



# Tecnológico de Monterrey

**Construcción de Software y Toma de Decisiones**

**Análisis de Requerimientos**

**Alumnos:**

Juan Pablo Cruz Rodríguez - A01783208

David Medina Domínguez - A01783155

Angel Adrian Alfonso - A01782545

Daniel Soult Gómez - A01782985

**Campus Santa Fe**

**Grupo 501**

**Mayo 2023**

## Requerimientos de Software y Metodología Ágil:

Los requerimientos de software son descripciones detalladas de las funciones y características que se necesitan para un sistema o aplicación de software. Estos requerimientos pueden incluir aspectos como la funcionalidad, la usabilidad, el rendimiento, la seguridad y otros aspectos técnicos y no técnicos. La importancia de los requerimientos de software radica en que proporcionan una guía clara y detallada para el diseño, desarrollo, implementación y prueba de un sistema o aplicación de software.

### Historias de Usuario:

Las historias de usuario son una técnica de gestión de requisitos en el desarrollo de software que se utiliza para describir las características y funcionalidades que se desean implementar en un producto o sistema. Las historias de usuario son descripciones breves y simples de una funcionalidad desde la perspectiva del usuario, que se escriben en un lenguaje natural y se enfocan en las necesidades y objetivos del usuario.

La importancia de las historias de usuario radica en que ayudan a los equipos de desarrollo de software a comprender las necesidades de los usuarios y a construir un producto que cumpla con sus expectativas. Las historias de usuario proporcionan una descripción clara y concisa de las funcionalidades requeridas, lo que ayuda a los desarrolladores a concentrarse en los detalles relevantes y a evitar el desperdicio de tiempo y esfuerzo en funcionalidades innecesarias.

ID Historia de Usuario	Como<Rol>	Quiero<Eventos>	Para<Objetivo>	Prioridad 1-5	Estimación
1	Socio Fo... ▾	que el juego sea RPG	establecer ciertas mecánicas y características	5	240h
2	Socio Fo... ▾	que el juego sea atractivo	el usuario le de ganas de jugar de nuevo	5	240h

ID Historia de Usuario (Esteban)	Como<Rol>	Quiero<Eventos>	Para<Objetivo>	Prioridad 1-5	Estimación
1	Product ... ▾	que se use una base de datos relacional	para consultar los datos de forma eficaz y optimizada	5	240h
2	Product ... ▾	se proteja la integridad de los datos	para evitar datos incorrectos, duplicados, faltantes o alterados	5	240h
3	Product ... ▾	usar llaves de optimización	para una búsqueda en la tabla de forma rápida	4	240h
4	Product ... ▾	normalizar la base de datos	reducir la redundancia de los datos y aumentar la integridad de los mismos	4	240h
5	Product ... ▾	utilizar una API	para conectar la página web, el juego y la base de datos	4	240h
6	Product ... ▾	usar datos atómicos para la tablas	desglosar la información a un punto fácil de entender	3	240h
7	Product ... ▾	implementar un nivel introductorio	reconocer los controles o historia del juego	3	240h
8	Product ... ▾	controles o mecánicas intuitivas	para facilitar el aprendizaje del jugador	4	240h
9	Product ... ▾	opción de inicio de sesión	para llevar un registro de los datos del usuario	3	240h

ID Historia de Usuario (Gilberto)	Como<Rol>	Quiero<Eventos>	Para<Objetivo>	Prioridad 1-5	Estimación
1	Product ... ▾	que el juego tenga un objetivo	para darle una razón al jugador para completar el juego	5	240h
2	Product ... ▾	implementar mecanica de crecimiento	para darle la sensación al jugador de estar mejorando en el juego	4	240h
3	Product ... ▾	usar máquinas de estados para la toma de de decisiones	para una mejor visualización de los posible caminos	4	240h

ID Historia de Usuario (Octavio)	Como<Rol>	Quiero<Eventos>	Para<Objetivo>	Prioridad 1-5	Estimación
1	Product ... ▾	que la página web sea segura	para mantener los datos del usuario	5	240h
2	Product ... ▾	que tenga un interfaz de usuario intuitiva	para que se le facilite al usuario ubicarse y encontrar cada apartado	3	240h
3	Product ... ▾	que tenga sonido y gráficos de calidad	para una experiencia inmersiva y gratificante	2	240h
4	Product ... ▾	que el juego sea desafiante	para mantener el interés y el compromiso del jugador	5	240h
5	Product ... ▾	implementar una función de registro	para guardar la información y la partida del jugador	3	240h

## Product Backlog:

El Product Backlog es una lista ordenada de requerimientos o funcionalidades que se desean incluir en un producto o sistema en desarrollo. El Product Backlog es un componente clave de la metodología ágil de desarrollo de software, y se utiliza para gestionar los requisitos del proyecto y planificar el trabajo a realizar en cada iteración.

- **Funcionales**

- Socio Formador

- **RPG**

- Variedad de personajes
      - Sistema de combate
      - Progresion de Personaje
      - Historia y Misiones
      - Sistema de Inventario
      - Jugabilidad Intuitiva

- **Juego Atractivo**

- Gráficos de Calidad
      - Sonidos de Calidad
      - Historia y Personajes Interesantes
      - Retos Desafiantes

- Product Owner Esteban Castillo Juarez

- **Utilizar una base de datos relacional**

- Almacenamiento y Organización de los datos
      - Integridad de los datos
      - Seguridad de los datos
      - Escalabilidad

- **Usar restricciones de integridad**

- Acceso y Recuperación de los Datos
      - Llaves Primarias y Foráneas
      - Acciones en Cascada
      - Reglas de Actualización
      - Restricción de Valores de Columnas

- **Normalizar la base de datos**

- Eliminación de Redundancia
      - Reducción de Errores y Ambigüedad
      - Simplificación de Tablas
      - Flexibilidad y Escalabilidad
      - Aumentar la Capacidad de Análisis

- **Mantener segura la base de datos**

- Autenticacion y Autorizacion
    - Protección de Datos Confidenciales
    - Registro de Acciones
    - Restricción de Acceso
    - Copias de Seguridad
    - Cumplimiento de Regulaciones
  - **Implementar una API**
    - Acceso a Datos y Funcionalidad
    - Interfaz Estandarizada
    - Seguridad
    - Pruebas Y Monitoreo
    - Documentacion Clara
  - **Uso de datos atómicos**
    - Integridad de los Datos
    - Unicidad de los Datos
    - Accesibilidad de los datos
    - Escalabilidad de los Datos
    - Compatibilidad de los Datos
    - Identificación de los Datos
  - **Implementación de tutorial**
    - Presentación de los Controles y la Historia
    - Interactividad con el Jugador
    - Progresión del Tutorial
    - Retroalimentación al Jugador
    - Duración corta
  - **Controles intuitivos**
    - Facilidad de Uso
    - Responsividad Instantánea
    - Accesibilidad Amplia
  - **Inicio de sesión**
    - Autenticación
    - Verificación de la Cuenta
    - Recordar la sesión
    - Restablecer contraseña
    - Accesibilidad
- Product Owner Gilberto Echeverria
- **Juego con objetivo**
    - Reglas del juego
    - Niveles
    - Enemigo u Obstáculos
  - **Crecimiento de personaje**
    - Progresión en el juego
    - Sistema de experiencia

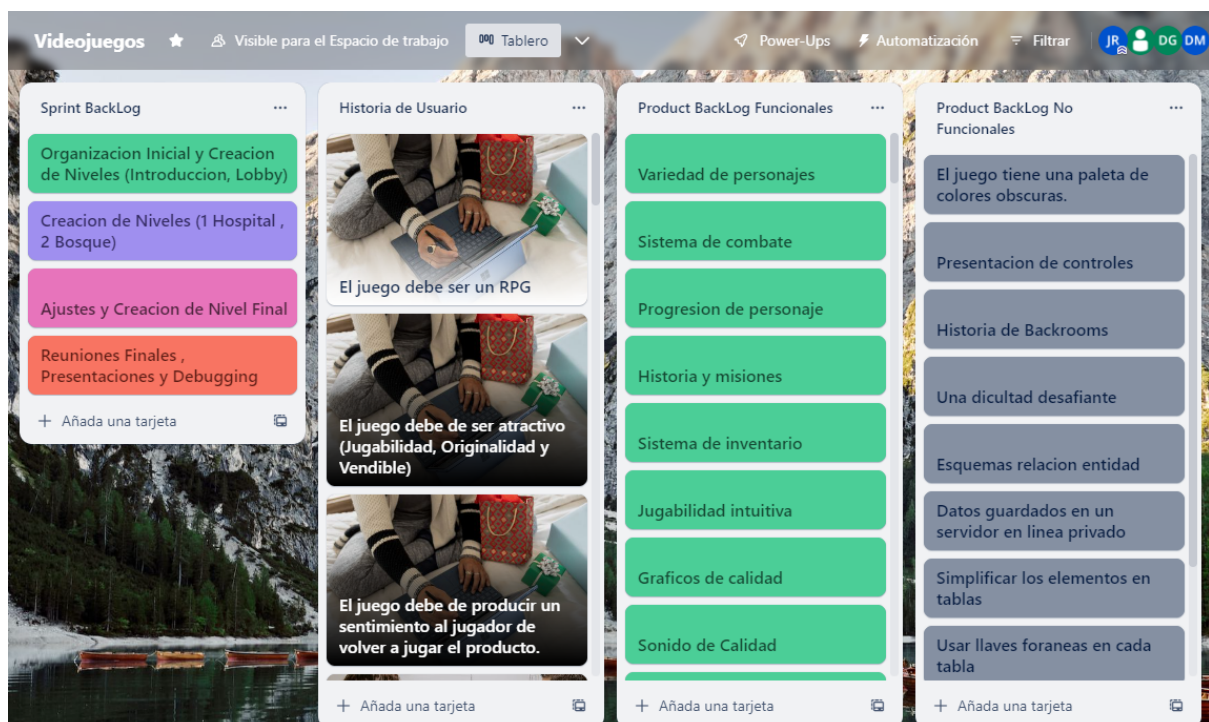
- Crecimiento de habilidad y atributos
  - Mejora de Equipamiento y Objetos
  - Dificultad del juego
  - **Máquinas de estados para decisiones**
    - Estados y Transiciones
    - Eventos
    - Acciones y Condiciones
    - Tolerancia a Fallos
- Product Owner Octavio Navarro
  - **Seguridad de la web**
    - Autenticación y Autorización
    - Encriptación de Datos
    - Prevención de Ataques
    - Monitoreo de Actividades
    - Actualización y Parches de Seguridad
  - **Interfaz de usuario**
    - Diseño Amigable e Intuitivo
    - Navegación Clara
    - Compatibilidad de los Dispositivos
    - Seguridad
  - **Efectos especiales**
    - Variedad de Efectos
    - Calidad de Efectos
    - Coherentes
    - Sincronizados
  - **Juego desafiante**
    - Dificultad escalable
    - Puntos de Control
    - Habilidades Especiales
    - Inteligencia Artificial Desafiante
  - **Función de registro**
    - Información de la Cuenta del Jugador
    - Creación de Perfil o Cuenta
    - Estadísticas y Clasificación
    - Características Adicionales
- **No Funcionales**
  - Paleta de Colores Oscura
  - Presentación Amigable de Controles
  - Controles del Teclado
  - Historia de los Backrooms
  - Una Sola Dificultad

- Diagramas y Esquemas
- Datos guardados en un Servidor en Línea
- Usar Claves Foráneas en cada Tabla
- Música y Ambiente Tétricos
- Pixel Art 16x16

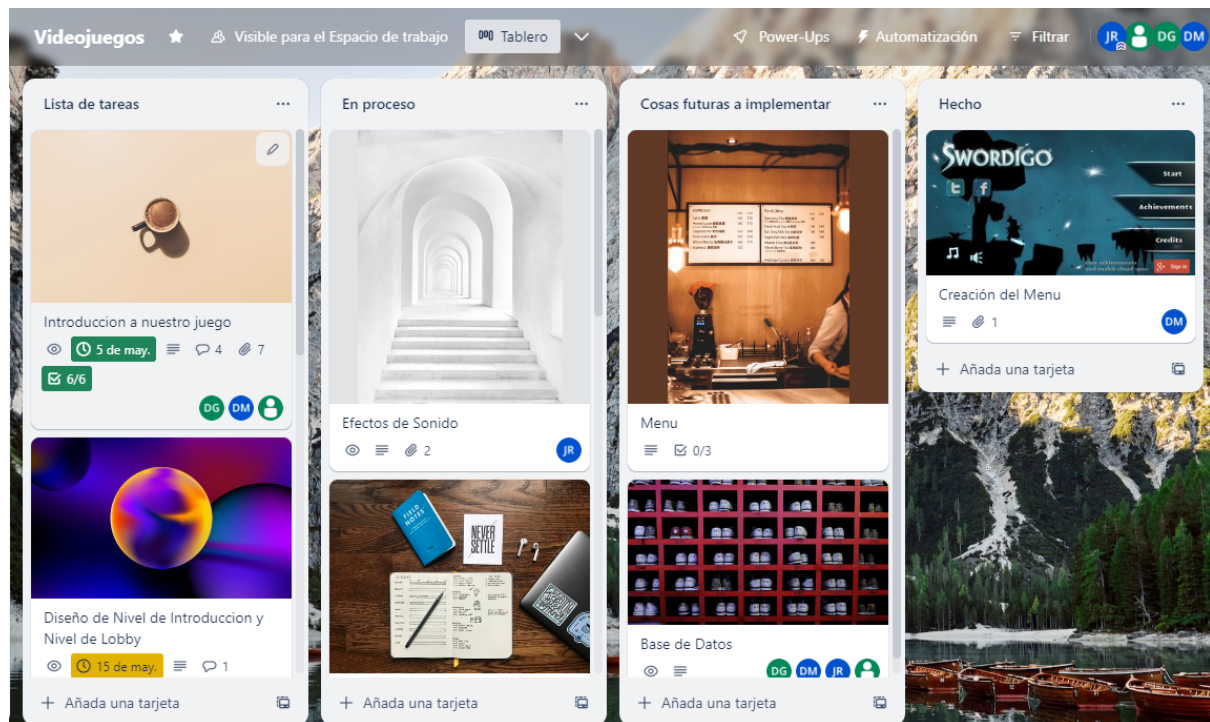
## Sprint Backlog:

El Sprint Backlog es una lista de tareas o actividades específicas que el equipo de desarrollo debe completar durante un Sprint en particular. El Sprint Backlog se crea a partir del Product Backlog y se utiliza para planificar el trabajo que se realizará durante el Sprint. El Sprint Backlog se actualiza diariamente durante el Sprint y se utiliza como una herramienta para el seguimiento del progreso del equipo y para asegurar que se cumplan los objetivos del Sprint.

- **Sprint: Organización Inicial y Creación de Niveles (Introducción, Lobby)**
  - Juan Pablo Cruz Rodríguez
- **Sprint: Creación de Niveles (1 Hospital , 2 Bosque)**
  - David Medina Domínguez
- **Sprint: Ajustes y Creación de Nivel Final**
  - Angel Adrian Alfonso
- **Sprint: Reuniones Finales , Presentaciones y Debugging**
  - Daniel Soult Gomez





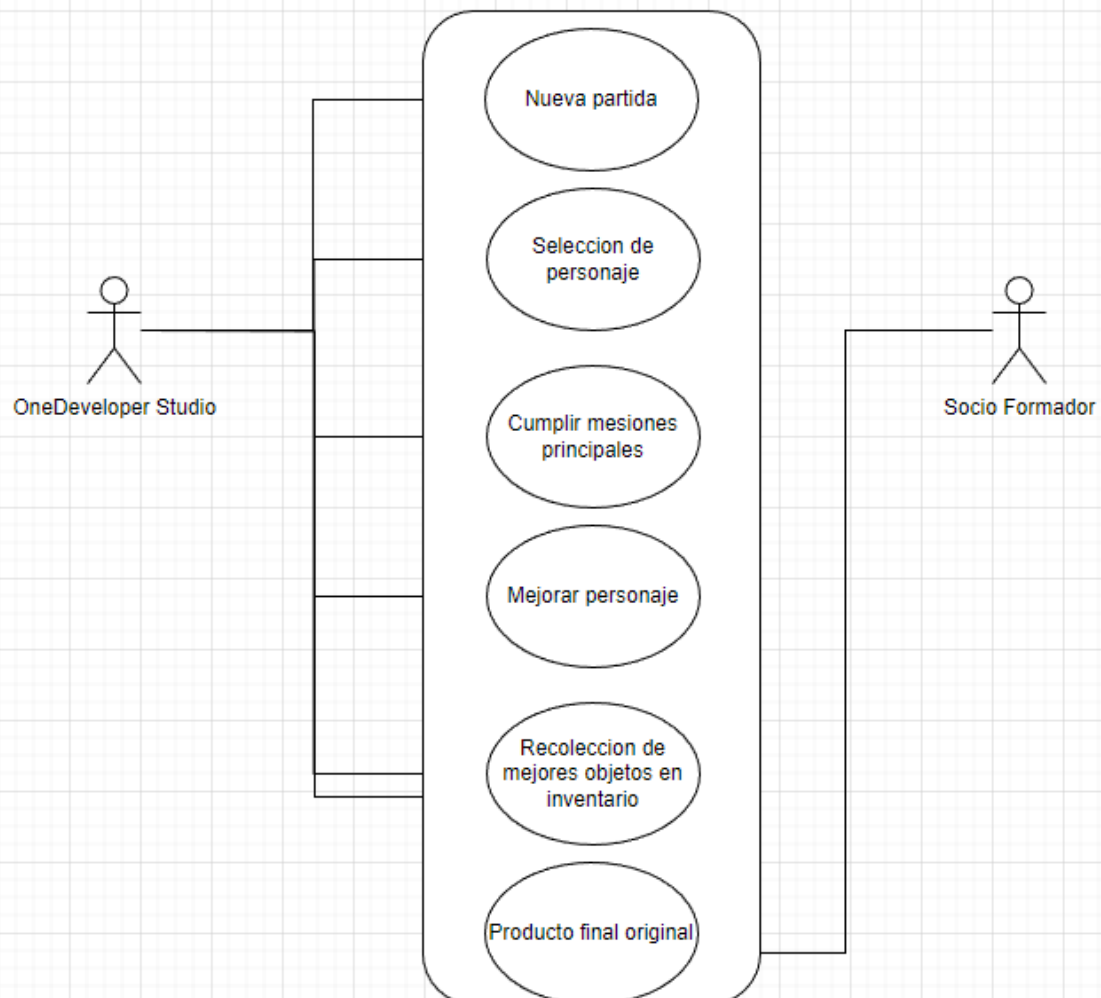


## [Trello: Espacio de Metodología Ágil SCRUM](#)

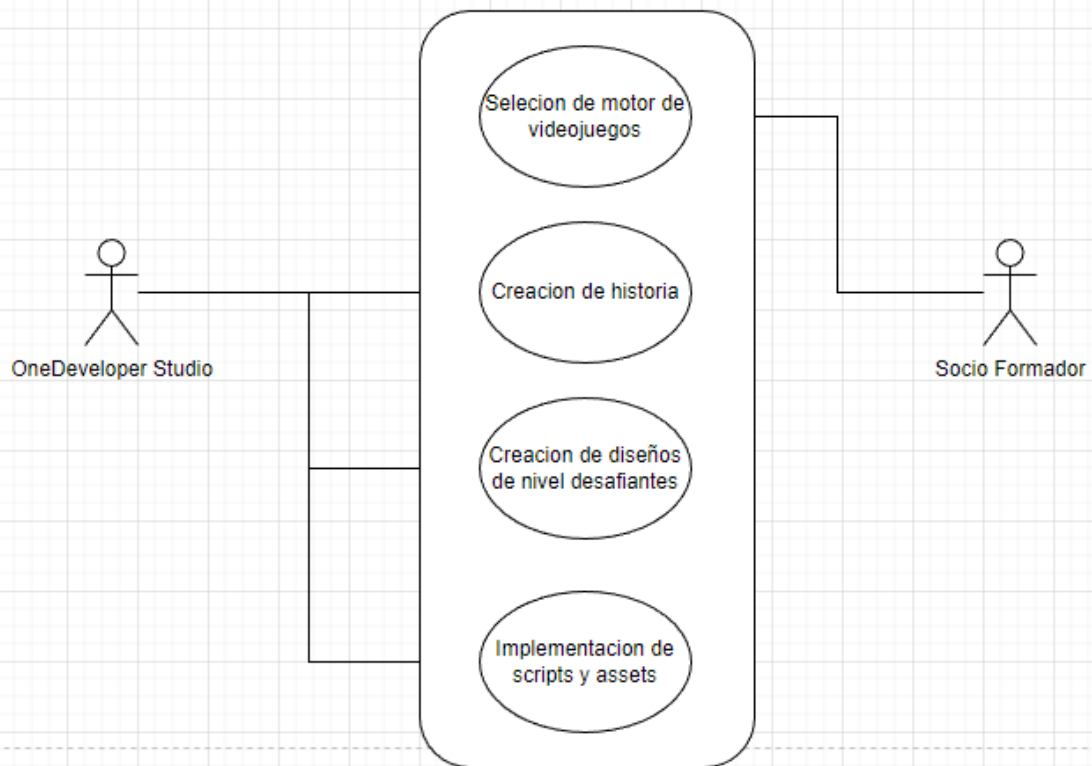
### Diagramas de Casos de Uso:

Los diagramas de caso de uso son una herramienta gráfica utilizada en la ingeniería de software para representar las funcionalidades del sistema desde la perspectiva del usuario. Se utilizan para describir cómo los usuarios interactúan con el sistema y para identificar los requisitos funcionales del sistema.

## Creacion de un RPG

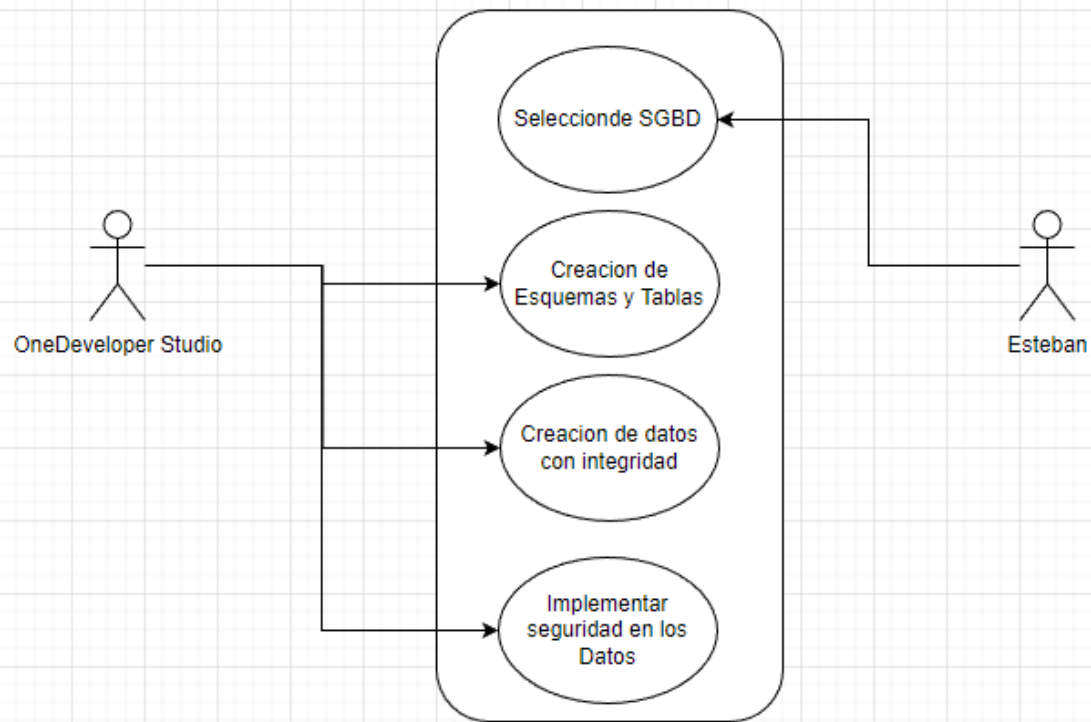


## Creacion de un Juego Atractivo

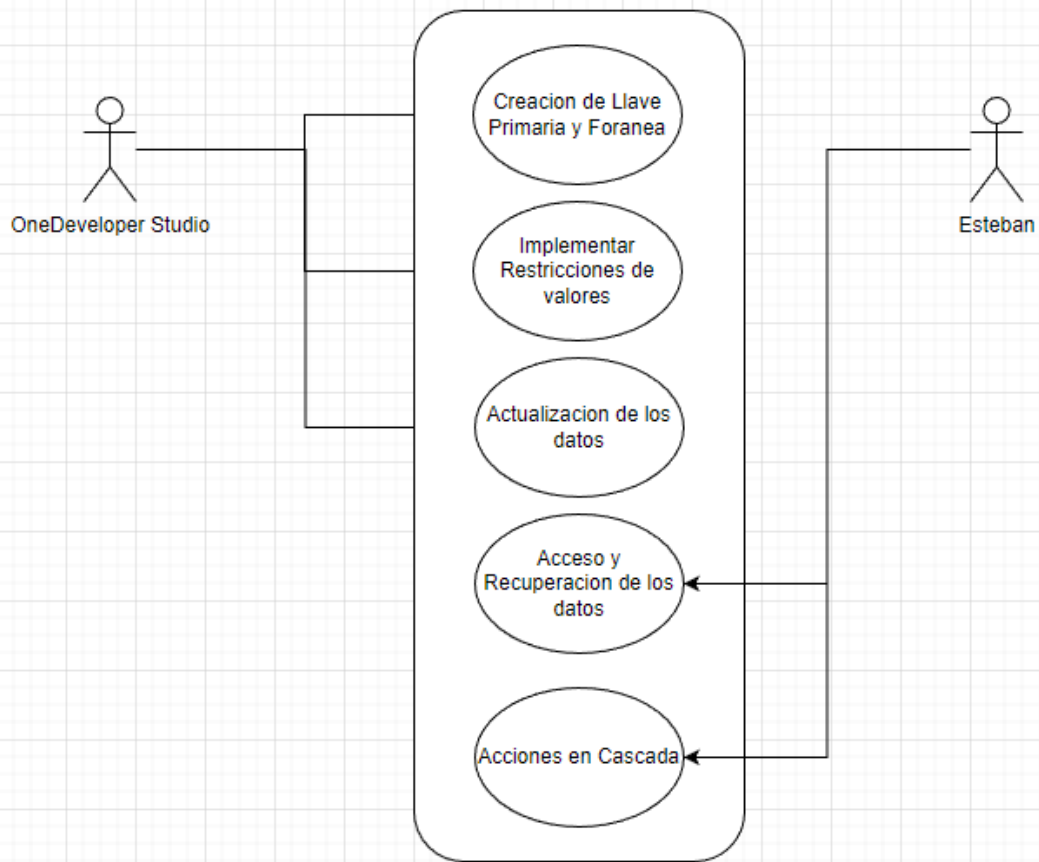


Base de datos relacional

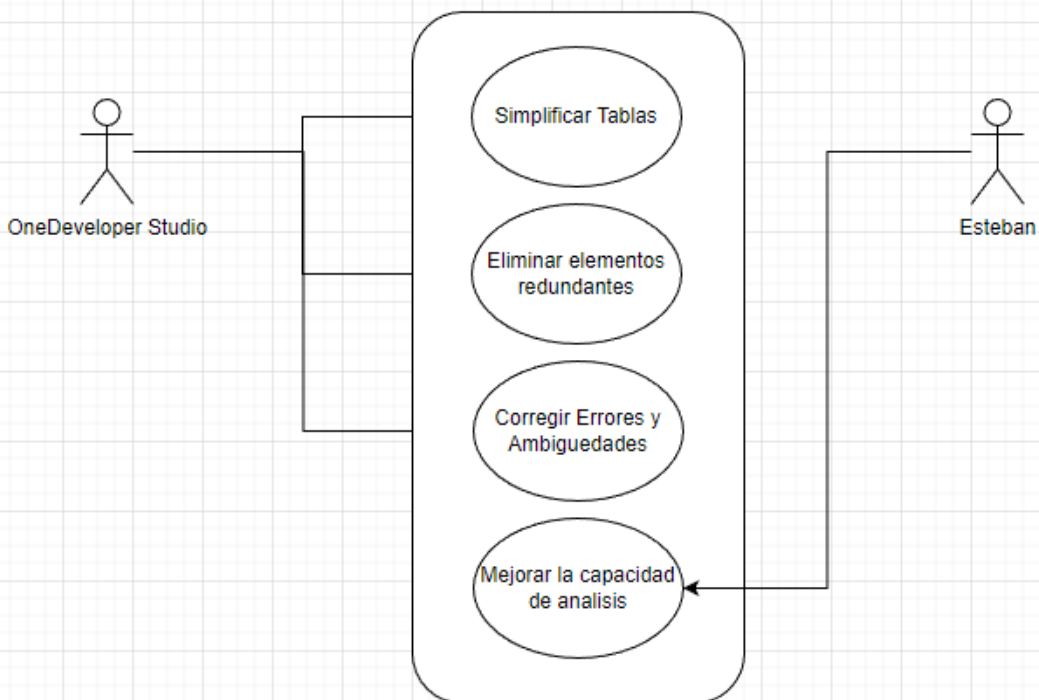
# Base de datos relacional



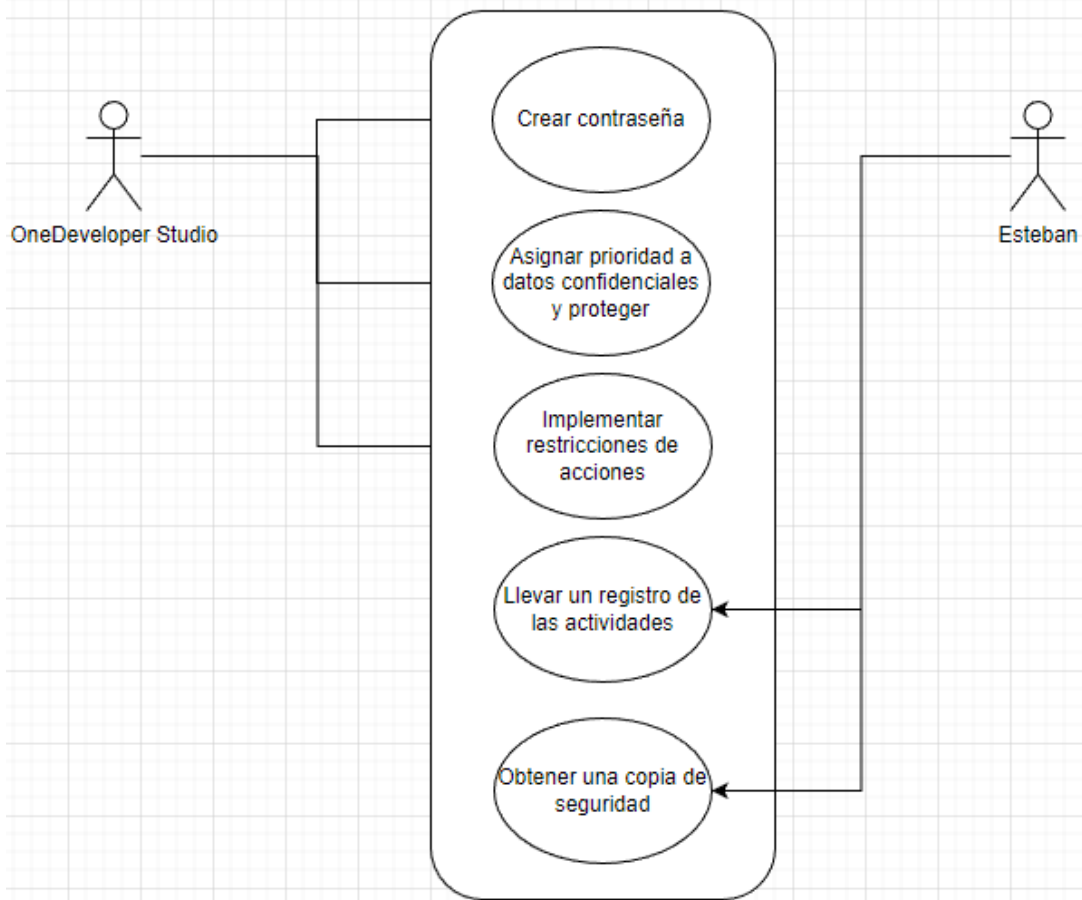
### Implementar Restricciones de Integridad



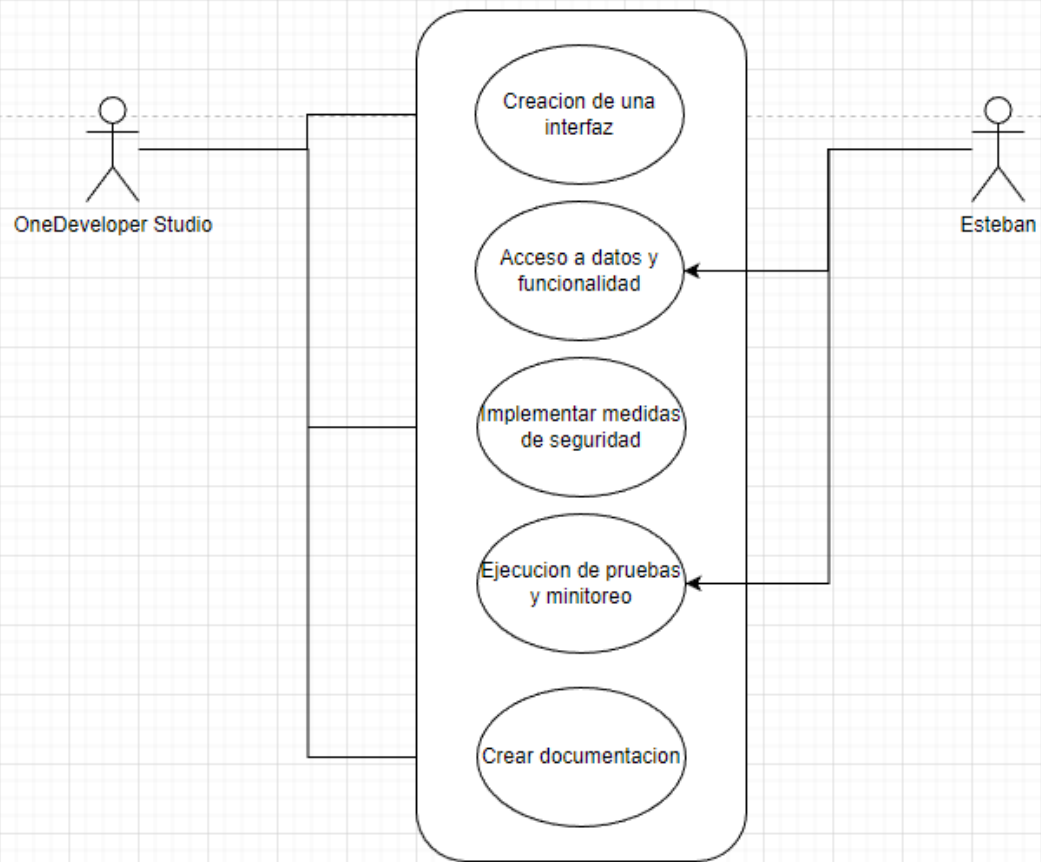
### Normalizacion de la base de datos



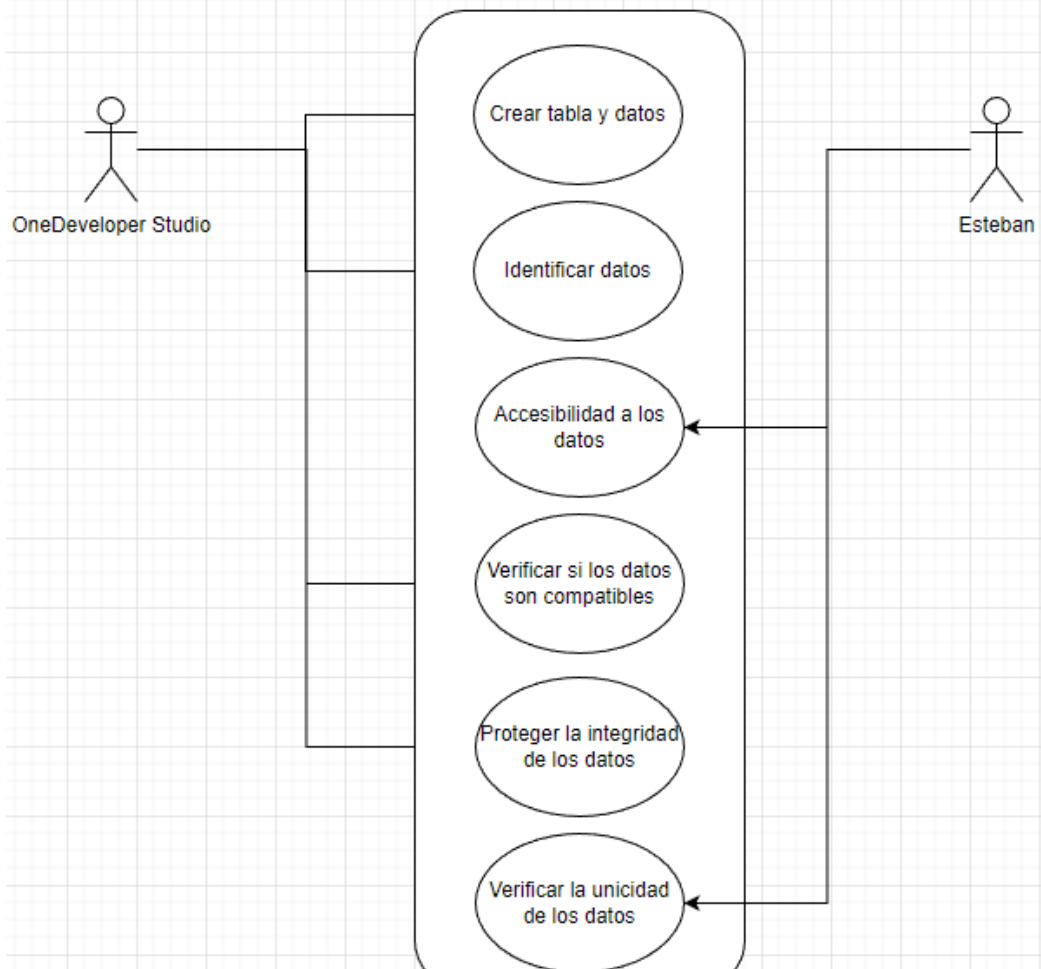
## Seguridad de la base de datos



## Creacion de API

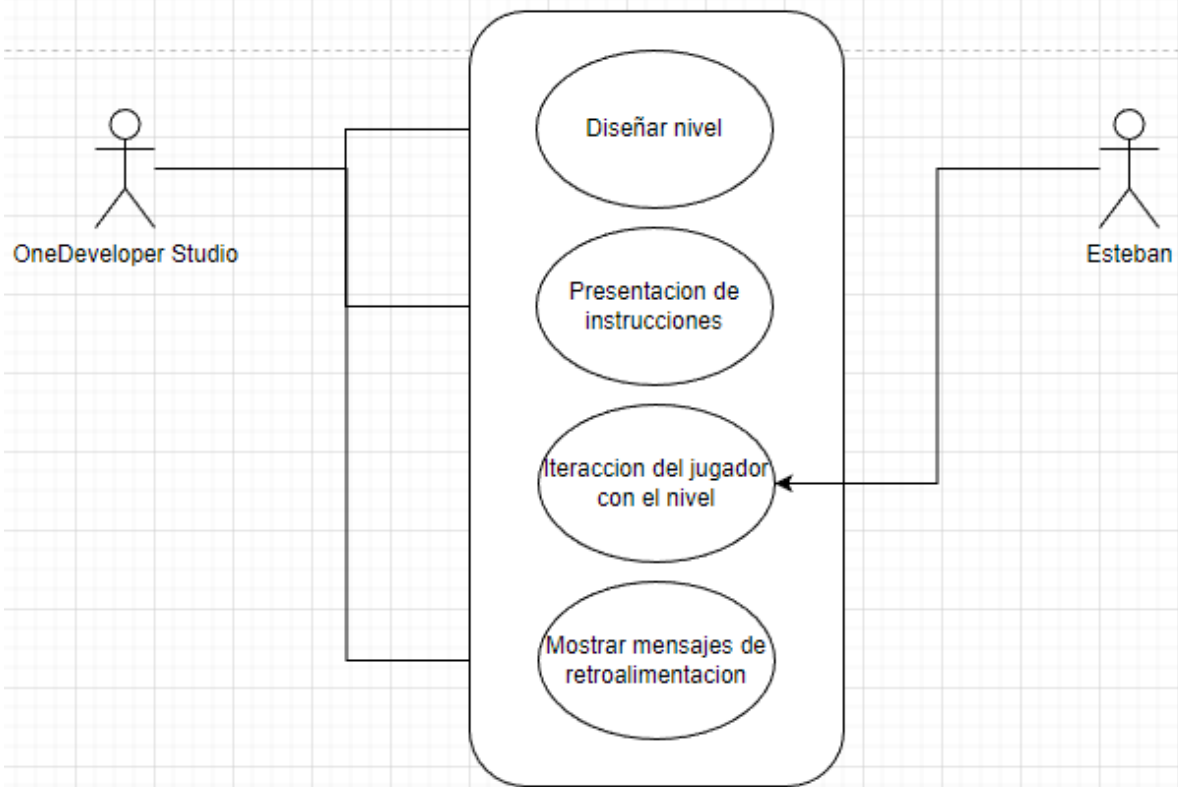


# Creacion de datos atomicos

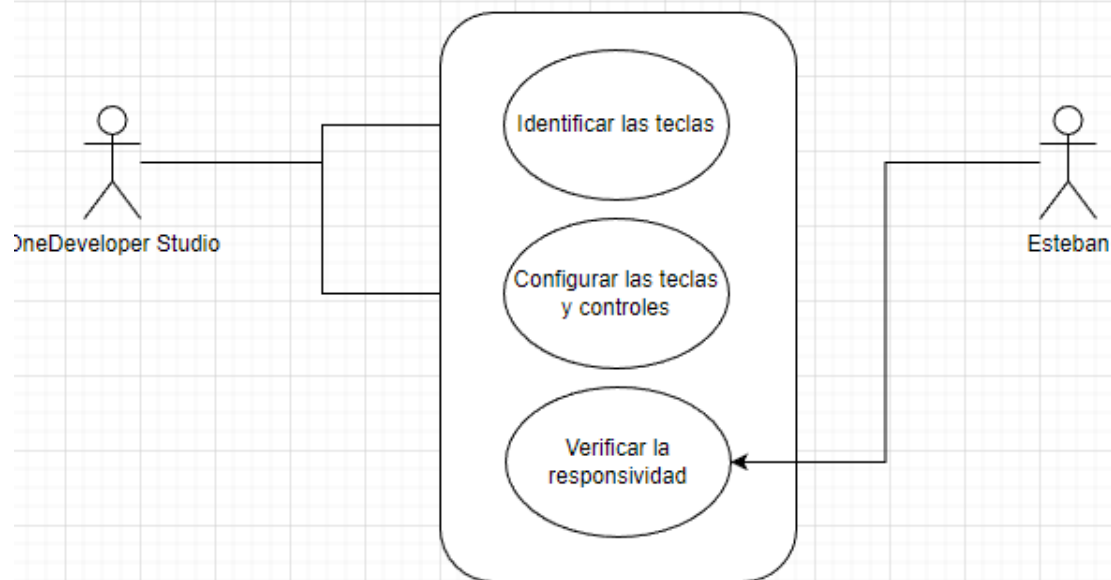


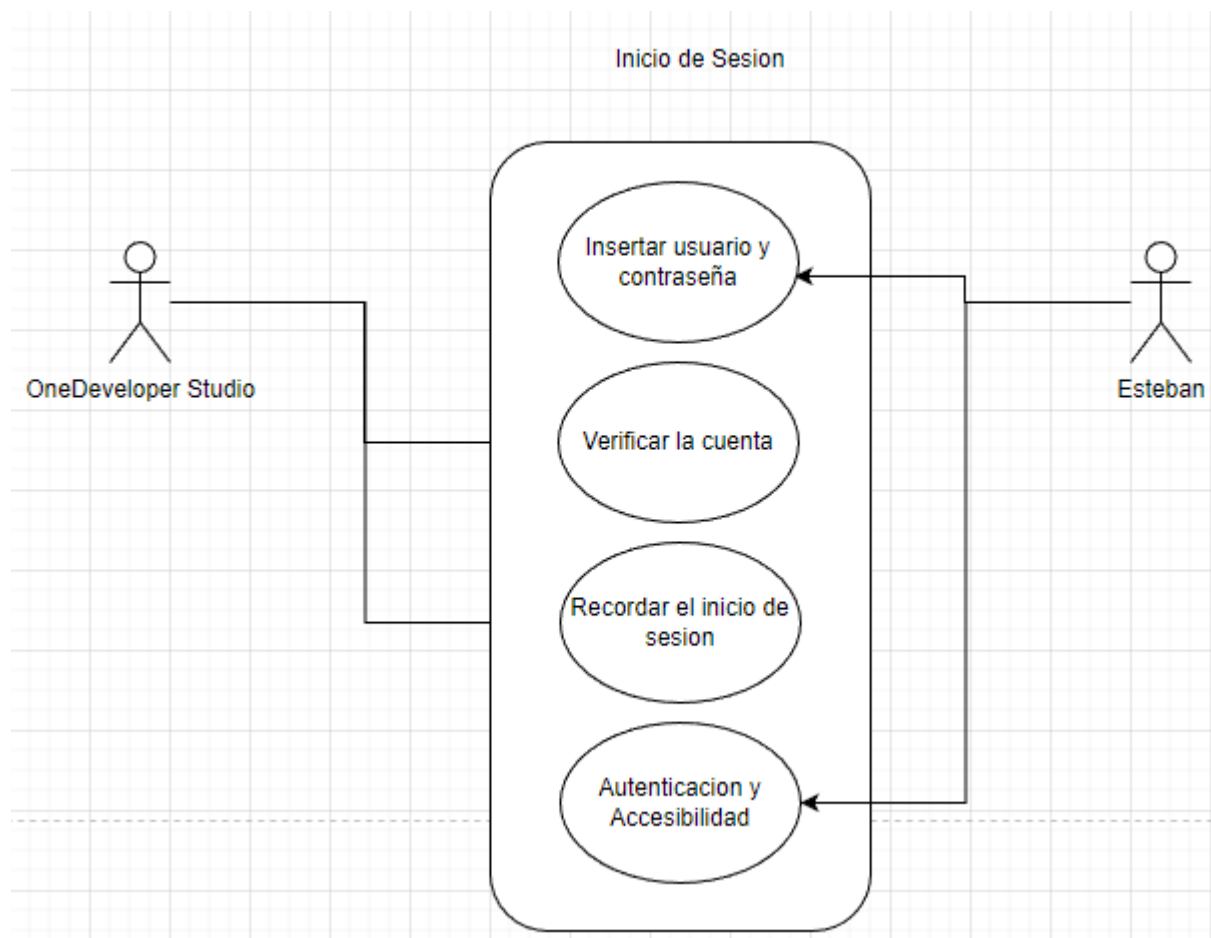


### Creacion de Tutorial



### Implementacion de Controles Intuitivos





## Tablas Descriptivas:

Las tablas descriptivas, también conocidas como tablas de casos de uso o tablas de especificación, son herramientas utilizadas en la ingeniería de software para describir de manera estructurada y detallada los casos de uso de un sistema o software. Estas tablas proporcionan información clara y concisa sobre los actores, los disparadores del caso, los flujos de ejecución y las postcondiciones de cada caso de uso.

<b>Caso de Uso</b>	Inicio de Nueva Partida
<b>Descripción</b>	Este caso de uso describe el inicio de una nueva partida en el videojuego. El jugador comienza una sesión de juego desde cero.
<b>Actores</b>	Jugador
<b>Disparador</b>	El jugador decide iniciar una nueva partida.

<b>Precondiciones</b>	El juego está en modo de inicio y el jugador ha seleccionado la opción de comenzar una nueva partida.
<b>Postcondiciones</b>	Se inicia una nueva partida y se carga el estado inicial del juego.
<b>Flujo de Ejecución:</b>	
1.	El jugador selecciona la opción "Nueva Partida" en el menú principal del juego.
2.	El juego muestra una pantalla de confirmación para asegurarse de que el jugador desea comenzar una nueva partida.
3.	El jugador confirma su decisión y el juego procede a reiniciar todos los elementos necesarios para una nueva partida.
4.	El juego carga el estado inicial, incluyendo la ubicación del jugador, su inventario vacío y cualquier configuración predeterminada necesaria.
5.	El jugador es presentado con una introducción o una secuencia inicial para establecer el contexto de la historia, si corresponde.
6.	El jugador obtiene el control del personaje y puede comenzar a jugar en la nueva partida.
7.	El caso de uso finaliza.

<b>Caso de Uso</b>	Selección de Personaje
<b>Descripción</b>	Este caso de uso describe el proceso de selección de personaje al inicio de un videojuego. El jugador elige al personaje que controlará durante el juego.
<b>Actores</b>	Jugador

<b>Disparador</b>	El jugador decide iniciar una nueva partida.
<b>Precondiciones</b>	El juego está en modo de selección de personaje y se han cargado los personajes disponibles..
<b>Postcondiciones</b>	El jugador selecciona un personaje y pasa al siguiente paso del juego.
<b>Flujo de Ejecución:</b>	
1.	El jugador accede a la pantalla de inicio o menú principal del juego.
2.	El juego muestra la opción "Nuevo Juego" o "Iniciar Partida"..
3.	El jugador selecciona la opción "Nuevo Juego" o "Iniciar Partida".
4.	El juego muestra la pantalla de selección de personaje con una lista de personajes disponibles..
5.	El jugador examina las opciones de personaje y sus características.
6.	El jugador elige un personaje seleccionándolo en la pantalla..
7.	El juego confirma la selección del personaje y muestra un mensaje de confirmación.
8.	El jugador acepta la confirmación y el juego guarda la selección del personaje.
9.	El juego carga el escenario o nivel inicial del juego con el personaje seleccionado.
10.	El jugador obtiene el control del personaje y puede comenzar a jugar.
11.	El caso de uso finaliza.

<b>Caso de Uso</b>	Misión Principal
--------------------	------------------

<b>Descripción</b>	Este caso de uso describe el inicio de una misión principal en un videojuego. El jugador se embarca en una tarea específica como parte de la historia principal del juego.
<b>Actores</b>	Jugador, Personajes No Jugadores (PNJ) relacionados con la misión.
<b>Disparador</b>	El jugador alcanza un punto en la historia del juego que activa una nueva misión principal..
<b>Precondiciones</b>	El juego está en modo de misión principal y se ha cargado la misión correspondiente con sus objetivos y personajes involucrados.
<b>Postcondiciones</b>	El jugador comienza la misión principal y se establece el estado de la misión.
<b>Flujo de Ejecución:</b>	
1.	El jugador se encuentra en un área del juego o recibe una comunicación que indica el inicio de una nueva misión principal.
2.	El juego muestra una notificación o mensaje en pantalla que informa al jugador sobre la nueva misión y sus objetivos.
3.	El jugador revisa los detalles de la misión, incluyendo los objetivos, el contexto de la misión y las recompensas potenciales.
4.	El jugador acepta iniciar la misión principal y confirma su participación.
5.	El juego actualiza el estado de la misión, marcándose como "activa" y estableciendo los objetivos iniciales.
6.	El jugador recibe indicaciones o pistas sobre la ubicación o acciones requeridas para progresar en la misión principal.
7.	El jugador explora el mundo del juego,

	interactúa con personajes no jugadores (PNJ) relevantes y cumple los objetivos de la misión principal.
8.	Durante la misión, el jugador puede enfrentar desafíos, resolver acertijos, participar en combates u otras actividades relacionadas con la misión.
9.	El jugador progresa en la misión principal al completar los objetivos intermedios y avanzar hacia el objetivo final.
10.	Una vez que el jugador ha cumplido con éxito los requisitos de la misión principal, el juego muestra una notificación de completado y proporciona las recompensas correspondientes..
11.	El juego actualiza el estado de la misión a "completada" y permite
12.	El caso de uso finaliza.

<b>Caso de Uso</b>	Mejorar al Personaje
<b>Descripción</b>	Este caso de uso describe el proceso de mejora del personaje del jugador en un videojuego. El jugador realiza acciones específicas para aumentar las habilidades, características o niveles del personaje.
<b>Actores</b>	Jugador
<b>Disparador</b>	El jugador decide mejorar al personaje después de alcanzar un punto específico en el juego, acumular experiencia, obtener nuevos recursos o cumplir ciertos requisitos.
<b>Precondiciones</b>	El jugador decide mejorar al personaje después de alcanzar un punto específico en el juego, acumular experiencia, obtener nuevos recursos o cumplir ciertos requisitos..

<b>Postcondiciones</b>	El personaje del jugador se mejora según las acciones realizadas, y se actualizan las estadísticas, habilidades o características correspondientes.
<b>Flujo de Ejecución:</b>	
1.	El jugador accede a una pantalla o menú específico para mejorar al personaje.
2.	El juego muestra las opciones y recursos disponibles para la mejora, como puntos de experiencia, habilidades desbloqueables, objetos especiales, etc.
3.	El jugador revisa las opciones y decide cómo desea mejorar al personaje.
4.	El jugador selecciona una opción de mejora, como asignar puntos de experiencia a características específicas, desbloquear una nueva habilidad o equipar un objeto mejorado.
5.	El juego verifica si el jugador cumple los requisitos necesarios para realizar la mejora seleccionada.
6.	Si los requisitos se cumplen, el juego realiza los cambios correspondientes en las estadísticas, habilidades o características del personaje.
7.	El juego muestra una confirmación de la mejora realizada y actualiza la interfaz del jugador para reflejar los cambios..
8.	El jugador puede repetir el proceso de mejora seleccionando nuevas opciones o recursos disponibles.
9.	El jugador puede guardar los cambios realizados en la mejora del personaje.
10.	El jugador puede continuar jugando con el personaje mejorado y aprovechar las ventajas o beneficios obtenidos.

11.	El caso de uso finaliza.
-----	--------------------------

<b>Caso de Uso</b>	Recolección de Objetos
<b>Descripción</b>	Este caso de uso describe el proceso de recolección de objetos por parte del jugador en un videojuego. El jugador busca, recoge y almacena diferentes tipos de objetos durante su progreso en el juego.
<b>Actores</b>	Jugador
<b>Disparador</b>	El jugador encuentra un objeto visible en el entorno del juego o interactúa con un elemento específico que contiene objetos recolectables..
<b>Precondiciones</b>	El juego está en modo de exploración o interacción, y existen objetos recolectables en el entorno del jugador.
<b>Postcondiciones</b>	El jugador recolecta el objeto, que se agrega a su inventario o se utiliza de alguna manera en el juego.
<b>Flujo de Ejecución:</b>	
1.	El jugador explora el entorno del juego o interactúa con elementos específicos, como cofres, cajas, enemigos derrotados, etc.
2.	El juego muestra un objeto recolectable en el entorno o en la recompensa obtenida al interactuar con el elemento.
3.	El jugador decide recolectar el objeto y realiza la acción correspondiente, como acercarse al objeto o interactuar con él.
4.	El juego verifica si el jugador cumple los requisitos necesarios para recolectar el objeto.
5.	Si los requisitos se cumplen, el juego realiza los cambios necesarios en el inventario del jugador y agrega el objeto recolectado.



6.	El juego muestra una confirmación de la recolección del objeto y actualiza la interfaz del jugador para reflejar los cambios en el inventario.
7.	El jugador puede repetir el proceso de recolección al encontrar nuevos objetos en el entorno del juego o al interactuar con elementos adicionales.
8.	El jugador puede utilizar los objetos recolectados de diversas formas, como equiparlos, utilizarlos como herramientas o recursos, combinarlos con otros objetos, etc.
9.	El jugador puede gestionar su inventario, descartando objetos no deseados, vendiéndolos o intercambiándose con otros personajes no jugadores (PNJ) en el juego.
10.	El caso de uso finaliza.

<b>Caso de Uso</b>	Entrega Final
<b>Descripción</b>	Este caso de uso describe el proceso de entrega final de un videojuego. El equipo de desarrollo completa y entrega el producto final del videojuego a los usuarios o clientes.
<b>Actores</b>	Equipo de Desarrollo, Usuarios o Clientes
<b>Disparador</b>	El desarrollo del videojuego ha llegado a su etapa final y está listo para ser lanzado o entregado a los usuarios o clientes.
<b>Precondiciones</b>	El desarrollo del videojuego ha alcanzado su estado final, todas las características y funcionalidades han sido implementadas y se han realizado las pruebas y correcciones necesarias.
<b>Postcondiciones</b>	El videojuego finalizado se entrega a los usuarios o clientes para su uso o distribución.

Flujo de Ejecución:	
1.	El equipo de desarrollo realiza las últimas pruebas y correcciones para garantizar que el videojuego esté libre de errores y cumpla con los requisitos establecidos.
2.	El equipo de desarrollo prepara los elementos necesarios para la entrega final, como la compilación final del videojuego, los recursos adicionales, los documentos de usuario y cualquier otro componente requerido.
3.	El equipo de desarrollo crea una versión final del videojuego que se ajusta a los estándares y formatos requeridos para su distribución o uso.
4.	El equipo de desarrollo genera la documentación necesaria, como manuales de usuario, guías de instalación, notas de lanzamiento, etc.
5.	El equipo de desarrollo realiza una revisión final de todos los elementos y documentos para asegurarse de que estén completos y listos para su entrega.
6.	El videojuego finalizado y los documentos relacionados se entregan a los usuarios o clientes según los acuerdos establecidos, ya sea a través de descarga digital, distribución física o mediante una plataforma específica.
7.	Los usuarios o clientes reciben el videojuego finalizado y los documentos asociados, y procede a instalarlo, utilizarlo o distribuirlo según sea necesario.
8.	El equipo de desarrollo puede realizar un seguimiento posterior a la entrega para recopilar comentarios, realizar actualizaciones o proporcionar soporte adicional según sea necesario.
9.	El caso de uso finaliza.

