



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):	https://github.com/JuanGonUMA/Proyecto-IIS
Espacio Trello (url):	https://trello.com/invite/b/1IRL1cra/ATTI7a3bdf3abf0f74fdc63f116521b8a63cE6B4EDE2/rpg-uma-studios
Memoria del proyecto (Office 365 o Google Docs url):	https://docs.google.com/document/d/1fovfAouLURRqUzD0CcRxI9wtmG2IF--xqf4KM0dquMo/edit?usp=sharing

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 5: 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.
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
Planificación	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
Personal	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
Riesgo de escritura	Conflicto en el guardado en Github	Media	Leve	Resolver conflicto en local y después ya actualizarlo en repositorio
Eliminación de archivo	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
Falta de comunicación	Falta de comunicación entre dos o más partes del equipo	Baja	Catastrófico	Una buena planificación y chequeos constantes debería de mitigar este problema
Problema de escalabilidad	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
Cambio de opinión de algún profesor	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

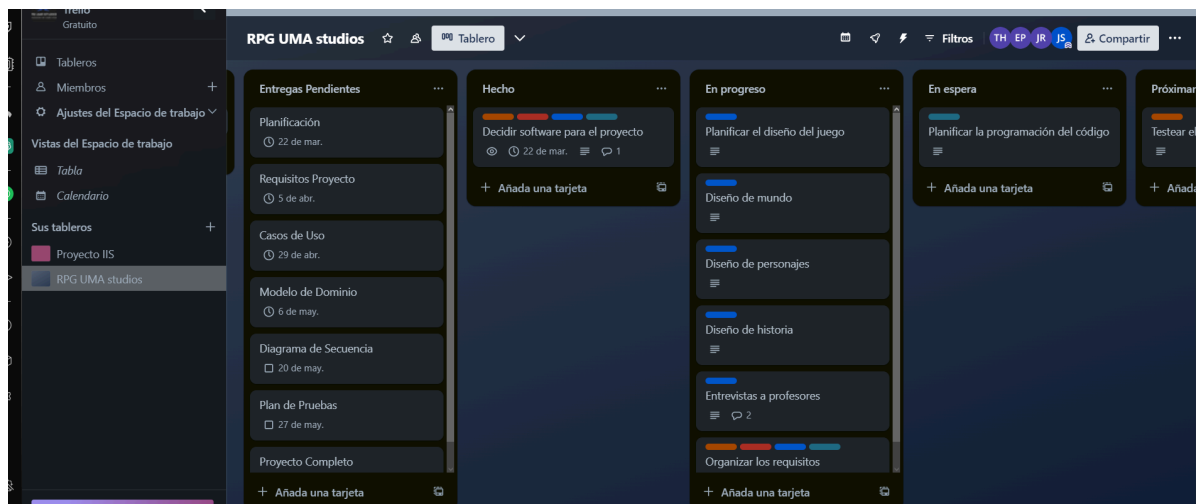


Sección 4:

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.





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.

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.

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.

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.

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 npc's (no 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.

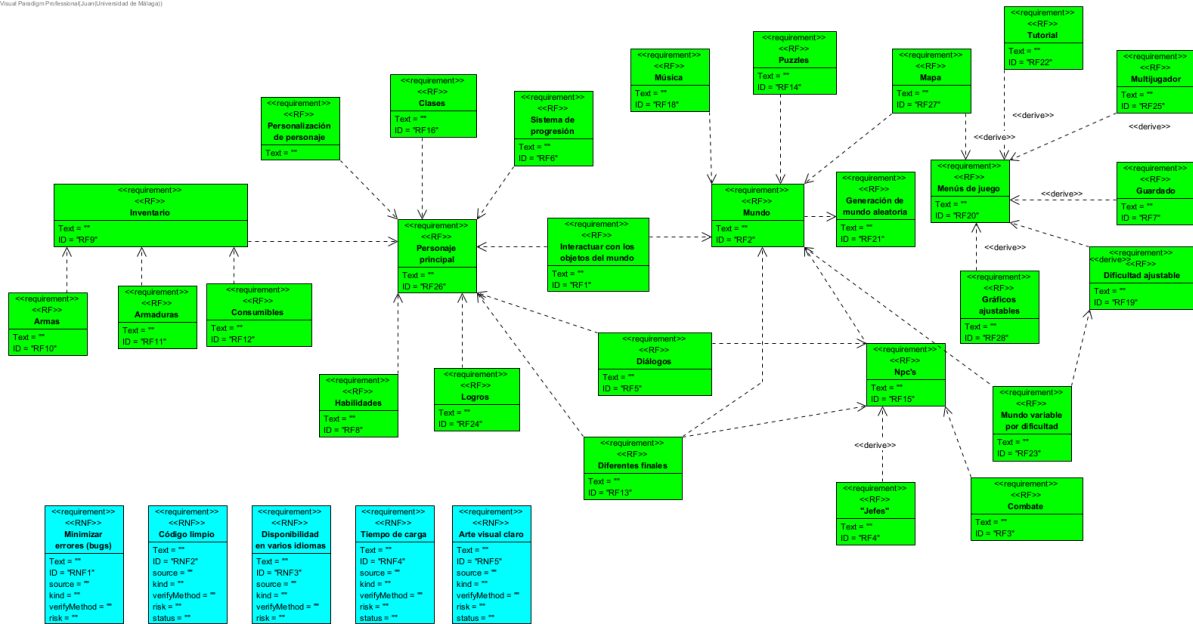
RNF5: Arte visual claro

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 6:

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 GameMaker.