

Desarrollo de un juego de gestión con múltiples metodologías ágiles

Autor: Pau Martínez Artigot
Tutor: Mateo Terrasa Torres
Profesor: Joan Arnedo Moreno

Grado de Multimedia
Videojuegos

Junio de 2024

Créditos/Copyright



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial - SinObraDerivada [3.0 España de Creative Commons.](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

| | |
|---|---|
| Título del trabajo: | <i>Desarrollo de un juego de gestión</i> |
| Nombre del autor: | <i>Pau Martínez Artigot</i> |
| Nombre del colaborador/a docente : | <i>Mateo Terrasa Torres</i> |
| Nombre del PRA: | <i>Juan Arnedo Moreno</i> |
| Fecha de entrega (mm/aaaa): | <i>06/2024</i> |
| Titulación o programa: | <i>Grado de Multimedia</i> |
| Área del Trabajo Final: | <i>Videojuegos</i> |
| Idioma del trabajo: | <i>español</i> |
| Palabras clave | <i>desarrollo, videojuegos, género de gestión</i> |
| Resumen del Trabajo (máximo 250 palabras): | |
| <p>Debido a la experiencia personal del equipo de desarrollo sobre la falta de juegos de gestión militar a nivel introductorio, este plantea la planificación, investigación de mercado y desarrollo de un juego de gestión multiplataforma (PC/Android) sobre la creación y mantenimiento de un grupo mercenario en el motor de videojuegos <i>Godot</i> usando metodologías ágiles para optimizar el proceso con pocos recursos disponibles. Primero se elaboran sistemas de planificación y organización relacionados con metodologías ágiles. Siguiendo los sistemas establecidos, se procede al desarrollo inicial del juego, desde un estudio de mercado hasta una versión jugable, adaptando las metodologías definidas anteriormente. El resultado son dos versiones básicas del juego para Windows y Android, con la capacidad de completar una partida en ambas plataformas. Aunque se esperaba poder desarrollar más del juego y crear una plataforma y presencia en la red, las metodologías ágiles y su capacidad de reestructuración sobre un proyecto en marcha con dificultades de tiempo y recursos han facilitado la priorización de tareas para terminar el proyecto con un producto jugable y satisfactorio.</p> <p>PEC02 vídeo: https://youtu.be/i_PUlitfKBI PEC03 vídeo: https://youtu.be/zvLRLdeid2E GitHub: https://github.com/PMartinezArtigot/PMC-Manager</p> | |

Abstract (in English, 250 words or less):

Due to the personal experience of the development team regarding the lack of entry-level military management videogames, they propose the planning, market research and development of a management videogame in several platforms (Windows and Android) revolving the creation and maintenance of a mercenary group in the videogame engine Godot, using Agile methodology to optimize the process with the small number of resources available. First, several planning and organization systems are developed. Following the established systems, the initial stage of the development process begins, from an extensive market research to the delivery of a first fully playable build, adapting the defined plans with the aforementioned Agile methodology. The final result are two basic but fully playable builds of the game on the stated preferred platforms, with the ability to complete a game. Although the development team wished to develop the game further, as well as establish an Internet presence, the Agile methodology and its abilities to restructure an ongoing project with both time and resource constraints have aided the prioritization of tasks to finish the project within the specified timeframe with a satisfactory and playable product.

Agradecimientos

Primero y ante todo, quiero agradecer al profesorado de la UOC todo su apoyo durante mi educación, especialmente, a Mateo Terrasa Torres por su apoyo durante el desarrollo de este proyecto, sus acertadas críticas y sus valiosas sugerencias han representado un impulso y una inspiración para este trabajo, y a Gerard Cases Bernaus, mi tutor durante mi estancia en la UOC, por su continua orientación, guía y apoyo durante los últimos 4 años.

También dedico mi agradecimiento a los artistas y desarrolladores que dedican su tiempo libre a crear contenido para el dominio público sin ánimo de lucro, su altruismo es un pilar fundamental para los desarrolladores independientes.

Y, por último, agradezco el respaldo de mi familia y amigos, que me han acompañado durante esta experiencia y han ofrecido una mano cuando lo he necesitado.

Abstract

Due to the personal experience of the development team regarding the lack of entry-level military management videogames, they propose the planning, market research and development of a management videogame in several platforms (Windows and Android) revolving the creation and maintenance of a mercenary group in the videogame engine Godot, using Agile methodology to optimize the process with the small number of resources available. First, several planning and organization systems are developed. Following the established systems, the initial stage of the development process begins, from an extensive market research to the delivery of a first fully playable build, adapting the defined plans with the aforementioned Agile methodology. The final result are two basic but fully playable builds of the game on the stated preferred platforms, with the ability to complete a game. Although the development team wished to develop the game further, as well as establish an Internet presence, the Agile methodology and its abilities to restructure an ongoing project with both time and resource constraints have aided the prioritization of tasks to finish the project within the specified timeframe with a satisfactory and playable product.

Resumen

Debido a la experiencia personal del equipo de desarrollo sobre la falta de juegos de gestión militar a nivel introductorio, este plantea la planificación, investigación de mercado y desarrollo de un juego de gestión multiplataforma (PC/Android) sobre la creación y mantenimiento de un grupo mercenario en el motor de videojuegos *Godot* usando metodologías ágiles para optimizar el proceso con pocos recursos disponibles. Primero se elaboran sistemas de planificación y organización relacionados con metodologías ágiles. Siguiendo los sistemas establecidos, se procede al desarrollo inicial del juego, desde un estudio de mercado hasta una versión jugable, adaptando las metodologías definidas anteriormente. El resultado son dos versiones básicas del juego para Windows y Android, con la capacidad de completar una partida en ambas plataformas. Aunque se esperaba poder desarrollar más del juego y crear una plataforma y presencia en la red, las metodologías ágiles y su capacidad de reestructuración sobre un proyecto en marcha con dificultades de tiempo y recursos han facilitado la priorización de tareas para terminar el proyecto con un producto jugable y satisfactorio.

Palabras clave

Godot, desarrollo de videojuegos, metodologías ágiles, gestión militar

Índice

| | | |
|-----------|---|-----------|
| 1. | Introducción..... | 12 |
| 1.1. | Introducción/Prefacio | 12 |
| 1.2. | Descripción..... | 14 |
| 1.3. | Objetivos generales | 17 |
| 1.3.1. | Objetivos principales..... | 17 |
| 1.3.2. | Objetivos adicionales | 18 |
| 1.4. | Metodología y proceso de trabajo | 19 |
| 1.5. | Planificación | 21 |
| 1.5.1. | Diagrama de Gantt..... | 21 |
| 1.5.2. | Tablero kanban | 22 |
| 1.6. | Presupuesto..... | 23 |
| 1.7. | Estructura del resto del documento | 25 |
| 2. | Análisis de mercado | 26 |
| 2.1. | Contexto del mercado | 26 |
| 2.2. | Público objetivo (i.e. <i>target audience</i>) y perfiles de usuario | 29 |
| 2.3. | Estado del arte..... | 31 |
| 2.3.1. | Fuentes de inspiración..... | 31 |
| 2.3.2. | Análisis de productos parecidos | 39 |
| 3. | Propuesta..... | 43 |
| 3.1. | Especificaciones del producto | 43 |
| 3.2. | Modelo de negocio..... | 44 |
| 3.3. | Estrategia de marketing | 44 |
| 4. | Diseño..... | 45 |
| 4.1. | Entorno de desarrollo y otras herramientas..... | 45 |
| 4.2. | Arquitectura de la información..... | 49 |
| 4.3. | Diseño de la interfaz | 50 |
| 4.4. | Diseño gráfico | 58 |
| 4.4.1. | Sprites y otros elementos interactivos | 58 |
| 4.4.2. | Logo, banderas y fondos de pantalla | 59 |

| | |
|---|------------------------------|
| 4.4.3. Estilo de la UI | 65 |
| 5. Demostración | 67 |
| 5.1. Prototipos..... | 67 |
| 5.1.1. Prototipo Lo-Fi..... | 67 |
| 5.1.2. Prototipo Hi-Fi | 70 |
| 5.2. Guía de usuario | 78 |
| 6. Conclusiones y líneas de futuro | 79 |
| 6.1. Conclusiones..... | Error! Bookmark not defined. |
| 6.2. Líneas de futuro | Error! Bookmark not defined. |
| 7. Bibliografía..... | 82 |

Figuras y tablas

Índice de figuras

| | |
|---|----|
| Figura 1: Boceto rápido de la pantalla principal (creación propia)..... | 15 |
| Figura 2: Boceto rápido de la pantalla de mercado negro (creación propia)..... | 16 |
| Figura 3: Boceto rápido de la pantalla de mejora de base (creación propia) | 16 |
| Figura 4: Primer parte del diagrama de Gantt..... | 21 |
| Figura 5: Segunda parte del diagrama de Gantt..... | 21 |
| Figura 6: Tablero kanban en la aplicación Trello | 22 |
| Figura 7: Captura propia de pantalla del pase de batalla de <i>Fortnite</i> | 26 |
| Figura 8: <i>Mother Base</i> en <i>Metal Gear Solid: Peace Walker</i> (Metal Gear Wiki) | 31 |
| Figura 9: <i>Mother Base</i> en <i>Metal Gear Solid V: The Phantom Pain</i> (Metal Gear Wiki) | 32 |
| Figura 10: Vista de personal en espera en MGS V (“ <i>Comprehensive Staff Guide</i> ” por Caihead - <i>Steam</i>) | 32 |
| Figura 11: Inventario de un soldado en XCOM 2 (“ <i>Complete Soldier Guide</i> ” por Spin - <i>Steam</i>)..... | 34 |
| Figura 12: Captura del juego <i>Rebel Inc.</i> (<i>Ndemio Creations – Steam</i>)..... | 35 |
| Figura 13: Captura propia del menú principal de <i>Girls’ Frontline</i> | 36 |
| Figura 14: Ejemplo de misiones de logística en <i>Girls’ Frontline</i> | 37 |
| Figura 15: Un escuadrón a punto de llevar a cabo una misión de logística. | 37 |
| Figura 16: Un ejemplo de un escuadrón del juego <i>Girls’ Frontline</i> Los rectángulos rojos indican la efectividad de combate total y de cada uno de los personajes | 38 |
| Figura 17: Ejemplo de una misión en el juego <i>Girls’ Frontline</i> Marcado en rojo el valor <i>STRENGTH</i> que informa al jugador de la dificultad del nivel | 38 |
| Figura 18: Captura de pantalla de <i>PMC Builder</i> (<i>Improper Works – itch.io</i>)..... | 39 |
| Figura 19: Captura de pantalla del juego <i>Uprising</i> (BLAKE – <i>itch.io</i>)..... | 40 |
| Figura 20: Captura propia del juego <i>PMC-Suite</i> | 41 |
| Figura 21: Captura de pantalla del juego <i>A war of a mad man’s making</i> (<i>Witching Metal Productions – itch.io</i>) .. | 42 |
| Figura 22: Diagrama propio de los elementos generales del juego | 49 |
| Figura 23: <i>Wireframe</i> propio de la pantalla de misiones | 51 |
| Figura 24: <i>Wireframe</i> propio de consulta de más información de una misión..... | 51 |
| Figura 25: <i>Wireframe</i> propio de la pantalla de asignación de un escuadrón..... | 52 |
| Figura 26: <i>Wireframe</i> propio de la pantalla del mercado negro | 53 |
| Figura 27: <i>Wireframe</i> propio de la pantalla de Base de Operaciones | 54 |
| Figura 28: <i>Wireframe</i> propio de la pantalla de mejora de instalaciones..... | 55 |
| Figura 29: <i>Wireframe</i> propio de la pantalla de la clínica | 55 |
| Figura 30: <i>Wireframe</i> propio de la pantalla de los barracones..... | 56 |
| Figura 31: <i>Wireframe</i> propio de la pantalla del almacén..... | 56 |
| Figura 32: <i>Wireframe</i> propio de la pantalla del garaje..... | 57 |
| Figura 33: Captura de los <i>sprites</i> creados por el equipo de desarrollo | 58 |
| Figura 34: <i>Info Icon - Freepik</i> | 58 |
| Figura 35: <i>Clipboard pixel art</i> (Bekzod Shoyakubov - Vecteezy) | 58 |
| Figura 36: <i>Black skull icon</i> (Alexey Gomankov - Wikimedia Commons) | 59 |

| | |
|---|----|
| Figura 37: “ <i>Fraternitas Livonica</i> ” shield (Vladusty - Wikimedia Commons) | 59 |
| Figura 38: Versión básica/original del logo de PMC Manager | 59 |
| Figura 39: Logo final (sin texto) para el juego PMC Manager | 60 |
| Figura 40: Bandera de la facción NUO dentro del juego PMC Manager | 60 |
| Figura 41: <i>Gun outline</i> (INVERTED - Wikimedia Commons) | 61 |
| Figura 42: <i>Wikimania2019 dove icon</i> (BFlores (WMF) - Wikimedia Commons) | 61 |
| Figura 43: <i>Balanced scale of Justice</i> (Perhelion - Wikimedia Commons) | 61 |
| Figura 44: Bandera de la facción EPE dentro del juego PMC Manager | 61 |
| Figura 45: Bandera de la facción FLR dentro del juego PMC Manager | 62 |
| Figura 46: <i>Fist</i> (Eugenio Hansen, OFS - Wikimedia Commons) | 62 |
| Figura 47: AK-47 (Skoll - Wikimedia Commons) | 62 |
| Figura 48: <i>Brown-dominant "highland" ERDL camouflage pattern swatch</i> (ERDL - Wikimedia Commons) | 63 |
| Figura 49: Fondo de pantalla principal del juego PMC Manager | 63 |
| Figura 50: Imagen satélite de una base militar – Planet Labs | 64 |
| Figura 51: Fondo de la pantalla Base de Operaciones en el juego PMC Manager | 64 |
| Figura 52: <i>Abandoned warehouse. Black and white image. noise scene.</i> (SAYAN - Adobe Stock) | 64 |
| Figura 53: Fondo de la pantalla del Mercado Negro en el juego PMC Manager | 65 |
| Figura 54: <i>dark mysterious man in a hoodie is hiding his face, hacker, anonymous</i> (spaxiax - Adobe Stock) | 65 |
| Figura 55: Mercader del juego PMC Manager | 65 |
| Figura 56: Captura de la pantalla inicial del prototipo básico de PMC Manager | 67 |
| Figura 57: Captura de la pantalla Lista de Misiones del prototipo básico de PMC Manager | 68 |
| Figura 58: Captura de la pantalla secundaria “Más información” de una misión del prototipo básico de PMC Manager | 68 |
| Figura 59: Captura de la pantalla secundaria “Asignar escuadrón” de una misión del prototipo básico de PMC Manager | 69 |
| Figura 60: Captura de la pantalla Base de Operaciones del prototipo básico de PMC Manager | 69 |
| Figura 61: Captura de la pantalla inicial del prototipo avanzado de PMC Manager | 70 |
| Figura 62: Captura de la pantalla secundaria general de “Información” del prototipo avanzado de PMC Manager | 70 |
| Figura 63: Captura de la pantalla “Lista de Misiones” del prototipo avanzado de PMC Manager | 71 |
| Figura 64: Captura de la pantalla secundaria general de “Misiones en curso” del prototipo avanzado de PMC Manager | 71 |
| Figura 65: Captura de la pantalla secundaria “Más información” de una misión del prototipo avanzado de PMC Manager | 72 |
| Figura 66: Captura de la pantalla secundaria “Asignar escuadrón” de una misión del prototipo avanzado de PMC Manager | 72 |
| Figura 67: Captura de la pantalla secundaria general “Informe de estado diario” del prototipo avanzado de PMC Manager | 73 |
| Figura 68: Captura de la pantalla secundaria general “Soldados caídos en combate” del prototipo avanzado de PMC Manager | 73 |
| Figura 69: Captura de la pantalla “Base de Operaciones” del prototipo avanzado de PMC Manager | 74 |
| Figura 70: Captura de la pantalla secundaria “Mejorar instalación” del prototipo avanzado de PMC Manager | 74 |

Figura 71: Captura de la pantalla secundaria “Clínica” del prototipo avanzado de PMC Manager 75

Figura 72: Captura de la pantalla secundaria “Barracones” del prototipo avanzado de PMC Manager 75

Figura 73: Captura de la pantalla “Mercado Negro” del prototipo avanzado de PMC Manager 76

Figura 74: Captura de la pantalla secundaria “Contratar soldado” del prototipo avanzado de PMC Manager 76

Figura 75: Captura de la pantalla secundaria general “Finalizar planes” del prototipo avanzado de PMC Manager 77

Figura 76: Captura de la pantalla de derrota del prototipo avanzado de PMC Manager 77

Índice de tablas

Tabla 1: Desglose de gastos 24

Tabla 2: Requerimientos mínimos de Godot en ordenadores y portátiles (Godot, 2024) 46

Tabla 3: Requerimientos recomendados de Godot en ordenadores y portátiles (Godot, 2024) 47

1.Introducción

1.1. Introducción/Prefacio

Desde que tengo memoria siempre he jugado a videojuegos. Desde la videoconsola *V.Smile* por *VTech*, la cual fue diseñada para niños de 3 a 9 años, pasando por casi todas las consolas de *Nintendo* y *Sony* que se han lanzado desde el principio del milenio hasta el día de hoy, incluyendo obviamente una gran librería de videojuegos en el ordenador. Durante todo este tiempo he probado todo tipo de juegos, desde los nombres más grandes como *Nintendo*, *GameFreak*, *Konami*, *Blizzard* hasta la época dorada de los juegos *Flash*, donde cualquier persona tenía la posibilidad de poder crear un juego y subirlo en la Web.

En especial adquirí un gran interés por todo lo militar, en parte, gracias a los titanes de los juegos de tiros en primera persona, *Battlefield* y *Call of Duty*, y también por las influencias de mi madre y su amor por el cine bélico (películas como *El puente sobre el río Kwai* y *La gran evasión* y series como *M.A.S.H.*, entre otras).

Esta influencia, más mi amor por los videojuegos, me llevó a querer crear algo para este tipo de juegos. Empecé a hacer mapas en *Hammer*, una herramienta de creación de niveles para los juegos que usan el motor gráfico *Source*, desarrollado por *Valve*, los cuales son mayormente juegos de tiros en primera persona (*Counter-Strike*, *Team Fortress 2* y *Half Life 2*). Con la experiencia que amasé durante los años, me di cuenta de mi creciente pasión por el desarrollo de este tipo de contenido. Esta pasión se materializó en mi trabajo de investigación de final de Bachillerato, el cual se basó en usar una copia de *Half-Life 2* y modificarla para crear un juego nuevo, un juego de tiros en primera persona llamado *Global Operator Taskforce*, donde el jugador se ponía en los pies de fuerzas especiales de diferentes partes del mundo para llevar a cabo una misión de eliminar matones y el jefe de una mafia en un edificio abandonado.

Durante el curso de mis estudios en el grado de Multimedia en la UOC, he programado una variedad de juegos, desde los clásicos como *Pong* o *Arkanoid* en Javascript y Java respectivamente, hasta una versión del juego de cartas siete y medio para navegadores en HTML/CSS/Javascript. No sólo eso, en mis últimas prácticas curriculares en TERMCAT (organismo del Departament de Cultura de la Generalitat de Catalunya), restauré un antiguo juego *Flash* en *Godot*, para que se pudiera volver a subir el juego en su sitio web.

Empecé el grado de Multimedia impulsado por mi pasión por los videojuegos y su creación, y como culminé mis estudios anteriormente con una modificación de un juego (aún debía acrecentar mi conocimiento técnico) y como, durante mi grado, mis proyectos favoritos dentro y fuera de la UOC han sido la creación de videojuegos, he decidido culminar mis estudios creando un juego desde cero, para poner a prueba mis conocimientos y cerrar esta etapa como siempre he querido.

1.2. Descripción

Una de las sagas de juegos que ha tenido más impacto en el mundo de los videojuegos ha sido la saga de *Metal Gear Solid*. En dos de los juegos (*Peace Walker* y *Metal Gear Solid V*), el personaje principal crea y hace crecer un grupo de mercenarios y su base de operaciones. El jugador acrecienta su pequeño ejército captando soldados y enviándolos a misiones, adquiriendo y usando recursos para gestionar la base. Principalmente esta mecánica es un elemento secundario del juego, una manera de desbloquear nuevo equipamiento y habilidades, por lo que acaba teniendo poca profundidad.

Si se elaborara sobre esta mecánica secundaria adaptándola para hacerla una mecánica principal, añadiendo una historia, objetivo principal y nuevas mecánicas, se podría crear un juego multiplataforma de un jugador que podría tener bastante éxito, ya que una rápida búsqueda en plataformas como *Steam*, *itch.io* y *Play Store* con palabras como *Mercenary* o *PMC* nos muestran pocos resultados, de los cuales, la mayoría son proyectos abandonados, por lo tanto, hay un vacío en el mercado que no se ha llenado.

Con la introducción del concepto de PMCs en videojuegos altamente populares como el *reboot* de la saga *Call of Duty: Modern Warfare* y uno de los juegos pioneros del género de juegos de tiros *extraction looter*, *Escape From Tarkov*, el jugador promedio está familiarizado con este concepto e incluso posiblemente tenga interés en probar un juego que contenga el mismo concepto, lo cual sería un punto de atracción ciertamente destacable.

Por lo tanto, el objetivo de este trabajo es generar un videojuego de gestión donde el jugador cree y mantenga su empresa militar privada, cumpliendo misiones, contratando soldados y dándoles equipamiento mientras gestiona varios recursos, con un concreto final: el jugador tendrá que dar apoyo a una de las tres grandes fuerzas en un conflicto mundial y hacer que ganen la guerra. Este juego sería idealmente multiplataforma, disponible en ordenador y Android y para garantizar que el proyecto sea lo más conveniente de llevar a cabo, se desarrollará en *Godot*, debido a la familiaridad y simplicidad del motor de juego, además de su capacidad de generar juegos multiplataforma con cambios ínfimos

El *game loop* del juego sería este:

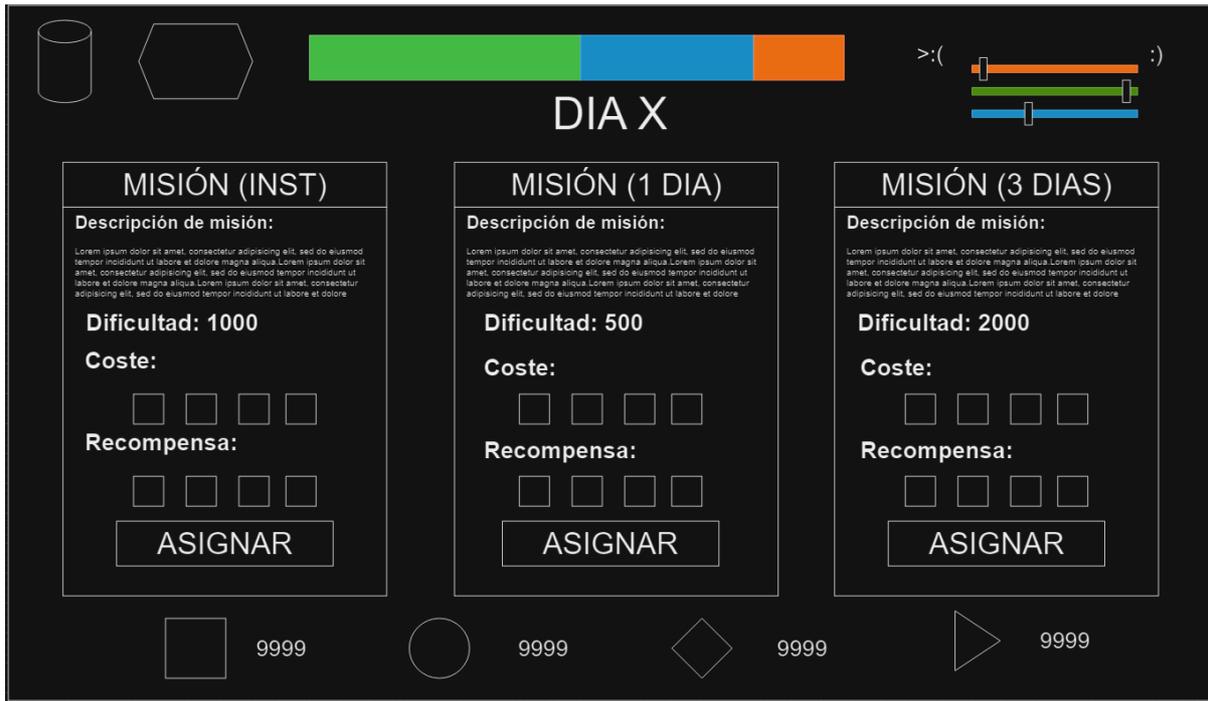


Figura 1: Boceto rápido de la pantalla principal (creación propia)

El jugador empieza el día X del conflicto y tiene un número de misiones disponibles (elegidas aleatoriamente) las cuales pueden ser neutras o apoyar a una de las tres fuerzas (e incluso perjudicar a otras). Antes de empezar, la misión el jugador puede ver el nivel de dificultad y lo podrá comparar con el nivel de combate de uno de sus escuadrones. Estas misiones durarán días, por lo tanto, el jugador podrá planificar su gestión de recursos de acuerdo con las misiones que tenga que llevar a cabo. La realización de estas misiones dará más recursos al jugador e incluso ítems para mejorar sus soldados. Si fracasa en la realización, perderá recursos y tendrá soldados heridos o incluso podrá perderlos en combate.

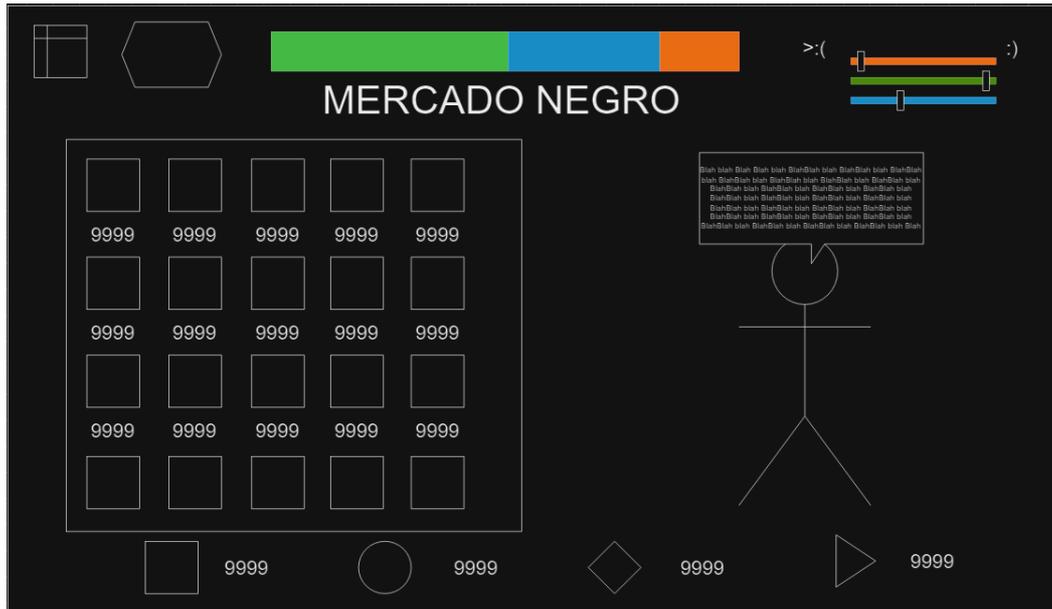


Figura 2: Boceto rápido de la pantalla de mercado negro (creación propia)

También tendrá a su disposición un mercado negro en otra pantalla, donde el usuario podrá comprar ítems y equipamiento para mejorar sus soldados. Este mercado se irá actualizando cada día.



Figura 3: Boceto rápido de la pantalla de mejora de base (creación propia)

Por último, tendrá una sección que mostrará aspectos de su base, los cuales podrá mejorar para tener beneficios (como acceso a más misiones, poder tener más escuadrones, más espacio de inventario, etc.).

1.3. Objetivos generales

1.3.1. Objetivos principales

El objetivo básico será el siguiente:

Crear un prototipo jugable sobre un juego de gestión donde el jugador funde y mantenga su empresa militar privada, cumpliendo misiones y contratando soldados, mientras gestiona varios recursos, con el objetivo final de dar suficiente apoyo a uno de los tres bandos del conflicto hasta que ganen o mantener la guerra activa ayudando a los tres bandos y adquiriendo la mayor cantidad de recursos disponibles.

Objetivos del juego:

- Crear los fundamentos para que cada *playthrough* se afronte de una manera diferente añadiendo elementos de aleatoriedad sobre mecánicas existentes.
- Incorporar mecánicas fundamentales para mostrar el futuro funcionamiento del juego
- Cubrir un vacío en el mundo de los juegos de gestión en el aspecto de gestión de mercenarios.

Objetivos para el jugador:

- Que el jugador tenga una vista anticipada de lo que será el producto final
- Que el jugador se vea involucrado en sus decisiones durante una *playthrough* y que dichas decisiones tengan impacto en el juego.
- Que el jugador sea estimulado mentalmente respecto a la gestión de sus recursos y sea castigado o premiado dependiendo de sus decisiones.

Objetivos personales del autor del TF:

- Expandir el conocimiento sobre el desarrollo y diseño de un videojuego.
- Adquirir experiencia sobre la creación de *sprites*, audio y otros elementos artísticos relacionados con el desarrollo de un videojuego.
- Aplicar los conocimientos sobre el motor de juegos *Godot* y expandirlos sobre la base de un juego con más complejidad que proyectos anteriores

1.3.2. Objetivos adicionales

Estos objetivos no son necesarios para considerar el TF completo. Son objetivos para una proyección de futuro o si se invirtiera más tiempo de desarrollo.

- Que el jugador desarrolle una conexión con sus unidades y su grupo de mercenarios.
- Plantear suficientes mecánicas para generar más *playtime* para justificar volver a jugar el juego después de una partida.
- Que el jugador se vea motivado para hacer más de una *playthrough*.
- Crear una campaña *on-line* o presencia para poner el juego al público
- Subir el juego en plataformas públicas como itch.io
- Hacer encuestas sobre *playtesters* para desarrollar el juego

1.4. Metodología y proceso de trabajo

La metodología ágil abarca una gran cantidad de técnicas de trabajo, las cuales giran alrededor de ciclos de trabajo cortos. Estos ciclos cortos son muy útiles debido a que ayudan a mostrar resultados a un supuesto cliente o generar progreso palpable a un superior, en vez de esperar a varias *deadlines* que suelen ser de larga duración. No solo eso, esta planificación de períodos de trabajo más cortos ayuda a flexibilizar la gestión de trabajo en caso de ciertos imprevistos.

Los beneficios de las metodologías ágiles son excelentes y encajan perfectamente con el desarrollo de este TF. Así que, se plantea usar diferentes metodologías ágiles para simplificar el flujo de trabajo:

- **Para grandes hitos y fechas clave:**

Para poder planificar fechas clave, grandes hitos (como entregas de PECs y versiones del juego), usaremos una metodología ágil *Scrum*, la cual divide el desarrollo de un producto en *sprints* o ciclos de trabajo. Teniendo en cuenta la última fecha de entrega de la memoria (PEC 4) siendo el día 16 de junio y empezando el desarrollo desde la semana que empieza el día 25 de marzo, nos encontramos con un plazo de 12 semanas para hacer el desarrollo de este TF. Cada *sprint* será una semana, por lo tanto, este proyecto será dividido en 12 *sprints* y en tres fases de desarrollo. Para asistir en la visualización y organización de esta metodología aplicada, usaremos un diagrama de Gantt, el cual se hará en la plataforma Canva y raramente será modificado.

- **Para el propio desarrollo del juego:**

Para organizar el desarrollo del juego a menor escala y los pasos de redacción de la memoria, la metodología ágil *Kanban* será una gran herramienta para definir qué tareas llevar a cabo y su prioridad, a la vez que facilitará la documentación del progreso del proyecto. *Kanban* o seguimiento visual del progreso del proyecto ayuda a la organización de trabajo dividiendo un proyecto en tareas escritas en tarjetas de señalización y clasificándolas de acuerdo con las necesidades del equipo (en este caso, el autor de este TF). Para la visión y organización *Kanban*, usaremos la plataforma Trello, la cual se irá actualizando de manera periódica con nuevas tareas y actualizando sus estados.

La estructura del proyecto abarca tres grandes divisiones, marcadas por la fecha de entrega de cada PEC:

- **Concepto y desarrollo temprano (25/03/2024-21/04/2024)**

En esta primera fase se definirán los conceptos básicos y mecanismos a partir del análisis de mercado, se categorizarán todas las *features* por urgencia/importancia en Trello, se definirán los diseños de las pantallas con *wireframes* y se creará una primera *build* conceptual, la cual servirá para mostrar las pantallas básicas y con un mínimo de jugabilidad (*low fidelity/alpha*).

- **Desarrollo (22/04/2024-19/05/2024)**

En esta segunda fase se llevará a cabo la mayoría de carga del desarrollo, teniendo en cuenta el trabajo anterior y planificación y definición de *features*. A la vez, se llevará a cabo la creación de elementos gráficos (*sprites*). Al final de este ciclo se espera tener una *build* beta donde el jugador pueda completar el juego, aunque falten mecánicas secundarias.

- **Retoques y *playtesting* (20/05/2024-16/06/2024)**

En esta última fase se preparará el *launch* privado, donde se ofrecerá *playtesting* a un grupo reducido de personas para probar el juego y dar sus opiniones a través de un formulario o entrevistas privadas. Aparte de este *launch* privado, se creará un tráiler para acompañar el evento. Todo el proyecto culminará en la grabación de la defensa del TF.

Se profundizará sobre el diagrama de Gantt, la distribución de *sprints* y fechas exactas más adelante, en el siguiente apartado.

1.5. Planificación

Como se ha dicho anteriormente, la planificación de este TF se rige por dos metodologías:

1.5.1. Diagrama de Gantt

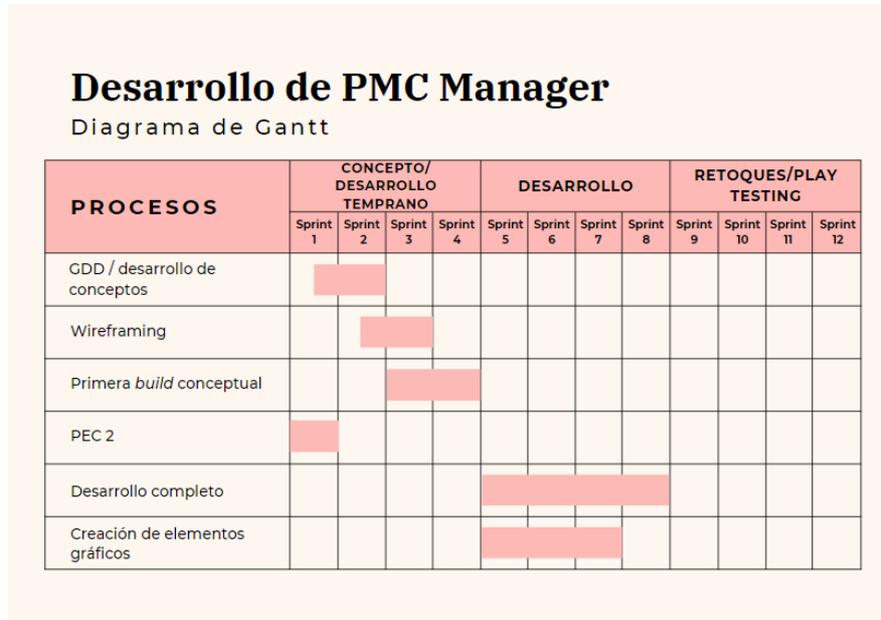


Figura 4: Primer parte del diagrama de Gantt

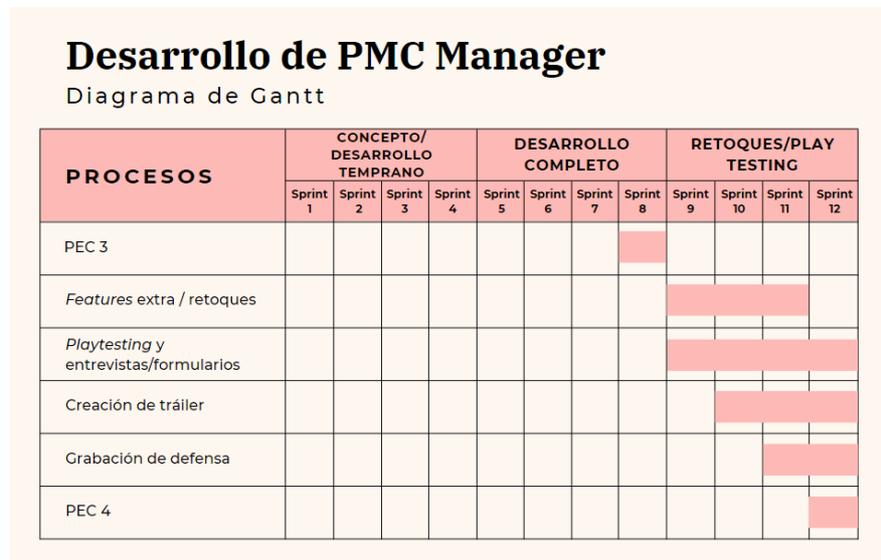


Figura 5: Segunda parte del diagrama de Gantt

Como se puede apreciar, las tareas generales se han dividido en diferentes procesos, con cada proceso perteneciendo a las divisiones mencionadas en el apartado anterior. Cada *sprint* se refiere a una semana entre el 25 de marzo hasta el 16 de junio, con cada PEC y su

semana de entrega asignada. Aunque no se pueden definir fechas exactas debido a la naturaleza del desarrollo de este TF más la temprana situación actual del proyecto, hemos podido organizar el desarrollo en *sprints* generales para asistir en la organización de las tareas durante las siguientes 12 semanas.

1.5.2. Tablero kanban

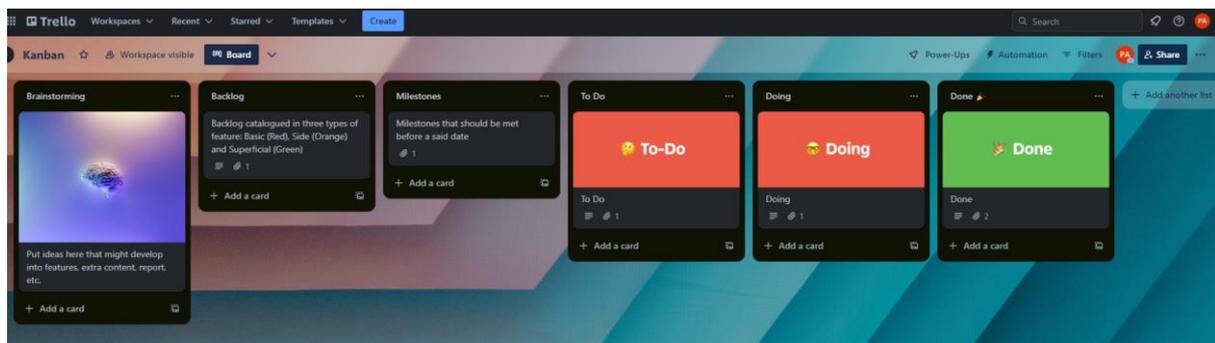


Figura 6: Tablero kanban en la aplicación Trello

El tablero kanban se llenará con las varias *features* y *milestones* durante los dos primeros *sprints* (el proceso nombrado “GDD / desarrollo de conceptos”). *Milestones* incluirá las entregas de PECs, finalizaciones de grandes tareas (como el tráiler, versiones del juego, etc.) mientras que *brainstorming* incluirá varias ideas para *features* u otros conceptos para el TF encontradas después de la primera fase. Estas ideas se acabarán transformando en tareas, las cuales serán las piezas que conjuntamente forman un *milestone*.

Se espera seguir un proceso como el siguiente:

Se escogen varias tareas del *backlog* dependiendo de su importancia (rojo = urgente o básico, naranja = secundario, verde = opcional) y se coloca en la lista *To Do* cada *sprint*. Durante ese *sprint* se deben ir desplazando las tareas a *Doing* hasta que son completadas y se desplazan a *Done*. En el caso de no haber más tareas urgentes o secundarias durante ese *sprint*, se pueden considerar ideas de la lista de *brainstorming* y volverlas en tareas para el *backlog*.

Este sistema es susceptible de cambio, como cambios de lista, pero el concepto general será el mencionado anteriormente.

1.6. Presupuesto

Aunque la idea de presupuesto no se tiene muy en cuenta durante este proyecto (este TF hace más énfasis en el propio desarrollo de un juego, no el proceso de monetización desde un punto de vista empresarial), podemos desglosar un presupuesto general basándonos en el tiempo de trabajo más el *software* necesario para llevarlo a cabo:

- **Horas de trabajo:**

Si en cada *sprint* se tiene en cuenta que se van a trabajar los siete días de la semana durante 4 horas cada día, sale a un total de 28 horas semanales, por 12 *sprints*, nos quedan 336 horas de trabajo. Teniendo en cuenta que el salario mínimo interprofesional es 926.10 euros mensuales por 28 horas trabajadas semanales [1] (Bermejo, 2024), las horas de trabajo, como mínimo, tendrían un coste de 2778.3 euros.

- **Software usado:**

Como se ha dicho anteriormente, este juego se creará a partir del motor de juegos *Godot*, el cual es completamente gratuito. Las herramientas de organización como *Canva* y *Trello* también tienen versiones gratuitas y de momento no se predice el uso de sus versiones *premium*. Por último, otras aplicaciones necesarias, como OBS para grabar la cámara web y los contenidos de la pantalla, o edición de audio como *Audacity* también serán gratuitas.

Se espera usar varias aplicaciones del paquete de *Adobe* para llevar a cabo el apartado gráfico del juego (*Photoshop*) y para crear contenido multimedia que acompañe el TF, como las grabaciones de defensa o los tráilers (*Premiere Pro*). Si se contratara el plan de particulares de Adobe [2] (Adobe, 2024) debido a que la UOC no proporcionase este paquete de forma gratuita, este tendría un coste de 36,29 euros mensuales, con un coste total de 108.87 euros.

El otro paquete de *software* necesario para redactar este documento sería el paquete de Office 365 de Microsoft. La versión personal de este paquete tendrá un coste de 7 euros mensuales, teniendo un coste total de 21 euros totales.

Tabla 1: Desglose de gastos

| | Horas de trabajo | Paquete de Adobe | Paquete de Office |
|--------------|-------------------------|-------------------------|--------------------------|
| Mes 1 | 926.10 | 36.29 | 7.00 |
| Mes 2 | 926.10 | 36.29 | 7.00 |
| Mes 3 | 926.10 | 36.29 | 7.00 |
| TOTAL | 2778.30 | 108.87 | 21.00 |

Por lo tanto, el estimado coste total de este proyecto será de **2908.17 euros**. Esta cifra no tiene en cuenta el precio de la electricidad ni el *hardware* necesario, ya que no se precisa de nuevo *hardware* y el coste de la electricidad no se verá afectado por esta operación

1.7. Estructura del resto del documento

- **Análisis de mercado:**

Se estudiarán otros juegos del género parecidos, sus altos y bajos, cómo han sido recibidos por su público a través de foros en línea y se analizarán los resultados para definir los aspectos básicos del juego.

- **Propuesta:**

Sobre la base de la información adquirida del análisis de mercado, se elabora una propuesta para definir completamente todos los aspectos del juego y sus características diferenciales respecto a otros juegos.

- **Diseño:**

Se muestran todos los pasos de diseño de creación del juego, desde los *wireframes*, paletas de color, tipografías, hasta la arquitectura interna del código.

- **Demostración:**

En este capítulo se mostrarán capturas desde las primeras versiones del juego hasta la última versión del juego que se tenga al tiempo de entrega de la memoria.

- **Conclusiones y líneas de futuro:**

En el último capítulo se hará un análisis final del proyecto, resaltando sus altos y bajos, sus mayores obstáculos y el futuro del juego y sus subsecuentes actualizaciones.

2. Análisis de mercado

2.1. Contexto del mercado

El mercado de videojuegos está más saturado que nunca, con un gran énfasis en juegos que requieren más y más tiempo del jugador para exprimir todo el contenido del producto. Un ejemplo serían los juegos como servicio (o *games as a service*), como *Fortnite*, *PUBG: Battlegrounds*, *Apex Legends*, *Call of Duty: Warzone*, etc. Estos son de los juegos más populares actualmente, con *Fortnite* manteniendo una cifra consistente de 2 a 4 millones de jugadores diarios [3] (Ashley, 2024), mientras que otros juegos parecidos mantienen cifras entre 300 y 600 mil jugadores diarios entre varias plataformas [4] (Tylor-Hill, 2023), [5] (Steam, 2024). Estos juegos son normalmente gratuitos y la progresión gira alrededor de la mecánica del pase de batalla (*battle pass*).

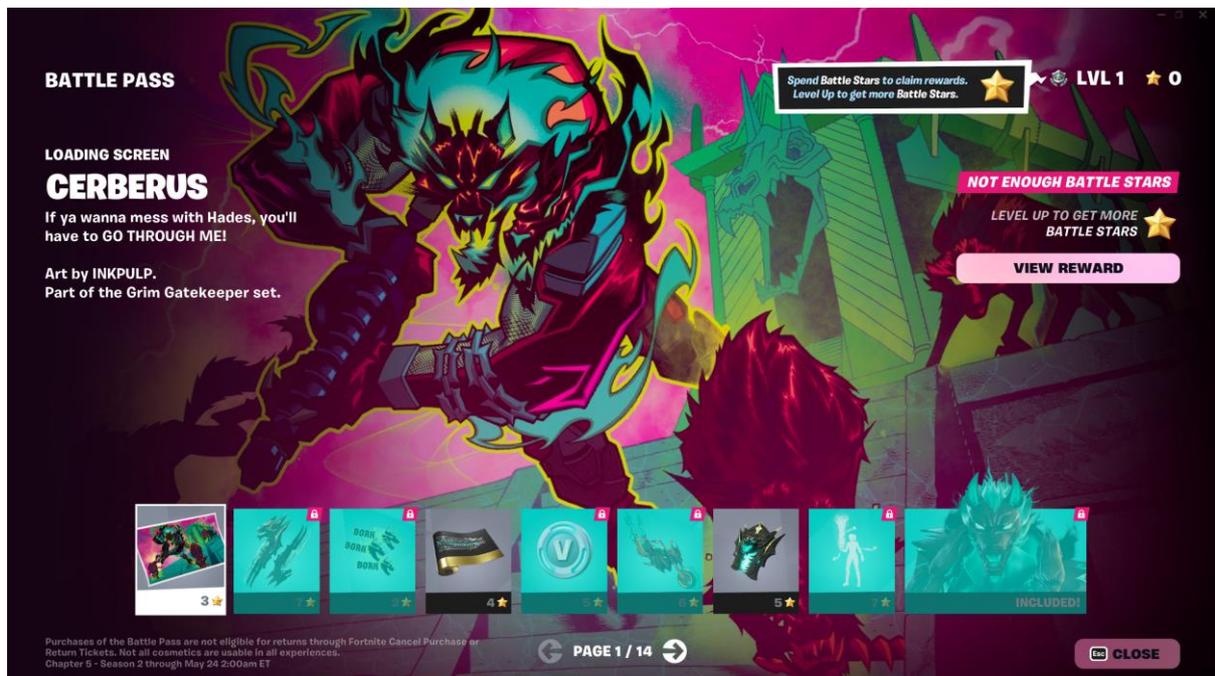


Figura 7: Captura propia de pantalla del pase de batalla de *Fortnite*

Este concepto fuerza al jugador a jugar muchas horas cumpliendo objetivos o retos para desbloquear nuevos cosméticos o bonificaciones para ganar más partidas. El problema se halla en que el tiempo para completar un pase de batalla es limitado. Estos pases de batalla aparecen por temporadas, normalmente duran alrededor de 2 o 3 meses y cuando este período termina, el contenido de este pase de batalla no se puede volver a conseguir bajo

ningún concepto. Esto induce al jugador un sentimiento de FOMO (*Fear Of Missing Out*) [6] (Komad, 2023), un término que describe el miedo a perder una oportunidad que no volverá a tener. Por lo tanto, el jugador se ve obligado a jugar para cumplir con esta *deadline* para conseguir todos los beneficios en el tiempo pactado.

La cuestionable ética de este tipo de prácticas (cuyas implicaciones merecerían un TF por sí mismas) trasciende el ámbito del presente TF, pero estas implicaciones deben exponerse en el contexto de análisis de mercado para poner en perspectiva cuál es la experiencia promedio de un usuario hoy en día y qué tipo de productos consume. El sistema de pase de batalla es simplemente una de muchas herramientas usadas en la industria para monopolizar el tiempo de juego del jugador, creando una relación que encadena al usuario al producto.

Otro aspecto a tener en cuenta en la industria de los videojuegos es el crecimiento de los juegos en móviles. En 2023, el mercado de los juegos en móviles ocupaba el 49% del mercado mundial, generando 90 mil millones de dólares en beneficios de los 184 que la industria entera había generado [7] (Knezovic, 2024). Una parte fundamental de los juegos de esta industria usan fórmulas simples pero adictivas, que buscan llenar “horas vacías”, como viajes en transporte público, tiempos de espera, descansos de trabajo, etc. Juegos como *Clash Royale*, *Subway Surfers*, *Angry Birds* y *Candy Crush* son ejemplos de juegos que han conseguido ocupar un espacio dentro de la industria al identificar este vacío y usar más tiempo del usuario.

Pero incluso el mercado de los juegos de móvil está relegando estos juegos simples, sustituyéndolos por productos de jugabilidad muy compleja, historias largas y enrevesadas, con una calidad gráfica comparable a las consolas de una o dos generaciones anteriores y requiriendo tanto tiempo de juego como los juegos tradicionales. Juegos como *Genshin Impact*, *Honkai: Star Rail* y *Fate / Grand Order*, los cuales se encuentran entre los 10 juegos con mayor recaudación durante 2023 [8] (Clement, 2024) son ejemplos de juegos que tienen mecánicas muy profundas, largas historias y que requieren horas de “farqueo” (repetir acciones dentro del juego para conseguir ítems de mejora, subir de nivel a tus personajes, etc.).

Por lo tanto, al plantear un nuevo juego con muy poca financiación en esta nueva era, donde los titanes de la industria intentan acaparar el mayor tiempo disponible de los jugadores en tantas plataformas, nos tenemos que plantear esta cuestión:

“¿Nuestro pequeño producto podrá competir con los otros?”

Cuando se opera desde una perspectiva *indie* ante grandes titanes AA-AAA, los cuales siguen usando este tipo de prácticas de retención de jugadores, se ha de valorar qué habilidad posee el equipo de desarrollo para poder generar suficiente atracción para que el jugador rompa uno de sus enlaces de tiempo para dedicarlo a un juego nuevo.

Aunque ese no es el único caso. El mercado del videojuego es tan sumamente vasto que la coexistencia de productos es el *statu quo* de la industria. Es decir, aunque el producto generado no pueda competir a nivel gráfico, técnico o narrativo con los productos AAA, hay suficientes jugadores que, teniendo una fórmula única y refinada, pueden elevar un juego a un nivel de éxito inesperado. Un ejemplo fundamental sería *Minecraft*, un juego *indie* inicialmente desarrollado por Markus Persson que ha acabado siendo el juego más vendido de toda la historia [9] (Brody, 2023). Otros ejemplos más comunes son *Terraria*, *Undertale*, *Hollow Knight* [10] (Krambs, 2024), juegos *indie* que han vendido millones de copias y son considerados piezas fundamentales de la industria.

En resumen, el mercado actual está liderado por los desarrolladores más grandes, que luchan de manera feroz para monopolizar el tiempo de juego del usuario con un abanico de prácticas de retención escondidas debajo de juegos de alta fidelidad y complejidad, pero aun así, los juegos *indie*, gracias a sus méritos, han encontrado su sitio en este vasto mercado. Por lo tanto, la estrategia que se plantea en este TF será la siguiente:

Al analizar la capacidad del equipo, se propone el desarrollo de un juego multiplataforma que se encuentre en el punto medio del abanico de complejidad y *storytelling*, con una fidelidad gráfica atractiva pero básica (se considera el estilo 16-bit) para rellenar el vacío del mercado de un juego que requiera una involucración moderada por parte del usuario sin obstruir su tiempo de juego de otros productos.

2.2. Público objetivo (i.e. *target audience*) y perfiles de usuario

Con la contextualización del mercado ya hemos expuesto una parte fundamental de nuestro público objetivo: jugadores ya establecidos en otros juegos que busquen una distracción secundaria que no les ocupe tanto tiempo. En el capítulo anterior, en el apartado de Descripción se señalan varios juegos relacionados con el concepto de empresas militares privadas (*Call of Duty*, *Escape From Tarkov*). El público objetivo serían jugadores que consumen este tipo de producto (juegos militares basados en un escenario moderno) y que tengan un interés por el mundo de la militaría. Hay que remarcar que no todos los jugadores que consumen este tipo de productos tienen un interés por el ámbito militar, pero sí que es cierto que la gran mayoría de gente que tiene interés por la militaría suele jugar a este tipo de juegos, por eso, los datos relacionados con este género nos pueden ser útiles.

Especificar un rango de edad en concreto sobre un género tan vasto como los juegos de tiros en primera persona no es estrictamente necesario para analizar nuestro público objetivo, pero existe un consenso general: un rango de jugadores varones entre 15 y 40 años suelen jugar a tiros en primera persona [11] (Kroupp, 2023)

Se podría argumentar que se está analizando una base errónea de jugadores, ya que se planea crear un juego de gestión, no de tiros en primera persona. Mi refutación sería que el perfil que se está intentando atraer no es todo tipo de jugadores de tiros en primera persona, sino que más bien tengan un interés en el mundo de la militaría y, por ese interés, acaban jugando a juegos de tiros en primera persona. Para esta relación, se infiere que la temática de los juegos de tiros en primera persona suele estar muy ligada a la temática militar y, con la novedosa idea que se propone, se espera atraer a estos jugadores a este género con un juego introductorio.

Cabe insistir en que, cuando se habla de “juegos de tiros en primera persona de temática militar”, no sólo se consideran los juegos más grandes como *Call of Duty* o *Battlefield*, los cuales históricamente han tenido una *playerbase* de jugadores casuales, sino que también se tienen en cuenta varios títulos, antiguos y nuevos, que hacen énfasis en el realismo hasta el punto de simulación (la saga *Armed Assault*, o *ArmA*, la saga *Rising Storm*, *Insurgency: Sandstorm*, *Escape From Tarkov*, *Enlisted*, *Verdun*, etc.).

En resumen, el perfil esperado para el público objetivo sería el siguiente:

- Varón
- Entre 15 y 40 años
- Género primario de juegos consumidos: juegos de tiros en primera persona
- Tiene un grado de interés en la militaría
- Está familiarizado con el concepto de empresas militares privadas

A continuación, se exponen varios perfiles hipotéticos de usuario:

Usuario 1:

Un chico de 16 años en formación académica (ESO/Bachillerato) usa su extenso tiempo libre para jugar a juegos militares o de tiros en primera persona. Ve que los juegos que consume normalmente tienen mucho *downtime* entre partidas e invierte este tiempo en usar su móvil para ver contenido rápido como *Tik Tok*, *Youtube Shorts* o *Instagram Reels*. Este busca una distracción que le mantenga más involucrado, pero tampoco hasta el punto de que tenga que dedicarle más atención al móvil que al juego principal que está consumiendo. Aunque no tiene una pasión o alto interés por la militaría, está muy familiarizado con el escenario de juegos bélicos en tiempos modernos debido a los juegos principales que suele consumir.

Usuario 2:

Un hombre de 30 años trabaja en una oficina bajo un contrato de jornada completa. Su sitio de trabajo se encuentra a más de media hora en transporte público desde su casa, por lo tanto, se encuentra con un tiempo muerto el cual quiere rellenar con una actividad. Éste busca juegos en las tiendas de aplicaciones en su móvil y ve que la mayoría son juegos que requieren mucha atención y dedicación por su parte o juegos que tienen un estímulo muy bajo para él y sabe que le aburrirán en poco tiempo. Debido a su interés por el mundo de la militaría, invierte su tiempo libre durante los fines de semana jugando en línea con sus amigos a simuladores militares, los cuales suelen tener bastante *downtime* entre que es abatido y puede participar de nuevo, y un juego con el que se pueda durante este tiempo también es algo que desea.

Usuario 3:

Un hombre de 25 años universitario ha empezado a desarrollar un interés por la militaría, siguiendo noticias de conflictos alrededor del mundo y a jugar a juegos que tengan temáticas parecidas, pero se encuentra con dificultades para introducirse, como el precio de entrada, el

nivel de habilidad de otros jugadores, etc. Su tiempo libre está bastante limitado debido a sus estudios y trabajo a media jornada, así que no dispone de muchas horas de ocio para desarrollar sus habilidades ni del dinero para comprar juegos de alta fidelidad.

2.3. Estado del arte

Usando el conocimiento de juegos existentes por parte del autor y la búsqueda de productos parecidos al proyecto propuesto por plataformas de distribución de juegos, se han recopilado varios juegos de diferentes plataformas que contienen (en algunos casos de forma parcial, en otros de forma completa) ideas de diseño que pueden aportar enriquecimiento al proyecto.

2.3.1. Fuentes de inspiración

Metal Gear Solid: Peace Walker (MGS: PW) y Metal Gear Solid V: The Phantom Pain (MGS V)

En ambos juegos, dirigidos por Hideo Kojima y desarrollados por Kojima Productions, el personaje principal Big Boss (*Snake*) controla una PMC (conocida como *Militaires Sans Frontières* en PW o *Diamond Dogs* en MGS V) desde una instalación parecida a una plataforma petrolífera llamada *Mother Base*. Dicha *Mother Base* tiene todo tipo de instalaciones, como barracones, hangares, galerías de tiro, enfermerías, un espacio de I+D, un centro de inteligencia, un comedor e incluso un calabozo.



Figura 8: *Mother Base* en *Metal Gear Solid: Peace Walker* (Metal Gear Wiki)



Figura 9: *Mother Base* en *Metal Gear Solid V: The Phantom Pain* (Metal Gear Wiki)

Para que estas facilidades funcionen, *Mother Base* requiere soldados asignados a diferentes departamentos. El jugador adquiere más personal para la base a través de la jugabilidad principal: las misiones de combate. El jugador, en vez de matar a los soldados enemigos que se encuentra en sus misiones, los puede noquear con balas tranquilizantes, con combate cuerpo a cuerpo o incluso hacer que se rindan. Entonces el jugador puede escanear dicho soldado para ver en qué departamento será más competente (I+D, combate, médico, inteligencia, etc.) con una asignación de letras entre E hasta A (de menos competente a más competente) en orden alfabético, con unos pocos siendo especialmente competentes, asignando el rango S. Estos también pueden tener una habilidad específica, como hablar un cierto idioma, ser más habilidoso con un tipo de arma, tener más puntos de vida, etc.



Figura 10: Vista de personal en espera en MGS V ("*Comprehensive Staff Guide*" por Caihead - Steam)

Cuanto más personal sea asignado a dichos departamentos, el jugador desbloquea nuevo equipamiento, más habilidades, expansión y mejora de las instalaciones, etc. Este equipamiento, más las mejoras y expansiones, requieren recursos (como metales comunes, metales menores, fuel) y una moneda conocida como GMP (o *Gross Military Product*). El jugador adquiere estos elementos a través de cumplir misiones, robar recursos de los enemigos o enviando a sus soldados a misiones. Estas misiones varían en duración, entre unas horas o incluso días, y pueden incluir ciertas limitaciones, como que uno de los soldados tenga una cierta habilidad, que todos tengan una valoración específica o superior, e incluso requieran un vehículo en concreto (el jugador puede adquirir vehículos de combate durante las misiones como recompensa o extrayéndolos y enviándolos a su base). Estas misiones recompensan al jugador con altas cantidades de recursos o con desbloques de nuevo equipamiento.

Se ha mencionado varias veces la inspiración de estos dos juegos sobre el proyecto propuesto, debido al énfasis que tienen en la creación y mantenimiento de una empresa militar privada, pero esta mecánica acaba siendo secundaria ya que la jugabilidad principal es la infiltración y espionaje por parte del jugador, por lo tanto, una extracción de esta mecánica y su introducción como un juego *standalone* acabaría con una experiencia de usuario decepcionante.

Como propuesta para el proyecto, las mecánicas de mejora y expansión de la base para desbloquear y expandir habilidades, el sistema de puntuación sobre las habilidades generales de los soldados, habilidades específicas y cómo pueden afectar en sus roles, el sistema de recursos y moneda y, por último, el sistema de enviar a soldados en misiones (con posibles limitaciones para forzar al jugador a crear un equipo acorde a la misión) a cambio de recursos serán piezas fundamentales del juego que se propone desarrollar.

XCOM: Enemy Unknown / 2

Este juego de estrategia y combate por turnos inspirado en el juego de estrategia *UFO: Enemy Unknown* (en EUA conocido como *X-COM: UFO Defense*) de los 90, desarrollado por *Firaxis Games*, contiene muchos elementos que podrían ser útiles para tener en cuenta al planear el desarrollo del proyecto propuesto.

El juego consiste en dirigir a tus soldados de élite contra una invasión alienígena de la Tierra. Entre misiones, el jugador deberá gestionar sus recursos para desarrollar nuevas tecnologías para luchar contra alienígenas, mantener su base y sus finanzas.

Se pueden apreciar muchos paralelismos con el proyecto ya propuesto, y un aspecto en concreto que el equipo de desarrollo desearía introducir sería la customización de equipamiento de cada soldado. En XCOM, cada soldado tiene una armadura, un arma principal, un arma secundaria y dos ítems de utilidad. Estos ítems no sólo suben los puntos de vida de los soldados o hace que hagan más puntos de daño, también pueden tener habilidades únicas, como las armaduras W.A.R. o el traje EXO, los cuales permiten usar armas pesadas.



Figura 11: Inventario de un soldado en XCOM 2 ("Complete Soldier Guide" por Spin - Steam)

En la propuesta del juego, aparte de ítems de mejora de los soldados o diferentes armas o armaduras que suban los puntos de daño o los puntos de vida, también se podrían introducir habilidades únicas en ciertos ítems de equipamiento, lo cual haría el sistema más profundo y atractivo.

Otro elemento de estos juegos son las relaciones entre las fuerzas XCOM y las diferentes naciones de la Tierra. Si el jugador ignora las misiones de una nación, estos pueden sucumbir a la invasión alienígena y supone un recorte de fondos para las fuerzas XCOM. Esta relación entre naciones y las fuerzas que controla el jugador es otra mecánica fundamental que se planea introducir en el proyecto propuesto.

Rebel Inc.

Rebel Inc. es un juego de estrategia multiplataforma desarrollado por *Ndemic Creations*, los creadores de *Plague Inc.* En este juego, en vez de esparcir una plaga que acabe con la población mundial, el objetivo del jugador es estabilizar una región después de que una gran guerra haya ocurrido en varios territorios. Este deberá usar un presupuesto mensual y gestionarlo para minimizar los problemas que ocurran en esta nación y derrocar cualquier fuerza insurgente que aparezca.



Figura 12: Captura del juego *Rebel Inc.* (*Ndemic Creations – Steam*)

Con el paso del tiempo y dependiendo de las medidas y decisiones del usuario, el juego genera eventos relacionados con dichas decisiones. Por ejemplo, si en una zona no da apoyo económico, de está podrá generarse una fuerza insurgente. O si se establecen fuerzas de mantenimiento de la paz durante demasiado tiempo, la felicidad de la zona irá disminuyendo. Incluso la construcción de infraestructura puede generar eventos, como un ataque terrorista, malversación de fondos o corrupción, etc.

Esta relación entre decisiones que toma el jugador y eventos que pueden ocurrirle serán útiles para hacer que cada *playthrough* sea única y fomente en el jugador la toma de diferentes decisiones durante cada una de ellas, así que se plantea introducir en alguna versión posterior del juego.

Girls' Frontline

Girls' Frontline es un juego de estrategia para móvil desarrollado por *Mica Team*, donde el jugador es el comandante de una empresa militar privada en el año 2062, después de una tercera guerra mundial, donde la radiación de las bombas usadas ha forzado a la humanidad a luchar sus combates con androides. El objetivo del jugador es controlar escuadrones de androides contra diferentes fuerzas enemigas para cumplir ciertos contratos y defenderse de fuerzas que quieren ver su extinción. El combate es por turnos y el objetivo del jugador es mover sus escuadrones por un campo de batalla predefinido y debe cumplir uno o varios objetivos (normalmente, capturar la base enemiga o eliminar todos los enemigos en el campo de batalla).



Figura 13: Captura propia del menú principal de *Girls' Frontline*

Como se puede apreciar en la esquina superior derecha de la figura anterior, el jugador tiene 4 tipos de recursos: *manpower* (mano de obra), *ammunition* (munición), *rations* (raciones) y *parts* (partes). La tarea principal del jugador es gestionar estos recursos para adquirir más personajes, mejorar sus habilidades, expandir su base, etc. Estos recursos se consiguen principalmente a través de misiones de logística, las cuales son misiones secundarias que no requieren ninguna acción por parte del jugador, estas simplemente requieren asignar un escuadrón y se completan solas después de un período de tiempo definido.

Las recompensas de estas misiones pueden doblarse dependiendo de un porcentaje que el juego muestra al jugador. Si el escuadrón asignado tiene un nivel muy alto, esta probabilidad aumentará.



Figura 14: Ejemplo de misiones de logística en *Girls' Frontline*



Figura 15: Un escuadrón a punto de llevar a cabo una misión de logística.

Marcado en rojo se ven los requerimientos de misión y la probabilidad de doblar la recompensa.

La probabilidad de doblar la recompensa de una misión dependiendo de la experiencia del escuadrón es una herramienta muy interesante para fomentar la mejora de un escuadrón dentro del proyecto que se propone.

Por otra parte, el juego tiene una manera muy efectiva de mostrar la dificultad de una misión y a la vez mostrar la capacidad de combate de un escuadrón. Esto se cumple con una unidad de medida conocida como *Combat Effectiveness* (Eficacia de Combate), un número que

aumenta dependiendo de varios factores, como el nivel de cada personaje, el nivel de sus habilidades, sus estadísticas base y el equipamiento que llevan.



Figura 16: Un ejemplo de un escuadrón del juego *Girls' Frontline*
Los rectángulos rojos indican la efectividad de combate total y de cada uno de los personajes

Este número también es útil para informar al usuario sobre la dificultad de una misión, indicado como “*STRENGTH*” en la selección de misiones, siendo el valor recomendado de CE de uno o más escuadrones para poder completar la misión.

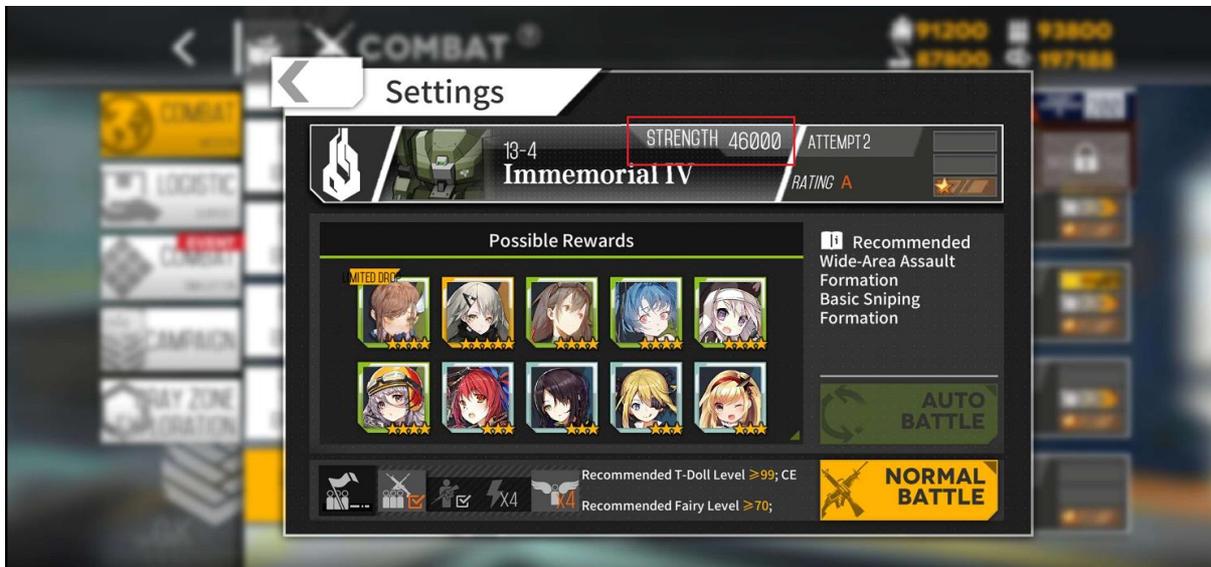


Figura 17: Ejemplo de una misión en el juego *Girls' Frontline*
Marcado en rojo el valor *STRENGTH* que informa al jugador de la dificultad del nivel

El concepto de CE como medida de poder de un escuadrón y dificultad de una misión será fundamental en los mecanismos del juego que se plantea realizar.

2.3.2. Análisis de productos parecidos

Las características del desarrollo del proyecto (equipo pequeño, publicación independiente) definen el producto resultado de este proyecto como un producto *indie*, con un coste de producción muy bajo. Hay un sitio web cuyo objetivo principal es la publicación libre de pequeñas producciones *indie* de videojuegos, entre otros elementos multimedia: itch.io.

Teniendo en cuenta que este es la competencia más inmediata, se ha llevado a cabo una investigación en este sitio web sobre juegos subidos por otros usuarios con conceptos parecidos. Estos se han escogido a través del uso específico de etiquetas en el género de estrategia y comparando la temática con el producto propuesto.

PMC Builder

De los primeros juegos que se han encontrado con una temática muy parecida es el juego “*PMC Builder*” por el desarrollador conocido como *Improper Works*. Este parece ser un juego *clicker*, un género de juego que trata sobre la generación de un recurso a través de pulsar un botón de forma repetitiva, usando dicho recurso para comprar mejoras y *auto clickers* para adquirir aún más de este recurso por segundo. Un ejemplo primordial sería el juego *Cookie Clicker* por el desarrollador Orteil.

La premisa de *PMC Builder* es relativamente sencilla: el jugador debe acrecentar su empresa militar privada con el dinero que genera a través de una mecánica *clicker*, construyendo instalaciones, mejorándolas y contratando soldados para enviarlos en misiones alrededor del globo.

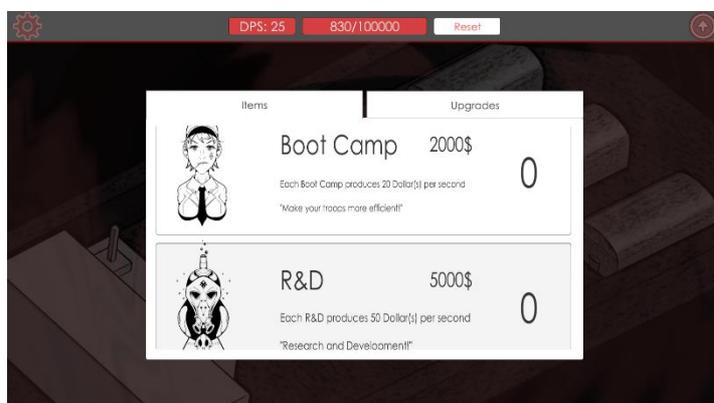


Figura 18: Captura de pantalla de *PMC Builder* (*Improper Works* – itch.io)

La única versión disponible es una primera versión conceptual, con solo una mecánica *clicker* integrada con poca profundidad. El uso de este recurso para acrecentar tu ejército solo se aplica en la mejora de tu base, pero parece ser que el juego principal de enviar tropas y llevar a cabo misiones aún

no se ha desarrollado, lo cual es un poco decepcionante al tener en cuenta que el desarrollador obliga a cualquier usuario que quiera probar el juego a pagar 70 céntimos por lo que es esencialmente un concepto sin ninguna sustancia.

Un par de puntos que merecen destacarse a favor del desarrollador serían la transparencia de los planes de desarrollo futuros y los gráficos que acompañan a los ítems y mejoras, los cuales están muy bien estilizados por un dibujante de gran habilidad. Desgraciadamente, viendo que las últimas actualizaciones en su perfil se hicieron hace 7 años y el juego no ha recibido más versiones desde la primera versión conceptual que se lanzó hace 8 años, podemos asumir que el desarrollo de este proyecto ha sido abandonado.

Uprising

Este juego de estrategia, desarrollado por el usuario BLAKE, es parecido en concepto al producto propuesto, pero en vez de contratar soldados para una empresa militar privada, el jugador ha de crear una revolución en un país a través de varias herramientas de influencia, desde instrumentos pacíficos, como folletos propagandísticos y protestas, hasta actos terroristas, como asesinatos de oficiales o detonación de explosivos. El jugador debe gestionar el dinero que posee y las relaciones con otros países para garantizar que su fuerza revolucionaria tenga éxito.

El objetivo final es convertir todo el país elegido a la ideología del jugador sin irse a bancarrota.

El juego también tiene eventos aleatorios a medida que pasa el tiempo dentro del juego, los cuales pueden afectar las decisiones (o la disponibilidad de estas) del jugador.

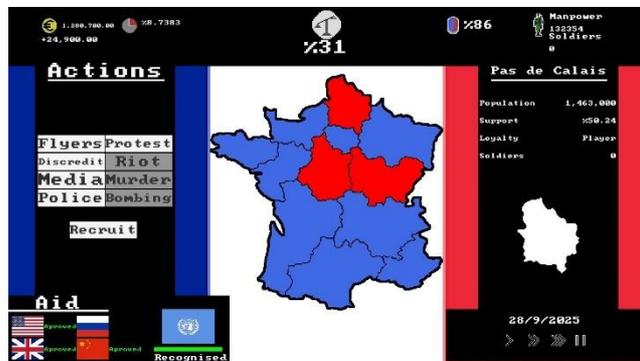


Figura 19: Captura de pantalla del juego *Uprising* (BLAKE – itch.io)

Este es un juego mucho más completo que el estudiado anteriormente, con una estética retro que recuerda a los juegos disponibles en los sistemas MS-DOS de los 80 y 90. Aparte de tener una buena banda sonora y una jugabilidad fácil de seguir, este tiene un problema al mostrar algunos bloques de texto dentro de las acciones a medias, confundiendo al usuario.

PMC-Suite

Este juego de estrategia desarrollado por KaadNet trata de la gestión de las operaciones de una empresa militar privada en un país ficticio conocido como Laguta, el cual se encuentra en medio de una guerra civil. El jugador puede escoger 4 “partidos” a los que dar apoyo: el gobierno de Laguta, un grupo rebelde, un grupo de fanáticos religiosos o las empresas petroleras que operan en el país.

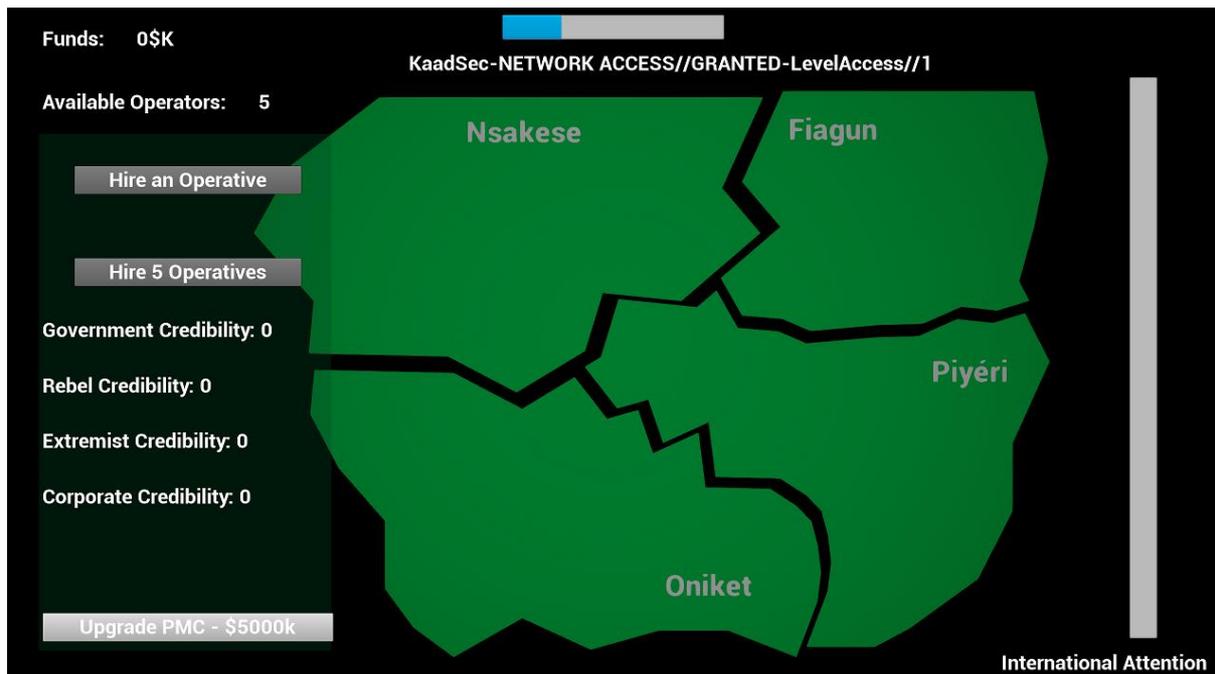


Figura 20: Captura propia del juego *PMC-Suite*

El jugador adquiere credibilidad si da apoyo a una de las 4 facciones en cada zona del país de Laguta. Cuanta más credibilidad tenga, el jugador tendrá acceso a mejores misiones, las cuales le darán mejores recompensas. Hay un medidor de atención internacional, aludiendo a que pueden interferir otros países, pero no parece ser funcional en esta versión del juego.

Como *PMC Builder*, este juego aún se encuentra en el estado de prototipado, con una jugabilidad muy simple y una UI básica. No solo eso, la jugabilidad es extremadamente frenética, en el sentido que cada 5 segundos hay nuevas misiones, eventos aleatorios, etc., lo cual puede agobiar al jugador rápidamente. Teniendo en cuenta que el juego no ha sido actualizado en los últimos 3 años, este también se puede considerar un proyecto abandonado.

A war of a mad man's making

Este juego de estrategia / novela visual desarrollado por *Witching Metal Productions* tiene mucho más énfasis narrativo que los proyectos mencionados anteriormente. El jugador se pone en la piel de un mariscal de campo de una nación ficticia bajo el régimen dictatorial de una figura conocida como el "Gran Líder".

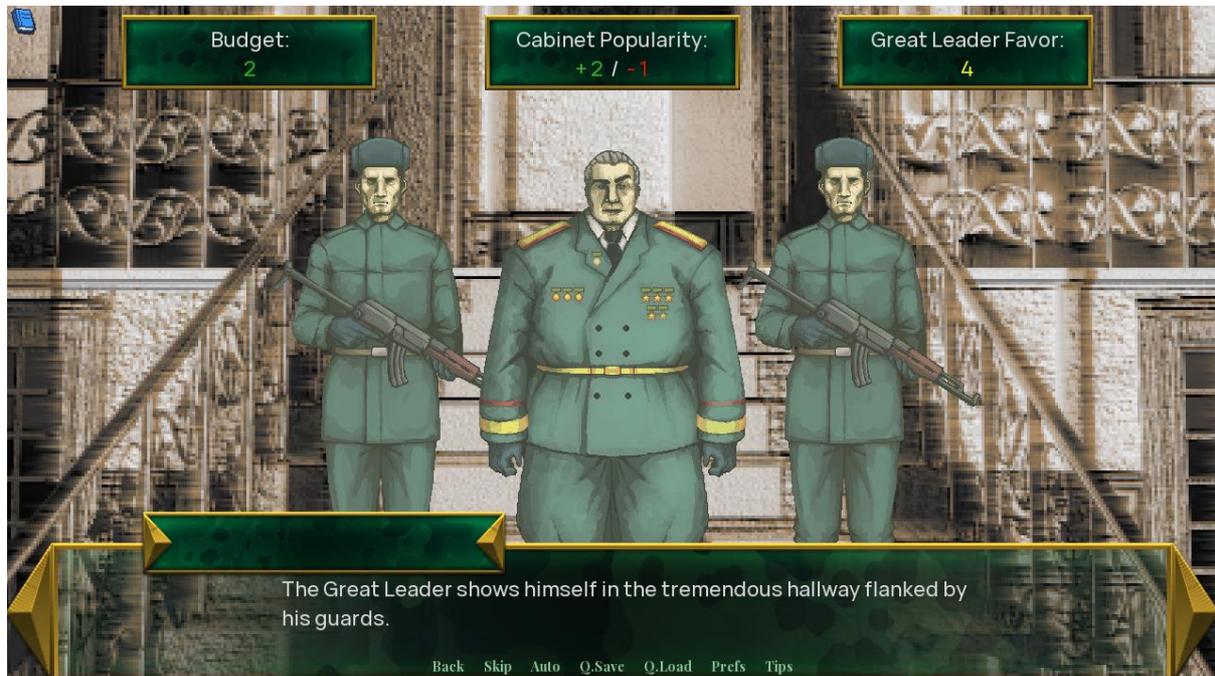


Figura 21: Captura de pantalla del juego A war of a mad man's making (*Witching Metal Productions* – *itch.io*)

El jugador tiene que gestionar el presupuesto, su reputación y favores del Gran Líder para sobrevivir el conflicto bélico a punto de estallar contra las naciones que le rodean.

El elemento de gestión en este juego es más bien secundario, influenciando el resultado de la historia en 5 diferentes finales, pero principalmente este juego es una novela visual, así que su inclusión en esta investigación puede parecer un poco desconcertante. A pesar de no tener un énfasis en mecánicas de gestión como otros juegos mencionados en este capítulo, es interesante remarcar (aparte del gran detalle gráfico o de la composición musical) algunos conceptos e ideas que pueden ser útiles o parecidos a las ideas ya preestablecidas para el proyecto que se propone.

Principalmente, el dinamismo, materializado en la diversidad de finales y caminos que puede tomar el jugador, dependiendo de sus decisiones, es una muy buena herramienta para

empujar al jugador a acometer otra *playthrough*, un logro que queremos conseguir para crear una retención sana de jugadores.

3.Propuesta

Como se ha expuesto en el capítulo anterior, otros juegos con ideas muy parecidas ya existen en el mercado *indie*, pero estos tienen grandes problemas, como el abandono de desarrollo, falta de mecánicas que den importancia a las decisiones del jugador y confusión general al entrar en un nuevo juego.

Así que la propuesta inicial de **“generar un videojuego de gestión donde el jugador cree y mantenga su empresa militar privada, cumpliendo misiones y contratando soldados mientras gestiona varios recursos, con un concreto final: el jugador tendrá que dar apoyo a una de las tres grandes fuerzas en un conflicto mundial y hacer que ganen la guerra”** se mantiene, con la promesa de mantener el desarrollo del juego, incluyendo mecánicas que tengan al jugador involucrado en el producto y con tutoriales e información para asistir con cualquier duda que tenga.

3.1. Especificaciones del producto

El producto final debería tener las siguientes especificaciones para cumplir con la visión del equipo de desarrollo:

Especificaciones básicas:

- Juego *offline*
- Multiplataforma (Android y PC)
- Gestión de recursos y moneda
- Expansión de la base del jugador, usando los recursos adquiridos a través de las misiones para desbloquear nuevas habilidades o mejorar existentes.
- Definición de tres facciones que interactúen entre ellas, cada una con características y objetivos propios
- Tienda donde el jugador puede usar sus recursos para adquirir nuevos soldados

Especificaciones futuras:

- Desarrollar la tienda extensivamente, donde el jugador puede usar sus recursos para adquirir equipamiento e ítems de mejora para sus soldados
- Historia dinámica dependiendo de las decisiones del jugador
- Customización de personajes (cambio de nombre y equipamiento)
- Sistema de eventos aleatorios que afecten de manera positiva (o negativa) al jugador para hacer el juego más difícil
- Sistema de guardado de partida y múltiples *save files*
- Disponibilidad de idioma entre español e inglés

3.2. Modelo de negocio

Hasta que el juego no se pueda completar por parte del jugador, el equipo de desarrollo propone un sistema de donaciones estilo *tip jar* proporcionado por terceros, como la plataforma Ko-fi o el sistema de donaciones integrado en *itch.io* al descargar el juego. Otro sistema a tener en cuenta es el de suscripción, como *Patreon*, el cual da acceso a portales específicos donde se puede compartir el desarrollo del juego y versiones del juego por una cantidad mensual.

Cuando el desarrollo del juego haya avanzado lo suficiente para considerarse un producto completo, una *milestone* que se considerará cumplida cuando el juego tenga un *endpoint* de victoria y derrota, se considera poner esta versión detrás de un pago fijo en *itch.io*. Esta versión se seguirá actualizando con más contenido, *bugfixes*, etc.

3.3. Estrategia de marketing

Debido a la naturaleza del proyecto y su bajo presupuesto, la herramienta principal de marketing será la difusión boca a boca en círculos en línea relacionados con el desarrollador y otros conocidos. Se plantea usar la presencia en línea existente para propagar el desarrollo del juego bajo un pseudónimo para mantener la privacidad. Se ofrecerá al posible jugador la habilidad de poder probar el juego en sus fases de desarrollo y consultar en redes sociales, como *Twitter*, donde se subirán actualizaciones del desarrollo del juego.

4. Diseño

4.1. Entorno de desarrollo y otras herramientas

Una de las decisiones más importantes y no precisamente fácil al crear un juego es escoger el entorno de desarrollo adecuado. Hay tantas opciones, como motores gráficos creados específicamente para el desarrollo de juegos como *Unity* o *Unreal Engine*, o elementos más complejos, como usar una estructura HTML5. Todas estas herramientas tienen sus pros y sus contras, pero, para este proyecto, se ha escogido el motor gráfico *Godot*, un motor relativamente poco conocido comparado con los grandes nombres mencionados anteriormente, pero que aporta un gran número de beneficios.

A modo de contexto, *Godot* es un motor gráfico multiplataforma de código abierto creado por dos argentinos, Ariel Manzur y Juan Linietsky, y ha sido mantenido por la propia comunidad de *Godot*.

Este motor gráfico se diferencia de los demás por su simplicidad, ya que usa un sistema de bloques y *scripts* muy intuitivo, usando su propio lenguaje conocido como *GDScript*, inspirado en *Python* y otros lenguajes parecidos. No sólo eso, este motor gráfico está altamente optimizado, por lo tanto, el producto resultante funcionará en la mayoría de dispositivos de gama baja y media. Hablando de dispositivos, un proyecto en *Godot* es capaz de generar productos finales para Linux, Windows, macOS, HTML5 y Android con ajustes mínimos. [12] (Godot, 2024)

Aparte de todos estos beneficios, el equipo de desarrollo ya tiene experiencia en este motor gráfico, habiendo creado proyectos anteriormente como *hobby* y a cargo de empresas, por lo tanto, esta era la mejor opción.

Los requerimientos mínimos de *Godot* para ordenador son los siguientes:

Tabla 2: Requerimientos mínimos de Godot en ordenadores y portátiles [13] (Godot, 2024)

| | |
|------------|--|
| <p>CPU</p> | <ul style="list-style-type: none"> - Windows: x86_32 CPU con instrucciones SSE2, o cualquier x86_64 CPU <i>Ejemplo: Intel Core 2 Duo E8200, AMD Athlon XE BE-2300</i> - macOS: x86_64 o ARM CPU (Apple Silicon) <i>Ejemplo: Intel Core 2 Duo SU9400, Apple M1</i> - Linux: x86_32 CPU con instrucciones SSE2, x86_64 CPU, ARMv7 o ARMv8 CPU <i>Ejemplo: Intel Core 2 Duo E8200, AMD Athlon XE BE-2300, Raspberry Pi 4</i> |
| <p>GPU</p> | <ul style="list-style-type: none"> - Método de renderizado Forward+: Gráficos integrados con soporte total de Vulkan 1.0 <i>Ejemplo: Intel HD Graphics 5500 (Broadwell), AMD Radeon R5 Graphics (Kaveri)</i> - Método de renderizado Mobile: Gráficos integrados con soporte total de Vulkan 1.0 <i>Ejemplo: Intel HD Graphics 5500 (Broadwell), AMD Radeon R5 Graphics (Kaveri)</i> - Método de renderizado Compatibility: Gráficos integrados con soporte total de OpenGL 3.3 <i>Ejemplo: Intel HD Graphics 2500 (Ivy Bridge), AMD Radeon R5 Graphics (Kaveri)</i> |

| | |
|-------------------|---|
| RAM | <ul style="list-style-type: none"> - Editor nativo: 4GB - Editor web: 8GB |
| Almacenamiento | 200MB (usados para el ejecutable, archivos del proyecto y caché). Exportar proyectos requiere descargar plantillas de exportación por separado (1.3GB después de la instalación). |
| Sistema operativo | <ul style="list-style-type: none"> - Editor nativo: Windows 7, macOS 10.13 (Compatibility) or macOS 10.15 (Forward+/Mobile), Distribuciones de Linux lanzadas después de 2016 - Editor web: Firefox 79, Chrome 68, Edge 79, Safari 15.2, Opera 64 |

Tabla 3: Requerimientos recomendados de Godot en ordenadores y portátiles [11] (Godot, 2024)

| | |
|-----|--|
| CPU | <ul style="list-style-type: none"> - Windows: x86_64 CPU con instrucciones SSE4.2, con 4 núcleos físicos o más <i>Ejemplo: Intel Core i5-6600K, AMD Ryzen 5 1600</i> - macOS: x86_64 o ARM CPU (Apple Silicon) <i>Ejemplo: Intel Core i5-8500, Apple M1</i> - Linux: x86_32 CPU con instrucciones SSE2, x86_64 CPU, ARMv7 o ARMv8 CPU <i>Ejemplo: Intel Core i5-6600K, AMD Ryzen 5 1600, Raspberry Pi 5 con overclocking</i> |
| GPU | <ul style="list-style-type: none"> - Método de renderizado Forward+: Gráficos dedicados con soporte total |

| | |
|-------------------|---|
| | <p>de Vulkan 1.2 <i>Ejemplo: NVIDIA GeForce GTX 1050 (Pascal), AMD Radeon RX 460 (GCN 4.0)</i></p> <ul style="list-style-type: none"> - Método de renderizado Mobile: Gráficos dedicados con soporte total de Vulkan 1.2 <i>Ejemplo: NVIDIA GeForce GTX 1050 (Pascal), AMD Radeon RX 460 (GCN 4.0)</i> - Método de renderizado Compatibility: Gráficos integrados con soporte total de OpenGL 4.6 <i>Ejemplo: NVIDIA GeForce GTX 650 (Kepler), AMD Radeon HD 7750 (GCN 1.0)</i> |
| RAM | <ul style="list-style-type: none"> - Editor nativo: 8GB - Editor web: 12GB |
| Almacenamiento | <p>1.5 GB (utilizados para el ejecutable, los archivos del proyecto , las plantillas de exportación y el caché)</p> |
| Sistema operativo | <ul style="list-style-type: none"> - Editor nativo: Windows 10, macOS 10.15, las distribuciones Linux lanzadas a partir del 2020 - Editor web: Últimas versiones de Firefox, Chrome, Edge, Safari, Opera |

Para la creación de elementos gráficos como *sprites*, se ha usado Adobe Illustrator, el cual permite crear elementos gráficos en forma vectorial que pueden adaptarse a diferentes tamaños en píxeles.

4.2. Arquitectura de la información

En *Godot* el desarrollo de un juego consiste en varias pantallas conocidas como “escenas”, y cada escena contiene elementos como botones, *labels*, bloques de texto, elementos gráficos, etc. que se conocen como nodos. Cada escena y nodo puede tener su propio *script*, el cual asiste en añadir interactividad y comunica acciones y reacciones entre todos los elementos involucrados. Las escenas se guardan en el formato *.tscn* y los *scripts*, en el formato *.gd*.

A continuación, se incluye un diagrama de los elementos generales:

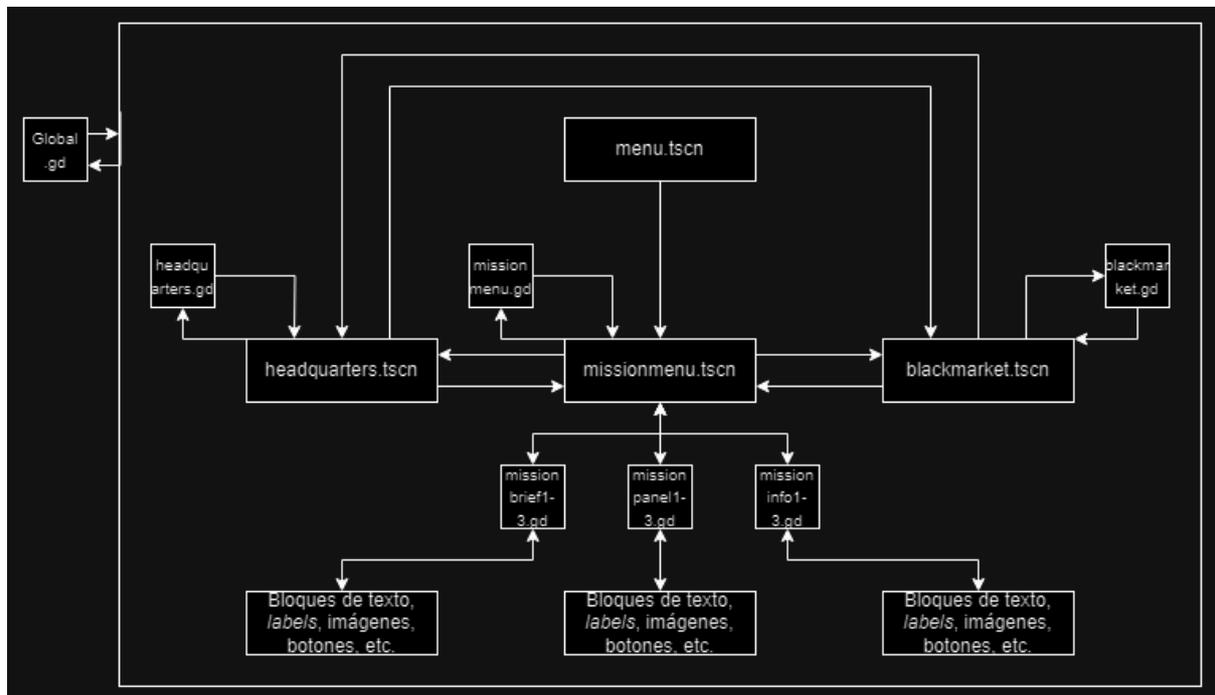


Figura 22: Diagrama propio de los elementos generales del juego

Cada escena incluye una pantalla diferente y, dentro de cada pantalla, un componente puede tener un *script* asociado. Hay que tener en cuenta que todos los elementos están conectados a un *script* llamado *Global.gd*, el cual almacena datos importantes, como la cantidad de recursos, los escuadrones, la lista de misiones, etc.

- “menu.tscn” es una simple pantalla principal, la cual sirve para albergar el título del juego, el botón de empezar el juego y más adelante la información de salvado de juego y otras saves.
- “missionmenu.tscn” es la pantalla principal del juego, incluye las misiones disponibles, la reputación con diferentes facciones, el progreso del conflicto, el contador de recursos, etc. Desde esta pantalla se lanzan las misiones, escogiendo el escuadrón más adecuado, indicado por un número que indica el porcentaje de victoria de la misión dependiendo del nivel del escuadrón.
- “headquarters.tscn” incluye las instalaciones del jugador, el cual puede mejorar para desbloquear más habilidades o mejorar las existentes. También podrá consultar sus soldados, asignarlos a escuadrones y usar ítems que ha adquirido a través de misiones o del mercado negro.
- “blackmarket.tscn” alberga al mercado negro, el cual cada día mostrará diferentes ítems, contratos de soldados y equipamiento para entregar a los soldados. Esta pantalla incluirá un mercader el cual dirá diferentes frases, dependiendo de los eventos del juego.

4.3. Diseño de la interfaz

Antes de empezar la programación del juego, se crearon varios *wireframes* de la mayoría de pantallas posibles para tener una idea tangible de la interfaz que se requeriría para llevar a cabo todas las mecánicas propuestas y usarlos como plantilla para la creación de pantallas.

Cada pantalla tendrá el mismo *header* y *footer*: en la esquina izquierda tendrá su contador de recursos, en el centro tendrá el medidor de progreso de la guerra (cada facción tendrá un color identificativo) y en la esquina derecha un medidor de reputación del jugador ante las tres facciones. En los lados derechos e izquierdos, el jugador podrá alternar entre las tres diferentes pantallas y finalizar la planificación de ese día y proceder al siguiente.



Figura 23: Wireframe propio de la pantalla de misiones



Figura 24: Wireframe propio de consulta de más información de una misión



Figura 25: Wireframe propio de la pantalla de asignación de un escuadrón

En las tres capturas de pantalla anteriores se puede apreciar la pantalla de misiones, donde el jugador podrá ver las misiones disponibles en un cierto día del conflicto. Podrá leer el coste de recursos para llevar a cabo la misión, la recompensa si la misión es exitosa y los requisitos para tener la máxima probabilidad de victoria en el juego. Se crea una pantalla alternativa a la misión que mostrará una pequeña descripción de la misión y los efectos que tendrá sobre el conflicto y las relaciones. Al hacer clic en “asignar escuadrón”, una pantalla secundaria se abrirá que mostrará los escuadrones disponibles y teniendo en cuenta si su HP no está a 0 y su HC (calculado a partir de un valor básico personal de HC por soldado y sumando HC extra dependiendo del nivel de cada habilidad), proporcionará una probabilidad de victoria sobre la misión.

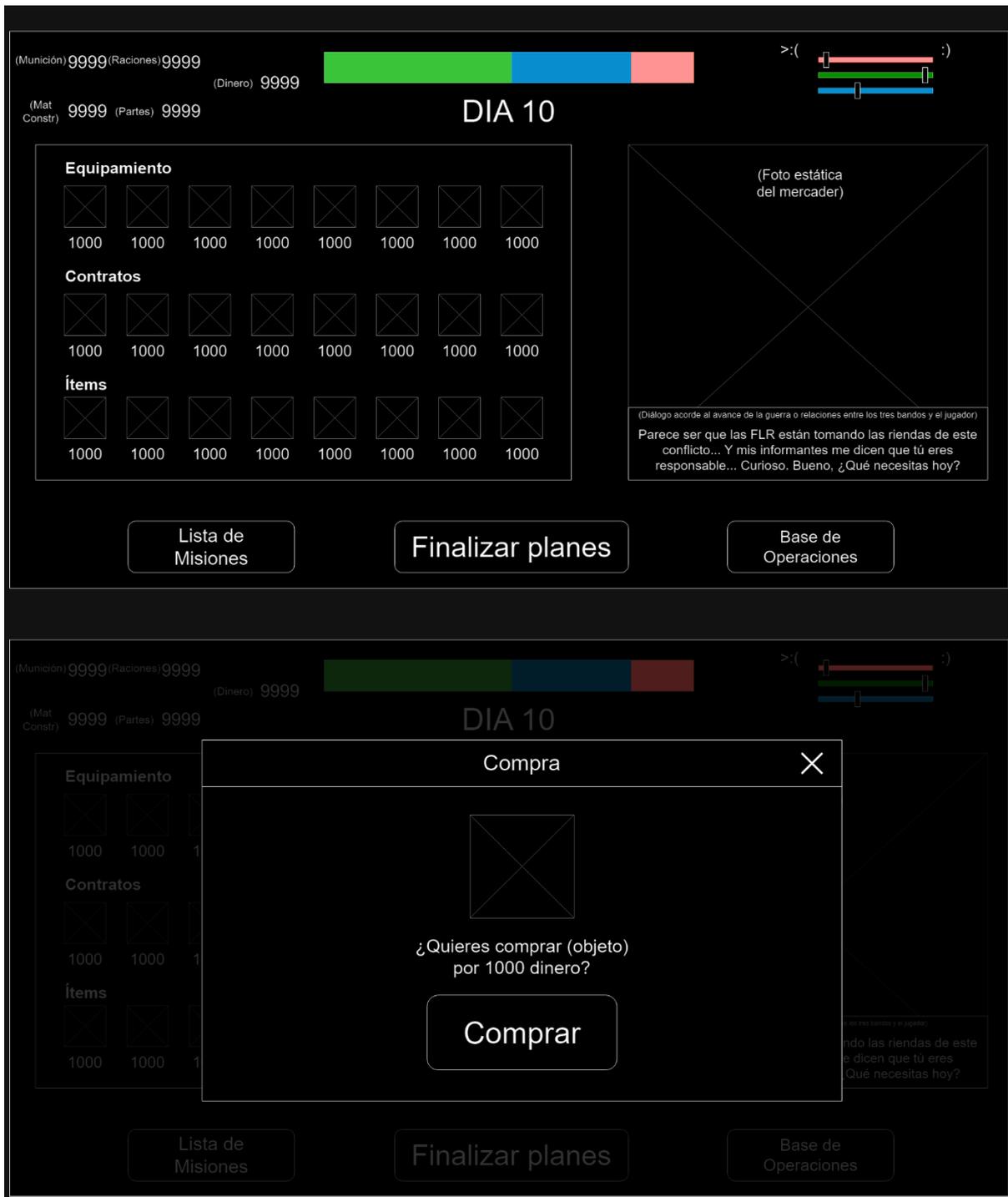


Figura 26: Wireframe propio de la pantalla del mercado negro

La pantalla de Mercado Negro incluye el único NPC del juego, un mercader encapuchado que dará un poco de información sobre el conflicto o hará diferentes comentarios cada vez que lo visites. Este venderá los contratos para tener más soldados y en un futuro, el equipamiento e

ítems que el jugador podrá utilizar para darle más ventaja y poder en sus misiones. Cuando el jugador intente hacer una compra, saltará una pantalla secundaria de confirmación.

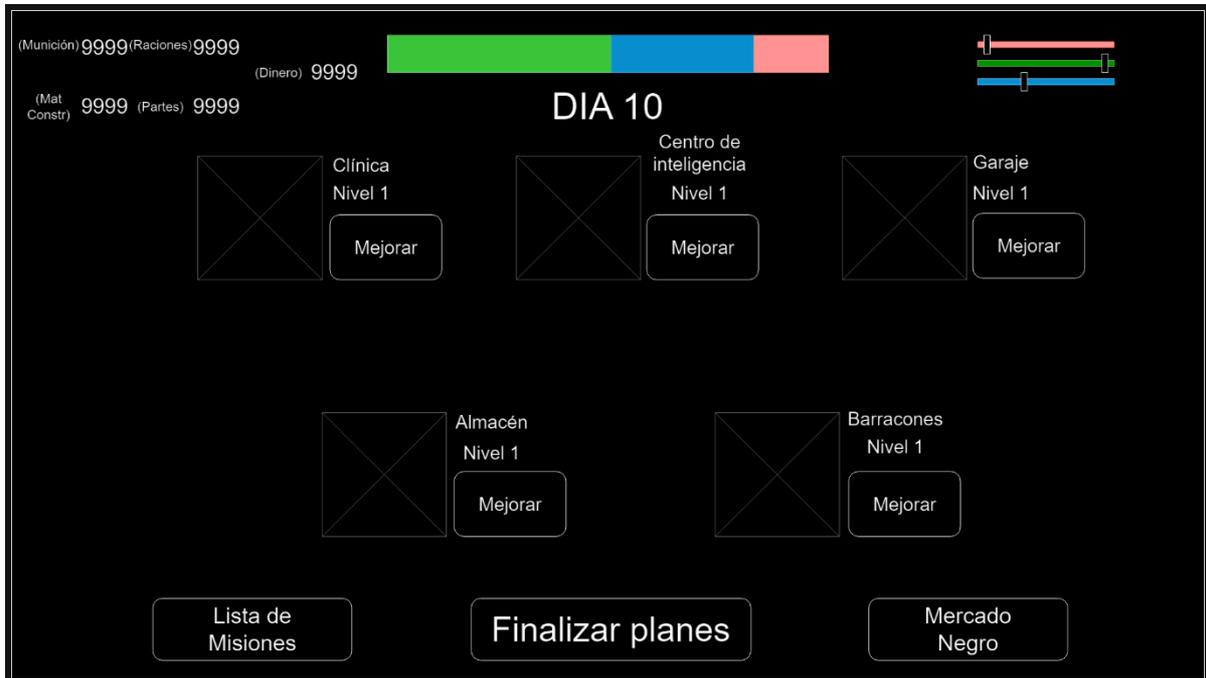


Figura 27: Wireframe propio de la pantalla de Base de Operaciones

La tercera y última pantalla principal incluirá las instalaciones dentro de la Base de Operaciones, donde el jugador podrá invertir sus recursos adquiridos a través de misiones exitosas para llevar a cabo el mantenimiento de su PMC, como curar a sus soldados.

- La clínica servirá para curar a los soldados heridos en combate, usando raciones como pago. Cada vez que pase un día del conflicto, si un escuadrón no está en una misión y tiene soldados heridos, estos recibirán una pequeña cantidad de puntos de vida. Si se mejora la clínica, esta cantidad diaria aumentará.
- El Centro de Inteligencia en un futuro desbloqueará nuevas misiones cuando se mejore y permitirá tener más misiones disponibles por día. Se considerará añadir más funcionalidad, como poder contactar con el líder de cada una de las facciones.
- El Garaje en un futuro tendrá una funcionalidad exacta a la clínica, pero con vehículos. Estos se podrán asignar a escuadrones que lo requieran para misiones más intensas y con mucho más riesgo, a cambio estas tendrán grandes recompensas.
- El Almacén en un futuro servirá para guardar todos los ítems y equipamiento no usado por los soldados. A medida que el jugador mejore su almacén, este podrá albergar una mayor cantidad de ítems y equipamiento

- Los Barracones servirán para gestionar la asignación de soldados en sus respectivos escuadrones o dejarlos en reservas. A medida que el jugador mejore esta instalación podrá tener más soldados en reserva y más escuadrones activos.



Figura 28: Wireframe propio de la pantalla de mejora de instalaciones



Figura 29: Wireframe propio de la pantalla de la clínica



Figura 30: Wireframe propio de la pantalla de los barracones



Figura 31: Wireframe propio de la pantalla del almacén



Figura 32: Wireframe propio de la pantalla del garaje

4.4. Diseño gráfico

El equipo de desarrollo desde un inicio deseaba que el juego tuviera un diseño gráfico que recordase a la era de los juegos de 16 bits, un estilo retro que instigara un sentimiento de nostalgia que muchos jugadores aprecian en el espacio *indie*. Por lo tanto, los varios aspectos de diseño que se van a exponer giran alrededor de esta filosofía.

4.4.1. Sprites y otros elementos interactivos

De momento, el juego incluye varios *sprites* hechos por el equipo de desarrollo en *Adobe Illustrator* y exportados para incluirlos en el juego. Estos *sprites* serían los recursos que el jugador manipulará durante su *playthrough*.



Figura 33: Captura de los *sprites* creados por el equipo de desarrollo

También en cada pantalla se han incorporado dos botones: uno para consultar información sobre las diferentes mecánicas y el contexto general del juego y el otro para ver qué misiones se están llevando a cabo. Debido a la tardía inclusión no planificada de estos botones, se ha optado por *sprites* creados por terceras partes y subidos en línea como imágenes de *stock*.

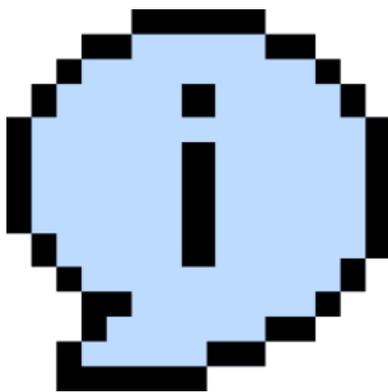


Figura 34: *Info Icon* - Freepik



Figura 35: *Clipboard pixel art* (Bekzod Shoyakubov - Vecteezy)

4.4.2. Logo, banderas y fondos de pantalla

Para la creación del logo del juego, las banderas de las facciones y los fondos de pantalla, se ha usado una amplia gama de imágenes de stock o de la colección de Wikimedia Commons y se han adaptado y modificado en la aplicación GIMP.

Empezando con el logo del juego, se han usado los siguientes gráficos de la librería de Wikimedia:

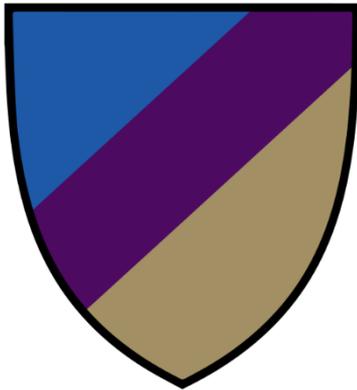


Figura 37: "Fraternitas Livonica" shield (Vladusty - Wikimedia Commons)



Figura 36: Black skull icon (Alexey Gomankov - Wikimedia Commons)

Para crear el siguiente logo:

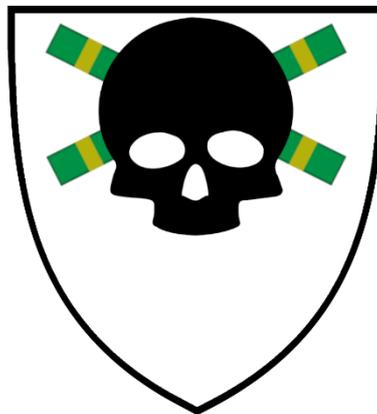


Figura 38: Versión básica/original del logo de PMC Manager

Al tener el logo creado, este se encuentra en muy alta resolución, lo cual no encaja temáticamente con la dirección de estilo propuesta originalmente. Por lo tanto, se usa una herramienta libre en línea llamada "Pixel It" por José Moreira, conocido como "giventofly" en Github.

Con esta herramienta se pueden convertir imágenes normales en *pixel art*, lo cual, teniendo en cuenta la exigua cantidad de tiempo y recursos durante esta fase final de desarrollo, representa el ahorro de una gran cantidad de tiempo que se invertiría en la creación de elementos gráficos desde cero. Por lo tanto, al introducir el logo mostrado anteriormente y escogiendo el *blocksize* de 16, acabamos con este resultado:

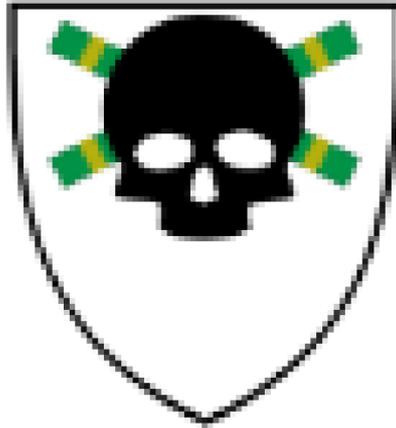


Figura 39: Logo final (sin texto) para el juego PMC Manager

Este proceso con la herramienta “Pixel It” se repetirá para todos los elementos gráficos que vienen a continuación.

La creación de las banderas sigue el mismo patrón, usando imágenes del banco de Wikimedia, con el cambio de que ahora el fondo de cada bandera es el color de cada facción del juego:

Naciones Unidas del Oeste (NUO)



Figura 40: Bandera de la facción NUO dentro del juego PMC Manager

El objetivo detrás del diseño de esta bandera era plasmar el sentimiento principal de la facción dentro del juego: un conjunto de naciones obsesionadas con la justicia, que ven sus fuerzas como jueces de “paz” y ejercen su juzgado a través de la violenta intervención militar enmascarada como una fuerza benevolente y pacífica.

Para esta bandera se han modificado varios elementos para encajarlos en un diseño básico de dos colores y pasado por el filtro de Pixel It.



Figura 41: *Gun outline* (INVERTED - Wikimedia Commons)

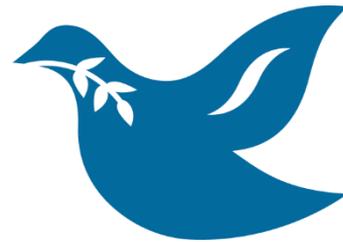


Figura 42: *Wikimania2019 dove icon* (BFlores (WMF) - Wikimedia Commons)



Figura 43: *Balanced scale of Justice* (Perhelion - Wikimedia Commons)

Ejército de los Pueblos del Este (EPE)



Figura 44: Bandera de la facción EPE dentro del juego PMC Manager

Esta bandera ha sido creada desde cero en GIMP, con la misma mentalidad de un diseño simple de dos colores, esta vez, con una iconografía simple con un cierto significado para caracterizar a esta facción. En este caso se quiere representar un sentimiento de unidad de diferentes naciones (las estrellas doradas), creando un nuevo símbolo a partir de la unificación de todas las figuras (el pentágono resultante), empujando un sentimiento de homogeneidad y de colaboración por un “bien mayor”.

Fuerzas de Liberación Rebelde (FLR)



Figura 45: Bandera de la facción FLR dentro del juego PMC Manager

El diseño de esta bandera es el más directo y brusco. Está enfocado a evocar un sentimiento revolucionario, el puño alzado, a través de cualquier medio, especialmente, los violentos (el Kalashnikov).

Para esta bandera se han modificado varios elementos para encajarlos en un diseño básico de dos colores y pasado por el filtro de Pixel It como se ha hecho anteriormente.



Figura 47: AK-47 (Skoll - Wikimedia Commons)



Figura 46: *Fist* (Eugenio Hansen, OFS - Wikimedia Commons)

Para los fondos de pantalla y el único NPC del juego, el responsable del Mercado Negro, se ha usado una mezcla de imágenes *stock* e imágenes de Wikimedia Commons, las cuales, aparte de haber sido transformadas con la aplicación Pixel It, también se han editado de antemano subiendo su contraste y saturación en GIMP para intentar replicar la paleta limitada de colores de un ordenador de 16 bits.



Figura 48: *Brown-dominant "highland" ERDL camouflage pattern swatch* (ERDL - Wikimedia Commons)

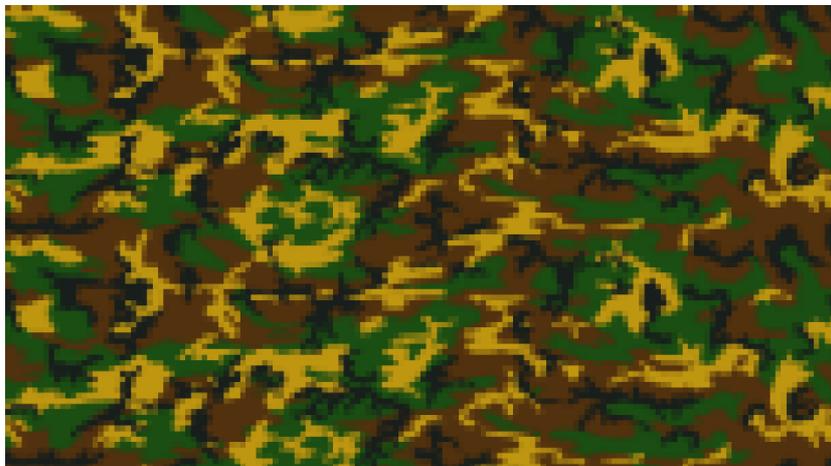


Figura 49: Fondo de pantalla principal del juego PMC Manager



Figura 50: Imagen satélite de una base militar – Planet Labs

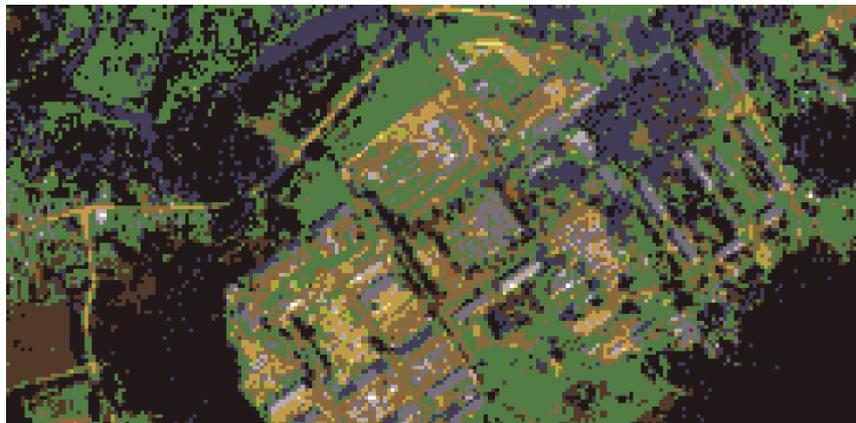


Figura 51: Fondo de la pantalla Base de Operaciones en el juego PMC Manager



Figura 52: Abandoned warehouse. Black and white image. noise scene. (SAYAN - Adobe Stock)



Figura 53: Fondo de la pantalla del Mercado Negro en el juego PMC Manager



Figura 54: *dark mysterious man in a hoodie is hiding his face, hacker, anonymous* (spaxiax - Adobe Stock)



Figura 55: Mercader del juego PMC Manager

4.4.3. Estilo de la UI

Siguiendo la temática militar bajo la perspectiva de un juego *retro* de 16 bits, escogemos la siguiente paleta de colores:



R: 26
G: 78
B: 18
HEX:#1A4E12



R: 80
G: 93
B: 67
HEX:#505D43



R: 61
G: 71
B: 50
HEX:#3D4732



R: 47
G: 56
B: 38
HEX:#2F3826



R: 242
G: 201
B: 0
HEX:#F2C900

El primer y último color serán los colores primarios para la mayoría de los botones y ciertos paneles, el primero corresponde al color de fondo de los elementos mencionados anteriormente y el último será el color del texto dentro de dichos elementos. Los colores restantes son secundarios, usados para los estados de botones y otros paneles.

Esta combinación de colores se ha escogido no solo para hacer mímica de la típica paleta de colores de una caja de munición o suministros militares, sino que también ayuda con el alto contraste que el equipo de desarrollo quiere mantener para remarcar ese sentimiento *retro*.

A la vez, el equipo ha comprometido esta filosofía para facilitar la lectura de los estados de botones, creando una amplia gama de verdes con diferentes tonalidades, y para no abrumar al usuario con una gran saturación de color, la mayoría de las pantallas secundarias usan una gama de grises, blanco y negro para facilitar su lectura y reducir el cansancio de la vista del usuario. Para encajar con la gama cromática del Mercado Negro, la mayoría de los elementos interactivos se han estilizado con blancos y negros.

En todas las pantallas se usa la misma fuente, una fuente gratuita conocida como "*Retro Gaming*", subida por Daymarius en el sitio web DaFont.com. La fuente tendrá en todos los casos un borde negro de 10 píxeles para facilitar su lectura sobre la gran variedad de colores de las tres diferentes pantallas.

5. Demostración

5.1. Prototipos

Durante el desarrollo de este TF, se han entregado dos versiones principales del juego:

- Un prototipo principal conceptual, mostrando la distribución básica de algunas de las pantallas y una versión muy rudimentaria del *gameloop* propuesto.
- Un prototipo bastante más completo, con el *gameloop* más elaborado, es decir, el juego tiene estados de victoria y derrota de una partida. También incluye un diseño con más carácter, más atractivo y definido, con las funcionalidades básicas secundarias para poder completar una partida y con nuevas adiciones para mejorar la comprensión de las acciones del jugador.

5.1.1. Prototipo Lo-Fi



Figura 56: Captura de la pantalla inicial del prototipo básico de PMC Manager



Figura 57: Captura de la pantalla Lista de Misiones del prototipo básico de PMC Manager

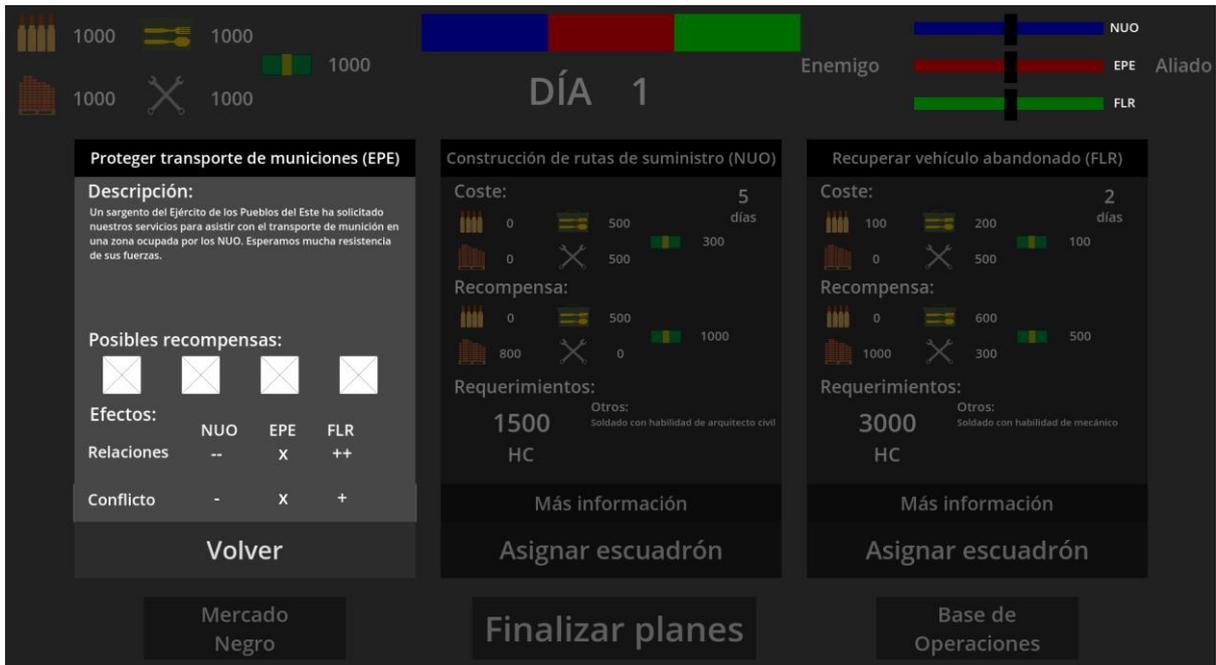


Figura 58: Captura de la pantalla secundaria "Más información" de una misión del prototipo básico de PMC Manager



Figura 59: Captura de la pantalla secundaria “Asignar escuadrón” de una misión del prototipo básico de PMC Manager

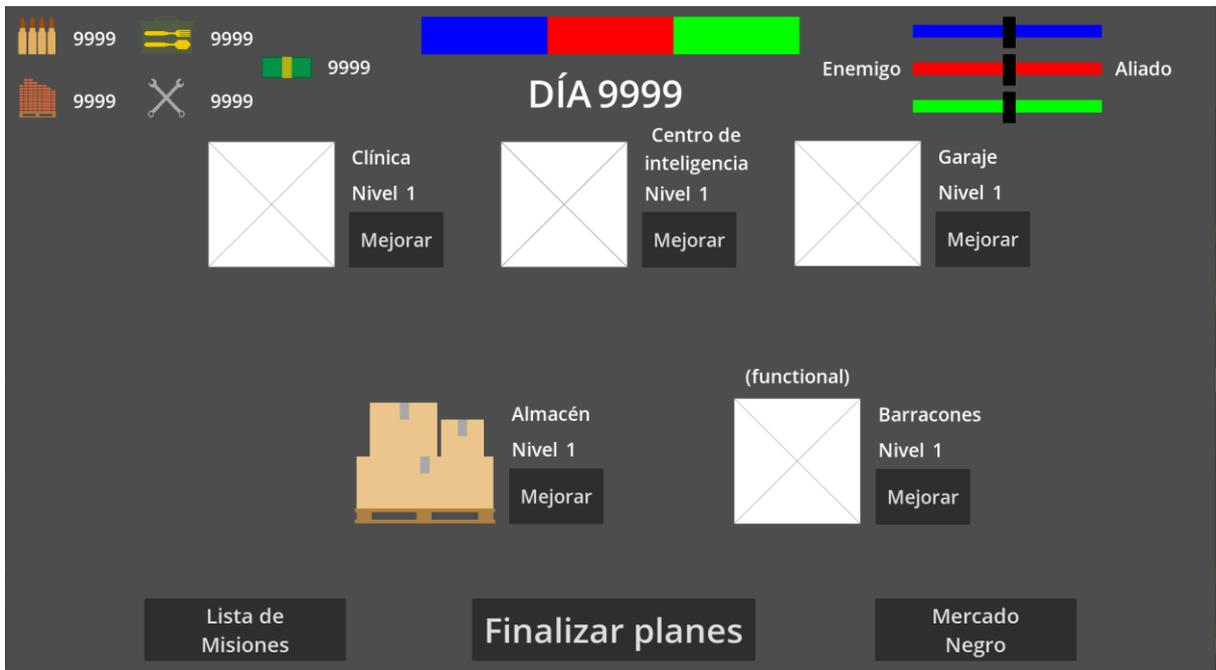


Figura 60: Captura de la pantalla Base de Operaciones del prototipo básico de PMC Manager

5.1.2. Prototipo Hi-Fi



Figura 61: Captura de la pantalla inicial del prototipo avanzado de PMC Manager



Figura 62: Captura de la pantalla secundaria general de "Información" del prototipo avanzado de PMC Manager



Figura 63: Captura de la pantalla "Lista de Misiones" del prototipo avanzado de PMC Manager



Figura 64: Captura de la pantalla secundaria general de "Misiones en curso" del prototipo avanzado de PMC Manager



Figura 65: Captura de la pantalla secundaria “Más información” de una misión del prototipo avanzado de PMC Manager



Figura 66: Captura de la pantalla secundaria “Asignar escuadrón” de una misión del prototipo avanzado de PMC Manager



Figura 67: Captura de la pantalla secundaria general "Informe de estado diario" del prototipo avanzado de PMC Manager



Figura 68: Captura de la pantalla secundaria general "Soldados caídos en combate" del prototipo avanzado de PMC Manager



Figura 69: Captura de la pantalla "Base de Operaciones" del prototipo avanzado de PMC Manager



Figura 70: Captura de la pantalla secundaria "Mejorar instalación" del prototipo avanzado de PMC Manager



Figura 71: Captura de la pantalla secundaria "Clínica" del prototipo avanzado de PMC Manager



Figura 72: Captura de la pantalla secundaria "Barracones" del prototipo avanzado de PMC Manager



Figura 73: Captura de la pantalla "Mercado Negro" del prototipo avanzado de PMC Manager



Figura 74: Captura de la pantalla secundaria "Contratar soldado" del prototipo avanzado de PMC Manager



Figura 75: Captura de la pantalla secundaria general "Finalizar planes" del prototipo avanzado de PMC Manager

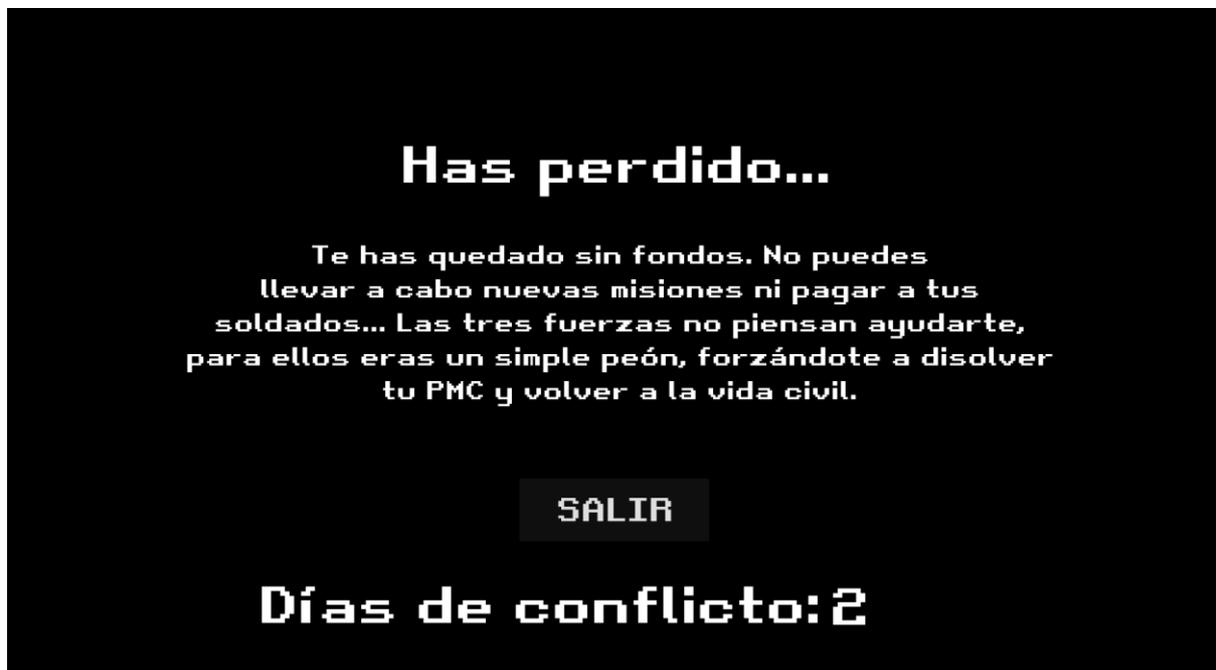


Figura 76: Captura de la pantalla de derrota del prototipo avanzado de PMC Manager

5.2. Guía de usuario

Para asistir al lector de este TF y como introducción al juego desarrollado, se redacta una pequeña guía de usuario para exponer, a través de varias listas de acciones, cómo sería la experiencia de juego:

El jugador quiere enviar un escuadrón en una misión

1. El jugador lee la información disponible en la pantalla general de Lista de Misiones y escoge una de las misiones ofrecidas.
2. El jugador pulsa “Más información” para saber qué consecuencias tendrá llevar a cabo esa misión, como su reputación con las tres facciones o el progreso de la guerra.
3. Si el jugador está convencido de llevar a cabo esta misión, este hará clic sobre el botón “Asignar escuadrón” y escogerá uno de sus escuadrones que tenga soldados. Idealmente escogerá el que más se ajuste a los requerimientos de la misión.
4. Cuando se asigna una misión a un escuadrón, la influencia de reputación es instantánea y ese escuadrón estará “ocupado” hasta que la misión termine.
5. Cuando la misión ha terminado, el usuario recibirá información sobre el resultado en la pantalla de informe diario, incluyendo también sus soldados caídos en combate. Si se da el caso, un soldado caído en combate no se puede curar y se considera perdido para siempre.

El jugador quiere curar a sus soldados

1. El jugador va a la pantalla principal de Base de Operaciones y hace clic en el botón “Clínica”
2. Aquí tendrá un listado de sus soldados en escuadrones y en reservas y podrá curar a sus soldados instantáneamente por un coste de raciones.
 - 1.1. En caso de que el usuario no quiera gastar recursos, puede dejar pasar días ya que cada soldado que no se encuentre ocupado se curará lentamente por sí solo.

El jugador quiere tener un escuadrón desde cero

1. El jugador va a la pantalla principal de Mercado Negro
2. Escoge qué contrato comprará y compra el número de soldados deseado.
3. El jugador navega a la pantalla de Base de Operaciones y hace clic en “Barracones”
4. En la nueva pantalla secundaria, el jugador verá los soldados comprados recientemente en sus reservas.
5. Seleccionando un soldado en concreto y haciendo clic en “Asignar soldado”, este podrá decidir de qué escuadrón formará parte el nuevo soldado, si hay espacio o está desbloqueado.

6. Conclusiones y líneas de futuro

6.1. Conclusiones

Después de varias versiones del juego y semanas de desarrollo, este TF concluye con, en primer lugar, una versión jugable en Android y Windows, donde el jugador es capaz de terminar una partida y experimentar la jugabilidad principal del producto y, en segundo lugar, un tráiler para promocionar el juego.

El proyecto comprendía un grado sustancial de dificultad y complejidad, por lo que el procedimiento de trabajo, con el tiempo, ha ido cambiando con el objetivo de superar obstáculos y adaptarse a los recursos y tiempo del equipo de desarrollo.

Sin embargo, estas dificultades han servido para aprender varias lecciones sobre la planificación de proyectos de desarrollo de software y el propio desarrollo. Entre dichas lecciones, la más obvia sería la filosofía de planificar un proyecto de esta magnitud a largo plazo. Inicialmente, con las metodologías ágiles, el equipo de desarrollo había planificado los 3 meses de duración del TF con altas expectativas, pero, a medida de que el proyecto se iba desarrollando, el equipo era más consciente de la dificultad de cumplir con algunos de los objetivos secundarios, lo cual motivaba la reestructuración de la planificación establecida.

Al principio esta situación se consideraba casi catastrófica, pero el equipo de desarrollo replanteó el tiempo restante sobre una nueva planificación ágil, afrontando las acertadas críticas y valiosas sugerencias del tutor, y se ha podido entregar el proyecto en un estado que se considera satisfactorio, sacrificando tareas secundarias, como el *playtest* y las encuestas, para asegurar el cumplimiento de tareas con más importancia, demostrando la capacidad de adaptabilidad de los sistemas ágiles.

Por lo que se refiere a los objetivos del juego y del jugador, es decir, los mencionados *playtest* y las encuestas, se han tenido que sacrificar por el bien del proyecto. Es decir, para asegurar que los objetivos del juego y personales se cumplieran, se han sacrificado los objetivos para el jugador, o mejor dicho, la demostración de estos. A modo de explicación, se planteaba llevar a cabo durante los últimos *sprints* de la tercera fase un *playtest* para ver si estos objetivos se habían cumplido, pero esta parte del proyecto se eliminó al principio de la tercera fase al

anticipar que, para el desarrollo satisfactorio del proyecto y para cumplir con los objetivos personales y del propio juego, aún se requería la sustancial inversión de recursos y tiempo.

Otra lección a tener en cuenta sobre el propio desarrollo del software es la esencial importancia de conocer y combinar una gran variedad de conocimientos multimedia para llevar a cabo el desarrollo de un videojuego. Desde conocimientos de distribución de UI, conocimientos artísticos de composición y contraste, cromatismo, el correcto uso de tipografías para la facilidad de lectura de texto en pantalla, sin olvidar la programación orientada a objetos y la edición de imagen, audio y vídeo.

En conclusión, un proyecto de esta magnitud ha puesto a prueba todos los conocimientos adquiridos por el equipo de desarrollo durante los últimos 4 años, además de educar al equipo sobre nuevos conceptos y tareas. Por lo tanto, en cuanto a los objetivos del autor propuestos para este TF, estos se consideran más que cumplidos.

6.2. Líneas de futuro

En cuanto a las mejoras a realizar en hipotéticas futuras versiones del proyecto, en el apartado anterior se menciona la exclusión de los *playtests* y encuestas del juego, pero esto no significa que se abandone la idea y el proyecto en general. Como se ha indicado al principio de este documento, el concepto detrás de este juego ha sido una idea que el equipo de desarrollo ha querido llevar a cabo desde hace años, y este TF ha sido solo el principio.

Como se puede apreciar en el subapartado “Objetivos adicionales” en “Objetivos generales” o en el subapartado “Especificaciones futuras” en “Especificaciones del producto”, el equipo de desarrollo plantea varios aspectos a desarrollar de cara al futuro, desde elementos técnicos como estados de guardado, traducción del juego al inglés y más mecánicas para mantener al jugador más involucrado en el juego (eventos aleatorios, historia dinámica dependiendo de la reputación con las tres facciones, modificación de soldados) hasta una presencia en línea, como subir el juego en *itch.io*, crear una pequeña campaña publicitaria en redes sociales, etc., y seguir desarrollando el juego hasta considerarlo un producto suficientemente competitivo como para publicarse en plataformas de pago, como *Steam* y *Google Play Store*.

En resumen, este TF es el primer paso de desarrollo de un proyecto que durará años. Este solo es el principio de un proyecto de vida, que ha estado años en la mente del equipo de desarrollo, con una larga lista de características que se quieren implementar y una inmensa pasión por los videojuegos.

7. Bibliografía

[1]

Adobe. (2024). *Planes y precios de suscripción a Creative Cloud*. Obtenido de Adobe:
<https://www.adobe.com/es/creativecloud/plans.html>

[2]

Bermejo, I. (17 de Enero de 2024). *Nuevo SMI en 2024: esto es lo que cobrará si trabaja media jornada*. Obtenido de La Razón:
https://www.larazon.es/economia/nuevo-smi-2024-esto-que-cobrara-trabaja-media-jornada_2024011765a7e6b367d53e0001f4b2a1.html#:~:text=25%20horas%3A%20826%2C87%20euros,horas%3A%20727%2C40%20euros.

[3]

Ashley, J. (2 de Enero de 2024). *WHAT IS THE FORTNITE PLAYER COUNT IN 2024?* Obtenido de esports.net: <https://www.esports.net/news/fortnite/fortnite-player-count/>

[4]

Tylor-Hill, G. (28 de Diciembre de 2023). *WARZONE PLAYER COUNT IN 2024: HOW MANY PEOPLE PLAY WARZONE?* Obtenido de esports.net:
<https://www.esports.net/news/cod/warzone-player-count/>

[5]

Steam. (20 de Abril de 2024). *Top 100 most played games*. Obtenido de Steam:
<https://store.steampowered.com/charts/mostplayed>

[6]

Komad, M. (5 de Agosto de 2023). *Product design and psychology: The Exploitation of Fear of Missing Out (FOMO) in Video Game Design*. Obtenido de Medium:
<https://medium.com/@milijanakomad/product-design-and-psychology-the-exploitation-of-fear-of-missing-out-fomo-in-video-game-design-5b15a8df6cda>

[7]

Knezovic, A. (8 de Abril de 2024). *200+ Mobile Games Statistics: Market & Revenue Report [2024]*. Obtenido de Udonis:
<https://www.blog.udonis.co/mobile-marketing/mobile-games/mobile-gaming-statistics>

[8]

Clement, J. (15 de Marzo de 2024). *Highest grossing mobile games worldwide in 2023*.

Obtenido de Statista:

<https://www.statista.com/statistics/1179913/highest-grossing-mobile-games/>

[9]

Broddy, Z. (15 de Octubre de 2023). *Minecraft crosses 300 million copies sold as it prepares to celebrate its 15th anniversary*. Obtenido de Windows Central:

<https://www.windowscentral.com/gaming/minecraft/minecraft-crosses-300-million-copies-sold-as-it-prepares-to-celebrate-its-15th-anniversary>

[10]

Krambs, G. (10 de Junio de 2024). *The Most Popular Indie Game of All Time, Ranked*.

Obtenido de StrawPoll: <https://strawpoll.com/most-popular-indie-game-all-time>

[11]

Kroupp, G. (4 de Setiembre de 2023). *First Person Shooters: \$150B Market Overview*.

Obtenido de LinkedIn:

<https://www.linkedin.com/pulse/first-person-shooters-150b-market-overview-guy-kroupp/>

[12]

Godot. (2024). *Main page*. Obtenido de Godot: <https://godotengine.org>

[13]

Godot. (2024). *Requerimientos del sistema*. Obtenido de Godot Docs:

https://docs.godotengine.org/es/4.x/about/system_requirements.html