

FLEETSTUDIOS



Logo original hecho para el proyecto

Nombre del grupo	Flota Studios
Titulación	Titulados en la Escuela Técnica de Informática
Grupo de la titulación	A Computadores D Informática
Grupo reducido	GR1
Numero de grupo	2
Repositorio Github (url)	https://github.com/ocxpke/FleetStudios
Espacio Trello (url)	https://trello.com/invite/b/dhYyIAOf/ATTIfc8a9ecc386642ca4090a9e62ec1592689B22737/gamedevelopment
Memoria del proyecto	https://1drv.ms/w/s!AiAMSw7wbdNogfsYbCG4ct8-rt8wMg?e=GWvBkt

MIEMBROS

N.º de miembro	Nombre	Apellidos	Correo electrónico UMA
1	Clara	León Becerra	clari6260@gmail.com
2	Francisco	León Jaime	duendeeditorial@uma.es
3	José Antonio	Ramírez Girón	joseramirez@uma.com
4	Álvaro	Rodríguez Samper	alroper04@gmail.com
5	Andrea	Soler Ortega	061083213@uma.es
6	Regina	Velasco Soto	rvelasco@uma.es

TABLA DE CONTENIDOS

-Sección 1: Introducción. Hay que explicar el problema que se intenta solucionar con el sistema software a desarrollar.

El objetivo será desarrollar una versión del conocido juego de hundir la flota, incorporando nuevas características que ofrezcan una versión redefinida y más completa.

Se ha podido observar que, con el auge de los videojuegos, al ser estos más llamativos y en constante renovación, este tipo de juegos de mesa han quedado “obsoletos” pues son juegos lineales y repetitivos que nunca consiguen sorprender con alguna novedad que enganche al jugador.

Tampoco existe ninguna plataforma líder para llevar al online este clásico juego.

Nosotros buscamos aprovechar esta oportunidad y sacar un proyecto que esperemos que triunfe en el mercado gracias a su constante soporte y actualizaciones, novedades y comunidad que lo apoye. Nuevas modalidades (diferentes tableros), modo campaña, diferentes personalizaciones (mapas, skins, ...), y una gran variedad de nuevas e impresionantes armas que se podrán obtener a través de jugar, ir obteniendo recompensas y superando logros.

-Sección 2: Roles. Indicar los roles que tiene cada integrante del equipo. Cada uno debe tener dos roles, y cada rol debe estar cubierto, al menos, por dos compañeros. [Recordad que tener un rol significa ser responsable de esa tarea, pero todos los integrantes del equipo deben saber de todas las tareas.]

CONCEPTO O IDEA: Todos

DISEÑO: Álvaro, Regina y Clara

PLANIFICACIÓN: Paco, Andrea, Álvaro

PROGRAMACIÓN: José Antonio, Álvaro

ILUSTRACIÓN: Andrea, Clara, Regina

PRUEBA Y DEPURACIÓN: Todos

BANDA SONORA: José Antonio, Paco

REDACCIÓN: Regina, Paco

- **Sección 3: Gestión del Riesgo.** Hay que identificar los riesgos del proyecto y, para cada uno, indicar: tipo de riesgo, descripción del riesgo, probabilidad (muy baja, baja, moderada, alta o muy alta), efectos del riesgo (catastrófico, serio, tolerable o insignificante) y estrategia para mitigarlo.

RIESGO 1:

Riesgo: Competencia directa contra los pioneros del juego.

Descripción del riesgo: Estamos entrando directamente en la zona de confort de una empresa que ya lleva tiempo fija y con el control en esta parte del sector de dicho producto por ende hay mucho por remar en contra.

Probabilidad: Muy alta.

Efectos del riesgo: Catastróficos.

Estrategia para mitigarlo: Aprovechar las ideas revolucionarias que queremos implementar y no dejar el juego obsoleto y sin actualizaciones como nuestros competidores.

RIESGO 2:

Riesgo: Incumplimiento de los plazos.

Descripción del riesgo: No ser capaces de cumplir los plazos pactados con los inversores y anunciados a los clientes finales.

Probabilidad: Baja.

Efectos del riesgo: Catastróficos.

Estrategia para mitigarlo: Establecer puntos claves en el proyecto junto a una fecha asociada e ir completando estos “checkpoints” con tiempo y planificación.

RIESGO 3:

Descripción del riesgo: Robo de datos de nuestros clientes.

Probabilidad: Media.

Efectos del riesgo: Catastróficos.

Estrategia para mitigarlo: Establecer un plan de ciberseguridad que identifique riesgos y aporte soluciones para evitarlo.

RIESGO 4:

Descripción del riesgo: Un integrante se sale del proyecto antes de finalizarlo.

Probabilidad: Baja.

Efectos del riesgo: Dificultad para encontrar con rapidez a la persona idónea. Tolerables.

Estrategia para mitigarlo: Redistribución del trabajo. Activar el protocolo comunicado en el contrato de trabajo en grupo.

RIESGO 5:

Descripción del riesgo: El cliente requiere un cambio que provoca un gran rediseño.

Probabilidad: Moderada.

Efectos del riesgo: Serios.

Estrategia para mitigarlo: Comunicarnos constantemente con el cliente para que se mantenga informado y se encuentre de acuerdo con la dirección del proyecto y su visión.

RIESGO 6:

Descripción del riesgo: Personal clave con los permisos del repositorio/documento está enfermo o no está disponible.

Probabilidad: Muy baja.

Efectos del riesgo: Tolerables.

Estrategia para mitigarlo: Asegurarnos desde el inicio que todos los miembros podemos acceder y modificar cualquier plataforma clave.

RIESGO 7:

Descripción del riesgo: La idea inicial del proyecto no permite la suficiente expansión.

Probabilidad: Moderada.

Efectos del riesgo: Serios.

Estrategia para mitigarlo: Buscar funciones que no tengan necesariamente que ver directamente con el tema principal del proyecto, pero permitan fusionarse.

RIESGO 8:

Descripción del riesgo: Se subestimó inicialmente la carga de trabajo que supone el desarrollo software que queríamos plantear.

Probabilidad: Media.

Efectos del riesgo: No cumplir plazos de entrega. Catastróficos.

Estrategia para mitigarlo: Realizar nueva valoración sobre cargas de trabajo, asignación y redefinición de plazos.

RIESGO 9:

Descripción del riesgo: La base de datos no es capaz de procesar el suficiente volumen de transacciones por segundo.

Probabilidad: Baja.

Efectos del riesgo: Serios.

Estrategia para mitigarlo: Actualizar la base de datos a un plan que permita más velocidad.

RIESGO 10:

Descripción del riesgo: Reestructuración de la empresa por problemas financieros provoca cambios directos en la organización del proyecto.

Probabilidad: Baja.

Efectos del riesgo: Reducción del presupuesto inicial. Catastróficos.

Estrategia para mitigarlo: Rediseño de objetivos para adaptarse al nuevo escenario y reasignación de tareas.

RIESGO 11:

Descripción del riesgo: Se cae el servidor de la página que ofrezca el servicio de distribución del proyecto en un momento crítico.

Probabilidad: Baja.

Efectos del riesgo: Interrupción del servicio. Catastróficos.

Estrategia para mitigarlo: Ir siempre con un margen de tiempo a la hora de la entrega y preparación del proyecto.

RIESGO 12:

Descripción del riesgo: A la hora de actualizar el código en el repositorio de GitHub encontrarse con un problema de incompatibilidad del código (debido a dos miembros trabajando al mismo tiempo) que no sepamos solucionar.

Probabilidad: Baja.

Efectos del riesgo: Bajos.

Estrategia para mitigarlo: Intentar no trabajar en el mismo instante desde dos ordenadores diferentes. En caso de suceder, que discutan los integrantes los beneficios de los cambios con el fin de decantarse por una implementación u otra.

RIESGO 13:

Descripción del riesgo: Diferencias de opinión entre los miembros del grupo a la hora de implementar o diseñar el proyecto.

Probabilidad: Moderada.

Efectos del riesgo: Tolerables.

Estrategia para mitigarlo: Mantener la comunicación en todo momento para que todos se sientan cómodos con la dirección en la que se encamina el proyecto. En caso de suceder, resolver la cuestión por medios democráticos.

RIESGO 14:

Descripción del riesgo: El desarrollo usando Eclipse puede plantear problemas sobre configuración, compatibilidad de complementos, gestión de dependencias y mantenimiento que pudieran afectar al rendimiento.

Probabilidad: Baja.

Efectos del riesgo: Tolerables

Estrategia para mitigarlo: familiarizarse con las mejores prácticas de Eclipse, mantener un entorno de desarrollo bien configurado y actualizado, y asegurarse de que cualquier complemento utilizado sea compatible y esté bien mantenido.

RIESGO 15:

Descripción del riesgo: Problemas financieros a la hora de expandir la velocidad de transacciones por segundo de la base de datos o a la hora de contratar un plan necesario en el software utilizado

Probabilidad: Baja.

Efectos del riesgo: Serios.

Estrategia para mitigarlo: Incluir en la planificación la posible necesidad de estas expansiones para poder presupuestarlo de antemano y realizar las gestiones convenientes para la obtención de préstamos.

RIESGO 16:

Descripción del riesgo: El índice de fallos es mayor al esperado

Probabilidad: Moderado.

Efectos del riesgo: Tolerable.

Estrategia para mitigarlo: Reestructurar la planificación o alternatively buscar una forma de sustituir la implementación de esa herramienta por otra.

- **Sección 4: Planificación.** Se debe indicar el modelo de proceso software elegido (y por qué se ha elegido) y pegar capturas de pantalla de los tableros Trello con las tareas identificadas y su organización. Si se han usado módulos (*Power-Ups*), pegar también capturas de pantalla con lo que muestran. Explicad brevemente las tareas.

Hemos decidido implementar una metodología tipo 'agile' por ser la que mejor se ajusta a nuestras necesidades, por brindarnos la posibilidad de adaptación a pequeños cambios que surjan durante el desarrollo del proyecto, además de la libertad que nos ofrece al explotar las fortalezas individuales de cada miembro.

Analizando las metodologías 'agile', nos hemos decantado por la metodología *SCRUM*, por ser una metodología ampliamente usada en el sector tecnológico a la hora de desarrollar software. Asimismo, nos permite desarrollar proyectos de un tamaño pequeño/mediano, como viene a ser el propuesto por *FLEETSTUDIOS*.

Al aplicar la metodología SCRUM a nuestro proyecto tuvimos que aplicar los roles ya predefinidos dentro de *SCRUM*:

Product Owner --> Andrea Soler Ortega.

Scrum Master --> Jose Ramírez Girón.

Team --> Clara León Becerra, Francisco León Jaime, Álvaro Rodríguez Samper, Regina Velasco Soto.



El *product owner* se pondrá en contacto con el cliente, en este caso el profesor, el cual transmitirá qué espera del proyecto a desarrollar, aspiración e ideas a implementar. El *product owner* será el encargado de generar el **product backlog**, que será necesario para transmitir la idea del cliente al equipo.

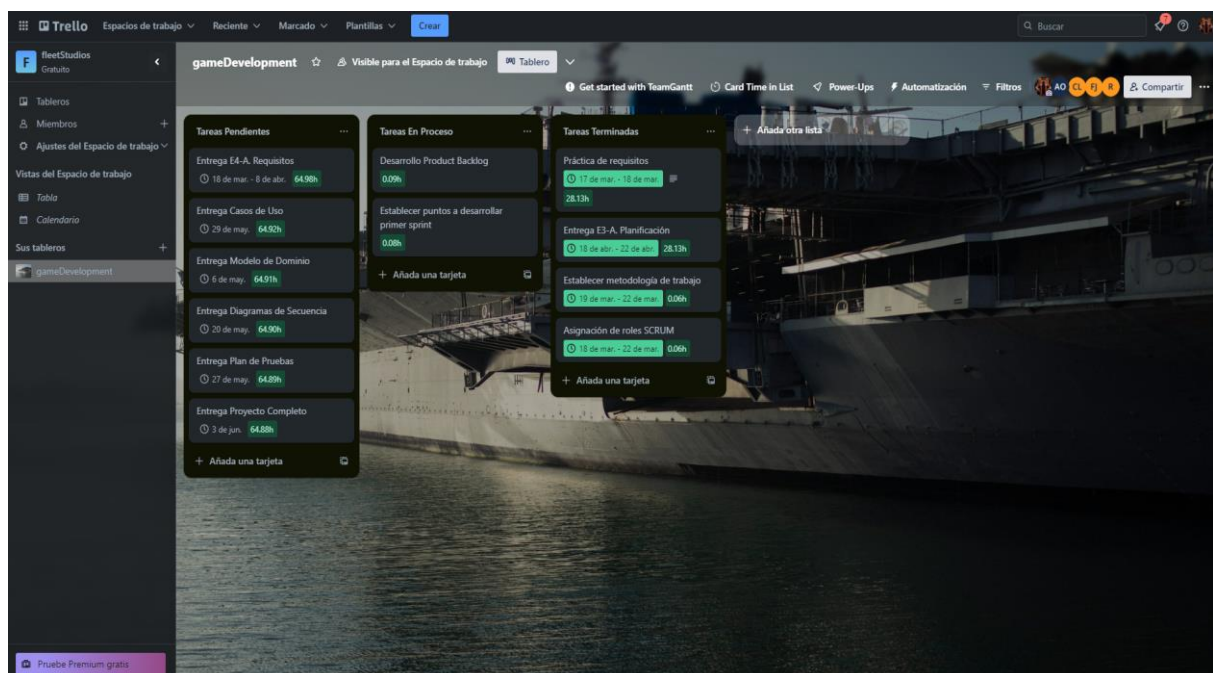
Una vez transmitida la idea, establecidos los puntos a desarrollar, componentes necesarios y establecido el orden de evolución de la aplicación, en resumen, una vez

organizado el desarrollo del proyecto, nos tocará establecer los **sprints** los cuales serán de 1 semana. Además, al final de cada uno, el scrum master se encargará de tener una pequeña reunión con cada miembro del *team* para que puedan compartir sus reflexiones acerca del desarrollo del producto, y que mejoras implementaría. Aquí, en los **sprints**, cada miembro del equipo escogerá una tarea dependiendo de sus fortalezas. Cabe recalcar que, durante los sprints, las reuniones con el cliente pueden tener lugar y generar cambios en el proyecto.

Al finalizar cada **sprint**, se generará el **sprint backlog** que será desarrollado por el *scrum master*, donde se recogerán los siguientes puntos y requisitos a desarrollar por el equipo para el siguiente **sprint**.

Al finalizar el proyecto se tendrá un último **sprint review** y se generará un **burndown chart** para ver el desempeño del equipo a lo largo del desarrollo del proyecto. Y, por último, una charla con el cliente (profesor) para mostrar el resultado final del proyecto.

En conclusión, por las razones definidas anteriormente y teniendo en cuenta los diferentes procesos software existentes, creemos que este es el que mejor se adapta al nuestro.



En el momento de la realización del tablero, solo tenemos las entregas del proyecto según están indicadas en la planificación de la asignatura. Según vayamos avanzando se irán añadiendo nuevas tareas.



Se han habilitado estos *power-ups* para ayudar a la hora del desarrollo del proyecto. A medida que se vaya avanzando en el proyecto se generarán y se aplicaran más reglas al tablero y se habilitarán los *power-ups* necesarios.

Las tareas expuestas en el tablero se describirán en 3 tablas, las cuales describen su estado. A estas, se le añadirán más tablas durante el desarrollo del proyecto.

- Sección 5: Herramientas software usadas durante la realización del proyecto. Indicar las herramientas software que se han usado para desarrollar el proyecto hasta la fecha. Cualquier tipo de herramienta: de comunicación, trabajo colaborativo, elaboración de documentos, etc.

JAVA - JAVA SWING – ECLIPSE – MYSQL – SQL – TRELLO – GIT – GITHUB -
PHOTOSHOP – WHATSAPP – DISCORD - WORD