



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

UDARE: Desarrollo del front-end de una red social para  
promover hábitos saludables

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Fabra García, Javier

Tutor/a: Letelier Torres, Patricio Orlando

CURSO ACADÉMICO: 2023/2024

*A mis padres, quienes con su amor y apoyo incondicional me han impulsado siempre a seguir adelante. A mi familia, por estar siempre a mi lado en los buenos y malos momentos. Y a mis amigos, por su compañía y aliento en este camino. A todos, gracias de corazón*

# Agradecimientos

---

A Patricio Letelier, por brindarme la posibilidad de desarrollar un TFG de emprendimiento, además de introducirnos al desarrollo software mediante las asignaturas de PSW y PIN.

A David Sanjaime, por ser un gran compañero de TFG, y por hacer más fácil y llevadero este proyecto.

A los participantes de los experimentos, por todas las encuestas, opiniones y críticas constructivas aportadas, por dedicar todo el tiempo requerido sin ánimo de lucro y por toda la paciencia que han tenido.

Y, sobre todo, a mi familia, cuyo amor y apoyo incondicional me han acompañado en todo momento, ayudándome a superar esta etapa de mi vida.

# Resumen

---

Las redes sociales han experimentado un gran crecimiento en los últimos años, convirtiéndose en un elemento imprescindible y esencial en la vida de muchos jóvenes. Este tipo de aplicaciones tienen una gran capacidad para transformar la forma en la que interactuamos y nos relacionamos con las demás personas. Además, suponen una poderosa herramienta para influir en el comportamiento y pensamiento de los usuarios que las consumen.

Este TFG abordará el desafío que supone crear una red social en el contexto de un proyecto de emprendimiento. Esta red social se centrará en promover hábitos saludables y en mejorar el estilo de vida de las personas. Se expondrá con detalle todo el proceso asociado al desarrollo completo de una idea de negocio, que abarcará desde la generación y evaluación de la idea de negocio hasta la descripción de los aspectos más técnicos y desafíos tecnológicos a los que nos hemos tenido que enfrentar.

Cabe resaltar que este TFG se ha desarrollado en el marco de Start.inf, el espacio de emprendimiento de la ETSINF. También hay que destacar que este TFG se ha realizado conjuntamente, dónde el autor de este trabajo final de grado, Javier Fabra García se encargará del desarrollo Front-end, mientras que David Sanjaime Bayarri será responsable del Back-end.

**Palabras clave:** emprendimiento, red social, estilo de vida, hábitos saludables.

## Abstract

---

Social media has experienced significant growth in recent years, becoming an indispensable and essential element in the lives of many young people. These applications have a great capacity to transform the way we interact and relate with others, and they work as powerful tools for influencing the behavior and thinking of their users.

This Final Degree Project (TFG) will address the challenge of creating a social network within the context of an entrepreneurial project. This social network will primarily focus on promoting healthy habits and improving people's lifestyles. The project will detail the entire process associated with the complete development of a business idea, encompassing everything from the generation and evaluation of the business idea to the description of the most technical aspects and technological challenges we have faced.

It should be noted that this TFG has been developed within the framework of Start.inf, the entrepreneurial space of ETSinf. It should also be noted that this final project (TFG) has been carried out jointly, with the student Javier Fabra García in charge of Front-end development, while David Sanjaime Bayarri is responsible for the Back-end.

**Keywords:** entrepreneurship, social network, lifestyle, healthy habits.

# Índice de contenidos

---

## Tabla de contenido

1. INTRODUCCIÓN.....	9
1.1 Motivación.....	9
1.2 Objetivos.....	11
1.3 Estructura de la memoria.....	12
2. EVALUACIÓN DE LA IDEA DE NEGOCIO.....	13
2.1 Clientes.....	13
2.2 Estudio de mercado y competidores.....	14
2.2.1 Rabbit.....	15
2.2.2 Challenge Accepted.....	17
2.2.3 21 Days.....	19
2.2.4 Tabla comparativa.....	22
2.3 Proyección de ingresos y gastos.....	24
2.4 Análisis DAFO.....	27
2.5 Lean Canvas.....	28
2.6 Conclusiones de la evaluación.....	29
3. TECNOLOGÍAS UTILIZADAS.....	31
3.1 Lenguaje de programación.....	31
3.2 Entorno de desarrollo.....	32
3.3 GITHUB.....	33
3.4 FIGMA.....	34
4. DESARROLLO DE LA SOLUCIÓN.....	36
4.1 Metodología.....	36
4.2 Requisitos.....	38
4.3 Diseño de la interfaz de usuario.....	43
4.3.1 Interfaz de usuario.....	43
4.3.2 Experiencia de usuario.....	46
4.4 Diseño del código y la base de datos.....	47
4.4.1 Arquitectura.....	48
4.4.2 Diagrama de clases.....	49
4.4.3 Diagrama de despliegue.....	50
4.4.4 Comunicación con el <i>backend</i> .....	51
4.5 Programación.....	52
4.5.1 Patrones de diseño.....	52

4.5.1.1 Singleton.....	53
4.5.1.2 Adapter .....	53
4.5.2 Refactoring .....	55
4.5.3 Desafíos de programación.....	56
4.6 Pruebas .....	57
5. CRONOLOGÍA DEL TFG.....	59
5.1 Desarrollo del primer MVP .....	59
5.2 Primer experimento .....	60
5.3 Desarrollo del segundo MVP .....	67
5.4 Segundo experimento .....	71
6. CONCLUSIONES Y TRABAJO FUTURO.....	75
6.1 Conclusiones .....	75
6.2 Trabajo futuro .....	76
REFERENCIAS .....	77
ANEXO.....	79
Anexo A.....	79
Anexo B.....	88



# Índice de figuras

Figura 1: Pantallas de progreso .....	15
Figura 2: Pantalla de rachas.....	16
Figura 3: Pantallas de perfil y plan premium.....	16
Figura 4: Pantallas de la aplicación Challenge Accepted .....	17
Figura 5: Pantallas de la aplicación Challenge Accepted .....	18
Figura 6: Pantallas de la aplicación Challenge Accepted .....	18
Figura 7: Pantallas de la aplicación 21 days.....	19
Figura 8: Pantalla aplicación 21 Days.....	20
Figura 9: Pantalla aplicación 21 Days.....	20
Figura 10: Pantalla aplicación 21 Days.....	21
Figura 11: Pantalla aplicación 21 Days.....	21
Figura 12: Gráfico de ingresos, gastos y total acumulado.....	26
Figura 13: Repositorio del front-end.....	34
Figura 14: Imagen global del proyecto figma.....	35
Figura 15: Tablero Kanban Trello primer sprint.....	38
Figura 16: Logotipo e imagotipo.....	44
Figura 17: Paleta de colores y tipografías .....	44
Figura 18: Iconos e ilustraciones .....	45
Figura 19: Feed principal y pantalla de perfil .....	45
Figura 20: Ranking.....	46
Figura 21 : Nuevos iconos .....	46
Figura 22 : Diagrama de actividades.....	47
Figura 23: Estructura del proyecto.....	48
Figura 24: Diagrama de clases .....	49
Figura 25: Diagrama de despliegue .....	50
Figura 26: Ejemplos uso RESTful.....	52
Figura 27: Métodos de la clase UserSingleton .....	53
Figura 28: RecyclerView vacío.....	54
Figura 29: Ítems RecyclerView.....	54
Figura 30: Ejemplo de la carga de imágenes mediante Glide.....	55
Figura 31: Ejemplo de seguimiento de pruebas de aceptación .....	58
Figura 32: Funcionalidades primer MVP .....	59
Figura 33: Funcionalidades por prioridad primer experimento.....	60
Figura 34: Pregunta cuestionario sobre usabilidad.....	62
Figura 35: Pregunta cuestionario sobre la interfaz .....	62
Figura 36: Pregunta cuestionario sobre el interés en la app .....	63
Figura 37: Pregunta cuestionario sobre apps similares .....	63
Figura 38: Pregunta cuestionario sobre el precio del premium.....	64
Figura 39: Pregunta cuestionario sobre el ranking.....	64
Figura 40: Pregunta cuestionario sobre las rachas .....	65
Figura 41: Pregunta cuestionario sobre los retos de comunidad .....	65
Figura 42: Pregunta cuestionario sobre mejores amigos.....	65
Figura 43: Pregunta cuestionario sobre funcionalidades que no gustan.....	66
Figura 44: Pregunta cuestionario sobre funcionalidades nuevas .....	66
Figura 45: Pregunta cuestionario .....	67

Figura 46: Funcionalidades por prioridad segundo MVP .....	67
Figura 47: Tablero Kanban Trello segundo sprint.....	69
Figura 48: Cronología del proyecto .....	70
Figura 49: Preguntas sobre el ranking .....	72
Figura 50: Preguntas sobre el editor de fotos.....	72
Figura 51: Preguntas sobre las rachas de retos.....	73
Figura 52: Preguntas sobre los retos de comunidad .....	73
Figura 53: Preguntas sobre nuevas funcionalidades.....	74
Figura 54: Pantallas de carga y landing page.....	79
Figura 55: Pantallas de inicio de sesión y registro.....	80
Figura 56: Feed principal.....	80
Figura 57: Cámara y pantalla comentarios .....	81
Figura 58: Pantalla del perfil.....	82
Figura 59: Cerrar sesión y perfil ajeno .....	82
Figura 60: Pantalla álbum .....	83
Figura 61: Pantalla ranking .....	83
Figura 62: Pantalla trofeos .....	84
Figura 63: Pantalla sugerencias.....	85
Figura 64: Pantalla seguidores.....	85
Figura 65: Pantalla siguiendo .....	86
Figura 66: Pantalla seleccionar reto .....	86
Figura 67: Pantalla filtros y confirmación.....	87





# Índice de tablas

---

Tabla 1: Comparativa competidores.....	23
Tabla 2: Planificación ingresos y gastos.....	26
Tabla 3: Análisis DAFO .....	27
Tabla 4: Lean Canvas.....	29

# 1. INTRODUCCIÓN

---

En este capítulo introductorio se presentarán las bases que han impulsado la creación de este proyecto. Para ello, se contextualizará nuestra idea de negocio dentro de la situación actual de las redes sociales, se expondrá el valor añadido que ofrecemos con nuestra aplicación y como esta se puede introducir en un mercado tan competitivo como es el de las redes sociales. Además, se detallarán los objetivos del proyecto, tanto generales como específicos, proporcionando una visión clara de lo que se pretende alcanzar con este TFG. En definitiva, en este capítulo se establecerán las bases necesarias para entender la importancia y propósito de nuestro proyecto.

## 1.1 Motivación

Vivimos en un mundo cada vez más interconectado, dónde las redes sociales se han convertido en una parte esencial de nuestras vidas. Estas han cambiado radicalmente la forma en la que nos relacionamos, comunicamos y consumimos información.

Desde Facebook<sup>1</sup> y Youtube<sup>2</sup> hasta LinkedIn<sup>3</sup>, estas plataformas nos permiten estar conectados de manera instantánea con amigos, familiares y personas de todo el mundo. A través de ellas, podemos compartir nuestros pensamientos, emociones, logros y experiencias en tiempo real.

Una de las mayores fortalezas de las redes sociales es su capacidad para llegar a un número tan grande de personas, y a su vez, de influir en ellas. Gracias a esta tecnología los usuarios pueden acceder a una amplia gama de personas con intereses similares. Ya sea promocionando productos y servicios, difundiendo información o simplemente compartiendo momentos del día a día, las redes sociales ofrecen un canal de comunicación muy potente y efectivo para llegar a una audiencia cada vez más globalizada.

A pesar de los numerosos beneficios asociados a estas tecnologías, también han surgido una serie de desafíos y problemas asociados a su uso excesivo, especialmente entre los más jóvenes. La dependencia de los jóvenes a las tecnologías y redes sociales se ha acentuado mucho en los últimos años, llevando a generar preocupantes problemas de adicción. Según un estudio de la *American Psychological Association*<sup>4</sup>, un porcentaje significativo de adolescentes ha mostrado signos de adicción a las redes sociales. Concretamente, el estudio revela que, en promedio, los adolescentes en Estados Unidos pasan de media unas 4,8 horas diarias utilizando las redes sociales más

---

<sup>1</sup> Facebook: <https://www.facebook.com>

<sup>2</sup> Youtube: <https://www.youtube.com>

<sup>3</sup> LinkedIn: <https://www.linkedin.com>

<sup>4</sup> American Psychological Association: <https://www.apa.org>

populares en la actualidad, donde Youtube, TikTok<sup>5</sup> e Instagram<sup>6</sup> constituyen el 87% del tiempo total frente a la pantalla [1].

Además, la sobreexposición a las redes sociales ha sido asociado con una gran variedad de efectos negativos en la salud mental de los usuarios que las consumen, siendo objeto de numerosos estudios y debates. Los riesgos asociados son numerosos y están ampliamente documentados en numerosos artículos científicos. Un metaanálisis publicado en *BMC Psychology*<sup>7</sup>, encontró una correlación significativa entre el uso excesivo de las redes sociales y un aumento considerable de los niveles de ansiedad y síntomas depresivos entre adolescentes y adultos jóvenes [2]. Por otra parte, un estudio publicado en *BMC Public Health*<sup>8</sup> señala que el uso excesivo de estas tecnologías tiende a evadir al usuario del mundo exterior, haciéndole propenso al aislamiento y reemplazando las interacciones sociales con su familia y amigos por tiempo en frente de la pantalla [3].

Otro importante problema que observamos actualmente en la población es el gran número de fracasos que hay de personas que intentan realizar un cambio a mejor en su vida, pero que no lo consiguen por falta de disciplina y motivación. Esto es multifactorial, pero en gran parte se debe al tiempo excesivo que le dedicamos a las redes sociales, las cuales no solo nos consumen una gran cantidad de nuestro tiempo, sino que también influye muy negativamente en el sistema de recompensas de nuestro cuerpo. La exposición prolongada a estos estímulos superficiales y a la gratificación instantánea que nos produce consumir contenido en redes sociales hace que nos resulte más difícil perseguir metas a largo plazo y adoptar hábitos saludables de manera sostenible en el tiempo [4].

A raíz de todos estos problemas, ha surgido una tendencia cada vez más creciente hacia los estilos de vida más sostenibles y saludables. Cada vez más personas se están dando cuenta de lo perjudicial que puede llegar a ser abusar de las redes sociales y están buscando maneras de intentar evadirse del mundo digital y volver a conectar con la realidad física y presente. Esta tendencia global también se ve reflejada en la cada vez mayor concienciación que se tiene acerca de la importancia de abordar los Objetivos de Desarrollo Sostenible<sup>9</sup>.

Es en este contexto donde surge nuestra motivación y la idea de negocio, la creación de una red social que intenta abordar todos estos problemas. Nuestra aplicación trata de ofrecer un equilibrio entre el entretenimiento de las redes sociales y el fomento de un estilo de vida más saludable a través de los hábitos. Creemos que la mejor forma de lograr cambios sustanciales en la vida de las personas y que estas se mantengan motivadas y comprometidas con su desarrollo es a través del componente social que ofrecen las redes sociales. La clave fundamental radica en identificar los aspectos positivos de las redes sociales y utilizarlos a nuestro favor para cambiar la vida de las

---

<sup>5</sup> TikTok: <https://www.tiktok.com>

<sup>6</sup> Instagram: <https://www.instagram.com>

<sup>7</sup> BMC Psychology: <https://bmcpyschology.biomedcentral.com/>

<sup>8</sup> BMC Public Health: <https://bmcpublikealth.biomedcentral.com/>

<sup>9</sup> Objetivos de Desarrollo Sostenible (ODS): <https://sdgs.un.org/goals>

personas, minimizando así al mínimo los problemas inherentes de este tipo de tecnologías.

## 1.2 Objetivos

Nuestro objetivo principal es desarrollar una aplicación móvil para Android<sup>10</sup> llamada UDARE dentro del contexto de un proyecto de emprendimiento. Nuestra aplicación será una red social enfocada en promover hábitos saludables y mejorar el estilo de vida de las personas. La idea es desarrollar una segunda versión mejorada a partir de una primera versión muy básica de la aplicación que se desarrolló en la asignatura Proyecto de Ingeniería de Software (PIN), de la rama de Ingeniería del Software. El objetivo es llevar a cabo tareas de refactorización para mejorar la calidad del código, y lo más importante, implementar nuevas funcionalidades según las necesidades y sugerencias de los usuarios. El objetivo final será desplegar la aplicación en un entorno de producción para que los usuarios puedan utilizarla sin ningún inconveniente.

Cabe resaltar que hay que hacer una clara diferenciación entre los objetivos generales relacionados con el proyecto de emprendimiento, y los objetivos específicos de este TFG. Los objetivos del proyecto de emprendimiento son muy ambiciosos y se salen del alcance de este TFG. Así mismo, los objetivos del TFG son metas más acotadas y realistas que contribuyen al desarrollo del proyecto, pero que solo abordarán una parte de ellos, debido a la limitación de horas de dedicación correspondientes a un TFG.

Los objetivos de este TFG son los que se detallan a continuación:

- Desarrollar el *front-end* de UDARE.
- Desplegar la aplicación en un entorno de producción.
- Realizar un segundo MVP<sup>11</sup> según las necesidades de los usuarios.

Los objetivos específicos de nuestra red social son los siguientes:

- Promover la actividad física, saludable, social y cultural.
- Evadir al usuario del ámbito digital.
- Animar al usuario a nutrirse y crecer en la realidad física y presente.
- Motivar a las personas a realizar retos diarios entretenidos y gratificantes.
- Impulsar la motivación y competencia mutua entre personas.

---

<sup>10</sup> Android: <https://www.android.com/>

<sup>11</sup> MVP: "Minimum Viable Product" que, en español, significa "Producto Mínimo Viable". No es más que la versión más sencilla de un producto y, por tanto, presenta sus funcionalidades más básicas. Funciona como prueba para validar la idea, aplicabilidad y adherencia de un producto en el mercado.



### 1.3 Estructura de la memoria

La memoria se estructura en diferentes capítulos enfocados cada uno en un aspecto del proyecto. En el índice podemos ver un avance de cómo se ha constituido, pero aquí explicaremos más en detalle cada sección:

En el capítulo dos nos encontraremos con la evaluación de la idea de negocio, donde se explica el estudio realizado para justificar si el emprendimiento es rentable o no, cómo debemos diseñar la estrategia para triunfar.

El tercer capítulo trata de las diferentes fases de proyecto, desde sus inicios al punto inicial. En él explicamos los avances durante el tiempo, los diferentes productos mínimos viables (MVP) realizados y los experimentos realizados sobre ellos.

En cuarto lugar, explicaremos las decisiones tomadas en cuanto a las tecnologías utilizadas para llevar a cabo la aplicación, tanto en lenguajes de programación, cómo en entornos de desarrollo y herramientas complementarias.

Seguidamente, tenemos los aspectos técnicos, apartado en el que se expone cómo se ha desarrollado la aplicación. Aquí explicaremos los diferentes desafíos encontrados, patrones de diseño, pruebas y la arquitectura del *front-end* de UDARE.

Finalmente se comentarán las conclusiones del proyecto, y el trabajo futuro de emprendimiento.

## 2. EVALUACIÓN DE LA IDEA DE NEGOCIO

---

La evaluación de la idea de negocio resulta de crucial importancia en el contexto de un proyecto de emprendimiento. Nos permite identificar y analizar los factores claves que van a determinar el éxito o fracaso de nuestro proyecto.

En este apartado se incluirá la identificación de nuestros clientes potenciales y aquellos que están dispuestos a pagar por nuestro producto. También se realizará un estudio exhaustivo del mercado para de esta forma identificar a nuestros competidores y ser capaces de destacarse de ellos. Además, nos apoyaremos con herramientas como el análisis *DAFO* [5] y el *Lean Canvas* [6] que nos ayudarán a sintetizar los puntos anteriores y nos proporcionarán una visión clara y resumida de la viabilidad del proyecto. Finalmente, se extraerán las conclusiones de la evaluación que justifican la viabilidad y la propuesta de valor que estamos incorporando al mercado con nuestra red social.

### 2.1 Clientes

La identificación y comprensión de los clientes desempeñan un papel fundamental. La segmentación de clientes es esencial para adaptar el producto a las necesidades específicas de cada grupo, lo que permite una estrategia de marketing más efectiva. Además, la identificación de los *Early Adopters*<sup>12</sup> puede proporcionar una valiosa retroalimentación que servirá de gran ayuda para perfeccionar y mejorar la aplicación.

En este contexto, el segmento de clientes para nuestra red social abarca a todo tipo de persona interesada en mejorar su estilo de vida mediante la realización de retos diarios. Esto puede incluir a personas preocupadas por su salud y bienestar, amantes del deporte, personas a las que les guste realizar actividades al aire libre o personas comprometidas con el desarrollo personal. La aplicación se dirige a aquellos usuarios que buscan una red social que contribuya a alcanzar sus metas personales y mejorar su estilo de vida, pudiendo compartir todo ese progreso y camino con otras personas. Nuestro público objetivo se encuentra principalmente entre los jóvenes, ya que estos son los más activos y los que más consumen este tipo de aplicaciones, aunque también trataremos de abarcar el máximo público posible, intentando incluir a personas de mayor edad.

En cuanto a las empresas, los potenciales clientes pueden ser aquellas que deseen patrocinar algunos de los retos, o crear los suyos propios. Empresas de productos relacionados con el bienestar, marcas de ropa deportiva, alimentación, y compañías comprometidas con causas sostenibles podrían beneficiarse con el patrocinio de algunos de nuestros retos. También son potenciales clientes las empresas que deseen

---

<sup>12</sup> *Early Adopter*: Individuos o empresas dispuestos a probar versiones preliminares de un producto o servicio antes de que sea ampliamente conocido por el público en general. Resultan de interés ya que pueden proporcionar retroalimentación valiosa para la mejora y desarrollo del producto [20].



publicitarse en nuestra aplicación de manera directa, mediante la inserción de anuncios en el *feed* principal.

Los *Early Adopters*, por otro lado, podrían ser de dos tipos. En primer lugar, estarían los *influencers* en redes sociales enfocados en contenido *fitness* o de desarrollo personal. Estos podrían escoger un grupo reducido de personas de su audiencia (unas 100 personas) para que prueben la aplicación, y respondan a un cuestionario dónde darían *feedback* sobre qué funcionalidades deberían de ser mejoradas y cuales podrían implementarse. La segunda opción, que es la que llevaremos a cabo, será realizar un experimento con personas de confianza de nuestro entorno para que prueben la aplicación y nos den sus opiniones a través un cuestionario. Como nuestros potenciales clientes se encuentran en un público joven, los *Early Adopters* serán personas de un rango de edad parecido al nuestro (de 16 años a 30).

## 2.2 Estudio de mercado y competidores

El mercado de las aplicaciones móviles y en especial el de las redes sociales ha experimentado un crecimiento exponencial en los últimos años. Estas aplicaciones han transformado la manera en la que consumimos información y nos relacionamos con el entorno. Desde el lanzamiento de las primeras aplicaciones como Facebook, hasta la actualidad, este sector ha ido en aumento y evolucionando constantemente para satisfacer la demanda de los usuarios.

Para poder introducirnos en este mercado con éxito debemos realizar un estudio exhaustivo de este para comprender cuales son las tendencias actuales, así como para identificar cuáles son las necesidades y expectativas de los usuarios. Identificar a nuestra competencia en el mercado es fundamental para poder ofrecer un producto innovador y que se diferencie del resto.

Este estudio tiene como objetivo identificar las principales aplicaciones que hay actualmente en el mercado que ofrezcan un servicio parecido al nuestro. De esta forma, podremos determinar cuáles son sus puntos débiles y como nuestra aplicación puede solucionarlos.

Para este estudio se ha optado por analizar las siguientes aplicaciones: Rabit<sup>13</sup>, 21 Days Challenge<sup>14</sup> y Challenge Accepted<sup>15</sup>. Todas ellas comparten la misma idea, que es promover la formación de hábitos más saludables para poder mejorar el estilo de vida de las personas a través de retos. A continuación, se realizará un estudio detallado de cada aplicación:

---

<sup>13</sup> Rabit: <https://apps.apple.com/us/app/rabit-daily-routine-planner/id1512605216>

<sup>14</sup> 21 Days Challenge: <https://21dayschallengeapp.com/>

<sup>15</sup> Challenge Accepted: <https://www.challengeacceptedapp.com/>

### 2.2.1 Rabit



Rabit es una aplicación que se enfoca principalmente en la creación de hábitos saludables, así como en la formación de rutinas. Esta aplicación ofrece a los usuarios la posibilidad de registrar y organizar sus tareas, añadir nuevos hábitos, llevar un registro de tu progreso a través de un calendario y la gamificación del proceso a través de rachas y trofeos.

La aplicación en general ofrece una interfaz muy amigable y que incita al usuario a ser constante y completar sus rutinas. El usuario tiene que introducir manualmente los hábitos que desea implementar en su rutina, y como se observa en la Figura 1, podrá en todo momento visualizar el estado de su progreso.



Figura 1: Pantallas de progreso

En la Figura 2 podemos ver que, para motivar al usuario a que no pierda los hábitos, Rabit utiliza un concepto clave que son las rachas. A medida que vayas completando los hábitos día a día tu racha irá incrementando.





Figura 2: Pantalla de rachas

Como se ve en la Figura 3, la aplicación nos ofrece un plan premium de pago en el que se nos desbloquean muchas funcionalidades. La primera pantalla que aparece en la figura salta muy frecuentemente en la aplicación, llegando a ser un poco intrusivo para el usuario. Además, las funcionalidades básicas gratis que ofrece la aplicación son muy limitadas, especialmente si el usuario hace mucho uso de la aplicación.



Figura 3: Pantallas de perfil y plan premium

## 2.2.2 Challenge Accepted



Challenge Accepted es una aplicación que nos permite hacer un seguimiento de los días en los que se cumple un reto. La aplicación nos da la opción de definir y registrar retos, establecer recordatorios y visualizar el progreso mediante gráficos. La aplicación ofrece también recompensas y permite compartir el progreso con familiares y amigos.

Se elige un tipo de reto, cómo ejercitarse durante 30 días, y dentro de la aplicación dispones de un calendario como el de la primera pantalla de la Figura 4, dónde anotas los días en los que lo has cumplido. Un aspecto para destacar es que se pueden crear, o unirse a equipos para ver el progreso de los componentes de éste en algún reto como en la segunda pantalla de la Figura 4.

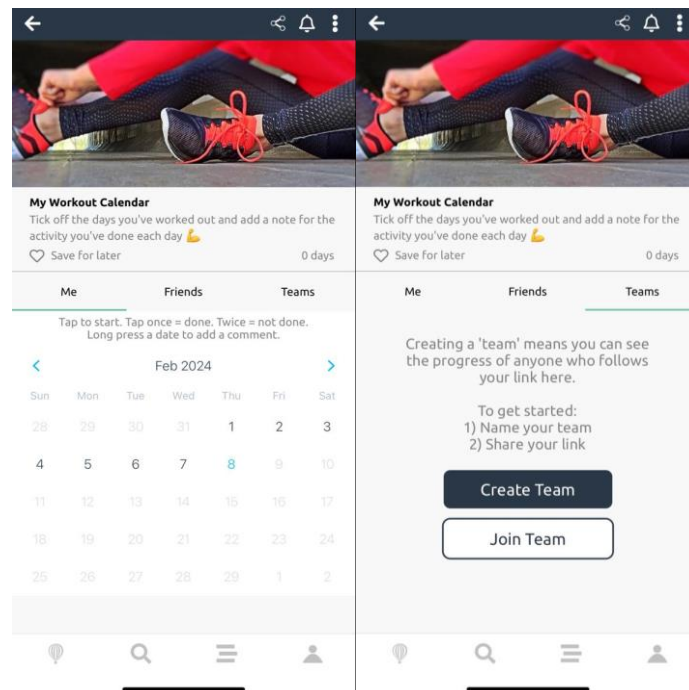


Figura 4: Pantallas de la aplicación Challenge Accepted

La aplicación está formada por 4 apartados, una pantalla inicial observable en la primera pantalla de la Figura 5, dónde se proponen diferentes retos y permite crear nuevos. Otro apartado dónde se pueden buscar directamente los retos mediante un buscador como en la segunda pantalla de la Figura 5.

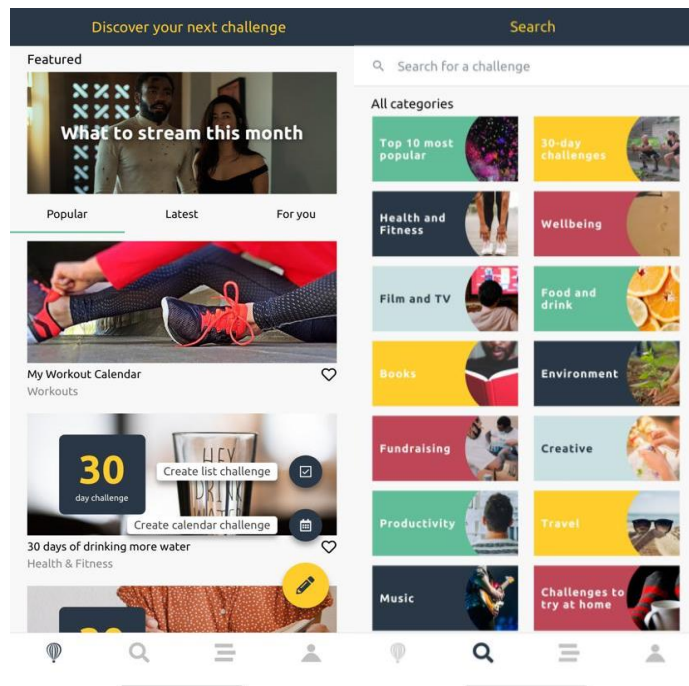


Figura 5: Pantallas de la aplicación Challenge Accepted

Un tercer apartado visible en la primera pantalla de la Figura 6 para hacer un seguimiento de los retos que se están completando y por último en la segunda pantalla de la Figura 6 podemos visualizar el perfil propio, y el de tus amigos.

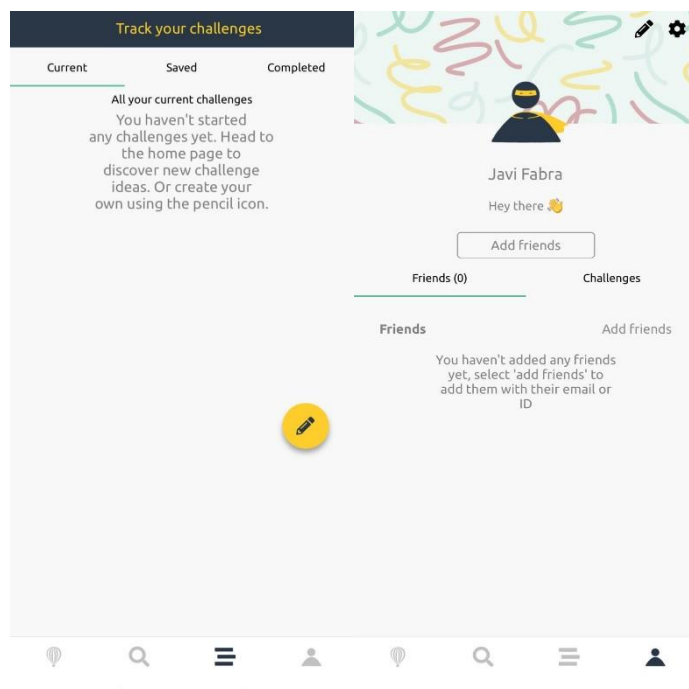


Figura 6: Pantallas de la aplicación Challenge Accepted

Esta aplicación es la más básica y sencilla de todas por lo que no dispone de gran variedad de funcionalidades.

### 2.2.3 21 Days

21 Days Challenge es una aplicación móvil que ofrece desafíos de diversas áreas, que se completan durante 21 días, con el objetivo de crear un hábito.

En esta aplicación podemos elegir los diferentes temas que nos interesan, y posteriormente nos muestra una lista de desafíos a cumplir relacionados con estos temas, como podemos ver en la Figura 7.

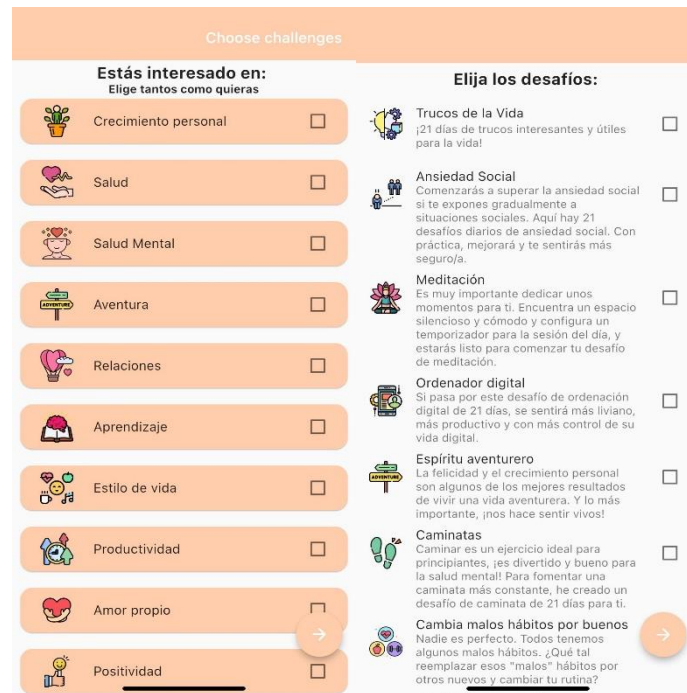


Figura 7: Pantallas de la aplicación 21 days

Tras elegir uno de los desafíos, puedes comenzar a completarlo el día 1, dónde te propondrá la aplicación una meta que deberás de cumplir relacionada con el desafío, y te mostrará un calendario, con los días que has completado, y los que quedan por completar, cómo se puede ver en la Figura 8.



Figura 8: Pantalla aplicación 21 Days

Un aspecto para destacar de esta aplicación es que la comunidad puede interactuar entre ella, pero solamente mediante texto. La aplicación dispone de un apartado llamado *feed* como el de la Figura 9 donde los usuarios pueden responder preguntas propuestas por la misma comunidad, por la aplicación y comentar en estas respuestas. También se pueden completar desafíos propuestos por la comunidad, además de proponerlos.

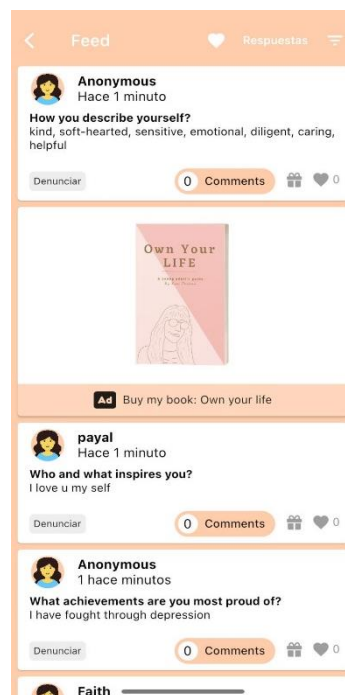


Figura 9: Pantalla aplicación 21 Days

21 Days Challenge también dispone de un apartado dónde puedes ver y modificar tu perfil como en la Figura 10. En este, puedes modificar tu avatar con las monedas que se obtienen completando desafíos, ver los *likes* que se han obtenido en el *feed*, y un recuento de los desafíos actuales y los completados. En este mismo apartado también se dispone de un diario, dónde se pueden hacer anotaciones en los diferentes días, y de una función llamada tareas, dónde se proponen diferentes acciones a hacer dentro de la aplicación para obtener monedas.



Figura 10: Pantalla aplicación 21 Days

Como muchas otras, esta aplicación también dispone de un plan premium de pago el cual se ve en la Figura 11, para quitar los anuncios, permitir descargar diferentes fondos de pantalla para la aplicación y otorga el doble de puntos cuando completas un desafío.

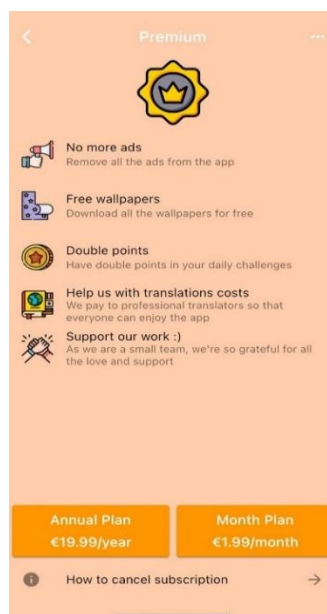


Figura 11: Pantalla aplicación 21 Days

## 2.2.4 Tabla comparativa

Analizadas las diferentes aplicaciones que se han considerado las principales competidoras, como se ilustra en la Tabla 1, se ha elaborado una tabla comparativa que refleja las diferentes características que ofrece cada aplicación. Esto se muestra en la tabla a continuación, que nos ayuda a identificar qué características deberían de ser implementadas para no quedarse atrás de la competencia, y cuales nos proporcionan una ventaja frente a ellas.

Propiedades	<i>Rabit</i>	<i>21 Days</i>	<i>Challenge Accepted</i>	UDARE
Recompensa por completar retos	✓	✓	✗	✓
Crear retos propios	✓	✓	✓	✓
Proponer retos para la comunidad	✗	✓	✗	✓
<i>Feed</i> con fotos	✗	✗	✗	✓
<i>Feed</i> con texto	✗	✓	✗	✓
Comentarios en las publicaciones del <i>feed</i>	✗	✓	✗	✓
Reacciones a las publicaciones del <i>feed</i>	✗	✗	✗	✓
Perfil del usuario	✓	✓	✓	✓
Lista de amigos	✗	✓	✓	✓
Mejores amigos	✗	✗	✗	✓
Ranking entre amigos	✗	✗	✓	✓
Subir fotos completando los retos	✗	✗	✗	✓
Hacer foto con ambas cámaras	✗	✗	✗	✓
Álbum de fotos de los retos	✗	✗	✗	✓

Calendario con los días que se han completado retos	✓	✓	✓	✓
Categorías de retos	✓	✓	✓	✓
Contador de puntos por categoría	✗	✗	✗	✓
Buscador de retos	✓	✓	✓	✓
Lista de retos en curso	✓	✓	✓	✓
Lista de retos acabados	✓	✓	✓	✓
Recordatorios de retos	✓	✓	✓	✓
Publicidad	✓	✓	✗	✓
Función premium	✓	✓	✗	✓
Retos patrocinados	✗	✗	✗	✓
Interfaz en diferentes idiomas	✗	✓	✗	✓
Ubicación de dónde se ha realizado el reto	✗	✗	✗	✓
Conexión con Spotify para compartir música	✗	✗	✗	✓

Tabla 1: Comparativa competidores

A modo de conclusión, hemos podido observar que existen muchas aplicaciones relacionadas con el seguimiento de hábitos que te intentan incentivar a llevar una vida más saludable, pero ninguna de estas aplicaciones tiene el enfoque de red social que nosotros ofrecemos en nuestra aplicación, y que es en esencia lo que nos hace únicos y nos diferencia de la competencia. Es decir, en nuestra aplicación, aparte de todo lo comentado anteriormente, tendrás ese componente social de las redes sociales que te permitirá interactuar con otras personas compartiendo, comentando o dando «*me gusta*» a los retos que suban tus amigos. Básicamente, podrás compartir tu estilo de vida y tu progreso con tus amigos y recibir *feedback* de estos, lo que permitirá impulsar la competitividad y motivación mutua.

Gracias a la realización de este análisis y la identificación de los puntos débiles de nuestra competencia, hemos podido encontrar una gran oportunidad de negocio, que,



en definitiva, se resume en la combinación del mundo del desarrollo personal y hábitos con el mundo de las redes sociales.

## 2.3 Proyección de ingresos y gastos

Se ha realizado una proyección de ingresos y gastos semestral de los primeros 5 años comparando los ingresos y costos que se asumirán durante la realización y mantenimiento del proyecto. Durante este periodo se tendrán tres tipos de ingresos, las contrataciones directas por empresas que quieran publicitarse en nuestra aplicación, ya sea patrocinando retos o por publicidad directa, y los ingresos provenientes de las suscripciones premium.

Los contratos con los patrocinadores de retos pueden ser diferentes, dependiendo del tipo de publicidad que requieran y de las visitas que obtengan, por lo que se ha estimado un ingreso medio de 3.750€ semestralmente por patrocinador los primeros dos años, y de 10.000€ a partir del tercero, ya que habrá un aumento de usuarios, y por consecuencia, de visitas.

En cuanto al pago por publicidad entre publicaciones, los ingresos dependen de qué tipo de anuncios se va a publicitar, por lo que se ha estudiado cuáles interesan más a los potenciales usuarios y cuáles son los ingresos de cada categoría. Teniendo esto en cuenta se ha buscado un punto medio para obtener beneficios, y no arruinar la experiencia del usuario con anuncios que no sean de su interés. Esto se ha calculado haciendo la media del ingreso por visita en Google Ads<sup>16</sup> de estas 4 categorías en la región de Europa, que será dónde se tendrán la mayoría de los usuarios, sobre todo en los inicios de la aplicación. [7]

Los gastos base para tener en cuenta para que la aplicación funcione son los de la base de datos, que se utilizará MongoDB<sup>17</sup>, toda la gestión de la autenticación de usuarios que se realizará mediante Firebase<sup>18</sup>, el almacenamiento de las imágenes en AWS S3<sup>19</sup>, el hosting del *back-end* con AWS EC2<sup>20</sup> y la herramienta AWS API GATEWAY<sup>21</sup> que se utilizará para gestionar las llamadas al *back-end*. El cálculo de esto es muy poco preciso ya que dependerá mucho de factores como la cantidad de usuarios, lo activos que sean estos o la cantidad de llamadas que se haga al api del *back-end*. Por esta razón se ha hecho una estimación al alza, para poder ofrecer el servicio al usuario sin ningún problema.

Un aspecto muy importante hoy en día es el *marketing*, por lo que desde el primer año se va a invertir en ello contratando una empresa externa para una campaña de

---

<sup>16</sup> Google Ads: <https://adsense.google.com/intl/es-419/start/>

<sup>17</sup> MongoDB: <https://www.mongodb.com/es/pricing>

<sup>18</sup> Firebase: <https://firebase.google.com/pricing?hl=es&authuser=0#blaze-calculator>

<sup>19</sup> Amazon Web Services S3: <https://calculator.aws/#/createCalculator/S3>

<sup>20</sup> Amazon Web Services EC2: <https://calculator.aws/#/createCalculator/ec2-enhancement>

<sup>21</sup> Amazon Web Services API GATEWAY: <https://calculator.aws/#/createCalculator/APIGateway>

contenidos, dando así a conocer el producto. A partir del segundo año se optará por contratar a un director de marketing que se encargará de gestionar nuestras redes sociales y de idear las campañas de publicidad. En el cuarto año se contratará a una segunda persona que se unirá al CMO para formar un equipo de marketing.

Respecto a los desarrolladores, inicialmente seremos dos, pero se ha decidido que en el segundo año se contratará a un desarrollador senior con experiencia en el desarrollo de aplicaciones móviles.

No será hasta el tercer año cuando hará falta una oficina para trabajar, ya que aparte de los dos desarrolladores iniciales, el director de marketing y el desarrollador senior se contratará un desarrollador junior en el tercer año, y otro en el cuarto, para ayudar en el mantenimiento y soporte de la aplicación, y a un director financiero, por lo que se alquilará una oficina y se invertirá en material informático como portátiles, pantallas, etc.

Finalmente, pero no menos importante, se tienen los gastos de administración, donde se incluyen los pagos a una gestoría durante los dos primeros años y a partir del tercer año se contratará un director Financiero que se encargará de las mismas funciones. En el quinto año debido al volumen de ingresos se contará también con un administrativo para ayudar al CFO.

Todo esto se encuentra detallado en la Tabla 2, donde se estiman los gastos de cada servicio y los ingresos que se obtendrán durante los primeros cinco años. Como se puede ver, la inversión inicial deberá alcanzar los 55129€, ya que es el máximo negativo en el resultado semestral acumulado. El punto de inflexión se refiere al momento en el cual una tendencia en los datos cambia de dirección, por lo que se sitúa en el primer semestre del tercer año. Podremos decir que hemos recuperado toda la inversión en los inicios del segundo semestre del cuarto año, donde el resultado semestral acumulativo superará la inversión inicial. Para obtener los precios definidos en la Tabla 2, hemos accedido a las páginas donde muestran los precios de MongoDB [8], AWS EC2 [9], AWS S3 [10], Firebase [11] y AWS ApiGateway [12].

También se ha realizado un gráfico, que se puede ver en la Figura 12, donde se resumen los ingresos, gastos, y el coste acumulado durante este tiempo. En este gráfico se puede observar que no será hasta el segundo semestre del tercer año donde el balance total será positivo, pese a que en el primer semestre de ese año ya se obtendrá beneficio.

# UDARE: Desarrollo front-end de una red social para promover hábitos saludables

Fuentes de Ingresos	Años y semestres									
	1		2		3		4		5	
	1	2	1	2	1	2	1	2	1	2
Pratrocinadores Nuevos	2	3	5	5	7	9	8	10	11	12
Usuarios Activos	25000	60000	150000	215000	300000	410000	500000	700000	1000000	1500000
Numero de visitas en publicidad	50000	100000	350000	500000	800000	1000000	1500000	2100000	3000000	4000000
Suscripciones premium	300	400	550	700	900	1100	1350	1500	1800	2300
<b>Ingresos Anuales</b>										
<b>Total Ingresos</b>	12.669 €	20.790 €	48.749 €	61.263 €	137.143 €	173.829 €	204.547 €	272.809 €	355.502 €	446.937 €
<b>Gastos Anuales</b>										
Base de datos mongodb	892 €	2.410 €	2.410 €	4.642 €	4.642 €	8.928 €	8.928 €	13.280 €	17.632 €	17.632 €
Gestión de autenticación firebase	0 €	330 €	2.760 €	4.692 €	6.900 €	9.900 €	14.250 €	17.940 €	19.500 €	21.750 €
AWS S3	1.044 €	1.326 €	1.752 €	2.316 €	2.742 €	3.588 €	4.296 €	5.424 €	6.696 €	8.754 €
AWS EC2	390 €	1.056 €	2.538 €	3.876 €	4.548 €	5.196 €	6.030 €	7.368 €	8.706 €	10.050 €
AWS API GATEWAY	1.050 €	1.890 €	2.100 €	2.625 €	3.675 €	5.250 €	6.300 €	7.875 €	9.450 €	10.500 €
Marketing	15.000 €	15.000 €	15.000 €	15.000 €	17.000 €	17.000 €	22.000 €	22.000 €	30.000 €	30.000 €
CMO - Director de Marketing	0 €	0 €	15.000 €	15.000 €	15.000 €	15.000 €	15.000 €	15.000 €	15.000 €	15.000 €
Empleado marketing	0 €	0 €	0 €	0 €	0 €	0 €	9.000 €	9.000 €	9.000 €	9.000 €
Material informático	0 €	0 €	0 €	0 €	5.000 €	5.000 €	5.000 €	5.000 €	5.000 €	5.000 €
Desarrollador senior	0 €	0 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €
Desarrollador junior	0 €	0 €	0 €	0 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €
Desarrollador junior	0 €	0 €	0 €	0 €	0 €	0 €	12.500 €	12.500 €	12.500 €	12.500 €
Oficina	0 €	0 €	0 €	0 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €
Gestoria	6.000 €	7.000 €	7.500 €	8.000 €	0 €	0 €	0 €	0 €	0 €	0 €
CFO - Director Financiero	0 €	0 €	0 €	0 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €
Administrativo	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	7.500 €	7.500 €
<b>Total Gastos</b>	24.376 €	29.012 €	69.060 €	76.151 €	122.007 €	132.362 €	165.804 €	177.887 €	203.484 €	210.186 €
<b>Resultado Semestral</b>	-11.707 €	-8.222 €	-20.312 €	-14.888 €	15.136 €	41.467 €	38.743 €	94.922 €	152.018 €	236.751 €
<b>Resultado Semestral Acumulado</b>	-11.707 €	-19.929 €	-40.241 €	-55.129 €	-39.993 €	1.475 €	40.217 €	135.139 €	287.157 €	523.908 €

Tabla 2: Planificación ingresos y gastos

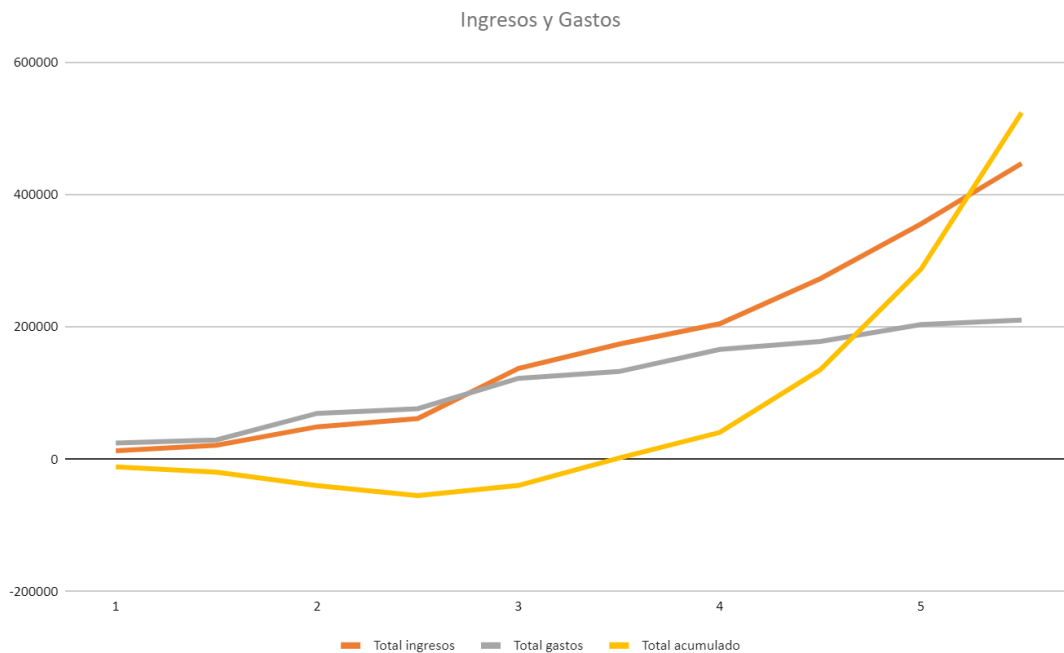


Figura 12: Gráfico de ingresos, gastos y total acumulado

## 2.4 Análisis DAFO

El análisis DAFO, también conocido como análisis *SWOT* en inglés (Fortalezas, Oportunidades, Debilidades y Amenazas), es una herramienta fundamental en el ámbito del emprendimiento. Esta herramienta permite evaluar tanto los factores internos como externos que pueden afectar a un proyecto [5].

Identificar las fortalezas y debilidades internas proporciona una comprensión interna de la situación, mientras que examinar las oportunidades y amenazas externas ofrece una visión del entorno competitivo. Al combinar estos elementos, el análisis DAFO ayuda a los emprendedores a tomar mejores decisiones, diseñar estrategias efectivas, minimizar riesgos y maximizar las oportunidades.

En la Tabla 3 tenemos el análisis DAFO de UDARE, dónde exponemos las fortalezas, debilidades, oportunidades y amenazas del proyecto.



Tabla 3: Análisis DAFO<sup>22</sup>

En cuanto a las debilidades internas, el proyecto se enfrenta a numerosos desafíos, destacando la falta de experiencia en el desarrollo de aplicaciones móviles y la falta de experiencia en la gestión empresarial de un proyecto de emprendimiento de esta envergadura. Además, se ve limitado por un tiempo reducido, los altos costos asociados

<sup>22</sup> Tabla 3: Análisis DAFO. Esta imagen se ha diseñado utilizando la herramienta en línea Canva, disponible en: <https://www.canva.com/>

con la escalabilidad y el mantenimiento de la red social y la falta de financiación. Todos estos inconvenientes podrían obstaculizar la viabilidad del proyecto a largo plazo.

En cuanto a las amenazas externas, el proyecto se enfrenta a un mercado de redes sociales altamente competitivo, dónde se encuentran grandes empresas como Facebook y TikTok que dominan actualmente el mercado.

La rápida evolución tecnológica y la necesidad constante de actualización son amenazas adicionales. Las cambiantes preferencias del usuario también representan un riesgo, ya que no se pueden prever y podría afectar a la adopción de la aplicación.

A pesar de las debilidades, el proyecto cuenta con fortalezas, incluyendo bajos costos iniciales, un equipo motivado dispuesto a aprender y que tiene una gran familiaridad con el sector de las redes sociales. En términos de oportunidades y factores externos, el mercado tecnológico y de las redes sociales está en crecimiento, además, en los últimos tiempos hay una clara tendencia general por parte de la población hacia los estilos de vida más saludables y sostenibles, impulsados principalmente por los ODS<sup>23</sup>. Nuestra propuesta innovadora, que está perfectamente alineada con estas nuevas tendencias, se posiciona como una solución atractiva y que podría hacerse un hueco en este mercado.

## 2.5 Lean Canvas

Hemos usado una adaptación del *Lean Canvas* basada en el *Canvas* de Modelo de Negocio diseñado por Alex Osterwalder que gracias a Ash Maurya se puede utilizar como una adaptación específica para emprendedores que están trabajando en su idea de negocio, sobre todo en sus primeras fases [6]. Es una herramienta para esbozar y analizar el modelo de negocio de una *startup* dónde el lienzo del modelo de negocio se divide en nueve bloques: clientes, problema, proposición de valor, solución, canales, ingresos, costos, métricas y ventaja competitiva, que se ha rellenado en ese mismo orden, ya que ese es el más ideal para conceptualizar la idea correctamente.

En la Tabla 4, se muestra la versión actual del documento que se obtiene mediante este proceso, aunque podrá cambiar en el futuro a medida que se desarrolle el proyecto. Cabe destacar de entre los diferentes bloques la proposición de valor y la ventaja competitiva, pues tienen un impacto significativo en la percepción e interés de nuevos inversores, debido a que una proposición clara y convincente, y una ventaja diferencial frente a otros competidores puede captar fácilmente la atención de los inversores, y muestra que se entienden las necesidades de los clientes.

---

<sup>23</sup> ODS: Conjunto de 17 metas globales establecidas por la Asamblea General de las Naciones Unidas en 2015, como parte de la Agenda 2030.

<p><b>2 Problema</b></p> <p>Aunque existen muchas aplicaciones enfocadas a la consecución de retos, ninguna de ellas ofrece a la vez funcionalidades sociales, un enfoque en retos saludables, y facilidades para que el usuario pueda encontrar y escoger retos a su medida.</p> <p>Este tipo de aplicaciones tampoco están enfocadas a ser redes sociales, sino que son más simples, y no consiguen que los usuarios interactúen entre ellos.</p>	<p><b>4 Solución</b></p> <p>Nuestra propuesta es ofrecer diferentes retos diarios para que los usuarios los completen y los compartan con sus seguidores, y hacer una aplicación cómoda y fácil de usar.</p>	<p><b>3 Proposición de valor</b></p> <p>Experimenta la diferencia con nuestra red social única. Cada día, te desafiamos con retos de diversas categorías. Supéralos, comparte tus logros con amigos y acumula puntos para desbloquear descuentos exclusivos. ¡Únete a nosotros y haz que tu vida sea más activa!</p>	<p><b>9 Ventaja competitiva</b></p> <p>Nuestra aplicación es única en su enfoque social, ya que ninguna rival del mercado se centra en compartir los retos con tus amigos, solamente se centran en la realización de los mismos.</p>	<p><b>1 Clientes</b></p> <p>El producto está enfocado a usuarios que tengan o busquen tener nuevos hábitos y busquen emplear una red social para compartirlas con otros usuarios con el mismo objetivo; del mismo modo, también la usarán múltiples empresas, que ofrecerán a los usuarios beneficios y descuentos.</p> <p>Los primeros usuarios serán los que busquen una aplicación de estas características, así como aquellas empresas que muestren interés inicial en relacionarse con nosotros.</p>
<p><b>7 Costos</b></p> <p>Los principales costes de la aplicación surgirán de sus empleados (un total de 5 programadores), el mantenimiento de sus servicios en línea (especialmente la base de datos) y el marketing.</p> <p>Nuestras estimaciones para el quinto año de vida de la aplicación indican unos costes de alrededor de 407800€.</p>	<p><b>6 Ingresos</b></p> <p>Nuestra principal fuente de ingresos será la publicidad de empresas que quieran anunciarse con nosotros a través de los retos patrocinados. Además en un futuro se podría implantar un método para impulsar el crecimiento de las cuentas.</p> <p>Nuestras estimaciones para el segundo año de vida de la aplicación indican unos ingresos de alrededor de 422054€.</p>	<p><b>8 Métricas</b></p> <p>Actualmente, nuestras expectativas de negocio se basan en el interés que han mostrado sus posibles usuarios en entrevistas y encuestas. En el futuro, nos basaremos en las reseñas para garantizar su interés y la sostenibilidad de la aplicación.</p>	<p><b>5 Canales</b></p> <p>La aplicación se descarga a través de las tiendas móviles habituales. Su uso será publicitado en las cadenas que se beneficien o que se relacionen con nosotros, además del marketing mediante redes sociales, como instagram, tiktok o X.</p>	

Tabla 4: Lean Canvas

## 2.6 Conclusiones de la evaluación

Expuesta la idea de negocio, UDARE ofrece una solución innovadora centrada en las necesidades del usuario aprovechando un segmento de mercado prometedor que no está ocupado por ninguna otra empresa.

En primer lugar, se han identificado unos clientes y se han seleccionado unos *Early Adopters* que serán los pilares clave para el éxito. Al dirigirnos a un rango de personas con un elevado nivel de uso de los dispositivos móviles, podemos aprovechar un mercado prometedor. Además, la retroalimentación de los *Early Adopters* será fundamental para perfeccionar la aplicación y garantizar su éxito y uso entre el público objetivo, en este caso los jóvenes entre 16 y 30 años.

Como ya hemos dicho, se ha identificado una oportunidad de mercado ofreciendo un producto innovador respecto a los competidores, lo que se demuestra en este estudio, donde se evidencia que las demás aplicaciones dentro del mismo ámbito carecen del enfoque singular basado en una red social, tal como lo ostenta UDARE. Este enfoque desempeña un papel fundamental en la viabilidad y el potencial de éxito del proyecto.

Tras realizar un análisis económico detallado del proyecto a lo largo de cinco años, dividido en semestres, se considera necesaria una inversión inicial de 55.000€, designada para cubrir una variedad de gastos, incluyendo salarios del personal, servicios en línea para mantener la aplicación operativa, actividades de marketing, así como costos asociados con la administración y gestión empresarial. Se proyecta que la

empresa comenzará a generar beneficios a partir del tercer año, con un balance positivo a partir del segundo semestre de ese mismo año.

En el marco del análisis DAFO se ha realizado una evaluación exhaustiva de las fortalezas, oportunidades, debilidades y amenazas inherentes al proyecto. A pesar de los desafíos asociados a la inexperiencia en el desarrollo de aplicaciones móviles y la gestión empresarial, así como la presencia de competidores destacados y la rápida evolución tecnológica, se han identificado fortalezas como los bajos costos iniciales, una propuesta innovadora y un equipo altamente motivado. Estos elementos posicionan al proyecto de manera favorable, ofreciendo oportunidades significativas en un mercado en expansión.

En definitiva, el proyecto se presenta como una propuesta viable y prometedora en el mercado actual. Al ofrecer una solución innovadora centrada en las necesidades del usuario y dirigirse a un segmento de mercado prometedor, la iniciativa está bien posicionada para alcanzar el éxito. La identificación y selección cuidadosa de *Early Adopters* refuerza aún más su potencial al garantizar la relevancia y utilidad de la aplicación entre el público objetivo. Además, el análisis económico detallado respalda la viabilidad financiera del proyecto, con una inversión inicial calculada para cubrir los gastos operativos y proyecciones de generación de beneficios a partir del tercer año.



## 3. TECNOLOGÍAS UTILIZADAS

---

### 3.1 Lenguaje de programación

En el desarrollo de aplicaciones móviles, es crucial seleccionar la tecnología adecuada que se ajuste a las necesidades específicas del proyecto, el equipo de desarrollo y los objetivos a medio y, sobre todo, a largo plazo. Existen diversas opciones tecnológicas, cada una con sus propias características, ventajas y desventajas. A continuación, se detallan las principales alternativas: desarrollo nativo, desarrollo multiplataforma y desarrollo de Aplicaciones Web Progresivas (PWA).

El desarrollo nativo implica la creación de aplicaciones específicas para cada plataforma móvil utilizando los lenguajes y herramientas proporcionados por los propios fabricantes. Para iOS<sup>24</sup>, esto significa usar Swift<sup>25</sup> u Objective-C<sup>26</sup> junto con Xcode<sup>27</sup> como entorno de desarrollo integrado (IDE). Para Android, se utiliza Kotlin<sup>28</sup> o Java<sup>29</sup> junto con Android Studio<sup>30</sup>.

Entre las principales ventajas del desarrollo nativo encontramos un rendimiento óptimo dado que se ejecutan directamente sobre el sistema operativo del dispositivo, acceso completo a las funcionalidades del dispositivo y una experiencia de usuario superior, debido a que se adaptan de una forma más estricta a las guías de diseño de cada plataforma. No todo son beneficios porque optar por un desarrollo nativo implica desarrollar y mantener dos bases de código separadas y, además, disponer de desarrolladores especializados en ambos lenguajes y plataformas.

El desarrollo multiplataforma busca crear una única base de código que pueda ejecutarse en múltiples sistemas operativos. Herramientas populares para este enfoque incluyen Flutter<sup>31</sup>, React Native<sup>32</sup> y Xamarin<sup>33</sup>. Este tipo de desarrollo aborda las deficiencias del desarrollo nativo, y carece de sus mayores ventajas, por lo que podríamos considerar que es el polo opuesto. Optando por esta metodología, aumentamos la eficiencia de desarrollo y disminuimos la necesidad de diferentes especialistas disponiendo así de un solo código, pero a cambio, tendremos un rendimiento inferior y un acceso limitado a funcionalidades nativas.

Por último, tenemos las PWA, aplicaciones web que se comportan como aplicaciones móviles, accesibles a través del navegador, pero con capacidades como trabajar offline, recibir notificaciones *push* y acceso a algunas características del dispositivo.

Esta opción amplifica las ventajas inherentes al desarrollo multiplataforma, pues ofrece un desarrollo más ágil y económico, garantiza compatibilidad con una amplia gama de plataformas y elimina la necesidad de instalación. No obstante, también exacerba sus

---

<sup>24</sup> iOS: <https://www.apple.com/ios/ios-17/>

<sup>25</sup> Swift: <https://www.apple.com/co/swift/>

<sup>26</sup> Objective-C: <https://apps.apple.com/us/app/obj-c-programming-language/id486204866>

<sup>27</sup> Xcode: <https://developer.apple.com/la/support/xcode/>

<sup>28</sup> Kotlin: <https://kotlinlang.org/>

<sup>29</sup> Java: <https://www.java.com/es/>

<sup>30</sup> Android Studio: <https://developer.android.com/studio?hl=es-419>

<sup>31</sup> Flutter: <https://flutter.dev/>

<sup>32</sup> React Native: <https://reactnative.dev/>

<sup>33</sup> Xamarin: <https://dotnet.microsoft.com/es-es/apps/xamarin>



limitaciones, restringiendo aún más la funcionalidad y deteriorando el rendimiento de la aplicación.

Tras un análisis exhaustivo de las distintas alternativas, se ha decidido desarrollar la aplicación utilizando el lenguaje nativo de Android, Kotlin. Este lenguaje ofrece varias ventajas significativas en comparación con otras opciones. En primer lugar, permite alcanzar un rendimiento óptimo y proporciona acceso completo a todas las funcionalidades del dispositivo, factores cruciales para garantizar una aplicación de alta calidad. Además, la sintaxis concisa y segura de Kotlin disminuye la probabilidad de errores y facilita el mantenimiento del código. Kotlin está plenamente integrado con Android Studio, lo que proporciona un entorno de desarrollo robusto y eficiente, accesible en cualquier dispositivo, a diferencia de Xcode. Aparte de los motivos técnicos, al inicio del proyecto ya se contaba con cierta experiencia en Kotlin por parte de un integrante del equipo, lo que evitó partir de cero y facilitó la elección de esta metodología, a pesar de los costos y el esfuerzo adicional que puede implicar el desarrollo nativo.

### 3.2 Entorno de desarrollo

Kotlin es un lenguaje de programación moderno que ha ganado popularidad por su interoperabilidad con Java y su sintaxis sencilla y concisa. Existen varias opciones y entornos para programar en Kotlin, cada uno con sus propias características, ventajas y desventajas. A continuación, se presentan las principales alternativas que se tuvieron en cuenta, incluyendo sus mayores virtudes y deficiencias:

#### INTELLIJ IDEA



IntelliJ IDEA <sup>34</sup> es un entorno de programación desarrollada por la misma empresa que creó Kotlin, por lo que es uno de los IDEs más robustos y completos para programar aplicaciones Android. Incluye características avanzadas como la navegación del código, autocompletado inteligente, refactorización segura y herramientas de depuración.

#### ECLIPSE CON KOTLIN PLUGIN



Eclipse <sup>35</sup> es uno de los IDEs más populares para el desarrollo Java, y que nos permite instalar el plugin de Kotlin para programar una aplicación Android. Esta opción, aunque ya lo conocíamos por haberlo utilizado durante la carrera, fue rechazada en todo momento debido a la mala experiencia que tuvimos en su uso.

#### VS CODE CON KOTLIN EXTENSION



Visual Studio Code <sup>36</sup> es un editor de código ligero y muy bien optimizado que funciona a la perfección en la mayoría de las máquinas con pocos recursos hardware, pero, aunque es muy buena opción para la programación en cualquier lenguaje, no proporciona las funcionalidades necesarias para desarrollar un proyecto de esta magnitud, por lo que también se rechazó su utilización, pese a la buena experiencia de todo el equipo con este IDE.

<sup>34</sup> IntelliJ IDEA: <https://www.jetbrains.com/idea/>

<sup>35</sup> Eclipse: <https://eclipseide.org/>

<sup>36</sup> Visual Studio Code: <https://code.visualstudio.com/>

## ANDROID STUDIO

Fruto de la colaboración con JetBrains<sup>37</sup> y Google es el IDE oficial para el desarrollo de Android. Basado en IntelliJ IDEA es el entorno recomendado para el desarrollo en Kotlin debido a su profunda integración con el SDK (*Software Development Kit*<sup>38</sup>) de Android. Proporciona herramientas específicas como el emulador de Android, herramientas de perfilado y un diseñador de interfaces visual. Además, al ser el IDE oficial, cuenta con soporte y actualizaciones continuas.

Una vez expuestas todas las opciones que se manejaban, se decidió rechazar Eclipse y VS Code por las razones ya expuestas, y quedaron los dos *frameworks* restantes como principales opciones. Finalmente, se decidió utilizar Android Studio por su integración completa con el ecosistema Android, el soporte oficial de Google y las herramientas especializadas, además de toda la documentación que se puede encontrar en la misma plataforma de Android Developers<sup>39</sup>.

## 3.3 GITHUB

GitHub<sup>40</sup> es una plataforma de desarrollo colaborativo que aloja proyectos en la nube utilizando el sistema de control de versiones Git. Desde su lanzamiento en 2008, se ha convertido en una herramienta indispensable en el trabajo colaborativo para desarrolladores y equipos de todo el mundo.

En nuestro contexto, al realizar un proyecto colaborativo nos ha sido de gran ayuda para mantener un historial detallado de todos los cambios realizados en el proyecto y compartir el código. También nos ha permitido mantener una gran consistencia del código para todos, y a través de las ramas hemos podido trabajar de forma aislada en diferentes funcionalidades sin tener ningún tipo de problema.

Para tener una mayor consistencia en la versión original, se decidió utilizar la metodología *GitFlow*, donde se definen una serie de ramas y flujos de trabajo que facilitan la gestión de versiones, la integración de nuevas funcionalidades y la resolución de errores. En esta metodología, se define una rama *master*, donde solo se añade código sin fallos, una rama *develop*, donde los diferentes desarrolladores hacen los *pull requests*, y donde se añaden los cambios para ser probados y una serie de ramas temporales, que añaden diferentes funcionalidades y así facilitar el flujo de trabajo. En la Figura 13 podemos ver el repositorio de *front-end* de UDARE, donde registramos y subimos todos los cambios del código.

---

<sup>37</sup> JetBrains: <https://www.jetbrains.com/es-es/>

<sup>38</sup> SDK (*Software Development Kit*): Conjunto de herramientas, bibliotecas y documentación que permite a los desarrolladores crear aplicaciones para una plataforma específica

<sup>39</sup> Android Developers: <https://developer.android.com/?hl=es-419>

<sup>40</sup> Github: <https://github.com/>

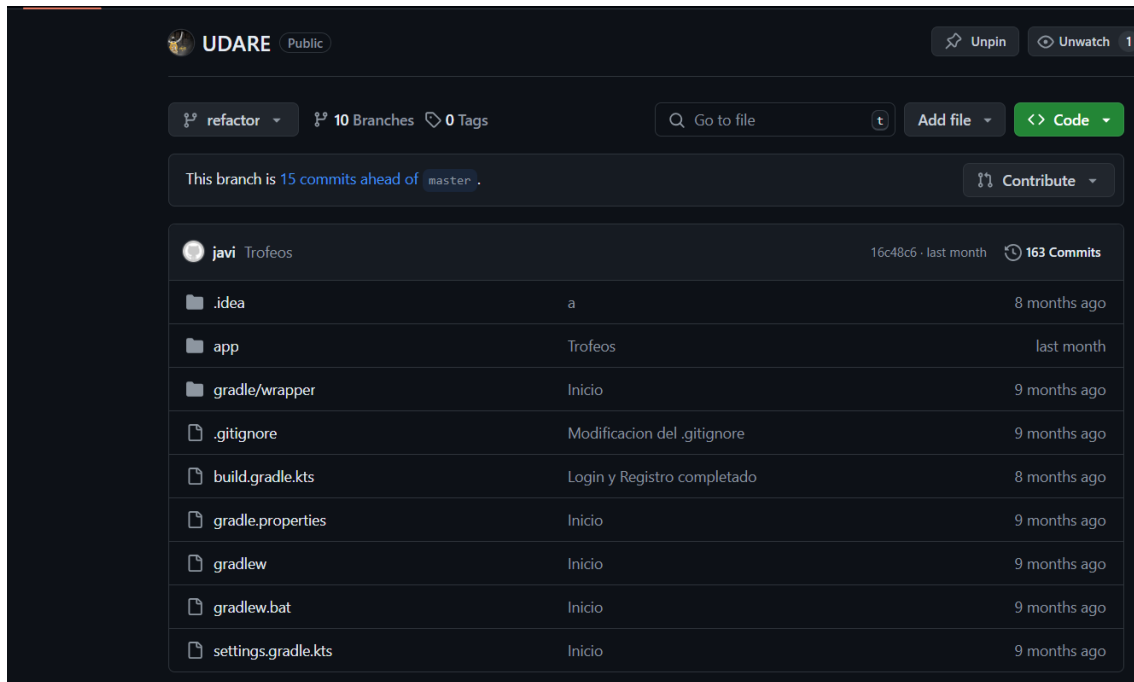


Figura 13: Repositorio del front-end

### 3.4 FIGMA

Figma <sup>41</sup> es una herramienta de diseño basada en la web que permite a los diseñadores crear, compartir y colaborar en proyectos de diseño en tiempo real. Esta herramienta destaca por su enfoque en la colaboración y la accesibilidad que, a diferencia de otras que requieren instalaciones locales y configuraciones específicas, Figma opera completamente en el navegador facilitando el acceso desde cualquier dispositivo con conexión a internet.

Una de sus virtudes es que permite la creación de diseños de alta fidelidad con una variedad de herramientas vectoriales y de edición de imágenes, asegurando que cada elemento esté alineado con las directrices de diseño de la plataforma. Otra de las principales ventajas de Figma es su capacidad para permitir la colaboración en tiempo real. Múltiples diseñadores y desarrolladores pueden trabajar simultáneamente en el mismo archivo, hacer comentarios y realizar cambios instantáneamente. En la Figura 14 se puede ver una vista global del proyecto Figma realizado para las interfaces de la aplicación.

<sup>41</sup> Figma: <https://www.figma.com/>

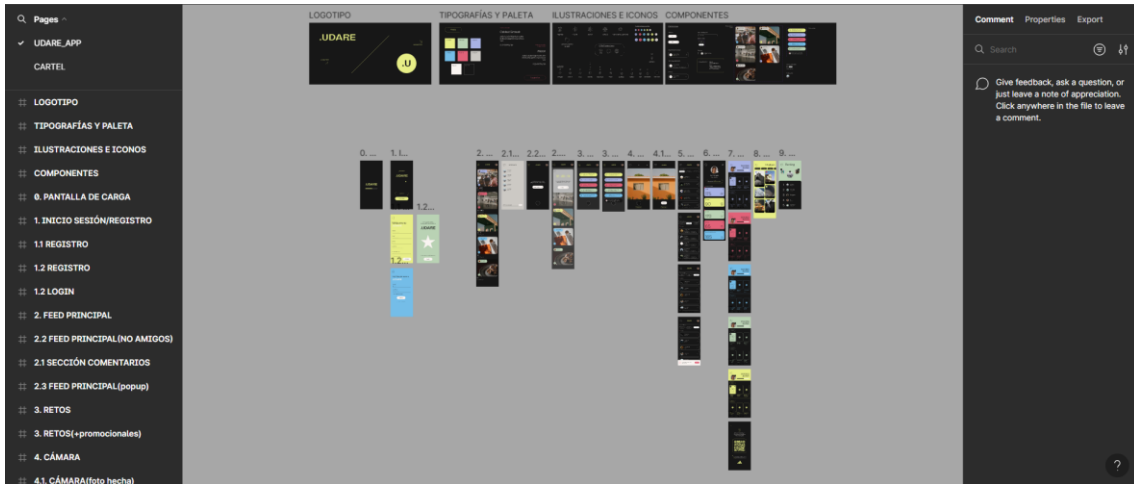


Figura 14: Imagen global del proyecto figma

## 4. DESARROLLO DE LA SOLUCIÓN

---

### 4.1 Metodología

Para el desarrollo de la solución se va a optar por seguir la metodología *Lean Startup* [12]. La idea de este método es ir adaptando el producto a la demanda del mercado y no a nuestra propia visión. Con esto se permite la creación de productos y servicios de una manera ágil e innovadora.

Su funcionamiento es simple, el primer paso consiste en el desarrollo de una versión básica del producto llamado MVP (*Minimum Viable Product*). Cada vez que se complete el desarrollo de un MVP, se realizarán experimentos con grupos reducidos de personas de nuestro público objetivo, que serán los *Early Adopters*. De esta forma se podrá recibir retroalimentación temprana de los usuarios, lo que puede permitir, por una parte, confirmar nuestras creencias acerca de lo que puede gustar al público, o, por otra parte, darse cuenta a tiempo de que la idea de negocio no acaba de encajar en el mercado, y que, por lo tanto, se tendrá que pivotar, cambiar las estrategias y, en definitiva, enfocar la idea de negocio de otra forma.

Todo esto es un proceso iterativo, es decir, se lanzarán tantos como sean necesarios para validar nuestras hipótesis y asegurarnos de que el producto va a satisfacer las necesidades reales de los usuarios. En cada iteración se irá mejorando y puliendo el producto de acuerdo con el *feedback* que nos va proporcionando el usuario, para que de esta manera se pueda desarrollar un producto final que realmente cumpla con las expectativas y satisfaga al público objetivo.

Cada MVP se desarrollará haciendo uso de algunas de las prácticas de los métodos ágiles más populares actualmente, como son Scrum<sup>42</sup> o Kanban<sup>43</sup> [8]. Concretamente, se incluirá en el proyecto la elaboración de un *backlog*, donde se tendrán ordenadas por prioridad las tareas pendientes de realizar en el proyecto, para la planificación del trabajo a realizar se utilizarán *Sprints*, que son iteraciones de trabajo asociado a un período de tiempo, también se trabajará con unidades de trabajo (UT), que se utilizarán como medida para definir tareas específicas y que podrán ser de diferentes tipos y tener distintos *workflows*. Por último, para llevar una gestión eficiente del trabajo hecho y por hacer, se utilizará la herramienta del tablero kanban.

En nuestro caso concreto, el primer MVP ya se desarrolló en la asignatura de PIN, lo que significa que, como ya se ha explicado anteriormente, pero queremos volverlo a remarcar, este TFG abarcaría el desarrollo de los experimentos sobre ese primer MVP, el desarrollo completo del segundo MVP y los experimentos del segundo MVP.

---

<sup>42</sup> *Scrum*: La metodología ágil Scrum es un marco de trabajo que facilita la colaboración y gestión de proyectos en equipos, promoviendo iteraciones cortas y entregas incrementales de producto [20].

<sup>43</sup> *Kanban*: La metodología ágil Kanban es un sistema visual de gestión de flujo de trabajo que optimiza la eficiencia mediante la visualización de tareas en columnas que representan diferentes estados de progreso [20].

La organización de un proyecto es crucial para su éxito. Antes de ponernos a programar debemos de tener una comprensión clara de las características que vamos a implementar, así como los plazos de entrega y el tiempo que nos va a llevar su desarrollo.

Para ello, hemos optado por utilizar *Trello*<sup>44</sup> para la organización general de nuestro proyecto. *Trello* es una herramienta en línea de gestión de proyectos que utiliza tableros, listas y tarjetas para organizar tareas y colaborar en equipo. En concreto haremos uso de un tablero kanban con las siguientes columnas:

- **Backlog:** Se trata de un contenedor donde depositaremos todas las unidades de trabajo no acabadas. Inicialmente contendrá todas nuestras UT.
- **Especificación de requisitos:** Consistirá básicamente en realizar una descripción detallada de cada UT, en especificar todas sus pruebas de aceptación y en realizar una breve estimación.
- **Programar:** A esta columna se moverán las UT de las que se haya previamente especificado los requisitos y que estén listas para su desarrollo e implementación.
- **Aplicar pruebas de aceptación:** Una vez se haya acabado el desarrollo de una UT, se le deberán pasar las pruebas de aceptación asociadas a esta. En el caso de que se encuentren fallos, se retornará a la columna de programar.
- **Terminado:** En esta columna se irán depositando todas las UT finalizadas, es decir, todas aquellas que hayan pasado todas sus pruebas de aceptación.

En la siguiente Figura 15 se muestran todas las características en el backlog, así como las columnas de nuestro tablero por las que deberán pasar todas ellas. A medida que vayamos avanzando en el desarrollo, se realizarán instantáneas del tablero para poder mostrar el progreso a lo largo de todo el desarrollo. Como se puede observar, cada UT tiene una etiqueta de un color asociada, que corresponde a la característica general a la que hace referencia.

---

<sup>44</sup> Trello: <https://trello.com/>

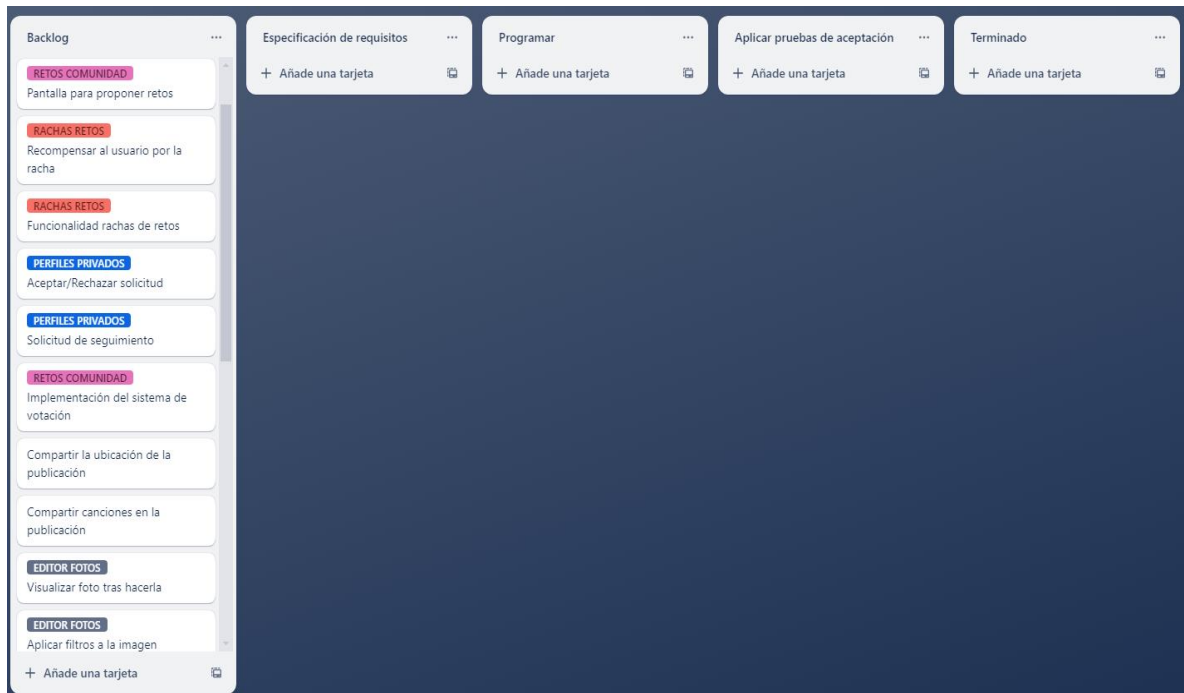


Figura 15: Tablero Kanban Trello primer sprint

## 4.2 Requisitos

Los requisitos funcionales son especificaciones detalladas de las funciones que un sistema debe cumplir. Definen las acciones que el sistema debe ser capaz de realizar, incluyendo las interacciones entre los usuarios y el sistema, así como las funcionalidades específicas que el sistema debe proporcionar. Antes de programar se debe tener una comprensión clara acerca de los requisitos de la aplicación. En este apartado se expondrán los requisitos funcionales generales.

- **Guardar sesión del usuario:** Esta función tiene como objetivo permitir a los usuarios poder iniciar sesión automáticamente al entrar en la aplicación sin la necesidad de tener que poner el correo y la contraseña cada vez. Para ello, se tendrá que guardar la sesión del usuario internamente, y permitir al usuario poder cerrar la sesión manualmente.

*UT Asociadas:* Cerrar sesión del usuario, Guardar la sesión del usuario.

*Pruebas de aceptación:*

- Iniciar sesión con un usuario, cerrar la aplicación, volver a abrirla y comprobar que se inicia la aplicación con el mismo usuario.
- Al pulsar el botón de cerrar sesión, cerrar la aplicación y volver a abrirla te pide los datos de autenticación.

- Cerrar sesión, iniciar sesión con otro usuario y comprobar que se inicia sesión con el segundo usuario, y se guarda correctamente tras cerrar y volver a abrir la aplicación.

*Estimación en HIP<sup>45</sup>: 20h*

- **Ranking:** Consta de una tabla de clasificación dónde el usuario puede ver que usuarios tienen mayor puntuación, clasificados de mayor a menor, tanto entre sus amigos, como con el resto de los usuarios de la aplicación.

*UT Asociadas:* Ranking de amigos, Ranking global.

*Pruebas de aceptación:*

- El ranking de amigos deberá mostrar a todos los amigos del usuario actual ordenados por puntuación.
- La tarjeta del usuario actual está resaltada con un color llamativo.
- El usuario deberá poder visualizar rápidamente su posición en el ranking.
- El ranking global mostrará una clasificación con el top 50 usuarios con mayor puntuación a nivel mundial.
- El ranking se actualizará en tiempo real cada vez que algún integrante obtenga nuevos puntos.

*Estimación en HIP: 25h*

- **Visitar perfiles ajenos:** Desde el buscador de usuarios, seguidores o seguidos, desde el ranking o desde las publicaciones, se debe de poder visitar el perfil de un usuario clicando en su tarjeta, es decir, en su foto de perfil o nombre.

*UT Asociadas:* Visitar perfil ajeno.

*Pruebas de aceptación:*

- Desde el buscador de usuarios se clica en la tarjeta de un usuario y se muestra el perfil de dicho usuario.
- Desde el buscador de seguidores se clica en la tarjeta de un usuario y se muestra el perfil de dicho usuario.
- Desde el buscador de seguidos se clica en la tarjeta de un usuario y se muestra el perfil de dicho usuario.
- Desde el ranking se clica en la tarjeta de un usuario y se muestra el perfil de dicho usuario.
- Desde el feed se clica en la foto o nombre del usuario y se muestra su perfil.

---

<sup>45</sup> *HIP (Horas Ideales de Programación):* Horas dedicadas a la tarea de programación, suponiendo que se realizan de forma ininterrumpida.





- Se muestra el perfil del usuario con todos sus datos e información correcta y actualizada.

*Estimación en HIP: 15h*

- **Retos de la comunidad:** A parte de los 5 retos pertenecientes a cada categoría que todos los usuarios tendrán, el usuario tendrá la opción de completar retos creados por la comunidad. Los usuarios podrán proponer retos, y los más votados por la comunidad (tanto desarrolladores como usuarios), serán publicados para su realización. Esta característica incluye varias subtarear: realización de una pantalla donde los usuarios puedan votar, realización de una pantalla donde el usuario pueda proponer retos, implementación de votación.

*UT Asociadas:* Pantalla de votación para los usuarios, Pantalla para proponer retos, Implementación del sistema de votación.

*Pruebas de aceptación:*

- El usuario debe tener la opción de realizar el reto de la comunidad.
- El usuario puede participar activamente en la decisión de qué retos de la comunidad serán publicados, para ello deberá tener la opción de votar los que más le gusten.
- El usuario deberá poder proponer retos a la comunidad y esperar a que estos sean votados.

*Estimación en HIP: 35h*

- **Notificaciones** El usuario recibirá diferentes tipos de notificaciones en función de los diferentes eventos que se activen en nuestra aplicación. Para cuando un amigo al que le has activado las notificaciones desde su perfil sube una publicación, debe llegarle al usuario una notificación de que ese usuario ha completado un nuevo reto. Cuando otro usuario le da al botón de seguir en tu perfil, o en el buscador de amigos, al usuario le llega una notificación de que tiene un seguidor nuevo (en el caso de que tenga el perfil público). En el caso de que un usuario tenga el perfil privado, recibirá notificaciones de solicitudes de seguimiento. Por último, cuando un amigo de un usuario lleve una racha considerable (por ejemplo 100 días seguidos completando un reto), le saldrá una notificación con dicha información.

*UT Asociadas:* Notificación de seguidor nuevo, notificación de publicación de amigo, notificación de solicitud de seguimiento y notificación racha de amigo.

*Pruebas de aceptación:*

- Activar las notificaciones de un amigo, que ese amigo suba una foto y comprobar que llega la notificación.
- Desactivar las notificaciones de un amigo, que ese amigo suba una foto y comprobar que no llega la notificación.

- Seguir a un usuario con perfil público y comprobar que le llega la notificación de nuevo seguidor.
- Enviar una solicitud de seguimiento a un perfil privado y comprobar que llega la notificación.
- Que un usuario complete una racha de 100 retos y comprobar que a sus amigos les llega una notificación.

*Estimación en HIP: 35h*

- **Rachas de retos:** El usuario dispondrá de una racha de retos que se irá incrementando a medida que vaya completando los retos. Inicialmente empezará en 0 y cuando el usuario consiga rachas significativas como racha de 20, 50, 100 o 200 días consecutivos completando el reto diario, se le recompensará con premios.

*UT Asociadas:* Recompensar al usuario por la racha, Funcionalidad rachas de retos.

*Pruebas de aceptación:*

- Inicialmente la racha será de 0.
- Si el usuario completa el reto diario, se incrementará en 1 la racha.
- Si el usuario no completa el reto del día, la racha se retornará al valor 0.
- Cuando el usuario llegué a un número de racha especificado, se le recompensará.
- La racha del usuario se podrá visualizar en su perfil.
- El usuario podrá visualizar la racha de sus amigos.

*Estimación en HIP: 25h*

- **Editor de fotos:** Tras hacer una foto, el usuario deberá poder editarla antes de subirla. El usuario podrá aplicar distintos filtros e incluso descartarla en caso de que no le guste.

*UT Asociadas:* Previsualizar la imagen, Aplicar filtros a la imagen, Descartar foto.

*Pruebas de aceptación:*

- Hacer una foto, aplicar un filtro, subirla y comprobar que se ha subido correctamente.
- Hacer una foto, eliminarla y comprobar que te devuelve a la cámara para poder repetirla, y que no se ha subido.

*Estimación en HIP: 20h*

- **Perfiles privados:** Permitir una opción al usuario de que su perfil sea privado. Esto implica que únicamente los usuarios que haya aceptado previamente puedan ver el perfil.



*UT asociadas:* Solicitud de seguimiento, aceptar/rechazar solicitud.

*Pruebas de aceptación:*

- Activar la opción del perfil privado en un usuario, cambiar de usuario y comprobar que este último no puede ver el perfil del primer usuario.
- Comprobar que un usuario puede enviar una solicitud a otro usuario que tenga el perfil privado.
- Comprobar que un usuario puede aceptar/rechazar una solicitud de seguimiento de otro usuario.
- Si el usuario acepta la solicitud de seguimiento, el solicitante efectivamente es capaz de ver su perfil
- Si el usuario no acepta la solicitud de seguimiento, el solicitante no podrá visualizar su perfil.

*Estimación en HIP: 20h*

- **Compartir la ubicación de la publicación:**

En el editor de fotos se añade una opción que permite al usuario compartir la ubicación desde dónde está subiendo la foto.

*UT Asociadas:* Compartir la ubicación de la publicación.

*Pruebas de aceptación:*

- Hacer una foto, no marcar la opción de compartir ubicación y comprobar que no se comparte en la publicación.
- Hacer una foto, marcar la opción de compartir ubicación y comprobar que se comparte en la publicación.

*Estimación en HIP: 15h*

- **Compartir canciones en la publicación:** En el editor de fotos se añade una opción que permite al usuario compartir la canción que está escuchando en ese momento en Spotify. En esta UT será necesaria la integración con un servicio externo como es Spotify.

*UT Asociadas:* Compartir canciones en la publicación.

*Pruebas de aceptación:*

- Hacer una foto, no marcar la opción de compartir canción y comprobar que no se comparte en la publicación.
- Hacer una foto, marcar la opción de compartir canción y comprobar que se comparte en la publicación.

*Estimación en HIP: 15h*

- **Recompensas:**

Esta UT trata sobre las recompensas que se le irán entregando al usuario en la aplicación. Estas recompensas se le otorgarán al usuario en forma de trofeo. Estos trofeos estarán visibles en el perfil del usuario, en una ventana especial. Inicialmente el usuario no tendrá ningún trofeo desbloqueado y a medida que vaya desbloqueando, se irán añadiendo a su perfil. Habrá dos tipos de recompensa, recompensas por llegar a un determinado número de puntos y recompensas por llegar a una determinada racha. Para los puntos, inicialmente recompensamos al usuario cuando alcance los 1000, 5.000, 10.000 y 50.000, y para las rachas, cuando el usuario alcance una racha de 10, 20, 50, 100 y 365. Cabe destacar que, si el usuario es recompensado por llegar a una racha de 20 días, por ejemplo, y posteriormente pierde la racha, el trofeo lo mantiene.

*UT Asociadas:* Recompensas por rachas, recompensas por puntos.

*Pruebas de aceptación:*

- Que un usuario alcance un determinado número de racha y que efectivamente se le entregue el trofeo correspondiente.
- Que un usuario alcance un determinado número de puntos y que efectivamente se le entregue el trofeo correspondiente.
- Que inicialmente todos los usuarios no tengan ningún trofeo.

*Estimación en HIP:* 20h

## 4.3 Diseño de la interfaz de usuario

Hoy en día las Interfaces de Usuario (*IU*) y la Experiencia de Usuario (*UX*) se han convertido en componentes críticos para el éxito de cualquier aplicación. La *IU* se refiere al punto de interacción entre el usuario y el sistema, mientras que la *UX* abarca la percepción y respuesta del usuario frente a esa interacción. La relevancia de estos conceptos ha crecido exponencialmente en la era digital actual, donde la competencia es feroz y las expectativas de los usuarios son cada vez más altas.

### 4.3.1 Interfaz de usuario

La interfaz de usuario es fundamental ya que es el medio a través del cual los usuarios interactúan con un sistema. Incluye elementos visuales como botones, menús, iconos, y ventanas, así como elementos interactivos como formularios y enlaces.

El objetivo principal de la *IU* es facilitar la interacción del usuario con el sistema de una manera eficiente y efectiva ya que un buen diseño no solo facilita la usabilidad, sino que también puede aumentar la eficiencia, accesibilidad, coherencia visual y satisfacción del usuario.

En el caso de este proyecto, la base de las interfaces de la aplicación, es decir, el logo, tipografía, paleta de color y el estilo de los iconos que se muestra en las Figuras 16,17



y 18, fueron diseñadas por un equipo de estudiantes de Bellas Artes, dónde, en una colaboración para un proyecto, se desarrolló parte de esta.



Figura 16: Logotipo e imagotipo

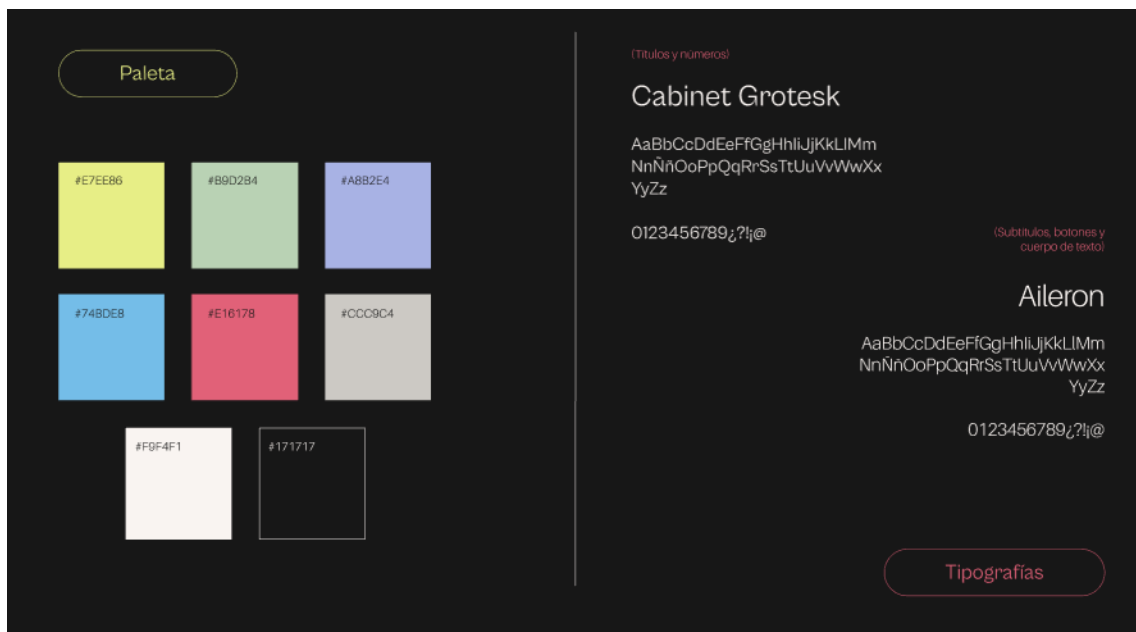


Figura 17: Paleta de colores y tipografías

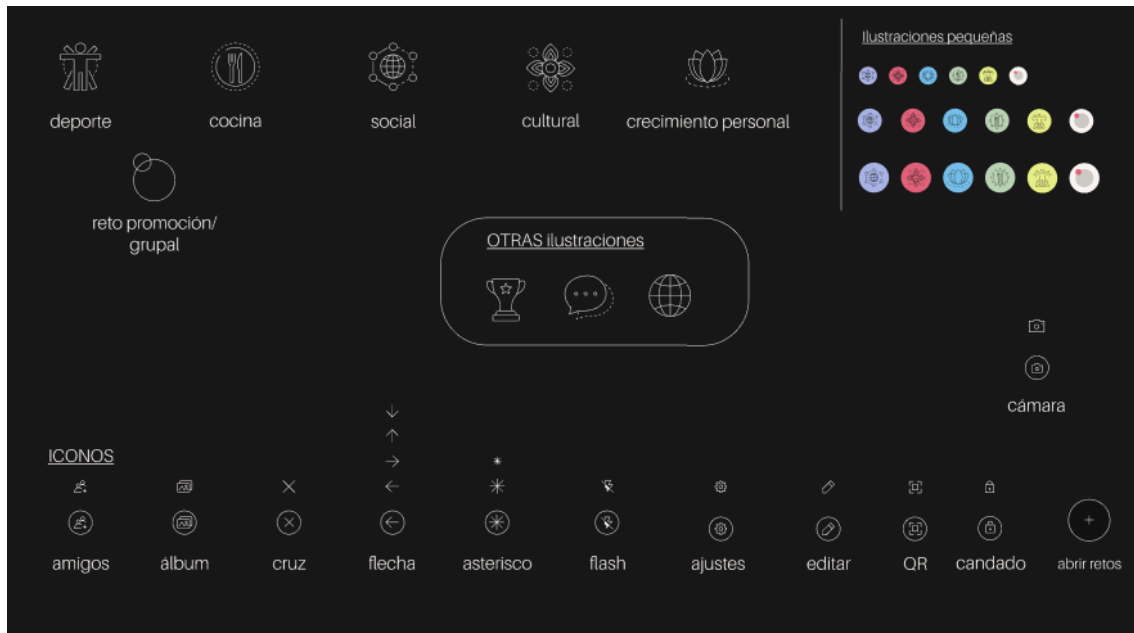


Figura 18: Iconos e ilustraciones

Algunas de las pantallas desarrolladas durante los inicios del desarrollo cómo la de inicio o la del perfil de las Figura 19, fueron diseñadas totalmente por este equipo.

Como se puede ver comparando el diseño con el resultado final de la aplicación, se ha intentado ser lo más fieles al diseño original, siguiendo todas las pautas marcadas, y siguiendo tanto las relaciones de márgenes y tamaños, como la calidad de los diseños, además de un diseño responsive para poder adaptarse a todo tipo de dispositivos.

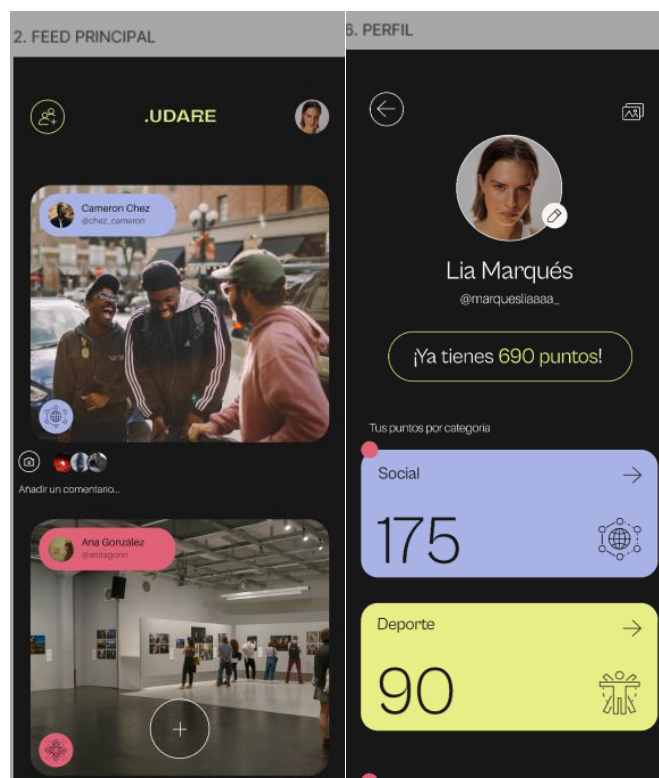


Figura 19: Feed principal y pantalla de perfil

### 4.3.2 Experiencia de usuario

Como ya se comentó en la redacción del primer *MVP*, los inicios de este proyecto pertenecen a una asignatura del grado, por lo que ésta llegó a su fin, y las interfaces y los iconos de las funcionalidades implementadas, como el ranking en la Figura 20 o la pantalla de trofeos, fueron diseñadas por el equipo de desarrollo, siendo fieles al estilo marcado por el equipo de diseño.

Para el diseño de las nuevas funcionalidades, se decidió implementar una pantalla por color de la paleta proporcionada. Es decir, así como el inicio de sesión, su fondo es azul, o el álbum amarillo; el ranking se decidió hacer verde y la pantalla de trofeos azul. Además, para los diseños adicionales, se diseñaron nuevos iconos como los que se pueden ver en la Figura 21.

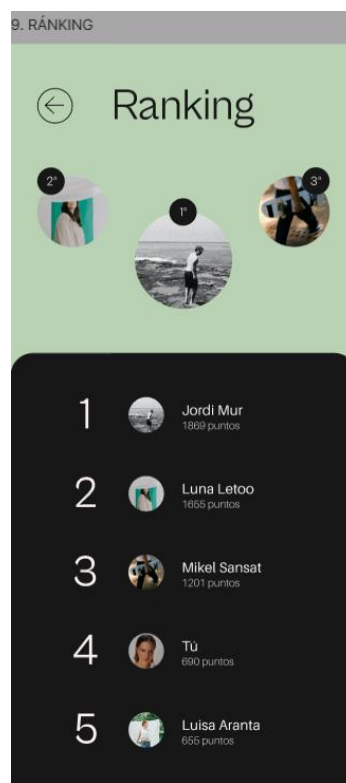


Figura 20: Ranking

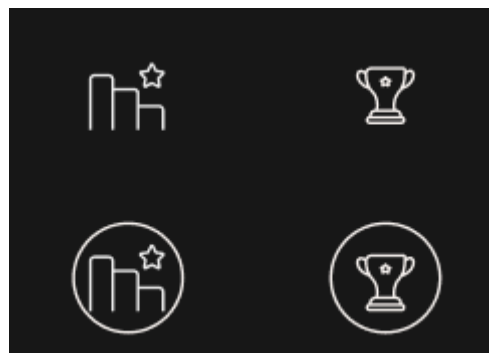


Figura 21 : Nuevos iconos

La experiencia de usuario se refiere a la percepción y respuesta del usuario resultante del uso de un producto, sistema o servicio. Abarca todas las interacciones del usuario con el sistema, incluyendo sus emociones, creencias, preferencias, percepciones, respuestas físicas y psicológicas, comportamientos y logros que ocurren antes, durante y después del uso, por lo que su principal objetivo es proporcionar una experiencia positiva, satisfactoria y libre de frustraciones.

El diseño de la *UX* requiere una planificación meticulosa para asegurar una experiencia de usuario fluida e intuitiva. Para lograr este objetivo se ha decidido hacer un diagrama de actividades, como se puede observar en la Figura 22. Un diagrama de actividades es una representación gráfica que ilustra el flujo de control y de datos de un proceso o sistema. Se centra en el comportamiento dinámico del sistema, mostrando las actividades, decisiones, flujos concurrentes y sincronizaciones que ocurren dentro de un proceso. Cada actividad representa una pantalla o vista específica de la aplicación, mientras que las transiciones indican la navegabilidad entre ellas. En este caso hemos decidido utilizar las líneas con flecha para las navegabilidades restringidas, es decir, que no se puede volver hacia atrás, y líneas sin flechas, que indica que puedes ir en ambos sentidos.

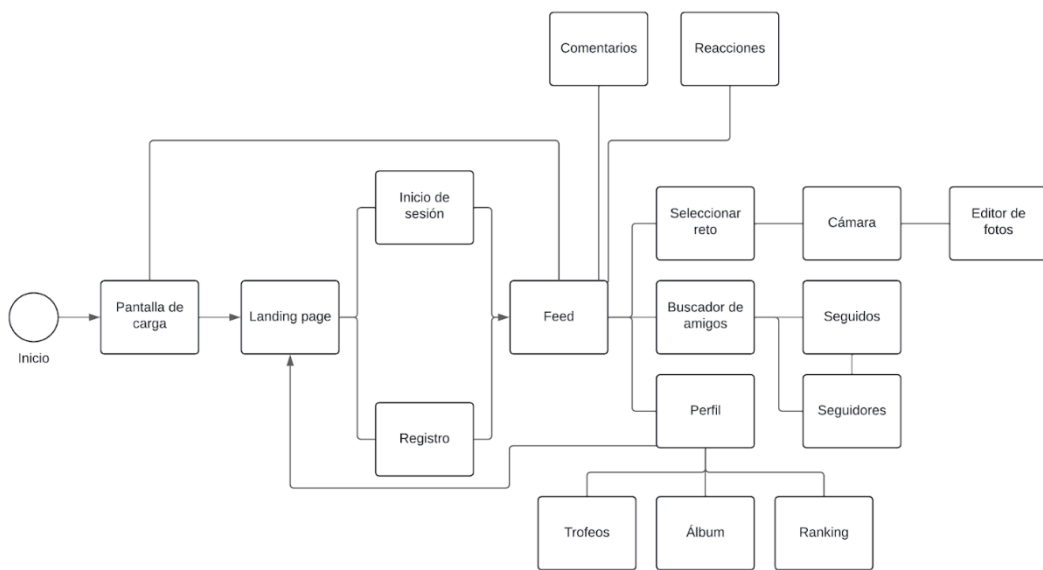


Figura 22 : Diagrama de actividades

#### 4.4 Diseño del código y la base de datos

El diseño de una aplicación es de suma importancia, ya que sienta las bases para el desarrollo de un producto final efectivo y exitoso. Esta etapa inicial no solo determina la apariencia visual y la experiencia del usuario, sino que también establece la estructura arquitectónica y funcional del software. En este proyecto se han intentado abarcar las máximas especificaciones posibles, y por eso se han decidido especificar los siguientes apartados.





#### 4.4.1 Arquitectura

Para diseñar la arquitectura de la aplicación, se ha optado por seguir la Clean Architecture [13], un patrón de diseño arquitectónico que promueve la separación de preocupaciones y la independencia de detalles técnicos específicos dentro de una aplicación. Este enfoque está concebido para desarrollar sistemas que sean fáciles de mantener y actualizar con el tiempo, facilitando las pruebas unitarias y promoviendo la escalabilidad. Se centra en minimizar la dependencia de detalles técnicos externos, enfocándose en la lógica central del negocio.

Un proyecto Android está dividido previamente en dos partes principales. Para aplicar esta arquitectura, se consideran dos particiones más generales. En la primera partición, siguiendo los principios mencionados, se encuentra la capa de lógica, responsable de implementar las diversas funcionalidades y de cargar datos en la interfaz de usuario. La segunda partición incluye una capa de servicios, encargada de la comunicación con el backend, y una última capa que define los modelos específicos de la aplicación. En la Figura 23 se pueden ver las carpetas que corresponden a lo redactado anteriormente.

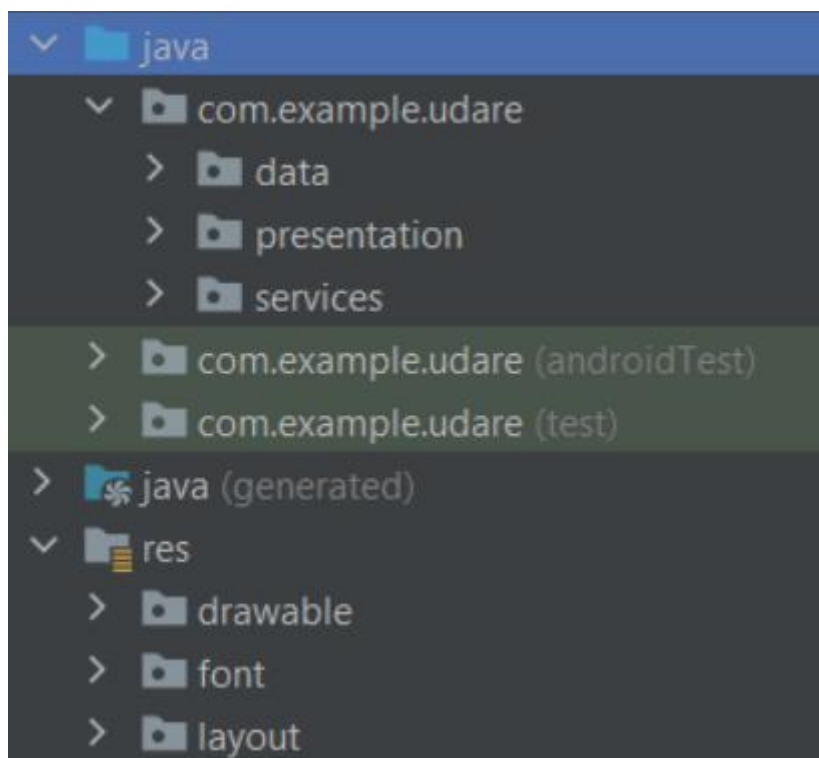


Figura 23: Estructura del proyecto

En la segunda parte del proyecto, donde se definen las interfaces, también se ha optado por estructurar en particiones específicas. Una de estas particiones alberga las pantallas que serán mostradas, asegurando así una gestión efectiva de la presentación visual del contenido. En una segunda partición se encuentran los diversos iconos y las imágenes estáticas que se utilizarán a lo largo de las diferentes pantallas, permitiendo una organización eficiente de los recursos visuales del proyecto. Finalmente, una última partición está reservada para gestionar las fuentes tipográficas de la aplicación,

asegurando consistencia y coherencia en la presentación del texto en toda la interfaz. Estas 3 secciones corresponden a las carpetas de la Figura 23 “layout”, “drawable” y “font” respectivamente.

#### 4.4.2 Diagrama de clases

Un diagrama de clases es una representación gráfica que pertenece al conjunto de herramientas de la metodología *UML (Unified Modeling Language)*<sup>46</sup>, que se emplea para visualizar, especificar, construir y documentar los artefactos de un sistema de *software*.

El diagrama de clases de la Figura 24, muestra la estructura estática del sistema representando las clases que lo componen, así como las relaciones entre ellas. Cada clase se ilustra mediante un rectángulo dividido en tres partes: el nombre de la clase, los atributos y las operaciones o métodos.

Para simplificar el diagrama y hacerlo más agradable visualmente se ha decidido incluir solamente las clases, ya que son las entidades básicas que encapsulan los datos y comportamientos, y las relaciones entre ellas, por lo que se han omitido los atributos y métodos.

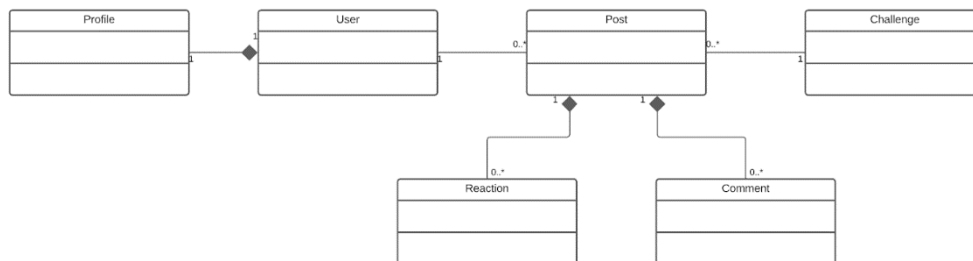


Figura 24: Diagrama de clases

En cuanto a las relaciones, hay de 4 tipos, de asociación que representa una relación entre dos clases, de herencia que denota una jerarquía, y la agregación y composición, que implican relaciones “todo/parte”.

La agregación, representada con un rombo transparente, es menos restrictiva ya que las partes pueden existir independientemente, y la composición, representada con un rombo negro, indica una relación más fuerte dónde las partes no pueden existir sin el todo.

<sup>46</sup> *UML (Unified Modeling Language)*: Lenguaje de modelado estandarizado que se utiliza para visualizar, especificar, construir y documentar los componentes de un sistema de software.



En el diagrama, aparecen dos tipos de relaciones, asociaciones y composiciones. De las del segundo tipo, se ha decidido considerarlas así porque la destrucción de un usuario conlleva la destrucción de su perfil, y lo mismo con los posts y las reacciones y comentarios.

Como se ha comentado anteriormente, no aparecen ni los atributos ni los métodos para simplificar el diagrama, pero a grandes rasgos, todas las clases tienen los atributos necesarios para su definición y los *getters* y *setters* correspondientes para poder acceder y modificarlos.

#### 4.4.3 Diagrama de despliegue

Para desarrollar mejor la arquitectura del sistema, también hemos definido el diagrama de despliegue que aparecen en la Figura 25, dado que el *front-end* se comunica con dos servicios, *Firebase* para llevar la gestión de autenticación y una máquina virtual *AWS*, utilizando el servicio *EC2*, dónde se está ejecutando un servidor en *Node.js*.

Esto se ha hecho debido a que *Firebase* proporciona *SDKs* fáciles de integrar con diversas plataformas de desarrollo, cómo *Android Studio*, y proporciona servicios de autenticación con diferentes servicios como *Google*, *Facebook* o *Apple* sin necesidad de desarrollar. Además de todos estos beneficios, también mejora y evita el desarrollo para el almacenamiento y transferencia de contraseñas, ya que las codifica ya por defecto, cosa que en el *back-end* deberíamos de haber desarrollado.

Esto funciona de tal manera que cuando se registra un nuevo usuario, lo hace en ambos servidores, en *firebase* se almacena la contraseña y el usuario, y en el *backend* un *uid* que pertenece al creado en *firebase*. Cuando se inicia sesión se obtiene ese *uid* desde *firebase*, se consulta en la base de datos y se obtienen todos los datos restantes del usuario.

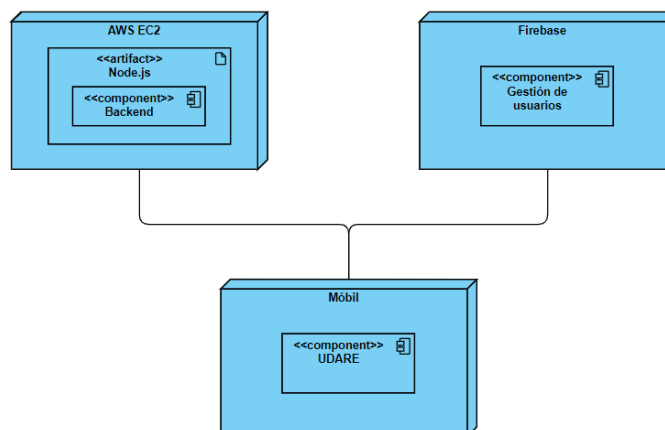


Figura 25: Diagrama de despliegue

#### 4.4.4 Comunicación con el *backend*

En cuanto a las llamadas a la *API*, se ha implementado una estructura de gestión de llamadas que sigue el principio de separación de responsabilidades de los principios *SOLID*<sup>47</sup>, que manifiesta que una clase, método o módulo debería de tener solamente una responsabilidad, es decir, solamente una razón para cambiar. Para ello, se ha dividido el código en varias capas: una capa de red y una de repositorios. Esta organización mejora la mantenibilidad y escalabilidad del código.

En la capa de red, se ha definido una interfaz dónde se definen los *endpoints* de la *API*, y su correspondiente implementación de la lógica para interactuar con ella utilizando *retrofit*<sup>48</sup>, una librería de cliente *HTTP* (*Hyperext Transfer Protocol*<sup>49</sup>) tipo *REST* (*Representational State Transfer*<sup>50</sup>) para *Android* y *Java* desarrollada por *Square*<sup>51</sup>. Esta librería se utiliza para simplificar las comunicaciones de red y convertir automáticamente las respuestas *JSON* (o *XML*) de una *API* en objetos *Java*, posteriormente convertidos en *Kotlin*, a través de llamadas *RESTful*.

En cuanto a la capa de repositorios, se ha diseñado una interfaz que define las operaciones disponibles desde el *front-end*, y su implementación a través de la capa de servicios. Estas operaciones serán utilizadas en la capa de presentación para implementar las distintas funcionalidades de la aplicación.

Como anteriormente se ha dicho, se han utilizado llamadas *RESTful*, que se refiere a la implementación de los principios *REST* en el diseño de servicios web.

En una *API RESTful*, los recursos son entidades clave que se pueden identificar de manera única a través de *URIs* (*Uniform Resource Identifiers*<sup>52</sup>). Cada recurso debe tener un *URI* que lo identifique de manera única y que sea accesible a través de métodos estándar de *HTTP* como *GET*, *POST*, *PUT* y *DELETE*.

Como se puede ver en la Figura 26, hay 3 ejemplos claros del uso desde el *front-end*, *GET* que solicita información al servidor, *POST* que crea nuevos registros y *PUT* para actualizar registros ya establecidos. En este caso, en la primera llamada se solicita una lista con todos los usuarios, en la segunda se añade un nuevo usuario a la base de datos y en la última se actualiza su foto de perfil.

---

<sup>47</sup> *SOLID*: Conjunto de cinco principios de diseño de software orientado a objetos que buscan mejorar la calidad y la mantenibilidad del código.

<sup>48</sup> *Retrofit*: <https://square.github.io/retrofit/>

<sup>49</sup> *HTTP* (*Hyperext Transfer Protocol*): Protocolo de comunicación utilizado para transferir datos a través de la web.

<sup>50</sup> *REST* (*Representational State Transfer*): Estilo de arquitectura para diseñar servicios web. Se basa en el uso de estándares como *HTTP* y define un conjunto de principios para la comunicación entre sistemas.

<sup>51</sup> *Square*: <https://squareup.co>

<sup>52</sup> *URIs* (*Uniform Resource Identifiers*): Cadena de caracteres que identifica un recurso en Internet de manera unívoca.

```
1 usage  👤 David Sanjaime +1
@GET("/users")
Call<List<User>> getAllUsers();

1 usage  👤 Miguel Wagner
@POST("/users/add")
Call<User> createUser(@Body User user);

1 usage  👤 Miguel Wagner *
@Multipart
@PUT("/users/updateProfilePic")
Call<User> updateProfilePic(
    @Part MultipartBody.Part image,
    @Part MultipartBody.Part user
);
```

Figura 26: Ejemplos uso RESTful

## 4.5 Programación

A continuación, procederemos a abordar aspectos relacionados con el desarrollo del código, tales como los patrones de diseño implementados, las refactorizaciones realizadas, y los diversos desafíos de programación enfrentados durante el desarrollo del proyecto.

### 4.5.1 Patrones de diseño

Los patrones de diseño son soluciones probadas y reutilizables para problemas comunes que surgen en el diseño y desarrollo de software. Estos patrones proporcionan un marco estandarizado y bien documentado para abordar desafíos específicos, facilitando la creación de sistemas robustos, mantenibles y escalables. Su utilización es esencial ya que promueven las mejores prácticas y ayudan a evitar errores comunes.

#### 4.5.1.1 Singleton

El patrón de diseño *Singleton* está diseñado para restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto. Su intención consiste en garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella. No se encarga de la creación de objetos en sí, sino que se enfoca en la restricción en la creación de un objeto [16].

Para el proyecto, se ha decidido implementarlo para restringir la creación de usuarios, y gestionar el acceso a su información de manera global y así poder acceder a cualquier variable que se necesite desde cualquier punto de la aplicación.

Para su implementación se ha decidido añadir una clase llamada *UserSingleton*, con un constructor privado para evitar la creación de instancias directas. En esta clase se crea un objeto instancia, que se establece una única vez al iniciar la aplicación, así nos aseguramos de que no tenemos dos usuarios iniciados a la vez. También se implementan los métodos necesarios para modificar atributos que se actualizan mediante el uso de la aplicación.

En la Figura 27 podemos ver diferentes métodos implementados a los que se accede al iniciar sesión, cerrar sesión, obtener el usuario para acceder a sus diferentes atributos y un último método para actualizar la foto del usuario.

```
public void logIn(User user) {this.user = user;}

public void logOut() { user = null; }

public User getUserLogged() { return this.user; }

public void updatePhoto(String url) { this.user.getProfile().setProfilePic(url); }
```

Figura 27: Métodos de la clase *UserSingleton*

#### 4.5.1.2 Adapter

El patrón de diseño *Adapter* se trata de un patrón estructural, es decir, se enfoca en cómo se componen y organizan los objetos para formar estructuras más grandes y complejas. Específicamente, en el caso de las interfaces de usuario, se usa para adaptar datos para ser mostrados en componentes visuales [17].

Un adaptador envuelve uno de los objetos para esconder la complejidad de la conversión que tiene lugar tras bambalinas. El objeto envuelto ni siquiera es consciente de la existencia del adaptador. Por ejemplo, puedes envolver un objeto que opera con metros y kilómetros con un adaptador que convierte todos los datos al sistema anglosajón, es decir, pies y millas.

En nuestro caso, se han creado los adaptadores para poder cargar información de diferentes objetos en un mismo *item*. Siendo más específicos, en la interfaz principal, o *feed* se muestran las fotos correspondientes a los *posts*, pero además también se muestra la información del usuario que ha subido la foto, y las reacciones de otros

usuarios a ese mismo *post*. Para conseguirlo, se ha creado una nueva clase a la cual se le pasa la información necesaria de cada *item*, y así la información del usuario y las diferentes reacciones corresponden con el *post*.

Posteriormente, una vez creado el nuevo objeto adaptador, se le pasa a otra clase dónde se procesarán los datos y se cargarán en la vista. Para esto se ha utilizado la librería proporcionada por *Android Studio* llamada *RecyclerView*<sup>53</sup>, que se encarga de cargar los diferentes *ítems*, con la misma estructura, pero con diferentes datos. Como podemos ver en la Figura 28, así es como se ve este objeto sin ninguna información cargada sobre él, y en la siguiente Figura 29, un ítem que se cargará en dicho contenedor.

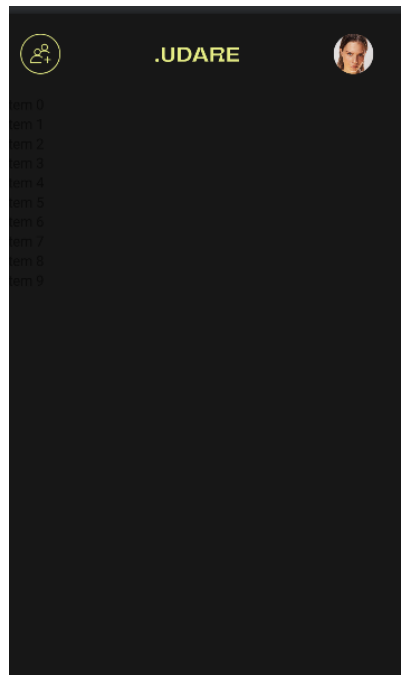


Figura 28: RecyclerView vacío

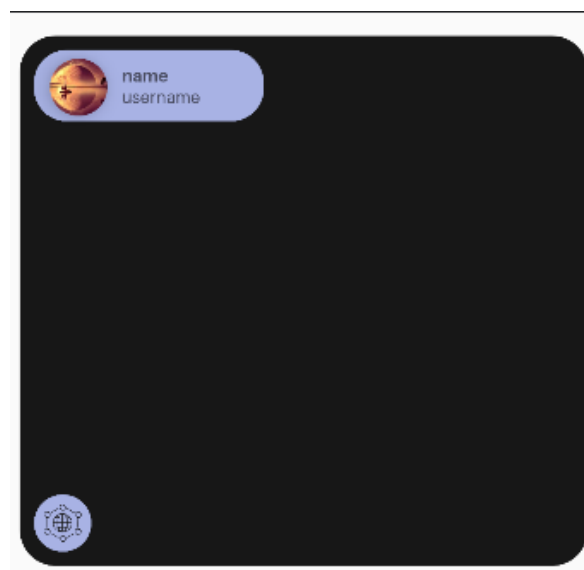


Figura 29: Item RecyclerView

<sup>53</sup> *RecyclerView*: <https://developer.android.com/jetpack/androidx/releases/recyclerview>

## 4.5.2 Refactoring

La refactorización del código es el proceso de modificar y reorganizar el código fuente de una aplicación sin cambiar su comportamiento externo. El objetivo principal de la refactorización es mejorar la estructura interna del código para hacerlo más legible, mantenible y eficiente. Este proceso implica aplicar una serie de técnicas y patrones de diseño que optimizan la calidad del código, reduciendo la complejidad y eliminando redundancias.

Este proyecto ha tenido mucho trabajo en esta tarea debido a que es fruto, como ya se ha comentado anteriormente, de un proyecto conjunto del grado, por lo que cada uno tenía un estilo de programación, con unas manías propias, y con poca experiencia en desarrollo, por lo que hay mucha diversidad de estilos.

### REFACTORING DE LA PANTALLA DE INICIO

El mayor trabajo de refactorización en UDARE ha sido provocado por la pantalla de inicio, ya que requiere una considerable cantidad de información del *back-end*, lo que ralentiza significativamente el rendimiento de la aplicación. Para mejorar este aspecto, se decidió llevar a cabo diversas refactorizaciones que se explicarán a continuación

Como primera medida, y que resultó en una notable mejora del rendimiento, se optó por solicitar las publicaciones al *back-end* en lotes de 10, para así no manejar un volumen tan grande de datos. La gestión del envío la realiza el *back-end*, pero en el *front-end* también se ha implementado una función que, al llegar al último *post*, carga las siguientes 10 imágenes.

Otra de las soluciones propuestas fue utilizar la carga progresiva de imágenes en *Android*. Para la carga se empleó la librería *Glide*<sup>54</sup>, que permite cargar imágenes procedentes de una *URL*. Esta librería proporciona una opción que muestra el progreso de carga de la imagen, presentándola primero de forma borrosa y, a medida que se carga, se va viendo más nítida. En la Figura 30 se puede observar el uso de esta librería y su método *thumbnail*, que implementa lo mencionado anteriormente.

```
Glide.with(holder.imageView) RequestManager
    .load(list[position].post.image) RequestBuilder<Drawable>
    .apply(options)
    .thumbnail( sizeMultiplier: 0.05f)
    .transition(DrawableTransitionOptions.withCrossFade())
    .diskCacheStrategy(DiskCacheStrategy.ALL)
    .into(holder.imageView)
```

Figura 30: Ejemplo de la carga de imágenes mediante Glide

<sup>54</sup> *Glide*: <https://github.com/bumptech/glide>



Finalmente, se decidió cambiar la ubicación de la petición y carga de imágenes. Las actividades en *Android* tienen un ciclo de vida que se resume en 7 métodos que se sobrescriben según las necesidades del programador: *onStart*, *onCreate*, *onResume*, *onPause*, *onStop*, *onDestroy* y *onRestart*. Inicialmente, debido al desconocimiento del funcionamiento de este ciclo, la gestión de petición y carga de imágenes se realizaba en el *onCreate*, pero se decidió trasladarla al *onStart* para mejorar su rendimiento.

Otra de las refactorizaciones llevadas a cabo en el *front-end* fue la mejora del flujo de actividades de la aplicación. Anteriormente, cada vez que se navegaba a otra pantalla, se cargaban los datos desde cero y no se cerraba la actividad anterior, lo que creaba un flujo infinito y una sobrecarga de la memoria, además de ralentizar el flujo debido al bajo rendimiento. Esto se implementó así debido al escaso conocimiento del funcionamiento de *Android* y la inexperiencia del equipo. No obstante, se llegó a una mejor solución añadiendo la opción de *finish()* al volver a la pantalla de procedencia, para no sobrecargar la memoria y disponer de datos previamente cargados.

Algunas refactorizaciones menos importantes para el producto final, pero a los que también se les ha dedicado tiempo han sido adaptar la nomenclatura de todo el código kotlin a la Kotlin Style Guide<sup>55</sup>, ya que al ser un proyecto grupal en sus inicios y no acordar un estilo, había nombres de variables y métodos en diferentes idiomas y estilos, por lo que era inviable para el futuro mantenimiento y escalado de la aplicación.

#### 4.5.3 Desafíos de programación

El mayor desafío de programación encontrado en la aplicación, y prácticamente en la mayoría de las aplicaciones móviles ha sido el de confeccionar un diseño responsive. Esto se basa en proporcionar una experiencia de usuario óptima en una amplia gama de dispositivos y tamaños de pantalla.

Además de incluirse como *refactoring*, ya que la mayoría de las pantallas ya hechas no incluían este principio, es un objetivo difícil de lograr debido a que las proporciones no siempre serán las mismas y los objetos para *Android* no se adaptan siempre a todo tipo de pantalla. Además de todo esto, se añade la dificultad de que no se puede probar en todos los dispositivos *Android* existentes, por lo que nunca se va a poder determinar esta parte como finalizada.

Esta problemática se solucionó utilizando *ConstraintLayouts* junto con *RelativeLayouts*, además de medidas responsive como porcentajes, o *match\_parent*, que se adapta al tamaño del contenedor padre.

Otro desafío de programación con el que nos encontramos fue el procesamiento de fotografías. En el desarrollo de la funcionalidad para tomar imágenes, cuando se capturaban se guardaban en la galería, y luego se mostraban en la interfaz, por lo que, al aplicar los filtros, no se aplicaban sobre la imagen guardada, si no sobre la que se está mostrando. Esto complicó el desarrollo de los filtros ya que tuvimos que cambiar la

---

<sup>55</sup> Kotlin Style Guide: <https://developer.android.com/kotlin/style-guide>

gestión de la captura de imágenes para que primero se muestre esa imagen por interfaz, y luego se guarde una vez aplicados los filtros.

## 4.6 Pruebas

Las pruebas son procesos sistemáticos diseñados para evaluar la calidad y funcionalidad de un *software*, asegurando que cumpla con los requisitos especificados y que esté libre de defectos. Representan una fase crítica dentro del ciclo de vida del desarrollo de *software*.

Para asegurar el buen funcionamiento del *software* se han realizado las pruebas de aceptación mencionadas en el apartado 4.2 Requisitos. Como bien comentamos en el apartado de metodología, todas las unidades de trabajo han tenido un *workflow*, y una de las tareas incluidas en este flujo es la de aplicar pruebas de aceptación.

En el primer MVP, como utilizamos la plataforma de gestión de proyectos de *worki*<sup>56</sup> en la asignatura de PIN, hicimos el seguimiento de estas pruebas mediante dicha plataforma. En esta, definimos todas las pruebas de aceptación de cada unidad de trabajo, y tras realizarla indicamos si cumplía con los requisitos o no.

En el segundo MVP, este seguimiento la herramienta de gestión de proyectos fue trello, por lo que utilizamos la función de *checklist* para hacer una lista con todas las pruebas de aceptación, y marcar con un *check* las que se cumplían como se puede ver en la Figura 31. Cuando había algún error no esperado en alguna prueba, y teníamos que reprogramar esa unidad de trabajo, escribíamos un comentario describiendo que prueba de aceptación había fallado, y cuál era el error no esperado.

---

<sup>56</sup> *Worki*: <https://cliente.tuneupprocess.com/web/>



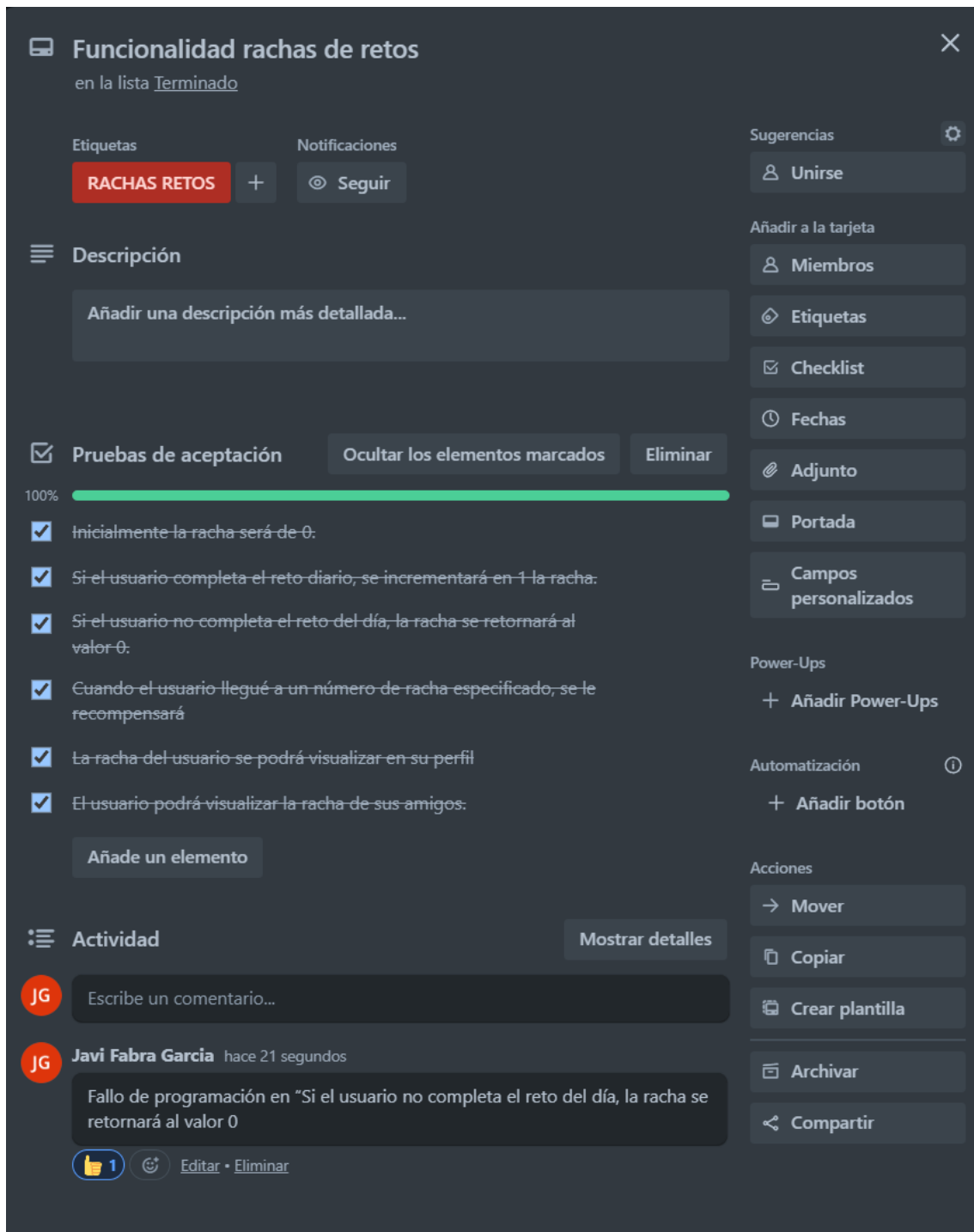


Figura 31: Ejemplo de seguimiento de pruebas de aceptación

## 5. CRONOLOGÍA DEL TFG

---

A continuación, se comentará la planificación del proyecto dónde se detallarán todos los *MVP* que se han realizado, así como los experimentos correspondientes.

### 5.1 Desarrollo del primer MVP

Como ya se ha comentado anteriormente, esta primera versión de la aplicación se llevó a cabo anteriormente y no se considera dentro de este TFG, por lo que no se entrará en mucho detalle acerca de su desarrollo. Lo único relevante y que nos concierne en este caso son las funcionalidades que se implementaron, ya que se necesitan conocer para poder realizar los experimentos correctamente.

En la Figura 32 se pueden observar todas las funcionalidades que se implementaron en este primer *MVP*. Entre estas, se pueden ver las funcionalidades más básicas de cualquier red social, como la autenticación de usuario, permitir subir, y mostrar las fotos en el *feed*, etc. También se incluyeron un par de funcionalidades más destacables entre el resto, como son las reacciones a las fotos, el calendario de fotos y los retos patrocinados.



Figura 32: Funcionalidades primer MVP

## 5.2 Primer experimento

El objetivo principal de este primer experimento es obtener retroalimentación de los usuarios acerca del producto que se ha desarrollado. Es muy importante saber qué opina nuestro público objetivo, para de esta forma saber si lo que se ha implementado realmente les gusta. También se utilizará este experimento para identificar cuáles son las funcionalidades que a los usuarios les gustaría que estuviesen en la aplicación y no están.

Para poder identificar cuáles podrían ser las nuevas funcionalidades para implementar en el segundo *MVP* y poder validarlas con los usuarios a través de este experimento, se optó por realizar una sesión de *Brainstorming*. El objetivo de la sesión fue crear un espacio de creatividad que diera lugar a la generación de ideas nuevas e innovadoras. Cada idea se iba apuntando en un papel, y una vez acabada la sesión, se le asignaba a cada idea una prioridad siguiendo el método *MoSCoW*, con el fin de ir descartando las menos interesantes. En la Figura 33 están las ideas clasificadas por colores y prioridad siendo las de color rojo las que pertenecen al *must*, las de azul al *should*, las de verde al *could* y las de amarillo al *won't*.

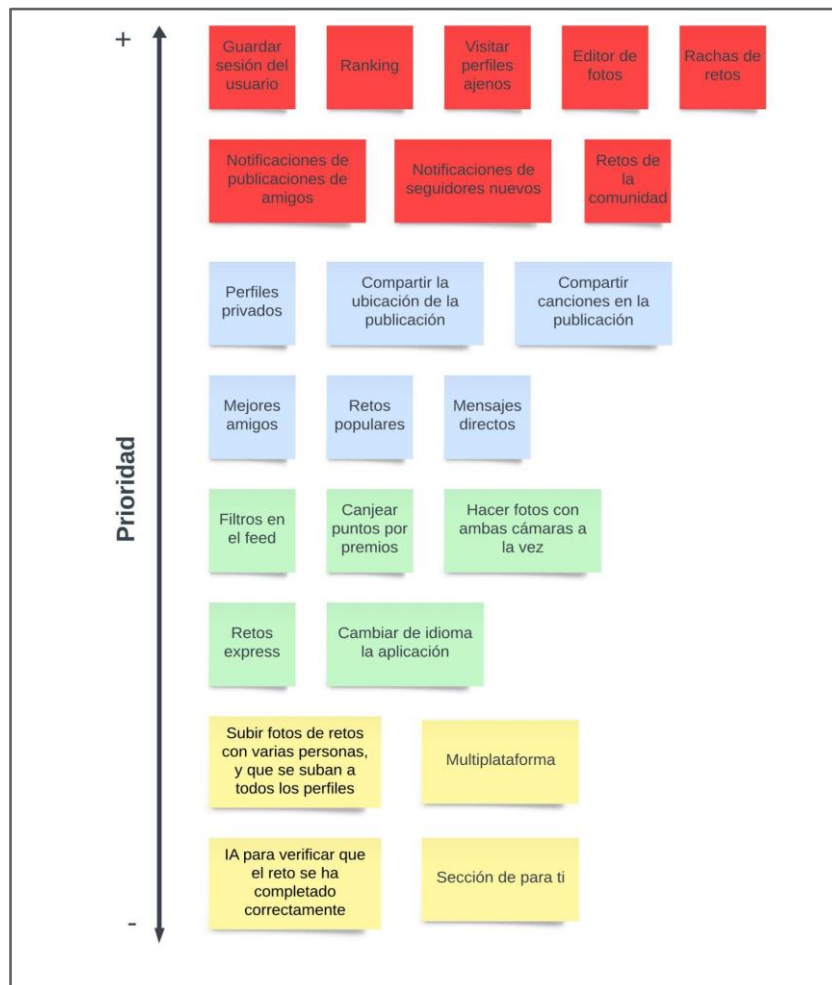


Figura 33: Funcionalidades por prioridad primer experimento

Posteriormente, se ha realizado un experimento compuesto por un grupo de 10 *Early Adopters*, que serán amigos cercanos o familiares dentro del público objetivo (de 16 a 30 años), para que prueben físicamente la aplicación libremente durante unos minutos y que posteriormente completen un cuestionario.

Se ha optado por realizar el experimento con un grupo tan reducido de gente cercana porque la aplicación no está desplegada en ningún servidor, por lo que el acceso a esta está muy limitado. Cabe destacar que es muy importante garantizar que cuando los usuarios prueben la aplicación lo hagan de una manera libre y sin intervención por nuestra parte, ya que se debe asegurarnos que la opinión del usuario sea totalmente objetiva y no esté influenciada por nosotros (los desarrolladores), básicamente porque nuestra perspectiva acerca de la aplicación puede estar sesgada debido a nuestro involucramiento directo en el proceso de desarrollo de la aplicación.

El cuestionario se realizará utilizando la herramienta de *Google Forms*, ya que de esta forma se pueden recopilar todas las respuestas de los usuarios de una manera eficiente y organizada. Esto nos permitirá posteriormente generar gráficas y tablas, lo que facilitará la comprensión de los resultados y nos ayudará a tomar mejores decisiones.

El cuestionario consta de una serie de preguntas que abordan diferentes aspectos de la aplicación. En primer lugar, se hacen preguntas acerca de la propia aplicación (experiencia de uso, interfaz, funcionalidades implementadas...). Seguidamente se pregunta si el usuario ha tenido la oportunidad de probar alguna aplicación similar a la nuestra anteriormente, con el fin de identificar la existencia de competidores directos y para corroborar que efectivamente es una aplicación innovadora. A continuación, se preguntan sobre cuánto estarían dispuestos los usuarios a pagar por una suscripción con ventajas adicionales, esto es crucial saberlo ya que nos proporciona información valiosa acerca del valor percibido por los usuarios y lo que estarían dispuestos a invertir en el producto, además, la estrategia financiera depende en gran parte de ello. Después se les pregunta acerca de la opinión que tienen sobre las posibles funcionalidades e ideas innovadoras que se han surgido durante la sesión de *Brainstorming* y que se podrían implementar. Por último, se hacen 3 preguntas abiertas, una para que el usuario nos indique libremente qué funcionalidades no le han gustado, otra para las que le gustaría que estuviesen y una tercera para que nos deje sus observaciones o sugerencias más personales.

Después de llevar a cabo el experimento, se hizo una reunión de retrospectiva para analizar los resultados obtenidos. Durante esta sesión, se llegaron a varias conclusiones significativas que se muestran junto a todas las preguntas y respuestas del cuestionario.

Los usuarios percibieron un alto nivel de usabilidad al interactuar con la aplicación como bien se muestra en la Figura 34.

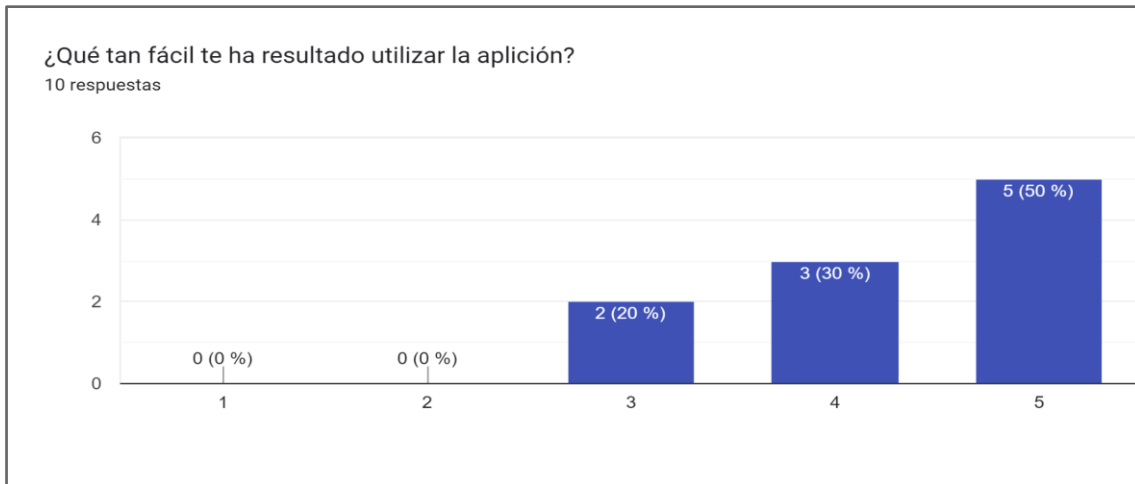


Figura 34: Pregunta cuestionario sobre usabilidad

En lo que respecta a la percepción estética de los usuarios, vemos como en la Figura 35 la gran mayoría la encontró atractiva y agradable.

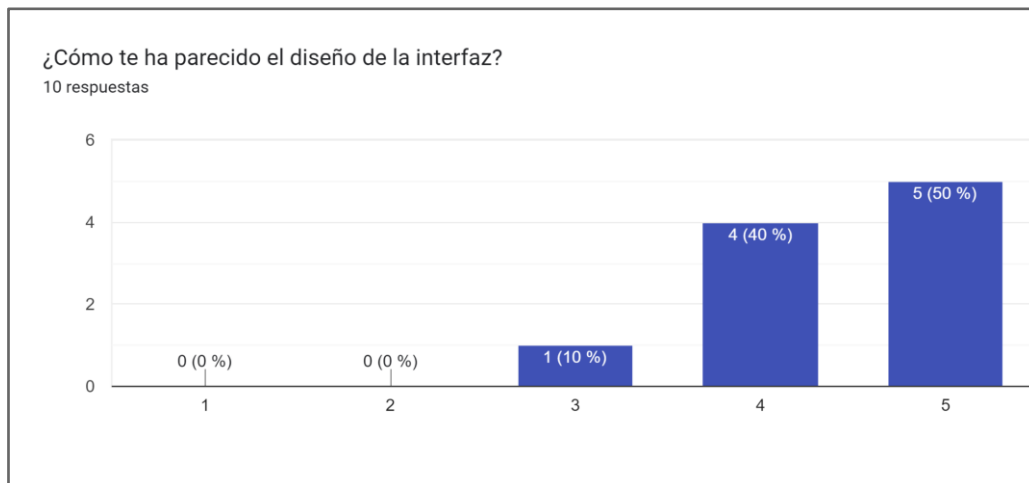


Figura 35: Pregunta cuestionario sobre la interfaz

En la Figura 36, ocho de los diez encuestados estarían interesados en descargarse y utilizar la aplicación.

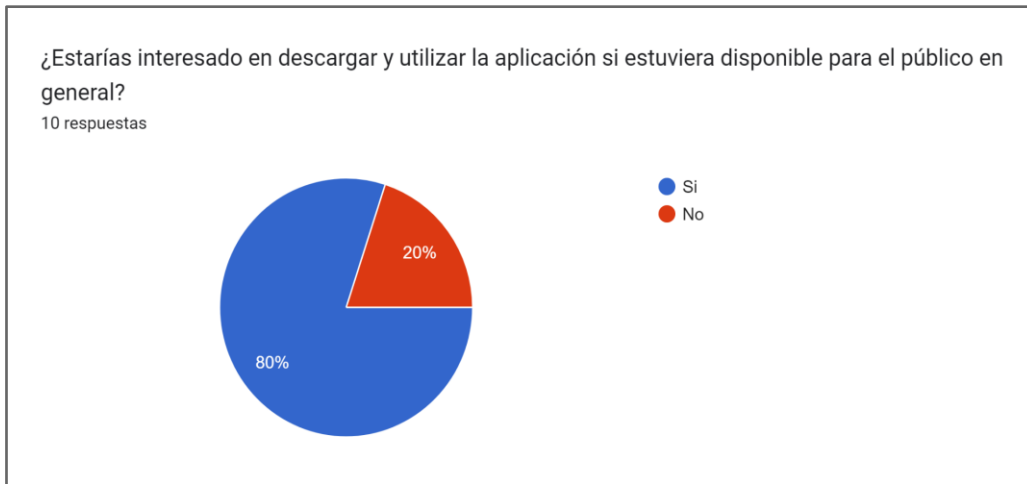


Figura 36: Pregunta cuestionario sobre el interés en la app

Además, en la Figura 37, un poco más de la mitad de los encuestados no han probado ni conocen una aplicación parecida.

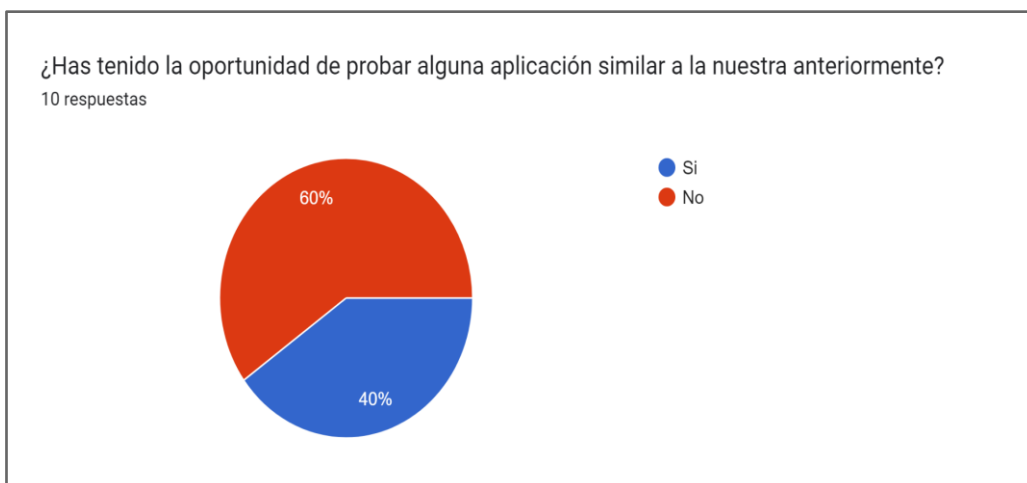


Figura 37: Pregunta cuestionario sobre apps similares

Aunque en la pregunta de la Figura 38 la mayoría no mostraron disposición por adquirir la función premium, un 20% si manifestó interés en hacerlo. Esta pregunta más allá de establecer el porcentaje de usuarios dispuestos a pagar nos proporciona una indicación acerca del intervalo de precios al que estarían dispuestos a comprometerse, situando dicho rango entre los 2€ y los 5€.



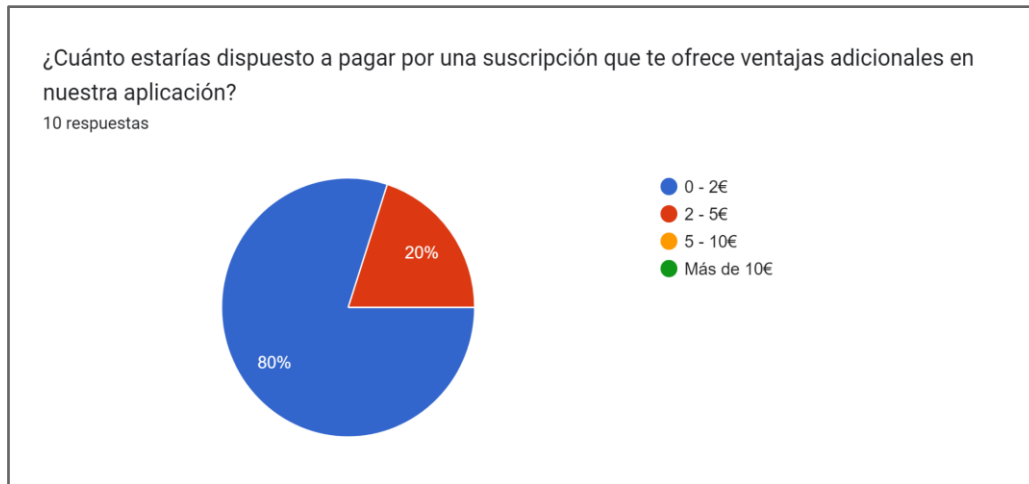


Figura 38: Pregunta cuestionario sobre el precio del premium

En lo que respecta a las preguntas relacionadas con las nuevas funcionalidades, como muestran las Figura 39, 40 y 41, los usuarios exhibieron mayor interés en las relativas a las rachas de retos, los retos de la comunidad y el ranking. Por otro lado, en relación con la opción de mejores amigos cuya pregunta aparece en la Figura 42, aunque algunos demostraron cierto interés, no generó un impacto significativo, dado que se percibe como una característica bastante común y repetitiva entre otras redes sociales.

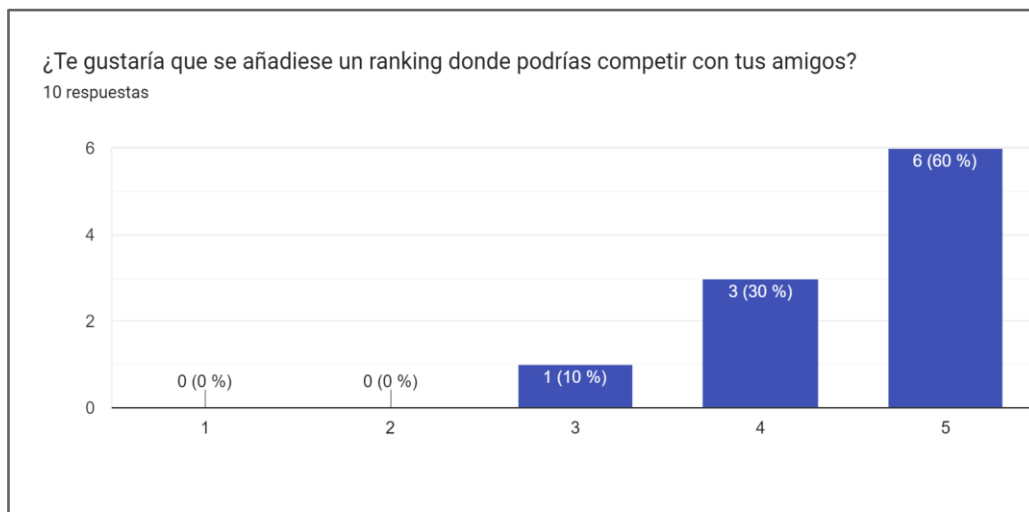


Figura 39: Pregunta cuestionario sobre el ranking

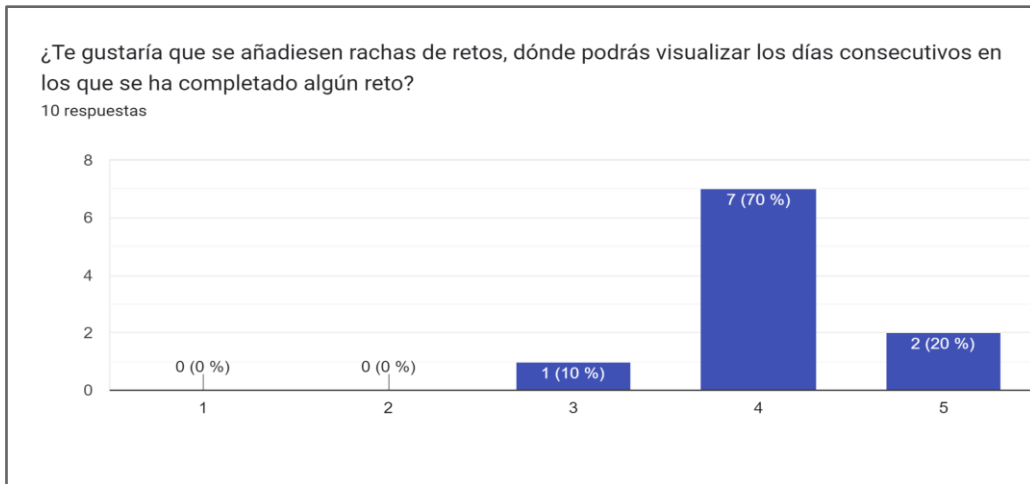


Figura 40: Pregunta cuestionario sobre las rachas

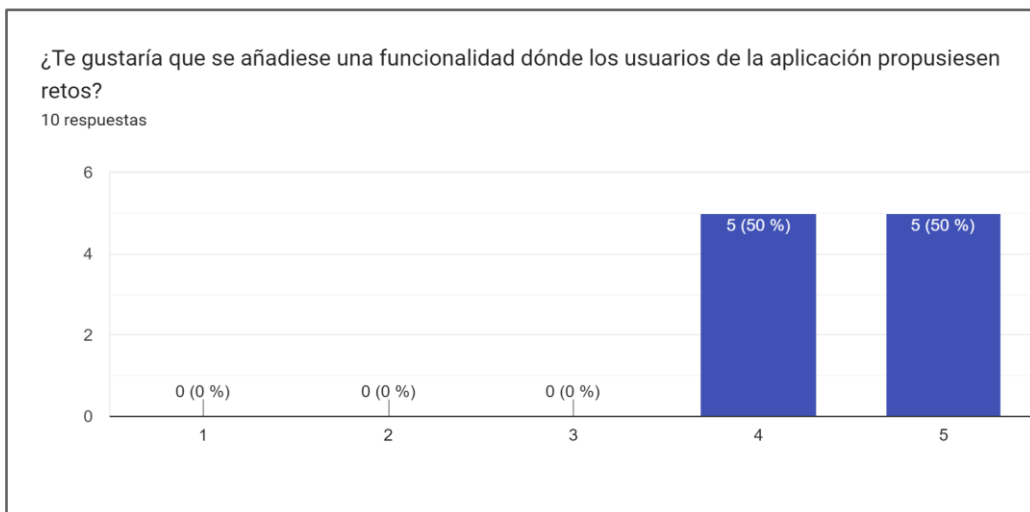


Figura 41: Pregunta cuestionario sobre los retos de comunidad

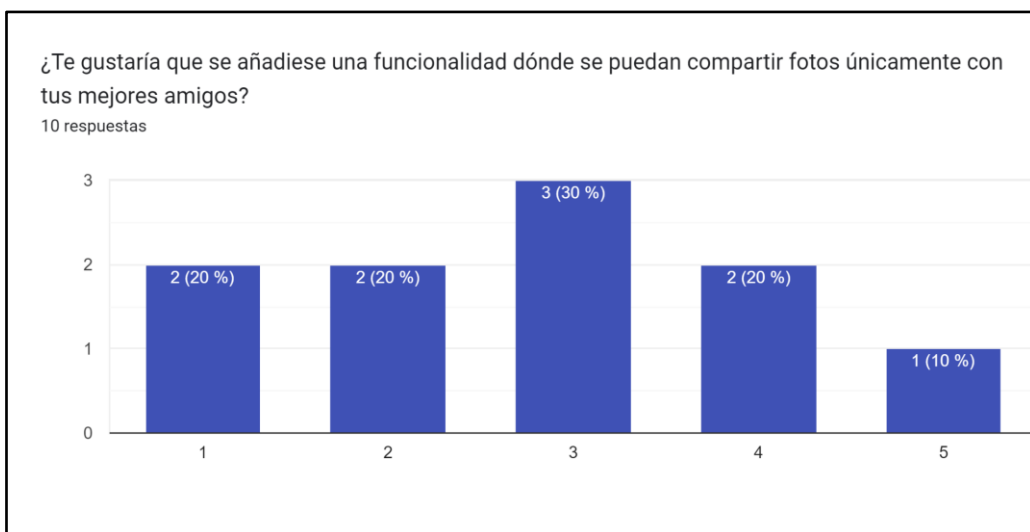


Figura 42: Pregunta cuestionario sobre mejores amigos

Entre las funcionalidades ya presentes, en la Figura 43, se recibieron críticas referentes a la imposibilidad de poder editar o repetir las fotos antes de subirlas, ya que estas se cargan automáticamente una vez capturadas. Asimismo, los usuarios expresaron disconformidad sobre la necesidad de introducir las credenciales cada vez que se accede a la aplicación, y se señaló la ausencia de la opción para establecer el perfil en privado

¿Qué funcionalidades de la aplicación no te han gustado y cambiarías?

5 respuestas

- Al subir la foto, no se puede editar y se sube automáticamente.
- Me gustaría poder poner mi perfil en privado
- Al subir una foto no me da la opción de editarla, se sube automáticamente.
- Hay que iniciar sesión cada vez que entras a la aplicación.
- Tengo que poner el correo y contraseña cada vez que entro a la aplicación

Figura 43: Pregunta cuestionario sobre funcionalidades que no gustan

Los usuarios expresaron interés en ciertas características adicionales que les gustaría que la aplicación incorporara. Entre estas, en la Figura 44, se encuentra la capacidad de explorar los perfiles y álbumes de otras personas, así como la posibilidad de plantear retos a amigos en concreto.

¿A parte de las comentadas anteriormente, qué funcionalidades crees que deberían haber o te gustaría que estuviesen en la aplicación?

5 respuestas

- Sería interesante que cada vez que se accede a la aplicación no fuese necesario introducir el nombre de usuario y la contraseña
- Poder proponer retos para mis amigos
- Quiero poder ver los perfiles de mis amigos
- Deberían poder verse los perfiles de mis amigos
- Ver los perfiles de mis amigos

Figura 44: Pregunta cuestionario sobre funcionalidades nuevas

Por último, una de las observaciones destacadas por los usuarios para mejorar la aplicación y que aparece en la Figura 45, es la necesidad de aumentar la velocidad de funcionamiento, tanto en la carga de imágenes y perfiles como en la rapidez al subir una fotografía.

Sugerencias y observaciones personales

3 respuestas

La aplicación está bastante bien, pero el concepto no me gusta porque se que no cumpliré los retos todos los días.

La aplicación va un poco lenta

Tarda mucho en subir una foto

Figura 45: Pregunta cuestionario

### 5.3 Desarrollo del segundo MVP

Tras el primer experimento, se ha reformulado el mapa de características, el cual se presenta en la Figura 46, añadiendo algunas funcionalidades básicas que consideramos durante el *Brainstorming* que no eran interesantes para el experimento, y decidiendo la prioridad de desarrollo. Las características en verde serán las que finalmente implementaremos en este segundo *MVP*.

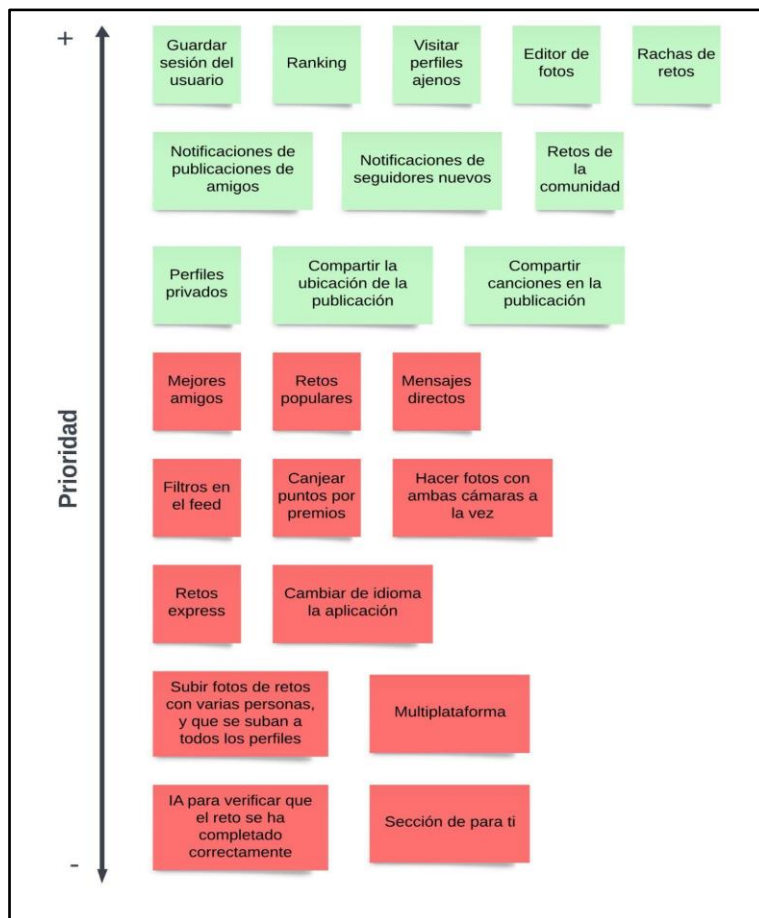


Figura 46: Funcionalidades por prioridad segundo MVP

A continuación, se realizará una descripción detallada de cada característica, así como la especificación de sus pruebas de aceptación. Por otra parte, también se hará una estimación orientativa del esfuerzo asociado a cada unidad de trabajo. El esfuerzo se estimará en Horas Ideales de Programación (*HIP*), es decir, las horas ininterrumpidas que se espera invertir en la actividad Programar.

En función de la complejidad de cada característica, algunas de ellas se desglosan en una o más unidades de trabajo (*UT*), para facilitar su gestión y desarrollo. Esta fase del proyecto se corresponde con la fase de especificación de requisitos:

A continuación, organizaremos el desarrollo de este segundo *MVP* y definiremos la cronología, así como las fechas claves para tener en cuenta:

Hemos decidido dividir el desarrollo del proyecto en 2 *sprints*. Cada *sprint* tendrá una duración de unas 3 semanas, y se le dedicarán aproximadamente 80 horas de *HIP* a cada uno (por persona).

El primer *sprint* abarcará desde el 6/05/2024 hasta el 26/05/2024. En este primer *sprint* le dedicaremos unas 20 horas a tareas de refactorización, mejora del código ya existente y a la aplicación de patrones de diseño. Teniendo en cuenta la duración del *sprint* y la estimación de tiempos realizada durante la fase de especificación de requisitos, hemos optado por implementar las siguientes funcionalidades: Guardar sesión del usuario, Ranking, Visitar perfiles ajenos, Editor de fotos y Rachas de retos.

El segundo *sprint* abarcará desde el 27/05/2024 hasta el 16/06/2024. Este *sprint* estará puramente enfocado a la implementación de las funcionalidades. Las funcionalidades para implementar serán las siguientes: Retos de comunidad, compartir la ubicación en la publicación, compartir canciones en la publicación, Perfiles privados y Notificaciones.

Justo las siguientes dos semanas después de acabar el segundo *sprint*, del 17/06/2024 al 1/07/2024, las dedicaremos a preparar y realizar el segundo experimento. Por lo tanto, a principios de Julio la aplicación debería estar prácticamente desarrollada, faltando únicamente acabar de redactar la memoria.

Cabe destacar que este es el plan inicial que llevaremos a cabo, pero en algún momento pueden surgir imprevistos durante el desarrollo que nos obliguen a ajustar nuestra planificación. Por ejemplo, nos podemos encontrar en la situación de que hemos estimado demasiado al alta y que todas las *UT* previstas para un *sprint* se completen antes de tiempo, lo que nos llevaría a la necesidad de añadir nuevas funcionalidades al proyecto. Por otra parte, también nos podemos encontrar en la situación de que hemos estimado a la baja y que, por lo tanto, nos vaya a tomar más tiempo desarrollar las *UT* del que habíamos previsto inicialmente. En este caso, deberíamos tomar decisiones acerca de qué funcionalidades deberíamos dejar fuera del alcance del proyecto.

**Final del primer *sprint*: 27/05/2024**

Tras la finalización del primer *sprint*, se llevó a cabo una reunión para analizar cómo fue su desarrollo y hacer un ejercicio de retrospectiva y de autocrítica. Esta reunión nos sirvió para identificar las funcionalidades que no se han llegado a completar (flecós) y para planificar el siguiente *sprint*.

En concreto, en la funcionalidad del ranking, durante las pruebas de aceptación nos dimos cuenta de que el propio usuario registrado en la aplicación no salía en el ranking, es decir, solo salían sus amigos. Una vez detectado el fallo, no nos dio tiempo a maniobrar y volver la *UT* a la actividad de programar ya que no teníamos tiempo, por lo que decidimos incluirla en el segundo *sprint* como fleco.

Con respecto a la funcionalidad de las rachas, decidimos orientar de otra forma las recompensas. Es decir, en vez de solo recompensar al usuario por las rachas, pensamos que resulta más interesante recompensar también por otros logros más genéricos, como, por ejemplo, premios por llegar a un determinado número de puntos. Por lo tanto, para el segundo *sprint* se creará una nueva *UT* llamada recompensas, definida también en los requisitos funcionales.

En conclusión, tras la reunión decidimos implementar para este segundo *sprint* el fleco del ranking y la *UT* de recompensas. El fleco al ser un pequeño error que tiene muy fácil solución no lo estimamos y no creamos una *UT* para ello, pero para las recompensas sí, ya que es una funcionalidad importante. Por esta razón debemos estimar esta nueva *UT* de las recompensas y ajustar este segundo *sprint* en función de nuestra capacidad. Finalmente se decidió quitar la funcionalidad de perfiles privados y añadir la de recompensas.

Esta decisión se debe a que la funcionalidad de perfiles privados no es innovadora y no aporta nada diferente que nos haga destacar frente a la competencia. Es una funcionalidad que seguro implementaremos en el futuro, ya que se trata de una característica básica en una red social, pero que para validar nuestra idea de negocio en un contexto siguiendo el método *Lean Startup* no resulta de gran interés.

El tablero *kanban* quedó como se muestra en la Figura 47, a fecha 27/05/2024, justo antes de comenzar el segundo *Sprint*.

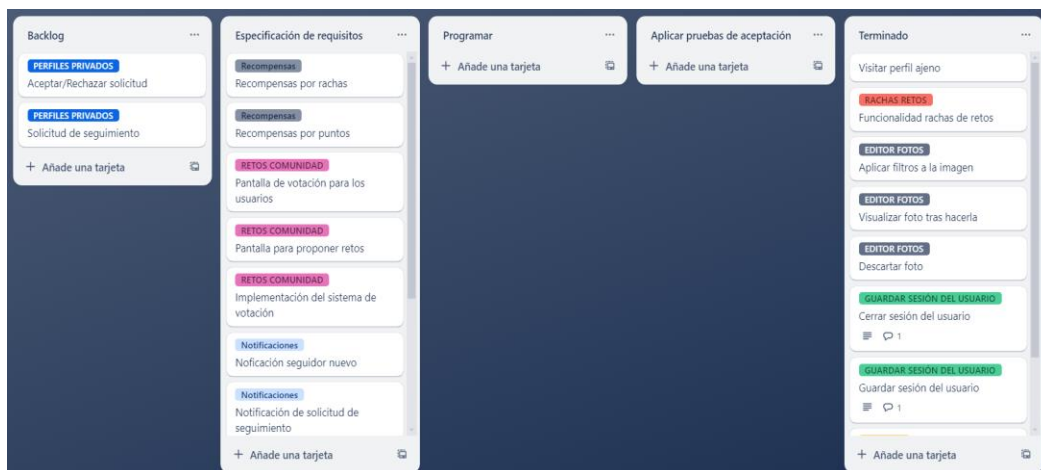


Figura 47: Tablero Kanban Trello segundo *sprint*

### Final del segundo *sprint*: 16/06/2024

Exactamente igual que en la finalización del primer *sprint*, se llevó a cabo una reunión de retrospectiva para analizar como fue el desarrollo de este último *sprint*. Este *sprint* se desarrolló sin ningún problema y se implementaron todas las *UT* previstas, quedando únicamente la realización del segundo experimento.

Durante la retrospectiva, el equipo destacó varios aspectos positivos, como la mejora en la comunicación y colaboración, que contribuyeron a la fluidez del desarrollo. También se identificaron áreas de mejora, como la necesidad de optimizar aún más la aplicación para aumentar su rendimiento.

Como conclusión tenemos la Figura 48 dónde se puede ver la cronología que se ha seguido durante el desarrollo del proyecto, desde que se finalizó el primer MVP y se realizó el experimento de la aplicación desarrollada en la asignatura de PIN el 23/04/2024, hasta la realización del segundo experimento el 17/06/2024 en el que se validó el segundo MVP realizado entre las dos fechas anteriormente mencionadas.



Figura 48: Cronología del proyecto

## 5.4 Segundo experimento

Este segundo experimento se ha realizado principalmente para valorar las nuevas funcionalidades desarrolladas durante el segundo *MVP*, para así conocer la opinión de los *Early Adopters* sobre UDARE, y darnos información sobre que funcionalidades de la aplicación deberíamos de mejorar y que otras podemos utilizar para explotar comercialmente la aplicación.

Para realizar este experimento, se ha optado por descargar un *APK* (*Android Package*<sup>57</sup>) para así poder aumentar el número de personas con el que hacer el experimento concurrentemente, y darles más tiempo para utilizar la aplicación. Esto ha sido posible ya que hemos desplegado el *back-end* en una máquina virtual de *AWS*, por lo que los usuarios han podido utilizar la aplicación desde su propia red. A diferencia del primer experimento, en este segundo se han buscado 30 *Early Adopters*, aumentando el rango de edad hasta los 50 años.

Tras una semana utilizando la aplicación por su cuenta, se decidió hacer una reunión con los máximos *Early Adopters* posibles para que nos dieran *feedback* rellenando una encuesta del mismo estilo que la del primer experimento, y nos contarán un poco sobre su experiencia usando UDARE.

En dicho cuestionario, hemos decidido realizar preguntas más específicas sobre las nuevas funcionalidades desarrolladas, y sobre funcionalidades futuras para la aplicación, que se deberían de desarrollar antes de publicar la aplicación, y por lo tanto habría que planificarlas para un tercer *MVP*.

Para la primera parte del cuestionario, como hemos dicho anteriormente, se han hecho ocho preguntas sobre las nuevas funcionalidades desarrolladas durante este segundo *MVP*.

Como se pueden ver en la Figura 49, las dos primeras preguntas tratan sobre la opinión de la funcionalidad del ranking, y que tipos de ranking se añadirían. La mayoría de las respuestas de la primera pregunta se sitúan entre el 4 y el 5, representando el 40% y el 26,6% respectivamente. En la segunda cuestión de pregunta abierta, hemos recibido 12 opiniones diferentes, ya que las de este tipo no son de respuesta obligatoria, dónde la gran parte piensa que sería interesante poder hacer grupos personalizados y así poder competir con diferentes entornos.

---

<sup>57</sup> *APK* (*Android Package*): Formato de archivo utilizado para distribuir e instalar aplicaciones en dispositivos Android.



¿Qué te ha parecido el ranking? \*

Muy malo    1    2    3    4    5    Muy bueno

○    ○    ○    ○    ○

¿Añadirías algún tipo de ranking más, aparte del global y el de amigos?

Texto de respuesta larga

Figura 49: Preguntas sobre el ranking

En la Figura 50 podemos observar las dos siguientes preguntas, relacionadas con el editor de fotos. Al igual de la primera pregunta, la primera trata sobre la opinión de la funcionalidad, donde un 30% han votado el 2 y un 35% el 3. En la segunda pregunta, 25 personas han contestado, siendo la mayoría de los comentarios relacionados con añadir efectos, *stickers* de todo tipo y poder añadir texto sobre las imágenes. Esto nos hace pensar que la funcionalidad tiene mucho margen de mejora, y que se debería plantear aumentar su funcionalidad en un futuro.

¿Qué opinión tienes sobre el editor de fotos? \*

Muy malo    1    2    3    4    5    Muy bueno

○    ○    ○    ○    ○

Añadirías alguna funcionalidad más al editor de fotos aparte de los filtros?

Texto de respuesta larga

Figura 50: Preguntas sobre el editor de fotos

A continuación, tenemos dos preguntas relacionadas con las rachas de retos, que se pueden ver en la Figura 51. La mayoría de las respuestas se encuentran entre el 3, con un 43,3% y el 4, con un 23,3%, cosa que nos dice que ha gustado, pero que deberíamos de mejorar algunos aspectos y perfeccionarla un poco más. En la pregunta de respuesta abierta ha habido 13 comentarios, de los cuales la mayoría sugiere que al entrar a la aplicación te informe de cuál es el estado actual de tu racha.

¿Te han gustado las rachas de retos? \*

1 2 3 4 5

Muy poco      Mucho

¿Cambiarías algo relacionado con las rachas, ya sea los premios, o la forma en la que muestra la racha?

Texto de respuesta larga

Figura 51: Preguntas sobre las rachas de retos

Finalizando las preguntas sobre las nuevas funcionalidades implementadas tenemos las expuestas en la Figura 52, relacionadas con los retos de la comunidad y su sistema de votación. Un 50% ha votado un 4, es decir, que les parece una buena funcionalidad, y un 26,6% un 5. Esto nos dice que es una funcionalidad que interesa a los usuarios, y que les ha gustado la forma en la que se ha implementado, ya que solo hemos recibido 7 comentarios sobre cambios en el sistema de votación y propuesta de retos. Entre estos, se sugiere que cuando se proponga un reto, sugiera otros parecidos propuestos por otros usuarios, y así poder votarlos de alguna manera para que aparezca en la pantalla de votación.

⋮

¿Qué piensas sobre los retos de la comunidad? \*

1 2 3 4 5

Muy malo      Muy bueno

¿Cambiarías la forma de votación y propuesta de retos de comunidad?

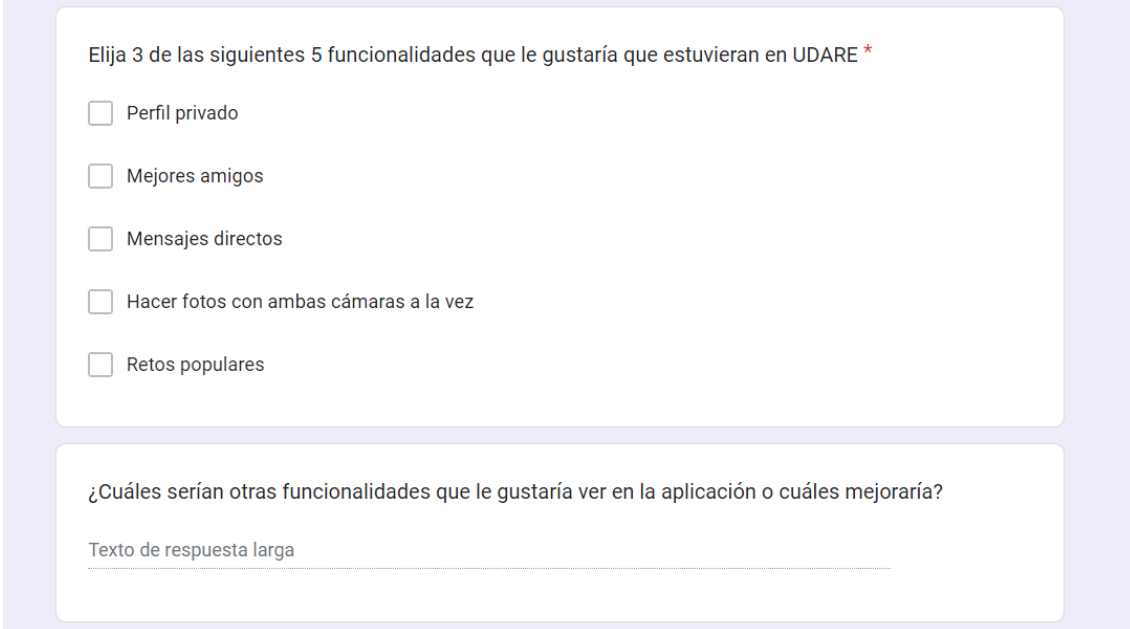
Texto de respuesta larga

Figura 52: Preguntas sobre los retos de comunidad

Tras las preguntas sobre las funcionalidades, encontramos dos preguntas enfocadas a las posibles nuevas funcionalidades que se podrían implementar. Hemos seleccionado por prioridad las siguientes 5 funcionalidades que no están implementadas y hemos preguntado cuales serían las más atractivas. Como se observa en la Figura 54, hemos hecho dos preguntas relacionadas con este tema, una primera de selección, dónde se deberán elegir las tres opciones que prefieran, y una segunda de respuesta abierta, para sugerir nuevas funcionalidades que no se encuentren en la lista, o modificaciones de algunas ya existentes.

En cuanto a la primera pregunta que se puede ver en la Figura 53, hemos obtenido un ranking siendo, por orden de la más votada a la menos, con 25 votos los perfiles

privados, con 23 hacer fotos con ambas cámaras a la vez, con 18 mejores amigos, con 13 retos populares y con 11 mensajes directos. La última pregunta se realizó con el objetivo de obtener ideas nuevas para el proyecto de fuera del equipo, ya que creemos que siempre es bueno escuchar opiniones ajenas, y nutrirse de opiniones diferentes.



Elija 3 de las siguientes 5 funcionalidades que le gustaría que estuvieran en UDARE \*

- Perfil privado
- Mejores amigos
- Mensajes directos
- Hacer fotos con ambas cámaras a la vez
- Retos populares

¿Cuáles serían otras funcionalidades que le gustaría ver en la aplicación o cuáles mejoraría?

Texto de respuesta larga

Figura 53: Preguntas sobre nuevas funcionalidades

## 6. CONCLUSIONES Y TRABAJO FUTURO

---

### 6.1 Conclusiones

Es emocionante ver el proceso de creación desde 0, hasta la formación final de una aplicación funcional que puede ser utilizada por cualquiera. Ha sido un proyecto que nos ha acompañado durante todo el último curso del grado y que ha ido evolucionando paralelamente a nosotros. Ver código de nuestros primeros desarrollos en *Kotlin*, y compararlo con el actual es algo gratificante dado que es visible la mejoría tanto en la lógica, como en el estilo.

Los objetivos por cumplir, expuestos en el principio de esta memoria podemos decir que se han cumplido ya que hemos sido capaces de desarrollar el *front-end* de una nueva aplicación *Android*, esta aplicación se ha desplegado en un entorno de producción, ya que hemos podido compartir un *APK* para que cada *Early Adopter* tuviese UDARE en sus propios móviles y además hemos adaptado el segundo MVP a las opiniones de los participantes en el primer experimento, siendo capaces de desarrollar un segundo MVP con todas las funcionalidades requeridas.

La realización de este TFG no habría sido posible sin los conocimientos adquiridos durante la carrera, por eso cabe mencionar las asignaturas de PSW (Proceso de Software) y PIN (Proyecto de Ingeniería de Software) las cuales nos han motivado a realizar un proyecto de emprendimiento, especialmente PIN, dónde empezamos esta aventura con otros 5 compañeros, siendo 3 informáticos más y 2 estudiantes de Bellas Artes. También hemos adquirido conocimientos de AVD (Análisis y validación de Software) para desarrollar test de calidad y ISW (Ingeniería del Software) y DDS (Diseño de Software) de dónde obtuvimos el conocimiento para asentar las bases y la estructura del proyecto.

Este proyecto ha sido de gran ayuda para crecer como informáticos, ya que hemos aprendido a programar aplicaciones móviles con *Android Studio* y en el lenguaje de programación de *Kotlin* desde 0, por lo que hemos adquirido experiencia en adaptarnos a nuevos entornos y lenguajes de programación sin tener conocimientos previos. Además, nos ha acercado un poco más a proyectos reales del mundo laboral, dónde se emplean los mismos métodos aplicados en este TFG.

Personalmente, ha sido un gran aprendizaje tanto técnicamente como en otras competencias como trabajar en grupo aplicando metodologías ágiles para gestionar el trabajo. Nos hemos dado cuenta de que el compromiso y las ganas de hacer un buen proyecto son las bases de cualquier trabajo.

De UDARE solo podemos sacar conclusiones positivas porque nos ha ayudado mucho en nuestro progreso como estudiantes y como ingenieros informáticos, así que nos lo llevamos como una bonita experiencia.

## 6.2 Trabajo futuro

Como proyecto de emprendimiento, el objetivo siempre ha de ser llegar al máximo número de personas y tener un alto número de descargas, pero a nuestro parecer hemos de ponernos objetivos alcanzables a corto/medio plazo e ir aumentando la dificultad de estos una vez se cumplen.

A corto plazo, deberíamos planificar un nuevo *MVP* dónde desarrollemos nuevas funcionalidades una vez analizada más a fondo la información obtenida en el segundo experimento, para así ofrecer una aplicación competitiva y que pueda hacerse un hueco en el mercado de las redes sociales.

Mirando un poco más lejos, nos gustaría poder publicar la aplicación en Google Play y planificar la migración a iOS, para así poder aumentar el número de posibles clientes. Para esto, se requiere una inversión en hardware, dado que para programar en el entorno de programación XCode necesitaremos un Mac.

## REFERENCIAS

- [1] American Psychological Association, «Teens are spending nearly 5 hours daily on social media. Here are the mental health outcomes,» [En línea]. Available: <https://www.apa.org/monitor/2024/04/teen-social-use-mental-health>. [Último acceso: Febrero 2024].
- [2] BMC Psychology, «Association between problematic social networking use and anxiety symptoms: a systematic review and meta-analysis,» [En línea]. Available: <https://doi.org/10.1186/s40359-024-01705-w>. [Último acceso: 04 2024].
- [3] BMC Public Health, «Social media use and social connectedness among adolescents in the United Kingdom: a qualitative exploration of displacement and stimulation,» [En línea]. Available: <https://doi.org/10.1186/s12889-021-11802-9>. [Último acceso: Febrero 2024].
- [4] Science in the News, Harvard University, «Dopamine, Smartphones & You: A battle for your time,» [En línea]. Available: <https://sitn.hms.harvard.edu/flash/2018/dopamine-smartphones-battle-time/>. [Último acceso: 04 2024].
- [5] Pepperdine Graziadio Business School, «Best Practices for Successful SWOT Analysis,» [En línea]. Available: <https://extension.psu.edu/conducting-a-swot-analysis>. [Último acceso: 04 2024].
- [6] A. Osterwalder y Y. Pigneur, *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*, Wiley, 2010.
- [7] Google, «Precios Google Ads,» [En línea]. Available: <https://adsense.google.com/intl/es-419/start/>. [Último acceso: Febrero 2024].
- [8] MongoDB, «Precios servicios MongoDB,» [En línea]. Available: <https://www.mongodb.com/es/pricing>. [Último acceso: Febrero 2024].
- [9] Amazon, «Precios servicio AWS EC2,» [En línea]. Available: <https://calculator.aws/#/createCalculator/ec2-enhancement>. [Último acceso: Febrero 2024].
- [10] Amazon, «Precios servicio AWS S3,» [En línea]. Available: <https://calculator.aws/#/createCalculator/S3>. [Último acceso: Febrero 2024].
- [11] Google, «Precios servicio Firebase,» [En línea]. Available: <https://firebase.google.com/pricing?hl=es&authuser=0#blaze-calculator>. [Último acceso: Febrero 2024].
- [12] Amazon, «Precios servicio AWS APIGATEWAY,» [En línea]. Available: <https://calculator.aws/#/createCalculator/APIGateway>. [Último acceso: Febrero 2024].
- [13] E. Ries, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*, Crown Currency, 2011.



- [14] P. Letelier, «¿Qué tienes en tu Backlog (y en tus Sprints)? Agility at work.» [En línea]. Available: <https://agilismoatwork.blogspot.com/>. [Último acceso: Junio 2024].
- [15] R. C. Martin, «Clean Architecture: A Craftsman's Guide to Software Structure and Design.» [En línea]. Available: <https://www.oreilly.com/library/view/clean-architecture-a/9780134494272/>. [Último acceso: Junio 2024].
- [16] R. Guru, «Patrones de diseño Singleton.» [En línea]. Available: <https://refactoring.guru/es/design-patterns/singleton>. [Último acceso: Junio 2024].
- [17] R. Guru, «Patrones de diseño Adapter.» [En línea]. Available: <https://refactoring.guru/es/design-patterns/adapter>. [Último acceso: Junio 2024].
- [18] Investopedia, «What Is an Early Adopter, and How Does It Work?.» [En línea]. Available: <https://www.investopedia.com/terms/e/early-adopter.asp>. [Último acceso: Mayo 2024].
- [19] D. J. Anderson, Kanban: Successful Evolutionary Change for Your Technology Business, Blue Book ed., 2010.
- [20] K. Schwaber y J. Sutherland, «The Scrum Guide.» 2017. [En línea]. Available: <https://www.scrum.org/scrum-guide-2020>. [Último acceso: Abril 2024].

# ANEXO

---

## Anexo A

A continuación, se va a mostrar cómo se usa la aplicación y sus distintas funcionalidades.

Para empezar, cuando se abre la aplicación se muestra una pantalla de carga, como la primera pantalla de la Figura 54, con el logo de fondo y una barra que indica el proceso de carga. Cuando la barra llega a su fin, se cambia de pantalla y se muestra una *landing page*, con el lema de la aplicación: “Te atreves?”, y dos botones, uno para iniciar sesión y otro para registrar un nuevo usuario, que se puede ver en la segunda pantalla de la figura anteriormente mencionada.

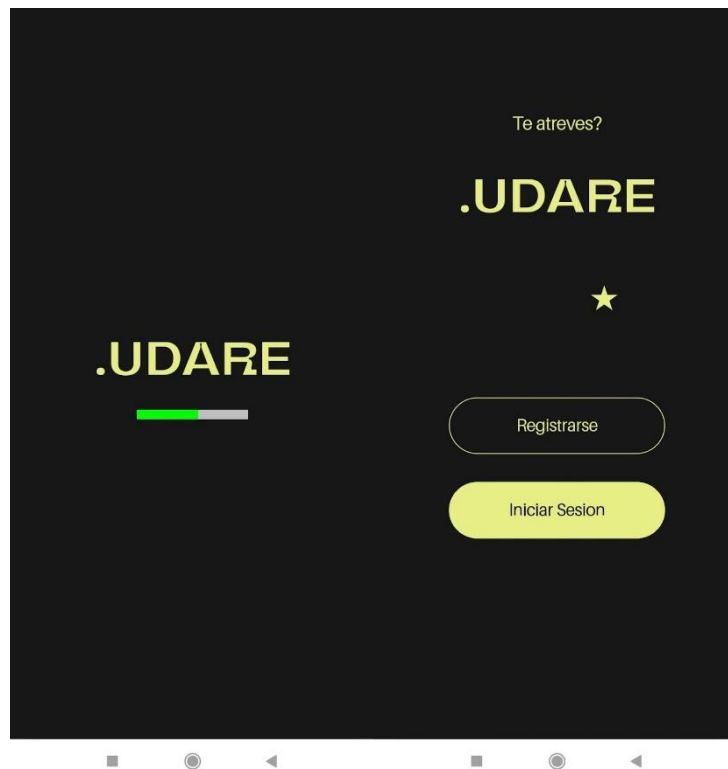


Figura 54: Pantallas de carga y landing page

Como se ha mencionado anteriormente, en la *landing page* disponemos de dos botones que aparecen en la primera pantalla de la figura Figura 55. Si pulsamos el botón de iniciar sesión aparecerá la pantalla de inicio de sesión donde nos pide las credenciales, que constan del correo y la contraseña del usuario, tal y como se muestra en la Figura 59. Desde aquí, se dispone de un acceso directo al formulario de registro, y de un texto que genera un *popup* cuando es clicado, pudiendo introducir el correo del usuario para el cambio de contraseña. En cambio, en la segunda pantalla de la Figura 55, vemos que, si pulsamos en registrarse, aparecerá un formulario donde introducir los datos necesarios para crear un usuario, además de tener acceso directo a la pantalla de inicio de sesión.



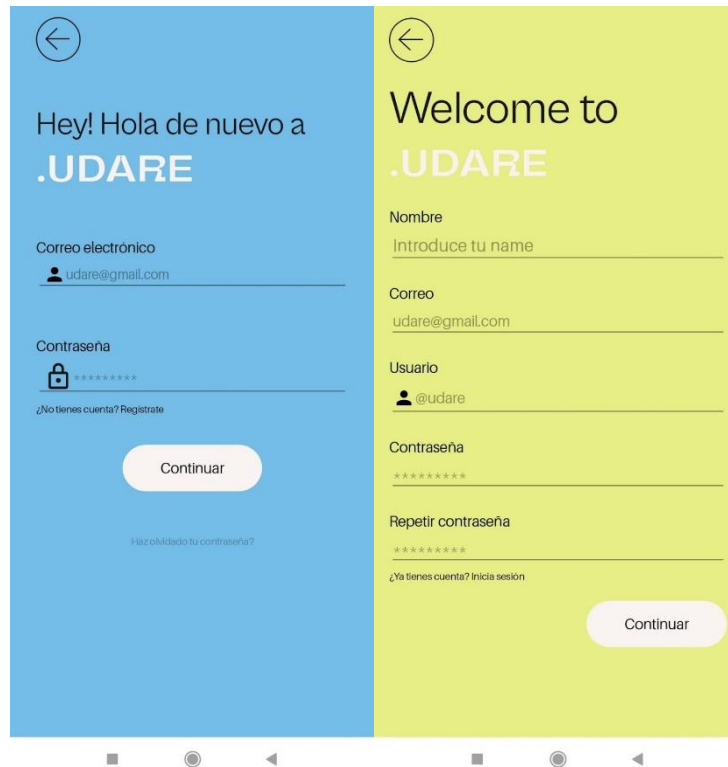


Figura 55: Pantallas de inicio de sesión y registro

Tras introducir unas credenciales correctas, o registrar un usuario, pasaremos al *feed* principal, dónde se muestran los diferentes posts que han subido las personas a las que sigues, además de accesos a diferentes funcionalidades. Como podemos ver en la Figura 56, en la parte superior derecha se muestra la foto de perfil del usuario que ha iniciado sesión, por lo que clicando en ella podremos acceder al perfil. En la misma barra superior, pero a la izquierda, tenemos un icono de añadir amigos, dónde si pulsamos en él, nos llevará a la pantalla de sugerencias de amigos.

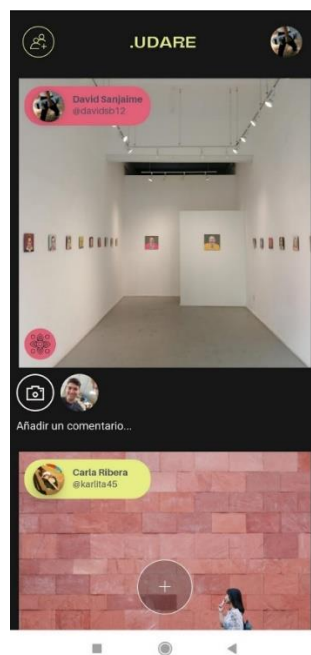


Figura 56: Feed principal

Desde el feed principal también podremos añadir comentarios y reacciones en los *posts* a través del símbolo de la foto debajo del *post*, para así abrir la cámara cómo en la primera pantalla de la Figura 57, o clicando en el texto de “Añadir comentario...”, dónde se nos abriría una pantalla como la de la segunda de la Figura 57. Otra de las funcionalidades a las que nos permite acceder esta pantalla es la de visitar perfiles ajenos. Pulsando en el recuadro dónde aparece la foto y el nombre de usuario del autor del *post*, nos llevará a su perfil. Finalmente, pero no menos importante, podremos acceder al seleccionador de retos pulsando en el círculo con un más en su interior, que se encuentra debajo de la pantalla.

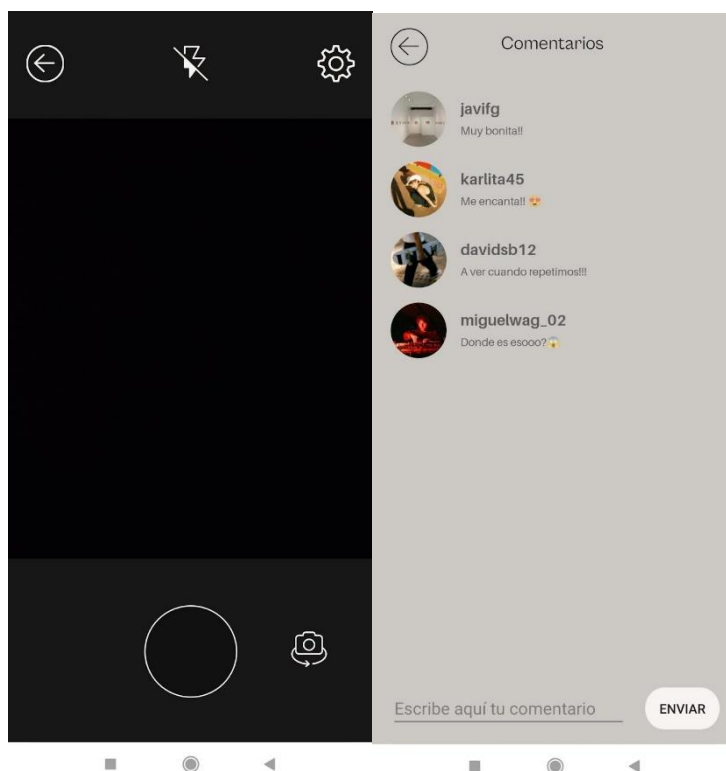


Figura 57: Cámara y pantalla comentarios

Expuestas todas las funcionalidades a las que se pueden acceder desde la pantalla de inicio, vamos a ir mostrando tanto que se puede hacer en ellas, cómo cuál es su interfaz. Siguiendo el mismo orden con el que las hemos expuesto tenemos el perfil que aparece en la Figura 58. En este, se mostrará la foto de perfil del usuario, desde dónde podremos cambiarla pulsando el icono con un lápiz, el nombre de usuario, el nombre, el número de días seguidos que lleva completando retos, los puntos totales y los puntos desglosados por categoría.



Figura 58: Pantalla del perfil

Haciendo *scroll* hasta el final, también encontraremos un texto, como se puede ver en la parte inferior de la primera pantalla de la Figura 59, dónde pulsando sobre él podremos cerrar sesión. Los perfiles ajenos tendrán la misma estructura que el perfil, como podemos ver en la segunda pantalla de la Figura 59, ocultando el botón para cambiar la foto de perfil, el de cerrar sesión, y los botones para acceder a los trofeos y al ranking.



Figura 59: Cerrar sesión y perfil ajeno

Desde esta pantalla también se accede a varias funcionalidades a través de los botones que se encuentran a la derecha de la foto de perfil. El botón superior nos muestra un álbum como el de la Figura 60, con todas las fotos que ha subido el usuario, y dónde se podrán filtrar por el último mes, semana, o por categorías.

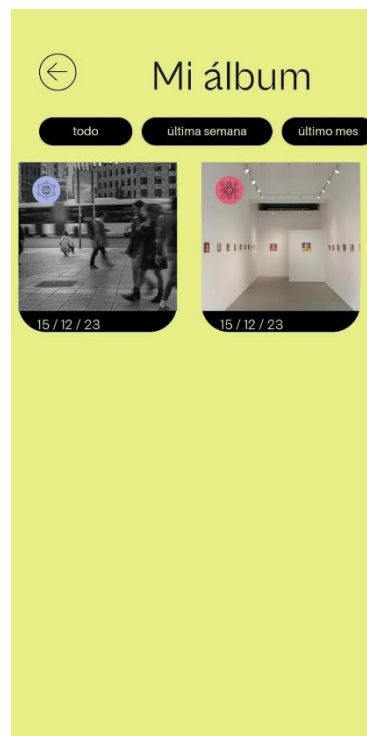


Figura 60: Pantalla álbum

El botón con un trofeo de la pantalla del perfil muestra un ranking, como el de la Figura 61, basado en los puntos totales. Este ranking también se puede filtrar por todos los usuarios de la aplicación, o por la gente que sigues pulsando el botón que se encuentra arriba a la derecha.

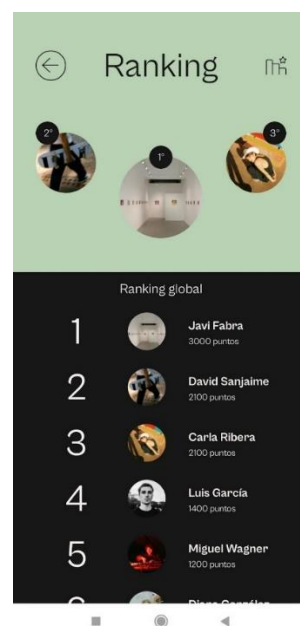


Figura 61: Pantalla ranking

El último botón de la columna nos mostrará los trofeos conseguidos por el usuario como se puede ver en la Figura 62. Estos trofeos están separados por los obtenidos mediante rachas y por puntos.

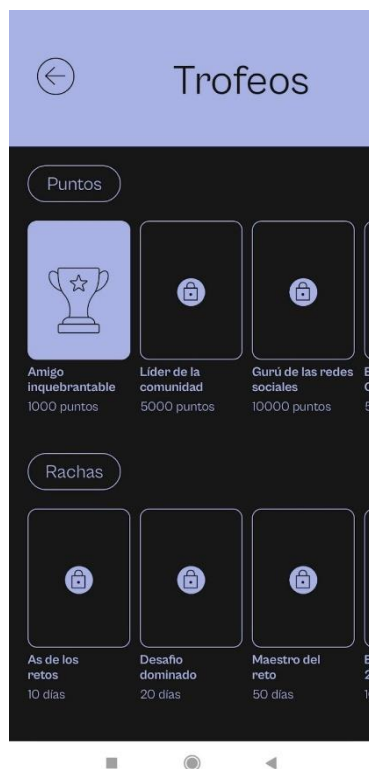


Figura 62: Pantalla trofeos

Siguiendo las funcionalidades a las cuáles se pueden acceder desde la pantalla de inicio, tenemos la pantalla de sugerencias, seguidores y seguidos. Esta pantalla dispone de 3 botones y una lista de usuarios, dónde dependiendo de qué botón tenga el fondo blanco, se mostrará una lista u otra.

En las sugerencias de la Figura 63, nos mostrará una lista de usuarios a los que tus seguidos siguen, pero tú no, además de que se puede buscar cualquier usuario registrado. Para los botones de seguidores y seguidos de las Figuras 64 y 65, mostrará cómo sus títulos indican, los seguidores y seguidos por el usuario. Desde esta pantalla también se podrá acceder al perfil propio clicando en la foto de perfil situada arriba a la derecha, y a los perfiles ajenos clicando sobre su foto de perfil.

Como se puede ver en la Figura 63, desde la pantalla de sugerencias tendremos un icono de más, dónde pulsando sobre él se añadirá ese usuario a la lista de seguidos. Lo mismo para la lista de seguidos, pero con el icono de menos, ya que si pulsamos sobre él dejaremos de seguir a ese usuario. Esto se podría rectificar en caso de pulsar sin querer ya que, tras clicar sobre el botón, el icono cambia al contrario y el usuario no desaparece hasta que no cambiamos de pantalla.

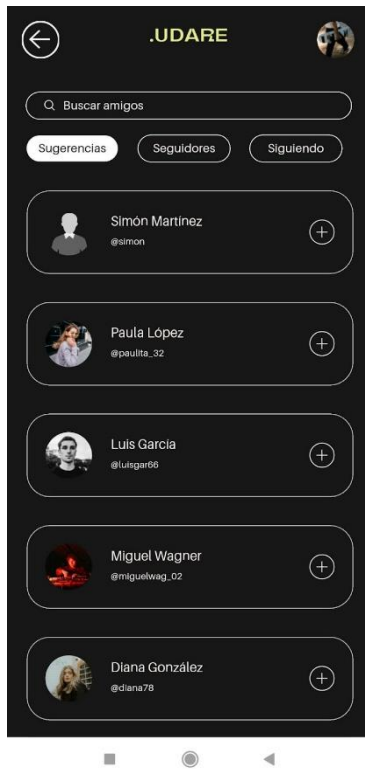


Figura 63: Pantalla sugerencias

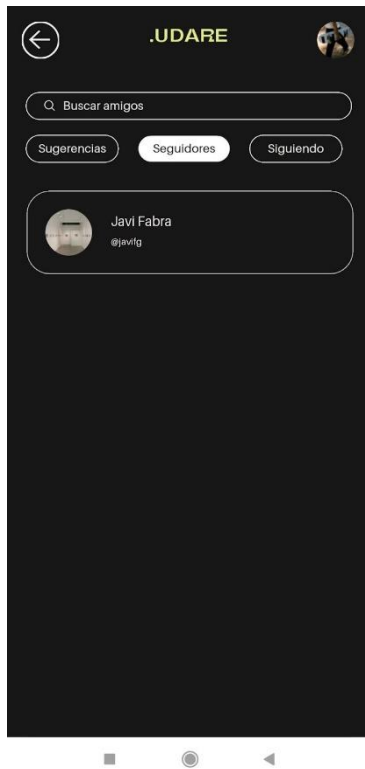


Figura 64: Pantalla seguidores

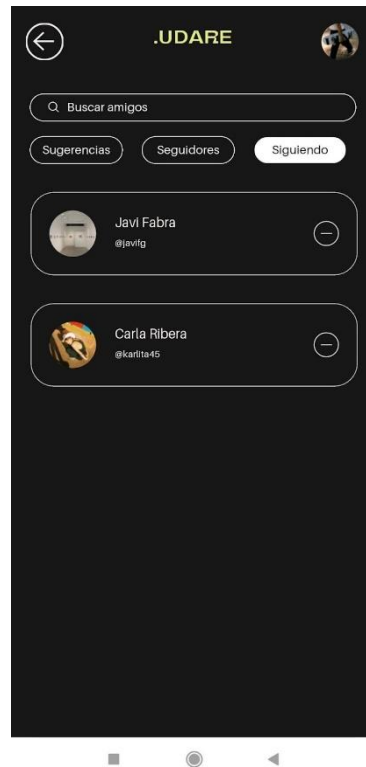


Figura 65: Pantalla siguiendo

Finalmente, y ya acabando todas las funcionalidades accesibles desde la pantalla de inicio tenemos la pantalla que muestra la Figura 66 dónde se puede seleccionar el reto que vamos a completar. En esta pantalla se propondrán cinco retos de cinco categorías distintas, más un reto patrocinado.



Figura 66: Pantalla seleccionar reto

Tras seleccionar uno de los retos, se nos volverá a mostrar una pantalla como la primera de la Figura 67 dónde tomaremos la foto que subiremos como *post*. Una vez hecha la foto, podremos aplicar los diferentes filtros disponibles, y decidir si subimos esa foto pulsando en el botón de subir, o repetirla pulsando la x de arriba a la izquierda, y confirmando el *popup* que se nos muestra. Una vez que pulsamos en el botón para subir la foto, aparecerá una pantalla que aparece en la segunda imagen de la Figura 67 felicitándonos por completar el reto del día.

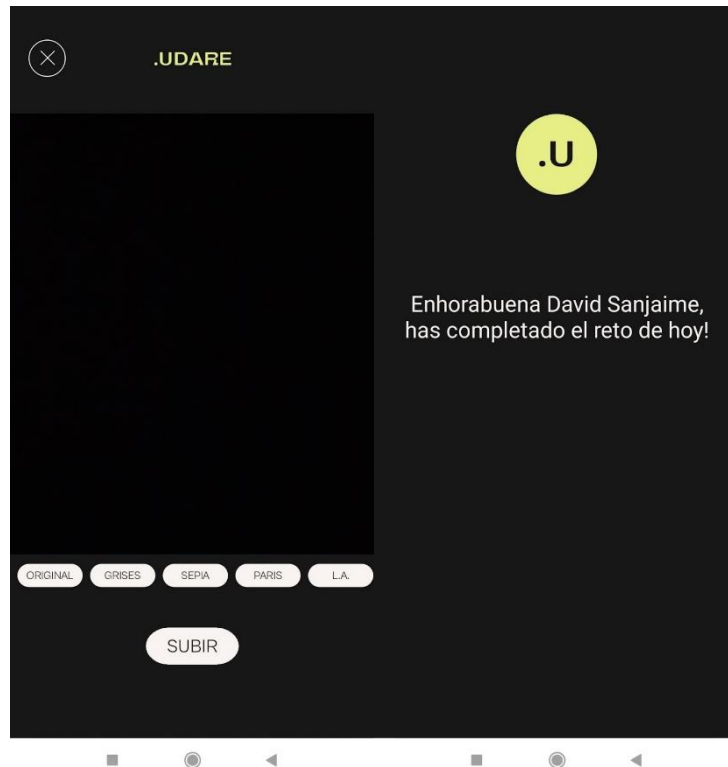


Figura 67: Pantalla filtros y confirmación



## Anexo B

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

<b>Objetivos de Desarrollo Sostenibles</b>	<b>Alto</b>	<b>Medio</b>	<b>Bajo</b>	<b>No Procede</b>
ODS 1. <b>Fin de la pobreza.</b>				X
ODS 2. <b>Hambre cero.</b>				X
ODS 3. <b>Salud y bienestar.</b>	X			
ODS 4. <b>Educación de calidad.</b>			X	
ODS 5. <b>Igualdad de género.</b>				X
ODS 6. <b>Agua limpia y saneamiento.</b>				X
ODS 7. <b>Energía asequible y no contaminante.</b>				X
ODS 8. <b>Trabajo decente y crecimiento económico.</b>				X
ODS 9. <b>Industria, innovación e infraestructuras.</b>				X
ODS 10. <b>Reducción de las desigualdades.</b>				X
ODS 11. <b>Ciudades y comunidades sostenibles.</b>				X
ODS 12. <b>Producción y consumo responsables.</b>				X
ODS 13. <b>Acción por el clima.</b>				X
ODS 14. <b>Vida submarina.</b>				X

ODS 15. <b>Vida de ecosistemas terrestres.</b>				X
ODS 16. <b>Paz, justicia e instituciones sólidas.</b>				X
ODS 17. <b>Alianzas para lograr objetivos.</b>				X

*Tabla 5: Relación de trabajo con los ODS*

El Objetivo de Desarrollo Sostenible más acorde con nuestra aplicación es claramente el número 3: Salud y bienestar, como podemos observar en la Tabla 5. Esto se debe a que se motiva e influencia a los usuarios a mantener un estilo de vida más saludable y a salir de la zona de confort para realizar actividades fuera de la rutina mediante los retos diarios propuestos. La intención de UDARE es animar a la gente a realizar nuevas actividades en ámbitos diferentes cada día, y así mejorar su calidad de vida.

También podemos relacionar el ODS 4, pero en un grado bajo porque algunos de los retos animan a culturizarse ya sea mediante la visita a museos, la lectura de libros o simplemente conociendo nuevos deportes o personas de otras culturas.