [41] Waterman (2014), *Topics of interest to 3D game artists*
<https://watermanp.wordpress.com/trends-in-the-3d-art-production-pipeline-2014/>
(última consulta 10/5/2016).
- [42] Ubisoft (2014), *Tetris Ultimate*
<https://www.ubisoft.com/es-ES/game/tetris-ultimate/>
(última consulta 10/5/2016).