Grupo 5

inDvelopers

Plan de dirección del proyecto



Registro de cambios

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Fecha | Responsables | Descripción |
| **1.0** | 14/03/2021 | Fernando Wals Rodriguez | Primera versión del plan de dirección |
| **1.1** | 25/03/2021 | Fernando Wals Rodriguez | Se han añadido descripción a las métricas de calidad, se han añadido nuevas métricas de calidad y de rendimiento. Se han añadido nuevas tareas, el cronograma se ha replanificado y se encuentra en un pdf a parte |
| **1.2** | 02/04/21 | Fernando Wals Rodriguez | Se ha replanificado el plan de dirección, añadiendo nuevas tareas y modificando el cronograma |
| **1.3** | 21/04/21 | Fernando Wals Rodriguez | Se ha replanificado el plan para el Sprint 3 |
| **1.4** | 24/04/21 | Fernando Wals Rodriguez | Se ha replanificado el plan para la segunda semana del Sprint 3. |

Datos del proyecto

|  |  |
| --- | --- |
|  |  |
| **Nombre** | inDvelopers |
| **Fecha** | 14/03/2021 |
| **Patrocinador** | Grupo 5 |
| **Project Manager** | Fernando Wals Rodriguez |
| **Grupo** | G5 |

Índice

[1. Información sobre el plan 5](#_Toc66727966)

[1.1. Plan de ejecución del proyecto 5](#_Toc66727967)

[1.2. Procesos y herramientas de control 5](#_Toc66727968)

[1.3. Ciclo de vida del proyecto 5](#_Toc66727969)

[1.4. Control de cambios 5](#_Toc66727970)

[1.5. Gestión de la configuración 6](#_Toc66727971)

[1.6. Introducción 6](#_Toc66727972)

[2. Plan de gestión del alcance 7](#_Toc66727973)

[2.1. Proceso de elaboración del enunciado del alcance 7](#_Toc66727974)

[2.2. Creación de la EDT/WBS 7](#_Toc66727975)

[2.3. Aprobación de la línea base del alcance 7](#_Toc66727976)

[3. Enunciado del alcance 8](#_Toc66727977)

[3.1. Descripción del alcance 8](#_Toc66727978)

[3.2. Entregables 9](#_Toc66727979)

[4. Plan de gestión de requisitos 10](#_Toc66727980)

[5. Documento de requisitos 11](#_Toc66727981)

[5.1. Casos de uso 13](#_Toc66727982)

[5.2 Diagrama UML 15](#_Toc66727983)

[6.EDT 16](#_Toc66727984)

[6.1 Lista de actividades 16](#_Toc66727985)

[7. Cronograma 20](#_Toc66727986)

[8. Plan de gestión de costes 21](#_Toc66727987)

[9. Estimación de costes 24](#_Toc66727988)

[9.1. Coste de personal 24](#_Toc66727989)

[9.2. Coste de equipo 25](#_Toc66727990)

[9.3. Riesgos iniciales 25](#_Toc66727991)

[9.4. Fondo de Riesgo 26](#_Toc66727992)

[9.5. Costes de Recursos 27](#_Toc66727993)

[10. Presupuesto final 28](#_Toc66727994)

[10. Análisis de metodología, calidad y rendimiento 29](#_Toc66727995)

[10.1. Análisis de metodología 29](#_Toc66727996)

[10.2. Análisis de calidad 29](#_Toc66727997)

[10.2.1 Métricas de calidad 29](#_Toc66727998)

[10.3. Análisis de rendimiento 33](#_Toc66727999)

[11. Plan de gestión de riesgos 34](#_Toc66728000)

[12. Plan de comunicaciones 36](#_Toc66728001)

[12.1. Interesados 36](#_Toc66728002)

[12.2. Información que comunicar 36](#_Toc66728003)

[12.3. Reuniones 36](#_Toc66728004)

[12.4. Reparto por subgrupos e intercomunicaciones 36](#_Toc66728005)

[12.5. Tecnologías 36](#_Toc66728006)

[12.6. Factores ambientales 37](#_Toc66728007)

[13. Anexo 38](#_Toc66728008)

[13.1. Equipo de trabajo 38](#_Toc66728009)

[13.1.1. Organigrama 38](#_Toc66728010)

[13.1.2. Características del equipo 39](#_Toc66728011)

[13.2. Estudio de las competencias 41](#_Toc66728012)

[13.2.1. Que nos diferencia del resto 46](#_Toc66728013)

[13.2.2. Diagrama DAFO 46](#_Toc66728014)

[13.3. Charter goal 48](#_Toc66728015)

[13.4. Actas de reunión durante la planificación 51](#_Toc66728016)

[13.4.1. Acta 0 51](#_Toc66728017)

[13.4.2. Acta 1 54](#_Toc66728018)

[13.4.3. Acta 2 56](#_Toc66728019)

[13.4.4. Acta 3 58](#_Toc66728020)

[13.4.5. Acta 4 60](#_Toc66728021)

[13.5. Tecnologías usadas, ALM 62](#_Toc66728022)

[13.5.1. Justificación 62](#_Toc66728023)

[13.5.2. Análisis de despliegues 63](#_Toc66728024)

[13.6. Respuestas de los formularios de nuestros usuarios piloto 64](#_Toc66728025)

[13.7. Tabla de beneficios 68](#_Toc66728026)

# 1. Información sobre el plan

## 1.1. Plan de ejecución del proyecto

El equipo va a estar organizado por un Project Manager, el cual colocará las tareas para cada iteración, se encargará de la supervisión del trabajo y realizará todas las tareas que tenga asignadas. Los programadores se encargarán de realizar todas las tareas que tengan asignadas. Todo lo que se implemente deberá ser testeado de las distintas formas (probar la aplicación, crear los distintos tests). Cada semana se harán dos reuniones para hablar y comentar sobre el trabajo realizado, lo que se hará y todos los problemas que surjan. Un miembro se encargará de escribir un acta de todo lo que se ha tratado en una reunión. Llegado el final de la iteración, se desplegará la aplicación en la nube y el Project Manager se encargará de prepararlo todo para su entrega.

## 1.2. Procesos y herramientas de control

El Project Manager se encargará de tomar las decisiones finales y aceptar o rechazar los cambios del proyecto. Para la dirección, se emplearán una serie de herramientas como el tablero de kanban, donde se pondrán las tareas y quienes deben hacerlas. En ese caso se utilizará GitHub Projects. Además, permitirá conocer en todo momento el estado del proyecto y de la iteración. También, se gestionará el tiempo gracias al cronograma, que permitirá ver el inicio y fin de las tareas. Mediante Clockify se puede contar el tiempo real que llevará, esto permitirá tomar decisiones sobre cambios y reajustes en las estimaciones.

## 1.3. Ciclo de vida del proyecto

El equipo de desarrollo seguirá algunas pautas de la metodología ágil llamada Scrum, para un control constante sobre el estado del proyecto. Se marcarán 5 iteraciones, las cuales serán: 3 Sprints de 2 semanas y las 2 últimas iteraciones relacionadas con el plan de publicidad, de negocio, presentación final, etcétera. En el fin de estos hitos, se realizará una review del proyecto para mostrar lo que se ha implementado en la iteración, acto seguido se realizará una reunión de retrospectiva. Donde el equipo expondrá que ha ido bien y que no, lo que se puede mejorar y qué cambios se harán en la siguiente iteración para mejorar el rendimiento del equipo. Cada semana se hará lo que se conoce como una “daily-stand up” (2 como se ha dicho), donde cada miembro contará que ha hecho, los problemas que ha tenido y lo que hará. Al inicio de cada iteración se hará una reunión de planning, donde se revisará el documento de planificación del proyecto para saber lo que se debe hacer en la siguiente iteración, acto seguido se establecerán las tareas en el tablero.

## 1.4. Control de cambios

Todo cambio que se quiera realizar deberá ser aceptado previamente por el Project Manager. Se realizará una reunión para proponer los cambios y hablar en grupo sobre todos los cambios, los efectos y consecuencias que conllevarían. Pero como se ha mencionado, será el Project Manager quien tome la decisión. Este se encargará de analizar el cambio y ver si es plausible. Si es aprobado, los responsables de la zona del proyecto que afecta el cambio lo implementarán y se reflejará en todos los elementos para el registro de los cambios, como en los documentos de seguimiento y control.

## 1.5. Gestión de la configuración

Los cambios de documentos que registrarán en los propios documentos en la sección de “registro de cambios”, donde se especifica la versión, fecha del cambio, quien lo ha hecho y una breve descripción. Con respecto a los cambios de código, se realizará un registro de cambios en el drive que utilizará el equipo

## 1.6. Introducción

Este trabajo consiste en la creación de una plataforma de videojuegos para desarrolladores indie, “inDvelopers”. Este proyecto será implementado por el grupo 5 de ISPP mediante las tecnologías de React (como Front-end), Spring (como Back-end) y MongoDB como base de datos. Se trabajará con Github para el control de versiones y Discord para hacer reuniones de grupo. Se empezará el 16 de Marzo y se acabará el 7 de Junio. La razón de este proyecto es para poder superar la asignatura y aprender a trabajar profesionalmente en informática, tal y como lo haría una empresa.

# 2. Plan de gestión del alcance

## 2.1. Proceso de elaboración del enunciado del alcance

El cómo se definirá el alcance para el producto “inDvelopers” será definido siguiendo las recomendaciones del manual de gestión de proyectos de PMBOK. Estará formado por dos apartados, los cuales se detallan a continuación.

En primera instancia tendremos el apartado de “Descripción del alcance”. Este apartado describe con sumo detalle los contenidos, características y funcionalidades que tendrá el proyecto a desarrollar: Sus secciones, tipos de productos, proceso de alquiler de los mismos, tipos de vistas que tendrá (por ejemplo, parte pública y privada de la aplicación). A fin de que se sepa cómo va a estar formado el producto, cómo va a funcionar y cómo los clientes potenciales interactúan con el mismo.

En segunda instancia tendremos el apartado de “Entregables”. Este apartado describe de manera textual todos los entregables que se desarrollarán y se darán a disposición durante el desarrollo del proyecto. Los entregables serán cualquier producto o resultado que se deberán cumplir para completar un proceso o fase del proyecto. Se especificará el número de entregables, las fechas a las que deben estar listos y una descripción detallada de los mismos. Dentro de los entregables entra cualquier resultado complementario como informes, actas y documentación del propio proyecto.

## 2.2. Creación de la EDT/WBS

La EDT se desarrollará siguiendo las descripciones detalladas del apartado “Descripción del alcance”, que será definido en el enunciado del alcance. Estará formada por las distintas iteraciones, de las cuales colgarán los paquetes de trabajo, en la lista de actividades se detalla que actividades forman parte de cada paquete de trabajo.

## 2.3. Aprobación de la línea base del alcance

El alcance del proyecto será aprobado mediante la revisión del documento por parte de los miembros del equipo, donde se revisará el mismo y se decidirá qué partes cambiar. Todos los miembros revisarán el alcance para aportar su opinión sobre el mismo y todas las posibles mejoras o cambios a realizar.

# 3. Enunciado del alcance

## 3.1. Descripción del alcance

“inDvelopers” es un portal dedicado para un grupo de desarrolladores, en concreto los desarrolladores indie o indie developers. Este tipo de desarrolladores se dedican principalmente al desarrollo de videojuegos de forma amateur, sin ninguna formación profesional previa. Tan solo se guían por su creatividad y su curiosidad por aprender sobre el mundo de los videojuegos y el trabajo y conocimiento que ello requiere. Por lo que son personas que en su mayoría son autodidactas y aprenden de todos los lugares que ofrece Internet (y por supuesto de la práctica).

Por ello, “inDvelopers” es una forma de demostrar al mundo y a las empresas tus cualidades como game developer. En esta aplicación, los indie developers podrán darse a conocer y mostrar a la comunidad todos sus trabajos hechos a lo largo del transcurso. Mediante la aplicación, podrás subir información detallada sobre ti: Información de contacto, descripción sobre las tecnologías que sueles emplear para el desarrollo de tus juegos, enlaces de interés y un listado de todos tus videojuegos hechos hasta la fecha. En cada videojuego podremos encontrar: vídeos, imágenes, descripción, requisitos del sistema, críticas y puntuaciones de la comunidad. Y por supuesto la opción de adquirirlo

La aplicación va a estar formada por distintas secciones: “Publicaciones”, “Novedades”, “Juegos”, “FAQs”, “Sobre nosotros” y login/signup

En la sección de publicaciones podemos ver todas las publicaciones que hace la comunidad. Estas publicaciones pueden ser textos informativos, enlaces de interés o imágenes. En esta misma sección se puede hacer Click en “crear publicación” para que el usuario publique algo.

En la sección de Novedades podemos ver los juegos con mejor puntuación de la comunidad. Todos los juegos que suelen tener una puntuación de 4 o más (la media de las puntuaciones de sus críticas) aparecerán en esta sección.

En la sección de “Juegos” podemos ver todo el listado de los juegos creados por la comunidad. Al hacer click en uno de ellos entrarás en los detalles, verás toda su información al detalle. Y tendrás la opción de comprarlo y/o descargarlo. Como usuario, puedes subir tus propios juegos, en la sección de juegos, haciendo click en subir un juego, te enviará al formulario para colgar tu videojuego. Si no eres usuario de pago, tendrás un límite de 5 juegos y estos no pueden tener precio, solo puedes colgar juegos gratuitos. Como propietario del juego, puedes modificarlo y eliminarlo si lo deseas.

En la sección de “FAQs” se pueden ver las preguntas frecuentes que le pueden surgir a los usuarios, junto con sus respectivas respuestas.

En la sección “Sobre nosotros”, nos muestra un párrafo describiendo la aplicación y sobre el equipo que lo ha creado.

Por último, en la sección de login/signup, los usuarios podrán entrar y registrarse. Si ya tienen una cuenta, pueden hacer login introduciendo sus credenciales de usuario. Al hacerlo, se cambiará esta sección por el nombre del usuario y al hacer click entraremos en los detalles de nuestra cuenta. Podremos modificar nuestros datos, ver nuestros juegos, estadísticas personales y acceder a la pestaña de subscripción. Esta pestaña nos permitirá pagar por una subscripción mensual que no permitirá colgar más juegos y con un precio. Hay dos subscripciones: La primera, por 7.99€ al mes te permite colgar 5 juegos con precio y la segunda por 11.99€ puedes colgar 10 juegos con precio. También habrá usuarios administradores, los cuales tendrán acceso a todo, a un panel de estadísticas de administrador y podrán eliminar juegos y usuarios.

## 3.2. Entregables

Habrá un total de 5 entregables al empezar el proyecto:

El sprint 1 se entregará el 5/04/21. En este entregable se espera todos los casos de uso core que se hayan definido, también se espera que la aplicación esté testeada y desplegada.

El sprint 2 se entregará el 26/04/21. En este entregable se espera tener implementado el sistema de pago y que los juegos estén almacenados en la nube. También se espera que la aplicación esté testeada y desplegada.

El sprint 3 se entregará el 10/05/21. En este entregable se espera que la aplicación haya sido mejorada y que todos los posibles fallos sean resueltos. También se espera que la aplicación esté testeada y desplegada.

La preparación del lanzamiento se entregará el 24/05/21. En este entregable se espera que la aplicación esté lista para su presentación al mundo y que se haya creado una campaña publicitaria junto con un plan de lanzamiento.

Por último, el plan de negocio se entregará el 7/06/21. Aquí se especificará la idea de negocio, planificación técnica y financiera, análisis SWOT, plan ROI.

# 4. Plan de gestión de requisitos

Priorización de requisitos

|  |  |
| --- | --- |
| Nivel | DESCRIPCIÓN |
| Alto | Imprescindible para la finalización del proyecto |
| Medio | No es imprescindible para la finalización pero si para un correcto desarrollo del proyecto |
| Bajo | No es necesario para la finalización del proyecto |

Tabla :Priorización de requisitos

Leyenda

|  |  |
| --- | --- |
| Tipo | Formato |
| Requisitos de información | RI-X |
| Requisitos Funcionales | RF-X |
| Requisitos No Funcionales | RNF-X |
| Requisitos de proyecto | RP-X |
| Requisitos de calidad | RC-X |
| Reglas de negocio | RN-X |

Tabla : Leyenda de requisitos

# 5. Documento de requisitos

Requisitos

|  |  |  |
| --- | --- | --- |
| ID | DESCRIPCIÓN | PRIORIDAD |
| RP-01 | Desarrollo acorde al PMBOK | Media |
| RP-02 | Índice en la planificación | Media |
| RP-03 | Control del tiempo en Clockify | Media |
| RP-04 | Control de asignación de actividades en Trello | Media |
| RP-05 | Realizar actas de reunión recogiendo los temas tratados en cada una | Media |
| RP-06 | Los documentos deben tener un lenguaje y estructura adecuada | Media |
| RP-07 | Análisis de competidores exhaustivo | Alta |
| RP-08 | Recopilación de usuarios piloto | Alta |
| RP-09 | Realizar encuestas a usuarios piloto | Alta |
| RP-10 | Análisis optimista y pesimista de las ganancias | Alta |
| RP-11 | Charter goal incluyendo los compromisos a los que debe adherirse el equipo | Media |
| RP-12 | Análisis para la elección de tecnologías | Alta |
| RP-13 | Se realizará un plan de marketing para dar a conocer la plataforma | Alta |
| RN-01 | Las contraseñas deberán tener un mínimo de 8 caracteres alfanuméricos | Alta |
| RN-02 | Únicamente un administrador podrá crear otro administrador | Alta |
| RN-03 | El inDveloper no podrá recibir ingresos por un juego si no se es premium | Alta |
| RN-04 | El inDveloper no podrá publicar más de 5 juegos si no es premium | Alta |
| RI-01 | El sistema deberá almacenar información sobre los inDvelopers | Alta |
| RI-02 | El sistema deberá almacenar información sobre los juegos | Alta |
| RI-03 | El sistema deberá almacenar información sobre las reseñas | Alta |
| RI-04 | El sistema deberá almacenar información sobre las publicaciones | Alta |
| RI-05 | El sistema deberá almacenar las imágenes asociadas al juego | Alta |
| RF-01 | El sistema debe redirigir a la plataforma de pago para asegurar seguridad | Alta |
| RF-02 | CRUD de juego | Alta |
| RF-03 | El inDveloper podrá dejar críticas en los juegos | Alta |
| RF-04 | Los clientes podrán registrarse en el sistema | Alta |
| RF-05 | Crear vista Sobre Nosotros | Media |
| RF-06 | El sistema renovará automáticamente la suscripción del cliente si este no la cancela | Alta |
| RF-07 | Los clientes podrán darse de baja en el sistema | Alta |
| RF-08 | El usuario administrador podrá acceder a una lista de los juegos disponibles | Alta |
| RF-09 | El usuario administrador podrá acceder a una lista de los inDvelopers registrados | Alta |
| RNF-01 | El sistema deberá estar disponible 24/7 | Alta |
| RNF-02 | Las imágenes deben tener buena resolución para mejorar la experiencia de usuario | Alta |
| RNF-03 | La moneda utilizada será el euro | Alta |
| RC-01 | Crear vista FAQs | Media |
| RC-02 | El sistema deberá resolver peticiones en un tiempo de respuesta mínimo | Alta |
| RC-03 | Utilización de una paleta de colores para lograr imagen de marca | Alta |
| RC-04 | El diseño de la página debe ser intuitivo | Alta |
| RC-05 | El sistema debe asegurar la confidencialidad de los datos de los clientes | Alta |

Tabla : Catálogo de requisitos

## 5.1. Casos de uso

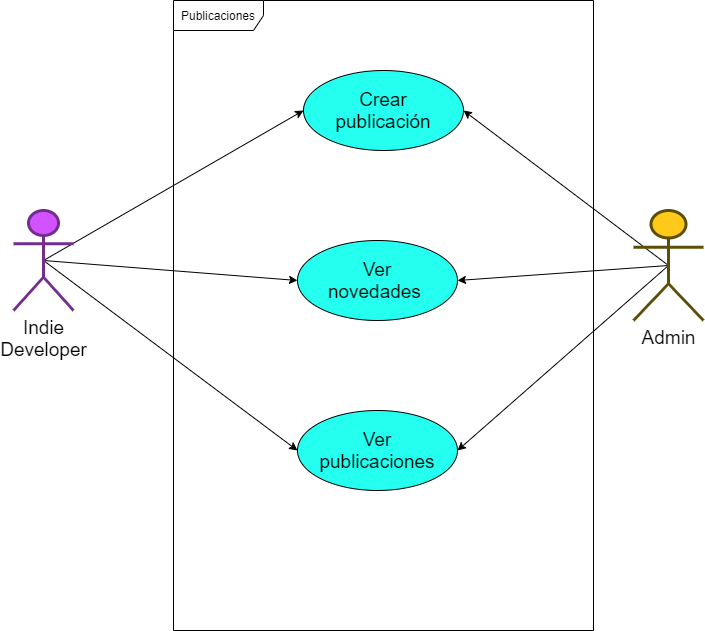


Ilustración :Caso de uso de publicaciones

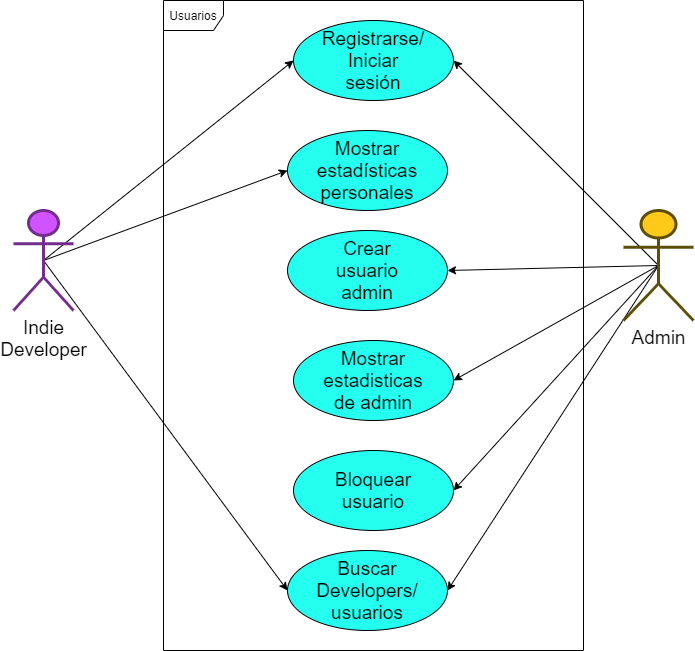


Ilustración :Caso de uso de usuarios

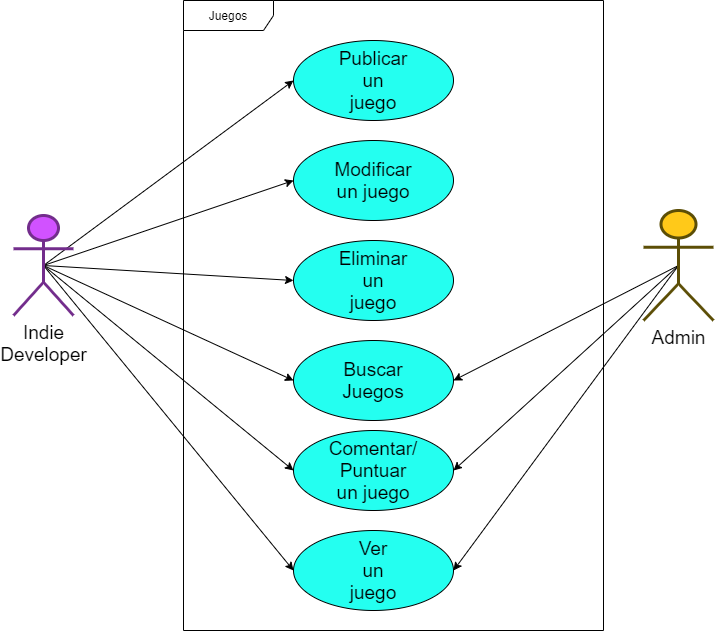
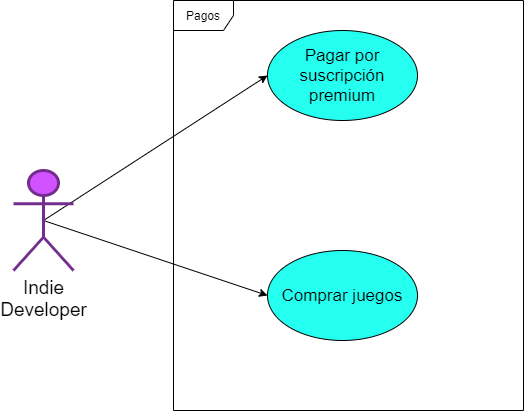


Ilustración : Caso de uso Pagos

Ilustración :Caso de uso Juegos

## 5.2 Diagrama UML

Ilustración : Diagrama UML de inDvelopers

# 6.EDT

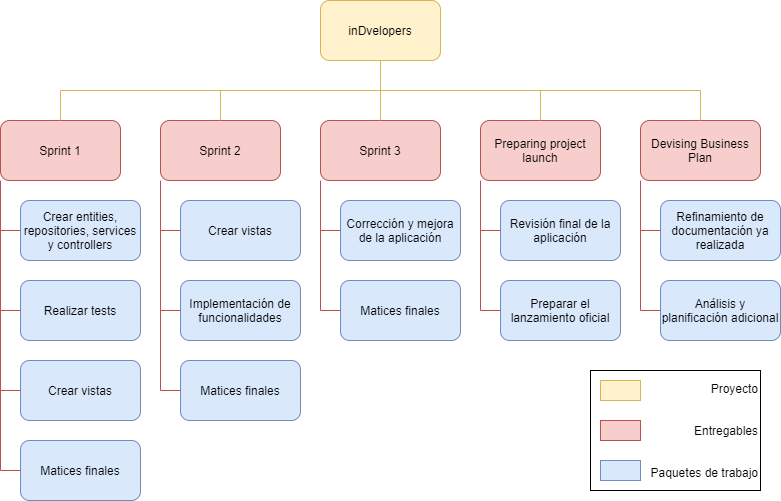


Ilustración : Estructura de desglose de trabajo

## 6.1 Lista de actividades

A continuación se definen en una tabla las iteraciones referentes a los entregables, los cuales se dividen en paquetes de trabajo, y estos a su vez en actividades que han de realizarse para el desarrollo óptimo de la web de comercio electrónico, ajustándose a los estándares de calidad propuestos:

|  |
| --- |
| **Sprint 1**   * Crear entities, repositories, services y controllers   + Crear entity, repository, service y controller de InDveloper   + Crear entity, repository, service y controller de Publication   + Crear entity, repository, service y controller de Game   + Crear entity, repository, service y controller de Review(Crítica) * Realizar tests   + Realizar tests de InDveloper   + Realizar tests de Publication   + Realizar tests de Game   + Realizar tests de Review * Crear vistas   + Crear el Header y Footer de la aplicación   + Crear la vista de registro de nuevo usuario   + Crear la vista de inicio de sesión   + Crear la vista de información de usuario   + Crear la vista de listado de juegos   + Crear la vista de crear un juego   + Crear la vista de ver juego   + Crear la vista de modificar juego   + Crear la vista de eliminar juego   + Crear la vista de ver publicaciones   + Crear la vista de crear publicación * Matices finales   + Crear la vista de mensaje de excepción para usuarios   + Crear la vista de crear review a un videojuego   + Crear la vista de listado de reviews en un videojuego   + Generar documentos complementarios durante el sprint   + Desplegar la aplicación en la nube   + Testear la aplicación en el despliegue |
| **Sprint 2**   * Crear vistas   + Crear la vista de novedades   + Crear la vista de FAQs   + Crear la vista de “Mis juegos”   + Crear la vista de “Sobre nosotros”   + Modificar la vista de ver juego para que los admin puedan descargar gratuitamente, borrar el juego y validarlo haciendo click en un check para marcarlo como validado   + Modificar la vista de listar juegos para que solo muestren los juegos verificados por los admin   + Crear la vista de juegos pendientes de revisión para admin   + Crear la vista de registrar administrador   + Crear una vista de listar usuarios siendo admin   + Modificar la vista de pago para que se haga un pago real   + Crear una vista de ver usuario para admin y que estos puedan borrarlo * Implementación de funcionalidades   + Implementar la creación de administrador   + Implementar la funcionalidad de mostrar los juegos pendientes de revisión para los administradores   + Implementar la funcionalidad de mostrar los mejores juegos de la comunidad(novedades)   + Implementar la funcionalidad de borrar juego siendo admin   + Implementar la funcionalidad de desbloquear juego siendo admin   + Implementar la funcionalidad de eliminar usuario siendo admin   + Implementar el método de pago   + Implementar el almacenamiento de los juegos en la nube * Matices finales   + Generar documentos complementarios durante el sprint   + Realizar/Corregir Tests   + Desplegar la aplicación en la nube   + Testear la aplicación en el despliegue |
| **Sprint 3**   * Crear vistas   + Crear la vista de estadísticas de InDveloper (DashBoard de usuario)   + Crear la vista de estadísticas de admin (DashBoard de admin)   + Añadir a la vista de listado de juegos el botón de “following games” para ver los juegos de los usuarios a los que sigues   + Modificar el listado de reviews para que puedas modificar una review que sea tuya.   + Modificar la vista de ver juegos para ver al autor y que tenga un botón para seguirle   + Modificar el listado de juegos para añadirle un buscador por nombre, categoría y precio.   + Modificar el registro de usuario y añadir un campo de fecha de nacimiento.   + Modificar el formulario de crear/ ver juego para ponerle los campos de pegi y categoría.   + Crear una sección de incidencias para que los usuarios puedan mandarlas y que lo vean los admin. Y que los admins tengan un listado para ver las incidencias y borrarlas una vez se hayan resuelto.   + Añadir una sección de términos y condiciones (en la barra de navegación).   + Modificar el registro para que te mande a los términos y condiciones.   + Modificar la vista de user details para poder acceder al listado de usuarios que te siguen y sigues. Para los usuarios que sigues, que puedas dejar de seguirlos   + (Los términos y condiciones deben decir: qué datos vamos a pedir, cuales vamos a usar, por qué les pedimos esos datos, decir que implementa protocolo seguro https, que hay garantía de privacidad, que un admin pueda notificar con el correo de los usuarios a estos si ha habido una brecha de seguridad (se puede especificar que el correo está para eso), que un usuario pueda mandarnos un correo para que le borremos o movamos sus datos y decir cómo vamos a gestionar las incidencias). * Implementación de funcionalidades   + Mejorar el DashBoard para administrador en back end   + Mejorar el DashBoard para usuarios en back end   + Crear un endPoint que devuelva los juegos de los usuarios a los que sigues   + Crear una entidad (repositorio, servicio, controlador) que represente la relación de seguimiento entre usuarios.   + Añadir a juego los atributos de categoría y pegi.   + Añadir al developer fecha de nacimiento.   + Crear un endPoint que busque juegos verificados por nombre, categoría o precio.   + Modificar el findVerified para que si el usuario es menor de edad que solo muestre los juegos que no son +18   + Modificar búsqueda de juego por título y por autor para que use findVerified, en lugar de findAll   + Mejorar el findTop para que muestre los juegos cuya media de puntuaciones de las reviews sea mayor o igual a 4   + Crear test de la entidad que relaciona el seguimiento entre dos usuarios.   + Crear tests de incidencia   + Crear nuevos tests de developer, juego y dashboard.   + Crear un endPoint que devuelva el listado de usuarios que te siguen.   + Crear un endpoint que devuelva el listado de usuarios que sigues   + Crear la entidad incidencia. * Corrección y mejora de la aplicación   + Añadir mejoras en la aplicación   + Corregir errores encontrados en la aplicación * Peticiones de los usuarios piloto   + Mejorar los campos para comprobar que no son Spam y que los admin puedan añadir palabras al diccionario (Front)   + Crear un diccionario que guarde las palabras Spam (Back)   + Mejorar el responsive (Front)   + Refactorizar las queries en Back y que el findByTitle, by category y price sea permitAll en la seguridad (Back)   + Mejorar el campo de la contraseña en el registro (Front)   + Añadir una progress bar y un aviso que indique que el juego está subido (Front)   + CRUD en publicaciones (Back)   + Que un usuario pueda editar, borrar sus publicaciones (Front)   + Que un usuario pueda borrar sus reviews (Front)   + Poner ofertas de videojuegos (Back)   + Que usuario pueda entrar en su juego y aplicar una oferta de x% de descuento (Front)   + Que se pueda recuperar la contraseña si se le ha olvidado al usuario (Front)   + Arreglos en el front(Descripción para el campo de tecnologías en el registro, poner un enlace en el footer que cuando se haga click en el símbolo de react o spring que te envíe a las páginas de las tecnologías, redirigir a /me cuando se modifica el perfil del usuario) * Testear la aplicación en el despliegue * Matices finales   + Generar documentos complementarios durante el sprint   + Realizar/Corregir Tests   + Desplegar la aplicación en la nube   + Testear la aplicación en el despliegue |
| **Preparing project launch**   * Revisión final de la aplicación   + Mejorar la aplicación   + Corregir errores encontrados en la aplicación   + Desplegar la aplicación en la nube   + Testear la aplicación en el despliegue * Preparar el lanzamiento oficial   + Plan de lanzamiento   + Campaña publicitaria |
| **Devising business plan**   * Refinamiento de documentación ya realizada   + Idea de negocio   + Equipo de proyecto   + Competidores * Análisis y planificación adicional   + Análisis SWOT   + Planificación técnica y financiera   + Plan ROI |
|  |

Tabla : Listado de actividades por paquete de trabajo

# 7. Cronograma

El cronograma se encuentra aparte en el drive del equipo y/o adjuntado en las futuras entregas

# 8. Plan de gestión de costes

El plan de gestión de costes se ha de realizar en base a las actividades que se especifican en el cronograma, calculando esos costes en base al tiempo de cada tarea.

Para llevar a cabo un plan de costes acertado debemos antes de nada identificar los tipos de costes plantea nuestro proyecto, los cuales serían

* Coste del personal
* Coste del equipo
* Costes fijos y operativos
* Fondo de riegos

Estimación y presupuesto

Coste del personal

Para el coste del personal se tiene en cuenta el cronograma y las actividades que se llevan a cabo por cada empleado. Así como los gastos derivados de impuestos y todo lo que afecta a calcular el sueldo anual bruto de cada persona.

Cuando se tiene el coste anual bruto, se calcula cuánto debería cobrar por hora dicho trabajador o trabajadora para alcanzar dicho sueldo en horario laboral según su contrato.

La estimación de este coste vendrá dado en formato de tabla como se muestra de ejemplo a continuación:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Recurso | Salario bruto (anual) | Coste total para la empresa | Coste empresa por mes | Coste empresa por hora |
| Elemento 1 | x€ | x€ | x€ | x€/hora |
| Costes totales de personal para el proyecto: | | **x€** | | |

Tabla 5: Esquema de tabla de costes de personal

Coste del equipo

Se establece un coste derivado del cálculo en base al cronograma y duración del proyecto en días naturales, ya que se realiza una división por día del coste de amortización de los activos a cinco años.

La estimación de este coste vendrá dado en formato de tabla como se muestra de ejemplo a continuación:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Recurso | Unidades necesarias | Coste unitario | Amortización diaria durante 5 años | Coste diario | Coste total en proyecto |
| Elemento 1 | x | x€ | x€ | x€ | x€ |
| Costes totales de equipo para el proyecto: | | **x€** | | | |

Tabla 6: Esquema de tabla de costes de equipo

Costes fijos y operativos

Hay que tener en cuenta el resto de variables como la luz, el agua, el suministro de internet, teléfono, seguros, etcétera.

Estos gastos se asumen normalmente en forma de mensualidades, por lo que se divide el gasto por día natural de duración del proyecto en base al cronograma y se obtiene el coste.

La estimación de este coste vendrá dado en formato de tabla como se muestra de ejemplo a continuación:

|  |  |  |
| --- | --- | --- |
| Operativo | Coste consumo diario | Coste total para la empresa |
| Elemento 1 | x€ | x€ |
| Costes totales de operativos para el proyecto: | | **x€** |

Tabla 7: Esquema de tabla de costes fijos y operativos

Fondos de riesgo

En este apartado, los riesgos se deben tener en cuenta y por ello se tiene en cuenta el análisis cuantitativo de los riesgos, en el que se determina el gasto en tiempo o dinero de cada uno de los riesgos identificados. Se hará la suma de esos gastos y se obtendrá el fondo para los riesgos. Decir que cuando el gasto se estima en tiempo, hay que multiplicarlo por el salario por hora del equipo en conjunto.

La estimación de este coste vendrá dado en formato de tabla como se muestra de ejemplo a continuación:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Impacto | Recurso  afectado | Probabilidad | Impacto | VE  (Impacto\*Prob) |
| Elemento 1 | - | % | € | x€ |
| Costes totales de recursos para el proyecto: | |  |  | **x€** |

Tabla 8: Esquema de tabla de costes de riesgos

Monitorización y control

Una vez el proyecto ha comenzado, es importante hacer un seguimiento de que los gastos diarios no superen el presupuesto establecido, ya que produciría cambios en el alcance o calidad del proyecto antes de pasar por una pérdida de beneficio para la organización. Es por ello que se debe de controlar un aspecto clave del proceso, los **riesgos**.

Así mismo, se considera más una gestión de riesgos que otro aspecto. Ante una situación indeseada, el presupuesto debe contar con un fondo mínimo basado en la probabilidad de que ocurra tal riesgo, con objeto de amortiguar el impacto económico en la empresa. Y en el momento que ocurre, modificar la estimación de estos fondos para gestionar los recursos en base a las prioridades de la empresa en que no ocurran ciertos riesgos que pondrían en jaque el presupuesto inicial en pos de un beneficio máximo.

# 9. Estimación de costes

En la estimación de costes podremos ver una predicción cercana y clara de cuánto va a costar la realización del proyecto.

La duración del proyecto según nuestro programa de trabajo será de 13 semanas.

## 9.1. Coste de personal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Recurso | Salario bruto (anual) | Coste total para la empresa | Coste empresa por mes | Coste empresa por hora |
| Project manager | 45.939€ | 59.675€ | 4,973€ | 29€/hora |
| Analista/  Programador | 32.337€ | 42.006€ | 3.501€ | 20€/hora |
| Scrum Master/  Programador | 42.026€ | 54.592€ | 4.550€ | 26€/hora |
| Director de marketing | 27.017€ | 35.096€ | 2.925€ | 17€/hora |
| Programador | 19.804€ | 25.726€ | 2.144€ | 13€/hora |
| Costes totales de personal para el proyecto: | | **107.143€** | | |

Tabla 9: Costes de personal

* **Recurso**: Rol del recurso en el equipo de proyecto
* **Salario bruto**: Salario bruto anual del trabajador o trabajadora
* **Coste total para la empresa**: Coste calculado de aplicar el porcentaje de cotizaciones de seguridad social al salario anual bruto
* **Coste empresa por mes**: Coste del recurso para la empresa calculado para un mes
* **Coste empresa por hora**: Coste del recurso para la empresa calculado para una hora, suponiendo 22 días de trabajo al mes y 8 horas al día
* **Costes totales de personal para el proyecto:** Coste total del proyecto con 1 Project manager, 1 analista, 1 Scrum master, 1 director de marketing y 8 programadores.

## 9.2. Coste de equipo

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Recurso | Unidades necesarias | Coste unitario | Amortización diaria durante 5 años | Coste diario | Coste total en proyecto |
| Portatil Dell XPS 13 | 12 | 1500€ | 0,82€ | 10€ | 910€ |
| Monitor | 12 | 100€ | 0,055€ | 0,66€ | 60,66€ |
| Ratón | 12 | 70€ | 0,038€ | 0,47€ | 42,47€ |
| Teclado | 12 | 80€ | 0,044€ | 0,53€ | 48,53€ |
| Impresora | 3 | 100€ | 0,055€ | 0,17€ | 15,16€ |
| Paquete Office | 12 | 16,90€ | - | 6,77€ | 338,4€ |
| Costes totales de equipo para el proyecto: | | **1.416€** | | | |

Tabla 10: Costes de equipo

* **Recurso**: Nombre o modelo del recurso
* **Unidades necesarias:** Unidades del recurso requeridas para el proyecto
* **Coste unitario:** Coste de compra/adquisición de un recurso
* **Amortización diaria durante 5 años:** Costo de amortización del recurso para un día
* **Coste diario:** Coste prorrateado por día del recurso
* **Coste total en proyecto:** Suma de costo del recurso en el proyecto

## 9.3. Riesgos iniciales

Un análisis inicial del proyecto ha llevado a la detección de unos posibles riesgos generales que se ven recogidos en la siguiente tabla:

|  |  |
| --- | --- |
|  | Riesgo |
| R1 | Actualizaciones del software |
| R2 | Baja de personal por coronavirus |
| R3 | Cambio en requisitos o tecnologías |
| R4 | Un competidor desarrolla un producto muy similar. |
| R5 | Pérdida de información o archivos durante el desarrollo del producto |
| R6 | Los canales de comunicación del equipo fallan. |
| R7 | No cumplir con alguna ley referente al proyecto |
| R8 | Una librería usada en el proyecto deja de estar disponible |
| R9 | Aparece un error crítico en una fase avanzada del proyecto |
| R10 | Problemas en despliegue en aplicaciones de condiciones gratuitas |

Tabla 11: Riesgos

## 9.4. Fondo de Riesgo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Impacto | Recurso  afectado | Probabilidad | Impacto | VE  (Impacto\*Prob) |
| R1 | Equipo | 30% | 500€ | 150€ |
| R2 | Personal | 40% | 1000€ | 400€ |
| R3 | Equipo | 20% | 1000€ | 200€ |
| R4 | Equipo | 50% | 5000€ | 2500€ |
| R5 | Equipo | 20% | 10000€ | 2000€ |
| R6 | Equipo | 10% | 500€ | 50€ |
| R7 | Personal | 10% | 5000€ | 500€ |
| R8 | Equipo | 20% | 10000€ | 2000€ |
| R9 | Equipo | 30% | 20000€ | 6000€ |
| R10 | Equipo | 40% | 15000€ | 6000€ |
| Costes totales de recursos para el proyecto: | |  |  | **19800€** |

Según el análisis cuantitativo de los riesgos, el coste en dinero aproximadamente serían **19800 euros**, para un desarrollo de 13 semanas, pudiendo este presupuesto variar según el estado del proyecto en las diferentes fases del desarrollo.

## 9.5. Costes de Recursos

Dinero destinado a los recursos durante el desarrollo del proyecto a lo largo de 13 semanas para todo el equipo de desarrollo.

|  |  |  |
| --- | --- | --- |
| Operativo | Coste Diario | Coste Total (13 semanas) |
| Electricidad | 0,322 € | 30 € |
| Agua | 0,032 € | 3 € |
| Seguros | 5,62 € | 512 € |
| Costes totales |  | **545 €** |

Tabla 12: Costes de recursos

# 10. Presupuesto final

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Determinación del presupuesto | | | | | | | | | | |
| Costes de personal | | | **Costes de equipo** | | | **Costes operativos** | | **Fondo de riesgo** | |
| Cantidad | **Rol** | **Coste** | **Cantidad** | **Activo** | **Coste** | **Operativos** | **Coste** | **Recurso** | **Coste** |
| 1 | Project  Manager | 16.079€ | 1 | Equipamiento hardware | 1.077,00€ | Electricidad | 30,00€ | Fondo de riesgo | 19800€ |
| 1 | Analista/  Programador | 11.303,00€ | 1 | Equipamiento software | 339,00€ | Agua | 3,00€ |  |  |
| 8 | Programador | 55.616,00€ |  |  |  | Seguros | 512,00€ |  |  |
| 1 | Scrum Master | 14.690,00€ |  |  |  |  |  |  |  |
| 1 | Directora de  marketing | 9.455,00€ |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | 107143€ |  |  | 1416€ |  | 545€ |  | 19800€ |
| TOTAL | | | | | | | | | 128904€ |

Tabla 13: Presupuesto

# 10. Análisis de metodología, calidad y rendimiento

## 10.1. Análisis de metodología

Se empleará la metodología SCRUM para el desarrollo de las iteraciones de desarrollo, puesto que es la que más se ajusta al proyecto que va a hacerse dado los requisitos sobre los continuos despliegues y evaluaciones del proyecto. En cada iteración se realizará un ciclo SCRUM completo, adaptándolo según las necesidades del equipo de desarrollo. Cada semana se realizarán dos reuniones en las que, bajo la vigilancia del Scrum Master para procurar que se cumplan los principios de SCRUM, se repartirán las tareas a los diferentes miembros del equipo y se pondrá en común lo ya realizado por cada uno con el fin de visualizar el estado general del proyecto.

## 10.2. Análisis de calidad

El proyecto se considerará finalizado y capaz de cumplir con los objetivos bajo los cuales fue creado cuando:

1. Todos los requisitos hayan sido satisfechos.
2. Todas las pruebas hayan sido correctamente construidas, ejecutadas y finalizadas con éxito.
3. El proyecto haya sido despegado correctamente junto a su correspondiente carga inicial de datos.
4. Se dé el visto bueno por parte de todos los miembros del equipo.

De no cumplirse alguna de las condiciones anteriormente mencionadas se propondrá una reunión extraordinaria con el fin de abordar la situación y buscar una solución acorde al problema, ya sea disminuir los estándares bajo los que se evalúa el proyecto o considerar que este no puede darse por acabado.

### 10.2.1 Métricas de calidad

Los criterios para medir la calidad de nuestro proyecto quedarán plasmados en los requisitos de calidad comentados en el plan de requisitos. Estos requisitos tienen cierta prioridad dentro del proyecto, así como un determinado criterio de aceptación, y en base a estos parámetros quedará establecido cómo vamos a medir la calidad de nuestro proyecto.

El proyecto además garantizará cierta calidad inicial que provea de distintos niveles de aceptación. Nuestro software deberá tener al menos los siguientes:

|  |  |
| --- | --- |
| Nivel de aceptación | Descripción |
| Compatibilidad | Debe ser compatible y accesible desde cualquier tipo de navegador. |
| Idioma | La aplicación debe estar completamente integrada en castellano. |
| Transparencia | La comunicación entre los miembros del equipo debe ser clara y precisa. |
| Integridad | Las actividades que se encuentran en el cronograma tienen como objetivo el desarrollo del proyecto. |

En relación a los criterios de aceptación mencionados, estableceremos el cumplimiento de dichos requisitos, bajo la supervisión del Scrum Master a lo largo del proyecto. También tendremos en cuenta las siguientes problemáticas que se subsanarán controlando las siguientes métricas de calidad:

|  |  |  |
| --- | --- | --- |
| ID | Objetivo | Preguntas |
| OB-1 | Planificar la gestión de requisitos de forma detallada e inteligible. | PR-1 ¿Cómo podemos asegurar que se ha definido una política eficiente para la priorización de los requisitos?  PR-2 ¿Cómo podemos saber si los requisitos cumplen con los objetivos de la aplicación? |
| OB-2 | Planificar la gestión del cronograma de forma detallada y precisa | PR3 ¿Cómo podemos asegurar que el cronograma es comprensible para todos los miembros del equipo? |
| OB-3 | Planificar la gestión de costes de forma adecuada | PR-4 ¿Cómo podemos estimar correctamente el tiempo de trabajo que se va a emplear?  PR-5 ¿Cómo podemos asegurar que todos los costes se cubren? |
| OB-4 | Planificar la gestión de riesgos de forma detallada y precisa | PR-6 ¿Cómo podemos asegurar que los riesgos están cubiertos?  PR-7 ¿Cómo podemos saber que la priorización de los riesgos es la correcta? |
| OB-5 | Diseñar vistas previas a la implementación | PR-8 ¿Cuántas vistas se estiman en el proyecto?  PR-9 ¿Satisface el diseño todos los requisitos funcionales? |
| OB-6 | Seleccionar las tecnologías que mejor respondan a las necesidades de nuestro proyecto | PR-10 ¿Podemos asegurar que el framework permite cumplir todos los requisitos? |
| OB-7 | Dedicar el tiempo planificado al desarrollo, evitando retrasos. | PR-11 ¿Cuál es el rendimiento del equipo? |
| OB-8 | Desarrollar código con unos estándares de calidad | PR-12 ¿Cómo sabemos que hemos hecho un código con buena calidad? |

**Preguntas**

|  |  |  |
| --- | --- | --- |
| Identificador | Pregunta | Métrica |
| PR-1 | ¿Cómo podemos asegurar que se ha definido una política eficiente para la priorización de los requisitos? | ME-1 |
| PR-2 | ¿Cómo podemos saber si los requisitos cumplen con los objetivos de la aplicación? | ME-2 |
| PR-3 | ¿Cómo podemos asegurar que el cronograma es comprensible para todos los miembros del equipo? | ME-3 ME-4 |
| PR-4 | ¿Cómo podemos estimar correctamente el tiempo de trabajo que se va a emplear? | ME-5 |
| PR-5 | ¿Cómo podemos asegurar que todos los costes se cubren? | ME-6, ME-7 |
| PR-6 | ¿Cómo podemos asegurar que los riesgos están cubiertos? | ME-8 |
| PR-7 | ¿Cómo podemos saber que la priorización de los riesgos es la correcta? | ME-9 |
| PR-8 | ¿Cuántas vistas se estiman en el proyecto? | ME-10 |
| PR-9 | ¿Satisface el diseño todos los requisitos funcionales? | ME-11 |
| PR-10 | ¿Podemos asegurar que el Framework permite cumplir todos los requisitos? | ME-12 |
| PR-11 | ¿Cuál es el rendimiento del equipo? | ME-13 |
| PR-12 | ¿Cómo sabemos que hemos hecho un código con buena calidad? | ME-14 ME-15 ME-16 ME-17 ME-18 |

**Métricas**

|  |  |  |
| --- | --- | --- |
| Identificador | Métrica | Descripción |
| ME-1 | Número de discrepancias en las prioridades de los requisitos. | Veces que nos hemos quejado de que es necesario un requisito antes de otro determinado. |
| ME-2 | Número de incidencias reportadas a causa de los requisitos. | Número de veces que hemos comentado que un determinado requisito no cumple nuestros objeticos y se tiene que modificar. |
| ME-3 | Porcentaje de tareas realizadas respecto a las que había que tener realizadas según el cronograma. | Tareas realizadas vs las que se deben (si se hacen todas sin problemas, significa que todos comprendíamos nuestras tareas del cronograma) |
| ME-4 | Número de actividades realizadas por roles diferentes a los asignados. | Número de tareas por rol, si cada uno tiene un rol y cumple al 100% sus tareas, se comprende que conoce, sabe y desarrolla su labor. |
| ME-5 | Número de retrasos producidos. | Si no hay un número elevado de retrasos, podemos decir que hemos planificado bien y hemos llevado todo según lo planificado. |
| ME-6 | Número de recursos sin coste presupuestado. | Recursos sin costes presupuestado (quiere decir qué si hay, hay una mala estimación porque no lo hemos tenido en cuenta esos determinados recursos) |
| ME-7 | Número de costes mal presupuestados. | Tanto si no lo hemos tenido en cuenta, como si hemos estimado mal el coste, es un problema igualmente. Se refiere a lo mal estimado tanto si sobra como que no. |
| ME-8 | Número de riesgos identificados que no se contemplen. | numero de los riesgos que nos hemos dado cuenta con posterioridad pero no hemos tenido en cuenta en la planificación. |
| ME-9 | Priorización correcta de requisitos | Número de veces que hemos decidido que un riesgo es anterior a otro. |
| ME-10 | Número de mockups para navegar viendo el diseño. | La estimación se satisface con el Número de mockups y estos son capaces de satisfacer el diseño de la app. |
| ME-11 | Porcentaje de requisitos funcionales cubiertos por el diseño. | % de los requisitos completados si obtenemos 100% no nos hemos dejado ningún requisito sin hacer y sin satisfacer |
| ME-12 | Número de funcionalidades requeridas no integradas con las tecnologías seleccionadas. | Con que haya una funcionalidad no realizada por motivo del framework podemos asegurar que no es adecuado y no nos permite satisfacer todos los requisitos. |
| ME-13 | Número de horas empleadas entre las disponibles por semana. | Horas realizadas/horas obligatorias de trabajo -> si sobra tiempo no se le está sacando el máximo rendimiento al equipo. |
| ME-14 | Nota de código de Codacy en el proyecto de Back-end | Nota dada por Codacy al proyecto de Back-end |
| ME-15 | Porcentaje de coverage de Codacy en el proyecto de Back-end | Porcentaje de cobertura de pruebas dado por Codacy en el Back-end |
| ME-16 | Nota de código de Codacy en el proyecto de Front-end | Nota dada por Codacy al proyecto de Front-end |
| ME-17 | Porcentaje de coverage de Codacy en el proyecto de Front-end | Porcentaje de cobertura de pruebas dado por Codacy en el Front-end |
| ME-18 | Porcentaje total de coverage del proyecto en sí | (%Coverage del Back-end +%Coverage del Front-end)/2 |

## 10.3. Análisis de rendimiento

Para analizar el rendimiento del equipo y de los respectivos miembros, se utilizarán los siguientes elementos:

|  |  |  |
| --- | --- | --- |
| Identificador | Métrica de rendimiento | Descripción |
| MR-1 | Número de commits de un miembro durante la semana | La cantidad de commits que ha hecho un miembro durante la semana |
| MR-2 | Número de commits de un miembro durante el Sprint | La cantidad de commits que ha hecho un miembro durante el Sprint |
| MR-3 | Tareas finalizadas por un miembro durante la semana | Cantidad de tareas que ha hecho un miembro durante una semana del Sprint |
| MR-4 | Tareas finalizadas por un miembro durante el Sprint | Cantidad de tareas que ha hecho un miembro durante el Sprint |
| MR-5 | Tareas finalizadas por un miembro dentro de los planificado | Tareas que ha hecho un miembro dentro de lo planificado |
| MR-6 | Tiempo dedicado por semana | Tiempo total que le ha dedicado un miembro durante la semana |
| MR-7 | Tiempo dedicado en el Sprint | Tiempo total que le ha dedicado un miembro durante el Sprint |
| MR-8 | Rendimiento del Project Manager durante el Sprint | 10-0.5\*(Número de incoherencias, malas estimaciones, tareas que no se han contemplado y dependencias que no se han contemplado) |

# 11. Plan de gestión de riesgos

En este plan de riesgos tendremos en cuenta todos los posibles problemas que podremos encontrar durante la ejecución del proyecto con la finalidad de localizarlos a tiempo y poder suprimirlos o evitarlos consiguiendo así mayor rendimiento. A continuación, se mostrará los riesgos potenciales que pueden tener un efecto perjudicial en el mismo, incluyendo un identificador, el nombre, la categoría a la que pertenece, una breve descripción y el impacto que podría suponer para el equipo:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Nombre** | **Categoría** | **Descripción** | **Impacto** |
| R1 | Actualizaciones del software | Organización | Se producen actualizaciones de sofware | Medio |
| R2 | Baja de personal por coronavirus | Desarrollo | Debido al coronavirus, el equipo sufre la pérdida de un integrante quedando así su parte del trabajo por realizar | Alto |
| R3 | Cambio en requisitos o tecnologías | Planificación | Se decide cambiar algunas de las tecnologías para el desarrollo del proyecto | Bajo |
| R4 | Un competidor desarrolla un producto muy similar. | Desarrollo | Aparece un competidor con un producto cuyas características son similares a las nuestras | Medio |
| R5 | Pérdida de información o archivos durante el desarrollo del producto | Desarrollo | Se pierden datos debido a la caída de la nube o porque algún miembro no lo ha subido y lo ha perdido en local | Medio |
| R6 | Los canales de comunicación del equipo fallan. | Organización | Los canales establecidos fallan por diversos motivos, haciendo imposible la transmisión de información o de celebrar reuniones | Bajo |
| R7 | No cumplir con alguna ley referente al proyecto | Organización | Algún miembro no cumple con la ley o norma establecida en el proyecto | Bajo |
| R8 | Una librería usada en el proyecto deja de estar disponible | Desarrollo | La librería que se utiliza en el proyecto deja de estar disponible debido a cambios en los términos de uso o desaparece | Medio |
| R9 | Aparece un error crítico en una fase avanzada del proyecto | Desarrollo | Se produce un error mayor en una etapa importante del proyecto | Alto |
| R10 | Problemas en despliegue en aplicaciones de condiciones gratuitas | Desarrollo | Aparecen problemas que dificultan el despliegue o que lo imposibilitan debido al cambio de condiciones | Alto |

Tabla 14: Riesgos

Además, hay una serie de acciones las cuales se deben poner en práctica en el caso de que se pudiera producir alguno de estos riesgos en el equipo. Dichas acciones están recogidas en un plan de respuesta a los riesgos, del cual debemos considerar lo siguiente:

|  |  |  |
| --- | --- | --- |
| **ID** | **Nombre** | **Acciones** |
| R1 | Actualizaciones del software | Se intentará volver a descargar y establecer la versión que se estableció inicialmente en el inicio del proyecto |
| R2 | Baja de personal por coronavirus | En caso de que la persona esté incapacitada por el coronavirus, se repartirán sus tareas al resto de miembro |
| R3 | Cambio en requisitos o tecnologías | Se deberá hacer una replanificación y se discutirá en una reunión sobre las tecnologías que se usarán para el proyecto |
| R4 | Un competidor desarrolla un producto muy similar. | En este caso, se deberá hacer una replanificación y añadir nuevas funcionalidades o mejoras que no tenga el competidor. Se realizará un análisis de las características del competidor |
| R5 | Pérdida de información o archivos durante el desarrollo del producto | Se volverá a crear nueva información que intente sustituir la perdida. Si hay copias de seguridad, entonces se utilizarán para repoblar la nube del equipo donde se encuentra toda la información |
| R6 | Los canales de comunicación del equipo fallan. | Se utilizará otras alternativas en caso de que fallen. Si discord falla, pasaríamos la reunión a Google meet. Si whatsapp falla, se cambiaría a Telegram. |
| R7 | No cumplir con alguna ley referente al proyecto | Si algún miembro no cumple con lo establecido en el proyecto, se le dará un aviso. Si persiste, el grupo se reunirá para decidir si echar a esa persona. |
| R8 | Una librería usada en el proyecto deja de estar disponible | Se realizará una búsqueda para encontrar alternativas. Los miembros que usan la librería que está en desuso deberán investigar. |
| R9 | Aparece un error crítico en una fase avanzada del proyecto | En caso de que una versión posea un error crítico que no se pueda corregir, se eliminará la rama o el commit con ese error. Para impedir que el error se extienda a ramas importantes. |
| R10 | Problemas en despliegue en aplicaciones de condiciones gratuitas | Si no se deja desplegar debido al cambio de las condiciones gratuitas, se pagará al servicio para poder desplegar. Si surgen otro tipo de problemas, el equipo intentará ayudar a los responsables del despliegue. |

Tabla 15: Acciones contra riesgos

# 12. Plan de comunicaciones

## 12.1. Interesados

Los interesados en las comunicaciones del proyecto son los integrantes de este grupo de trabajo. Por tanto, las comunicaciones se harán entre los mismos.

## 12.2. Información que comunicar

La información que se ha de comunicar recae sobre todo en el reparto de tareas y en el seguimiento de estas. Para ello, se llevarán a cabo reuniones periódicas cada semana, necesarias para poder obtener un incremento en el producto, ya nos refiramos a la propia planificación, como al desarrollo del producto.

## 12.3. Reuniones

Debido a la variedad de compatibilidades de los integrantes del grupo de trabajo, se acaba por concretar reuniones semanales periódicas, las cuales se llevarán a cabo los miércoles a las 12:45 y los sábados a las 10:30, concretándose la hora exacta de la misma telemáticamente a través de la aplicación **Whatsapp**. El encargado de concretar dicha hora es el jefe de proyecto, Fernando Wals. Cabe indicar que cualquier reunión fuera de este horario sería comunicada con suficiente antelación para que todos los miembros del grupo puedan estar al tanto.

No se llevará a cabo ningún seguimiento automatizado para estas reuniones.

Las decisiones que se tomen en las reuniones quedarán plasmadas en actas de reunión, en las que se recoge el orden del día de cada reunión y las decisiones tomadas por parte del grupo.

## 12.4. Reparto por subgrupos e intercomunicaciones

El grupo de trabajo se dividirá en 2 subgrupos principales, el de Front-end y el de Back-end. Una persona dentro de estos subgrupos será la encargada de comunicar al jefe de proyecto cualquier incidencia o duda en el desarrollo de las tareas. De todas formas, las reuniones se celebrarán con el grupo completo, mientras otra medida no sea necesaria.

## 12.5. Tecnologías

Como se ha indicado anteriormente, uno de los canales usados para la comunicación se tratará de la aplicación **Whatsapp** para concretar disponibilidades, o preguntar dudas. Las reuniones tienen lugar mediante el uso de la aplicación **Discord**, la cual permite la realización de videollamadas y el compartir pantallas para concretar información. Para el intercambio de archivos, se hará uso de **Google Drive** y para la gestión del avance del proyecto se usará un repositorio de **Github**.

Con respecto a la organización del proyecto, el grupo ha estado utilizando la aplicación Trello. De todas formas, debido a decisiones tomadas en función a las indicaciones del profesorado, se ha optado recientemente por cambiar al sistema de **Github Projects.**

## 12.6. Factores ambientales

Como inciso, y ante la situación actual de emergencia sanitaria, se ha concretado que todas las reuniones serán telemáticas a través de **Discord**, como se ha indicado anteriormente.

# 13. Anexo

## 13.1. Equipo de trabajo

### 13.1.1. Organigrama

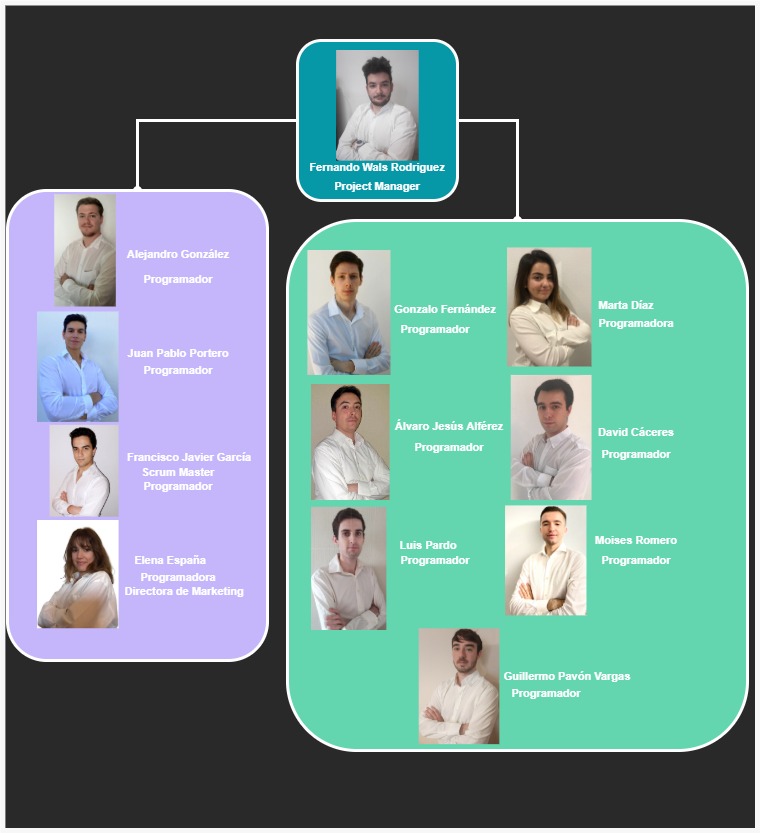


Ilustración : Organigrama del equipo

### 13.1.2. Características del equipo

A continuación, se exponen tanto las fortalezas como las debilidades de cada uno de los miembros del equipo de trabajo, para evaluar las capacidades de este:

Project Manager:

* *Fernando Wals Rodríguez* aprende nuevas tecnologías de manera eficaz, sobre todo las referentes a computación gráfica y diseño 3D, y tiene mucha habilidad a la hora de hablar en público y comunicarse con el equipo. Sin embargo, no posee la paciencia necesaria en algunos aspectos, como con los sistemas de integración continua como Travis. Las tecnologías que mejor usa son **MongoDB**, **React**, **Java** y **C++**.

Front End:

* *Francisco Javier García Roales* tiene una buena capacidad de organización y comunicación además de una gran adaptación a los cambios si se presentaran, pero su debilidad es la creatividad. En cuestión de tecnologías sus puntos fuertes son **Java, MySQL, Spring y React**.
* *Elena España González* es muy buena tanto en el ámbito de aportar ideas creativas, como en el de organizarse y comunicarse con el resto del equipo. Sin embargo, al no haber cursado la asignatura de Diseño y Pruebas debido a Erasmus, no posee el conocimiento de algunas tecnologías como Spring. En cuanto a sus tecnologías mejor usadas se encuentran **Java y Python**.
* *Alejandro González Martin* sabe trabajar bien con el Front-end y posee creatividad en lo referente a logos y diseños, pero no es tan bueno con el Back-end y le cuesta un poco comunicarse y presentar en público. **Java**, el framework de **Spring** y **CSS** son sus mejores tecnologías.
* *Juan Pablo Portero Montaño* tiene la capacidad de adaptarse muy bien en un equipo y tiene mucha constancia cuando se trata de aprender nuevas tecnologías, pero no es muy bueno exponiendo, y además le es muy difícil implementar mucho código en poco tiempo. En cuanto a las tecnologías que mejor usa se encuentran **Spring**, **Java** y **React**.

Back End:

* *Álvaro Jesús Alférez Richarte* es bueno a la hora de adaptarse a diferentes tecnologías y lenguajes, además de ser muy persistente con su trabajo, aunque posee poca experiencia con algunas tecnologías como MongoDB y con grupos tan grandes. Las tecnologías en las que es más diestro son **Java, Spring, PHP y CSS/Boostrap.**
* *David Cáceres Romero* tiene una creatividad muy buena para resolver problemas, y su capacidad de adaptación y comunicación también es positiva. Sin embargo, no trabaja muy bien al principio con tecnologías nuevas o con las que tenga poca experiencia. Además, tampoco es muy bueno en lo que respecta al tema de realizar exposiciones y presentaciones. Como tecnologías que mejor utiliza se pueden mencionar **C#, C++, Java** y el framework de **Spring**.
* *Marta Diaz Fernández* posee como cualidades una buena capacidad para aprender nuevas tecnologías y trabajar en equipo, aunque no se le dan tan bien los temas referentes a la creatividad, ni los referentes a presentaciones en público. Las tecnologías en las que mayor destreza tiene son **SpringBoot, Python, Java y CSS.**
* *Gonzalo Fernández Jiménez* trabaja bien con la tecnología Spring en lo referente al Back-end, además de saber buscar eficientemente requisitos e información. Por otro lado, no tiene mucha experiencia en el tema de Front-end y no es tan bueno hablando en público. **SpringBoot, Java y C#** son las tecnologías que mejor utiliza.
* *Luis Pardo López* puede trabajar muy bien una vez sabe cómo realizar una tarea, pero le cuesta exponer y aprender bien el funcionamiento de nuevas tecnologías, por lo que suele necesitar ayuda al principio. Las tecnologías que mejor usa son **Java** y **Spring**.
* *Guillermo Pavón Vargas* posee la cualidad de aprender bien el funcionamiento de nuevas tecnologías y de aportar ideas creativas al equipo de trabajo, aunque sus debilidades son exponer en público y su poca experiencia en el ámbito del Front-end. **Java**, **Python** y **Spring** son sus tecnologías mejor utilizadas.
* *Moisés Romero Romo* sabe realizar una buena documentación, además de tener una gran capacidad para aprender nuevas tecnologías. Sin embargo, su habilidad a la hora de exponer en público es menor y además al no haber cursado la asignatura de Diseño y Pruebas debido a Erasmus, no ha trabajado previamente con algunas tecnologías como Spring. En contraposición, tiene mayor soltura y destreza en tecnologías como **Java**, **Javascript** y **CSS**.

## 13.2. Estudio de las competencias

A continuación se efectuará un estudio de otras páginas que comparten similitudes con inDvelopers, con el fin de compararlas y extraer ventajas y desventajas de los competidores.

13.2.1 Competencias

#### 13.2.1.1. Steam

Steam es la aplicación de servicio de venta online de videojuegos más popular del mundo. Posee un catálogo enorme de juegos de todo tipo y antigüedad. Pone a la venta desde triples A hasta juegos indies. Pero todos estos juegos deben pasar por una serie de contratos, sobre todo referidos a las ganancias, controlados por steam.

Que hace bien Steam:

* Tener un gran catálogo. Permite comprar tanto nuevos juegos como descatalogados en consola.
* Ofertas. Suele tener un gran sistema de ofertas que incentiva a los usuarios a comprar juegos por un precio que pocas veces sale en tiendas físicas.

Que podríamos hacer nosotros mejor:

* En el tema de juegos indies solo algunos afortunados se abren paso para llegar al top 20 juegos más comprados. Eso se debe a que steam no hace favores y por tanto si eres un desarrollador o un equipo indie es muy difícil que alcances el nivel mediático de los grandes lanzamientos.
* Nosotros ponemos juegos indies a competir contra otros juegos indies, de manera que si alguno de estos juegos triunfa seguramente dejará la plataforma tarde o temprano para comercializarse en Steam, por ejemplo. Al hacer esto evitará eclipsar a otros juegos con potencial.
* Nosotros solo pedimos una suscripción mientras que Steam pide adicionalmente el 30% de las ganacias del videojuego.

#### 13.2.1.2. Xbox Game Store

Xbox Store es un servicio de venta online de videojuegos para la popular consola de Microsoft. El programa de Microsoft se llama Xbox Live Creators y permite a los creadores publicar los juegos sin necesidad de aprobación de concepto, simplemente tienen que seguir los estándares de calidad de Microsoft Store.

Que hace bien Xbox Store:

* Posee una tarifa de registro como desarrollador única.

Que podríamos hacer nosotros mejor:

* No necesitas cumplir con estándares de calidad.
* Nosotros solo pedimos una suscripción mientras que Xbox Store pide adicionalmente el 30% de las ganacias del videojuego.

#### 13.2.1.3. Nintendo Shop

Nintendo Shop es un servicio de venta online de videojuegos para las consolas de Nintendo.

Que hace bien Nintendo Shop:

* Da un kit de desarrollo que se adapta al extraño (en comparación con el resto de consolas y PC) hardware de Nintendo.

Que podríamos hacer nosotros mejor:

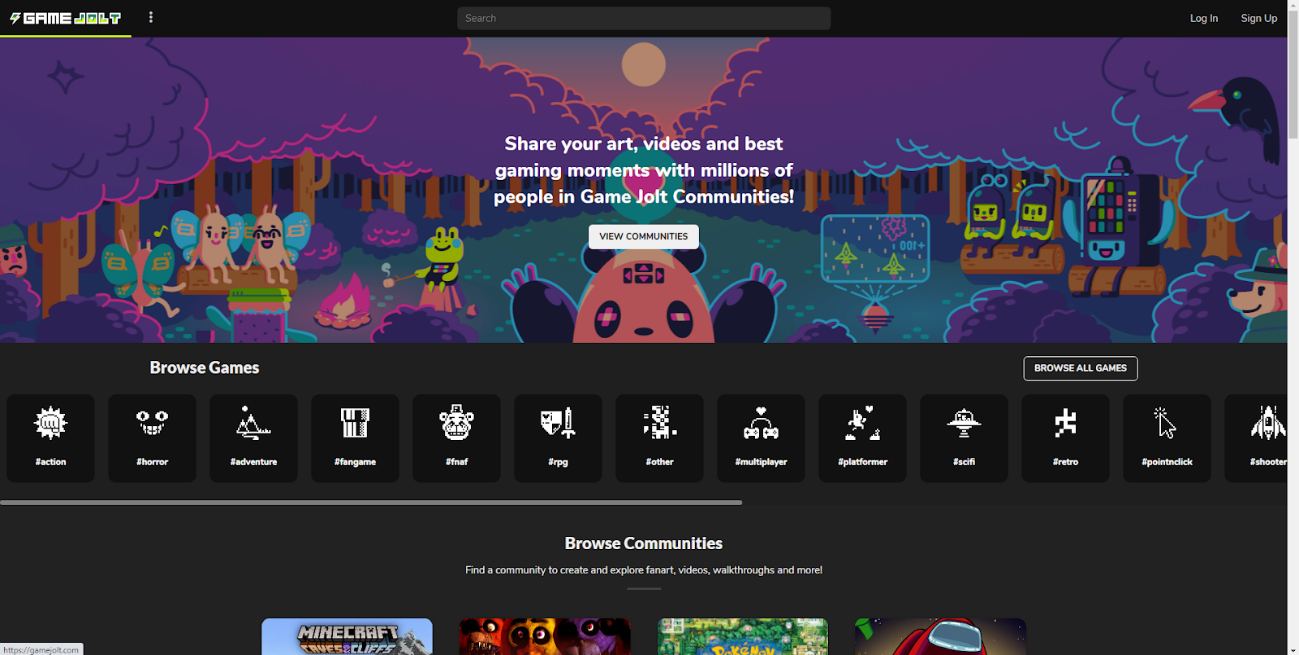
* No pedimos una cantidad tan grande para empezar como los 418€ que pide Nintendo.
* Nosotros solo pedimos una suscripción mientras que Nintendo pide el 30% de las ganacias del videojuego.

#### 13.2.1.4. GameJolt

Gamejolt es un portal en el que un desarrollador puede compartir sus creaciones con el fin de recibir feedback del público general. También contiene una sección donde la comunidad de amantes de videojuegos puede interactuar entre sí mediante la subida de fanart o temáticas de discusión, o simplemente hablar con otros usuarios. Esta última sección se acerca mucho al concepto de blog.

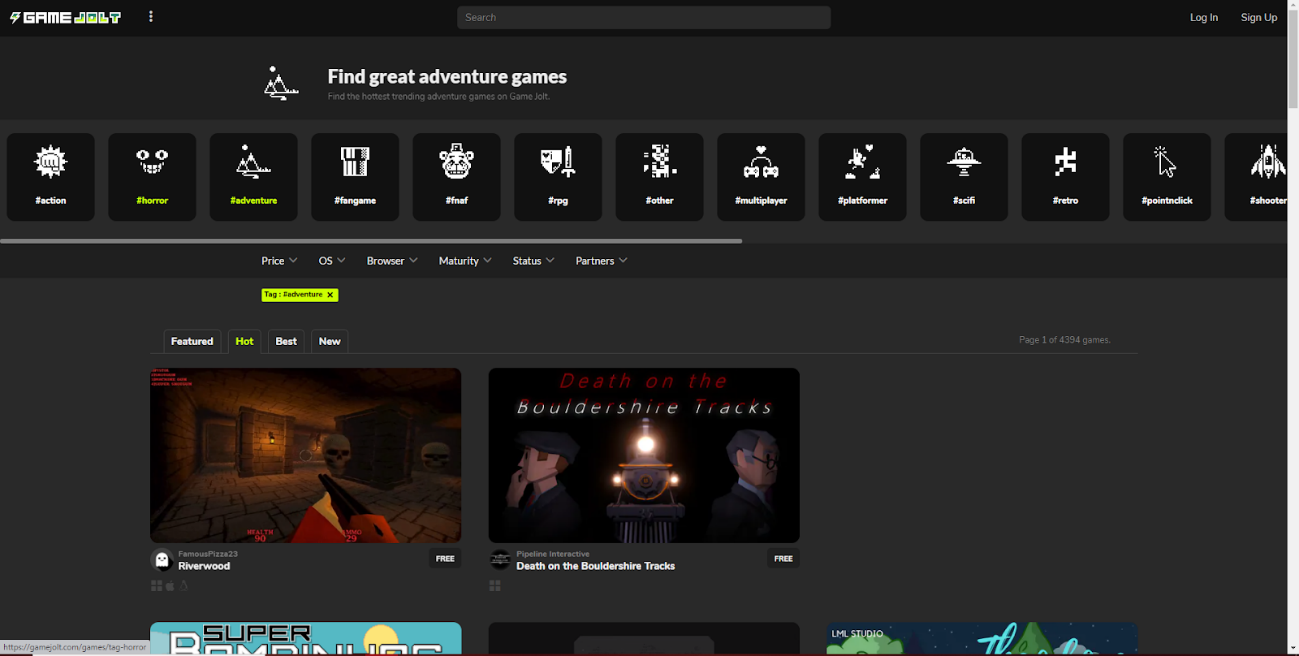
Navegabilidad:

Al acceder a la página, lo primero que podemos ver es una interfaz simple con varios elementos diferenciados, que se pueden navegar de manera simple e intuitiva



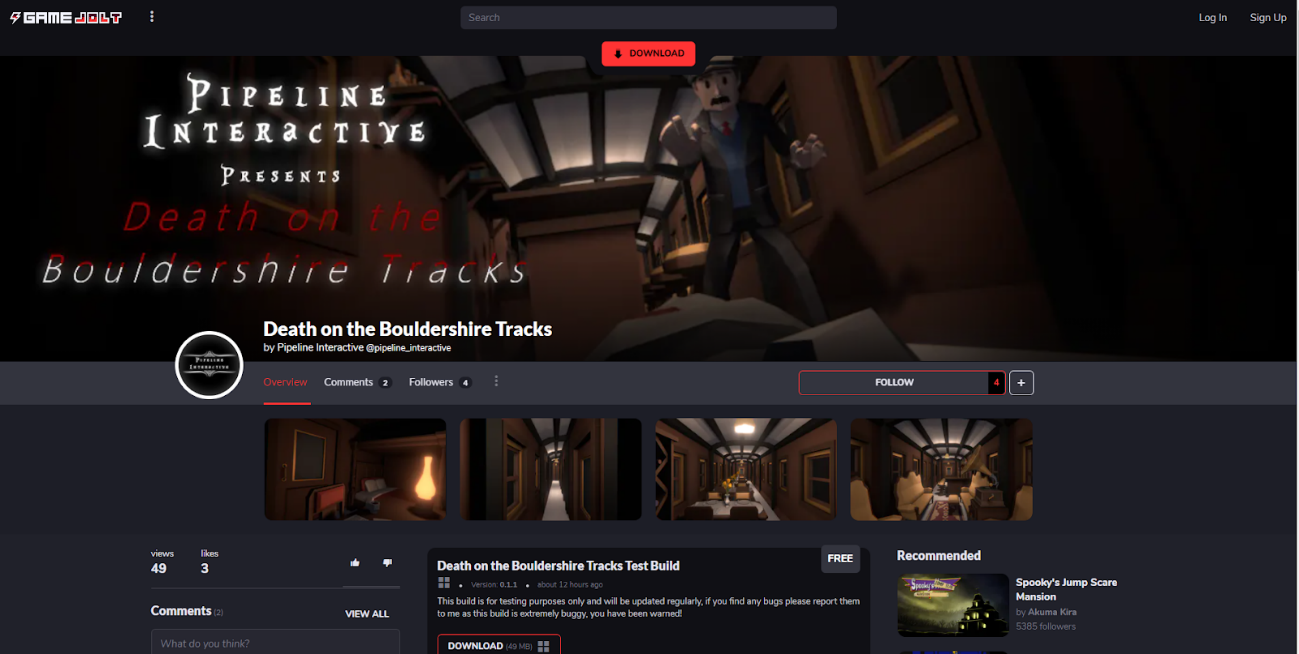
Hay una sección que atrae al usuario a leer sobre la sección de “blog” de la página, otra sección en la que el usuario puede elegir entre los distintos tipos de géneros de juegos para encontrar resultados relevantes a sus preferencias, y una última sección donde se pueden ver las distintas comunidades a las que el usuario puede acceder basado en sus preferencias.

La página también tiene un buscador en la parte superior y opciones para investigar sobre todos los juegos, en lugar de sólo un tipo.



Al acceder a un tipo determinado de juegos, podemos ver los resultados en la parte inferior de la página, y estos se pueden filtrar por popularidad, orden de estreno en la web o incluso por mejores críticas.

A su vez, se pueden clasificar por precio, sistema operativo para el que están diseñados, extensiones de buscador con las que son compatibles, nivel de madurez o incluso estado del producto.



Cuando elegimos un juego, podemos ver que tiene mucha información acerca del mismo en la página resultante. Tiene una sección donde el creador explica la idea principal del juego, imágenes y vídeos que enseñan la estética del juego, e incluso una sección donde se pueden ver comentarios de otros usuarios acerca del juego o una lista de los interesados que quieran recibir actualizaciones sobre el estado del mismo.

En caso de que el juego esté disponible para descargar, el usuario verá la opción de hacerlo en la parte superior de la pantalla, o si no lo está, podrá añadirse a sí mismo a la lista de interesados para recibir notificaciones si se hace algún avance o si pasa a estar disponible.

Por lo general, la página es simple e intuitiva, con una estética uniforme que recuerda al usuario continuamente que está en la misma página.

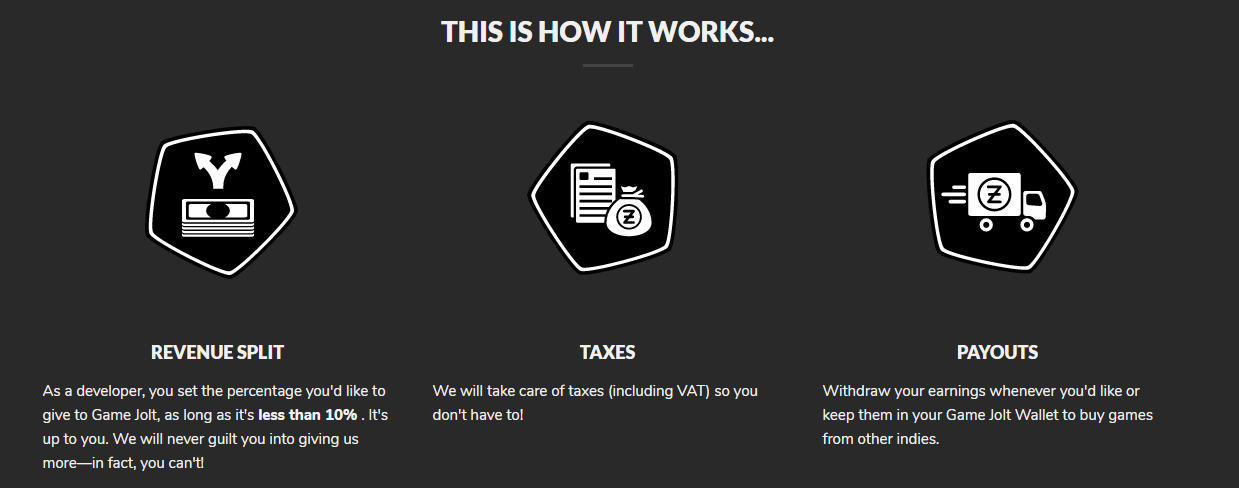
Contenido

El contenido de gamejolt.com es en su absoluta mayoría proporcionado por la comunidad de usuarios de la web. Se trata tanto de juegos disponibles al usuario como de anuncios de juegos que pronto estarán al alcance de todos. En la sección de “comunidades”, la que mencionamos anteriormente como un blog, el contenido es el que la comunidad sube a los distintos grupos.

Toda página acerca de un juego o la creación de algún usuario tiene la opción de escribir comentarios con el fin de interactuar con la persona que sube el contenido, dando lugar a la comunidad de expresarse.

Ingreso de beneficios

gamejolt.com obtiene sus beneficios de dos maneras principales. La primera es la venta de “merchandising” con la temática de la página, la cual podemos ver en esta fotografía tomada de su tienda. La segunda viene explicada en la sección “marketplace” de la web.



La imagen describe el proceso de vender un juego como creador del mismo, en el que el propio desarrollador elige qué porcentaje de los beneficios generados por su juego cede a gamejolt.com, mientras este porcentaje sea menor al 10% del precio completo del producto.

También indican que ellos se encargan de los impuestos y que cualquier usuario puede retirar su dinero en cualquier momento de su cartera de gamejolt.

No se ha encontrado ningún anuncio en la página, por lo cual la opción de ingresar beneficios por publicidad queda descartada.

#### 13.2.1.5. Epic Store

Epic Store es un servicio de venta online de videojuegos relacionados con el motor de desarrollo Unreal Engine.

Que hace bien Epic Store:

* No pide dinero mientras no supere los 10.000$, en cuyo caso hay que pagar la licencia del motor de desarrollo.

Que podríamos hacer nosotros mejor:

* No es necesario pagar ninguna licencia.
* Nosotros solo pedimos una suscripción mientras que Epic Store pide el 12% de las ganacias del videojuego.

#### 13.2.1.6. Google Play Store

Google Play es un servicio de venta online de videojuegos móviles.

Que hace bien Google Play Store:

* Solo pide el pago para ser desarrollador una vez, de 25$.

Que podríamos hacer nosotros mejor:

* Nosotros solo pedimos una suscripción mientras que Google Store pide el 30% de las ganacias del videojuego.

#### 13.2.1.7. Apple Store

Apple Store es un servicio de venta online de videojuegos del sistema operativo iOS.

Que hace bien Epic Store:

* Nada destacable.

Que podríamos hacer nosotros mejor:

* Nosotros solo pedimos una suscripción mientras que Google Store pide el 30% de las ganacias del videojuego y una subscripción anual de 99$.
* Apple impone una gran cantidad de pasos y controles para la publicación de juegos, nosotros apenas pedimos ninguno.

### 13.2.1. Que nos diferencia del resto

1. Tenemos una tarifa gratuita para subir juegos sin precios o demos. Con esto damos una incentiva a que cualquier usuario que quiera recibir feedback de la comunidad pueda subirlo gratuitamente.
2. Al estar centrados en juegos indies podemos evitar muchos puntos burocráticos que tengan que ver con impuestos, precios, ganancias y contratos de empresa.
3. La mayoría de las aplicaciones de venta de juegos están basados en juegos finales y corporaciones. Mientras que nuestra aplicación se centra en dar visibilidad a desarrolladores indies que en otro tipo de plataformas se arriesgan a ser ignorados o eclipsados por grandes lanzamientos.

**IMPORTANTE: No es que no haya competencia de grandes empresas a nuestros productos, SINO que nuestro producto, al centrarse solo en Indies, evita que los desarrolladores Indies tengan que competir con juegos de grandes corporaciones.**

1. Al tener un sistema de suscripciones con un coste constante mensual/anual para las ganancias y mantenimiento de servidores, los usuarios tienen total libertad de elegir el precio y no tienen por qué estar preocupados por limitaciones de ganancias.

### 13.2.2. Diagrama DAFO

|  |  |  |
| --- | --- | --- |
|  | INTERNO | EXTERNO |
| NEGATIVO | **Debilidades:**   * Nuestros ingresos dependen más de la popularización que de la calidad, si poca gente usa nuestra aplicación poca gente entrará nueva y podrá desembocar en fracaso. Empezar con buen pie es importante. | **Amenazas:**   * Ya existen muchos competidores con experiencia y un público que puede dificultar captar nuevo público que esté dispuesto echar un vistazo a nuevas optativas. |
| POSITIVO | **Fortalezas:**   * Sistema que no penaliza a los desarrolladores por triunfar. * Los indies no tienen porque compartir escaparate con grandes empresas. Más justo para los desarrolladores indies. | **Oportunidades:**   * Muchas empresas se centran en sus propias IPs o abarcar en general el mercado. Nosotros nos centramos en los desarrolladores Indies, dándoles toda la atención, lo que es algo no tan común en la industria. |

Tabla 16: DAFO

Tabla resumen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Costes para desarrollar | Mercado | Ventajas | Desventajas |
| Steam | 30% ganancias | PC | Gran público | Coste de 100$ de entrada, triunfe o no el juego.  Los juegos indies pueden ser eclipsados por los grandes lanzamientos. |
| Epic Store | 12% ganancias + 5% si recoge grandes ganancias | PC | Asegura mínimos de calidad | Los juegos indies pueden ser eclipsados por los grandes lanzamientos. |
| Google Play Store | 25$ + 30% ganancias | Mobile/  Tablet | Gran público | Limitado temas para juegos. No sexualidad, violencia explícita, drogas, etc. Solo una gran empresa puede sortear estas limitaciones. |
| Good Old Games | 30% ganancias | PC Indies | Desarrolladores Indies no tienen que competir contra grandes desarrolladoras | *Ninguna destacable* |
| GameJolt | <10% ganancias | PC Indies | Desarrolladores Indies no tienen que competir contra grandes desarrolladoras | Tienen un sistema de publicidad en juegos que puede perjudicar la calidad de los mismos. |
| Itch.io | <10% ganancias | PC Indies | Desarrolladores Indies no tienen que competir contra grandes desarrolladoras | *Ninguna destacable* |

## 13.3. Charter goal

En este documento se intenta expresar el objetivo común entre los integrantes del proyecto a realizar, así como redactar un compromiso firmado por todos los miembros del equipo en el que nos comprometemos a realizar un cierto trabajo, destacando las consecuencias de incumplir dicho compromiso.

Todos los integrantes del grupo tenemos el objetivo de acabar consiguiendo el mejor producto posible, trabajando en el proyecto de manera interesada y activa de forma que la aplicación resultante así como la documentación a realizar y todo el resto de tareas tengan la mejor calidad dentro de nuestras capacidades. Debido a esto el equipo en su totalidad debe comprometerse a realizar las tareas que le han sido encomendadas, así como ejercer su rol dentro del proyecto lo mejor posible. Para lograr un resultado adecuado cada miembro del equipo deberá trabajar un mínimo de 6 horas.

Cabe destacar que el interés y la participación de forma activa son vitales para el desarrollo de calidad de cualquier producto. Por este motivo resulta de gran importancia que los miembros del equipo no se limiten a realizar únicamente lo que se les encomienda, sino que se preocupen por el desarrollo del elemento a desarrollar y sean proactivos a la hora de realizar tareas, ayudar a quien lo necesite, aportar ideas, buscar información, o cualquier otro aspecto que pueda ser de utilidad para el proyecto.

En caso de no cumplir con los aspectos mencionados con anterioridad, el equipo, de manera consensuada, podría llegar a tomar la decisión de expulsar a dicho miembro, o cualquier otra acción a tomar que el equipo en conjunto considere oportuna.

Este acuerdo, al igual que el resto de documentos e ideas del proyecto, podría llegar a modificarse en un futuro, siempre y cuando todos los miembros del equipo estén de acuerdo con dichas modificaciones y teniendo que ser firmado de nuevo por cada uno de los miembros del mismo.

Firmas

|  |  |
| --- | --- |
| Nombre | Firma |
| Alvaro Jesus Alferez Richarte |  |
| David Caceres Romero |  |
| Marta Díaz Fernández |  |
| Elena España González |  |
| Gonzalo Fernandez Jimenez |  |
| Francisco Javier Garcia Roales |  |
| Alejandro Gonzalez Martin |  |
| Luis Pardo Lopez |  |
| Guillermo Pavon Vargas |  |
| Juan Pablo Portero Montaño |  |
| Moises Romero Romo |  |
| Fernando Wals Rodriguez |  |

## 13.4. Actas de reunión durante la planificación

### 13.4.1. Acta 0

|  |  |
| --- | --- |
| **Acta nº 0** |  |
| **Fecha** | Dia 27 del 02 de 2021 |
| **Hora de inicio** | 11:05 h |
| **Hora de fin** | 12:52 h |
| **Tipo de reunión** | Reunión inicial |

|  |  |
| --- | --- |
| **Nombre** | **Asistencia** |
| Alvaro Jesus Alferez Richarte | x |
| David Caceres Romero | x |
| Marta Diaz Fernandez | x |
| Elena España González | x |
| Gonzalo Fernandez Jimenez | x |
| Francisco Javier Garcia Roales | x |
| Alejandro Gonzalez Martin | x |
| Luis Pardo Lopez | x |
| Guillermo Pavon Vargas | x |
| Juan Pablo Portero Montaño | x |
| Moises Romero Romo | x |
| Fernando Wals Rodriguez | x |

1. Orden del dia

1. Elección de proyecto
2. Elección de tecnología a utilizar
3. División de los miembros del equipo en grupos de trabajo
4. Reparto de tareas de la presentación
5. Reparto de roles

2. Temas tratados.

Se comentan las dos ideas que fueron más votadas para recordarlas al resto del grupo y se realiza una votación para decidir, gana por dos votos a favor la opción de Fernando Wals. Se elige como nombre de la aplicación “Indvelopers”.

Distintos miembros del grupo mencionan tecnologías que podrían utilizarse con la intención de escoger una con la que una mayoría esté ya familiarizado.

Se plantean como opciones:

* React, Spring y MongoDB
* Vue.js, Django y Postgresql
* Bootstrap, Spring y MongoDB

La votación finaliza con 9 votos a favor de React, Spring y MongoDB y 2 para Bootstrap, Spring y MongoDB, eligiendo por tanto la primera como opción.

Para mejorar el reparto de tareas se divide el equipo principalmente en tareas de front end, back end y se decide que haya al menos dos miembros del equipo que se encarguen más de la documentación para asegurar que sea coherente. Dado que un porcentaje alto de los miembros del equipo tienen un nivel alto de inglés se plantea la posibilidad de realizar la documentación en este idioma.

Se realiza un brainstorming para elegir un logo provisional que identifique a nuestro proyecto. Elena España muestra una idea inicial que se decide modificar para ajustarla más a las necesidades e identidad del proyecto.

Se añaden las tareas necesarias para la presentación del martes 2 de marzo al Trello y se adjudican a los distintos miembros del equipo con fecha límite entre el domingo y el lunes.

Los roles principales a repartir son Project Manager, DevOps, Scrum Master, directora de marketing. El rol de Programador será adjudicado a todos los miembros del equipo y en los casos en que un mismo miembro tenga más de un rol las tareas se repartirán de forma que las cargas de trabajo sean igualitarias.

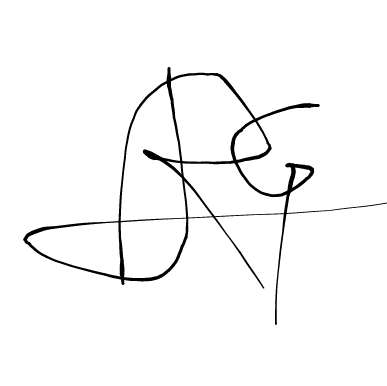
El reparto de roles queda de la siguiente forma

|  |  |
| --- | --- |
| **Nombre** | **Rol** |
| Alvaro Jesus Alferez Richarte | Programador |
| David Caceres Romero | Programador |
| Marta Diaz Fernandez | Programador |
| Elena España González | Directora marketing, Programador |
| Gonzalo Fernandez Jimenez | Programador |
| Francisco Javier Garcia Roales | Scrum Master, Programador |
| Alejandro Gonzalez Martin | Programador |
| Luis Pardo Lopez | Programador |
| Guillermo Pavon Vargas | Programador |
| Juan Pablo Portero Montaño | Programador |
| Moises Romero Romo | Programador |
| Fernando Wals Rodriguez | Project Manager, Programador |

3. Acuerdos 

1. Se elige como proyecto final la opción planteada por Fernando Wals, plataforma destinada a desarrolladores indies donde puedan ponerse en contacto y mostrar su trabajo.
2. Se eligen como tecnologías para el desarrollo React, Spring y MongoDB.
3. Se acuerda el uso de Trello.
4. Se utilizará GitHub como repositorio para el proyecto

4. Cierre de la reunión

             Francisco Javier Garcia Roales,

### 13.4.2. Acta 1

|  |  |
| --- | --- |
| **Acta nº 1** |  |
| **Fecha** | Dia 03 del 03 de 2021 |
| **Hora de inicio** | 12:45 h |
| **Hora de fin** | 14:10h |
| **Tipo de reunión** | Reunión seguimiento |

|  |  |
| --- | --- |
| **Nombre** | **Asistencia** |
| Alvaro Jesus Alferez Richarte |  |
| David Caceres Romero | x |
| Marta Diaz Fernandez | x |
| Elena España González | x |
| Gonzalo Fernandez Jimenez | x |
| Francisco Javier Garcia Roales | x |
| Alejandro Gonzalez Martin | x |
| Luis Pardo Lopez | x |
| Guillermo Pavon Vargas | x |
| Juan Pablo Portero Montaño | x |
| Moises Romero Romo | x |
| Fernando Wals Rodriguez | X |

1. Orden del día

1. Revisión de la idea
2. Revisión del método de obtención de beneficios
3. Preparación de la próxima presentación
4. Reparto de tareas

2. Temas tratados

Teniendo en cuenta el feedback recibido durante la clase del 2 de marzo se plantea concretar más la idea del proyecto para conseguir un MVP más asequible conforme al tiempo de desarrollo.

La idea, planteada por Fernando Wals, consiste en reducir el número de casos de uso para centrar el desarrollo en desarrolladores, juegos, publicaciones y críticas.

La idea se acepta, planteando la posibilidad de añadir funcionalidades en caso de que dé tiempo.

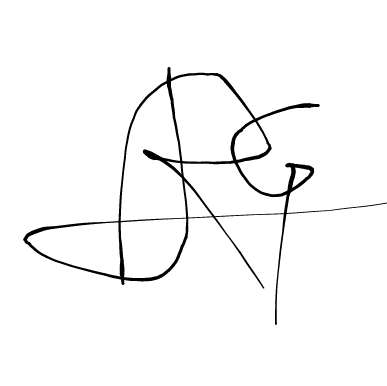
Se modifica el método de obtención de beneficios para complementar a los posibles ingresos de publicidad de acuerdo con los comentarios recibidos en clase.

Para mejorar la presentación de los competidores se decide realizar un checklist con aspectos a comprobar para cada competidor y poder hacer un estudio más exacto y coherente.

Se realiza una lista de Mockups a realizar y se reparten entre varios miembros del equipo, se decide hacer un mockup modelo para asegurar que todos tengan el mismo estilo.

Para facilitar la realización de la documentación se decide hacer un control de versiones evitando tener que realizar al final toda la documentación lo que podría provocar un cúmulo de trabajo muy alto.

4. Cierre de la reunión

             Francisco Javier Garcia Roales,

### 13.4.3. Acta 2

|  |  |
| --- | --- |
| **Acta nº 2** |  |
| **Fecha** | Dia 06 del 03 de 2021 |
| **Hora de inicio** | 11:00 h |
| **Hora de fin** | 12:20 h |
| **Tipo de reunión** | Reunión seguimiento |

|  |  |
| --- | --- |
| **Nombre** | **Asistencia** |
| Alvaro Jesus Alferez Richarte | x |
| David Caceres Romero | x |
| Marta Diaz Fernandez | x |
| Elena España González | x |
| Gonzalo Fernandez Jimenez | x |
| Francisco Javier Garcia Roales | x |
| Alejandro Gonzalez Martin | x |
| Luis Pardo Lopez | x |
| Guillermo Pavon Vargas | x |
| Juan Pablo Portero Montaño | x |
| Moises Romero Romo | x |
| Fernando Wals Rodriguez | x |

1.Orden del día

* 1. Revisión de avances

2.Temas tratados

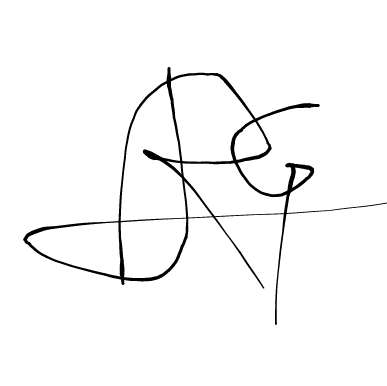
Para  que todo el equipo esté al tanto de los avances se muestran, de cada miembro del trabajo realizado en los últimos días. Al mostrar el trabajo surgen algunas dudas sobre la mecánica de la aplicación que se resuelven para todo el equipo.

Se recuerda que se debe investigar las tecnologías que se van a utilizar en el desarrollo para facilitar el desarrollo y evitar problemas más adelante.

Se realiza el UML de la aplicación en conjunto para poner en común las distintas

ideas de los miembros del equipo.

3. Cierre de la reunión

             Francisco Javier Garcia Roales,

### 13.4.4. Acta 3

|  |  |
| --- | --- |
| **Acta nº 3** |  |
| **Fecha** | Dia 10 del 03 de 2021 |
| **Hora de inicio** | 12:45 h |
| **Hora de fin** | 14:00 h |
| **Tipo de reunión** | Reunión seguimiento |

|  |  |
| --- | --- |
| **Nombre** | **Asistencia** |
| Alvaro Jesus Alferez Richarte | x |
| David Caceres Romero | x |
| Marta Diaz Fernandez | x |
| Elena España González | x |
| Gonzalo Fernandez Jimenez | x |
| Francisco Javier Garcia Roales | x |
| Alejandro Gonzalez Martin | x |
| Luis Pardo Lopez | x |
| Guillermo Pavon Vargas | x |
| Juan Pablo Portero Montaño | x |
| Moises Romero Romo | x |
| Fernando Wals Rodriguez |  |

1.Orden del día

* 1. Revisión del feedback
  2. Creación nuevas tareas
  3. Reparto de tareas

2.Temas tratados

Debido a que el Project Manager no puede asistir a la reunión, se encarga de dirigirla Francisco Javier Garcia como Scrum Master.

Se decide hacer un cambio en la imagen de la aplicación para diferenciarla de otras como Steam. Se considera el feedback relacionado con la claridad del logo llegando a la conclusión de que habría que asegurar que se vea más claramente durante la presentación para evitar confusiones.

En cuanto al análisis de competidores hubo algunos parámetros que no estuvieron bien planteados como la reducción de burocracia por lo que se plantea hacer un análisis más exhaustivo de la que se lleva a cabo en nuestros competidores.

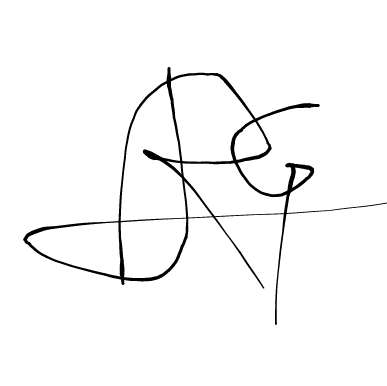
Se acuerda por mayoría absoluta eliminar la subida del precio de suscripción en proporción al éxito que se tenga puesto que provocaría que estos clientes se fuesen de la aplicación. Para solventar estos problemas se recalca la necesidad de estudiar la carga inicial necesaria.

Las fotos de equipo deben ser iguales por lo que se decide llevar camisa blanca y utilizar un fondo neutro de color blanco.

Para asegurar que el proyecto esté correctamente organizado se decide realizar un documento de planificación incluyendo gestión de alcance, comunicaciones, costes y calidad.

Se decide probar Heroku para desplegar el proyecto puesto que en caso de ser necesario se pueden crear más cuentas. Se recuerda que debe usarse Github, Travis y se añade como tecnologías a analizar. Se realiza el reparto de tareas en Trello.

3. Cierre de la reunión

             Francisco Javier Garcia Roales,

### 13.4.5. Acta 4

|  |  |
| --- | --- |
| **Acta nº 4** |  |
| **Fecha** | Dia 13 del 03 de 2021 |
| **Hora de inicio** | 10:35 h |
| **Hora de fin** | 12:10h |
| **Tipo de reunión** | Reunión seguimiento |

|  |  |
| --- | --- |
| **Nombre** | **Asistencia** |
| Alvaro Jesus Alferez Richarte | x |
| David Caceres Romero | x |
| Marta Diaz Fernandez | x |
| Elena España González | x |
| Gonzalo Fernandez Jimenez | x |
| Francisco Javier Garcia Roales | x |
| Alejandro Gonzalez Martin | x |
| Luis Pardo Lopez | x |
| Guillermo Pavon Vargas | x |
| Juan Pablo Portero Montaño | x |
| Moises Romero Romo | x |
| Fernando Wals Rodriguez | x |

1.Orden del día

* 1. Revisión de las tareas realizadas.
  2. Trabajo pendiente

2.Temas tratados

Fernando muestra el set up del proyecto en github que ha realizado por lo que se pide al resto del equipo que envíe sus cuentas de github para que puedan contribuir al proyecto.

Cada componente del equipo muestra los avances que han hecho en las tareas que han realizado, se recuerda que deben marcarse en Trello las tareas realizadas para poder llevar un buen control de los avances.

Se recomienda al equipo de front end familiarizarse con librerías que conectan React con Spring.

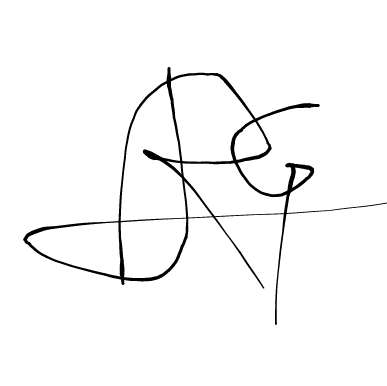
Moisés comenta cambios en el método de suscripción para maximizar las posibles ganancias de forma que el cliente pueda aumentar su suscripción para subir 5 juegos más. Para motivar al cliente a continuar pagando la suscripción se retiraría el soporte de la aplicación a sus juegos si dejase de pagar.

Alvaro muestra la landing page que ha desarrollado para que todo el equipo este familiarizado con la misma. Puesto que todos los miembros han investigado las tecnologias que tendran que utilizar, Fernando muestra un proyecto de prueba que ha creado para mostrar como se conectan los distintos componentes.

Para mejorar la presentación del proximo martes Fernando pide al resto del equipo consejos para solventar algunos errores que pudo cometer en la anterior y mejorar el ritmo y tono de la presentación.

De cara a la entrega del lunes se realiza un indice del documento de planificación a entregar para asegurar que no se deja nada atrás y se decide entregar documentos adyacentes a la planificación como anexos a está.

3. Cierre de la reunión

             Francisco Javier Garcia Roales,

## 13.5. Tecnologías usadas, ALM

Las tecnologías que vamos a usar para el desarrollo de la aplicación son Spring para el Backend, React para el Frontend y MongoDB para la Base de Datos.

Pero además de las tecnologías mencionadas vamos a usar también GitHub como repositorio para alojar el código, Travis para la integración contínua del código, y GitHub Projects para coordinar y seguir el avance de nuestro proyecto. Por último, en cuanto al despliegue tenemos pensado usar para las aplicaciones Heroku, aunque esta última decisión no es definitiva, y MongoDB Atlas para desplegar la base de datos.

Para complementar estas tecnologías también usamos Google Drive para alojar la documentación que realizamos, y desde el principio del proyecto hemos usado Trello aunque ésta probablemente cambiará a GitHub Projects como se ha mencionado con anterioridad.

Ninguna de estas tecnologías nos supone un coste económico y nos ofrecen una buena solución para las necesidades que nos plantea el proyecto.

En cuanto al desarrollo del proyecto, en el equipo de Frontend será de utilidad formar al menos con lo básico a Elena y Alejandro ya que no son proficientes con React, al igual que en el Backend habrá que formar básicamente a Moisés. Esto se realizará para que puedan desempeñar sus tareas de forma eficiente y el equipo entero pueda llevar más o menos el mismo ritmo de trabajo.

### 13.5.1. Justificación

* Con respecto al Backend hemos decidido utilizar Spring por diferentes razones:
* Es una tecnología que ya ha sido usada por casi todos los miembros del equipo y, por ello, nos desenvolvemos mejor usando Spring que otro tipo de tecnologías para el Backend.
* Spring es muy flexible, y eso nos permite realizar prácticamente cualquier idea que se nos ocurra con respecto a la aplicación.
* Es una tecnología muy rápida, productiva y segura, por lo que encaja muy bien con el resultado final que esperamos de la aplicación y consideramos que es una de las mejores opciones para conseguir este fin.
* En el caso del Frontend vamos a utilizar React por las razones mencionadas a continuación:
* React es una tecnología tanto simple como fácil de aprender, y esto permite que con poco tiempo de adaptación al uso de React nos centremos más en las cosas que producimos en lugar de terminar de aprender una tecnología nueva.
* Las actualizaciones de React son muy simples y son testeadas en la propia aplicación de Facebook, por lo que incrementa la confianza que podemos depositar en esta tecnología.
* Además, React tiene un gran rendimiento debido a que minimiza los cambios del DOM utilizando una Virtual DOM.
* Por último, React es muy fácil de testear, y eso nos permitiría tener código de una gran calidad en la aplicación.
* Como base de datos vamos a usar MongoDB por las siguientes razones:
* Como mencionamos con anterioridad la calidad es uno de nuestros objetivos principales a la hora de desarrollar la aplicación, y MongoDB nos permite realizar queries de forma más rápida que con otro gestor de bases de datos.
* MongoDB nos permite realizar modificaciones de manera mucho más sencilla ya que no se basa en esquemas y permite la modificación de los datos a medida que se desarrolla la aplicación.
* Para finalizar, algo que consideramos muy importante es el poder escalar la aplicación en un futuro, y MongoDB en lugar de escalar “verticalmente”, cosa que requiere de servidores muy pesados, permite escalar “horizontalmente” e ir añadiendo nuevos servidores del tamaño que nos convenga a la hora de escalar la aplicación, además de que no tendríamos que dejar sin funcionamiento a la aplicación a la hora de añadir nuevos servidores.

### 13.5.2. Análisis de despliegues

El proyecto InDvelopers tendrá que ser desplegado y, cumpliendo con los estándares de calidad, estar disponible 24/7 (24 horas al día, 7 días a la semana). Además, tendrá que tener cinco versiones diferentes desplegadas al mismo tiempo (una para cada iteración del proyecto). Bajo estas condiciones, se han valorado varias opciones para el despliegue de la aplicación como son Zeit, Github Pages, Heroku o Netlify, resultando finalmente en la elección de Heroku para este objetivo. Esta elección se debe principalmente a cuatro motivos que hace que destaquen sobre el resto:

Plan gratuito:

Las ofertas incluidas en el plan gratuito de Heroku cumplen con lo necesario para nuestro proyecto y es, con diferencia y por esta razón, la mejor opción de cuantas plataformas de despliegue se ha investigado.

Posibilidad de versionado

Mediante la creación de una aplicación nueva para cada iteración se espera conseguir que se pueda tener desplegadas las cinco versiones del proyecto simultáneamente. Si se diese el caso de que, por restricciones de la plataforma, la persona encargada del despliegue no pudiera realizar tal cosa, se propondría un nuevo candidato que se encargase del despliegue de esa versión y así sucesivamente a lo largo de las iteraciones.

Comodidad de despliegue

Dado que el proyecto deberá integrarse con Travis-CI, se procurará permitir un despliegue automático de la aplicación cada vez que se pasen las pruebas correctamente.

Familiaridad del equipo

Todo el equipo ha trabajado en el pasado con esta plataforma, al contrario que con las otras ya mencionadas.

La mayor desventaja de Heroku es su carencia de base de datos decentes en su plan gratuito, por lo que se ha estudiado la posibilidad de desplegar la base de datos del proyecto en MongoDB Atlas y conectarla a la aplicación desde ahí.

## 13.6. Respuestas de los formularios de nuestros usuarios piloto

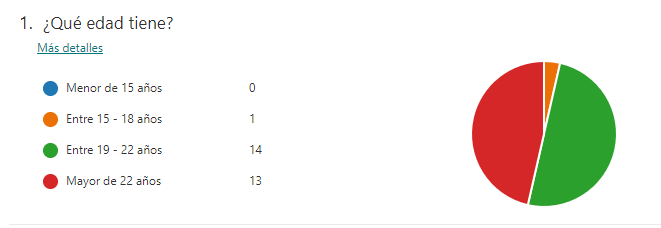


Ilustración : Gráfica de la pregunta 1

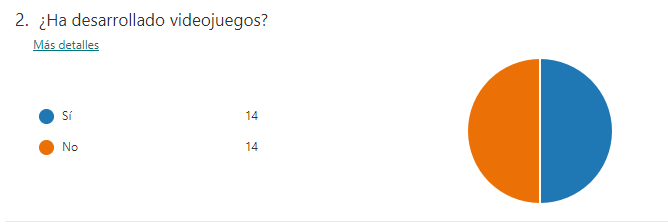


Ilustración : Gráfica de la pregunta 2

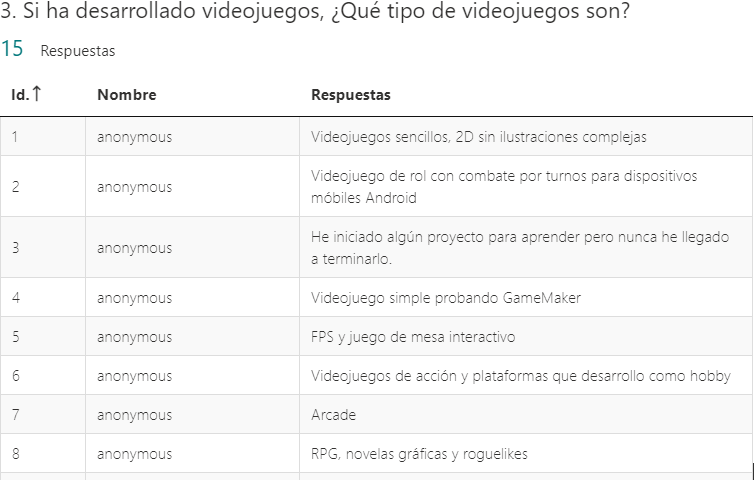


Ilustración : Respuestas a la pregunta 3

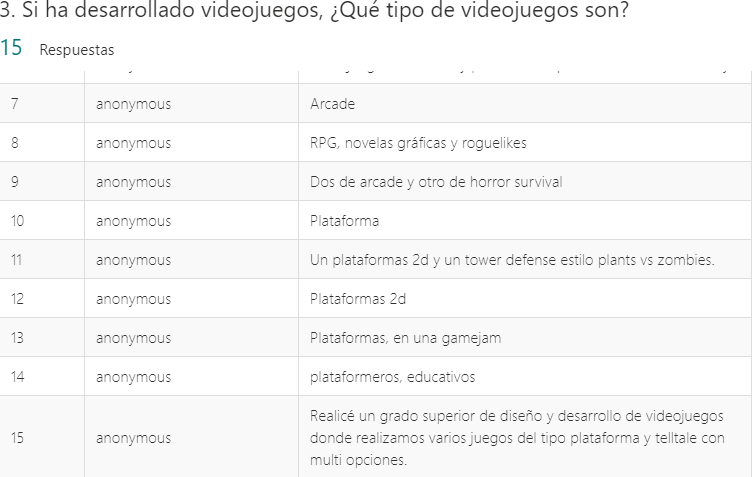


Ilustración : Pregunta 3, parte 2

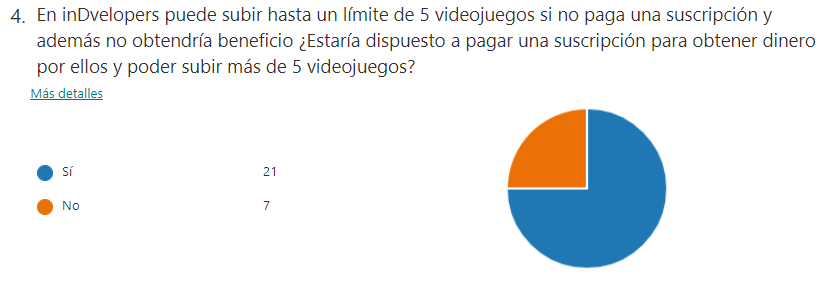


Ilustración : Gráfica de resultados de la pregunta 4

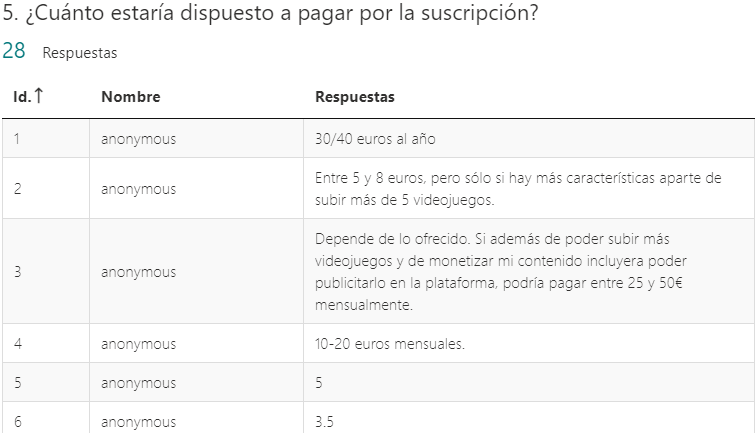


Ilustración :Respuestas de pregunta 5

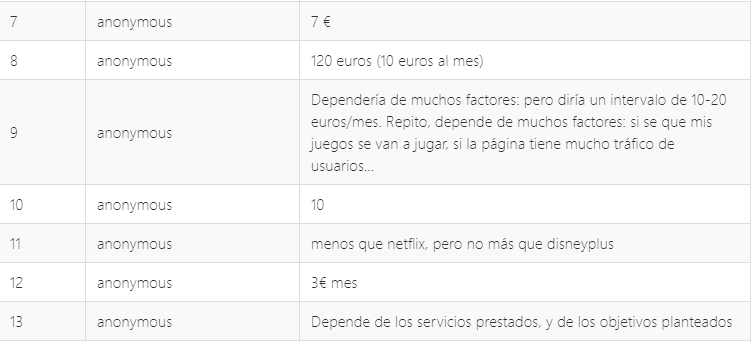


Ilustración : Respuestas de pregunta 5, parte 2

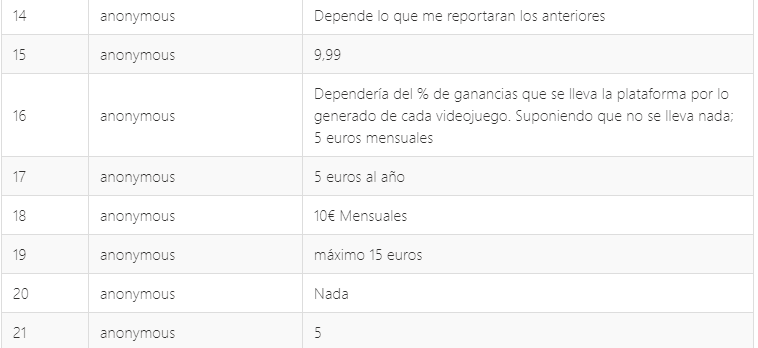


Ilustración : Respuestas de preguntas, parte 3

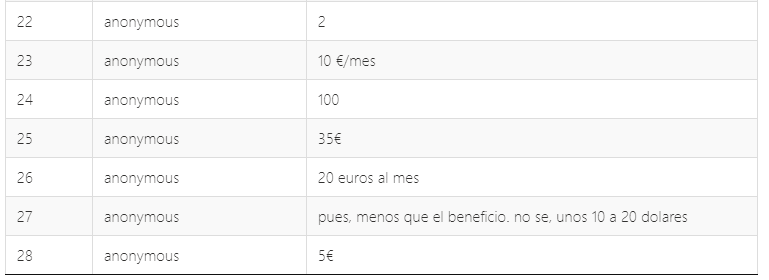


Ilustración : Respuestas de preguntas, parte 4

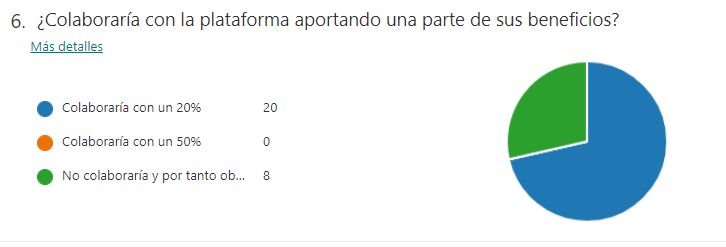


Ilustración : Gráfica de los resultados de la pregunta 6

## 13.7. Tabla de beneficios

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Mes | Usuarios | Optimista | Neutro | Pesimista | Inversión | Inversión + Burocracia |
| 1 | 10 | 120 | 90 | 60 | 128.904 | 130.704 |
| 2 | 20 | 360 | 270 | 180 | 128.904 | 132.504 |
| 3 | 50 | 960 | 720 | 480 | 128.904 | 134.304 |
| 4 | 100 | 2240 | 1680 | 1120 | 128.904 | 136.104 |
| 5 | 200 | 4800 | 3600 | 2400 | 128.904 | 137.904 |
| 6 | 300 | 8640 | 6480 | 4320 | 128.904 | 139.704 |
| 7 | 400 | 14.400 | 10800 | 7.200 | 128.904 | 141.504 |
| 8 | 500 | 21.600 | 16200 | 10.800 | 128.904 | 143.304 |
| 9 | 600 | 30.240 | 22680 | 15.120 | 128.904 | 145.104 |
| 10 | 700 | 41.440 | 31.080 | 20.720 | 128.904 | 146.904 |
| 11 | 800 | 54.240 | 40.680 | 27.120 | 128.904 | 148.704 |
| 12 | 900 | 68.640 | 51.480 | 34.320 | 128.904 | 150.504 |
| 13 | 1.000 | 84.640 | 63.480 | 42.320 | 128.904 | 152.304 |
| 14 | 1.100 | 102.240 | 76.680 | 51.120 | 128.904 | 154.104 |
| 15 | 1.200 | 121.440 | 91.080 | 60.720 | 128.904 | 155.904 |
| 16 | 1300 | 142.240 | 106.680 | 71.120 | 128.904 | 157.704 |
| 17 | 1400 | 164.640 | 123.480 | 82.320 | 128.904 | 159.504 |
| 18 | 1500 |  | 141.480 | 94.320 | 128.904 | 161.304 |
| 19 | 1600 |  | 160.680 | 107.120 | 128.904 | 163.104 |
| 20 | 1700 |  | 181.080 | 120.720 | 128.904 | 164.904 |
| 21 | 1800 |  |  | 135.120 | 128.904 | 166.704 |
| 22 | 1900 |  |  | 150.320 | 128.904 | 168.504 |
| 23 | 2000 |  |  | 166.320 | 128.904 | 170.304 |
| 24 | 2020 |  |  | 182.480 | 128.904 | 172.104 |

Tabla 17: Tabla de beneficios

Esta tabla muestra los beneficios en los casos optimista, neutro y pesimista. En el caso optimista tenemos lo que ganaríamos si todos los usuarios de ese mes pagasen por la subscripción más alta, en el neutro si pagasen un 75% de los usuarios y en el pesimista si son el 50%. La columna de “inversión+burocracia” indica lo que nos costaría con el paso del tiempo el trámite burocrático.

Como podemos ver, en el caso optimista recuperaríamos la inversión en el mes 17. En el neutro en el mes 20 y en el pesimista en 2 años.

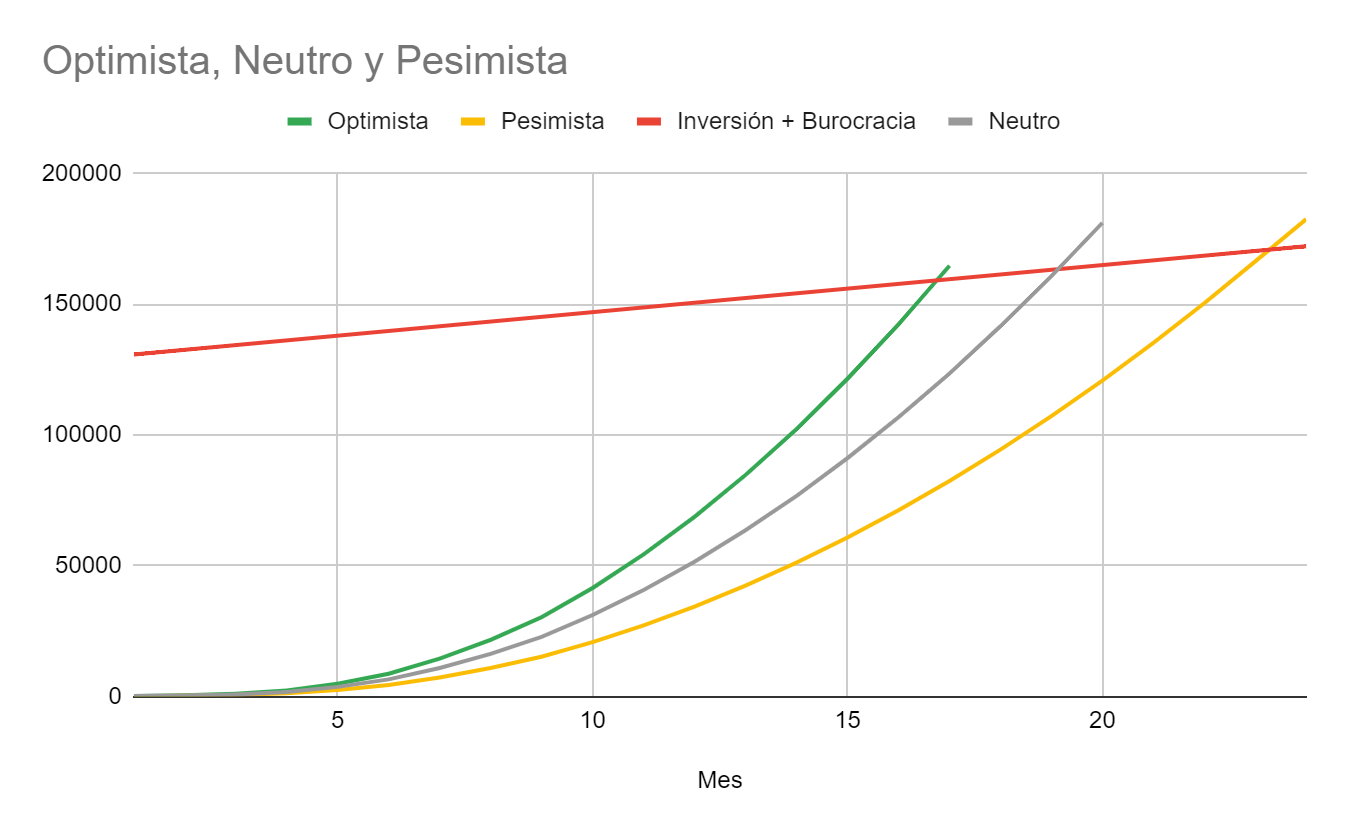


Ilustración : Gráfica de beneficios