

Especificación de requerimientos de software

Guasacaca Studios

Samantha Covarrubias Figueroa

Cristina Alessandra González

Valeria Martínez Martínez

Mau Tumalán

Sylvia Fernanda Colomo Fuente

Historias de usuario

| Cliente | Historia | Requerimientos |
|----------------|--|--|
| General | <ul style="list-style-type: none">Como cliente busco que se desarrolle un videojuego RPG | <ul style="list-style-type: none">Elaborar diagrama UML.Elaborar casos de uso.Elaborar diagramas de flujo de actividad. |
| | <ul style="list-style-type: none">Como cliente busco que se expliquen los controles y jugabilidad del juego | <ul style="list-style-type: none">Elaborar manual de instrucciones en el juegoImplementar manual de instrucciones en el juego |
| Socio Formador | <ul style="list-style-type: none">Como cliente busco que el jugador encuentre una experiencia entretenida e innovadora. | <ul style="list-style-type: none">Crear sprites del videojuegoBuscar e implementar música para el videojuegoBuscar e implementar efectos de sonido para el videojuego |
| | <ul style="list-style-type: none">Como cliente busco que exista una historia interesante. | <ul style="list-style-type: none">Creación de NPCsImplementación de NPCs.Creación de cutscenes.Implementación de cutscenes. |
| | <ul style="list-style-type: none">Como cliente busco que el videojuego implemente distintas mecánicas RPG | <ul style="list-style-type: none">Implementación de Skilltree para personaje principal (chef)Elementos de personalización visual de personaje principal (chef) |
| | <ul style="list-style-type: none">Como cliente busco que el usuario encuentre un deseo de mejorar su desempeño, manteniendo un equilibrio entre la recompensa de ganar y la frustración de perder. | <ul style="list-style-type: none">Implementación de dificultadesImplementación de clases en los aliados para permitir diferentes acercamientos a las estrategias realizadas dentro del nivel.Implementación de puntaje de recompensas para SkillTree |
| | <ul style="list-style-type: none">Como cliente quiero que existan elementos dentro del juego que reten al | <ul style="list-style-type: none">Implementación de clases en los enemigos para permitir diferentes acercamientos a las |

| | | |
|------------|---|--|
| | <p>usuario a buscar la mejor estrategia y sea retado para pasar el nivel.</p> | <p>estrategias realizadas dentro del nivel.</p> <ul style="list-style-type: none"> Implementación de dificultades en los diferentes enemigos |
| Videojuego | <ul style="list-style-type: none"> Como cliente quiero visualizar la evolución del protagonista, personajes aliados y enemigos durante el juego. | <ul style="list-style-type: none"> Diseño de evoluciones del personaje acorde a su nivel en el SkillTree Diseño de evoluciones de aliados acorde a las mejoras desbloqueadas en el SkillTree Diseño de evoluciones de enemigos acorde a la dificultad. Diseño de evoluciones de enemigos acorde a progreso en el juego. |
| | <ul style="list-style-type: none"> Como cliente quiero que el videojuego implemente mecánicas de estrategia en tiempo real | <ul style="list-style-type: none"> Implementación de selección de personajes aliados y movimiento de estos Implementación de aparición de enemigos y movimiento de estos Implementación de curación a los aliados Implementación de time-delay (que el tiempo vaya más lento mientras se selecciona la posición del personaje) Implementación de ataque entre personajes aliados y enemigos Implementación de ataque de enemigos a la base (carrito de comida) |
| | <ul style="list-style-type: none"> Como cliente quiero que el usuario tenga que defender e interactuar con una base | <ul style="list-style-type: none"> Implementación de carrito de comida Implementación de menú de interacción para creación de comida Implementación de cooldown de chef durante la creación de comida |
| | <ul style="list-style-type: none"> Como cliente quiero que el videojuego tenga la menor cantidad de periféricos (controles) y que tengan una razón de ser. | <ul style="list-style-type: none"> Implementación de uso de periféricos sencillos para la jugabilidad del juego. |
| | <ul style="list-style-type: none"> Como cliente quiero que el | <ul style="list-style-type: none"> Implementación de distintas |

| | | |
|---------------|---|---|
| | juego sea rejugable. | nacionalidades. <ul style="list-style-type: none"> Implementación de bifurcaciones en el SkillTree. |
| Base de datos | <ul style="list-style-type: none"> Como cliente quiero que la información del jugador se guarde en un repositorio de información. | <ul style="list-style-type: none"> Implementar la base de datos con la herramienta de MYSQL. |
| | <ul style="list-style-type: none"> Como cliente quiero que la información se pueda insertar, actualizar y eliminar en la base de datos. | <ul style="list-style-type: none"> Implementación del modelo CRUD en la base de datos. |
| | <ul style="list-style-type: none"> Como cliente quiero que se pueda enviar y recibir información en la base de datos, acerca del videojuego en la página web. | <ul style="list-style-type: none"> Permitir la comunicación entre la BD, la página y el juego mediante las API's correspondientes |
| | <ul style="list-style-type: none"> Como cliente quiero que se pueda acceder a la base de datos fácilmente. | <ul style="list-style-type: none"> Conectar la base de datos a la página web y el juego |
| | <ul style="list-style-type: none"> Como cliente quiero que se pueda almacenar información de checkpoints, inventario, ítems, escenarios, vida ataque, velocidad, mana en tablas separadas. | <ul style="list-style-type: none"> Crear el diagrama entidad relación para diseñar el modelo. Identificar las distintas tablas a implementar. |
| Web | <ul style="list-style-type: none"> Cómo cliente quiero que la página web cuente con un ambiente que sea atractivo visualmente y que sea interactivo. | <ul style="list-style-type: none"> Diseñar distintas opciones de posibles prototipos de páginas web Implementar el diseño front end que el cliente elija Asegurarse de tener una página dinámica (interactuable) |
| | <ul style="list-style-type: none"> Cómo cliente quiero que exista un apartado para iniciar sesión en el juego por medio de la página web. | <ul style="list-style-type: none"> Añadir una nueva sección dónde se pueda cargar el unity (uso de API) Mediante el API comunicar la base de datos con el juego de Unity para poder hacer partidas distintas |
| | <ul style="list-style-type: none"> Como cliente quiero que el juego se pueda jugar desde la página web | <ul style="list-style-type: none"> Crear el apartado para el juego de Unity y llamarlo desde el código en JS. |

| | | |
|--|---|--|
| | <ul style="list-style-type: none"> • Como cliente quiero que en la página web se muestren estadísticas y gráficas con los datos guardados en el juego. | <ul style="list-style-type: none"> • Implementación de estadísticas • Crear la comunicación de la BD y el código de la página para poder tener resultados visuales de las estadísticas (API) |
| | <ul style="list-style-type: none"> • Como cliente quiero que exista un apartado con un manual de usuario en la página. | <ul style="list-style-type: none"> • Crear el apartado • Implementarlo como parte del front end de la página |

“Product backlog” (Numeración de requerimientos):

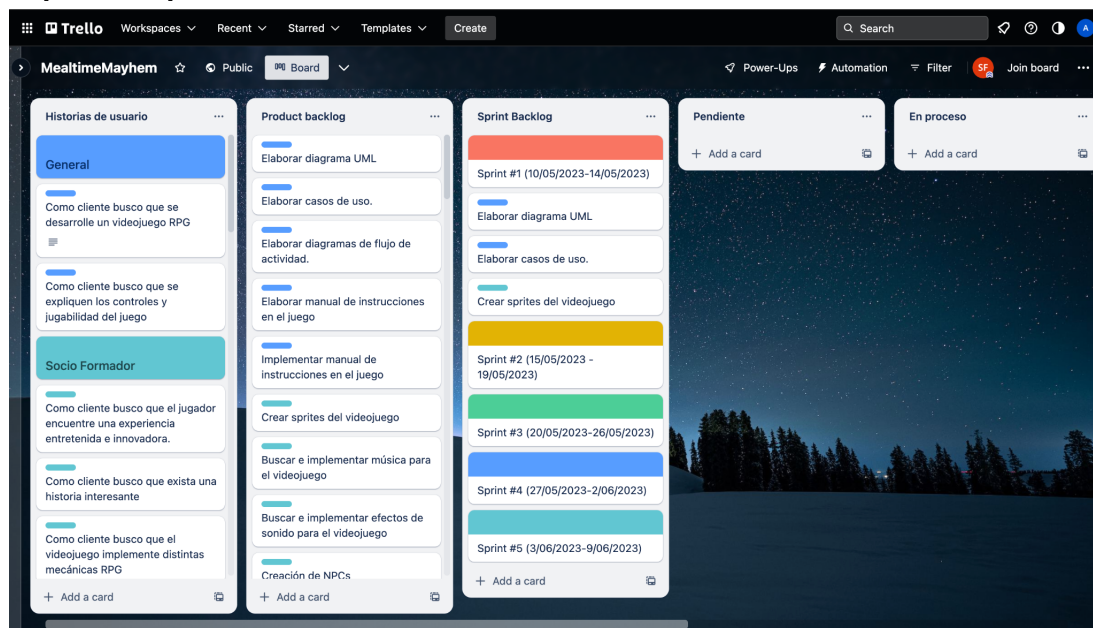
1. Elaborar diagrama UML.
2. Elaborar casos de uso.
3. Diseñar e implementar las capturas de las pantallas requeridas.
4. Crear el documento GDD con las especificaciones del juego.
5. Crear póster del videojuego.
6. Elaborar e implementar el manual de instrucciones en el juego
7. Crear sprites y animaciones del videojuego
8. Buscar e implementar música y efectos de sonido para el videojuego
9. Creación e implementación de NPCs
10. Creación e implementación de cutscenes.
11. Implementación de distintas nacionalidades.
12. Implementación de Skilltree para personaje principal (chef)
13. Elementos de personalización visual de personaje principal (chef)
14. Implementación de dificultades en los diferentes enemigos.
15. Creación e implementación de tilemap del overworld.
16. Overworld (Interactuable con el jugador).
17. Implementación de clases en los aliados para permitir diferentes acercamientos a las estrategias realizadas dentro del nivel.
18. Implementación de bifurcaciones y puntaje de recompensas para SkillTree
19. Implementación de clases en los enemigos para permitir diferentes acercamientos a las estrategias realizadas dentro del nivel.
20. Diseño de evoluciones del personaje acorde a su nivel en el SkillTree
21. Diseño de evoluciones de aliados acorde a las mejoras desbloqueadas en el SkillTree
22. Diseño de evoluciones de enemigos acorde a la dificultad.
23. Implementación de selección de personajes aliados y movimiento de estos
24. Implementación de aparición de enemigos y movimiento de estos
25. Implementación de time-delay (que el tiempo vaya más lento mientras se selecciona la posición del personaje)
26. Implementación de ataque entre personajes aliados y enemigos
27. Implementación de ataque de enemigos a la base (carrito de comida)
28. Implementación de carrito de comida
29. Implementación de menú de interacción para creación de comida
30. Implementación de Healthbar.
31. Implementación de uso de periféricos sencillos para la jugabilidad del juego.
32. Implementar la base de datos con la herramienta de MYSQL.
33. Implementación del modelo CRUD en la base de datos.

34. Permitir la comunicación entre la BD, la página y el juego mediante las API's correspondientes
35. Crear el diagrama entidad relación para diseñar el modelo.
36. Identificar las distintas tablas a implementar.
37. Diseño e implementación del front end (Crear secciones para Intro, Juego, Manual, Estadísticas y Créditos)
38. Comunicar la Web con la DB
39. Implementar el SignUp y Login
40. Mediante el API comunicar la base de datos con el juego de Unity para poder hacer partidas distintas
41. Embeber el juego en la Web
42. Crear la comunicación de la BD y el código de la página para poder tener resultados visuales de las estadísticas (API)
43. Implementación de curación a los aliados

“Sprint backlog” con las tareas a ser realizadas en el primer sprint:

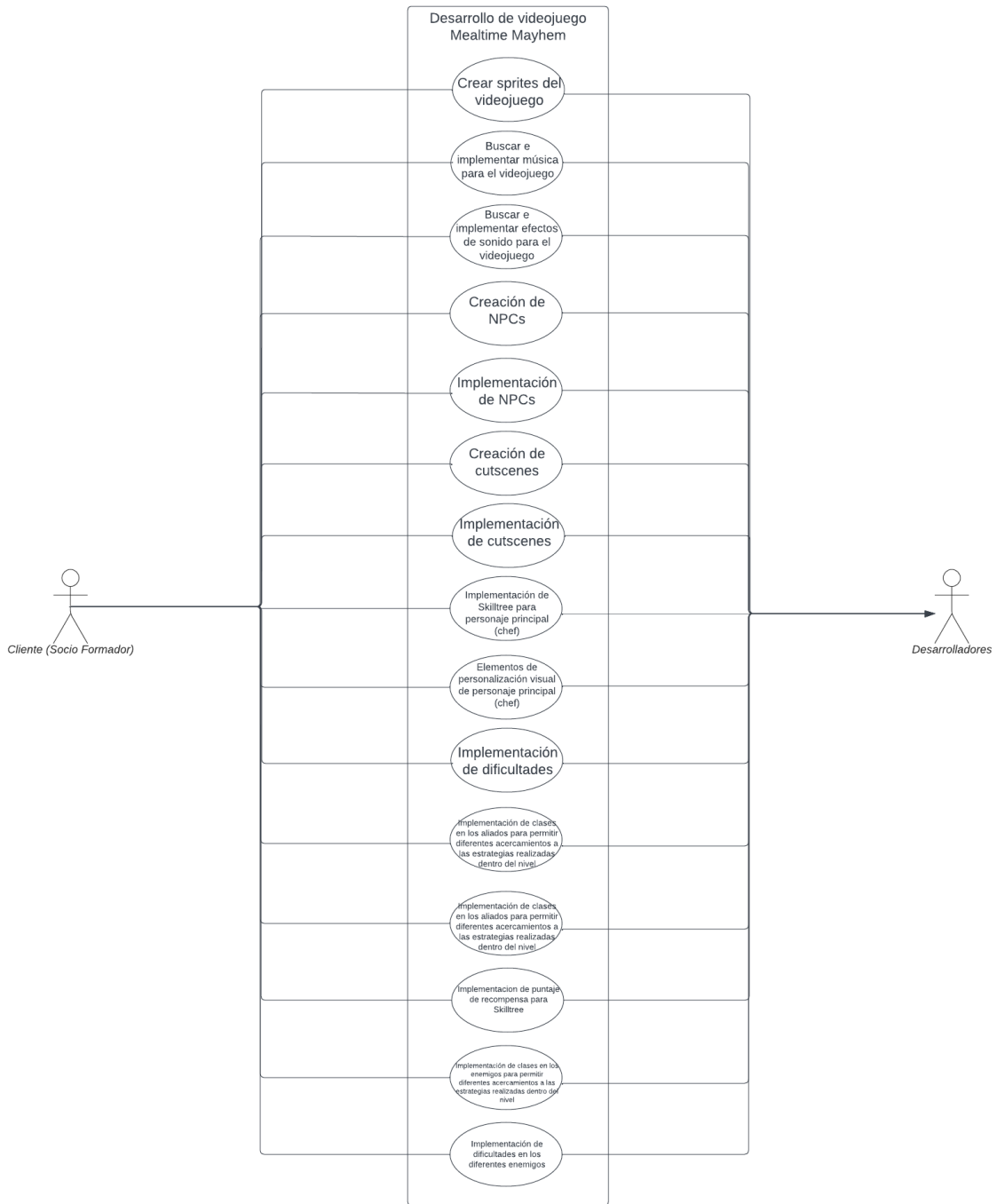
<https://trello.com/invite/b/Dtdwlh8r/ATTI4391f493d2473d255296fd70497099a27B890CAC/mealtimemayhem>

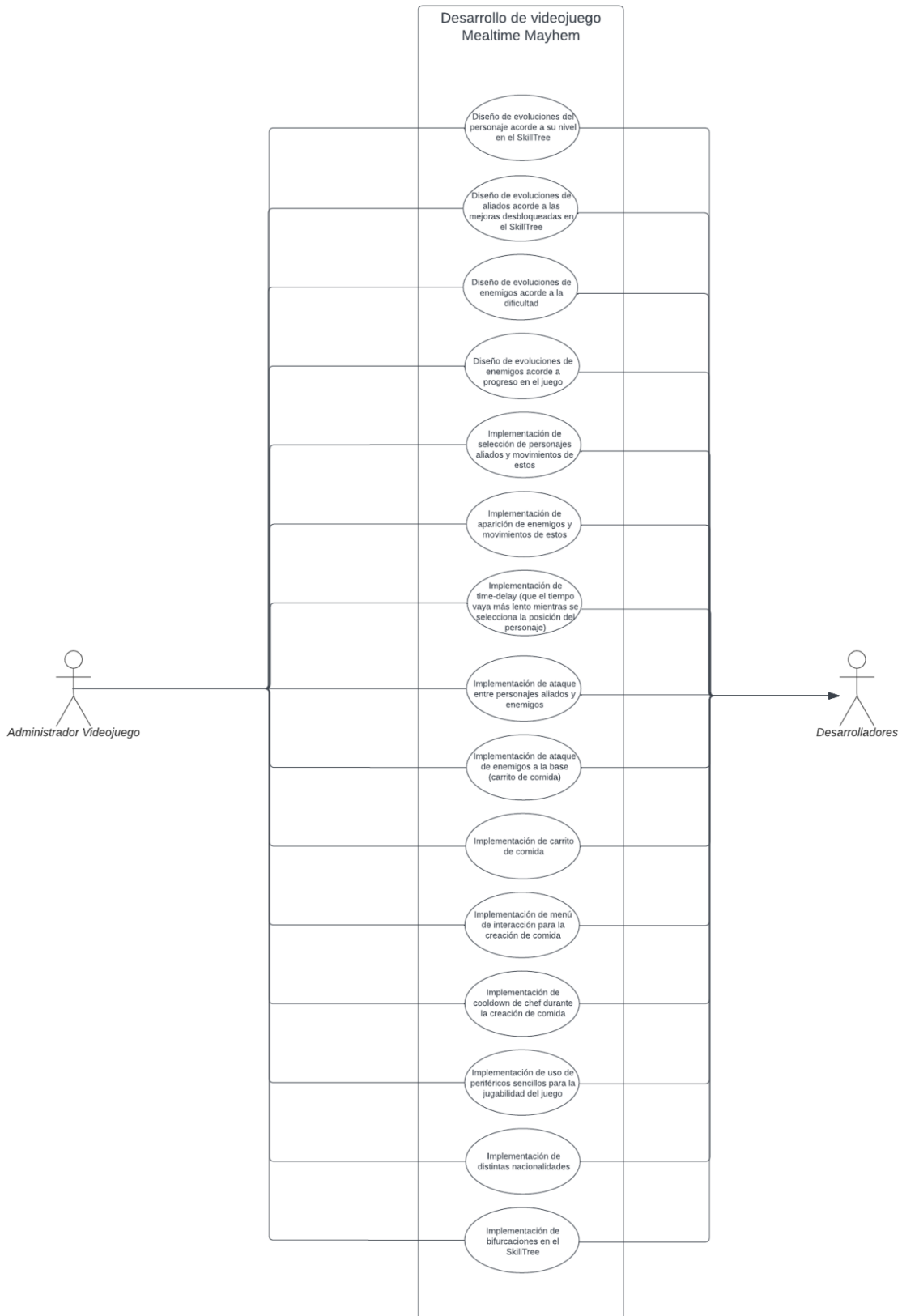
Captura de pantalla del tablero virtual

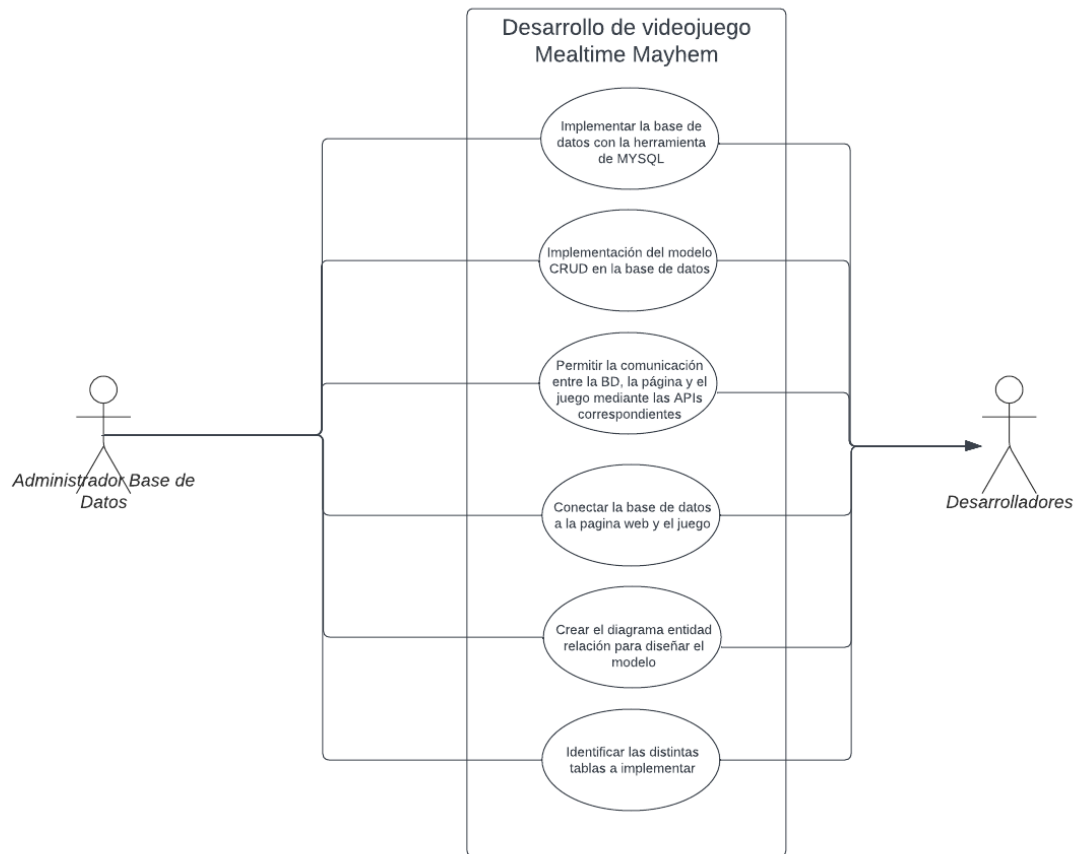


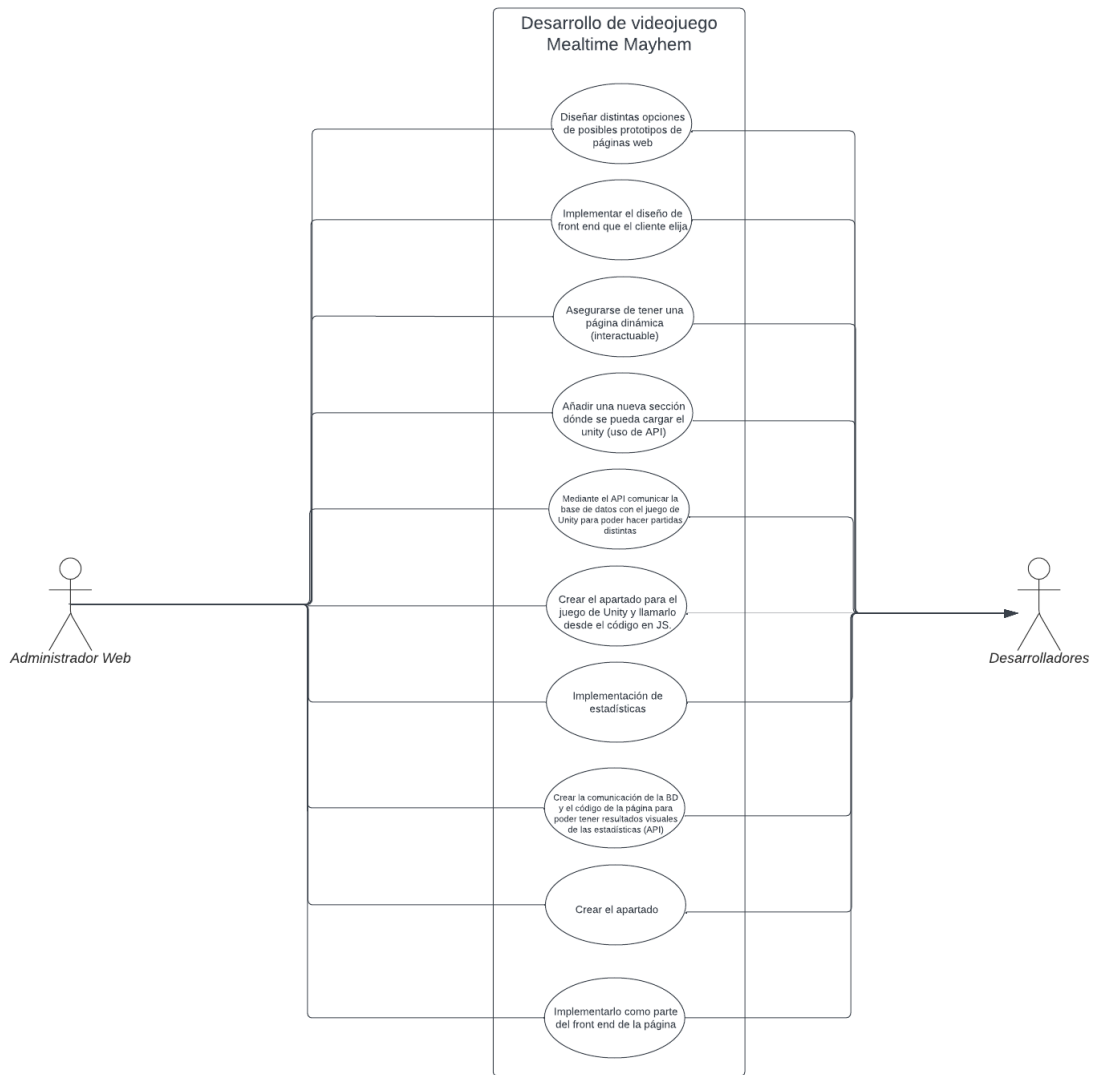
Diagramas de casos de uso:

📄 Casos de uso.png









Descripción de cada uno de los casos de uso del diagrama UML:

| Caso de uso | Elaborar diagrama UML |
|--------------------------------------|--|
| Requerimientos relacionados | N/A |
| Meta en contexto | Elaborar un diagrama UML que tenga requerimientos pertinentes para el correcto desarrollo del videojuego. |
| Precondiciones | Definir las mecánicas y desarrollo del videojuego para encontrar todos los posibles requisitos para su desarrollo. |
| Condición para un fin exitoso | Diagrama completo, con todos los requerimientos necesarios para el desarrollo exitoso del videojuego. |
| Condición para un fin fallido | Diagrama modificado, con cambios por hacer para incluir todos los requerimientos no contemplados. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Administrador del videojuego |
| Trigger | Los desarrolladores deben realizar y desambiguar todos los requerimientos acorde a las necesidades del administrador del videojuego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente las mecánicas y demás características del videojuego. |
| 2 | Definir completamente los requerimientos que serán necesarios para su realización. |
| 3 | Desarrollar el diagrama UML que incluya todos los requerimientos previamente definidos. |
| Extensiones | N/A |

| Caso de uso | Elaborar casos de uso |
|-------------------------------|---|
| Requerimientos relacionados | 1 |
| Meta en contexto | Diagramas con todos los requerimientos claramente definidos. |
| Precondiciones | Haber definido claramente los requerimientos necesarios en cada área y sus respectivos actores. |
| Condición para un fin exitoso | Diagrama con requerimientos definidos acorde a su área, con el fin de evitar modificaciones a lo largo del proyecto. |
| Condición para un fin fallido | Diagramas que requieran de modificación, debido a falta o mala representación de información como áreas o actores correspondientes. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Administradores |
| Trigger | Los desarrolladores deben definir los requerimientos acorde a las necesidades de los administradores. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Escuchar activamente las necesidades de cada administrador. |
| 2 | Definir completamente los requerimientos que serán necesarios en cada área. |
| 3 | Desarrollar diagramas de uso para cada área y sus respectivos actores. |
| Extensiones | N/A |

| Caso de uso | Diseñar e implementar las capturas de las pantallas requeridas. |
|-------------------------------|--|
| Requerimientos relacionados | 1, 4 |
| Meta en contexto | Crear las pantallas requeridas para juegos |
| Precondiciones | Definir las pantallas a implementar en el juego |
| Condición para un fin exitoso | Pantallas finalizadas con todos los elementos necesarios en ellas |
| Condición para un fin fallido | Pantallas incompletas o con necesidad de ser modificadas |
| Actores primarios | Desarrolladores |
| Actores secundarios | Administradores |
| Trigger | Los desarrolladores deben de diseñar e implementar las capturas de pantalla requeridas para el juego |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Diseñar las pantallas del juego |
| 2 | Crear las pantallas del juego |
| 3 | Implementar las pantallas del juego |
| Extensiones | N/A |

| Caso de uso | Crear el documento GDD con las especificaciones del juego. |
|-------------------------------|---|
| Requerimientos relacionados | 1, 2 |
| Meta en contexto | Crear el documento GDD incluyendo todos los puntos importantes del juego |
| Precondiciones | Definir los distintos elementos del juego |
| Condición para un fin exitoso | El documento GDD contiene toda la información y especificaciones del juego |
| Condición para un fin fallido | El documento GDD está incompleto o contiene errores en alguno de los elementos que contiene |
| Actores primarios | Desarrolladores |
| Actores secundarios | Administradores |
| Trigger | Los desarrolladores deben especificar y plasmar toda la información relevante del juego en el documento GDD |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Diseñar el juego |
| 2 | Definir detalladamente cada elemento del juego |
| 3 | Documentar los elementos del juego en el documento GDD |
| Extensiones | N/A |

| Caso de uso | Crear póster del videojuego. |
|-------------------------------|--|
| Requerimientos relacionados | 1 |
| Meta en contexto | Realizar el póster con elementos visuales del juego |
| Precondiciones | Contar con todos los elementos visuales del juego |
| Condición para un fin exitoso | El póster cuenta con todos los elementos clave para mostrar el contenido del juego |
| Condición para un fin fallido | El póster no cuenta con todos los elementos del videojuego necesarios |
| Actores primarios | Desarrolladores |
| Actores secundarios | Administradores |
| Trigger | Los desarrolladores deben crear el póster con todos los elementos principales del juego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Crear el juego |
| 2 | Definir los elementos clave a agregar en el póster |
| 3 | Realizar el póster con los elementos del juego |
| Extensiones | N/A |

| Caso de uso | Elaborar e implementar el manual de instrucciones en el juego |
|-------------------------------|---|
| Requerimientos relacionados | 1 |
| Meta en contexto | Crear el manual con las instrucciones del juego |
| Precondiciones | Contar con las mecánicas principales del juego |
| Condición para un fin exitoso | El manual del juego contiene la información relevante del éste |
| Condición para un fin fallido | El manual del juego está incompleto o no contiene los elementos necesarios |
| Actores primarios | Desarrolladores |
| Actores secundarios | Administradores |
| Trigger | Los desarrolladores deben crear el manual del juego con todos los elementos principales de éste |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Diseñar las mecánicas del juego |
| 2 | Implementar las mecánicas en el juego |
| 3 | Realizar el manual del juego |
| Extensiones | N/A |

| Caso de uso | Elaborar diagramas de flujo de actividad |
|-------------------------------|--|
| Requerimientos relacionados | 1, 2 |
| Meta en contexto | Los diagramas de flujo de actividad muestran gráficamente el flujo de desarrollo del videojuego y las conexiones entre las administraciones. |
| Precondiciones | Contar con los requerimientos claramente definidos, diagramas de caso de uso UML. |
| Condición para un fin exitoso | Se muestra correctamente en los diagramas de flujo, la relación de los requerimientos con las distintas administraciones del videojuegos que cumplen correctamente con todas las necesidades |
| Condición para un fin fallido | Se necesitan realizar modificaciones a los diagramas ya que no se muestra correctamente la relación entre las administraciones |
| Actores primarios | Desarrolladores |
| Actores secundarios | Administradores |
| Trigger | Los desarrolladores deberán escuchar las necesidades de los administradores, definir los requerimientos adecuados y relacionarlos entre sí con un diagrama de flujo de actividades |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Escuchar activa a las necesidades de cada administración |
| 2 | Definición de los requerimientos con sus respectivos diagramas de casos de uso UML |
| 3 | Desarrollar el diagrama de flujo con todas las actividades |
| Extensiones | N/A |

| Caso de uso | Elaborar e implementar el manual de instrucciones en el juego |
|-------------------------------|---|
| Requerimientos relacionados | 1, 2, 3, 4 |
| Meta en contexto | Crear un manual que permita al jugador tener el conocimiento necesario para navegar por el juego. |
| Precondiciones | Establecer y definir claramente las mecánicas, su funcionamiento e interacciones, así como los personajes para elaborar un manual detallado. |
| Condición para un fin exitoso | Elaboración de un manual detallado, con el funcionamiento del juego que permita al usuario desarrollar un entendimiento lo suficientemente profundo del mismo para que su navegación sea agradable. |
| Condición para un fin fallido | No es comprensible el funcionamiento del juego debido a una redacción inapropiada de las instrucciones. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Estando en la página web, el usuario hace clic en la sección de instrucciones, causando que estas se desplieguen. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir las mecánicas, interacciones y funcionamiento. |
| 2 | Redactar de manera detallada cada instrucción. |
| Extensiones | N/A |

| Caso de uso | Crear sprites y animaciones del videojuego |
|-------------------------------|--|
| Requerimientos relacionados | 4 |
| Meta en contexto | Elaborar los sprites y animaciones que serán parte del videojuego, pantallas, personajes con movimiento, ítems, etc. |
| Precondiciones | Haber definido una apariencia para los personajes, mecánicas, obstáculos y ambiente de juego. |
| Condición para un fin exitoso | Los sprites y animaciones pueden ser usados adecuadamente en el resto del desarrollo del videojuego. |
| Condición para un fin fallido | Se requieren modificaciones a los sprites y animaciones para su correcta implementación en Unity y en el videojuego en general. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Administrador del videojuego |
| Trigger | El administrador de videojuego define qué tipo de juego busca. Los desarrolladores crean a partir de esto un boceto para los sprites y animaciones del videojuego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente el videojuego, sus mecánicas, ítems y apariencia general. |
| 2 | Corroborar la congruencia entre las necesidades del administrador y lo desarrollado por el equipo. |
| 3 | Crear los sprites y animaciones para el videojuego. |
| Extensiones | N/A |

| Caso de uso | Buscar e implementar música y efectos de para el videojuego |
|--------------------------------------|--|
| Requerimientos relacionados | 4 |
| Meta en contexto | Buscar e implementar música y efectos libre de derechos en congruencia con la historia y el ambiente del juego. |
| Precondiciones | Definir las mecánicas y entornos del videojuego. |
| Condición para un fin exitoso | Existe una congruencia entre las mecánicas y entornos del juego con las piezas musicales que se ocupan. |
| Condición para un fin fallido | El formato de las piezas no es el apropiado o no existe una relación entre estas y el ambiente del juego. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Diferentes cambios de escena son reconocidos por el usuario gracias a la música y efectos de sonido. Los desarrolladores hacen esto posible por medio de la búsqueda e implementación de esta. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente las mecánicas y entornos del videojuego. |
| 2 | Clasificar las mecánicas y entornos de acuerdo a su rol en el juego. |
| 3 | Buscar e implementar la música y efectos de sonido que tienen congruencia con las mecánicas y entornos previamente definidos. |
| Extensiones | N/A |

| Caso de uso | Creación e Implementación de NPCs |
|-------------------------------|---|
| Requerimientos relacionados | 9 |
| Meta en contexto | Encontrar un rol adecuado para los personajes NPCs que ayuden a la experiencia de juego del usuario. |
| Precondiciones | Definir totalmente las mecánicas y desarrollo del videojuego. |
| Condición para un fin exitoso | Los NPCs tienen un papel relevante y real dentro de la historia además de que interactúan con el jugador principal. |
| Condición para un fin fallido | Los NPCs no tienen un rol relevante, no muestran información adecuada o no son implementados. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben realizar NPCs que al interactuar con el usuario den información relevante y sean parte de la historia del videojuego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente el videojuego y las interacciones NPCs - usuario. |
| 2 | Producir NPCs relevantes a la historia y desarrollo del videojuego. |
| 3 | Ensamblar el videojuego considerando el rol de los NPCs en la historia y desarrollo. |
| Extensiones | N/A |

| Caso de uso | Creación e implementación de cutscenes |
|--------------------------------------|--|
| Requerimientos relacionados | 4 |
| Meta en contexto | Implementar escenas que sirvan para añadir coherencia al desarrollo de la historia. |
| Precondiciones | Definir totalmente la historia del videojuego, determinando a los personajes adecuados. |
| Condición para un fin exitoso | Las cutscenes tienen un papel importante dentro de la historia y ayudan al jugador a sumergirse más dentro del juego. |
| Condición para un fin fallido | Las cutscenes no tienen sentido dentro de la historia. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Al terminar un nivel el usuario es presentado con una cutscene que hila la historia con el siguiente nivel. Los desarrolladores diseñan una cutscene congruente con la historia. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente la historia del juego. |
| 2 | Diseñar las cutscenes relevantes. |
| 3 | Crear las cutscenes por medio de un programa de diseño o animación. |
| Extensiones | N/A |

| Caso de uso | Implementación de distintas nacionalidades. |
|--------------------------------------|---|
| Requerimientos relacionados | 4 |
| Meta en contexto | Implementar escena para elegir la nacionalidad en la que el jugador quiera jugar |
| Precondiciones | Definir el total de nacionalidades a implementar en el juego |
| Condición para un fin exitoso | Las nacionalidades contienen distintos beneficios diferentes al crear una partida |
| Condición para un fin fallido | Las nacionalidades no están implementadas o existe algún error en su creación |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Al comenzar una partida el usuario tiene la opción de elegir entre varias nacionalidades, la que más le guste según las habilidades que adquiere al principio del juego |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente las nacionalidades en el juego. |
| 2 | Diseñar las habilidades principales para cada nacionalidad |
| 3 | Crear la escena con las opciones de las nacionalidades |
| Extensiones | N/A |

| Caso de uso | Implementación de Skilltree para personaje principal (chef) |
|-------------------------------|---|
| Requerimientos relacionados | 4 |
| Meta en contexto | Implementar sistema de mejoras para el personaje principal. |
| Precondiciones | Definir las habilidades que tendrá el personaje principal. |
| Condición para un fin exitoso | El personaje principal tiene un cambio en cuanto a habilidades dentro del juego y en relación a la mecánicas de este gracias al uso del skilltree. |
| Condición para un fin fallido | El personaje principal no presenta algún cambio o cambio positivo dentro del juego después de hacer uso del skilltree. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores diseñan un skilltree con opciones pertinentes al desarrollo del juego, del cual el usuario puede hacer uso al comienzo de este. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente las mejoras pertinentes al personaje principal dentro del juego. |
| 2 | Implementar las mejoras del skilltree en Unity. |
| 3 | Comprobar el funcionamiento adecuado de las mejoras planteadas. |
| Extensiones | N/A |

| Caso de uso | Elementos de personalización visual de personaje principal (chef) |
|-------------------------------|--|
| Requerimientos relacionados | N/A |
| Meta en contexto | Lograr un nivel de personalización del personaje principal característico de un RPG. |
| Precondiciones | Definir el personaje principal y sus posibles apariencias. |
| Condición para un fin exitoso | El personaje principal es personalizable con consistencia a lo largo del desarrollo del juego. |
| Condición para un fin fallido | El personaje principal no es personalizable o no mantiene un nivel de personalización constante a lo largo del juego. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben realizar un sistema de personalización del personaje principal a medida que el usuario vaya avanzando en el juego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir al personaje principal y los aspectos que serán personalizables de este. |
| 2 | Crear visualmente las personalizaciones pertinentes. |
| 3 | Implementar en Unity los componentes visuales junto con los aspectos técnicos que permitan hacer uso de estos componentes. |
| Extensiones | N/A |

| Caso de uso | Implementación de dificultades en los diferentes enemigos |
|--------------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Lograr un cambio de dificultad por cada horda y nivel que el jugador supera. |
| Precondiciones | Definir totalmente las mecánicas y desarrollo del videojuego. |
| Condición para un fin exitoso | El usuario se encuentra con un nivel de dificultad creciente conforme supera ciertos desafíos durante su experiencia al jugar. |
| Condición para un fin fallido | El usuario se encuentra con un juego de dificultad lineal, sea muy fácil o muy difícil, lo que afecta su experiencia de juego. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Al superar cada horda o nivel el usuario enfrenta una dificultad mayor para superar los desafíos que se le presentan, la cual es designada por los desarrolladores. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Diseñar la estructura del juego, principalmente los niveles y hordas por niveles. |
| 2 | Determinar cómo será representada la dificultad a incrementar por horda y nivel. |
| 3 | Implementar los cambios previamente determinados en dificultad al juego en Unity. |
| Extensiones | N/A |

| Caso de uso | Creación e implementación de tilemap del overworld. |
|--------------------------------------|---|
| Requerimientos relacionados | 4 |
| Meta en contexto | Lograr crear el mundo del overworld con elementos para explorar en él |
| Precondiciones | Definir visualmente el diseño del overworld |
| Condición para un fin exitoso | El overworld contiene elementos visuales de acuerdo con el tema del juego |
| Condición para un fin fallido | El overworld está incompleto o no va con la temática del juego |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Al crear el mapa del overworld se crea un ambiente agradable para jugar |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Diseñar el mapa del juego |
| 2 | Determinar los elementos importantes a implementar |
| 3 | Implementar el mapa del overworld |
| Extensiones | N/A |

| Caso de uso | Overworld (Interactuable con el jugador). |
|--------------------------------------|--|
| Requerimientos relacionados | 4, 15 |
| Meta en contexto | Implementar elementos interactivables en el overworld |
| Precondiciones | Implementar el mapa del overworld |
| Condición para un fin exitoso | El overworld contiene elementos interactivables en el juego |
| Condición para un fin fallido | Los elementos para generar interacción en el overworld están incompletos |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Al crear elementos interactivables en el juego, se crea un ambiente más ameno y real para brindarle al usuario una experiencia muy grata |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Diseñar los elementos interactivables de overworld en juego |
| 2 | Determinar los espacios dónde se van a implementar los elementos interactivables |
| 3 | Implementar los elementos interactivables del overworld |
| Extensiones | N/A |

| Caso de uso | Implementación de clases en los aliados para permitir diferentes acercamientos a las estrategias realizadas dentro del nivel |
|-------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Crear las clases correspondientes a los personajes que acompañan al personaje principal con sus distintas habilidades |
| Precondiciones | Definir totalmente las mecánicas y desarrollo del videojuego. |
| Condición para un fin exitoso | Los aliados (acompañantes del personaje principal) tienen diferentes habilidades que ayudan al personaje principal a combatir a los enemigos dentro del juego |
| Condición para un fin fallido | Los aliados no cuentan con habilidades variadas o su funcionamiento no es el correcto. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben realizar aliados que al interactuar con el juego, se genere en el usuario un sentido de estrategia dependiendo de las habilidades de los aliados elegidos |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente el las habilidades de cada tipo de aliado |
| 2 | Establecer la forma en la que se mostrarán las habilidades de los aliados |
| 3 | Implementar las habilidades de cada aliado con los aspectos técnicos correspondientes en Unity |
| Extensiones | N/A |

| Caso de uso | Implementación de clases en los enemigos para permitir diferentes acercamientos a las estrategias realizadas dentro del nivel |
|-------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Crear las clases correspondientes a los personajes que acompañan al personaje principal con sus distintas habilidades |
| Precondiciones | Definir totalmente las mecánicas y desarrollo del videojuego. |
| Condición para un fin exitoso | Los enemigos |
| Condición para un fin fallido | Los aliados no cuentan con habilidades variadas o su funcionamiento no es el correcto. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben realizar aliados que al interactuar con el juego, se genere en el usuario un sentido de estrategia dependiendo de las habilidades de los aliados elegidos |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente el las habilidades de cada tipo de aliado |
| 2 | Establecer la forma en la que se mostrarán las habilidades de los aliados |
| 3 | Implementar las habilidades de cada aliado con los aspectos técnicos correspondientes en Unity |
| Extensiones | N/A |

| Caso de uso | Implementación de clases en los enemigos para permitir diferentes acercamientos a las estrategias realizadas dentro del nivel |
|-------------------------------|--|
| Requerimientos relacionados | N/A |
| Meta en contexto | Lograr un sistema de ataque por parte de los enemigos que permita derrotar a estos estratégicamente |
| Precondiciones | Definir totalmente las mecánicas y desarrollo del videojuego. |
| Condición para un fin exitoso | Los enemigos tienen un esquema de ataque estructurado que permite al jugador crear estrategias para derrotarlos |
| Condición para un fin fallido | Los enemigos no contienen ataques estructurados por lo que es muy difícil implementar una estrategia favorable por el usuario |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben realizar enemigos que sigan un orden de ataque el cual involucre al usuario a desarrollar estrategias que le permitan vencer a los enemigos de la mejor manera posible |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente el comportamiento de los enemigos en el juego |
| 2 | Establecer las formas de daño que pueden recibir y hacer los enemigos |
| 3 | Implementar las mecánicas necesarias para la creación de la estructura de ataque de los enemigos |
| Extensiones | N/A |

| Caso de uso | Implementación de dificultades en los diferentes enemigos |
|-------------------------------|--|
| Requerimientos relacionados | N/A |
| Meta en contexto | Lograr un cierto nivel de dificultad para cada tipo de enemigo en el juego |
| Precondiciones | Definir totalmente las mecánicas, items y enemigos en el juego |
| Condición para un fin exitoso | Cada tipo de enemigo cuenta con cierto nivel de daño dependiendo de su clasificación |
| Condición para un fin fallido | Todos los enemigos generan el mismo nivel de daño por lo que el nivel de dificultad es constante en todo momento en el juego |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben realizar varios tipos de enemigos que generen diferentes niveles de daño haciendo que la dificultad varíe dependiendo del enemigo al que el usuario ataque |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir los distintos tipos de enemigos |
| 2 | Establecer los rangos de daño que genera cada tipo de enemigo |
| 3 | Implementar diversos niveles con la variedad de enemigos en Unity |
| Extensiones | N/A |

| Caso de uso | Diseño de evoluciones del personaje acorde a su nivel en el SkillTree |
|-------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Implementar un sistema de evolución donde se muestren las mejoras actuales del personaje . |
| Precondiciones | Definir totalmente las mecánicas y desarrollo del videojuego. |
| Condición para un fin exitoso | Se muestra la evolución por las mejoras del personaje principal en el juego |
| Condición para un fin fallido | No se muestra claramente la evolución por las mejoras del personaje principal en el juego |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben realizar un sistema de evolución del personaje de acuerdo a las mejoras del personaje |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Establecer el tipo de mejoras que se pueden hacer por el SkillTree |
| 2 | Definir el tipo de evolución que se mostrará al realizar las mejoras por medio del SkillTree |
| 3 | Implementar la evolución del personaje principal |
| Extensiones | N/A |

| Caso de uso | Diseño de evoluciones de aliados acorde a las mejoras desbloqueadas en el skilltree |
|--------------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Diseñar un cambio en las habilidades de los aliados al jugador principal en relación a las mejoras realizadas en el skilltree. |
| Precondiciones | Definir las mejoras presentadas en el skilltree y su posible aplicación a los personajes aliados. |
| Condición para un fin exitoso | Los personajes aliados tienen una mejora en habilidad propiciada por una elección en el skilltree. |
| Condición para un fin fallido | Los personajes aliados no presentan mejoras en habilidad o las mejoras no mantienen congruencia con las seleccionadas en el skilltree. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben implementar la extensión de las habilidades en el skilltree a los aliados, las cuales serán aprovechadas por el usuario en su experiencia a lo largo del juego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir completamente el skilltree a ser usado en el juego. |
| 2 | Adaptar las mejoras ofrecidas en el skilltree para el personaje principal a sus aliados. |
| 3 | Desarrollar el uso de las mejoras previamente definidas en Unity a lo largo del videojuego para los personajes aliados. |
| Extensiones | N/A |

| Caso de uso | Diseño de evoluciones de enemigos acorde a la dificultad |
|-------------------------------|--|
| Requerimientos relacionados | N/A |
| Meta en contexto | Diseñar un cambio en las habilidades de los enemigos conforme una dificultad base designada. |
| Precondiciones | Diseñar a los enemigos y el momento dentro del desarrollo del videojuego en el que se presentarán. |
| Condición para un fin exitoso | Los enemigos presentan diferentes habilidades base a lo largo del juego. |
| Condición para un fin fallido | Los enemigos no presentan habilidades iniciales o no presentan un cambio en habilidades a lo largo del juego. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben implementar las evoluciones de los diferentes enemigos con los cuales el usuario se encontrará a lo largo del juego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir a los enemigos que se encontrarán a lo largo del juego. |
| 2 | Diseñar los cambios que presentarán los diversos tipos de enemigos. |
| 3 | Implementar las evoluciones de los enemigos a lo largo del desarrollo de la historia, hordas y niveles del juego. |
| Extensiones | N/A |

| Caso de uso | Diseño de evoluciones de enemigos acorde a progreso en el juego |
|-------------------------------|--|
| Requerimientos relacionados | N/A |
| Meta en contexto | Diseñar un cambio en las habilidades de los enemigos conforme al progreso del usuario en el juego. |
| Precondiciones | Definir los cambios/evoluciones que pueden presentar los enemigos ocupando su dificultad base como referencia. |
| Condición para un fin exitoso | Los enemigos presentan un cambio en dificultad conforme al progreso del usuario en el juego. |
| Condición para un fin fallido | Los enemigos no presentan habilidades o no presentan un cambio incremental en habilidades a lo largo del juego. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben implementar las evoluciones de los diferentes enemigos con los cuales el usuario se encontrará a lo largo del juego en relación a su progreso. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir a los enemigos que se presentarán a lo largo del juego. |
| 2 | Diseñar los cambios que presentarán los enemigos en relación al progreso del usuario. |
| 3 | Implementar las evoluciones de los enemigos en Unity conforme al punto en el que se encuentra el usuario dentro del juego. |
| Extensiones | N/A |

| Caso de uso | Implementación de selección de personajes aliados y movimientos de estos |
|--------------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Establecer un entorno de selección de personajes aliados y movimientos para el jugador principal. |
| Precondiciones | Definir los personajes aliados y movimientos que estos pueden tener. |
| Condición para un fin exitoso | Los personajes aliados y sus posibles movimientos pueden ser seleccionados dentro del juego. |
| Condición para un fin fallido | Los personajes aliados no pueden ser seleccionados o únicamente pueden ser seleccionados los personajes sin posibilidad de elección de movimientos. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Los desarrolladores deben implementar un entorno de selección que permita al usuario designar a sus aliados y sus movimientos al inicio del juego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir las elecciones para aliados y movimientos. |
| 2 | Diseñar el entorno dentro del juego que permita realizar una selección fácil y completa para el usuario. |
| 3 | Implementar en Unity el entorno y las posibles elecciones designadas previamente. |
| Extensiones | N/A |

| Caso de uso | Implementación de aparición de enemigos y movimiento de estos |
|--------------------------------------|--|
| Requerimientos relacionados | N/A |
| Meta en contexto | Determinar los momentos y lugares en los que los enemigos aparecerán dentro del juego además de los movimientos que tendrán dentro de este. |
| Precondiciones | Definir a los enemigos y los posibles movimientos que pueden tener dentro del juego. |
| Condición para un fin exitoso | Los enemigos aparecen en los momentos y lugares adecuados además de que presentan un movimiento acorde al diseño del juego. |
| Condición para un fin fallido | Los enemigos aparecen en momentos o lugares arbitrarios o presentan movimientos incongruentes con el diseño del juego. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | El usuario se encuentra con los diferentes enemigos activos a lo largo de su experiencia de juego, los cuales tendrán su aparición y movimientos determinados por los desarrolladores. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir a los enemigos y los posibles movimientos que pueden tener en el juego. |
| 2 | Establecer congruencia entre los momentos, lugares y movimientos de los enemigos y el resto del desarrollo del juego. |
| 3 | Implementar los aspectos previamente determinados de los enemigos en Unity, cuidando mantener coherencia entre todos los rubros considerados. |
| Extensiones | N/A |

| Caso de uso | Implementación de time-delay (que el tiempo vaya más lento mientras se selecciona la posición del personaje) |
|--------------------------------------|--|
| Requerimientos relacionados | N/A |
| Meta en contexto | Determinar la mecánica donde el tiempo se haga más lento al seleccionar la posición del personaje. |
| Precondiciones | Definir el movimiento de los personajes aliados y las distintas dificultades. |
| Condición para un fin exitoso | El tiempo se hace más lento de forma correcta acorde a la dificultad una vez se está seleccionando la posición del personaje |
| Condición para un fin fallido | El tiempo no se hace más lento o no cambia el efecto acorde a la dificultad. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Al seleccionar un personaje aliado |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir un reloj general en la escena |
| 2 | Definir la reducción del paso del tiempo dentro de la escena acorde a la dificultad seleccionada por el jugador. |
| Extensiones | N/A |

| Caso de uso | Implementación de ataque entre personajes aliados y enemigos |
|-------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Determinar la acción de ataque dada entre los personajes aliados y enemigos |
| Precondiciones | Definir variables de vida y ataque acorde a las clases de los personajes |
| Condición para un fin exitoso | Los enemigos y los aliados se atacan entre sí, y mueren en caso de que su variable de vida llegue a cero |
| Condición para un fin fallido | Los enemigos y los aliados no se atacan, o no mueren en caso de que su variable de vida llegue a cero |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Cuando un enemigo y un aliado se encuentran dentro del tablero del juego. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir variables de vida y ataque acorde a la clase de cada aliado y enemigo. |
| 2 | Definir interacciones de ataque al encontrarse dentro de un rango dependiendo del aliado o enemigo. |
| 3 | Implementar los aspectos previamente determinados de los enemigos en Unity, cuidando mantener coherencia entre todos los rubros considerados. |
| Extensiones | N/A |

| Caso de uso | Implementación de ataque de enemigos a la base (carrito de comida) |
|-----------------------------|--|
| Requerimientos relacionados | 24 |

| | |
|--------------------------------------|--|
| Meta en contexto | Lograr que los enemigos hagan daño al carrito de comida si se genera una colisión entre ambos. |
| Precondiciones | Definir variables de vida (carrito) y ataque acorde a las clases de los personajes |
| Condición para un fin exitoso | Los enemigos llegan al carrito y la vida de este baja cuando se detecta la colisión. |
| Condición para un fin fallido | La vida del carrito no es afectada, incluso cuando se detecta la colisión con un enemigo. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Cuando un enemigo se dirige al carrito de comida. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir variables de vida (carrito) y ataque enemigo. |
| 2 | Crear condición de colisión. |
| 3 | Ver que la vida del carrito es afectada en caso de una colisión. |
| Extensiones | N/A |

| Caso de uso | Implementación de carrito de comida |
|-------------------------------|--|
| Requerimientos relacionados | 24 |
| Meta en contexto | Tener un carrito de comida funcional dentro de cada nivel. Con funcional nos referimos a que tenga una vida y que se puedan “cocinar” la comida interactuando con el mismo durante cada nivel. |
| Precondiciones | Definir variables de vida (carrito) |
| Condición para un fin exitoso | Si el chef está cerca y hace click, aparecerá un soldado de comida más en el tablero. |
| Condición para un fin fallido | No se genera ningún soldado |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Cuando el chef se acerca al carro y el usuario da click al mismo. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir la cantidad de comida por crear por nivel. |
| 2 | Crear el carrito como prefab para poder usarlo entre cada nivel. |
| 3 | Crear el evento de click con la condición de que el chef se encuentre cerca. |
| Extensiones | N/A |

| Caso de uso | Implementación de menú de interacción para la creación de comida |
|-------------------------------|--|
| Requerimientos relacionados | 24,26 |
| Meta en contexto | Tener un carrito funcional en el cual se pueda cocinar comida y perder si se queda sin vida. |
| Precondiciones | Definir variables de vida (carrito). |
| Condición para un fin exitoso | Se podrá crear comida si la cantidad permitida no ha sido excedida y si el carrito tiene vida. |
| Condición para un fin fallido | Que la vida no afecte la partida o se cree más comida de la debida |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Cuando el chef está cerca se habilita ese menú. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Crear el menú |
| 2 | Añadir los eventos correspondientes |
| 3 | Crear comida |
| Extensiones | N/A |

| Caso de uso | Implementación de la healthbar |
|-------------------------------|--|
| Requerimientos relacionados | 24 |
| Meta en contexto | Tener una barra de salud por cada soldado/comida, chef y carrito de comida. |
| Precondiciones | Definir variables de vida para todos los personajes. |
| Condición para un fin exitoso | Que se vea una barra de salud sobre los soldados y que se pueda ver también la salud del chef y el carrito de comida dentro del nivel, así como su actualización continua por si llegara un enemigo a atacar a cualquiera. |
| Condición para un fin fallido | No se puede ver la barra de salud o no se afecta cuando hay una colisión con algún enemigo. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Cuando hay una colisión entre un enemigo y cualquiera de los tres personajes anteriores (carrito de comida, chef o soldado) |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Definir entidades principales del juego y posibles atributos. |
| 2 | Refinar las opciones del paso 1 para poder interactuar |
| 3 | Programar que si hay un ataque, se vea reflejado en la barra de la healthbar. |
| Extensiones | N/A |

| Caso de uso | Identificar las distintas tablas a implementar. |
|-------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Hacer una planeación de las entidades y los atributos que se van a guardar en la base de datos acorde con el juego. |
| Precondiciones | Tener definido la historia que el videojuego va a seguir y cuáles datos son relevantes para el jugador y para el juego mismo. |
| Condición para un fin exitoso | Planeación de un modelo entidad-relación normalizado que represente correctamente los datos del juego. |
| Condición para un fin fallido | Un modelo entidad-relación inadecuado o incompleto respecto al juego |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | N/A |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Elegir un diseño a seguir |
| 2 | Adaptar ese diseño en código de front end con sus archivos HTML y CSS correspondientes. |
| 3 | Una vez listo el esqueleto, empezar a añadir elementos visuales (imágenes, colores, degradados, videos, íconos, etc) a la página. |
| Extensiones | N/A |

| Caso de uso | Diseño e implementación del front end (Crear secciones para Intro, Juego, Manual, Estadísticas y Créditos) |
|-------------------------------|---|
| Requerimientos relacionados | N/A |
| Meta en contexto | Crear una página web y diseñar la parte del front end, junto con sus debidas secciones tales como, Introducción, Juego, Manual, Estadísticas y Créditos. La página será ordenada con un buen diseño estético. |
| Precondiciones | Tener definida la paleta de colores y elementos gráficos del juego para añadir al diseño de la página. |
| Condición para un fin exitoso | Tener una página visualmente atractiva con las secciones listas para ser editadas. |
| Condición para un fin fallido | Una página mal diseñada, confusa para el usuario. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | N/A |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Elegir un diseño a seguir |
| 2 | Adaptar ese diseño en código de front end con sus archivos HTML y CSS correspondientes. |
| 3 | Una vez listo el esqueleto, empezar a añadir elementos visuales (imágenes, colores, degradados, videos, íconos, etc) a la página. |
| Extensiones | N/A |

| Caso de uso | Comunicar la Web con la DB |
|-------------------------------|--|
| Requerimientos relacionados | 38 |
| Meta en contexto | Lograr una comunicación exitosa entre la BD y la Web para poder obtener datos desde la misma. |
| Precondiciones | Tener lista una página web y el código para hacer las consultas requeridas junto con un espacio para comprobar el funcionamiento dentro de la web o en la consola. |
| Condición para un fin exitoso | Lograr obtener datos y enviar datos mediante GET y POST/INSERT, respectivamente. |
| Condición para un fin fallido | No poder acceder a la información ni poder agregar nueva a las tablas seleccionadas. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Uso de botones para cada acción/consulta. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Crear un archivo para la api (app.js) |
| 2 | Crear los endpoints necesarios para cada consulta. |
| 3 | Tener los archivos de .js para poder trabajar con la información obtenida dentro del espacio dado en el HTML (definido con un id). |
| Extensiones | N/A |

| Caso de uso | Implementar el SignUp y Login |
|-------------------------------|---|
| Requerimientos relacionados | 38,39 |
| Meta en contexto | Tener lista la opción de iniciar sesión o crear una cuenta dentro de la página web. |
| Precondiciones | <ul style="list-style-type: none"> Tener lista una página web y el código para hacer las consultas requeridas junto con un espacio para comprobar el funcionamiento dentro de la web o en la consola. Lograr una comunicación exitosa entre la BD y la Web para poder obtener datos desde la misma. |
| Condición para un fin exitoso | Tener la interfaz para cada usuario lista y operando, capaz de mandar datos y comprobar si existen o sino hacer el POST correspondiente. |
| Condición para un fin fallido | Fallos en inicio de sesión y no poder crear usuarios nuevos. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | Uso de botones para cada acción/consulta. |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Crear las interfaces junto con botones para SignUp y LogIn dentro de la web junto con sus correspondientes secciones. |
| 2 | Crear los endpoints necesarios para cada consulta. |
| 3 | Habilitar los botones en la barra de navegación de la web para poder iniciar sesión o crear una cuenta nueva. |
| Extensiones | N/A |

| Caso de uso | Mediante el API comunicar la base de datos con el juego de Unity para poder hacer partidas distintas |
|-------------------------------|---|
| Requerimientos relacionados | NA |
| Meta en contexto | Lograr crear la funcionalidad de sesiones desde Unity que haga uso de la base de datos por medio de la API. |
| Precondiciones | Tener un apartado de sesiones (game selector) en Unity y haber creado una tabla de sesiones en el esquema relacional. |
| Condición para un fin exitoso | Creación de sesiones desde Unity que mantengan integridad y guarden los elementos necesarios. |
| Condición para un fin fallido | No se crean las sesiones o no guardan adecuadamente los elementos que deberían. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | N/A |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Crear scripts en Unity para que los botones o elementos se conecten a la API. |
| 2 | Modificar el archivo de javascript para recibir los endpoints necesarios. |
| 3 | Comprobar conexión echando a andar el servidor y haciendo pruebas elementales. |
| Extensiones | N/A |

| Caso de uso | Embeber el juego en la Web |
|-------------------------------|--|
| Requerimientos relacionados | 38 |
| Meta en contexto | En la sección predeterminada para el juego, poder poner el build del unity para poder jugarlo dentro de la página web. |
| Precondiciones | Tener una página web, responsiva, funcional junto con la sección para el juego. |
| Condición para un fin exitoso | Que se pueda jugar dentro de la web. |
| Condición para un fin fallido | No poder jugar. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | N/A |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Hacer el build para web desde el Unity |
| 2 | Conseguir el html y canvas del Unity para ponerlo dentro de la sección. |
| 3 | Comprobar paths y jugabilidad dentro de la web. Si es necesario reparar errores. |
| Extensiones | N/A |

| | |
|--------------------------------------|---|
| Caso de uso | Crear la comunicación de la BD y el código de la página para poder tener resultados visuales de las estadísticas (API) |
| Requerimientos relacionados | 38, 39 |
| Meta en contexto | Poder crear gráficas en la sección de estadísticas con datos obtenidos de la BD y por ende del juego. |
| Precondiciones | <ul style="list-style-type: none"> • Tener una página web funcional • Tener lista la comunicación de la BD con la web mediante el api |
| Condición para un fin exitoso | Que las gráficas se puedan ver dentro de su sección con los datos correctos. |
| Condición para un fin fallido | No poder ver las gráficas o gráficas con datos erróneos. |
| Actores primarios | Desarrolladores |
| Actores secundarios | Usuario |
| Trigger | N/A |
| Flujo Principal | |
| Pasos | Acciones |
| 1 | Hacer las consultas correctas para cada gráfica. |
| 2 | Mandar los resultados al canvas predeterminado anteriormente en el HTML con su id correspondiente. |
| 3 | Visualizar los resultados y comprobar si son correctos. |
| Extensiones | N/A |