



|   |   |
|---|---|
| Logo del grupo:   |   |
| Nombre del grupo:   | RPG UMA studios   |
| Titulación:   | Ingeniería Informática  |
| Grupo de la titulación (A/B/C/D/...):   | A   |
| Grupo reducido (GR1/GR2/...):   | GR2   |
| Número de grupo, dentro del grupo reducido (Por ejemplo, si tu grupo es el "GR1-03", tendrías que escribir "03"): | 02  |
| Repositorio Github (url):   | <a href="https://github.com/JuanGonUMA/Proyecto-IIS">https://github.com/JuanGonUMA/Proyecto-IIS</a>   |
| Espacio Trello (url):   | <a href="https://trello.com/invite/b/1IRL1cra/ATTI7a3bdf3abf0f74fdc63f116521b8a63cE6B4EDE2/rpg-uma-studios">https://trello.com/invite/b/1IRL1cra/ATTI7a3bdf3abf0f74fdc63f116521b8a63cE6B4EDE2/rpg-uma-studios</a> |
| Memoria del proyecto (Office 365 o Google Docs url):  | <a href="https://docs.google.com/document/d/1fovFAouLURRqUzD0CcRxI9wtmG2IF--xqf4KM0dquMo/edit?usp=sharing">https://docs.google.com/document/d/1fovFAouLURRqUzD0CcRxI9wtmG2IF--xqf4KM0dquMo/edit?usp=sharing</a>   |

| Número de miembro | Nombre  | Apellidos          | Correo electrónico UMA |
|-------------------|---------|--------------------|------------------------|
| 1                 | Steven  | Rodríguez Santos   | stevenrs191@uma.es     |
| 2                 | Tiago   | Fernández Hidalgo  | tiagoferh@uma.es       |
| 3                 | Juan    | González Sierra    | 0611137347@uma.es      |
| 4                 | Enrique | Moragues Picatoste | kiikemp10@uma.es       |
| 5                 | Jaime   | Canalejo Rodriguez | canalejojaime@uma.es   |



## **Tabla de contenidos:**

Sección 1: Introducción.

Sección 2: Roles.

Sección 3: Gestión del riesgo.

Sección 4: Planificación.

Sección 5: Requisitos.

Sección 6: Casos de uso.

Sección 7: Modelo de dominio.

Sección 8: Diagramas de secuencia.

Sección 9: Herramientas software usadas durante la realización del proyecto.



## Sección 1:

Nuestro proyecto consiste en desarrollar un rpg (role playing game) ambientado en la época de los piratas (1620-1683), donde deberás superar una serie de desafíos relacionados para así vencer el juego.

El problema que intentamos satisfacer es algo más complicado de definir para nuestro proyecto ya que este es un videojuego, pero principalmente podríamos decir que lo que busca es combatir el aburrimiento, ofrecer a los estudiantes un juego que les permita disfrutar del mismo.



## Sección 2:

Los roles que hemos definido y los integrantes del grupo que lo asumen son los siguientes:

1. Coordinador. Garantiza el cumplimiento de plazos, soluciona los errores que surjan en GitHub, gestiona los incidentes y supervisa que el proyecto se desarrolle como se espera. Además, es encargado de las comunicaciones con el cliente y se asegura de reunir la información relevante que este pueda proporcionar. Encargados: Jaime y Kike.
2. Diseñador. Define conceptos clave como los niveles, las dimensiones de las habitaciones, las entidades presentes, etc. Su labor incluye establecer la estructura principal del juego. Encargados: Tiago, Juan y Steven.
3. Programador. Implementa el proyecto mediante la escritura de su código con todo lo que ello conlleva: manejar la lógica del juego, crear las estructuras de datos, corregir errores... Encargados: Juan, Kike y Steven.
4. Tester. Verifica que el programa funciona como se esperaba, probando todas las funciones y características del juego, identificando posibles errores, fallos de rendimiento, comportamientos inesperados... Su retroalimentación es clave para los otros roles. Encargados: Tiago y Jaime.



## Sección 3:

La gestión de riesgos viene dada por la siguiente tabla:

| Tipo de Riesgo | Descripción  | Probabilidad | Efectos del Riesgo | Estrategia para Mitigar   |
|----------------|--|--------------|--------------------|---|
| Proyecto       | Desviaciones en los plazos de entrega  | Baja         | Catastrófico       | Realizaremos una planificación realista y bien gestionada que considere pequeños contratiempos y reparta el tiempo de la forma más eficiente posible  |
| Negocio        | Ausencia por viaje o accidente   | Baja         | Serio              | Si sucede, tendremos que modificar la planificación y reasignar los trabajos de la persona que va a ausentarse  |
| Proyecto       | Conflicto en el guardado en Github   | Media        | Leve               | Resolver conflicto en local y después actualizarlo en repositorio   |
| Proyecto       | Algún archivo, ya sea por error de software o humano, sea eliminado                              | Muy baja     | Catastrófico       | Tener diferentes copias del archivo, tanto localmente como en el repositorio de Github  |
| Negocio        | Falta de comunicación entre dos o más partes del equipo  | Baja         | Catastrófico       | Una buena planificación y chequeos constantes debería de mitigar este problema  |
| Producto       | El proyecto es mucho más grande de lo que podemos abordar y por ende necesitamos recortar código | Baja         | Serio              | Igual que la anterior, una buena planificación del proyecto debería de mitigar este problema  |
| Proyecto       | Algún profesor decide que no quiere ser incluido como personaje en el proyecto                   | Baja         | Leve               | Escogeremos a aquellos profesores que consideremos que no cambiarán de opinión y se presten a este tipo de peticiones, en caso de que igualmente no quieran participar, bastará con sustituir su personaje por otro NPC diferente |



## Sección 4:

Hemos optado por una metodología fuertemente inspirada en la Scrum, a la cual hemos decidido denominar "Light Scrum". Consiste en planificar una serie de tareas que deberán ser cumplidas en un determinado plazo, así, en lugar de reunirnos cada día, podemos avisar de manera asíncrona cuando una tarea ha sido completada y continuar planificando nuevas tareas. Esto nos permite adaptarnos y amoldar el proyecto según nuestras necesidades temporales.

Una de las razones por las que nos hemos decidido por este estilo de trabajo y no por la Scrum tradicional es la dificultad de compenetrar los diferentes horarios de trabajo de cada uno, además, intentar realizar una reunión diaria podría ser ineficiente en nuestro caso ya que no vamos a trabajar cada día, por ello es mucho mejor tomar estas reuniones de forma asíncrona y cuando consideremos conveniente pudiendo además aligerar la carga de trabajo en situaciones donde tengamos tareas más prioritarias provenientes de otras asignaturas o, por el contrario, centrarnos más en el proyecto en momentos más tranquilos.

No hemos utilizado power-ups de trello todavía, ya que no nos hemos visto en la necesidad de utilizarlo aunque no descartamos la posibilidad de usarlos en un futuro.

Las tareas que existen actualmente en el Trello son:

1. Tareas relacionadas a las entregas: estas tareas sirven como recordatorio de las diferentes entregas del Campus Virtual, teniendo asignadas las fechas de entrega para que no se nos pase ninguna.
2. Decidir software para el proyecto: esta tarea solo sirve para decidir el software con el que implementaremos el código del proyecto final.
3. Organizar los requisitos: una tarea para hacer el reparto de tareas relacionadas a la planificación.
4. Realizar los casos de uso: una tarea para indicar los casos de uso que cada miembro realizará, para no solaparse, los cuales se detallarán en la memoria (este mismo documento).
5. Planificar el diseño del juego: esta tarea sirve para planificar y repartir las tareas relacionadas a la implementación del código del proyecto final.
6. Testear el juego: esta tarea sirve para comenzar a planificar la implementación de las pruebas JUnit del proyecto final.
7. Recursos: tenemos unas "tareas" al final que sirven como fuente de recursos para el resto de tareas, sobre todo, para la implementación del código, con documentación útil.



**RPG UMA studios** ☆ 🔍 Tablero

Filtros TH EP JR JS & Compartir ...

| Entregas Pendientes                   | Hecho  | En progreso                      | En espera                             | Próximo             |
|---------------------------------------|--|----------------------------------|---------------------------------------|---------------------|
| Planificación<br>🕒 22 de mar.         | Decidir software para el proyecto<br>🕒 22 de mar. 🗨️ 1 | Planificar el diseño del juego   | Planificar la programación del código | Testear el juego    |
| Requisitos Proyecto<br>🕒 5 de abr.    | + Añada una tarjeta                                    | Diseño de mundo                  | + Añada una tarjeta                   | + Añada una tarjeta |
| Casos de Uso<br>🕒 29 de abr.          |  | Diseño de personajes             |                                       |                     |
| Modelo de Dominio<br>🕒 6 de may.      |  | Diseño de historia               |                                       |                     |
| Diagrama de Secuencia<br>🕒 20 de may. |  | Entrevistas a profesores<br>🗨️ 2 |                                       |                     |
| Plan de Pruebas<br>🕒 27 de may.       |  | Organizar los requisitos         |                                       |                     |
| Proyecto Completo                     |  | + Añada una tarjeta              |                                       |                     |
| + Añada una tarjeta                   |  |                                  |                                       |                     |

Left Sidebar: Tableros, Miembros, Ajustes del Espacio de trabajo, Vistas del Espacio de trabajo, Tabla, Calendario, Sus tableros (Proyecto IIS, RPG UMA studios)



## Sección 5:

### RF1: Interactuar con objetos del mundo

**Como** diseñador,  
**quiero** que los jugadores puedan interactuar con los objetos del mapa.  
**Para** poder, en esencia, jugar.

#### Pruebas de aceptación

El jugador pulsará la tecla vinculada a interactuar estando en contacto con un objeto, esto desencadenará diferentes eventos.  
El jugador obtendrá efectos diferentes (o ninguno) a los iniciales si trata de interactuar con un objeto con el cual ya lo hizo.

### RF2: Mundo

**Como** diseñador,  
**Quiero** que el jugador tenga su experiencia de juego dentro de un mundo donde suceden todos los eventos, este debe ser amplio con zonas diferenciadas.  
**Para** que los jugadores tengan que explorarlo y descubrir así enemigos, equipamiento, aliados...

### RF3: Combate

**Como** diseñador,  
**quiero** que los jugadores puedan enfrentarse a enemigos con diversas habilidades e historias.  
**Para** que su experiencia de juego sea satisfactoria.

### RF4: “Jefes”

**Como** diseñador,  
**quiero** que determinados enemigos sean más poderosos de lo normal y tengan unas mecánicas determinadas para poder vencerlos.  
**Para** que los jugadores experimenten combates más largos y exigentes en los que deberán esforzarse para salir victoriosos.

#### Pruebas de aceptación

Cuando el jugador se encuentre con un jefe, este supondrá un reto, esto se refleja en que durará más en pie en comparación a otros enemigos y su combate será más difícil e intenso.





### RF5: Diálogos

**Como jugador,**

**quiero** poder tener diferente tipos de diálogos y respuestas a estos

**Para** hacer que el juego sea rejugable y que sienta que puedo tomar mis propias decisiones.

#### Pruebas de aceptación

El jugador pulsa la tecla correspondiente y se abre una caja en la parte inferior de la pantalla sobre la cual aparecerá un texto. Con la tecla enter, más texto irá apareciendo hasta completar el diálogo, momento en el cual la caja con el texto desaparecerá. Durante algunos diálogos es posible que se presenten diferentes opciones para dar como respuestas, si es el caso, el jugador seleccionará la opción deseada y el diálogo continuará.

### RF6: Sistema de progresión

**Como diseñador,**

**quiero** que los jugadores tengan un nivel que pueda aumentar conforme supere diferentes hitos y estos le otorguen beneficios.

**Para** darles una motivación a la hora de jugar y sentir que progresan.

#### Pruebas de aceptación

El jugador obtiene experiencia al superar diferentes hitos, esto se le hace saber al jugador gracias a una barra en el menú principal.

Con suficiente progreso, subirá de nivel, lo que hace que la barra vuelva a 0 pero se incrementa en uno su nivel, esto debería incrementar determinadas características del jugador, como su vida o daño.

Alcanzar ciertos niveles podría provocar algunos cambios, como algunos diálogos alterados, si el jugador obtiene estos niveles, dichos diálogos deberán cambiar.

### RF7: Guardado

**Como jugador,**

**quiero** poder guardar el progreso de juego.

**Para** continuar jugando por donde lo dejé la última vez y no tener que empezar de nuevo.

#### Pruebas de aceptación

Desde el menú correspondiente, el jugador elige la opción de guardar su progreso en la ranura de memoria deseada.

En el menú inicial, el jugador puede seleccionar una partida guardada y seguir jugando desde el mismo punto donde lo dejó.

**RF8: Habilidades**

**Como** jugador,

**quiero** poseer diversas habilidades.

**Para** poder utilizarlas en combate y tanto planear estrategias como adaptar mi estilo de juego a la hora de acabar con los enemigos.

**RF9: Inventario**

**Como** jugador,

**quiero** tener un inventario que incluya diferentes tipos de objetos.

**Para** poder utilizarlos como considere y así prepararme a la hora de enfrentar diferentes desafíos.

**Pruebas de aceptación**

El jugador pulsa la tecla vinculada al inventario y se muestra el inventario.

El jugador puede seleccionar con el cursor objetos y utilizarlos, tirarlos o equiparlos, hacer esto provocará determinados cambios en el inventario.

**RF10: Armas**

**Como** diseñador,

**quiero** que el inventario (RF9) pueda contener un tipo de objeto arma.

**Para** que el jugador pueda escoger la que crea conveniente o encaje con su estilo de juego a la hora de dañar a los enemigos.

**RF11: Armaduras**

**Como** diseñador,

**quiero** que el inventario (RF9) pueda contener un tipo de objeto armadura..

**Para** que el jugador pueda escoger la defensa que considere más conveniente a la hora de enfrentarse a enemigos.

**RF12: Consumibles**

**Como** diseñador,

**quiero** que el inventario (RF9) pueda contener un tipo de objeto consumible.

**Para** que el jugador pueda aprovechar sus efectos de forma limitada y en momentos de mayor relevancia.

**RF13: Diferentes finales**

**Como** diseñador,

**quiero** que el juego tenga diferentes desenlaces según las decisiones que el jugador haya tomado.



**Para** que el juego pueda ofrecer experiencias más personalizadas y permitir que este pueda jugarse de nuevo.

#### **RF14: Puzzles**

**Como** diseñador,

**quiero** que haya diferentes puzzles repartidos por el mapa que escondan tesoros.

**Para** que los jugadores se entetengan resolviéndolos y obtengan diferentes recompensas por resolverlos.

#### **RF15: Npc's**

**Como** diseñador,

**Quiero** que los jugadores tengan a su disposición diferentes npcs (non playable characters) con los cuales pueda interactuar y obtener diálogos, misiones, combates...

**Para** que sea más inmersivo jugar.

#### **RF16: Clases**

**Como** jugador,

**quiero** ser capaz de elegir una clase que se enfoque en un estilo de lucha y tenga habilidades concretas.

**Para** poder elegir el estilo de combate que más satisfactorio y sencillo me parezca.

#### **RF17: Personalización de personaje**

**Como** jugador,

**Quiero** personalizar mi personaje a mi gusto, por ejemplo el tipo de pelo, color de pelo, color de ojos, etc.

**Para** tener la libertad de crear un personaje cuyo aspecto sea el que yo quiera.

#### **RF18: Música**

**Como** diseñador,

**Quiero** que los jugadores mientras estén jugando escuchen música la cual cambie acorde a la zona en la que se encuentre o los eventos que estén sucediendo.

**Para** ofrecer una experiencia más agradable e inmersiva.

**RF19: Dificultad ajustable**

**Como** diseñador,

**quiero** que el juego tenga varios niveles de dificultad, haciendo a los enemigos más fuertes o menos fuertes. Esta dificultad es elegida por el jugador y puede ser cambiada en cualquier momento.

**Para** no tener una experiencia abrumadora si el juego es demasiado complicado o una experiencia aburrida si el juego es demasiado sencillo.

**RF20: Menús del juego**

**Como** diseñador,

**quiero** que haya diferentes menús de juego que permitan al jugador cambiar diferentes ajustes del propio juego.

**Para** que el juego sea más personalizable y se adapte a las necesidades individuales de cada usuario.

**RF21: Generación de mundo aleatoria**

**Como** jugador,

**Quiero** que cada vez que empiece una nueva partida, esta no sea exactamente igual a las que haya jugado antes.

**Para** que la experiencia sea distinta y pueda volver a jugar el juego sin aburrirme.

**RF22: Tutorial**

**Como** jugador,

**quiero** que exista la opción de jugar un tutorial.

**Para** poder aprender las mecánicas del juego y no ir perdido en mi primera partida.

**RF23: Mundo variable por dificultad**

**Como** diseñador,

**quiero** que el mapa cambie según la dificultad elegida.

**Para** que cada dificultad se sienta distinta y única.

**RF24: Logros**

**Como** jugador,

**quiero** que haya la posibilidad de conseguir logros a medida que avanzo en mi partida.

**Para** tener otro incentivo para jugar, poder conseguir algo extra, no solo jugar por jugar.

**RF25: Multijugador**

**Como** jugador,

**quiero** que sea posible conectarse con mis amigos y jugar juntos.

**Para** que la experiencia sea más divertida y más personas puedan disfrutar del juego con sus amigos.

**Pruebas de aceptación**

El jugador manda una invitación a un amigo y este entra en su mundo.

El jugador manda una solicitud a un amigo y este la acepta, permitiendo así que el primero entre en el mundo del segundo.

**RF26: Personaje principal**

**Como** diseñador,

**quiero** que el jugador controle a un personaje, con el cual interactuará con el juego en general.

**Para** que los jugadores tengan el medio con el que comunicarse con el mundo.

**RF27: Mapa**

**Como** diseñador,

**Quiero** que el jugador tenga la capacidad de abrir un mapa que le permita ver el mundo desde una perspectiva de planta y marcar cosas en el mismo.

**Para** que los jugadores puedan guiarse y desplazarse más fácil por el mundo.

**RF28: Gráficos ajustables**

**Como** diseñador,

**quiero** que el jugador tenga la capacidad de modificar diferentes aspectos de los gráficos del juego.

**Para** que los jugadores tengan una experiencia visual más satisfactoria y puedan también reducir la calidad gráfica en busca de un mejor rendimiento.

**RNF1: Minimizar errores (bugs)**

**Como** tester,

**quiero** que el proyecto pase por las pruebas de calidad necesarias.

**Para** que la experiencia de juego sea fluida y satisfactoria.



### **RNF2: Código limpio**

**Como** programador,

**quiero** que el código esté ordenado y comentado.

**Para** poder entender fácil y rápido cómo funciona cada parte del juego y seguir así fácilmente expandiendo el código cuando este sea considerablemente largo.

### **RNF3: Disponibilidad en varios idiomas**

**Como** coordinador,

**quiero** que el juego esté disponible en diferentes idiomas.

**Para** que este pueda llegar a más público y así más personas puedan entender el texto.

### **Pruebas de aceptación**

Desde el menú correspondiente, se cambia el idioma a uno diferente. Todo el texto que se muestre ahora está en el idioma escogido.

### **RNF4: Tiempo de carga**

**Como** programador

**quiero** que el tiempo de carga sea el menor posible

**Para** que el jugador no esté atascado en pantallas de carga demasiado largas.

### **RNF5: Arte visual claro**

**Como** diseñador,

**quiero** que el juego tenga un estilo artístico, bonito y claro.

**Para** que visualmente el juego sea agradable a la vista del jugador y dé una sensación de un juego bien realizado.

## Visual Paradigm Professional{Juan(Universidad de Málaga)}





## Sección 5: Casos de uso:

### Identificador único

- CU0. Abrir el juego.

### Contexto de uso

- Cuando el usuario quiera jugar, deberá ejecutar la aplicación del videojuego para poder interactuar con el juego.

### Precondiciones y activación

- El juego no está iniciado aún.

### Postcondición

- El juego estará ya abierto, disponible para interactuar con el usuario.

### Escenario principal

1. El usuario está en el escritorio con el juego cerrado.
2. El usuario ejecuta la aplicación.
3. En la pantalla del usuario aparece el juego iniciado con el menú de inicio.
4. El juego se ejecutará hasta que el usuario lo cierre.

### Escenarios alternativos

- El usuario ya tenía inicialmente el juego abierto y lo vuelve a abrir, esto no ejecuta nuevas instancias del juego sino que pone en primer plano el que estaba suspendido.

### Identificador único

- CU1. Moverse.

### Contexto de uso

- Cuando el usuario necesite mover al personaje podrá hacerlo.

### Precondiciones y activación

- El personaje es visible (no está en un menú).

### Postcondición

- El personaje se mueve en la dirección que ha solicitado el usuario, hasta que deje el usuario de pulsar la tecla.

### Escenario principal

1. El usuario pulsa una de las teclas de movimiento.
2. Su personaje se mueve por el mundo en la dirección vinculada a la tecla hasta que se deje de pulsar la tecla.
3. El personaje se queda quieto si no se continúa pulsando la tecla de movimiento.

### Escenarios alternativos

- El usuario pulsa a la vez dos teclas de movimiento que están en direcciones opuestas, en este caso, el personaje se quedará quieto.

### Identificador único

- CU2. Atacar.





### Contexto de uso

- Cuando el usuario quiera atacar a un enemigo para acabar con él, usará el arma que tenga equipada.

### Precondiciones y activación

- El jugador no está en ningún menú y tiene un arma equipada.

### Postcondición

- El jugador habrá atacado provocando las consecuencias de su ataque en el objetivo y junto con la animación de atacar. Después del ataque el personaje se quedará quieto hasta que el usuario decida realizar otra acción.

### Escenario principal

1. El usuario pulsa la tecla de atacar
2. El personaje realiza la animación de ataque si tiene un arma equipada.
3. El personaje quitará puntos de salud a un enemigo si es que lo hubiese.

### Escenarios alternativos

### Identificador único

- CU3. Atacar con habilidad.

### Contexto de uso

- Cuando el usuario quiera atacar a un enemigo, podrá usar determinadas habilidades ofensivas en vez de atacar de forma normal.

### Precondiciones y activación

- El personaje deberá estar fuera de cualquier menú. Además, debe tener los recursos suficientes para ejecutar la habilidad.

### Postcondición

- El personaje habrá realizado el ataque con la habilidad, la cual provocará diferentes efectos en el objetivo y en el atacante. Además, la habilidad entrará en enfriamiento y deberá esperar un tiempo determinado según la habilidad usada para volver a lanzarla.

### Escenario principal

1. El usuario pulsa la tecla vinculada a la habilidad.
2. El personaje realiza la animación del ataque con esa habilidad, provocando los efectos del mismo.
3. El personaje perderá recursos según indique la habilidad y esta se pondrá en enfriamiento.

### Escenarios alternativos

- El personaje no tiene recursos suficientes, por lo que no realizará el ataque con la habilidad.
- La habilidad está en enfriamiento, por lo que el personaje no puede usarla.

### Identificador único

- CU4. Crear personaje.

### Contexto de uso

- Antes de poder jugar es necesario un personaje, el usuario podrá crear su propio personaje a la hora de crear una partida.



### Precondiciones y activación

- Debe de haber disponible un hueco de personaje.

### Postcondición

- El personaje se habrá creado y se quedará guardado.

### Escenario principal

1. El usuario decide crear un nuevo personaje.
2. El usuario decide qué tipo de clase elegir.
3. El usuario tendrá la opción de personalizar su personaje.
4. Cuando acabe, se quedará guardado para poder usarlo cuando desee.

### Escenarios alternativos

- Cuando el usuario intente crear un personaje y no haya hueco.

### Identificador único

- CU5. Abrir menú principal.

### Contexto de uso

- Cuando el jugador quiera, podrá abrir y cerrar el menú principal.

### Precondiciones y activación

- El jugador no está en ningún menú.

### Postcondición

- El jugador se encuentra en el menú principal.

### Escenario principal

1. El jugador pulsa la tecla asociada a abrir el menú principal.
2. Se muestra el menú con diferentes opciones, incluida la de cerrar el propio menú principal.

### Escenarios alternativos

### Identificador único

- CU6. Guardar progreso.

### Contexto de uso

- Cuando el jugador quiera, puede guardar el progreso realizado en el juego.

### Precondiciones y activación

- El jugador se encuentra en el menú principal.

### Postcondición

- La partida se guarda.

### Escenario principal

1. El jugador selecciona la opción de partidas guardadas.
2. El menú muestra varias ranuras diferentes donde guardar el progreso.
3. El jugador selecciona una y elige la opción de guardar.
4. La partida se guarda en la ranura seleccionada.

### Escenarios alternativos

- La partida no se pudo guardar satisfactoriamente y se muestra un mensaje de error.



### Identificador único

- CU7. Cargar progreso.

### Contexto de uso

- Cuando el jugador quiera, puede cargar una ranura donde tenga datos de guardado y seguir jugando desde ese punto.

### Precondiciones y activación

- El jugador se encuentra en el menú principal.

### Postcondición

- Todos los elementos del juego cambian para ser los mismos que los que el jugador guardó en la ranura seleccionada.

### Escenario principal

1. El jugador selecciona la opción de partidas guardadas.
2. El menú muestra varias ranuras diferentes de las cuales se puede cargar el progreso.
3. El jugador selecciona una y elige la opción de cargar.
4. La partida se carga.

### Escenarios alternativos

- La partida no se pudo cargar satisfactoriamente y se muestra un mensaje de error.
- Si se carga una ranura vacía, el estado del mundo será el del comienzo.

### Identificador único

- CU8. Consultar inventario.

### Contexto de uso

- Cuando el jugador lo desee, puede ver su inventario e interactuar con lo que guarda.

### Precondiciones y activación

- El jugador no está en un diálogo.

### Postcondición

- Aparece un menú en la pantalla que muestra todo lo que el jugador tiene guardado.

### Escenario principal

1. El jugador pulsa la tecla vinculada al inventario.
2. Se abre el menú de inventario que muestra todo lo que el jugador tiene guardado.

### Escenarios alternativos

### Identificador único

- CU9. Utilizar un objeto del inventario.

### Contexto de uso

- Cuando el jugador lo desee, puede interactuar con algún objeto de su inventario.

### Precondiciones y activación

- El jugador se encuentra con el menú de inventario abierto.

### Postcondición

- Uno de los objetos del inventario se ha consumido, ha sido equipado, ha cambiado...



### Escenario principal

1. El jugador selecciona un objeto del inventario.
2. Aparece un menú contextual que muestra diferentes opciones para el objeto: “usar”, “tirar”, “mostrar detalles” y, en caso de armas/armaduras, “equipar”.
3. El jugador selecciona lo que quiere hacer con el mismo.
4. El objeto es utilizado, desaparece del inventario, se muestra un texto con más detalles del mismo o se marca con un símbolo para indicar que está siendo equipado.

### Escenarios alternativos

- Si el jugador selecciona un objeto que tenía equipado, la opción de “equipar” se sustituye por “desequipar”, lo que permite deshacer el equipamiento.

### Identificador único

- CU10. Cambiar la dificultad.

### Contexto de uso

- Cuando el jugador quiera, podrá cambiar la dificultad del mundo.

### Precondiciones y activación

- El jugador se encuentra en el menú principal.

### Postcondición

- La dificultad del mundo cambia.

### Escenario principal

1. El jugador selecciona la opción de cambiar dificultad.
2. El menú muestra las diferentes opciones, desde fácil hasta difícil.
3. El jugador selecciona la dificultad deseada.
4. Todos los elementos del mundo que deban cambiar, lo hacen acorde a la dificultad escogida y el menú destaca la dificultad en la que se encuentra el mundo.

### Escenarios alternativos

- Si el jugador selecciona la dificultad que el mundo ya tenía, no sucede nada.

### Identificador único

- CU11. Ver progreso de misiones.

### Contexto de uso

- Cuando el jugador quiera ver cuánto ha avanzado en las misiones del juego.

### Precondiciones y activación

- El jugador se encuentra en el menú principal.

### Postcondición

- El jugador sigue en el menú dentro del juego..

### Escenario principal

1. El jugador selecciona el apartado de misiones dentro del menú.
2. Se abrirá un apartado con todas las misiones y cuanto lleva completado de cada una.
3. El jugador podrá cerrar este menú para volver al anterior.

### Escenarios alternativos



- El jugador se equivoca de apartado y selecciona otro.

### Identificador único

- CU12. Seleccionar una misión.

### Contexto de uso

- Cuando el jugador quiera, podrá seleccionar una misión en específico para completarla.

### Precondiciones y activación

- El jugador se encuentra en el menú de las misiones

### Postcondición

- Arriba a la derecha de la pantalla se muestra la misión que el jugador está siguiendo.

### Escenario principal

1. El jugador selecciona la misión que quiere completar dentro del menú de misiones.
2. Al seleccionar la misión, se le devuelve al juego cerrando el menú.
3. Ahora le sale la misión arriba a la derecha y el porcentaje del progreso total que lleva en la misma misión.

### Escenarios alternativos

- Pulsa otro botón y abandona el menú de las misiones.

### Identificador único

- CU13. Usar habilidad defensiva.

### Contexto de uso

- El jugador está por recibir un ataque.

### Precondiciones y activación

- El jugador tiene una habilidad de defensa que puede activar para desviar o reducir el daño de los ataques enemigos.

### Postcondición

- Los ataques enemigos son mitigados o desviados.

### Escenario principal

1. El jugador activa la habilidad de defensa.
2. Realiza la animación de defenderse y provoca los efectos de la habilidad.

### Escenarios alternativos

- El jugador usa la habilidad a destiempo y la falla, sufriendo los efectos del ataque.

### Identificador único

- CU14. Curarse en el combate.

### Contexto de uso

- El jugador está en medio de una batalla y necesita recuperar salud.

### Precondiciones y activación

- El jugador tiene una habilidad de curación disponible.



- El jugador selecciona la habilidad de curación y la utiliza.

#### Postcondición

- El jugador recupera salud.

#### Escenario principal

1. El jugador pulsa la teclas asociada a la habilidad de curación
2. Realiza la animación de curación
3. Sube la barra de salud del personaje y con ello su cantidad de vida.

#### Escenarios alternativos

- Se equivoca de botón y se utiliza otra habilidad.
- Se intenta curar estando con la vida llena, lo que no provoca ningún efecto ni se consume la habilidad.

#### Identificador único

- CU15. Interactuar con NPC.

#### Contexto de uso

- Cuando el jugador lo necesite, puede interactuar con un NPC.

#### Precondiciones y activación

- El jugador debe de estar al lado de un NPC y pulsar la tecla de interactuar.

#### Postcondición

- El NPC habrá dado una ventaja o una misión al jugador ya sea dando una pista, un arma o asignando una misión.

#### Escenario principal

1. Al estar próximo a un NPC y pulsar la tecla saltará un cuadro de diálogo.
2. El NPC le podrá dar o un consejo u mejora o le asignará una misión al jugador.
3. Si le da un objeto, el objeto se irá al inventario, si le da una misión, la misión se guardará en el menú de misiones.

#### Escenarios alternativos

- El NPC es hostil y no se puede hablar con él o el jugador ya ha hablado con él y no tiene nada nuevo que decir.

#### Identificador único

- CU16. Desbloquear una parte del mapa.

#### Contexto de uso

- Cuando el jugador pueda, podrá avanzar a nuevas partes del mapa

#### Precondiciones y activación

- El jugador debe cumplir una serie de requisitos para poder acceder a diferentes partes del mapa, una vez cumplidos los requisitos puede viajar a esa parte del mapa

#### Postcondición

- Una parte del mapa se desbloquea para que el jugador pueda explotarla.

#### Escenario principal



1. El jugador cumple los requisitos, ya sea cumpliendo una misión o alcanzando cierto nivel, obteniendo un mensaje que le informa sobre haber desbloqueado una nueva parte del mapa.
2. Se elimina lo que impedía que el jugador pudiera avanzar hacia esa parte del mapa.
3. El jugador ya puede explorar esa nueva parte del mapa

### Escenarios alternativos

- Volver a cumplir los requisitos o cumplir otros alternativos no desbloquea de nuevo una zona ya desbloqueada.

### Identificador único

- CU17. Desbloquear historia.

### Contexto de uso

- Cuando un jugador busca avanzar en la trama principal del juego o desbloquear nuevas partes de la historia al completar eventos o misiones específicas.

### Precondiciones y activación

- El jugador debe tener acceso a eventos o misiones que estén relacionados con la progresión de la historia principal del juego.

### Postcondición

- Se desbloquea una nueva parte de la trama principal o se revela información adicional sobre la historia del juego.

### Escenario principal

1. El jugador participa en un evento especial o completa una misión que está vinculada con la trama principal del juego.
2. Se revelan detalles relevantes sobre la historia del juego o se presenta una situación que impacta directamente en la trama.
3. El jugador toma decisiones o realiza acciones que tienen consecuencias en la historia, como desbloquear nuevos personajes, ubicaciones o misiones.
4. Una vez completado el evento o la misión, se muestra una secuencia de eventos o diálogos que avanza la trama principal del juego.

### Escenarios alternativos

- Si el jugador ha completado todas las misiones disponibles en ese punto de la historia, se le proporciona información adicional sobre eventos futuros o se desbloquean nuevas misiones secundarias que podrían revelar más detalles sobre la trama principal.

### Identificador único

- CU18. Comerciar objetos con NPC.

### Contexto de uso

- El jugador podrá intercambiar, comprar o vender objetos a los NPC.

### Precondiciones y activación

- El jugador debe estar en una ubicación donde el NPC esté disponible para comerciar y tener los objetos que desea intercambiar en su inventario.

### Postcondición



- Los objetos seleccionados se intercambian con el NPC y pueden ser adquiridos por otros objetos o moneda dentro del juego.

### Escenario principal

1. El jugador interactúa con el NPC abriendo su cuadro de diálogo.
2. El jugador selecciona la opción de comerciar con el NPC en el diálogo.
3. El NPC presenta su inventario de objetos disponibles para comerciar.
4. El jugador selecciona los objetos que desea intercambiar y los coloca en la interfaz de comercio.
5. El NPC muestra los objetos o moneda que está dispuesto a ofrecer a cambio de los objetos del jugador.
6. El jugador revisa la oferta del NPC y confirma el intercambio si está de acuerdo.
7. Los objetos intercambiados se transfieren entre el inventario del jugador y el inventario del NPC.
8. Se cierra el menú de comercio y vuelve al diálogo.

### Escenarios alternativos

- El jugador no tiene suficientes objetos para realizar un intercambio, el NPC muestra un mensaje indicando la falta de recursos suficientes.
- El jugador cancela el comercio antes de confirmar, se cierra la interfaz de comercio y no se realiza ningún intercambio.
- El jugador intenta comerciar con un NPC que no ofrece esta función, el NPC muestra un mensaje indicando que no puede/quiere comerciar.

### Identificador único

- CU19. Reparar un objeto con NPC.

### Contexto de uso

- El jugador podrá reparar objetos dañados.

### Precondiciones y activación

- El jugador debe estar en una ubicación donde un NPC ofrezca el servicio de reparación y tener objetos dañados en su inventario.

### Postcondición

- Los objetos dañados son reparados por el NPC a cambio de una cantidad de monedas y/o recursos del juego.

### Escenario principal

1. El jugador interactúa con el NPC abriendo su cuadro de diálogo.
2. El jugador selecciona la opción de reparar objetos.
3. El NPC muestra una lista de objetos dañados en el inventario del jugador.
4. El jugador selecciona los objetos que desea reparar.
5. El NPC muestra el costo de reparación para cada objeto seleccionado.
6. El jugador confirma la reparación y paga el costo requerido.
7. El NPC repara los objetos seleccionados y los devuelve al inventario del jugador.

### Escenarios alternativos

- El jugador no tiene suficientes recursos para pagar la reparación de todos los objetos seleccionados, el NPC muestra un mensaje indicando la falta de recursos.





- El jugador cancela la reparación antes de confirmar, se cierra la interfaz de reparación y no se realiza ninguna reparación.
- El jugador intenta reparar objetos que no pueden ser reparados por el NPC, el NPC muestra un mensaje indicando que esos objetos no son reparables por él.

### Identificador único

- CU20. Luchar junto a NPC como aliado.

### Contexto de uso

- Cuando un jugador desea tener la ayuda de un NPC en combate como aliado durante una misión o evento específico dentro del juego.

### Precondiciones y activación

- El jugador debe encontrarse en una situación de combate donde sea posible la participación del NPC como aliado.

### Postcondición

- El jugador y el NPC aliado combaten juntos contra los enemigos, elevando las posibilidades de éxito en la batalla.

### Escenario principal

1. Durante un combate, el jugador interactúa con el NPC abriendo su cuadro de diálogo.
2. El jugador selecciona la opción de solicitar ayuda al NPC para que se una a la lucha como aliado.
3. El NPC acepta la solicitud y se une al jugador en la batalla como aliado, cerrando el diálogo.
4. El jugador y el NPC aliado coordinan estrategias de combate y luchan juntos contra los enemigos presentes.
5. El jugador y el NPC aliado colaboran para derrotar a los enemigos y completar los objetivos de la misión o evento.
6. Una vez finalizada la batalla, el NPC aliado regresa a su ubicación original o continúa acompañando al jugador, según el contexto del juego.

### Escenarios alternativos

- El jugador no solicita la ayuda del NPC aliado durante el combate, el NPC permanece inactivo y no participa en la batalla.
- El NPC aliado es derrotado en combate, el jugador puede intentar revivirlo si cuenta con los medios necesarios dentro del juego o el NPC queda incapacitado durante el resto del combate.
- El NPC no acepta ayudar al jugador y por tanto no se une a su equipo.

### Identificador único

- CU21. Multijugador.

### Contexto de uso

- Cuando un jugador quiera poder jugar con otras personas.

### Precondiciones y activación

- El jugador debe estar en el menú principal.
- Tendrá que tener a la otra persona agregada en su lista de amigos para que se pueda unir a la partida.



### Postcondición

- Ambos jugadores juegan juntos la misma partida, en el mismo mundo.

### Escenario principal

1. El jugador está en el menú para crear una nueva partida.
2. Elige la opción de crear un mundo multijugador.
3. Se muestra su lista de amigos.
4. Invita al amigo con el que quiera jugar seleccionándolo.

### Escenarios alternativos

- El jugador pulsa el botón de salir y abandona el menú de creación de partida.
- El jugador que recibe la invitación la rechaza, por lo que no se une a su mundo.

### Identificador único

- CU22. Jugar tutorial.

### Contexto de uso

- Cuando el jugador quiera aprender los controles básicos del juego y cómo funciona.

### Precondiciones y activación

- El jugador debe estar en el menú principal del juego.

### Postcondición

- El jugador juega el tutorial.

### Escenario principal

1. El jugador está en el menú principal.
2. Selecciona la opción de tutorial.
3. El jugador entra al tutorial y aprende los controles, como derrotar enemigos, interactuar con el mapa y más conceptos básicos del juego.

### Escenarios alternativos

- El jugador cancela el tutorial mediante el menú principal, volviendo a donde se encontraba..

### Identificador único

- CU23. Resolver un puzzle.

### Contexto de uso

- El jugador podrá resolver puzzles para seguir avanzando.

### Precondiciones y activación

- El jugador debe encontrar la zona donde se encuentra el puzzle.
- El jugador deberá tener cierto nivel o cierto objeto para poder completar el puzzle.

### Postcondición

- El puzzle queda resuelto.

### Escenario principal

1. El jugador llega a la zona donde se encuentra el puzzle e interactúa con el mismo.
2. Se le explica al jugador cual es su objetivo, o los requerimientos para poder intentarlo
3. El jugador interactúa con el puzzle para resolverlo, si lo logra, provoca un efecto, ya sea desbloquear una zona, proporcionar algún objeto al jugador, etc.



### Escenarios alternativos

- El jugador no cumple con los requisitos mínimos para intentar el puzzle, por lo que no puede hacerlo.
- El jugador no tiene en su inventario algún objeto necesario para hacer el puzzle, por lo que no puede hacerlo.
- El jugador no sabe resolver el puzzle, por lo que lo abandona temporalmente y el progreso con el mismo se guarda para su posterior retorno.
- El puzzle ya fue resuelto, por lo que no puede interactuar con el mismo.

### Identificador único

- CU24. Mejorar un objeto con NPC.

### Contexto de uso

- Cuando el jugador tenga un objeto y pueda mejorarlo.

### Precondiciones y activación

- El jugador debe tener el objeto a mejorar en el inventario.
- Debe tener los recursos necesarios para mejorarlo.
- Debe estar en la zona donde se encuentra el NPC que le permita la mejora de objetos.

### Postcondición

- Los objetos son mejorados y el jugador pierde los recursos que le ha costado la mejora.

### Escenario principal

1. El jugador llega a la zona donde el NPC le puede mejorar.
2. El jugador interactúa con el NPC abriendo su cuadro de diálogo y selecciona la opción de mejorar un objeto.
3. Se muestran los objetos con posible mejora en el inventario del jugador.
4. El jugador elige el objeto a mejorar.
5. Se muestra el costo de la mejora.
6. El jugador confirma la mejora, obteniendo con ello el objeto mejorado en cuestión y perdiendo los recursos que le costó el mejorarlo.
7. El jugador vuelve al diálogo anterior.

### Escenarios alternativos

- El jugador no dispone de los objetos requeridos para realizar la mejora, por lo que no se realiza la mejora y el NPC le informa de que le faltan recursos.

### Identificador único

- CU25. Ajustar gráficos.

### Contexto de uso

- El jugador podrá cambiar diferentes opciones gráficas del juego.

### Precondiciones y activación

- Estar dentro del menú principal.
- Acceder al menú de gráficos.

### Postcondición

- Los gráficos del juego cambian.



### Escenario principal

1. El jugador selecciona la opción de gráficos.
2. Selecciona los gráficos que le parezcan más adecuados.
3. Selecciona el botón de guardar cambios y estos se aplican.

### Escenarios alternativos

- El jugador no guarda los cambios y sale del menú, por lo que no se efectúa ningún cambio.

### Identificador único

- CU26. Ajustar sonido.

### Contexto de uso

- El jugador podrá cambiar diferentes opciones de sonido del juego.

### Precondiciones y activación

- Estar dentro del menú principal.
- Acceder al menú de sonido

### Postcondición

- Las propiedades del sonido cambian.

### Escenario principal

1. El jugador selecciona la opción de sonido.
2. Selecciona los ajustes que le parezcan más adecuados.
3. Selecciona el botón de guardar cambios y estos se aplican.

### Escenarios alternativos

- El jugador no guarda los cambios y sale del menú, por lo que no se efectúa ningún cambio.

### Identificador único

- CU27. Menú inicial.

### Contexto de uso

- El jugador verá el menú inicial cuando inicie el juego.

### Precondiciones y activación

- Abrir el juego.

### Postcondición

- El jugador se encuentra en el menú inicial.

### Escenario principal

1. El jugador abre el juego y aparece el menú inicial.

### Escenarios alternativos

### Identificador único

- CU28. Cargar, eliminar o crear partida.

### Contexto de uso



- El jugador podrá cargar, eliminar o crear una nueva partida mediante el menú inicial.

### Precondiciones y activación

- Estar en el menú inicial.

### Postcondición

- El jugador habrá eliminado una partida o iniciará una.

### Escenario principal

1. El jugador selecciona la opción de cargar o crear un mundo nuevo, por lo que la partida se inicia.

### Escenarios alternativos

- El jugador selecciona la opción de eliminar una partida guardada, por lo que esta se borra luego de confirmar que el jugador desea borrarla.

### Identificador único

- CU29. Abrir mapa.

### Contexto de uso

- El jugador podrá abrir un mapa.

### Precondiciones y activación

- El jugador no se encuentra en combate ni en un diálogo.

### Postcondición

- Se abre el mapa.

### Escenario principal

1. El jugador pulsa la tecla asociada al mapa.
2. El mapa se muestra.

### Escenarios alternativos

- El jugador pulsa la tecla para abrir el mapa estando en el mapa, por lo que este se cierra.

### Identificador único

- CU30. Usar habilidad.

### Contexto de uso

- El jugador podrá usar habilidades para diferentes propósitos.

### Precondiciones y activación

- El personaje deberá estar fuera de cualquier menú. Además, debe tener los recursos suficientes para ejecutar la habilidad.

### Postcondición

- El personaje habrá usado la habilidad, la cual provocará diferentes efectos. Además, la habilidad puede entrar en enfriamiento y el jugador deberá esperar un tiempo determinado según la habilidad usada para volver a lanzarla.

### Escenario principal

1. El usuario pulsa la tecla vinculada a la habilidad.
2. El personaje realiza la animación de la habilidad, provocando los efectos de la misma.
3. El personaje perderá recursos según indique la habilidad y esta podría entrar en enfriamiento.



### Escenarios alternativos

- El personaje no tiene recursos suficientes, por lo que no usa la habilidad.
- La habilidad está en enfriamiento, por lo que el personaje no puede usarla.

### Identificador único

- CU31. Cambiar de sala.

### Contexto de uso

- El jugador podrá, mediante una puerta u otro tipo de enlace, cambiar de una sala a otra.

### Precondiciones y activación

- El personaje deberá estar fuera de cualquier menú. Además, debe estar en contacto con la puerta que le permitirá ir a otra sala.

### Postcondición

- El personaje se encontrará en una sala diferente a la que estaba anteriormente.

### Escenario principal

1. El jugador pulsa la tecla vinculada a interactuar.
2. Aparece una transición breve.
3. El personaje habrá cambiado su ubicación.

### Escenarios alternativos

- La puerta no se puede utilizar, en ese caso, se mostrará un cuadro de diálogo y no cambiará de sala.

### Identificador único

- CU32. Interactuar con un objeto del mundo.

### Contexto de uso

- El jugador podrá interactuar con determinados objetos del mundo.

### Precondiciones y activación

- El personaje deberá estar fuera de cualquier menú. Además, debe estar en contacto con el objeto con el que quiere interactuar.

### Postcondición

- El personaje habrá interactuado con el objeto, lo que puede provocar diferentes efectos, como haber abierto algún puzzle, cambiado una parte del mapa, obtenido algo...

### Escenario principal

1. El jugador pulsa la tecla vinculada a interactuar.
2. Se desencadenan los eventos necesarios según el objeto con el que el jugador interactuó.

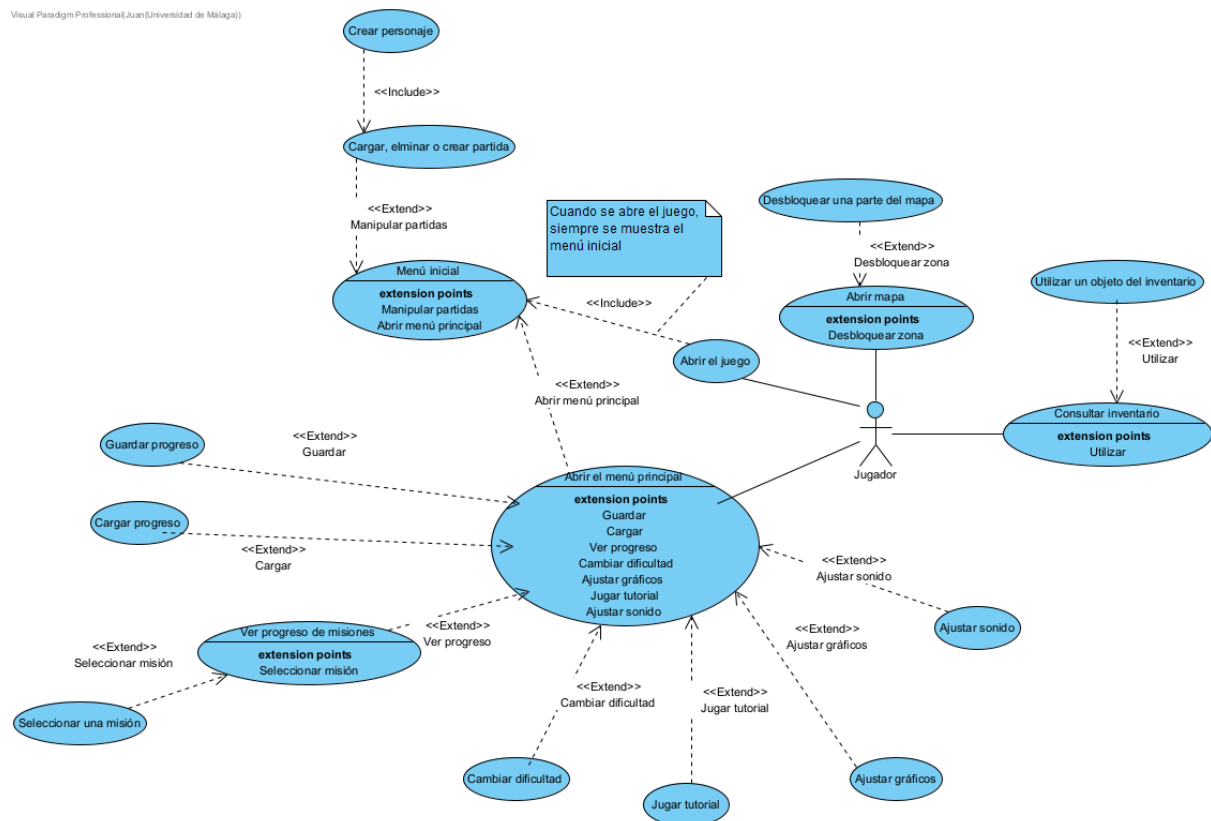
### Escenarios alternativos

- El jugador trata de interactuar con un objeto con el que ya interactuó, esto no provoca de nuevo su efecto, sino que en su lugar, o no sucede nada, o se muestra un cuadro de diálogo para indicarle al jugador que ya ha interactuado con el objeto en cuestión.

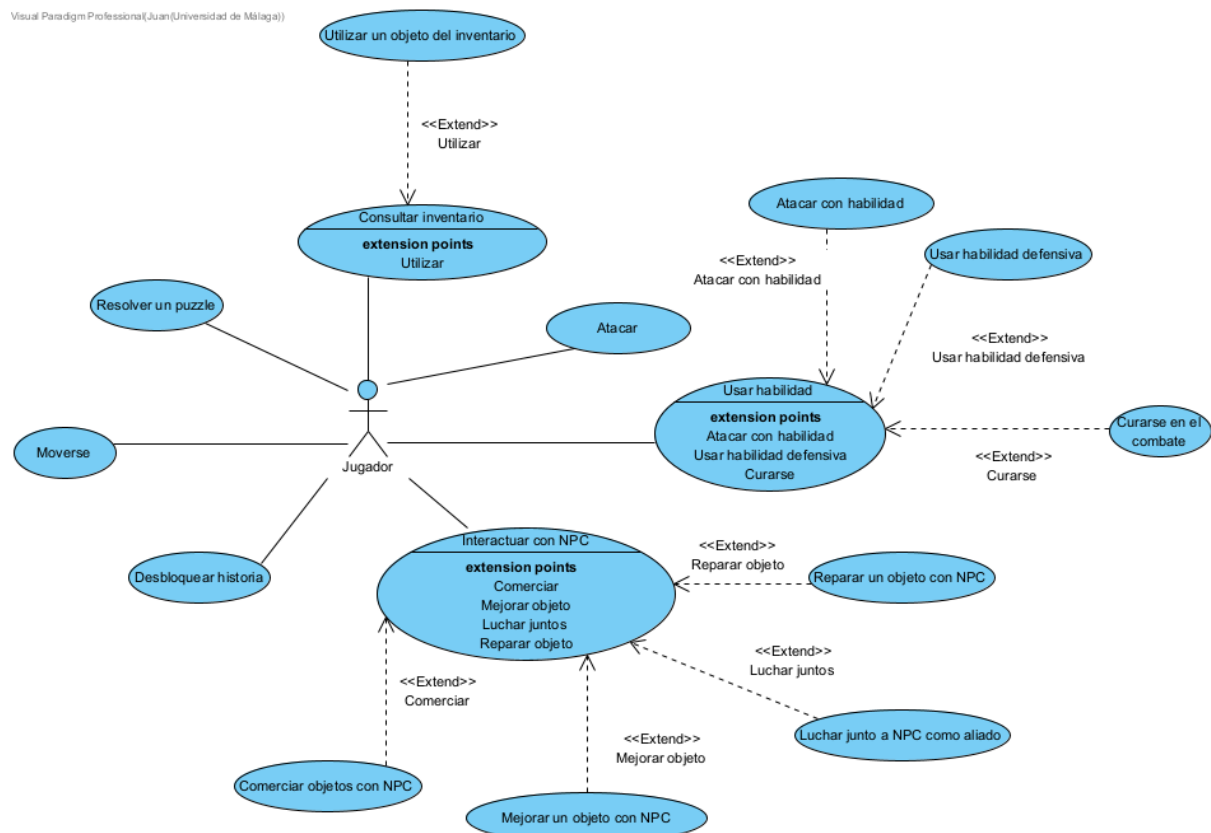


# Diagramas de Visual Paradigm:

Visual Paradigm Profesional(Juan(Universidad de Málaga))



Visual Paradigm Profesional(Juan(Universidad de Málaga))





## Sección 7:

**Clase Inventario:** el inventario del jugador, el cual contiene todos los objetos que guarde. Atributos: objetos almacenados, capacidad máxima, capacidad actual. Relación de asociación uno a muchos con Objeto, pues un inventario puede contener muchas armas (o ninguna).

**Clase Objeto:** un objeto que puede ser guardado y utilizado por el jugador. Atributos: nombre, peso, descripción, precio, rareza.

**Clase Arma:** un tipo de objeto que permite ser equipado para infligir daño en combate. Atributos: daño, habilidad activa, habilidad pasiva. Relación de generalización de objeto.

**Clase Armadura:** un tipo de objeto que permite ser equipado para mitigar daño en combate. Atributos: defensa, habilidad pasiva. Relación de generalización de objeto.

**Clase Consumible:** un tipo de objeto que permite ser consumido para obtener un determinado efecto. Atributos: efecto, duración. Relación de generalización de objeto.

**Clase NPC:** personaje no jugable, entidades con las cuales el usuario podrá interactuar. Atributos: nombre, vida máxima, vida actual, historia, movimientos, ataques, diálogos, capacidad de comerciar, capacidad de asignar misiones, capacidad de unirse al equipo.

**Clase Boss:** un tipo de NPC más fuerte y con más relevancia que además será parte de un enfrentamiento a mayor escala. Atributos: habilidades, recompensa al ser derrotado. Relación de generalización de NPC.

**Clase Habilidad:** las habilidades que el jugador (u otras entidades) podrán realizar. Atributos: coste de uso, descripción, enfriamiento, tipo (ofensiva, defensiva o ninguno). Relación de asociación muchos a muchos con Jugador y con Boss, pues ambas clases pueden aprender estas habilidades.

**Clase Mapa:** el mapa que el jugador podrá consultar. Atributos: localización del jugador, sitios de interés, puntero destino.

**Clase Jugador:** es el personaje que controlará el usuario. Atributos: nombre, vida máxima, vida actual, experiencia, nivel, defensa, arma equipada, habilidades, armadura equipada.

**Clase Menú:** un menú que el jugador puede abrir para seleccionar distintas opciones. Atributos: opciones (este será un array u otra estructura de datos que almacena todas las opciones posibles del menú en cuestión).

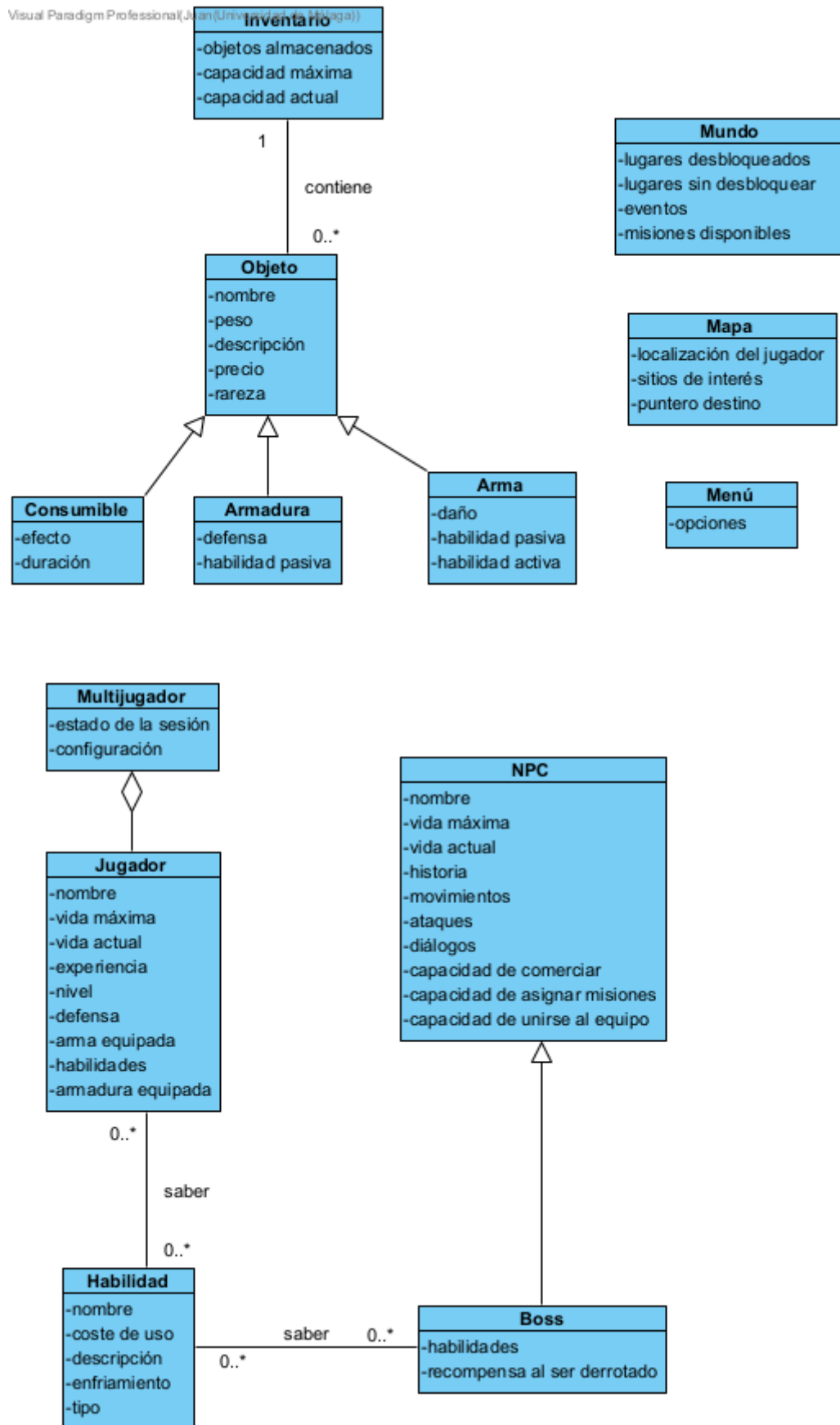
**Clase Mundo:** el escenario donde se desarrolla toda la trama y el usuario juega. Atributos: lugares desbloqueados, lugares sin desbloquear, eventos, misiones disponibles.

**Clase Multijugador:** la clase encargada de gestionar el modo multijugador. Atributos: estado de la sesión, configuración. Relación de agregación con Jugador.





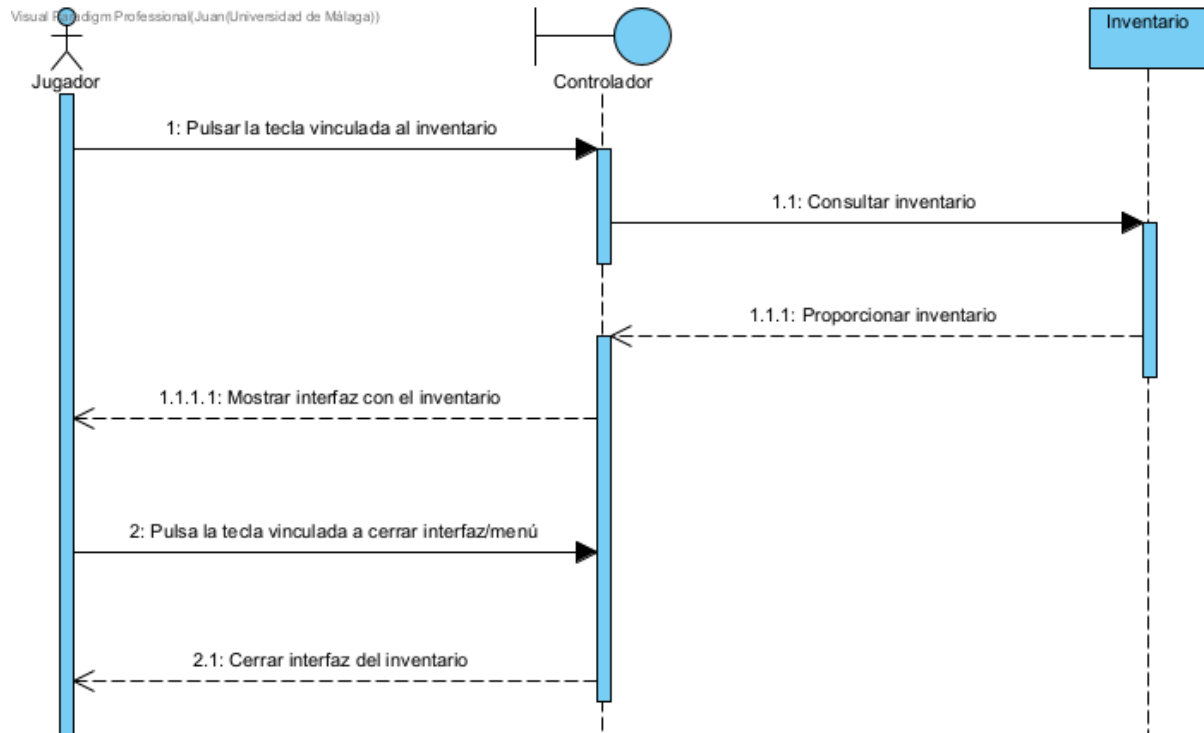
## Diagrama de Visual Paradigm:



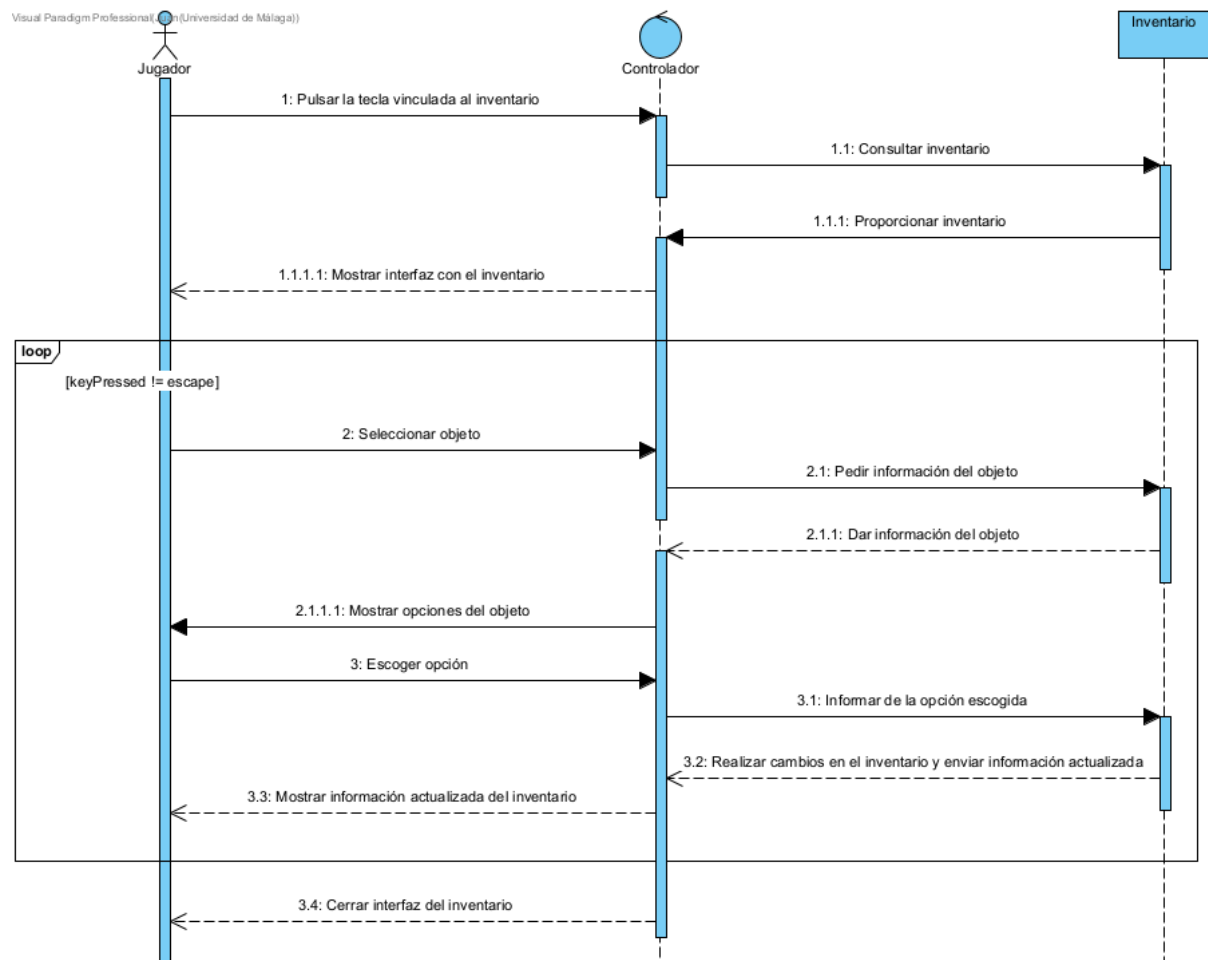


## Sección 8:

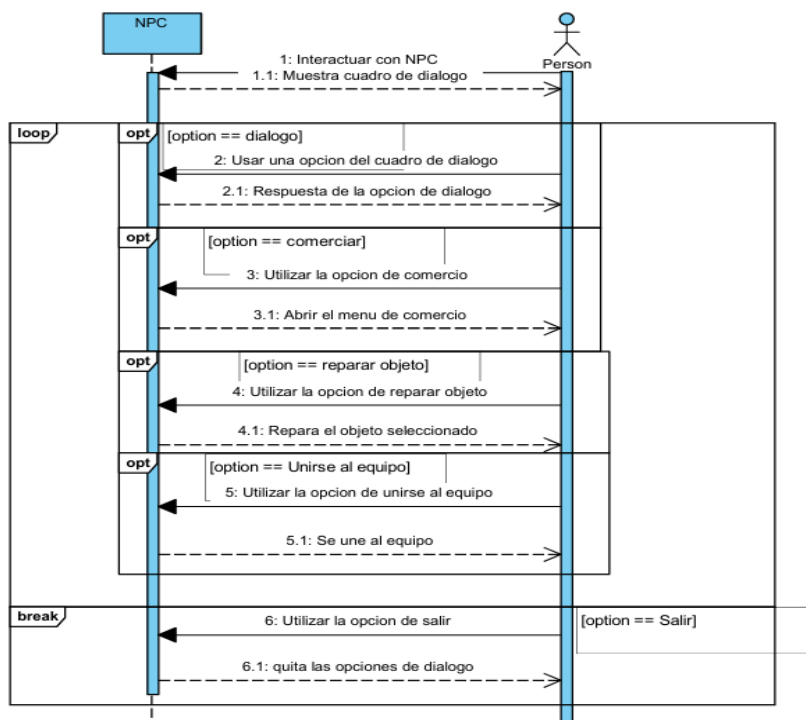
1. Consultar inventario: el jugador simplemente solicita ver el inventario, cuando lo desee, podrá cerrarlo.



2. Utilizar un objeto del inventario: el jugador quiere consultar información más concreta sobre algún/os objeto/s del inventario, hasta que desee cerrarlo. El inventario se actualizará en función de lo que el jugador haga con el objeto.

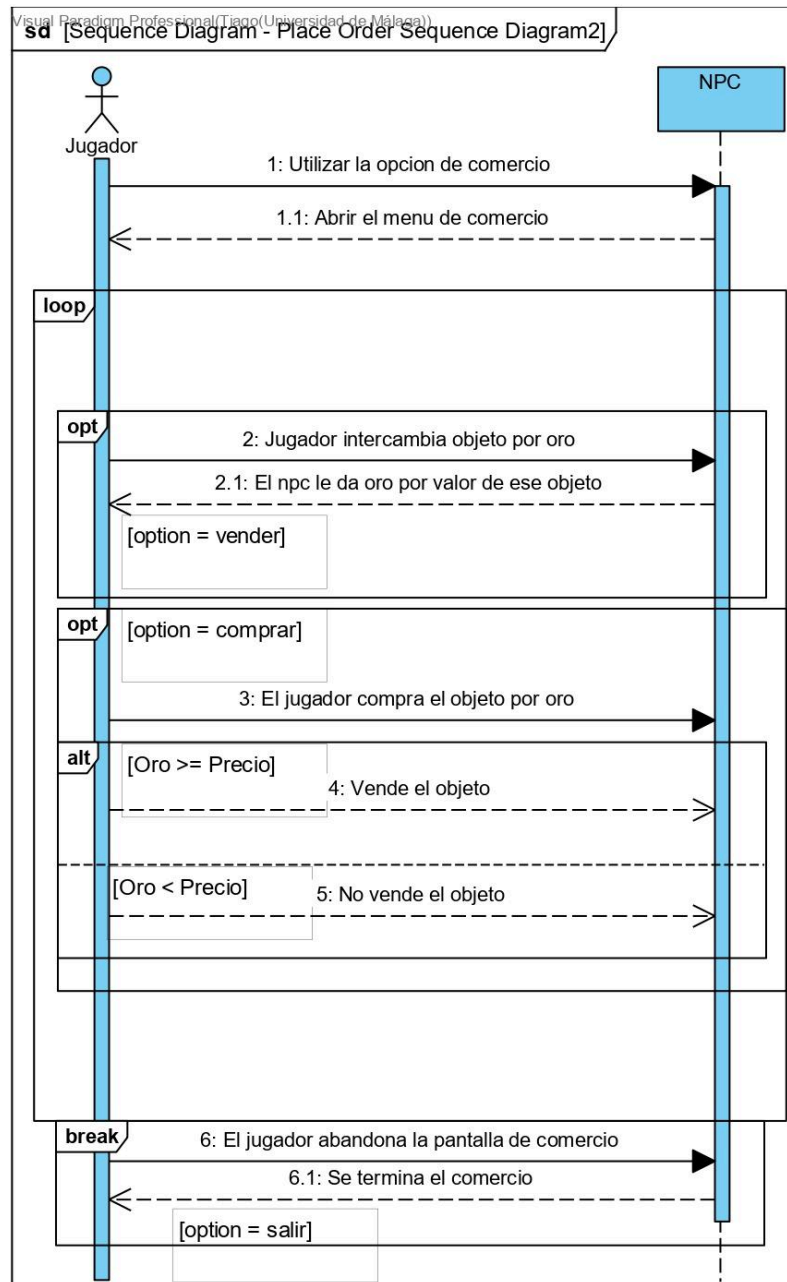


3. Interactuar con NPC: el jugador pulsando una tecla, interactúa con un Npc y escoge una opción de diálogo.

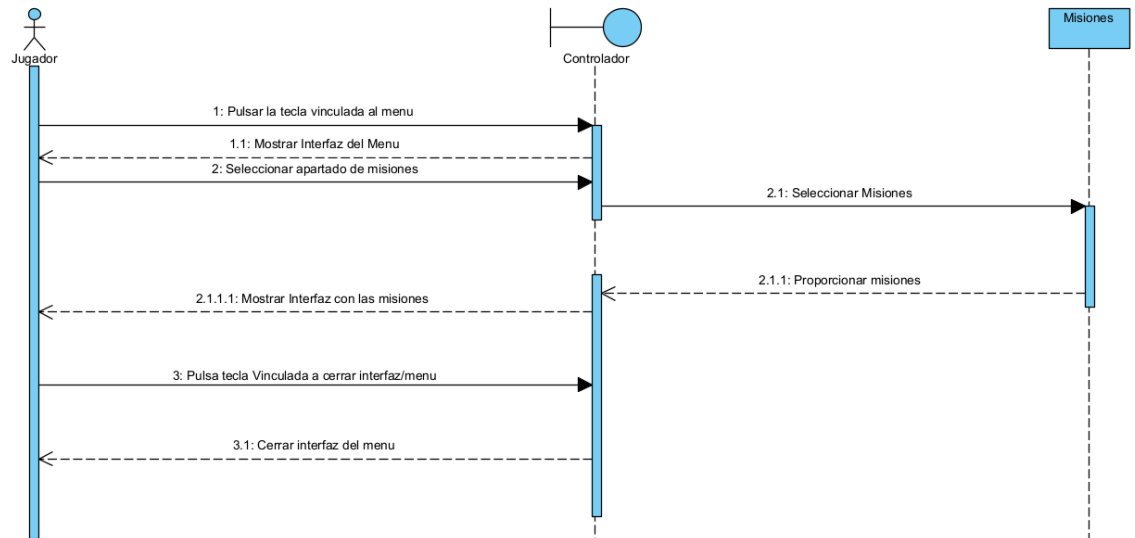




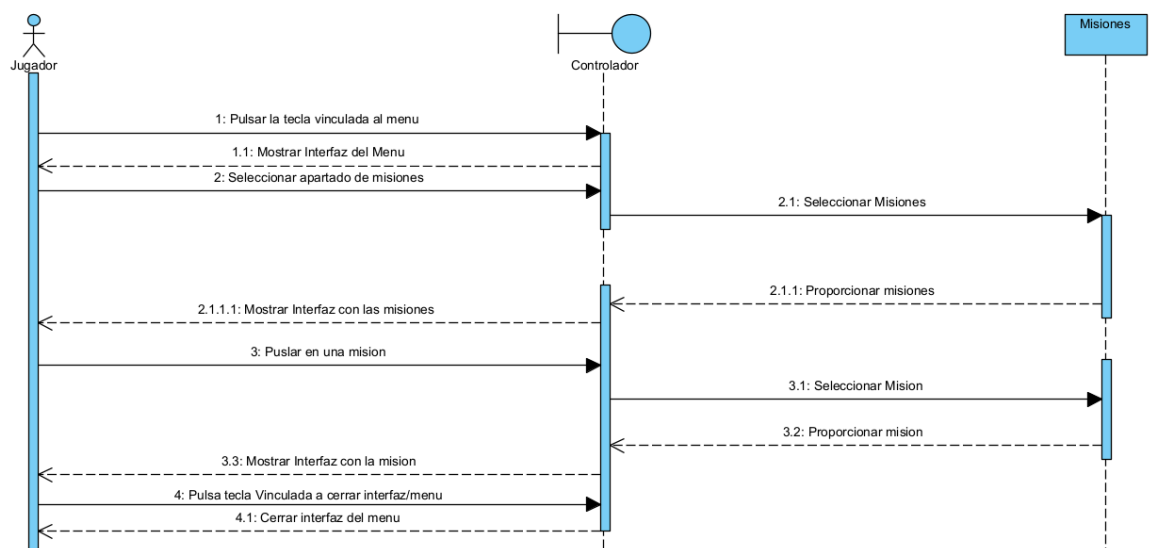
4. Comerciar objetos con NPC: el jugador escoge la opción de comerciar y abre un menú de comercio donde puede intercambiar objetos a cambio de oro.



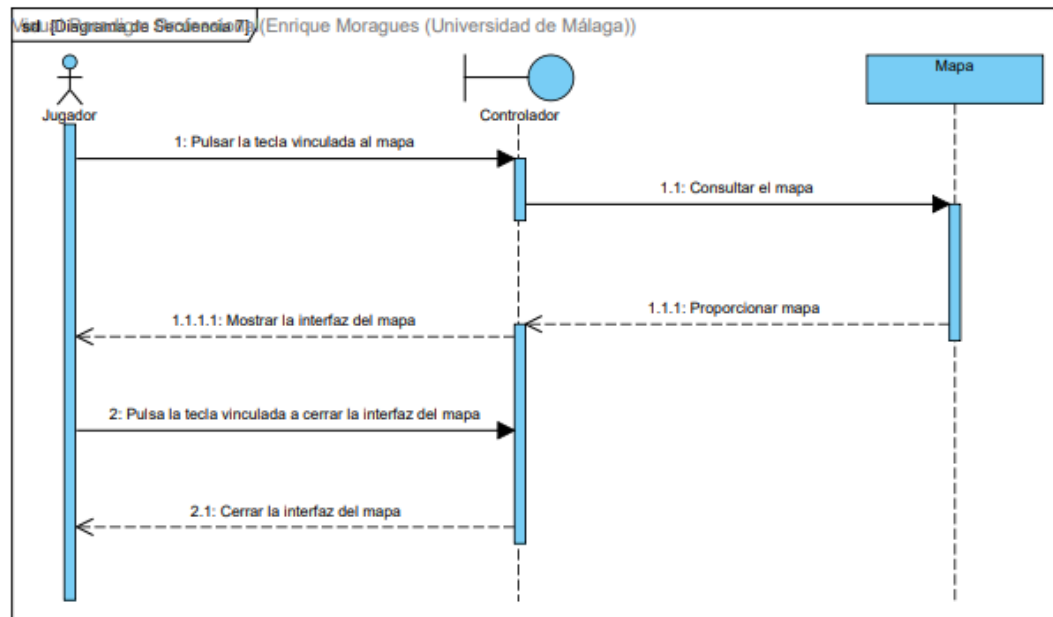
5. Ver progreso de misiones: el jugador pulsa la tecla vinculada al menú principal, posteriormente, selecciona las misiones para consultarlas y luego cierra este menú para seguir jugando.



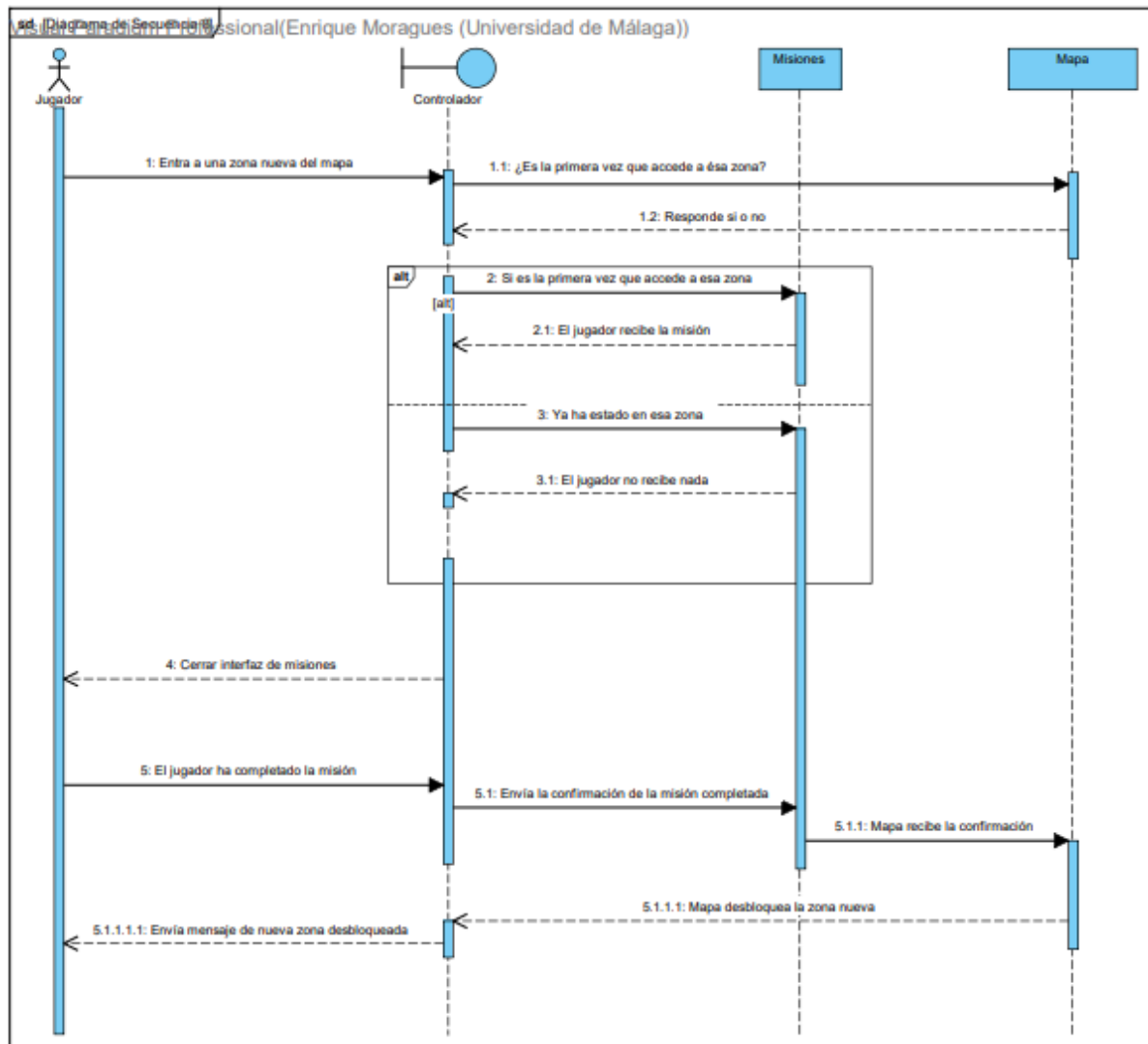
6. Seleccionar una misión: el jugador abre el menú principal, selecciona las misiones y posteriormente selecciona la misión deseada, luego de eso, cierra el menú.



7. Abrir el mapa: el jugador pulsa la tecla correspondiente al mapa, y este se abre.

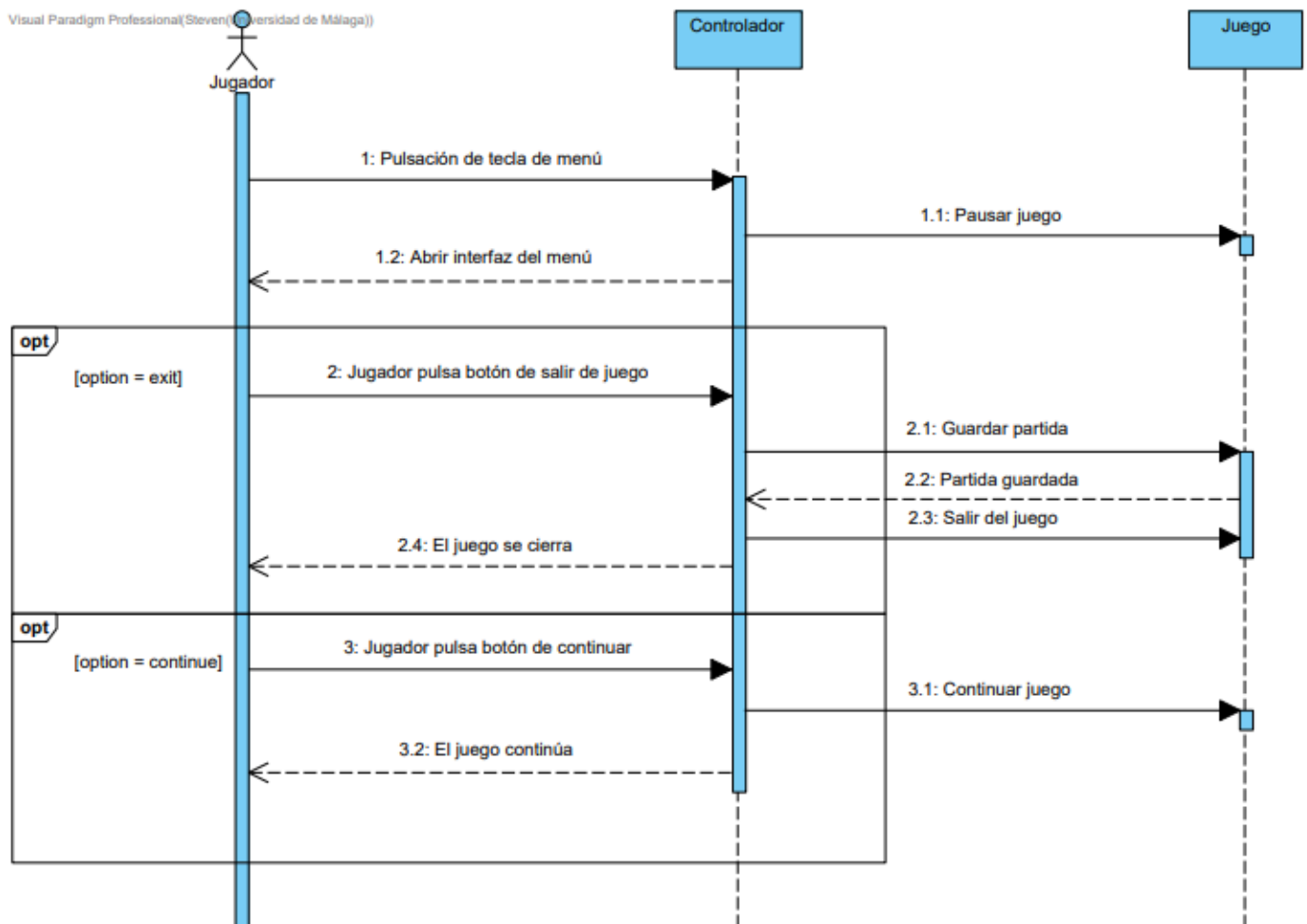


8. Desbloquear una parte del mapa: el jugador llega a una zona nueva del mapa. Aquí obtendrá una misión para poder desbloquear la nueva zona si no la tenía antes. Una vez la complete podrá acceder a la parte desbloqueada.





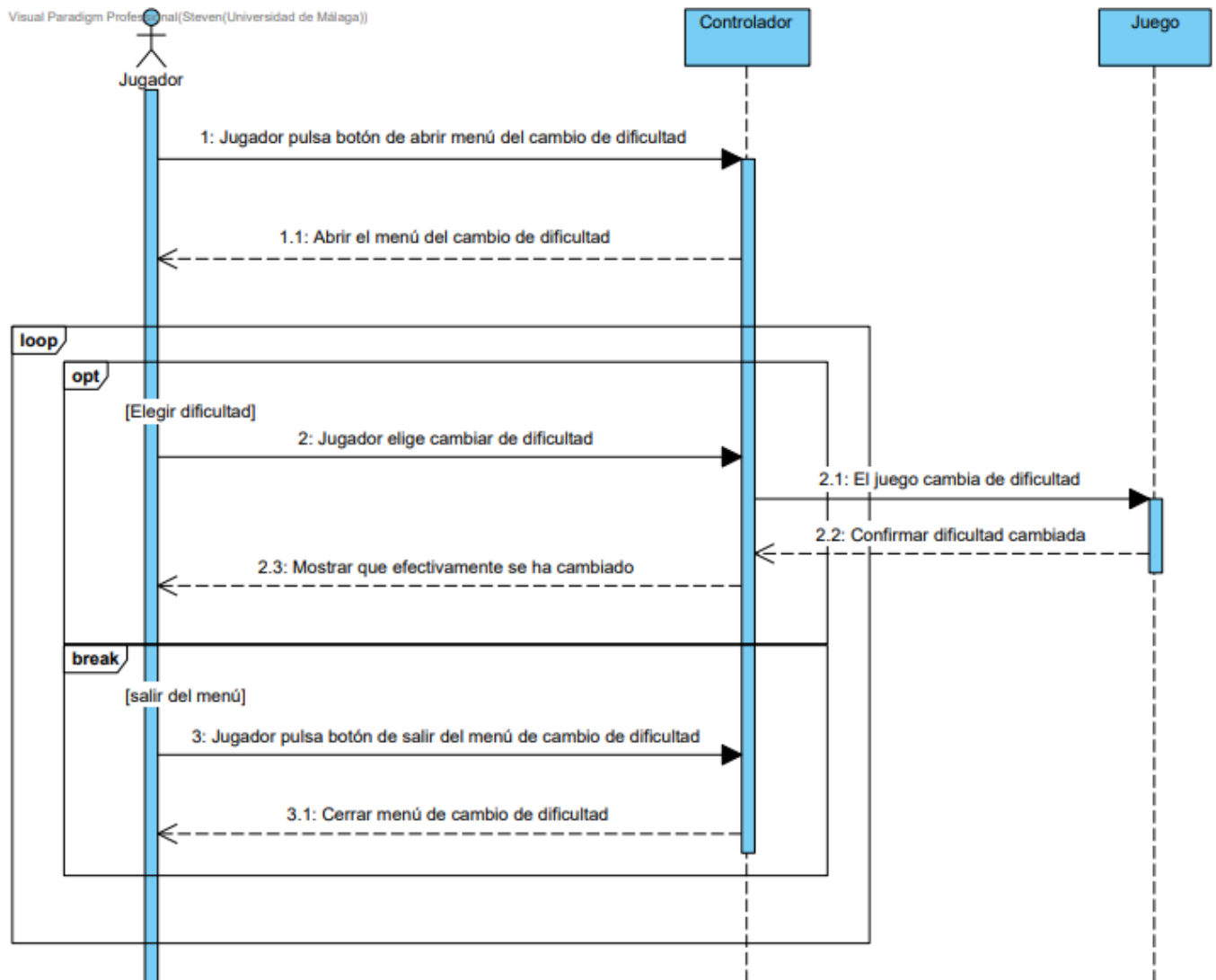
9. Abrir menú principal: el usuario puede pulsar la tecla del menú principal. El juego se pausará y lo llevará al menú principal donde puede elegir salir del juego (el juego se cierra), o seguir jugando.







10. Cambiar dificultad: el usuario también podrá pulsar un botón para abrir el menú de cambio de dificultad. Una vez abierto podrá elegir cambiar la dificultad a su gusto hasta que le dé al botón de salirse, por lo que el menú se cerrará.





## Sección 9:

Por el momento hemos usado whatsapp para comunicarnos y google docs para la elaboración de este mismo documento, además de GitHub para controlar los diferentes documentos requeridos por las entregas, Trello para la planificación del proyecto y Visual Paradigm para los requisitos.

En cuanto a la herramienta que emplearemos para el código en sí mismo (para elaborar la demo del producto), utilizaremos LibGDX con Eclipse.