



Tecnológico de Monterrey Campus Santa Fe

Software Requirements Specification

Paula Verdugo Marquez A01026218

Mariel Gómez Gutiérrez A01275607

Dulce García Ruíz A01748013

Andrés Tarazona Solloa A01023332

Santiago Rodríguez Palomo A01025232

Grupo 501

Construcción de Software y Toma de Decisiones

Mayo 4, 2023

Especificación de Requerimientos de Software

Historias de Usuario

Videojuego		
	Historia de usuario	Requerimientos
1	Como dueño del videojuego, quiero un comienzo y objetivo claro, que al terminarlo dé una sensación de final.	<ul style="list-style-type: none">● Programar que cuando los <i>health points</i> del jugador bajen de cero, sea un <i>lose condition</i>.● Crear una escena de Unity para que sea un nivel tutorial.● Programar que cuando el jugador derrote al <i>boss</i>, sea una <i>win condition</i>.
2	Como dueño del videojuego, quiero que los controles utilizados sean mínimos, y utilizados de la mejor manera.	<ul style="list-style-type: none">● Codificar una función en el script del jugador para el sistema de movimiento.● Crear funciones en scripts para apuntar y disparar con el mouse.● Programar un script para que la cámara siga al personaje.● Programar un script que despliegue la información del jugador, como qué gadgets posee y cómo son sus stats.
3	Como dueño del videojuego, quiero un personaje que pueda evolucionar y mejorar a través de las decisiones del jugador.	<ul style="list-style-type: none">● Generar dentro de un script la clase abstracta del personaje.● Asignar animaciones y colliders en el <i>gameobject</i> del personaje.● Crear una pantalla donde se pueda personalizar el personaje.
4	Como dueño del videojuego, quiero que la dificultad del juego incremente conforme progrese el jugador.	<ul style="list-style-type: none">● Generar la clase abstracta para los enemigos.● Crear el prefab para enemigo guardia.● Crear el prefab enemigo cámara.● Escribir un algoritmo similar a A* para el pathfinding de los guardias al jugador.
5	Como dueño del videojuego, quiero historia y cutscenes integradas al juego.	<ul style="list-style-type: none">● Generar una escena en Unity para que sea una cutscene al principio del juego.● Crear instancias de NPCs que ayuden a contar la historia del juego.

6	Como dueño del videojuego, quiero un juego que se pueda jugar varias veces.	<ul style="list-style-type: none"> • En el manejador del estado del juego, configurar el comportamiento de <i>respawn</i>. • Cargar los tiles, y crear los tilemaps para: ground, damaging_ground, walls, obstacles. • Generar y configurar un scene para que sea el primer nivel. • Generar y configurar un scene para que sea el segundo nivel. • Generar y configurar un scene para que sea el nivel boss. • Agregar efectos de sonido para el jugador y los enemigos. • Utilizar un controlador global que maneje la música del juego.
7	Como dueño del videojuego, quiero que las clases se sientan diferentes, pero que tengan posibilidades iguales.	<ul style="list-style-type: none"> • Codificar en un script la clase Cybergliadiator. • Codificar en un script la clase Codebreaker. • Codificar en un script la clase Ghostwalker. • Generar la clase abstracta para los gadgets. • Programar las clases para los gadgets del Cybergliadiator. • Programar las clases para los gadgets del Codebreaker. • Programar las clases para los gadgets del Ghostwalker.

Base de datos		
	Historia de usuario	Requerimientos
8	Como dueño de la base de datos, quiero tener un diagrama entidad-relación para una mejor comprensión.	<ul style="list-style-type: none"> • Crear diagramas entidad relación que demuestran como la base de datos que manejamos es relacional.
9	Como dueño de la base de datos, quiero una base de datos relacional, con tablas separadas, ambiente seguro, y con reglas de integridad.	<ul style="list-style-type: none"> • Crear un esquema de base de datos en MySQL que esté en la tercera forma normal.

10	Como dueño de la base de datos, necesito que mi base tenga un ambiente seguro y con reglas de integridad.	<ul style="list-style-type: none"> ● Implementar llaves primarias, elementos no nulos (NU) y cascading schemas.
11	Como dueño de la base de datos, quiero un API que permita un acceso fácil y seguro a la información guardada.	<ul style="list-style-type: none"> ● Crear un API web donde se implementen las operaciones CRUD.
12	Como dueño de la base de datos, quiero almacenar información acerca de escenarios, checkpoints, vida del jugador, daño causado por el jugador.	<ul style="list-style-type: none"> ● Crear un esquema de datos relacional donde se muestre el dominio y la naturaleza de las tablas.
13	Como dueño de la base de datos, quiero tener una medida de protección entre el front-end, back-end, y el middle-end.	<ul style="list-style-type: none"> ● Crear vistas en SQL para proteger la información y restringir el acceso a ciertos usuarios. ● Generar triggers en SQL que mantengan la integridad de la información.

Página web		
	Historia de usuario	Requerimientos
14	Como dueño de la página web, quiero una página web atractiva visualmente, que contenga toda la información necesaria. (Especificar las secciones)	<ul style="list-style-type: none"> ● Establecer una paleta de colores en todas las partes de la página web. ● Dibujar la planeación de secciones para el contenido de la página.
15	Como dueño de la página web, quiero que se puedan consultar las estadísticas de los usuarios a través de gráficas.	<ul style="list-style-type: none"> ● Obtener información desde la API y hacer múltiples visualizaciones de las estadísticas.
16	Como dueño de la página web, quiero secciones que expliquen el juego, y sus controles, como un manual.	<ul style="list-style-type: none"> ● Crear un video tutorial de cómo son las dinámicas del juego.
17	Como dueño de la página web, quiero que se utilicen Javascript, CSS, HTML, para el desarrollo, así como Node JS si es necesario.	<ul style="list-style-type: none"> ● Establecer un archivo CSS para los estilos globales. ● Escribir un archivo HTML y CSS para la explicación de juego, controles, y manual de usuario. ● Escribir un archivo HTML y CSS para

		<p>la sección about us.</p> <ul style="list-style-type: none"> ● Escribir un archivo HTML y CSS para la sección donde se va a embeber el juego. ● Escribir un archivo HTML y CSS para mostrar las estadísticas. ● Hostear la página web en la nube.
18	Como dueño de la página web, quiero una sección que tenga embebida el videojuego.	<ul style="list-style-type: none"> ● Subir el videojuego a la página y verificar su uso.

Requerimientos

Funcionales

Videojuegos

1. Programar que cuando los health points del jugador bajen de cero, sea un lose condition.
2. Crear una escena de Unity para que sea un nivel tutorial.
3. Programar que cuando el jugador derrote al boss, sea una win condition.
4. Codificar una función en el script del jugador para el sistema de movimiento.
5. Crear funciones en scripts para apuntar y disparar con el mouse.
6. Generar dentro de un script la clase abstracta del personaje.
7. Asignar animaciones y colliders en el gameobject del personaje.
8. Generar la clase abstracta para los enemigos.
9. Crear el prefab para enemigo guardia.
10. Crear el prefab enemigo cámara.
11. Escribir un algoritmo similar a A* para el pathfinding de los guardias al jugador.
12. Generar una escena en Unity para que sea una cutscene al principio del juego.
13. Crear instancias de NPCs que ayuden a contar la historia del juego.
14. En el manejador del estado del juego, configurar el comportamiento de respawn.
15. Cargar los tiles, y crear los tilemaps para: ground, damaging_ground, walls, obstacles.
16. Generar y configurar un scene para que sea el primer nivel.
17. Generar y configurar un scene para que sea el segundo nivel.
18. Generar y configurar un scene para que sea el nivel boss.
19. Agregar efectos de sonido para el jugador y los enemigos.
20. Utilizar un controlador global que maneje la música del juego.
21. Codificar en un script la clase Cybergliadiator.
22. Codificar en un script la clase Codebreaker.
23. Codificar en un script la clase Ghostwalker.
24. Generar la clase abstracta para los gadgets.
25. Programar las clases para los gadgets del Cybergliadiator.
26. Programar las clases para los gadgets del Codebreaker.
27. Programar las clases para los gadgets del Ghostwalker.

28. Programar un script para que la cámara siga al personaje.
29. Programar un script que despliegue la información del jugador, como qué gadgets posee y cómo son sus stats.

Base de datos

30. Crear diagramas entidad relación que demuestran como la base de datos que manejamos es relacional.
31. Crear un esquema de base de datos en MySQL que esté en la tercera forma normal.
32. Implementar llaves primarias, elementos no nulos (NU) y cascading schemas.
33. Crear un API web donde se implementen las operaciones CRUD.
34. Crear un esquema de datos relacional donde se muestre el dominio y la naturaleza de las tablas.
35. Crear vistas en SQL para proteger la información y restringir el acceso a ciertos usuarios.
36. Generar triggers en SQL que mantengan la integridad de la información.

Web

37. Obtener información desde la API y hacer múltiples visualizaciones de las estadísticas.
38. Crear un video tutorial de cómo son las dinámicas del juego.
39. Establecer un archivo CSS para los estilos globales.
40. Escribir un archivo HTML y CSS para la explicación de juego, controles, y manual de usuario.
41. Escribir un archivo HTML y CSS para la sección about us.
42. Escribir un archivo HTML y CSS para la sección donde se va a embeber el juego.
43. Escribir un archivo HTML y CSS para mostrar las estadísticas.
44. Subir el videojuego a la página y verificar su uso.

No Funcionales

45. Establecer una paleta de colores en todas las partes de la página web.
46. Crear una pantalla donde se pueda personalizar el personaje.
47. Dibujar la planeación de secciones para el contenido de la página.
48. Hostear la página web en la nube.
49. Crear un *hub world* donde se puedan checar stats y hablar con NPCs antes de respawnear.

Casos de Uso

Nombre		
Historias de usuario relacionadas		Historias de usuario:
Objetivo en contexto		
Precondiciones		
Condición de éxito		
Condición de fallo		
Actores primarios		
Actores secundarios		
Trigger		
Main Flow	Paso	Acción
	1	

Videojuegos

Nombre		Programar que cuando los health points del jugador bajen de cero, sea un lose condition.
Historias de usuario relacionadas		Historias de usuario: 2 y 6
Objetivo en contexto		Crear un script en el que se indique que al terminarse los health points del jugador, este pierde de manera automática (lose condition).
Precondiciones		Debe existir un jugador con el atributo health points.
Condición de éxito		Se crea de manera correcta el script donde si el contador de health points baja a cero, el juego se termina.
Condición de fallo		Existen errores en el script, no cumple con la condición dada anteriormente.
Actores primarios		Las y los desarrolladores
Actores secundarios		El usuario promedio
Trigger		Ninguno
Main Flow	Paso	Acción
	1	El usuario cuenta con un número de health points mayores a cero.
	2	Cada vez que el jugador sufre daño, el contador baja.
	3	Si el contador llega a cero, se termina el juego (loser condition).

Nombre		Crear una escena de Unity para que sea un nivel tutorial.
Historias de usuario relacionadas		Historias de usuario: 1 y 2
Objetivo en contexto		Idear una escena de Unity en la que el usuario pueda intuitivamente aprender los controles del videojuego.

Precondiciones		Deben servir de forma debida las mecánicas del juego.
Condición de éxito		Se logra atravesar el nivel entero sin bugs y con entendimiento de los controles.
Condición de fallo		Se causan bugs dentro del juego, o no es suficientemente intuitivo como para demostrar las
Actores primarios		Las y los desarrolladores
Actores secundarios		El usuario
Trigger		Primer inicio del juego.
Main Flow	Paso	Acción
	1	El usuario entra al nivel.
	2	Atraviesa el nivel donde se le presentan los controles y las mecánicas importantes.

Nombre	Programar que cuando el jugador derrote al boss, sea un win condition.
Historias de usuario relacionadas	Historias de usuario: 1, 5 y 6
Objetivo en contexto	Crear una condición donde los puntos de vida del boss son 0, en la que el jugador entonces gana el juego.
Precondiciones	Debe de haber mecánicas superables en el boss.
Condición de éxito	El jugador puede acabar los puntos de vida del boss.
Condición de fallo	El que los puntos de vida del boss lleguen a 0 no implica una condición de <i>win</i> .
Actores primarios	El usuario promedio.
Actores secundarios	Ninguno
Trigger	Acabar con los <i>health points</i> del boss.

Main Flow	Paso	Acción
	1	El jugador se enfrenta al boss.
	2	El jugador logra quitarle la vida entera al boss.
	3	El jugador gana el juego.

Nombre		Codificar una función en el script del jugador para el sistema de movimiento.
Historias de usuario relacionadas		Historias de usuario: 2 y 6.
Objetivo en contexto		El usuario debe de poder mover al jugador de manera fluida e intuitiva.
Precondiciones		Debe existir un jugador para mover.
Condición de éxito		El movimiento es fluido e intuitivo: un usuario con experiencia promedio lo puede entender sin explicaciones, no hay interrupciones, funciona como se esperaría.
Condición de fallo		El movimiento no es fluido ni intuitivo: no es claro cómo mover al personaje, se ve visualmente interrumpido, hay bugs o errores visibles, se alenta el movimiento.
Actores primarios		El usuario promedio.
Actores secundarios		Ninguno.
Trigger		El usuario presiona una tecla (w, a, s, d).
Main Flow	Paso	Acción
	1	El usuario dentro del juego presiona una tecla (w, a, s, d), mientras no esté en un cutscene.
	2	El jugador se mueve sin interrupciones dentro de los límites del mapa.

Nombre		Crear funciones en scripts para apuntar y disparar con el mouse.
Historias de usuario relacionadas		Historias de usuario: 2 y 6
Objetivo		Un usuario promedio debe, sin mucha dificultad, entender y controlar el sistema de ataque del jugador, utilizando el mouse.
Precondiciones		Necesita existir el jugador, así como un prefab para el láser.
Condición de éxito		Un usuario promedio, utilizando el mouse, puede disparar un láser. El láser se mueve a una velocidad constante, en la dirección que el usuario apuntó, y al impactar con una pared o un enemigo, es destruido, y hace daño si es aplicable.
Condición de fallo		El láser no se mueve, no hace daño, no se mueve en la dirección correcta, hay errores visuales, u otros errores o comportamientos no esperados.
Actores primarios		El usuario promedio.
Actores secundarios		Ninguno.
Trigger		Un movimiento o click del mouse.
Main Flow	Paso	Acción
	1	El usuario mueve el mouse a una coordenada dentro del juego.
	2	El usuario da un click, o presiona el mouse.
	3	Un proyectil láser emerge desde el jugador, con dirección a las coordenadas del mouse.
	4	El proyectil se mueve de manera continua hasta impactar con una pared, un obstáculo, o un enemigo guardia.
	5	Si impactó con un enemigo, le hace daño.

Nombre		Generar dentro de un script la clase abstracta del personaje.
Historias de usuario		Historias de usuario: 2, 3, 6, 7

relacionadas			
Objetivo			Tener la base a partir de la cual se pueden las clases de código para las clases jugables, heredando de una misma base.
Precondiciones			Ninguna.
Condición de éxito			Se crea un script en C# en la cual se defina una clase base para los personajes, siguiendo los diagramas UML de clase.
Condición de fallo			Hay errores en el script, no se siguieron los diagramas UML.
Actores primarios			Las y los desarrolladores.
Actores secundarios			Ninguno.
Trigger			Ninguno.
Main Flow	Paso	Acción	
	1	Ninguno. El script es para desarrollar procesos posteriormente.	

Nombre	Asignar animaciones y colliders en el gameobject del personaje.
Historias de usuario relacionadas	Historias de usuario: 3 y 6
Objetivo	Que los movimientos y comportamientos del personaje, se vean animados de manera correcta. Además, poner un box collider para el hitbox del personaje, y un circle trigger collider que será utilizado para ciertas habilidades.
Precondiciones	El gameobject del personaje.
Condición de éxito	Las animaciones tienen una velocidad apropiada, las animaciones utilizadas en cualquier momento corresponden con los <i>inputs</i> del usuario o estado del jugador. Los colliders corresponden con la posición del sprite y animación del jugador.
Condición de fallo	Las animaciones no se visualizan de manera correcta, o no corresponden con los <i>inputs</i> del usuario o estado del jugador. Los colliders están desfasados.
Actores primarios	Usuario promedio.
Actores secundarios	Ninguno.

Trigger		Movimiento del jugador, disparo del jugador, ningún <i>input</i> , muerte del jugador.
Main Flow	Paso	Acción
	1	Si no hay ningún input, se visualiza la animación <i>idle</i> .
	2	Si hay input de movimiento, se visualiza la animación de <i>run</i> .
	3	Si hay input de disparo, se visualiza la animación de <i>shoot</i> .
	4	Si el jugador muere, se visualiza la animación de <i>die</i> .

Nombre		Generar la clase abstracta para los enemigos.
Historias de usuario relacionadas		Historias de usuario: 4 y 6
Objetivo		Tener la base a partir de la cual se pueden las clases de código para los diferentes enemigos, heredando de una misma base.
Precondiciones		Ninguna.
Condición de éxito		Se crea un script en C# en la cual se defina una clase base para los enemigos, siguiendo los diagramas UML de clase.
Condición de fallo		Hay errores en el script, no se siguieron los diagramas UML.
Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno.
Trigger		Ninguno.
Main Flow	Paso	Acción
	1	Ninguno. El script es para desarrollar procesos posteriormente.

Nombre		Crear el prefab para enemigo guardia.
Historias de usuario relacionadas		Historias de usuario: 4 y 6.
Objetivo		Tener un prefab, que puede ser reutilizado en el desarrollo de los niveles. El prefab tendrá casi todas las funcionalidades del

	guardia, a excepción del pathfinding.	
Precondiciones	Existe la clase abstracta para los enemigos.	
Condición de éxito	Se crea un prefab, con scripts, colliders, animaciones, siguiendo los comportamientos descritos en el GDD y en los diagramas de clase UML.	
Condición de fallo	Al prefab le falta un elemento de los descritos anteriormente, o no sigue los comportamientos y clases previamente planeados.	
Actores primarios	Las y los desarrolladores.	
Actores secundarios	El usuario promedio.	
Trigger	Ninguno.	
Main Flow	Paso	Acción
	1	Cuando el usuario promedio juega, los guardias se comportan de una manera esperada.

Nombre	Crear el prefab enemigo cámara.	
Historias de usuario relacionadas	Historias de usuario: 4 y 6.	
Objetivo	Tener un prefab, que puede ser reutilizado en el desarrollo de los niveles. El prefab tendrá todas las funcionalidades de las cámaras.	
Precondiciones	Existe la clase abstracta para los enemigos.	
Condición de éxito	Se crea un prefab, con scripts, colliders, siguiendo los comportamientos descritos en el GDD y en los diagramas de clase UML.	
Condición de fallo	Al prefab le falta un elemento de los descritos anteriormente, o no sigue los comportamientos y clases previamente planeados.	
Actores primarios	Las y los desarrolladores.	
Actores secundarios	El usuario promedio.	
Trigger	Ninguno.	
Main Flow	Paso	Acción

	1	Cuando el usuario promedio juega, las cámaras se comportan de una manera esperada.
--	----------	------------------------------------------------------------------------------------

Nombre	Escribir un algoritmo similar a A* para el pathfinding de los guardias al jugador.	
Historias de usuario relacionadas	Historias de usuario: 4.	
Objetivo	Desarrollar un algoritmo de pathfinding para que los enemigos puedan tener sus paths definidos, pero igual puedan adaptarse a estados en alarma, entre otros.	
Precondiciones	Los guardias pueden moverse dentro de la escena y hay forma de dividir el tileset para poder implementar A*.	
Condición de éxito	Los enemigos pueden usar su pathfinding para tener movimiento claro.	
Condición de fallo	Los enemigos no siguen un algoritmo A*.	
Actores primarios	Las y los desarrolladores.	
Actores secundarios	Ninguno.	
Trigger	Ninguno.	
Main Flow	Paso	Acción
	1	Se definen puntos de entrada y de salida para un enemigo.
	2	Se utiliza A* para lograr llevar al enemigo de un punto de entrada a uno de salida, o viceversa si el ciclo termina.

Nombre	Generar una escena en Unity para que sea una cutscene al principio del juego.
Historias de usuario relacionadas	Historias de usuario: 1.
Objetivo	Generar cutscenes para el inicio del juego en Unity.

Precondiciones		La escena en Unity está preparada de forma correcta para animar la cutscene y que pueda hacer transición de forma clara.
Condición de éxito		La cutscene se ve de forma correcta y es entendible al vidente.
Condición de fallo		La cutscene no se ve, o no se logra discernir la historia.
Actores primarios		Las y los desarrolladores.
Actores secundarios		El usuario promedio.
Trigger		Primer inicio del juego.
Main Flow	Paso	Acción
	1	El jugador pone en “jugar” por primera vez.
	2	La cutscene se pone de forma automática para el inicio del juego.

Nombre		Crear instancias de NPCs que ayuden a contar la historia del juego.
Historias de usuario relacionadas		Historias de usuario: 5.
Objetivo		Generar personajes no jugables que contengan diálogo relevante al juego.
Precondiciones		Tener diálogo definido, y sprites para la implementación del NPC.
Condición de éxito		Hay un NPC con el que se puede conversar.
Condición de fallo		No se logra implementar el NPC.
Actores primarios		Las y los desarrolladores.
Actores secundarios		El jugador promedio.
Trigger		Clic en el NPC.
Main Flow	Paso	Acción
	1	El jugador hace clic sobre el NPC.
	2	Se despliega el diálogo del NPC.

Nombre		En el manejador del estado del juego, configurar el comportamiento de respawn.
Historias de usuario relacionadas		Historias de usuario: 2, 4 y 6.
Objetivo		Que haya un sistema de respawn, donde el jugador pueda volver a empezar el juego desde el primer nivel, manteniendo un cierto nivel de progreso con sus gadgets.
Precondiciones		Se puede empezar el juego una y otra vez sin que se vuelva a poner la cutscene de inicio, o el nivel de tutorial.
Condición de éxito		Después que el jugador pierda todos sus puntos de vida, se reinicia el juego, y el jugador hace respawn.
Condición de fallo		Cuando el jugador muere, el juego hace crash o termina solamente.
Actores primarios		Las y los desarrolladores.
Actores secundarios		El usuario promedio.
Trigger		Muerte del jugador.
Main Flow	Paso	Acción
	1	El jugador muere.
	2	Se hace respawn en el nivel 1, con la posibilidad de volver a intentar pasar el juego.

Nombre		Cargar los tiles, y crear los tilemaps para: ground, damaging_ground, walls, obstacles.
Historias de usuario relacionadas		Historias de usuario: 2, 5 y 6
Objetivo		Crear una paleta de tiles, así como tilemaps diferentes para diferentes partes del escenario. Tener un nivel pequeño en donde se pueda probar el desarrollo antes de empezar la creación de los niveles.

Precondiciones		Ninguna.
Condición de éxito		Se crean los cuatro diferentes tilemaps, utilizando una paleta de tiles, y se crea un nivel pequeño en donde todos los tiles se ven bien.
Condición de fallo		No se crean los cuatro tilemaps, algún tile se ve mal, o no se ven bien las transiciones entre tiles.
Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno.
Trigger		Ninguno.
Main Flow	Paso	Acción
	1	Ninguno. No es un proceso, pero es necesario para desarrollo posterior.

Nombre		Generar y configurar un scene para que sea el primer nivel.
Historias de usuario relacionadas		Historias de usuario: 1, 3 y 6.
Objetivo		Configurar el tilemap y el posicionamiento de todos los obstáculos/guardias, para que formen el primer nivel definido originalmente.
Precondiciones		Se pueden editar los niveles, y se tienen todas las partes hechas, como por ejemplo las cámaras o los guardias.
Condición de éxito		El nivel 1 se puede pasar sin problema alguno.
Condición de fallo		El nivel 1 presenta algún problema, o no se logra crear en absoluto.
Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno.
Trigger		Ninguno.
Main Flow	Paso	Acción
	1	Desarrollo completo del tilemap.

	2	Se posicionan los enemigos y las cámaras.
--	----------	-------------------------------------------

Nombre		Generar y configurar un scene para que sea el segundo nivel.
Historias de usuario relacionadas		Historias de usuario: 1, 3 y 6.
Objetivo		Configurar el tilemap y el posicionamiento de todos los obstáculos/guardias, para que formen el segundo nivel definido originalmente.
Precondiciones		Se pueden editar los niveles, y se tienen todas las partes hechas, como por ejemplo las cámaras o los guardias.
Condición de éxito		El nivel 2 se puede pasar sin problema alguno.
Condición de fallo		El nivel 2 presenta algún problema, o no se logra crear en absoluto.
Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno.
Trigger		Ninguno.
Main Flow	Paso	Acción
	1	Desarrollo completo del tilemap.
	2	Se posicionan los enemigos y las cámaras.

Nombre		Generar y configurar un scene para que sea el nivel boss.
Historias de usuario relacionadas		Historias de usuario: 1, 3 y 6.
Objetivo		Configurar el tilemap y el posicionamiento de todos los obstáculos/guardias, para que formen el nivel de boss definido originalmente.
Precondiciones		Se pueden editar los niveles, y se tienen todas las partes hechas, como por ejemplo las cámaras o los guardias. Hay mecánicas de

	boss suficientemente variadas para el uso de las tres clases independientemente.	
Condición de éxito	El nivel se puede pasar sin problema alguno, con cualquiera de las clases.	
Condición de fallo	El nivel presenta algún problema, o no se logra crear en absoluto, también si acaso no se puede pasar el nivel con alguna clase específicamente.	
Actores primarios	Las y los desarrolladores.	
Actores secundarios	Ninguno.	
Trigger	Ninguno.	
Main Flow	Paso	Acción
	1	Desarrollo completo del tilemap.
	2	Se posicionan los enemigos y las cámaras.

Nombre	Agregar efectos de sonido para el jugador y los enemigos.	
Historias de usuario relacionadas	Historias de usuario: 6	
Objetivo en contexto	Poder añadir elementos de audio al jugador y al enemigo para generar un ambiente más llamativo y completo.	
Precondiciones	Existencia del jugador y enemigo.	
Condición de éxito	Los efectos de sonido tiene cohesión y son de apoyo para el videojuego, generando un ambiente cohesivo.	
Condición de fallo	Los efectos de sonido tienen algún fallo y no son aptos para el contexto.	
Actores primarios	Las y los programadores.	
Actores secundarios	Ninguno	
Trigger	Ninguno	
Main Flow	Paso	Acción

	1	Seleccionar los elementos de audio.
	2	Añadirlos al proyecto y a las acciones del jugador y enemigo.

Nombre		Utilizar un controlador global que maneje la música del juego.
Historias de usuario relacionadas		Historias de usuario: 1 y 5.
Objetivo en contexto		Añadir música para todos los niveles, que vaya evolucionando conforme los niveles vayan progresando.
Precondiciones		Existen canciones independientes para todos los niveles, así mismo como motivos de transición entre ellas.
Condición de éxito		La música suena, de forma correcta, en todos los niveles que se puedan jugar.
Condición de fallo		Alguno de los tracks de música no se puede reproducir, o no se logra conseguir independencia musical entre los niveles.
Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno.
Trigger		Inicio de cada nuevo nivel.
Main Flow	Paso	Acción
	1	Se desarrolla la música para todos los niveles.
	2	Se toca la música correspondiente a ese nivel, en todos los niveles.

Nombre		Codificar en un script la clase Cybergadiator.
Historias de usuario relacionadas		Historias de usuario: 2, 3, 4, 6, y 7.
Objetivo		Crear un script, una clase que hereda de la clase base del jugador, la cual será utilizada para la clase jugable de Cybergadiator.

Precondiciones		La clase base del jugador.
Condición de éxito		Se crea un script en unity para la clase de Cybergladiator, siguiendo los lineamientos establecidos en el GDD y los diagramas de clase UML.
Condición de fallo		Hay errores en el script, o se sale de los lineamientos.
Actores primarios		Las y los desarrolladores.
Actores secundarios		El usuario promedio.
Trigger		Si el usuario escoge esta clase.
Main Flow	Paso	Acción
	1	Si el usuario escoge esta clase, va a ser la clase principal a través de la cual el usuario interactúa con el jugador.

Nombre		Codificar en un script la clase Codebreaker.
Historias de usuario relacionadas		Historias de usuario: 2, 3, 4, 6, y 7.
Objetivo		Crear un script, una clase que hereda de la clase base del jugador, la cual será utilizada para la clase jugable de Codebreaker.
Precondiciones		La clase base del jugador.
Condición de éxito		Se crea un script en unity para la clase de Codebreaker, siguiendo los lineamientos establecidos en el GDD y los diagramas de clase UML.
Condición de fallo		Hay errores en el script, o se sale de los lineamientos.
Actores primarios		Las y los desarrolladores.
Actores secundarios		El usuario promedio.
Trigger		Si el usuario escoge esta clase.
Main Flow	Paso	Acción
	1	Si el usuario escoge esta clase, va a ser la clase principal a través de la cual el usuario interactúa con el jugador.

Nombre		Codificar en un script la clase Ghostwalker.
Historias de usuario relacionadas		Historias de usuario: 2, 3, 4, 6, y 7.
Objetivo		Crear un script, una clase que hereda de la clase base del jugador, la cual será utilizada para la clase jugable de Ghostwalker.
Precondiciones		La clase base del jugador.
Condición de éxito		Se crea un script en unity para la clase de Ghostwalker, siguiendo los lineamientos establecidos en el GDD y los diagramas de clase UML.
Condición de fallo		Hay errores en el script, o se sale de los lineamientos.
Actores primarios		Las y los desarrolladores.
Actores secundarios		El usuario promedio.
Trigger		Si el usuario escoge esta clase.
Main Flow	Paso	Acción
	1	Si el usuario escoge esta clase, va a ser la clase principal a través de la cual el usuario interactúa con el jugador.

Nombre		Generar la clase abstracta para los gadgets.
Historias de usuario relacionadas		Historias de usuario: 2, 3, 4, 6 y 7
Objetivo		Generar en un script de C#, una clase abstracta la cual será utilizada para desarrollar los gadgets posteriormente.
Precondiciones		Ninguna.
Condición de éxito		Se siguen los lineamientos establecidos en el GDD y en el diagrama de clases UML.
Condición de fallo		Hay errores en el script o no se siguieron los lineamientos.
Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno.

Trigger		Ninguno.
Main Flow	Paso	Acción
	1	Al ser una clase abstracta, no hay proceso, pero es necesario para el desarrollo posterior.

Nombre	Programar las clases para los gadgets del Cybergladiator.	
Historias de usuario relacionadas	Historias de usuario: 2, 3, 4, 6 y 7	
Objetivo en contexto	Crear funciones en scripts de C# para los 3 gadgets del Cybergladiator. Igualmente se asigna un gráfico respectivamente.	
Precondiciones	Clase Cybergladiator inicializada.	
Condición de éxito	Cada gadget se comunica con la clase Cybergladiator efectivamente y se cumplen los lineamientos establecidos en el GDD y en el diagrama de clases UML.	
Condición de fallo	Hay errores en el script o no se siguieron los lineamientos.	
Actores primarios	El usuario promedio.	
Actores secundarios	Ninguno.	
Trigger	El usuario escoge en el menú el gadget, subsecuentemente recibe habilidades diferentes.	
Main Flow	Paso	Acción
	1	El usuario encuentra un gadget en el nivel.
	2	El usuario escoge su gadget en el menú.
	3	El usuario gana habilidades respectivamente.

Nombre	Programar las clases para los gadgets del Codebreaker.	
Historias de usuario relacionadas	Historias de usuario: 2, 3, 4, 6 y 7	
Objetivo en contexto	Crear funciones en scripts de C# para los 3 gadgets del Codebreaker. Igualmente se asigna un gráfico respectivamente.	
Precondiciones	Clase Codebreaker inicializada.	
Condición de éxito	Cada gadget se comunica con la clase Codebreaker efectivamente y se cumplen los lineamientos establecidos en el GDD y en el diagrama de clases UML.	
Condición de fallo	Hay errores en el script o no se siguieron los lineamientos.	
Actores primarios	El usuario promedio.	
Actores secundarios	Ninguno.	
Trigger	El usuario escoge en el menú el gadget, subsecuentemente recibe habilidades diferentes.	
Main Flow	Paso	Acción
	1	El usuario encuentra un gadget en el nivel.
	2	El usuario escoge su gadget en el menú.
	3	El usuario gana habilidades respectivamente.

Nombre	Programar las clases para los gadgets del Ghostwalker.	
Historias de usuario relacionadas	Historias de usuario: 2, 3, 4, 6 y 7	
Objetivo en contexto	Crear funciones en scripts de C# para los 3 gadgets del Codebreaker. Igualmente se asigna un gráfico respectivamente.	
Precondiciones	Clase Ghostwalker inicializada.	
Condición de éxito	Cada gadget se comunica con la clase Ghostwalker	

		efectivamente y se cumplen los lineamientos establecidos en el GDD y en el diagrama de clases UML.
Condición de fallo		Hay errores en el script o no se siguieron los lineamientos.
Actores primarios		El usuario promedio.
Actores secundarios		Ninguno.
Trigger		El usuario escoge en el menú el gadget, subsecuentemente recibe habilidades diferentes.
Main Flow	Paso	Acción
	1	El usuario encuentra un gadget en el nivel.
	2	El usuario escoge su gadget en el menú.
	3	El usuario gana habilidades respectivamente.

Nombre		Programar un script para que la cámara siga al personaje.
Historias de usuario relacionadas		Historias de usuario: 2 y 5
Objetivo en contexto		Crear un script para que la cámara del usuario siga al personaje que se encuentra estático.
Precondiciones		Debe de existir un jugador.
Condición de éxito		Cuándo el usuario se mueve con los controles, el personaje sigue visible y el escenario se va moviendo conforme avance.
Condición de fallo		El personaje no es visible o la cámara sale de escena. Se siguen los lineamientos establecidos en el GDD y los diagramas de clase UML.
Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno.
Trigger		Cuando se inicializa el personaje y cuando el usuario utiliza los controles para moverse.
Main Flow	Paso	Acción
	1	El usuario comienza su juego y su personaje es inicializado.

	2	El usuario utiliza los controles para moverse y la cámara lo sigue.
--	----------	---------------------------------------------------------------------

Nombre	Programar un script que despliegue la información del jugador, como qué gadgets posee y cómo son sus stats.	
Historias de usuario relacionadas	Historias de usuario: 2.	
Objetivo en contexto	Programar un script que tome el input de alguna tecla predeterminada, y pueda desplegar los datos importantes del usuario, como sus gadgets y sus stats importantes.	
Precondiciones	Que existan datos que desplegar.	
Condición de éxito	Se puede hacer toggle del <i>overlay</i> de datos importantes.	
Condición de fallo	No se logra hacer ningún toggle, o no se despliegan apropiadamente los datos.	
Actores primarios	Las y los desarrolladores.	
Actores secundarios	El usuario promedio.	
Trigger	Alguna tecla predefinida.	
Main Flow	Paso	Acción
	1	Se presiona la tecla.
	2	Los stats se despliegan en una pantalla predeterminada.

Base de datos

Nombre	Crear diagramas entidad relación que demuestra como la base de datos que manejamos es relacional.
Historias de usuario relacionadas	Historias de usuario: 8 y 9
Objetivo en contexto	Crear diagramas que muestren de manera clara y concisa la

		estructura de nuestra base de datos relacional.
Precondiciones		Ninguna
Condición de éxito		Se crean los diagramas tomando en cuenta las características descritas para la base de datos. Son claros y representan un apoyo visual.
Condición de fallo		Los diagramas no están bien contruidos, faltan condiciones o datos que son importantes àra la construcción de la base de datos
Actores primarios		Las y los programadores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Definir las características de la base de datos.

Nombre	Crear un esquema de base de datos en MySQL que esté en la tercera forma normal.
Historias de usuario relacionadas	Historias de usuario: 9
Objetivo en contexto	Crear de manera correcta y estructurada una base de datos que cumpla con la tercera forma normal.
Precondiciones	Ninguno
Condición de éxito	La base de datos creada en MySQL está normalizada (tomando en cuenta hasta la tercera forma normal) y estructurada de manera correcta.
Condición de fallo	La base de datos no está normalizada o presenta algún fallo en su estructura.
Actores primarios	Las y los programadores
Actores secundarios	Ninguno

Trigger		Ninguno
Main Flow	Paso	Acción
	1	Ninguno. No es un proceso, pero es necesario para desarrollo posterior.

Nombre	Implementar llaves primarias, elementos no nulos (NU) y cascading schemas.	
Historias de usuario relacionadas	Historias de usuario: 10	
Objetivo	Implementar de manera correcta llaves primarias, elementos no nulos (NU) y cascading schemas.	
Precondiciones	Tener nuestros diagramas entidad relación	
Condición de éxito	La base de datos implementa de manera correcta llaves primarias, elementos no nulos (NU) y cascading schemas.	
Condición de fallo	Existen fallas en la base de datos o no se emplean de manera correcta las llaves.	
Actores primarios	Las y los programadores	
Actores secundarios	Ninguno	
Trigger	Ninguno	
Main Flow	Paso	Acción
	1	Ninguno. No es un proceso, pero es necesario para desarrollo posterior.

Nombre		Crear un API web donde se implementen las operaciones CRUD
Historias de usuario relacionadas		Historias de usuario: 11
Objetivo		Poder analizar la información que el cliente nos solicitó, además, permitirá un acceso fácil y seguro a la información guardada.
Precondiciones		Una implementación correcta de la base de datos.
Condición de éxito		Se crea un API web donde se implementan las operaciones CRUD y se pueden ejecutar correctamente.
Condición de fallo		No se pueden implementar las operaciones CRUD.
Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Se crea la API web

Nombre		Crear un esquema de datos relacional donde se muestre el dominio y la naturaleza de las tablas.
Historias de usuario relacionadas		Historias de usuario: 12
Objetivo en contexto		Almacenar información acerca de escenarios, checkpoints, vida del jugador, daño causado por el jugador.
Precondiciones		Diseñar la base de datos de manera correcta.
Condición de éxito		Se crea un esquema de datos relacional donde se muestra el dominio y la naturaleza de las tablas.

Condición de fallo		No se muestra el dominio ni la naturaleza de las tablas en la creación del esquema de datos relacionales.
Actores primarios		Las y los desarrolladores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Crear diagramas de entidad
	2	Generar tablas que estén bajo la tercera forma normal
	3	Administrar la base de datos.

Nombre		Crear vistas en SQL para proteger la información y restringir el acceso a ciertos usuarios.
Historias de usuario relacionadas		Historias de usuario: 13
Objetivo en contexto		Tener una medida de protección entre el front-end, back-end, y el middle-end.
Precondiciones		Tener implementada la base de datos.
Condición de éxito		Se logró proteger la información y restringir el acceso a ciertos usuarios.
Condición de fallo		No se mantuvo la protección de la información.
Actores primarios		Las y los desarrolladores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Agregar las líneas necesarias en el código SQL para generar

		las vistas.
--	--	-------------

Nombre		Generar triggers en SQL que mantengan la integridad de la información.
Historias de usuario relacionadas		Historias de usuario: 13
Objetivo en contexto		Tener una medida de protección entre el front-end, back-end, y el middle-end.
Precondiciones		Tener implementada la base de datos.
Condición de éxito		Se mantuvo la protección del front-end, back-end, y el middle-end.
Condición de fallo		No se mantuvo la protección de la información pues se modificó la base de datos.
Actores primarios		Las y los desarrolladores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Agregar las líneas necesarias en el código SQL para generar los triggers.

Web

Nombre		Obtener información desde la API y hacer múltiples visualizaciones de las estadísticas
Historias de usuario relacionadas		Historias de usuario: 14, 15 y 17

Objetivo en contexto		En la página web existe una sección donde se encuentran métricas de las estadísticas obtenidas.
Precondiciones		Existe un registro de métricas del juego de usuarios pasados.
Condición de éxito		Se muestran gráficas que enseñan datos a tiempo real de las diferentes métricas seleccionadas.
Condición de fallo		Las gráficas no se actualizan, los datos mostrados son incorrectos, o no se muestran los datos.
Actores primarios		Las y los desarrolladores.
Actores secundarios		El usuario promedio.
Trigger		Ninguno.
Main Flow	Paso	Acción
	1	El usuario juega una partida en la página Web.
	2	El juego del usuario se comunica con la base de datos y se añaden los resultados y se comparan con los de usuarios anteriores.
	3	La página muestra resultados de todas las estadísticas que se encuentran en la base de datos.

Nombre	Crear un video tutorial de cómo son las dinámicas del juego.
Historias de usuario relacionadas	Historias de usuario: 16
Objetivo en contexto	Crear un video que le muestre a un usuario promedio a jugar el videojuego enseñando las dinámicas básicas con las que este cuenta.
Precondiciones	Definir dinámicas de juego. Creación de personajes e historia.
Condición de éxito	Se crea un video corto y entendible donde se muestran las dinámicas del juego. El usuario promedio puede comprenderlo.
Condición de fallo	El video no explica de manera clara las dinámicas. Su comprensión es difícil.

Actores primarios		Las y los programadores
Actores secundarios		El usuario promedio
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Definir las dinámicas e historia del juego
	2	Crear de manera gráfica la explicación de las mismas.

Nombre		Escribir un archivo HTML y CSS para la sección donde se va a embeber el juego.
Historias de usuario relacionadas		Historias de usuario: 17
Objetivo en contexto		Que la sección donde se va a embeber el juego tenga una buena presentación
Precondiciones		Ninguna
Condición de éxito		Establecer correctamente los archivos HTML y CSS para la sección en donde se embebió el juego.
Condición de fallo		No se establecen correctamente los archivos HTML y CSS.
Actores primarios		Las y los desarrolladores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Generar el archivo

Nombre		Escribir un archivo HTML y CSS para mostrar las estadísticas.
Historias de usuario relacionadas		Historias de usuario: 17
Objetivo en contexto		Mostrar las estadísticas
Precondiciones		Ninguna
Condición de éxito		Se logran mostrar las estadísticas.
Condición de fallo		No se logran mostrar las estadísticas
Actores primarios		Las y los desarrolladores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Generar el archivo

Nombre		Subir el videojuego a la página y verificar su uso.
Historias de usuario relacionadas		Historias de usuario: 18
Objetivo en contexto		Verificar el correcto desarrollo del videojuego en la sección que tenga lo tenga embebido.
Precondiciones		El correcto desarrollo de todo el software.
Condición de éxito		Se ejecuta correctamente el videojuego dentro de la página web
Condición de fallo		No se ejecuta correctamente
Actores primarios		Las y los desarrolladores
Actores secundarios		Ninguno
Trigger		Ninguno

Main Flow	Paso	Acción
	1	Subir el videojuego a la página.
	2	Verificar su uso

Nombre	Establecer un archivo CSS para los estilos globales.	
Historias de usuario relacionadas	Historias de usuario: 16	
Objetivo en contexto	Crear un script CSS que contenga los estilos globales que utilizaremos en la página web, con el fin que se vea unificada y armoniosa.	
Precondiciones	Ninguna	
Condición de éxito	Se crea un archivo CSS que contiene todos los estilos globales que dan el formato deseado.	
Condición de fallo	El archivo presenta fallas o un funcionamiento diferente al deseado.	
Actores primarios	Las y los programadores	
Actores secundarios	Ninguno	
Trigger	Ninguno	
Main Flow	Paso	Acción
	1	No es un proceso, pero facilita el desarrollo de la página web.

Nombre	Escribir un archivo HTML y CSS para la explicación de juego, controles, y manual de usuario.	
Historias de usuario relacionadas	Historias de usuario: 16	

Objetivo		Tener un archivo HTML y CSS para la sección destinada a explicar el juego a un usuario promedio.
Precondiciones		Ninguna
Condición de éxito		Los archivos contienen secciones claras que sean entendibles para el usuario promedio y comuniquen de manera clara nuestras dinámicas.
Condición de fallo		Los archivos contienen secciones poco estructuradas y no cumplen con lo requerido para explicarle al usuario el juego.
Actores primarios		Los programadores
Actores secundarios		El usuario promedio
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Se define lo que se le quiere comunicar al usuario (dinámicas del juego)
	2	Se plantea una estructura para esta sección

Nombre	Escribir un archivo HTML y CSS para la sección about us.
Historias de usuario relacionadas	Historias de usuario: 16
Objetivo	Tener un archivo HTML y un CSS que contengan la sección de about us.
Precondiciones	Ninguna
Condición de éxito	La sección está bien estructurada, comunica de manera correcta el contenido y es atractiva para el usuario promedio.
Condición de fallo	La sección no es atractiva y no cuenta con estructura o contenido solicitado.
Actores primarios	Las y los programadores
Actores secundarios	El usuario promedio
Trigger	Ninguno

Main Flow	Paso	Acción
	1	No es un proceso, pero facilita el desarrollo de la página web.

No funcionales

Nombre		Establecer una paleta de colores en todas las partes de la página web.
Historias de usuario relacionadas		Historias de usuario: 14
Objetivo en contexto		Poder tener una paleta de colores unificada para una correcta implementación de elementos gráficos en la página web.
Precondiciones		Ninguna
Condición de éxito		La paleta refleja el tema del juego y es llamativa para el usuario promedio
Condición de fallo		La paleta de colores no es llamativa y tampoco presenta cohesión.
Actores primarios		Las y los programadores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	No es un proceso, pero facilita el desarrollo de la página web.

Nombre		Crear una pantalla donde se pueda personalizar el personaje.
Historias de usuario relacionadas		Historias de usuario: 3
Objetivo en contexto		Se puede personalizar al personaje por medio de una pantalla
Precondiciones		Tener un personaje y poderlo elegir.
Condición de éxito		Se puede ver una pantalla correctamente donde se pueda personalizar al personaje
Condición de fallo		No se puede visualizar la pantalla
Actores primarios		Las y los desarrolladores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Generar la pantalla

Nombre		Dibujar la planeación de secciones para el contenido de la página.
Historias de usuario relacionadas		Historias de usuario: 14
Objetivo		Tener un lineamiento gráfico para agilizar el proceso del desarrollo web, así como plasmar las ideas y que el equipo de desarrollo esté en la misma página.
Precondiciones		Ninguna.
Condición de éxito		Se crea, a través de un dibujo o una herramienta tecnológica, un planteamiento gráfico estético para la página web, que incluya todas las secciones: manual de usuario, about us, home (donde se embebe el juego), y las estadísticas.
Condición de fallo		No están todas las secciones completas, o la página web no es coherente ni estética.

Actores primarios		Las y los desarrolladores.
Actores secundarios		Ninguno.
Trigger		Ninguno.
Main Flow	Paso	Acción
	1	No es un proceso, pero facilita el desarrollo de la página web.

Nombre		Hostear la página web en la nube.
Historias de usuario relacionadas		Historias de usuario: 17
Objetivo en contexto		Que la página web esté correctamente hosteada para su visualización y su uso.
Precondiciones		Implementar correctamente el hosting para la página web
Condición de éxito		Se hostea correctamente la página web
Condición de fallo		No se puede visualizar la página web
Actores primarios		Las y los desarrolladores
Actores secundarios		Ninguno
Trigger		Ninguno
Main Flow	Paso	Acción
	1	Hostear la página web correctamente.