



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escola Tècnica  
Superior d'Enginyeria  
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica  
Universitat Politècnica de València

# Aplicación móvil para la generación de rutas turísticas basadas en las preferencias del usuario

Trabajo Fin de Grado

**Grado en Ingeniería Informática**

**Autor:** Natalia Alcarria Romero

**Tutor:** David de Andrés Martínez

2019-2020



## Resumen

---

Este proyecto se centra en el desarrollo de una aplicación Android para la creación de rutas turísticas basadas en las preferencias de los usuarios.

La aplicación está orientada principalmente a usuarios que quieran viajar y descubrir nuevos lugares. Para ello, desde la aplicación pueden generar rutas automáticamente en función a sus intereses, aunque también es posible editar las rutas manualmente, añadiendo o eliminando lugares. Se ofrece la posibilidad de guardar estas rutas para poder acceder a ellas posteriormente y editarlas o volver a realizarlas.

También es posible utilizar la aplicación simplemente para encontrar un lugar concreto gracias a la vista de mapa en la que se muestran los lugares con marcadores personalizados según los intereses del usuario. Además, es posible añadir los lugares a una lista de favoritos para poder acceder a ellos fácilmente en cualquier momento.

Otra de las funcionalidades que ofrece es poder descubrir nuevos lugares que, aun estando fuera de las preferencias del usuario, pueden llegar a interesarle.

También cuenta con un sistema de votación de lugares, es decir, el usuario puede valorar los lugares. Los lugares mejor votados aparecen destacados en el mapa para todos los usuarios, independientemente de si pertenece a sus intereses o no.

Para el desarrollo se ha utilizado el lenguaje de programación Kotlin, así como el lenguaje de marcado XML para las interfaces de usuario. Para el *backend* se ha utilizado los servicios proporcionados por Firebase, tanto para la autenticación de los usuarios, como para almacenar y obtener datos en tiempo real.

**Palabras clave:** Android, Kotlin, turismo, Firebase.

## Resum

---

Aquest projecte se centra en el desenvolupament d'una aplicació Android per a la creació de rutes turístiques basades en les preferències dels usuaris.

L'aplicació està orientada principalment a usuaris que vulguen viatjar i descobrir nous llocs. Per a això, des de l'aplicació poden generar rutes automàticament en funció als seus interessos, encara que també és possible editar les rutes manualment, afegint o eliminant llocs. S'ofereix la possibilitat de guardar aquestes rutes per a poder accedir a elles posteriorment i editar-les o tornar a realitzar-les.

També és possible utilitzar l'aplicació simplement per a trobar un lloc concret gràcies a la vista de mapa en la qual es mostren els llocs amb marcadors personalitzats segons els interessos de l'usuari. A més, és possible afegir els llocs a una llista de favorits per a poder accedir a ells fàcilment en qualsevol moment.

Una altra de les funcionalitats que ofereix és poder descobrir nous llocs que, fins i tot estant fora de les preferències de l'usuari, poden arribar a interessar-li.



També compta amb un sistema de votació de llocs, és a dir, l'usuari pot valorar els llocs. Els llocs més ben votats apareixen destacats en el mapa per a tots els usuaris, independentment de si pertany als seus interessos o no.

Per al desenvolupament s'ha utilitzat el llenguatge de programació *Kotlin*, així com el llenguatge de marcat XML per a les interfícies d'usuari. Per al *backend* s'ha utilitzat els serveis proporcionats per *Firebase*, tant per a l'autenticació dels usuaris, com per a emmagatzemar i obtenir dades en temps real.

**Paraules clau:** Android, Kotlin, turisme, Firebase.

---

## Abstract

This project focuses on the development of an Android application for creating tourist routes based on user preferences.

The application is mainly aimed at users who wants to travel and discover new places. To do this, from the app they can automatically generate routes based on their interests, although it is also possible edit the routes manually, adding or removing places. It offers the possibility of save these routes so that the user can access them later and edit or do it again.

It is also possible to use the application simply to find a specific place thanks to the map view that shows the places with custom markers according to the user's interests. In addition, it is possible to add the places to a list of favourites so the user can easily access them at any time.

Another feature that it offers is to be able to discover new places that, even if outside user's preferences, may become of interest to him.

It also has a system of place rating, it means that the user can rate the places. The best-voted places are showed on the map for all users, regardless of whether they belong to their interests or not.

The development has used the Kotlin programming language as well as the XML markup language for user interfaces. The backend has used the services provided by Firebase, both for authentication of users, and for storing and obtaining data in real time.

**Keywords:** Android, Kotlin, tourism, Firebase.

# Tabla de contenidos

---

1. Introducción.....	11
1.1 Motivación.....	11
1.2 Objetivos.....	12
1.3 Estructura .....	12
1.4 Metodología y planificación .....	13
1.4.1 Scrum.....	14
1.5 Relación con los estudios cursados .....	14
2. Contexto tecnológico .....	15
3. Especificación de requisitos .....	19
3.1 Diagrama de casos de uso.....	19
3.2 Descripción de los casos de uso .....	20
3.2.1 Actor no autenticado.....	20
3.2.2 Actor autenticado.....	21
3.3 Diagrama de clases .....	28
4. Diseño .....	29
4.1 Arquitectura .....	29
4.2 Modelo de datos .....	31
4.3 Prototipos de la interfaz .....	32
4.3.1 Inicio de sesión.....	32
4.3.2 Registro.....	33
4.3.3 Elección de intereses .....	34
4.3.4 Menú principal .....	35
4.3.4.1 Pestaña lugares .....	35
4.3.4.2 Pestaña descubrir .....	36
4.3.4.3 Pestaña mapa.....	37
4.3.4.4 Pestaña mis rutas .....	38
4.3.4.4 Pestaña perfil .....	39
4.4 Diagrama de navegación .....	41
4.5 Material Design .....	42
5. Implementación .....	43
5.1 Herramientas para el desarrollo .....	43
5.1.1 Android Studio .....	43
5.1.2 GitLab.....	43



5.1.3 InkScape .....	43
5.2 Lenguajes de programación.....	44
5.3 Android .....	44
5.3.1 Características .....	45
5.3.2 Arquitectura.....	45
5.4 Base de datos.....	46
5.4.1 Firebase .....	47
5.5 Estructura interna de la aplicación .....	48
5.5.1 Capa de persistencia.....	48
5.5.2 Capa de negocio .....	49
5.5.3 Capa de presentación.....	49
5.6 Patrones de diseño .....	49
5.7 Flujos .....	50
6. Resultados .....	55
6.1 Resultado de la interfaz de usuario .....	55
6.1.1 Inicio de sesión.....	55
6.1.2 Registro.....	57
6.1.3 Intereses.....	57
6.1.4 Menú .....	58
6.1.4.1 Mapa.....	58
6.1.4.2 Lugares.....	59
6.1.4.3 Descubrir .....	60
6.1.4.4 Mis rutas .....	61
6.1.4.5 Perfil.....	61
6.1.5 Detalles de lugar.....	62
6.1.6 Detalles de ruta .....	63
6.1.7 Mis lugares favoritos.....	63
6.1.8 Mis votos .....	64
6.1.9 Mis intereses .....	64
6.1.10 Cambiar contraseña .....	65
6.2 Diagrama de navegación .....	66
6.3 Pruebas .....	66
7. Conclusión.....	73
7.1 Trabajo futuro.....	73
8. Bibliografía .....	74
Anexo I.....	78



# Índice de figuras

---

Figura 1 - Captura de pantalla de la aplicación Batch Geocode .....	15
Figura 2 - Captura de pantalla de la aplicación Calimoto .....	16
Figura 3 - Captura de pantalla aplicación My phone tour .....	16
Figura 4 - Captura de pantalla de la aplicación MyRoute-app.....	17
Figura 5 - Captura de pantalla de la aplicación Spain Travel Guide.....	17
Figura 6 - Diagrama de casos de uso usuario autenticado. ....	19
Figura 7 - Diagrama de casos de uso usuario no autenticado. ....	20
Figura 8 - Diagrama de clases.....	28
Figura 9 - Arquitectura de Firebase .....	29
Figura 10 - Modelo de datos.....	31
Figura 11 - Prototipo inicio de sesión.....	32
Figura 12 - Prototipo registro I.....	33
Figura 13 - Prototipo registro II .....	33
Figura 14 - Prototipo elegir intereses.....	34
Figura 15 - Prototipo pestaña “Lugares” .....	35
Figura 16 - Prototipo información de un lugar .....	35
Figura 17 - Prototipo pestaña "Descubrir" .....	36
Figura 18 - Prototipo información de un lugar .....	36
Figura 19 - Prototipo pestaña “Mapa” .....	37
Figura 20 - Prototipo pestaña “Mis rutas” .....	38
Figura 21 - Prototipo información de una ruta.....	38
Figura 22 - Prototipo pestaña “Perfil” .....	39
Figura 23 - Prototipo de menú desplegable .....	39
Figura 24 - Prototipo pantalla “Mis datos” .....	40
Figura 25 - Prototipo pantalla “Mis lugares favoritos” .....	40
Figura 26 - Prototipo pantalla “Mis intereses” .....	40
Figura 27 - Diagrama de secuencia prototipos .....	41
Figura 28 - Diagrama de secuencia iniciar sesión .....	51
Figura 29 - Diagrama de secuencia generar ruta.....	52
Figura 30 - Diagrama de secuencia modificar intereses .....	53
Figura 31 - Interfaz de inicio de sesión .....	55
Figura 32 - Interfaz recuperar contraseña.....	56
Figura 33 - Correo cambiar contraseña .....	56
Figura 34 - Cambio de contraseña .....	56
Figura 35 - Interfaz de registro I.....	57
Figura 36 - Interfaz de registro II.....	57
Figura 37 - Interfaz elegir intereses .....	57
Figura 38 - Interfaz pestaña “Mapa” .....	58
Figura 39 - Filtros pestaña “Mapa”.....	58
Figura 40 - Detalle de lugar en la pestaña “Mapa” .....	59
Figura 41 - Interfaz pestaña “Lugares”.....	59
Figura 42 - Interfaz pestaña “Descubrir” .....	60
Figura 43 - Interfaz pestaña “Mis rutas” .....	61
Figura 44 - Interfaz pestaña “Perfil” .....	61



Figura 45 - Menú lateral .....	61
Figura 46 - Interfaz detalles de lugar .....	62
Figura 47 - Interfaz detalle de ruta.....	63
Figura 48 - Interfaz pantalla “Mis lugares favoritos” .....	63
Figura 49 - Interfaz pantalla “Mis votos”.....	64
Figura 50 - Interfaz pantalla “Cambiar contraseña”.....	65
Figura 51 - Diagrama de navegación I.....	66
Figura 52 - Diagrama de navegación II.....	67
Figura 53 - Resultados cuestionario sección 3.1 .....	70
Figura 54 - Resultados cuestionario sección 3.2 .....	70
Figura 55 - Resultados cuestionario sección 3.3 .....	71
Figura 56 - Resultados cuestionario sección 3.4 .....	71
Figura 57 - Resultados cuestionario sección 3.5 .....	71
Figura 58 - Encuesta de usabilidad sección 1.....	78
Figura 59 - Encuesta de usabilidad sección 2.1.....	79
Figura 60 - Encuesta de usabilidad sección 2.2.....	80
Figura 61 - Encuesta de usabilidad sección 2.3.....	80
Figura 62 - Encuesta de usabilidad sección 3.1.....	81
Figura 63 - Encuesta de usabilidad sección 3.2.....	82

# Índice de tablas

---

Tabla 1 - Planificación .....	13
Tabla 2 - Comparativa aplicaciones significativas.....	18
Tabla 3 - Caso de uso registrarse - Descripción .....	20
Tabla 4 - Caso de uso registrarse - Interacción .....	20
Tabla 5 - Caso de uso registrarse - extensiones síncronas .....	20
Tabla 6 - Caso de uso elegir intereses - Descripción.....	20
Tabla 7 - Caso de uso elegir intereses - Interacción .....	21
Tabla 8 - Caso de uso iniciar sesión - Descripción .....	21
Tabla 9 - Caso de uso iniciar sesión - Interacción .....	21
Tabla 10 - Caso de uso iniciar sesión - Extensiones síncronas .....	21
Tabla 11 - Caso de uso listar lugares - Descripción.....	21
Tabla 12 - Caso de uso listar lugares - Interacciones .....	22
Tabla 13 - Caso de uso ver detalle de lugar - Descripción.....	22
Tabla 14 - Caso de uso ver detalle de lugar - Interacción.....	22
Tabla 15 - Caso de uso añadir lugar a favoritos - Descripción.....	22
Tabla 16 - Caso de uso añadir lugar a favoritos- Interacción .....	22
Tabla 17 - Caso de uso listar lugares favoritos - Descripción .....	22
Tabla 18 - Caso de uso listar lugares favoritos - Interacción.....	23
Tabla 19 - Caso de uso votar lugar - Descripción .....	23
Tabla 20 - Caso de uso votar lugar - Interacción .....	23
Tabla 21 - Caso de uso ver lugares votados - Descripción .....	23
Tabla 22 - Caso de uso ver lugares votados - Interacción .....	23
Tabla 23 - Caso de uso añadir lugar a ruta - Descripción .....	24
Tabla 24 - Caso de uso añadir lugar a ruta - Interacción .....	24
Tabla 25 - Caso de uso ver ruta - Descripción.....	24
Tabla 26 - Caso de uso ver ruta - Interacción .....	24
Tabla 27 - Caso de uso editar ruta - Descripción.....	24
Tabla 28 - Caso de uso editar ruta - Interacción .....	25
Tabla 29 - Caso de uso guardar ruta - Descripción.....	25
Tabla 30 - Caso de uso guardar ruta - Interacción.....	25
Tabla 31 - Caso de uso listar rutas realizadas - Descripción .....	25
Tabla 32 - Caso de uso listar rutas realizadas - Interacción.....	25
Tabla 33 - Caso de uso borrar ruta - Descripción .....	26
Tabla 34 - Caso de uso borrar ruta - Interacción .....	26
Tabla 35 - Caso de uso editar intereses - Descripción.....	26
Tabla 36 - Caso de uso editar intereses - Interacción.....	26
Tabla 37 - Caso de uso editar intereses - Extensiones síncronas.....	26
Tabla 38 - Caso de uso cambiar contraseña - Descripción.....	27
Tabla 39 - Caso de uso cambiar contraseña - Interacción .....	27
Tabla 40 - Caso de uso cambiar contraseña - Extensiones síncronas.....	27
Tabla 41 - Caso de uso cerrar sesión - Descripción .....	27
Tabla 42 - Caso de uso cerrar sesión - Interacción.....	27
Tabla 43 - Resultados .....	68
Tabla 44 - Cálculos de los resultados.....	69



# 1. Introducción

---

Este proyecto se centra en el desarrollo de una aplicación para dispositivos Android para la creación de rutas turísticas en base a los intereses del usuario.

El objetivo del proyecto es ayudar al turista a encontrar nuevos lugares que no suelen estar indicados en las guías turísticas, pero, sobre todo al turismo en lugares despoblados o aislados, en los que no existe la figura del guía turístico, permitiendo así encontrar lugares de interés, tanto a nivel nacional como internacional.

En la actualidad, los usuarios hacen uso de los dispositivos móviles para buscar cualquier tipo de lugar que se va a visitar, por lo que se planteó como una idea con la que los usuarios pudieran localizar estos lugares con mayor facilidad.

Con esta aplicación, los usuarios pueden buscar sitios de interés en cualquier lugar, en base a sus propios intereses. Además, si lo desean, pueden generar una ruta entre dichos lugares.

El usuario tiene la posibilidad de guardar sus rutas para volver a realizarlas, o añadir los lugares a favoritos para poder visitarlos en otro momento. La aplicación también permite valorar los lugares, haciéndolos más visibles a otros usuarios, ya que un lugar con una puntuación alta se convierte en un lugar destacado.

Previamente al desarrollo se ha realizado un estudio de las aplicaciones similares que ya existen, para tener en cuenta las necesidades que debe satisfacer la aplicación y que funcionalidad debería tener.

Para el desarrollo se ha elegido el IDE<sup>1</sup> Android Studio, en cuanto a la implementación se ha utilizado el lenguaje Kotlin, así como el lenguaje de marcado XML para el desarrollo de las interfaces de usuario.

## 1.1 Motivación

Con el aumento del uso de los *smartphones*, también ha aumentado la demanda de aplicaciones, los usuarios quieren disponer de todos los servicios en sus dispositivos portables.

El turismo es una de las principales actividades de ocio en la actualidad, cada vez hay más gente que viaja y que está interesada en conocer nuevos lugares.

La idea de este proyecto surge a raíz de unir los dos conceptos anteriores, ya que, si la gente cada vez viaja más, y quieren tener todos los servicios en sus dispositivos. Es necesaria una aplicación que les facilite la tarea de encontrar lugares que de verdad les interese visitar.

Por otro lado, este proyecto también está enfocado a potenciar el turismo en zonas rurales y aisladas para las que no es tan sencillo dar a conocer los puntos de interés que puedan tener. Con esta aplicación es mucho más sencillo encontrar lugares que interesen al usuario sin importar la zona donde se encuentren.

## 1.2 Objetivos

En este apartado se van a describir los objetivos que se pretenden alcanzar con el desarrollo de la aplicación.

La finalidad principal de este proyecto es proporcionar una herramienta que facilite al usuario su experiencia a la hora de realizar rutas turísticas, para ello, se han propuesto los siguientes objetivos:

- Facilitar la creación de rutas turísticas.
- Facilitar la búsqueda de lugares de interés, sea cual sea su ubicación.
- Descubrir nuevos lugares que puedan interesar al usuario.
- Promover la visita a los lugares más interesantes mediante el uso de un sistema de puntuación.

## 1.3 Estructura

Este TFG se compone de seis puntos. En primer lugar, se hace una breve descripción del proyecto, la motivación para su realización y los objetivos que debe cumplir la aplicación. Se explica la metodología que se ha seguido para el desarrollo y se detallan las actividades realizadas en cada iteración. Por último, se habla de la relación que tiene el proyecto con los conocimientos adquiridos en el grado.

En el segundo, se expone el resultado de un estudio realizado sobre aplicaciones similares ya existentes, comparando sus características para establecer una base de funcionalidades que debe tener la aplicación y ver cómo mejorar otras.

En tercer lugar, se definen los requisitos funcionales de la aplicación mediante el uso de diagramas de casos de uso y de clases.

En el cuarto punto se explica el diseño de la arquitectura y del modelo de datos, así como de la interfaz de usuario mediante imágenes de los prototipos iniciales.

En el quinto punto se explican las herramientas utilizadas para el desarrollo de la aplicación, también se explican con más detalle algunos de los flujos. Además, se explica la arquitectura interna de la aplicación

En el sexto punto, se muestra el diseño final de la aplicación y se realiza una explicación detallada de las funciones que ofrece cada pantalla de esta. También se exponen los resultados de las pruebas realizadas con usuarios finales tras terminar el desarrollo.

Para finalizar, se incluye un punto de conclusiones donde se hace una reflexión acerca de los problemas encontrados a la hora de realizar este proyecto y como se han resuelto, así como algunas funcionalidades que sería interesante implementar en futuras versiones de la aplicación. También se incluye una bibliografía con referencias a las fuentes utilizadas para la obtención de información, y un glosario de términos para aclarar conceptos técnicos que se encuentran a lo largo de este documento y un anexo donde se puede encontrar información adicional acerca de las pruebas realizadas.

## 1.4 Metodología y planificación

En este apartado se habla sobre la metodología y la planificación seguidas para el desarrollo del proyecto. Como se puede ver en la tabla 1, el desarrollo se ha dividido en nueve *sprints* o iteraciones, además de una iteración adicional llamada *Sprint 0*, la cual es característica en la metodología que se ha seguido y en la que se ha realizado la definición de requisitos y se han establecido las funcionalidades que debería tener la aplicación.

Tabla 1 - Planificación

<b>Sprint</b>	<b>Tareas</b>	<b>Fecha de inicio</b>	<b>Fecha de fin</b>
Sprint 0	➤ Definición de requisitos	03/02/2020	16/02/2020
Sprint 1	➤ Estructuración del proyecto en Android Studio ➤ Configuración de Firebase ➤ Configuración y activación de la API de Google	17/02/2020	01/03/2020
Sprint 2	➤ Autenticación de usuarios ➤ Gestión de intereses ➤ Interfaz gráfica de registro, inicio de sesión y elección de intereses ➤ Interfaz gráfica del menú principal	02/03/2020	15/03/2020
Sprint 3	➤ Gestión de lugares con la API de Google ➤ Interfaz gráfica pestaña "Mapa" ➤ Interfaz gráfica pestaña "Lugares"	16/03/2020	29/03/2020
Sprint 4	➤ Gestión de lugares por intereses ➤ Creación de iconos personalizados para los lugares ➤ Interfaz gráfica "Detalle de lugar" ➤ Interfaz gráfica pestaña "Descubrir"	30/03/2020	12/04/2020
Sprint 5	➤ Creación de rutas manualmente ➤ Interfaz gráfica "Mis Rutas" ➤ Guardado de rutas ➤ Gestión de votos de lugares	13/04/2020	26/04/2020
Sprint 6	➤ Generación automática de rutas ➤ Gestión de lugares favoritos ➤ Filtros en pantallas "Mapa", "Descubrir" y "Lugares" ➤ Interfaz gráfica pestaña "Perfil"	27/04/2020	10/05/2020
Sprint 7	➤ Interfaz gráfica "Mis lugares favoritos" ➤ Interfaz gráfica "Mis intereses" ➤ Interfaz gráfica "Mis votos" ➤ Cierre de sesión ➤ Cambio de contraseña	11/05/2020	24/05/2020
Sprint 8	➤ Optimización de funciones ➤ Ajustes en las interfaces gráficas	25/05/2020	07/06/2020
Sprint 9	➤ Pruebas con usuarios finales	08/06/2020	21/06/2020

### 1.4.1 Scrum

En este apartado se va a describir la metodología seguida en el desarrollo de este proyecto. Se ha optado por elegir *Scrum* ya que es una metodología que ya se conocía de proyectos anteriores. En este caso se ha adaptado para llevarla a cabo por una persona.

En *Scrum* se realizan entregas parciales y regulares del proyecto final, en las que se marcan unos plazos para que no se alarguen demasiado las entregas y se llegue al final del proyecto en los plazos establecidos, llevando a cabo un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. [1]

En esta metodología se utiliza el *Sprint 0* como un paso previo al desarrollo, esta iteración no es obligatoria, pero es muy útil para tener unas bases claras antes de comenzar el desarrollo.

En este caso, se han seguido plazos de entrega cortos y de duración fija, con iteraciones de dos semanas. Partiendo de una lista de objetivos marcados que actúan como plan del proyecto.

Para gestionar las iteraciones se ha utilizado Taiga, una herramienta ágil, de código abierto y gratuita, creada para gestionar y colaborar en proyectos ágiles. Es una herramienta sencilla e intuitiva. [2]

## 1.5 Relación con los estudios cursados

En este apartado se va a hablar de la relación de los conocimientos adquiridos en el grado que se han utilizado para llevar a cabo este proyecto. Ha sido necesario combinar contenidos de diferentes asignaturas y a continuación se van a destacar los aspectos más relevantes.

Por un lado, asignaturas como “Bases de datos y sistemas de la información” y “Tecnología de bases de datos” ambas cursadas en el tercer curso del grado, han sido de utilidad para gestionar y lograr una conexión estable con la base de datos.

Por otro lado, también ha servido de ayuda para la organización del proyecto la asignatura de “Gestión de Proyectos”, ya que me ha aportado una visión general de cómo llevar a cabo un proyecto y de la planificación que hay que seguir.

Por último, hay que destacar la base adquirida en todas las asignaturas relacionadas directamente con la programación.

## 2. Contexto tecnológico

---

En la actualidad, existen aplicaciones que se utilizan para la búsqueda de sitios de interés en una ubicación determinada, en los que se muestran imágenes y una breve descripción del lugar elegido. Con esta aplicación se pretende que, además de poder listar los lugares que hay en una ubicación, el usuario pueda crear una ruta en base a sus preferencias.

Se han tomado como referencia cinco aplicaciones. A continuación, se va a hacer una breve descripción de cada una de ellas y una tabla comparativa en las que se listarán los pros y contras de cada una de ellas.

En primer lugar, la aplicación *Batch Geocode*, [3] cuya interfaz principal se muestra en la figura 1.



Figura 1 - Captura de pantalla de la aplicación *Batch Geocode*

Esta aplicación genera una ruta en base a los puntos de geocalización<sup>2</sup> que recorre el usuario en tiempo real, permite listar los puntos en los que se detiene el usuario, añadir, actualizar y guardar una ruta realizada. Además, tiene una opción con la que poder filtrar las rutas realizadas.

Por otro lado, la aplicación, aunque no se encuentra en la Play Store como una aplicación de pago, tiene una versión PRO y la funcionalidad de la modalidad estándar es muy limitada.

Aplicación móvil para la generación de rutas turísticas basadas en las preferencias del usuario

En segundo lugar, la aplicación *Calimoto*, [4] cuya interfaz principal se muestra en la figura 2.



Figura 2 - Captura de pantalla de la aplicación *Calimoto*

Se trata de una aplicación específicamente para rutas en moto. En la que puedes crear rutas en base a la forma de las carreteras, por ejemplo, puedes elegir que sea una ruta con muchas curvas. En la ruta puedes elegir tu ubicación actual o una ubicación elegida.

Esta aplicación además permite guardar las rutas realizadas, monitorizar tu ruta en el momento que la realizas, etc.

Existe una versión premium en la que te permite acceder a más mapas y añadir nuevos filtros a las rutas. Además, esta versión mejora la recogida de datos de la ruta y los avisos personalizados para la ruta.

En tercer lugar, la aplicación *My phone tour*, [5] cuya interfaz principal se muestra en la figura 3.



Figura 3 - Captura de pantalla aplicación *My phone tour*

*My phone tour* es una aplicación que crea una ruta en base a los monumentos principales de la ciudad que se eligen. Además, muestra una pequeña descripción de esos lugares. Es una aplicación gratuita.

Sin embargo, esta aplicación ofrece muy poca funcionalidad ya que no permite modificar la ruta que ofrece. Dependiendo de la ciudad indicada tiene definidos los puntos de interés. Además, solo se pueden elegir las principales ciudades de España como destino para crear la ruta.

En cuarto lugar, la aplicación *MyRoute-app*, [6] cuya interfaz principal se muestra en la figura 4.



Figura 4 - Captura de pantalla de la aplicación *MyRoute-app*

Esta aplicación ofrece una gran variedad de opciones, además muestra actividades para realizar según la ubicación. También se pueden crear rutas y guardarlas tanto en modo privado como público, pudiendo compartirlas en este último. Es una aplicación gratuita.

Sin embargo, se trata de una aplicación que usa una vista Web para todas sus pantallas, lo cual reduce la usabilidad en dispositivos móviles.

En último lugar, la aplicación *Spain Travel Guide*, [7] cuya interfaz principal se muestra en la figura 5.



Figura 5 - Captura de pantalla de la aplicación *Spain Travel Guide*

Se trata de una aplicación en la que se pueden ver los diferentes lugares que hay una ciudad, con la posibilidad de filtrar por categoría. Es una aplicación gratuita.

A pesar de ello, la aplicación no permite crear una ruta, sino que únicamente permite visualizar los lugares de las principales ciudades de España.

Existe una versión de esta aplicación para distintos países, siendo única para ese país.

Una vez realizada una pequeña descripción de las aplicaciones más significativas que se han encontrado, se va a hacer una tabla comparativa de sus funcionalidades, como se muestra en la tabla 2.

Tabla 2 - Comparativa aplicaciones significativas

	<b>Batch Geocode</b>	<b>Calimoto</b>	<b>My phone tour</b>	<b>MyRoute-app</b>	<b>Spain</b>
<b>Generación de rutas</b>	✓	✓		✓	
<b>Filtro por preferencias</b>		✓			✓
<b>Ubicación actual</b>	✓	✓		✓	
<b>Guardar rutas</b>		✓		✓	
<b>Gratuita</b>			✓	✓	✓
<b>Descripción de lugares</b>			✓	✓	✓
<b>Navegación en tiempo real por la ruta</b>	✓			✓	

Atendiendo a las aplicaciones que se han analizado, se ha encontrado la necesidad de crear una aplicación que, además de ofrecer un listado de los lugares que hay en cada ubicación, permita crear una ruta en base a las preferencias que tenga el usuario. Además de que sea posible usarla en cualquier parte del mundo, sin necesidad de aplicaciones adicionales o compras internas. También, se pretende mejorar algunas de las funcionalidades encontradas en otras aplicaciones ya existentes, como por ejemplo ofrecer la posibilidad de modificar la ruta en tiempo real mientras se está realizando. Esta aplicación será gratuita.

# 3. Especificación de requisitos

En este apartado se describen las funciones que debe seguir la aplicación, para ello se van a utilizar diagramas de casos de uso y de clases.

## 3.1 Diagrama de casos de uso

Con el diagrama de casos de uso se especifica la funcionalidad de la aplicación desde el punto de vista del usuario.

En el diagrama, que se puede ver en las figuras 6 y 7, se han definido dos actores diferentes que representan a un usuario no autenticado y a un usuario autenticado. En primer lugar, un usuario no autenticado solo puede registrarse en la aplicación.

Por otro lado, una vez que el usuario se ha registrado puede listar los lugares, votar lugares y añadirlos a sus favoritos, así como ver más detalles de un lugar y añadirlo a una ruta.

Además, un usuario registrado puede crear una ruta añadiendo lugares a la misma, modificarla y guardarla, también puede listar las rutas guardadas, ver detalle de cada una de ellas y borrarlas.

Por otro lado, el usuario puede como modificar sus intereses, listar sus lugares favoritos, ver sus votos, cambiar la contraseña o cerrar sesión.



Figura 6 - Diagrama de casos de uso usuario autenticado.



Figura 7 - Diagrama de casos de uso usuario no autenticado.

## 3.2 Descripción de los casos de uso

A continuación, se van a describir con más detalle cada uno de los casos de uso vistos en las figuras 6 y 7.

### 3.2.1 Actor no autenticado

En este apartado se van a describir los casos de uso correspondientes a la figura 7.

Tabla 3 - Caso de uso registrarse - Descripción

<b>Caso de uso</b>	Registrarse
<b>Actor</b>	Usuario no autenticado
<b>Propósito</b>	Crear una cuenta
<b>Resumen</b>	El usuario introduce los datos solicitados y se registra.
<b>Precondiciones</b>	El usuario no está registrado.
<b>Postcondiciones</b>	El usuario queda registrado.

Tabla 4 - Caso de uso registrarse - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario abre la aplicación.	2. El sistema muestra la pantalla de autenticación.
3. El usuario accede a la pantalla de registro.	4. El sistema pide los datos al usuario.
5. El usuario rellena sus datos en los formularios	5. El sistema comprueba que los datos introducidos cumplen las restricciones y registra al usuario.

Tabla 5 - Caso de uso registrarse - extensiones síncronas

Extensiones síncronas
#1. Si no se introducen datos en los campos obligatorios o no cumplen con el formato correcto, aparecerá un mensaje de error. (5)

Tabla 6 - Caso de uso elegir intereses - Descripción

<b>Caso de uso</b>	Elegir intereses
<b>Actor</b>	Usuario no autenticado
<b>Propósito</b>	Elegir los intereses del usuario
<b>Resumen</b>	El usuario selecciona entre las opciones mostradas sus intereses
<b>Precondiciones</b>	El usuario ha introducido los datos solicitados en el registro.
<b>Postcondiciones</b>	Se guardan los intereses para el usuario

Tabla 7 - Caso de uso elegir intereses - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario termina el registro.	2. El sistema muestra la pantalla de selección de intereses.
3. El usuario elige sus intereses.	4. El sistema guarda los intereses.

### 3.2.2 Actor autenticado

En este apartado se van a describir los casos de uso correspondientes al diagrama de la figura 6.

Tabla 8 - Caso de uso iniciar sesión - Descripción

<b>Caso de uso</b>	Iniciar sesión
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Acceder a la aplicación
<b>Resumen</b>	El usuario introduce el correo electrónico y la contraseña y accede a la aplicación
<b>Precondiciones</b>	El usuario se ha registrado
<b>Postcondiciones</b>	-

Tabla 9 - Caso de uso iniciar sesión - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario abre la aplicación.	2. El sistema muestra la pantalla de autenticación y pide las credenciales al usuario.
3. El usuario introduce su correo electrónico y contraseña.	4. El sistema comprueba que los datos son correctos
	5. El sistema muestra la pantalla principal.

Tabla 10 - Caso de uso iniciar sesión - Extensiones síncronas

Extensiones síncronas
#1. Si se introducen unas credenciales para el usuario que no están registradas en la base de datos, aparecerá un mensaje de error. (4)

Tabla 11 - Caso de uso listar lugares - Descripción

<b>Caso de uso</b>	Listar lugares
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Visualizar los lugares existentes en una ubicación
<b>Resumen</b>	El usuario puede visualizar información de los lugares que hay en la ubicación indicada
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 12 - Caso de uso listar lugares - Interacciones

Intenciones de usuario	Obligaciones del sistema
1. El usuario selecciona en el menú la opción "Lugares"	2. El sistema muestra la lista de lugares

Tabla 13 - Caso de uso ver detalle de lugar - Descripción

<b>Caso de uso</b>	Ver detalles de lugar
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Obtener más información acerca de un lugar indicado
<b>Resumen</b>	El usuario marca el lugar que le interesa y obtiene una información más detallada de este.
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 14 - Caso de uso ver detalle de lugar - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario pulsa en un lugar.	2. El sistema muestra los detalles del lugar seleccionado.

Tabla 15 - Caso de uso añadir lugar a favoritos - Descripción

<b>Caso de uso</b>	Añadir lugar a favoritos
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Marcar como favorito un lugar que le ha gustado y quiere volver a ver
<b>Resumen</b>	El usuario marca como favorito un lugar
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	Se añade el lugar a la lista de favoritos del usuario

Tabla 16 - Caso de uso añadir lugar a favoritos- Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario pulsa el botón para añadir un lugar a sus favoritos.	2. El sistema añade el lugar a su lista de favoritos.

Tabla 17 - Caso de uso listar lugares favoritos - Descripción

<b>Caso de uso</b>	Listar lugares favoritos
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Ver los lugares que más le gustan al usuario
<b>Resumen</b>	El usuario accede a la lista en la que se muestran sus lugares favoritos
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 18 - Caso de uso listar lugares favoritos - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario selecciona en el menú la opción "Perfil".	2. El sistema muestra la pantalla de perfil.
3. El usuario abre el menú lateral.	4. El sistema muestra las opciones.
5. El usuario elige la opción mis favoritos	6. El sistema muestra la lista de lugares favoritos.

Tabla 19 - Caso de uso votar lugar - Descripción

<b>Caso de uso</b>	Votar lugar
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Votar un lugar para que el usuario pueda elegir en base a las votaciones
<b>Resumen</b>	El usuario da su opinión de un lugar y lo marca con una votación
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	Se marca el lugar con la votación de los usuarios

Tabla 20 - Caso de uso votar lugar - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario selecciona un lugar.	2. El sistema muestra la pantalla de detalle de lugar.
3. El usuario introduce una votación.	4. El sistema muestra la votación actualizada y lo añade a la lista de mis votos.

Tabla 21 - Caso de uso ver lugares votados - Descripción

<b>Caso de uso</b>	Ver lugares votados
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Poder ver los lugares puntuados
<b>Resumen</b>	El usuario accede a la lista con los lugares que ha votado
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 22 - Caso de uso ver lugares votados - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario selecciona en el menú la opción "Perfil".	2. El sistema muestra la pantalla de perfil.
3. El usuario abre el menú lateral.	4. El sistema muestra las opciones.
5. El usuario elige la opción mis votos	6. El sistema muestra la lista de lugares votados.

Tabla 23 - Caso de uso añadir lugar a ruta - Descripción

<b>Caso de uso</b>	Añadir lugar a ruta
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Añadir un lugar a una ruta
<b>Resumen</b>	El usuario ve un lugar que quiere visitar y lo marca para que se añada a la ruta
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	El lugar marcado se añade a la ruta

Tabla 24 - Caso de uso añadir lugar a ruta - Interacción

<b>Intenciones de usuario</b>	<b>Obligaciones del sistema</b>
1. El usuario pulsa el botón para añadir un lugar a la ruta	2. El sistema añade el lugar a la ruta.
	3. El sistema actualiza el mapa y muestra la ruta con el lugar añadido.

Tabla 25 - Caso de uso ver ruta - Descripción

<b>Caso de uso</b>	Ver ruta
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Ver una ruta guardada
<b>Resumen</b>	El usuario accede a una ruta ya creada
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 26 - Caso de uso ver ruta - Interacción

<b>Intenciones de usuario</b>	<b>Obligaciones del sistema</b>
1. El usuario pulsa en el menú en la opción "Mis rutas".	2. El sistema muestra la lista de las rutas guardadas.
3. El usuario pulsa en una ruta	4. El sistema muestra los detalles de la ruta seleccionada

Tabla 27 - Caso de uso editar ruta - Descripción

<b>Caso de uso</b>	Editar ruta
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Modificar una ruta creada
<b>Resumen</b>	El usuario accede a una ruta que ha creado y la modifica
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 28 - Caso de uso editar ruta - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario pulsa en el menú en la opción "Mis rutas".	2. El sistema muestra la lista de las rutas guardadas.
3. El usuario pulsa en una ruta.	4. El sistema muestra los detalles de la ruta seleccionada.
5. El usuario pulsa el botón de modificar.	5. El sistema muestra la vista de edición.
6. El usuario edita la ruta.	7. El sistema actualiza la información mostrada.
8. El usuario confirma los datos de la ruta.	9. El sistema guarda la ruta editada.

Tabla 29 - Caso de uso guardar ruta - Descripción

<b>Caso de uso</b>	Guardar ruta
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Crear una ruta que se va a realizar posteriormente
<b>Resumen</b>	El usuario crea una ruta y la guarda para realizarla posteriormente
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	La ruta se guarda

Tabla 30 - Caso de uso guardar ruta - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario pulsa en el menú en la opción "Mapa".	2. El sistema muestra el mapa.
3. El usuario pulsa el botón flotante de la parte inferior derecha de la pantalla.	4. El sistema despliega las opciones.
5. El usuario pulsa el botón de guardar ruta.	6. El sistema guarda la ruta y la añade a las rutas del usuario.

Tabla 31 - Caso de uso listar rutas realizadas - Descripción

<b>Caso de uso</b>	Listar rutas realizadas
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Ver las rutas creadas
<b>Resumen</b>	El usuario puede ver las rutas que ha creado
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 32 - Caso de uso listar rutas realizadas - Interacción

Intenciones de usuario	Obligaciones del sistema
1. El usuario pulsa la opción del menú "Mis rutas".	2. El sistema muestra la lista de rutas del usuario.

Tabla 33 - Caso de uso borrar ruta - Descripción

<b>Caso de uso</b>	Borrar ruta
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Poder borrar una ruta realizada
<b>Resumen</b>	El usuario elimina una ruta
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 34 - Caso de uso borrar ruta - Interacción

<b>Intenciones de usuario</b>	<b>Obligaciones del sistema</b>
1. El usuario pulsa en el menú en la opción "Mis rutas".	2. El sistema muestra la lista de rutas realizadas.
3. El usuario pulsa en la ruta que quiere borrar.	4. El sistema muestra el detalle de la ruta marcada.
5. El usuario pulsa el botón borrar.	6. El sistema elimina la ruta.

Tabla 35 - Caso de uso editar intereses - Descripción

<b>Caso de uso</b>	Editar intereses
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Cambiar la selección de lugares que se muestran en la aplicación en base a su ubicación
<b>Resumen</b>	El usuario accede a su lista de interés y puede eliminar o añadir intereses nuevos.
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 36 - Caso de uso editar intereses - Interacción

<b>Intenciones de usuario</b>	<b>Obligaciones del sistema</b>
1. El usuario selecciona en el menú la opción "Perfil".	2. El sistema muestra la pantalla de perfil.
3. El usuario abre el menú lateral.	4. El sistema muestra las opciones.
5. El usuario elige la opción editar intereses.	6. El sistema muestra la lista de intereses.
7. El usuario modifica sus intereses.	8. El sistema guarda los intereses del usuario.

Tabla 37 - Caso de uso editar intereses - Extensiones síncronas

<b>Extensiones síncronas</b>
#1. Si se han elegido menos de tres intereses se muestra un error. (7)

Tabla 38 - Caso de uso cambiar contraseña - Descripción

<b>Caso de uso</b>	Cambiar contraseña
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Cambiar la contraseña
<b>Resumen</b>	El usuario puede cambiar la contraseña ya sea porque la ha olvidado o porque quiere poner otra nueva
<b>Precondiciones</b>	-
<b>Postcondiciones</b>	-

Tabla 39 - Caso de uso cambiar contraseña - Interacción

<b>Intenciones de usuario</b>	<b>Obligaciones del sistema</b>
1. El usuario selecciona en el menú la opción "Perfil".	2. El sistema muestra la pantalla de perfil.
3. El usuario selecciona el botón cambiar contraseña.	4. El sistema muestra el formulario para cambiar la contraseña.
5. El usuario introduce la contraseña actual y la nueva contraseña	6. El sistema comprueba que la contraseña es correcta
	7. El sistema guarda la contraseña

Tabla 40 - Caso de uso cambiar contraseña - Extensiones síncronas

<b>Extensiones síncronas</b>
#1. Si la contraseña no es correcta, se muestra un error. (6)

Tabla 41 - Caso de uso cerrar sesión - Descripción

<b>Caso de uso</b>	Cerrar sesión
<b>Actor</b>	Usuario autenticado
<b>Propósito</b>	Salir de la aplicación
<b>Resumen</b>	El usuario sale de la aplicación
<b>Precondiciones</b>	El usuario ha iniciado sesión
<b>Postcondiciones</b>	-

Tabla 42 - Caso de uso cerrar sesión - Interacción

<b>Intenciones de usuario</b>	<b>Obligaciones del sistema</b>
1. El usuario selecciona en el menú la opción "Perfil".	2. El sistema muestra la pantalla de perfil.
3. El usuario abre el menú lateral.	4. El sistema muestra las opciones.
5. El usuario elige la opción cerrar sesión	6. El sistema cierra la aplicación

### 3.3 Diagrama de clases

En este apartado se van a representar las entidades que se han utilizado en la aplicación. Para ello, se va a usar un diagrama de clases.

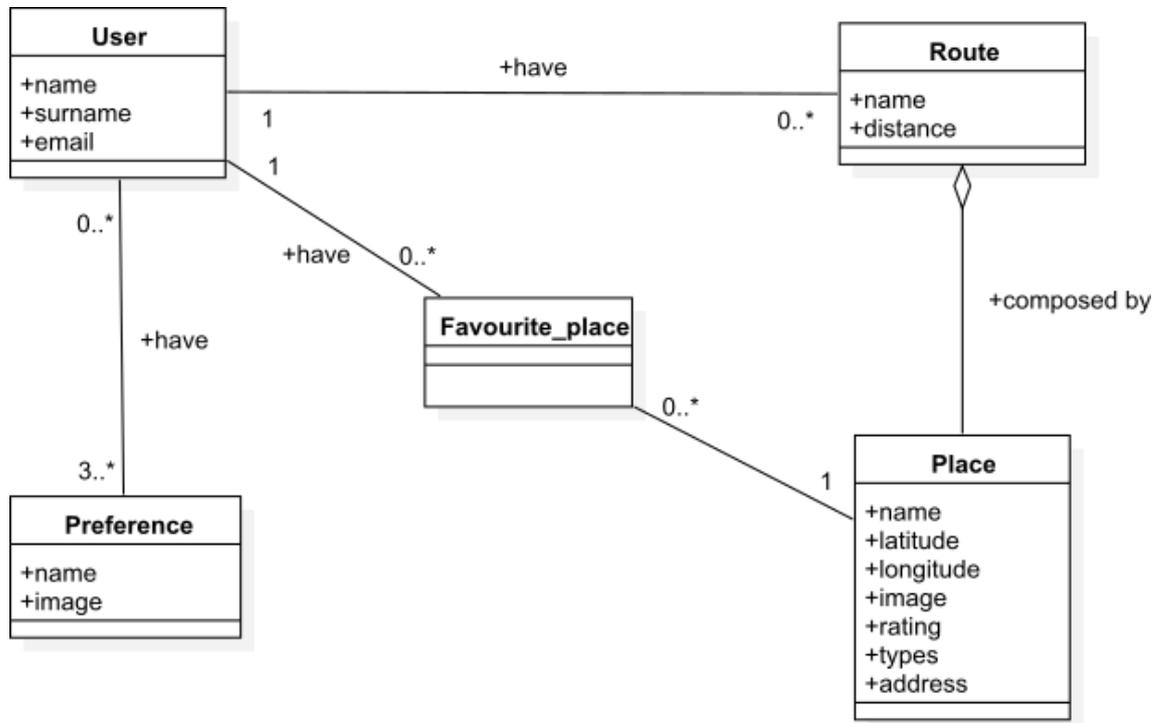


Figura 8 - Diagrama de clases

Como se puede ver en el diagrama de la figura 8, un usuario puede seleccionar tres o más preferencias. Por otro lado, puede crear rutas, las cuales están compuestas por lugares. Para crear la ruta se utilizan los puntos físicos de los lugares, determinados por sus coordenadas. Además, un usuario puede añadir un lugar a sus favoritos.

# 4. Diseño

---

En este apartado se introduce la arquitectura de la aplicación y como se conecta con los distintos servicios utilizados. También se explica el modelo de datos mediante un diagrama entidad-relación.

Por último, se muestran capturas de los prototipos iniciales y el resultado final de la interfaz de usuario, así como una explicación de la funcionalidad de cada uno de ellos y la relación que tiene cada uno de los prototipos con los casos de uso.

## 4.1 Arquitectura

La arquitectura de la aplicación se divide en dos partes:

- Una parte *backend*<sup>9</sup> en tiempo real realizada con Firebase.
- Otra parte *frontend*<sup>4</sup> diseñada para dispositivos Android, implementada en lenguaje Kotlin.

Siguiendo las directrices extraídas de Google Cloud, los servicios de *backend* de aplicaciones para dispositivos móviles deben cumplir los siguientes requisitos: [8]

- Limitar el almacenamiento de datos en el dispositivo.
- Permitir la sincronización de datos entre varios dispositivos.
- Manejar el modo sin conexión con facilidad.
- Envío de notificaciones y mensajes.
- Minimizar el gasto de batería.

Para llevar a cabo estos requisitos, se ha elegido un diseño que utiliza la nube de Google para crear servicios de *backend*, el diseño se muestra en la figura 9, este patrón proporciona una sincronización automática de datos, servicios de autenticación de usuario, mensajería, almacenamiento de archivos, estadísticas, entre otros.

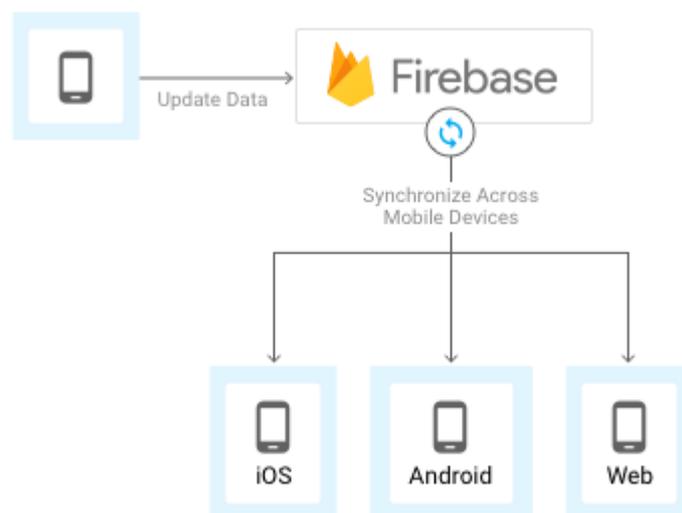


Figura 9 – Arquitectura de Firebase

Además, para la gestión de los lugares, las rutas y los mapas se han utilizado las siguientes APIs<sup>5</sup> que proporciona Google, concretamente:

- **Directions API** es un servicio que calcula rutas entre varias ubicaciones mediante una solicitud HTTP<sup>6</sup>, indicando el punto de inicio y destino, el modo de transporte en que se desea realizar dicha ruta, entre otros parámetros. [9]
- **Distance Matrix API** es un servicio que da información acerca de la distancia y tiempo de viaje entre dos lugares. [10]
- **Maps SDK<sup>7</sup> for Android**, esta API permite agregar mapas basados en datos de Google Maps a la aplicación. En la llamada a la API se permite añadir marcadores, polígonos, cambiar la vista del usuario, entre otros. [11]
- **Places API** es un servicio que devuelve información sobre los lugares mediante solicitudes HTTP. Dentro de esta API se permiten las siguientes solicitudes: [12]
  - Búsqueda de lugares basada en la ubicación.
  - Detalles de un lugar.
  - Place Photos, da acceso a fotos relacionadas con el lugar.
  - Place Autocomplete, rellena información de un lugar a medida que el usuario escribe el nombre o la dirección.
  - Query Autocomplete, predice las consultas por búsquedas que el usuario ha realizado, sugiriendo consultas al usuario mientras escribe.

La conexión con las APIs de Google se realiza mediante una petición REST<sup>8</sup>, se trata de una arquitectura de desarrollo web que se apoya en el estándar HTTP, el intercambio de los datos proporcionados por la API se hace en formato JSON<sup>9</sup>.

Para gestionar la comunicación con Firebase, con las APIs y el resto de los servicios, se han utilizado diferentes librerías:

- **Gson**: Librería de código abierto desarrollada por Google para convertir un objeto en su representación JSON y viceversa. Se puede utilizar sobre objetos de cualquier tipo.
- **OkHttp**: Es una librería de código abierto que permite realizar peticiones HTTP en Java<sup>10</sup>. Facilita la gestión de archivos multimedia ya que evalúa las características de la conexión y determina como operar de la mejor manera.
- **Data Binding**: Es una librería Android para vinculación de datos. Permite vincular los componentes de la interfaz gráfica a las fuentes de datos de la aplicación. Esta librería permite reducir llamadas al *framework*<sup>11</sup> de la interfaz de usuario en las actividades, resultando más sencillas y fáciles de mantener, mejorando el rendimiento de la aplicación. [13]

## 4.2 Modelo de datos

El almacenamiento de datos se ha hecho a través de la plataforma Firebase, esta permite que los datos estén actualizados en tiempo real, por lo que se puede acceder a ellos desde cualquier dispositivo. Estos datos se guardan en una nube de Google por lo que están accesibles para los usuarios en cuestión de milisegundos.

La base de datos utilizada en Firebase es NoSQL<sup>12</sup>, los datos no se almacenan en tablas ni registros, sino que se almacenan en documentos y estos se organizan en colecciones.

Cada documento contiene un conjunto de pares clave-valor. Firebase está optimizado para almacenar grandes colecciones de documentos pequeños, además cada colección puede contener subcolecciones de datos primitivos o complejos.

Los documentos disponen de un identificador que permite localizarlos y tienen una estructura similar al formato JSON.

A pesar de esto, para las lecturas y escrituras a los nodos sí que se hace a través de información extraída de otros nodos. Por ejemplo, cuando un lugar se añade a una ruta, este lugar se queda registrado en el nodo de la ruta. Por lo tanto, sí que se puede representar la estructura de información de una forma más comprensible como se muestra en la figura 10.

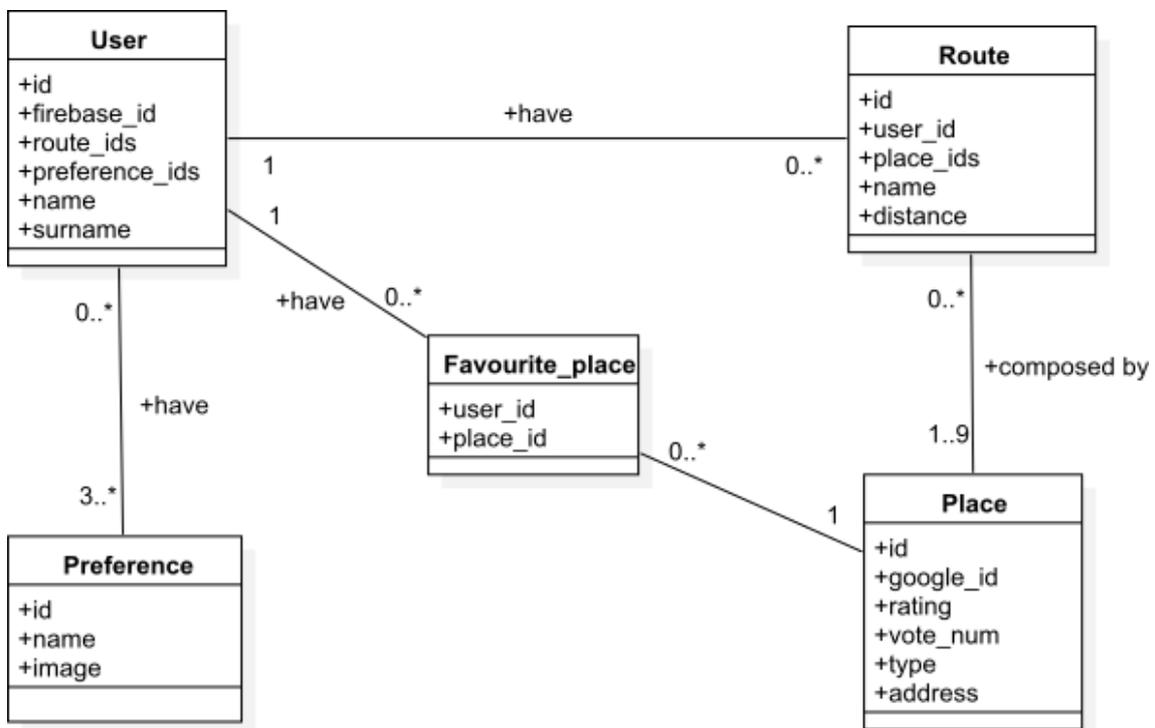


Figura 10 - Modelo de datos

Como se puede ver en el diagrama de la figura 10, todas las tablas tienen un identificador único que se utiliza como clave primaria. Y como claves ajenas los atributos con el sufijo *\_id*. Por ejemplo, en la tabla “*Route*”, *user\_id* hace referencia al usuario que crea la ruta y *place\_ids* a los identificadores de los lugares que la forman.

## 4.3 Prototipos de la interfaz

En este apartado se van a mostrar los prototipos iniciales de la aplicación, así como una breve explicación de la funcionalidad que se va a llevar a cabo y los casos de uso que se implementan en cada uno.

### 4.3.1 Inicio de sesión

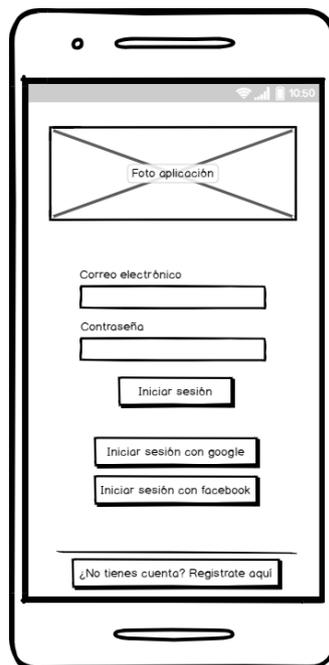


Figura 11 - Prototipo inicio de sesión

En la figura 11, se muestra el prototipo de la pantalla de inicio de sesión, en esta pantalla el usuario puede acceder con su correo electrónico y contraseña una vez ya se ha registrado en la aplicación. También puede iniciar sesión con su cuenta de Google o Facebook. Si el usuario no está registrado, puede acceder al registro pulsando en el botón de la parte inferior.

**Casos de uso:** Iniciar sesión, cambiar contraseña

### 4.3.2 Registro

Este prototipo muestra una pantalla de registro con dos pestañas: "Datos personales" (seleccionada) e "Inicio de sesión". El formulario solicita los siguientes datos:

- Nombre
- Apellidos
- Fecha de nacimiento
- Población
- Teléfono

En la parte inferior derecha hay un botón con una flecha hacia adelante. En la parte inferior izquierda hay un botón con una flecha hacia atrás. En la parte inferior central hay un botón que dice "¿Ya tienes cuenta? Inicia sesión aquí".

Figura 12 - Prototipo registro I

Este prototipo muestra la segunda pantalla de registro, con las pestañas "Datos personales" e "Inicio de sesión" (seleccionada). El formulario solicita:

- Correo electrónico\*
- Contraseña\*
- Repita contraseña\*

Debajo de los campos hay un botón que dice "Registrarse". En la parte inferior izquierda hay un botón con una flecha hacia atrás. En la parte inferior central hay un botón que dice "¿Ya tienes cuenta? Inicia sesión aquí".

Figura 13 - Prototipo registro II

En este apartado se muestran los prototipos de la pantalla de registro, el registro se divide en dos pantallas distintas. En la figura 12, se le solicitan al usuario los datos personales y en la figura 13, el correo electrónico y la contraseña con la que va a acceder a la aplicación. Si el usuario accede a la pantalla registro por error, es posible volver a la pantalla de inicio de sesión pulsando en el botón de la parte inferior de la pantalla, además el usuario puede moverse entre las dos pestañas libremente pulsando las flechas de delante y atrás. Una vez rellenados los campos obligatorios de manera correcta, el usuario pulsará el botón "Registrarse" con el que podrá acceder a la elección de sus intereses, como se ve en la figura 14.

**Casos de uso:** Registrarse

### 4.3.3 Elección de intereses

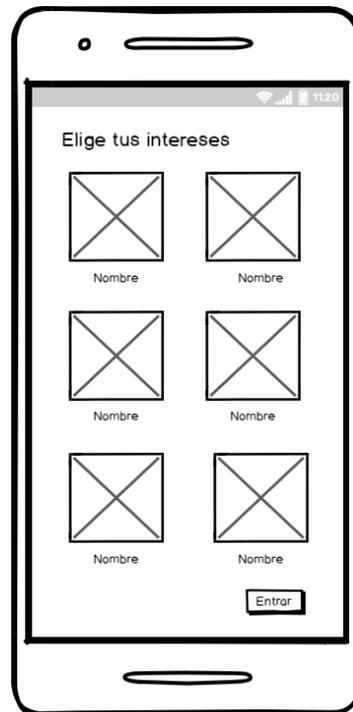


Figura 14 - Prototipo elegir intereses

En la figura 14 se muestra el prototipo de la pantalla de elección de intereses, en esta pantalla el usuario elige sus preferencias entre las disponibles. Una vez elegidos los intereses el usuario pulsará el botón "Entrar" con el que accederá al menú principal de la aplicación.

**Casos de uso:** Elegir intereses

## 4.3.4 Menú principal

### 4.3.4.1 Pestaña lugares

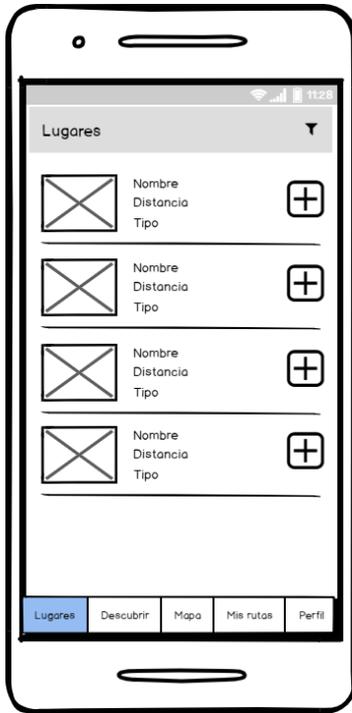


Figura 15 - Prototipo pestaña "Lugares"

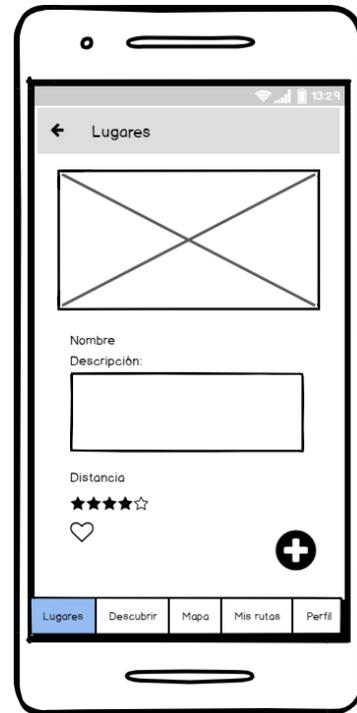


Figura 16 - Prototipo información de un lugar

En la figura 15, se muestra el prototipo de la pestaña lugares, esta pestaña se encuentra en el menú principal.

En esta pantalla se listan los lugares cercanos a la ubicación del usuario en función de sus intereses. Se muestra una lista con la imagen, nombre del lugar, distancia a la que se encuentra en usuario y tipo de interés que es.

Si el usuario pulsa en cualquier elemento de la lista, accede a la vista de la figura 16 en la que se puede observar en mayor detalle la descripción del lugar seleccionado. Pulsando el botón "+" se puede añadir el lugar a la ruta que está creando.

**Casos de uso:** Listar lugares, añadir lugar a ruta, votar lugar, añadir lugar a favoritos, ver detalles de lugar.

#### 4.3.4.2 Pestaña descubrir

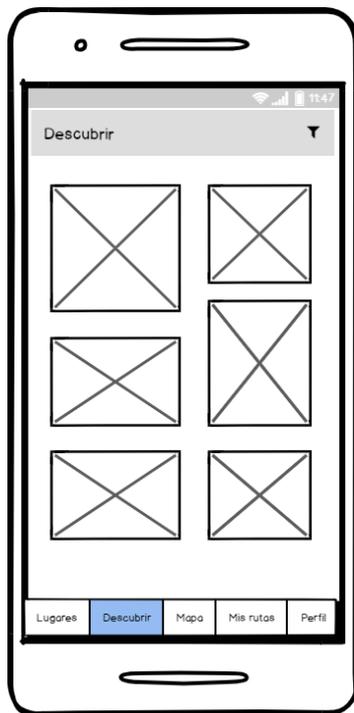


Figura 17 - Prototipo pestaña "Descubrir"

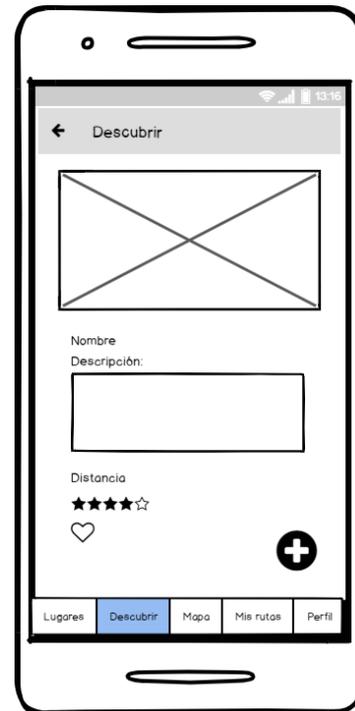


Figura 18 - Prototipo información de un lugar

En la figura 17 se muestra el prototipo de la de pestaña descubrir, una de las pestañas dentro del menú principal de la aplicación.

En esta pestaña se muestran las imágenes de lugares que pueden interesar al usuario, independientemente de la ubicación donde se encuentre, pudiendo añadir filtros para mejorar la búsqueda.

Si el usuario selecciona cualquier imagen de la lista, podrá ver una descripción más detallada del lugar seleccionado como se muestra en la figura 18, en esta pantalla el usuario puede además de ver información del lugar, votarlo, añadirlo a sus favoritos o a una ruta.

**Casos de uso:** Listar lugares, añadir lugar a ruta, votar lugar, añadir lugar a favoritos, ver detalles de lugar.

#### 4.3.4.3 Pestaña mapa

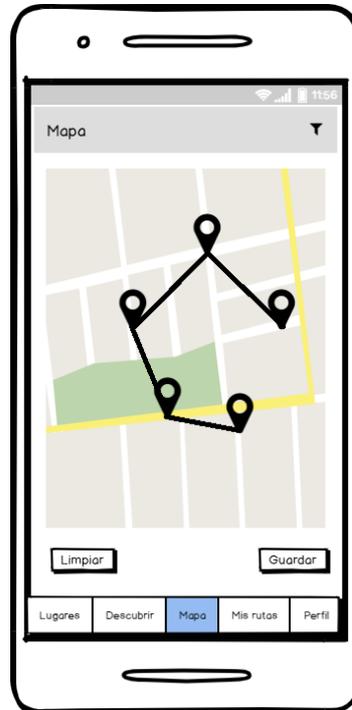


Figura 19 - Prototipo pestaña "Mapa"

En la figura 19 se muestra el prototipo de la pestaña principal del menú de la aplicación, esta será la vista que se mostrará al iniciar sesión.

En esta vista se puede ver una vista de un mapa en él se muestran marcados con marcadores los lugares cercanos a la ubicación del usuario en función de los intereses establecidos previamente. El usuario observará estos lugares unidos, formando una ruta que podrá guardar para realizarla posteriormente o realizar en el momento de su creación. Si se selecciona un marcador, aparece una ventana flotante en la que se muestra una descripción del lugar.

En la parte inferior hay dos botones, uno para guardar la ruta creada y otro para eliminar la ruta que se muestra actualmente en el mapa.

También existe la opción de añadir filtros para mejorar la búsqueda.

**Casos de uso:** Ver ruta, editar ruta, añadir lugar a favoritos, ver detalles de lugar, guardar ruta

#### 4.3.4.4 Pestaña mis rutas

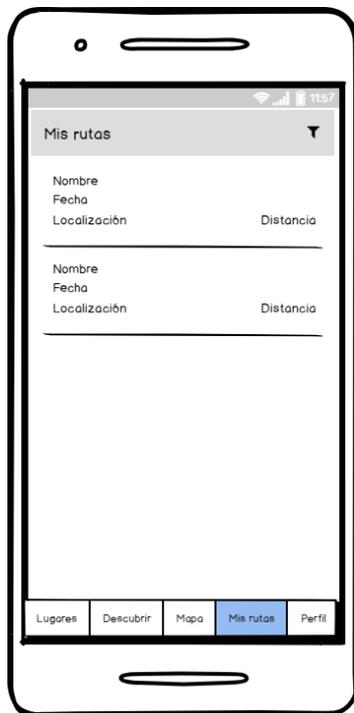


Figura 20 - Prototipo pestaña "Mis rutas"

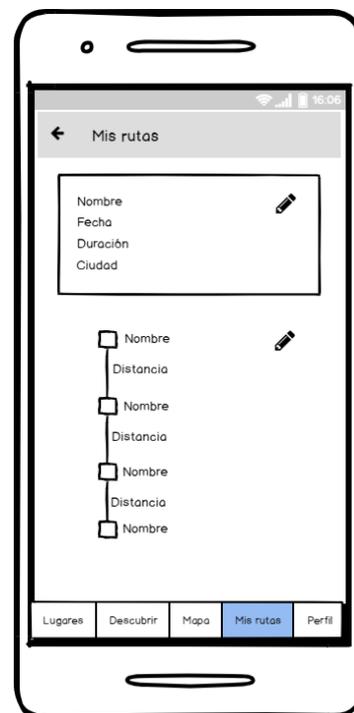


Figura 21 - Prototipo información de una ruta

En la figura 20, se muestra la vista de la pestaña "Mis rutas" del menú principal. En este prototipo se muestra una pequeña descripción de las rutas guardadas, se puede filtrar por alguno de los campos mostrados para buscar la ruta guardada deseada.

En la figura 21, se puede ver el prototipo que muestra una ruta con más detalle, en el que se permite editar los datos de esta.

**Casos de uso:** listar rutas realizadas, editar ruta, borrar ruta

#### 4.3.4.4 Pestaña perfil

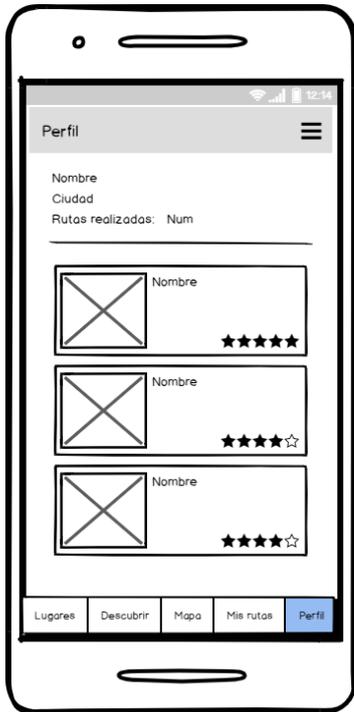


Figura 22 - Prototipo pestaña "Perfil"

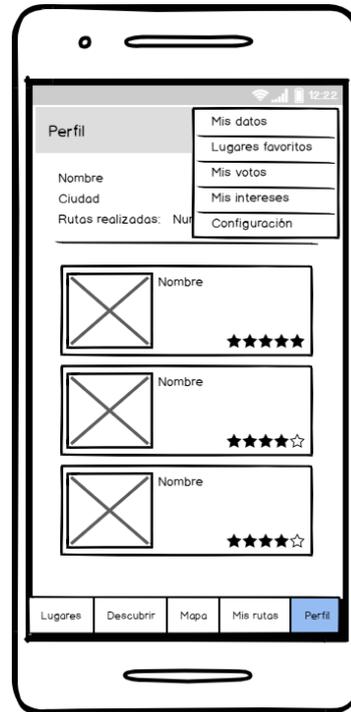


Figura 23 - Prototipo de menú desplegable

En la pestaña "Perfil" se muestra la vista mostrada en la figura 22, en esta se muestra información del usuario, el número de rutas realizadas y además los lugares favoritos del usuario. Además, pulsando en el botón de la barra superior se accede a un menú lateral mostrado en la figura 23 en el que aparecen distintas opciones que permiten al usuario ver sus lugares favoritos o editar sus intereses entre otras opciones como se puede ver en las figuras 24, 25 y 26.

**Casos de uso:** Listar intereses, editar intereses, listar lugares favoritos, ver votos, editar datos personales, cambiar contraseña, cerrar sesión

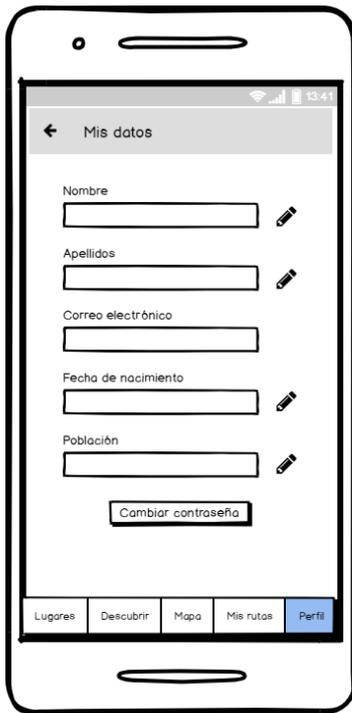


Figura 24 - Prototipo pantalla "Mis datos"

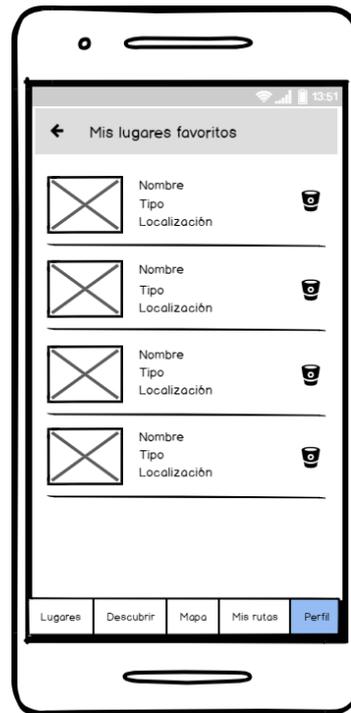


Figura 25 - Prototipo pantalla "Mis lugares favoritos"

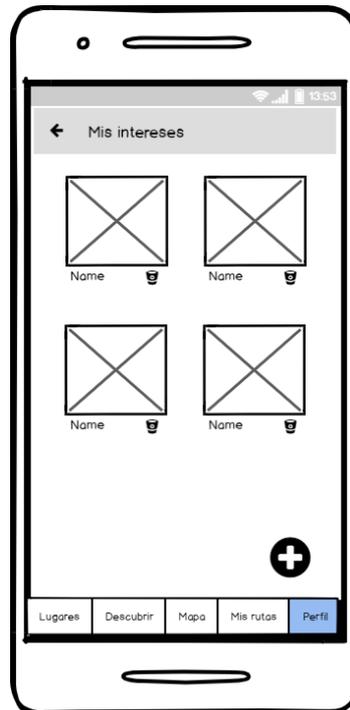


Figura 26 - Prototipo pantalla "Mis intereses"

## 4.4 Diagrama de navegación

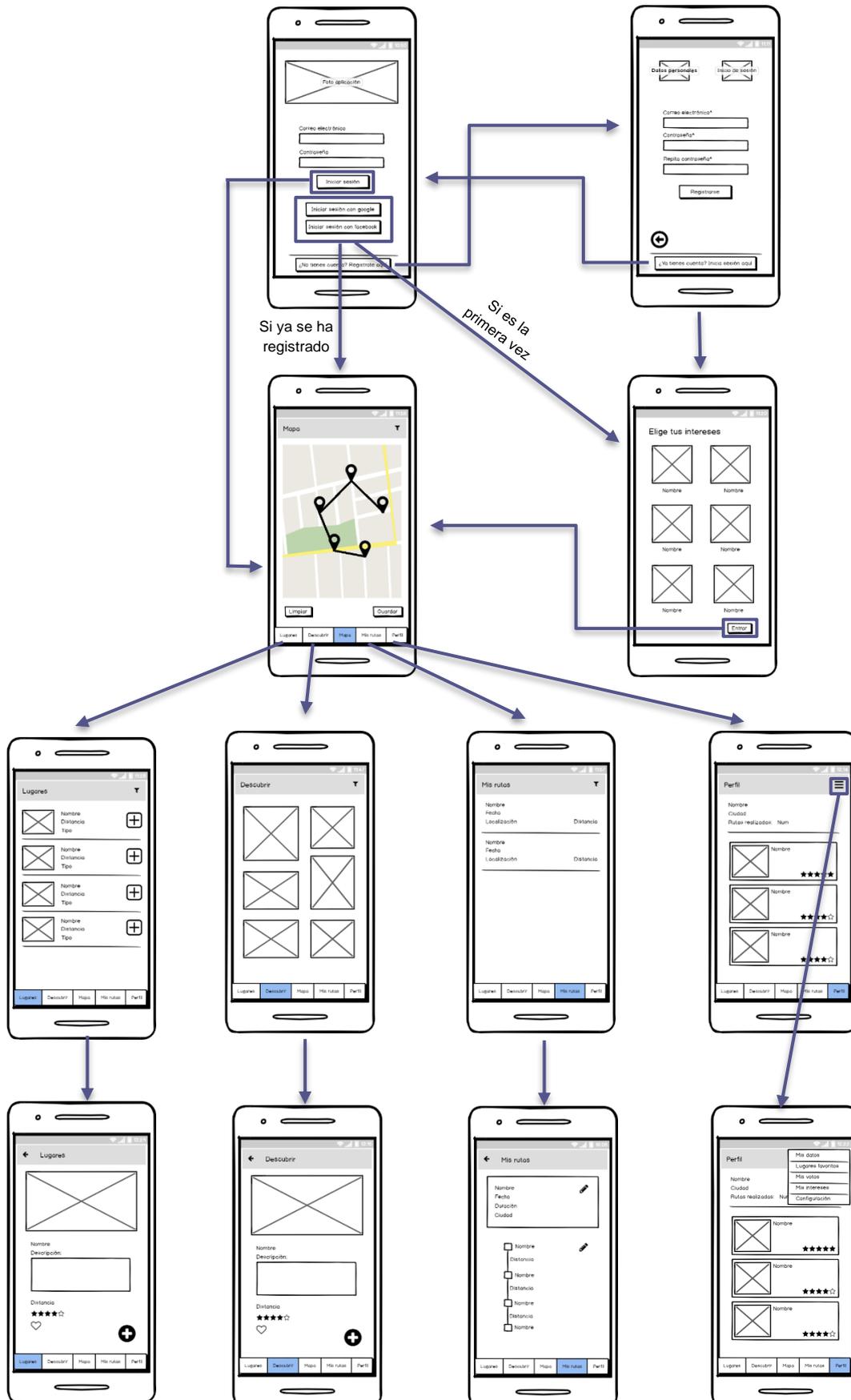


Figura 27 - Diagrama de secuencia prototipos

## 4.5 Material Design

Material Design es un conjunto de buenas prácticas que debe seguir una aplicación, utilizadas para conseguir que los objetos que la forman se aproximen a la realidad y sean fácilmente reconocibles por el usuario. [14]

Fue creado por Google con el objetivo de unificar sus interfaces y sus principios clásicos son el buen diseño, la adaptabilidad y la innovación.

En Android se empezó a utilizar a partir de la versión 5.0. No se utiliza únicamente en dispositivos móviles, sino que se puede utilizar en todo tipo de contenidos digitales como páginas web, aplicaciones para ordenadores, etc. [15]

Se caracteriza por emplear un estilo visual que se asemeja a un espacio real, el cual hace que la comodidad del usuario se incremente haciendo énfasis en los siguientes puntos:

- Minimalismo y colores planos, siendo el color gris el elemento de separación, reforzando así el resto de los colores llamativos.
- Sombreado para simular un efecto de relieve entre distintos niveles.
- Iconos vectoriales para evitar pixelado entre los diferentes tamaños de pantalla
- Animación de elementos ya que ayuda a aclarar el significado de estos y evita añadir textos aclaratorios.
- Concepto espacial 3D, para que el usuario perciba las formas de una forma más natural.
- Homogeneidad en las vistas de todas las pantallas.

# 5. Implementación

---

## 5.1 Herramientas para el desarrollo

En este apartado se va a explicar el software utilizado para desarrollar la aplicación, así como las razones de la utilización de cada una de ellas.

### 5.1.1 Android Studio

Android Studio [16] es un IDE basado en IntelliJ IDEA<sup>13</sup> para el desarrollo de aplicaciones Android. Se ha escogido esta herramienta porque es un editor de códigos muy potente y además es el entorno oficial para el desarrollo de aplicaciones Android.

Android Studio permite combinar código Java y Kotlin en un mismo proyecto, pudiendo crear clases en ambos lenguajes de programación y traducir directamente cualquier código Java a Kotlin y viceversa.

Dispone de un editor visual muy potente e intuitivo, en el cual se pueden crear las distintas vistas simplemente arrastrando los componentes, además se pueden editar los atributos de cada uno de estos. Este editor tiene una vista de árbol en la que se muestran todos los componentes que hay en la interfaz actualmente. También se permite su edición mediante código XML.

Por otra parte, Android Studio compila el código en un APK<sup>14</sup> que se puede instalar y ejecutar en el emulador o en un dispositivo Android. Para esto se utiliza un sistema de construcción basado en Gradle, un paquete de herramientas de compilación avanzadas, para automatizar y flexibilizar este proceso.

### 5.1.2 GitLab

Esta herramienta se ha utilizado para el control de versiones, ya que es un sistema de alojamiento de repositorios para proyectos, gestionados por el sistema de versiones Git y de software libre. [17]

Se ha elegido esta opción frente a otras como GitHub, subversión, etc. Por el amplio conjunto de servicios que ofrece y sobre todo porque se ofrece como un software libre que se puede descargar e instalar en cualquier servidor, pudiéndolo utilizar en cualquier organización sin ningún coste adicional.

### 5.1.3 Inkscape

Es un software libre de vectores gráficos de código abierto, que utiliza SVG<sup>15</sup> como formato nativo. Dispone de una interfaz sencilla, con funcionalidades muy fáciles de comprender. Ofrece herramientas muy potentes para la edición de iconos y logos.

En este caso se ha utilizado para el diseño de los iconos, así como el logo principal de la aplicación.

## 5.2 Lenguajes de programación

En este apartado se va a hablar de los lenguajes utilizados para la implementación de la aplicación, Kotlin y XML.

Por un lado, Kotlin es un lenguaje de programación funcional pensado para funcionar con Máquina Virtual de Java (JVM<sup>16</sup>) y Android, se caracteriza por una perfecta combinación de características que mejoran la funcionalidad durante la programación, centrándose en la seguridad, la claridad y la interoperabilidad. [18]

Los programas Kotlin pueden utilizar *frameworks* y librerías de Java, y su interoperabilidad no requiere de capas de adaptación, por lo que es posible tener clases Kotlin y Java sin que surjan problemas.

Google declaró en 2017 a Kotlin como lenguaje oficial en Android, equiparándolo con Java y C++<sup>17</sup>.

Se ha elegido Kotlin para el desarrollo de la aplicación por las facilidades que ofrece frente a Java. Como, por ejemplo, la simplicidad a la hora de escribir código, ya que Kotlin al ser estáticamente tipado no necesita que se defina el tipo de las variables. También trata internamente los errores por valores nulos, ya que existe una versión *nullable* para todos los tipos, con solo añadir un signo de interrogación después del tipo, lo que permite inicializar esa variable a valor nulo evitando así que se lancen excepciones por esta razón.

Otra de las ventajas que ofrece es que permite reducir la cantidad de código. Además, no es necesario el uso de *getters* y *setters* ya que se generan automáticamente.

Para el desarrollo de la aplicación se ha seguido la guía oficial de codificación para Kotlin, en esta guía se detallan los estándares a seguir para la codificación de Android para el código fuente. [19]

Por otro lado, para el diseño de las interfaces de usuario, se ha utilizado, además del editor gráfico de Android Studio, el lenguaje XML, ya que para las modificaciones de algunos componentes es más sencillo realizarlo desde código. XML también se puede aplicar para el análisis de datos o la lectura de textos creados.

## 5.3 Android

Android es un sistema operativo pensado para dispositivos con pantalla táctil. Está basado en el Kernel de Linux. Este proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono de manera sencilla.

### 5.3.1 Características

En este apartado se van a describir brevemente las características principales del sistema operativo Android. [20]

- **Se adapta a muchas pantallas y resoluciones**

Android se ejecuta en dispositivos con diferentes tamaños y densidades de píxeles de pantalla, el sistema se encarga de adaptar las interfaces para cada uno de los dispositivos, realizando un escalado y modificación de tamaños básicos.

- **Utiliza SQLite para el almacenamiento de datos**

Android incluye una librería para almacenar datos de forma puntual en una base de datos SQLite, esto permite que el desarrollo sea más simple

- **Soporte de Java**

El código Java no se ejecuta, sino que se compila a un lenguaje intermedio, que se interpreta más adelante.

- **Soporte de diferentes formatos multimedia**

La plataforma Android soporta una gran variedad de formatos multimedia, los cuales pueden estar tanto codificados como decodificados. Algunos de los formatos son genéricos para todos los dispositivos Android y hay algunos que en función del modelo de móvil que se utilice puede añadir unos formatos nuevos. [21]

- **Soporte para hardware**

Android soporta distintos tipos de hardware como por ejemplo cámaras de fotos, de vídeo, GPS, sensores de proximidad y de presión entre otros.

- **Google Play**

Google Play es una tienda de aplicaciones gratuitas y de pago para Android donde se pueden descargar o actualizar aplicaciones para los dispositivos.

- **Multitarea real de aplicaciones**

Android permite ejecutar varias aplicaciones al mismo tiempo, gestionando las que no están en primer plano con ciclos de reloj.

### 5.3.2 Arquitectura

En este apartado se va a explicar la arquitectura de la plataforma explicando cada uno de sus componentes. [22]

- **Kernel de Linux**

Es la base de Android, este permite que se puedan utilizar funciones básicas del sistema, como pueden ser de seguridad o la gestión de memoria y procesos.



- **Capa de abstracción de hardware (HAL)**

Ofrece interfaces estándares para implementar funcionalidades sin afectar al resto de niveles de la arquitectura. Se basa en varios módulos de biblioteca, cada uno de ellos implementa una interfaz distinta. Al realizar una llamada al hardware del dispositivo, el sistema carga el módulo de biblioteca necesario

- **Tiempo de ejecución de Android (ART)**

A partir de la versión de Android 5.0, cada aplicación ejecuta sus propios procesos con sus propias instancias del tiempo de ejecución de Android. ART puede ejecutar varias máquinas virtuales en dispositivos con memoria baja, fue diseñado específicamente para Android y optimizado para ocupar un espacio de memoria mínimo. Sus principales funcionalidades son la compilación *ahead-of-time* (AOT<sup>18</sup>), *just-in-time* (JIT<sup>19</sup>) y la recolección optimizada de elementos no utilizados, estas funcionalidades mejoran la compatibilidad con la depuración.

- **Bibliotecas C/C++ nativas**

Android proporciona APIs de Java para exponer la funcionalidad de algunas bibliotecas nativas a las aplicaciones.

- **Framework de la API de Java**

Ofrece todas las funcionales de Android. Es la base para el desarrollo de aplicaciones, están escritas en Java. Entre estas funciones se incluyen:

- Sistema de vista para compilar la interfaz de usuario de una aplicación.
- Administrador de recursos que permite un acceso a recursos como cadenas de texto, gráficos y archivos de diseño.
- Administrador de notificaciones que permite mostrar alertar personalizadas en la barra de estado.
- Administrador de actividad para controlar el ciclo de vida de las aplicaciones.
- Proveedores de contenido que permite que las aplicaciones puedan aplicaciones o compartir sus propios datos.

- **Apps del sistema**

En Android se incluyen un conjunto de aplicaciones básicas para el usuario como el correo electrónico, mensajería SMS, calendarios, navegación en Internet y contactos, entre otras muchas.

## 5.4 Base de datos

En este apartado se va a hablar del modelo de datos utilizado para desarrollar la aplicación. En este caso se ha optado por una base de datos NoSQL, en particular una base de datos documental. Se va a explicar el porqué de su utilización y las opciones que se han barajado.

## 5.4.1 Firebase

En este apartado se va a hablar de Firebase, se trata de una plataforma digital que se utiliza para facilitar el desarrollo de aplicaciones web o móvil y cuyo principal objetivo es mejorar el rendimiento de las aplicaciones mediante la implementación de diversas funcionalidades, con las que se logra una aplicación más manejable, segura y de fácil acceso para los usuarios. [23]

Se ha decidido utilizar Firebase ya que permite una gestión de usuarios segura sin necesidad de tener que desarrollar servicios o módulos dentro de la aplicación, además permite la identificación de los usuarios con diferentes proveedores, por ejemplo, Google o Facebook. También ofrece un método de autenticación básica con usuario y contraseña, esto permite simplificar el código.

Firebase permite tener una visión de los datos en la nube y además poder subir ficheros, modificarlos sin necesidad de tener un servidor en el que tener que montar y desplegar la base de datos.

Algunos de los servicios que ofrece Firebase son:

- **Almacenamiento en la nube**, con Firebase Storage y Firebase Cloud Firestore se proporciona almacenamiento con carga y descarga de archivos y datos para aplicaciones, sin importar la calidad de la red.
- **Servicios de analítica**, Firebase Analytics es una aplicación gratuita que ofrece una visión del uso de la aplicación por parte de los usuarios.
- **Autenticación**, Firebase Auth es un servicio que permite autenticar a los usuarios utilizando únicamente código del lado del cliente. Incluyendo también la autenticación mediante proveedores como Google y Facebook. Este servicio permite facilitar la creación de sistemas de autenticación, así como mejorar la incorporación, acceso y seguridad de los usuarios. Además, permite otras funcionalidades como la recuperación y verificación de cuentas, todo gestionado mediante los servidores plataforma
- **Notificaciones push**, mediante Firebase Cloud Messaging se permite crear notificaciones push con la que tener una comunicación directa e instantánea con el usuario de la aplicación.

Dentro de Firebase hay disponibles dos soluciones de bases de datos en la nube y accesibles que admiten sincronización en tiempo real. Ambas soluciones son bases de datos NoSQL. [24]

Por un lado, Realtime Database es la base de datos original de Firebase, es una solución eficiente y de baja latencia para aplicaciones que necesitan estar sincronizados en tiempo real. Almacenando los datos en un árbol JSON.

Por otro lado, Cloud Firestore es un base de datos reciente en Firebase para el desarrollo de aplicaciones móviles. Coge los mejores aspectos de *Realtime Database* en un modelo datos nuevo y más intuitivo. Además, permite realizar consultas rápidas y el escalado tiene un nivel más alto que *Realtime Database*. Almacena los datos en subcolecciones dentro de documentos por lo que mejora la organización.



En este caso se ha optado por elegir la segunda opción, *Cloud Firestore*, ya que hace una recuperación, ordenación y filtrado de las consultas de una manera más rápida y ordenada. Además, tiene una disponibilidad mayor a nivel de regiones en todo el mundo y dispone de un escalamiento es automático.

## 5.5 Estructura interna de la aplicación

En este apartado se va a hablar de la estructura interna de la aplicación, se ha decido seguir una arquitectura a tres capas, este modelo de desarrollo software tiene como principal objetivo la separación de las partes que componen el sistema software.

Este modelo tiene como ventaja que el desarrollo se puede llevar a cabo en distintos niveles, y en el caso de que se necesite realizar algún cambio en algún nivel, solo afectará al nivel en el que se encuentra. Además, permite distribuir el trabajo de creación de una aplicación por niveles, proporcionando un nivel de abstracción entre cada grupo de trabajo.

A continuación, se va a describir con más detalle cada una de las capas.

### 5.5.1 Capa de persistencia

Es la capa que se encarga de gestionar los datos, aquí se encuentran las conexiones con las bases de datos o cualquier otro lugar donde se almacenen los datos.

En Android se suele usar una base de datos SQLite, ya que el SDK de Android incluye soporte. SQLite permite almacenar información persistente en dispositivos con capacidades hardware<sup>19</sup> limitadas. Una de las principales ventajas es que no necesita ser instalado, ni se debe controlar su ejecución ya que no funciona como un proceso independiente, sino que está totalmente integrada con la aplicación.

SQLite tiene también algunos inconvenientes, principalmente el riesgo que presenta escribir las sentencias SQL manualmente en el código y que estas no se validen durante la compilación. Esto puede provocar errores si en algún momento se modifica el modelo de datos.

Aunque existen librerías que abstraen el uso de SQLite, como Room, que facilita en gran medida la creación de consultas a la base de datos, finalmente se ha decido no usar esta tecnología debido a que la conexión con Firebase es rápida y estable y no se hacen una gran cantidad de peticiones a la base de datos, por lo que no vale la pena asumir los riesgos que conlleva usar SQLite en este caso.

Por otro lado, existen los objetos *SharedPreferences*. Son colecciones que permiten almacenar pequeñas cantidades de información del tipo clave-valor y que después es muy accesible gracias a las API que ofrece. Por ejemplo, se ha utilizado *SharedPreferences* para gestionar la casilla “Recuérdame” que aparece en el inicio de sesión y que permite al usuario cerrar y abrir la aplicación sin necesidad de tener que volver a introducir sus credenciales.

## 5.5.2 Capa de negocio

En esta capa se gestiona toda la lógica de la aplicación. En ella se hace toda la funcionalidad para tratar los datos que se encuentran en la capa de persistencia.

En esta capa se ha hecho uso de tres componentes básicos en Android, que son actividades, fragmentos y adaptadores.

Por un lado, una actividad es un componente que contiene la pantalla principal que ve el usuario, este componente puede contener listas, botones, campos de texto, imágenes, entre otros.

Por otro lado, un fragmento representa un comportamiento o una parte de la interfaz de usuario en una actividad, se pueden combinar varios fragmentos dentro de una misma actividad, así como utilizar el mismo fragmento para diferentes actividades. Estos fragmentos son independientes de la actividad, es decir, tienen su propio ciclo de vida, aunque sí que dependen de cuando se está ejecutando la actividad, ya que, si se detiene una actividad o se cierra, los fragmentos que la contienen también lo harán.

Por último, un adaptador es un mecanismo que hace de puente entre la vista de la interfaz de usuario y los datos de esa vista. Se han utilizado principalmente para rellenar las listas que hay en la aplicación.

La aplicación se compone de una actividad principal sobre la que se desarrollan todas las acciones. Esta actividad tiene un menú inferior desde el cual se abren cada uno de los fragmentos que van desplazándose sobre la misma actividad.

Además, se han utilizado actividades cuando, por ejemplo, se ha necesitado mostrar más detalle de algún objeto seleccionado. La interfaz que muestra el detalle del lugar seleccionado está implementada en una actividad a parte para facilitar la comunicación.

## 5.5.3 Capa de presentación

Es la capa que ve el usuario, también se conoce como capa de usuario o interfaz gráfica, captura la información que se va a mostrar al usuario. Debe ser intuitiva y homogénea. Esta capa se comunica con la capa de negocio.

El diseño final de las interfaces de usuario se puede ver en el punto seis de la memoria.

## 5.6 Patrones de diseño

Un patrón de diseño describe un problema que se repite en un entorno, además aporta información de cómo resolver el problema. Por lo que resuelven problemas generales en un contexto determinado.



Existe una clasificación de patrones según su propósito:

- De creación: Son patrones que intervienen en el proceso de creación de los objetos.
- De estructura: Estos patrones sirven para establecer una interconexión entre las distintas clases y objetos que forman una estructura, buscando una similitud entre ellas, para que, si se produce un cambio de los requisitos, se vean afectados los mínimos componentes posibles.
- De comportamiento: Especifican como se deben comportar entre ellos los diferentes objetos de un programa.

Para la implementación de la aplicación se ha utilizado el patrón observador, este patrón está clasificado dentro de los patrones de comportamiento, se ha utilizado para actualizar las listas en tiempo real.

En el objeto “sujeto” se mantiene una lista de dependencias, es decir, mantiene una lista con los objetos de los que depende, así si se produce un cambio de estado, el sujeto notifica a los demás objetos su nuevo estado.

## 5.7 Flujos

En este apartado se van a explicar algunos flujos de la aplicación mediante diagramas de secuencia, para ello se va a representar la interacción que hace el usuario con la aplicación y las acciones que realiza internamente el sistema.

En primer lugar, se va a explicar brevemente la notación UML utilizada para la llevar a cabo los diagramas de secuencia. Se han usado cinco objetos:

- **Usuario:** Es el actor que realiza la acción con la aplicación
- **IU:** “Interfaz de Usuario”. El usuario realiza la interacción directamente con este objeto.
- **:OC:** “Objeto de control”. Actúa como intermediario entre la Interfaz de usuario y Firebase, también realiza las peticiones a las APIs de Google.
- **Firestore Auth:** “Autenticación de Firestore”. Servicio de autenticación de usuarios ofrecido por Firestore.
- **Cloud Firestore:** Se encarga de almacenar los datos de la aplicación en Firestore.
- **Google API:** Servicios REST ofrecidos por Google donde se realizan las consultas sobre los lugares y las rutas.

En el flujo de la figura 28 se muestra el diagrama de secuencia de inicio de sesión mediante el método de usuario y contraseña. En este flujo se supone que el usuario ya se ha registrado previamente.

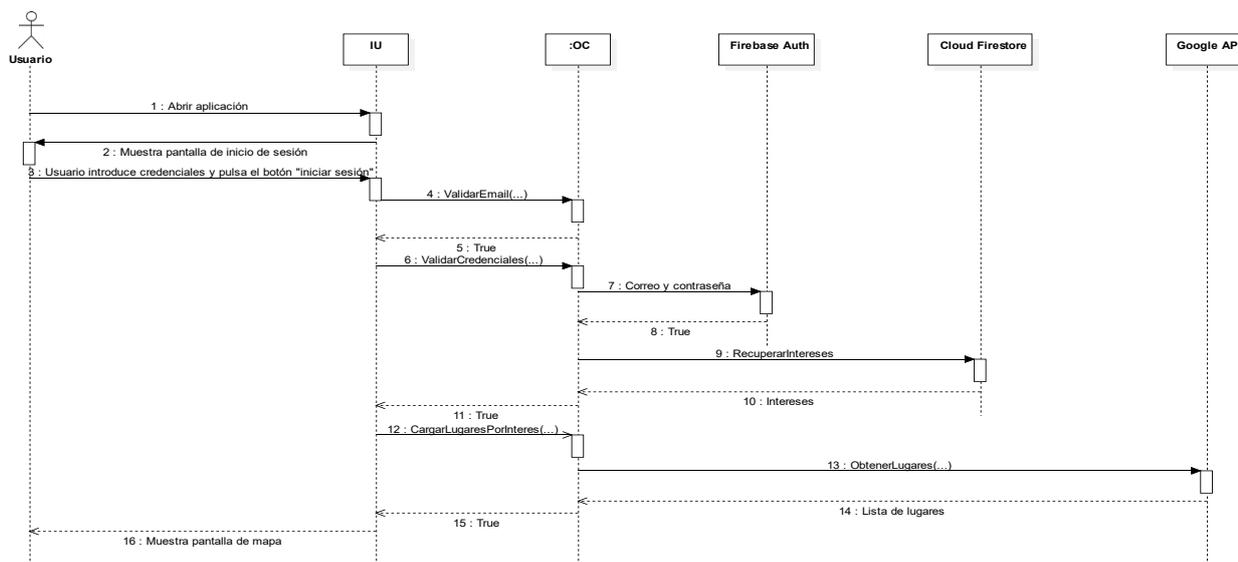


Figura 28 - Diagrama de secuencia iniciar sesión

En el diagrama se muestra como el usuario accede a la aplicación e introduce su correo y contraseña, al pulsar el botón de inicio de sesión, primero se comprueba que la estructura del correo electrónico introducido es correcta. Después se da paso a la comprobación de las credenciales contra *FirebaseAuth*.

Tras la comprobación de credenciales, se recuperan, desde la base de datos, los intereses seleccionados por el usuario.

Por último, se hace una petición a la API de Google para recuperar los lugares cercanos correspondientes a los intereses del usuario. Esto se hace mediante una tarea asíncrona, por lo que la aplicación sigue ejecutándose y cuando la API devuelve un resultado se muestran los lugares en el mapa mediante marcadores personalizados.

En el diagrama de la figura 29 se propone el flujo para el siguiente escenario:

Un usuario que ya ha iniciado sesión en la aplicación desea generar una ruta automáticamente entre los lugares de la zona que más le interesan. Tras generarla, se da cuenta de que hay otro lugar que le gustaría visitar y lo añade a la ruta actual. Además, añade ese lugar a su lista de favoritos para poder volver a visitarlo en otra ocasión.



## Aplicación móvil para la generación de rutas turísticas basadas en las preferencias del usuario

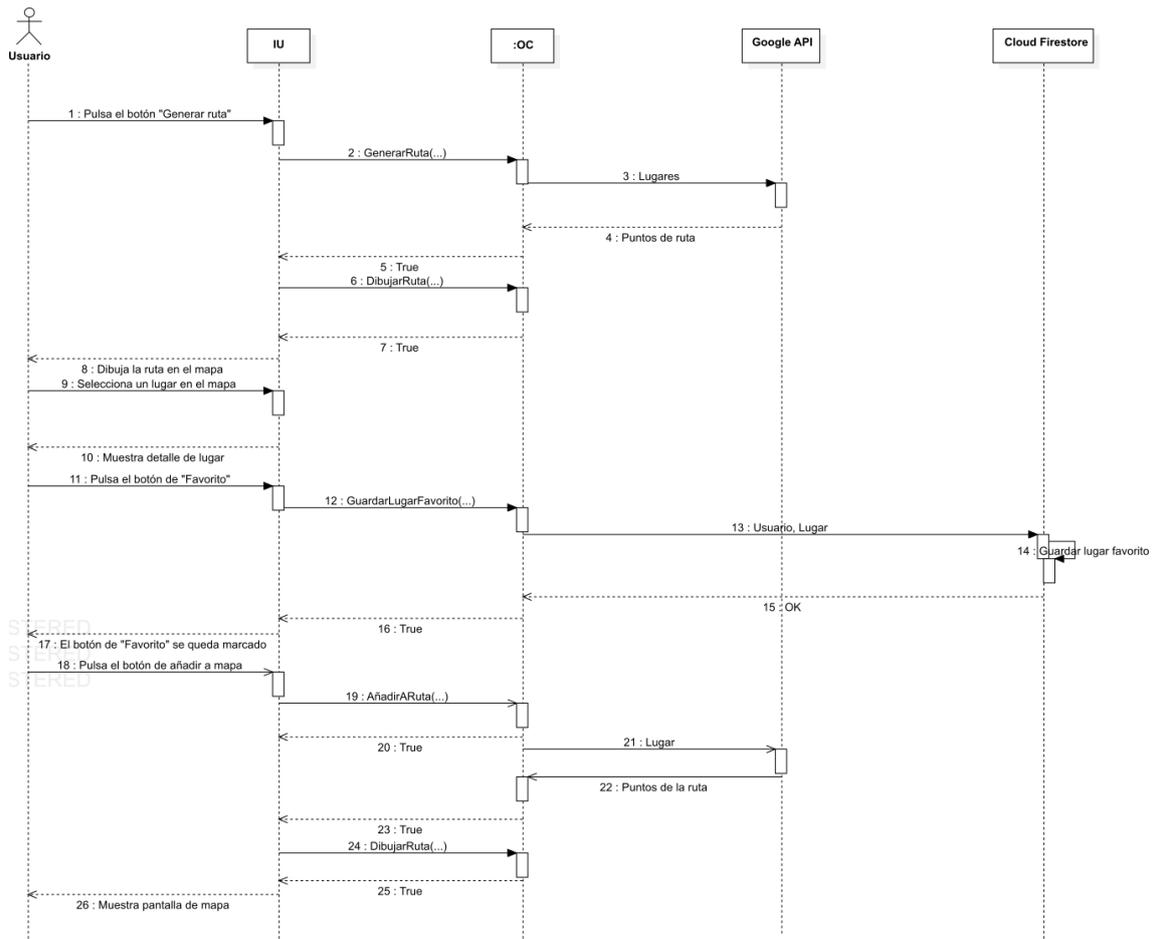


Figura 29 - Diagrama de secuencia generar ruta

En primer lugar, desde la vista de mapa, el usuario selecciona la opción de generar ruta, esto provoca que el sistema haga una petición a la API de Google, a la que le indica los lugares que se deben visitar, para obtener los distintos puntos que componen la ruta. Los lugares que se envían a la API ya se conocen desde el momento en que se ha iniciado sesión.

Cuando la API devuelve el resultado con los puntos de la ruta, se lanza la función que va a plasmar dicha ruta en el mapa.

Tras tener la ruta dibujada en el mapa, el usuario selecciona un lugar que no pertenece a la ruta actual y pulsa el botón correspondiente para añadirlo a sus lugares favoritos. Al hacerlo, se le pasan a la base de datos los identificadores del usuario y el lugar y esta se encarga de añadirlo a la lista de favoritos de dicho usuario. Cuando acaba este proceso, el botón de favoritos se resalta para indicar que ya se ha añadido.

Ahora el usuario quiere añadir el lugar a la ruta actual, para ello usa el botón correspondiente. Por defecto, el nuevo lugar se añade después del último punto de la ruta actual, por lo que simplemente se envía una petición a API para que devuelva los puntos de la ruta entre el último lugar de la ruta actual y el nuevo lugar añadido. Al devolverlo, se vuelve a dibujar la ruta y se muestra en el mapa.

Por último, en el diagrama de la figura 30 se muestra la interacción que hace el usuario modificar sus intereses. Para este flujo se supone que el usuario ha iniciado sesión previamente.

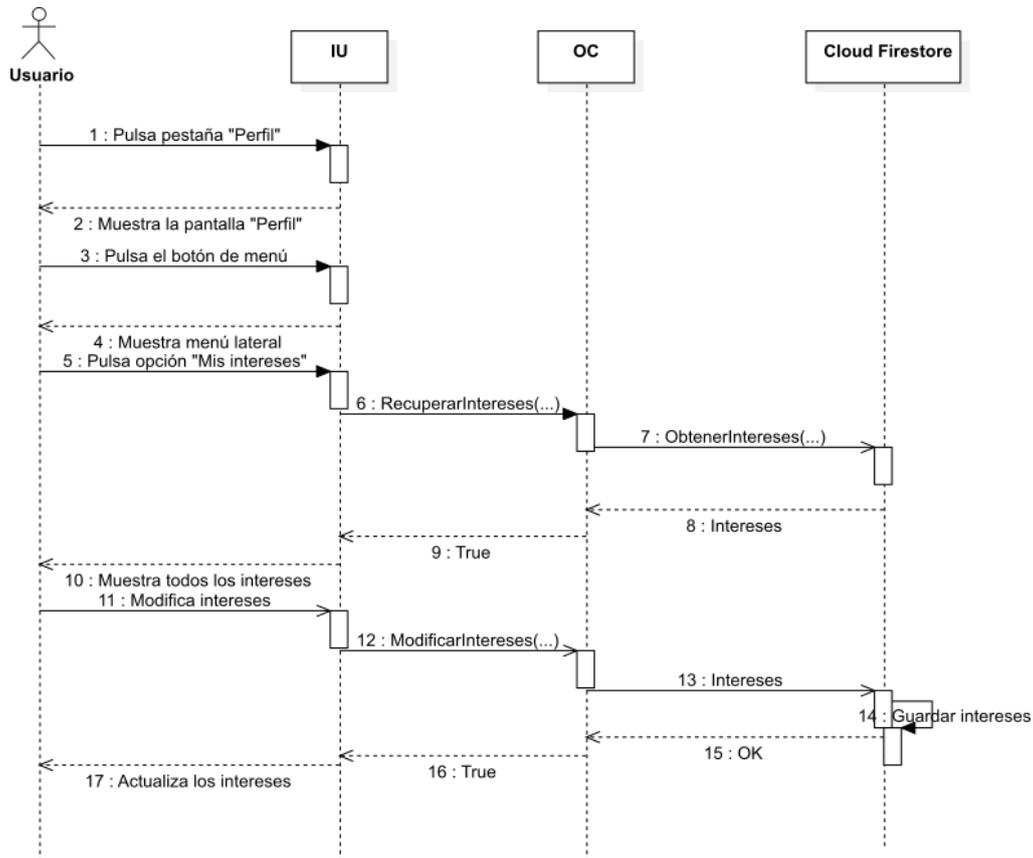


Figura 30 - Diagrama de secuencia modificar intereses

En primer lugar, el usuario accede a su perfil haciendo uso de la pestaña “Perfil” en el menú inferior, una vez en la pantalla de perfil, abre el menú desplegable y pulsa la opción “Mis intereses”.

Primero, se recuperan todos los intereses disponibles en la aplicación, ya que en la siguiente pantalla se muestran todos ellos y no solo los del usuario. Esto permite que el usuario pueda elegir nuevos intereses de entre los disponibles.

Se permite al usuario seleccionar y deseleccionar intereses y, al pulsar el botón de guardar, primero se comprueba que al menos haya tres intereses seleccionados y, tras esto, se envían los identificadores del usuario y dichos intereses a la base de datos, donde se almacenan.



# 6. Resultados

---

## 6.1 Resultado de la interfaz de usuario

En este apartado se va a explicar la funcionalidad de cada una de las pantallas que componen la aplicación.

### 6.1.1 Inicio de sesión



Figura 31 - Interfaz de inicio de sesión

En la figura 31 se muestra el diseño final de la interfaz de la pantalla de inicio de sesión, esta pantalla aparece al abrir la aplicación si no se ha iniciado sesión. El usuario tiene tres opciones para iniciar sesión, usando una cuenta creada desde la misma aplicación, o con su cuenta de Google o Facebook. En caso de entrar usando usuario y contraseña, se validan los campos y aparece un mensaje de error en caso de que algún dato sea incorrecto.

Debajo del formulario de inicio de sesión se pueden encontrar dos opciones, el botón de la izquierda sirve para mantener la sesión abierta si se cierra la aplicación. El de la derecha se utiliza para cambiar la contraseña en caso de que el usuario la haya olvidado, como se observa en la figura 32, al pulsarlo se abre un formulario en el que el usuario debe introducir un correo electrónico válido y, si existe en la base de datos, se envía un correo con un enlace en el que se puede establecer la nueva contraseña, como se puede ver en las figuras 33 y 34.

Por último, usando el botón de la parte inferior, se accede a la pantalla de registro de usuarios.

Aplicación móvil para la generación de rutas turísticas basadas en las preferencias del usuario



Figura 32 – Interfaz recuperar contraseña

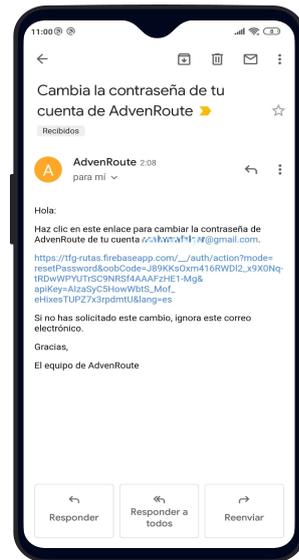


Figura 33 - Correo cambiar contraseña



Figura 34 - Cambio de contraseña

## 6.1.2 Registro

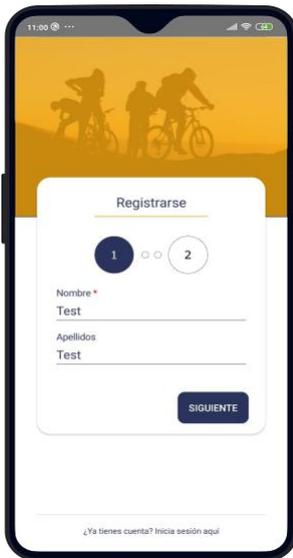


Figura 35 - Interfaz de registro I

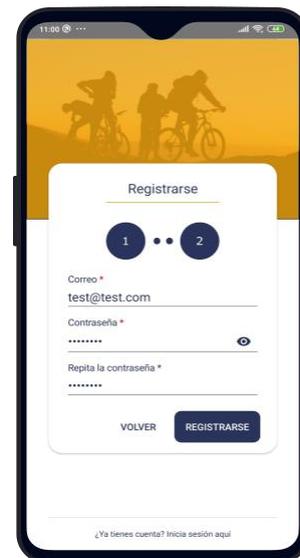


Figura 36 - Interfaz de registro II

El registro de usuarios se divide en dos partes, primero se deben introducir los datos personales, el nombre está marcado con un asterisco por ser un campo obligatorio. Al pulsar el botón “siguiente” aparece un nuevo formulario en el que se deben introducir los datos de acceso a la aplicación, es decir, correo electrónico y contraseña, teniendo que repetir esta para mayor seguridad. Como se puede observar, el campo de contraseña dispone de un botón que permite visualizarla. En este caso hay dos botones, uno para volver al formulario anterior y otro para finalizar el registro.

En la parte inferior hay un botón que permite volver a la pantalla de inicio de sesión.

## 6.1.3 Intereses

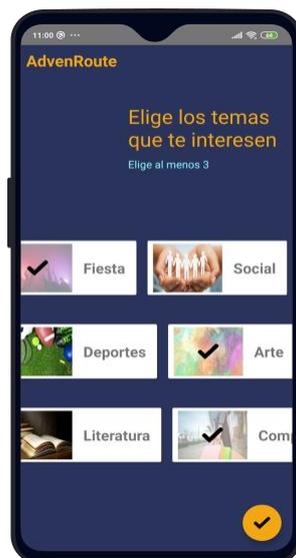


Figura 37 - Interfaz elegir intereses

Tras finalizar el registro, el usuario debe elegir sus intereses, para ello, en la pantalla se muestra una lista en formato horizontal donde debe seleccionar las categorías que desee. Como mínimo, se deben seleccionar tres intereses, de lo contrario no es posible avanzar ya que hasta que esto no se cumple, no aparece el botón para avanzar a la pantalla principal de la aplicación.

### 6.1.4 Menú

Esta es la pantalla principal de la aplicación, cada elemento del menú corresponde a una de las funciones que proporciona, pero el menú es común a todos ellos y la barra superior cambia dinámicamente mostrando las opciones adecuadas para cada pantalla.

#### 6.1.4.1 Mapa



Figura 38 - Interfaz pestaña "Mapa"

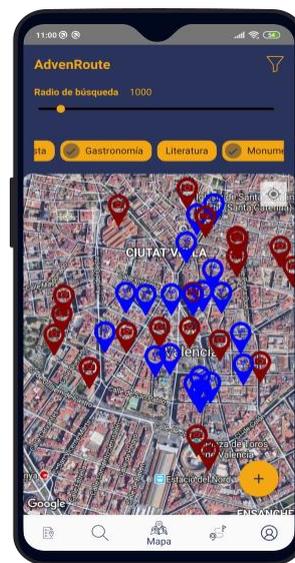


Figura 39 - Filtros pestaña "Mapa"

Al entrar a la aplicación, se muestra esta pantalla por defecto, en ella se puede observar un mapa en el que se indica, mediante marcadores personalizados para cada interés, los lugares correspondientes a los intereses que ha escogido el usuario al registrarse.

En el mapa se pueden observar diversos botones, en la parte superior derecha hay un botón que permite centrar la vista del mapa en la posición actual del usuario. En la parte inferior, hay un botón flotante el cual, muestra tres opciones al pulsarlo. Estas opciones son, de arriba a abajo, generación de ruta automática recorriendo los puntos de mayor interés para el usuario, eliminar la ruta actual del mapa y guardar la ruta actual, para poder acceder a ella más tarde y editarla o repetirla desde la opción del menú "Mis rutas".

En la parte superior de la pantalla, hay un botón que, al pulsarlo, provoca que el mapa se deslice hacia abajo mostrando los filtros disponibles. Como se puede observar en la figura 39 se puede filtrar por distancia, mostrando lugares dentro del rango

indicado. Y además se puede filtrar por intereses, permitiendo así al usuario encontrar lugares que estén fuera de sus preferencias iniciales.

Al pulsar en cada marcador, se abre un dialogo en el que se muestran algunos detalles del lugar, como el nombre o la dirección, así como algunas fotos entre las que se puede navegar, así como ampliar al mantener pulsadas. Desde aquí, también es posible ver la puntuación actual del lugar seleccionado. Si se presiona encima del nombre del lugar, se accede a una pantalla que muestra más detalles sobre este, como se puede ver en la figura 40. Por último, con el botón de la derecha, se puede añadir el lugar a la ruta actual.



Figura 40 - Detalle de lugar en la pestaña "Mapa"

### 6.1.4.2 Lugares

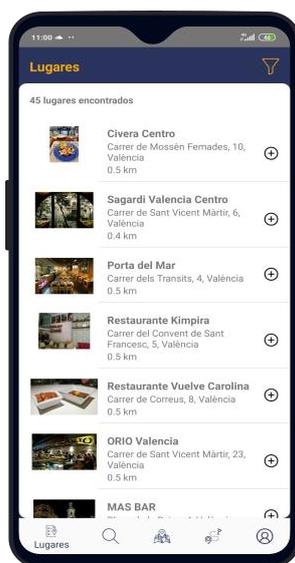


Figura 41 - Interfaz pestaña "Lugares"

En esta pantalla se muestra, al igual que en el mapa, los lugares cercanos interesantes para el usuario, pero esta vez en formato de lista. En cada elemento de la

lista, se muestra una foto del lugar, el nombre y la dirección, así como la distancia a la que se encuentra el usuario actualmente. Desde el botón de la derecha se puede añadir un lugar a la ruta actual.

En la parte superior de la lista se muestra un contador de los lugares que se han encontrado. Los filtros se muestran de la misma manera que en la pantalla de mapa y contienen las mismas opciones.

Por último, al pulsar cada elemento, se abre una pantalla en la que se muestran más detalles del lugar.

### 6.1.4.3 Descubrir

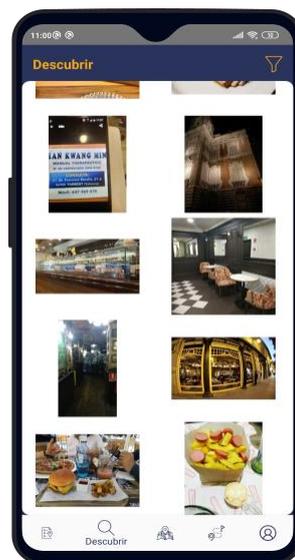


Figura 42 - Interfaz pestaña "Descubrir"

Esta pantalla está pensada para que el usuario encuentre nuevos lugares, que puedan ser o no de su interés. Es decir, se muestra una lista de fotografías de todos los lugares cercanos a la ubicación del usuario, este puede pulsar en las imágenes y ver el lugar con más detalle.

En esta pantalla no es posible filtrar por categoría ya que la idea es que el usuario descubra lugares nuevos sin importar de que tipo sean. Por tanto, solo es posible filtrar por distancia.

### 6.1.4.4 Mis rutas

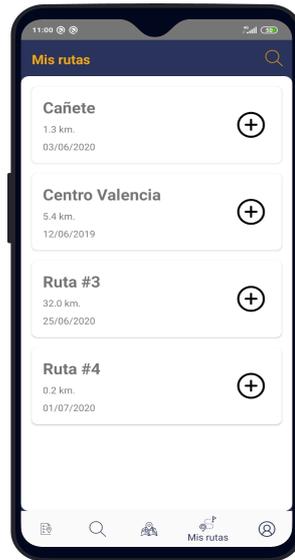


Figura 43 - Interfaz pestaña "Mis rutas"

En este apartado se pueden ver las rutas guardadas previamente por el usuario en forma de lista. Cada elemento muestra el nombre de la ruta, la longitud de esta y la fecha en la que se guardó. Al pulsar el botón de la derecha, la ruta se dibuja en el mapa para poder repetirla.

Al pulsar en cada elemento de la lista, se accede a una pantalla donde se muestran los lugares que componen la ruta, así como la distancia que hay entre cada uno de ellos, como se puede ver en la figura 47.

En la parte superior de la pantalla, se puede observar que el menú ha cambiado ya que en esta pantalla no existen filtros, sino que es posible realizar una búsqueda por nombre de la ruta.

### 6.1.4.5 Perfil

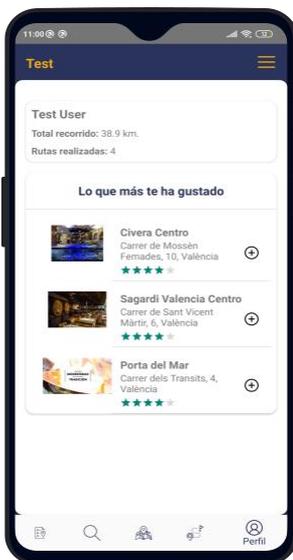


Figura 44 - Interfaz pestaña "Perfil"

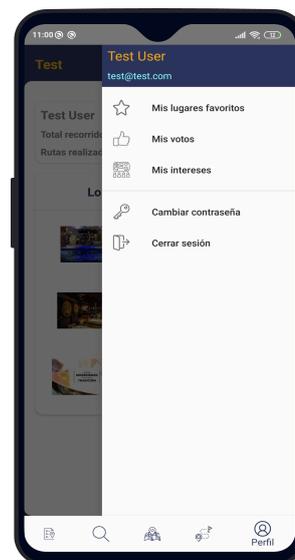


Figura 45 - Menú lateral

En esta pantalla se muestra la información del usuario. En la parte superior se puede ver los datos personales del usuario y algunas estadísticas de uso de la aplicación como el número de rutas realizadas o los kilómetros recorridos. El usuario puede editar sus datos personales desde aquí.

Abajo se muestran algunos de los lugares que más han gustado al usuario, es decir, que mejor puntuación han recibido.

En la parte superior derecha, hay un botón que muestra un menú lateral en el que se hay varias opciones adicionales, como, por ejemplo, ver los lugares guardados como favorito, ver y editar los intereses, entre otros. Desde este menú también se permite al usuario cerrar sesión.

### 6.1.5 Detalles de lugar

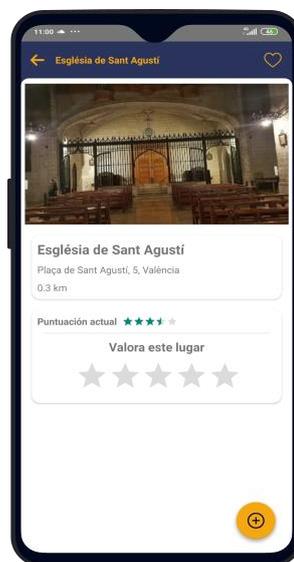


Figura 46 - Interfaz detalles de lugar

A esta pantalla se puede acceder desde varias partes de la aplicación, ya que siempre que se muestran los datos de algún lugar, existe la opción de apretar en estos para acceder a una vista más detallada del mismo. Por ejemplo, es posible acceder desde la pestaña “Lugares” o “Descubrir”, así como apretando en la información que aparece al seleccionar un marcador en el mapa.

En esta pantalla se pueden ver todas las imágenes disponibles del lugar, así como datos más detallados como la puntuación actual.

También se ofrecen más interacciones con el lugar, ya que desde esta pantalla es posible añadir el lugar a la lista de favoritos usando el botón situado en la parte superior derecha de la pantalla o la posibilidad de valorar el lugar haciendo uso de las estrellas de la parte inferior. Además, también es posible añadirlo a la ruta actual desde aquí.

## 6.1.6 Detalles de ruta

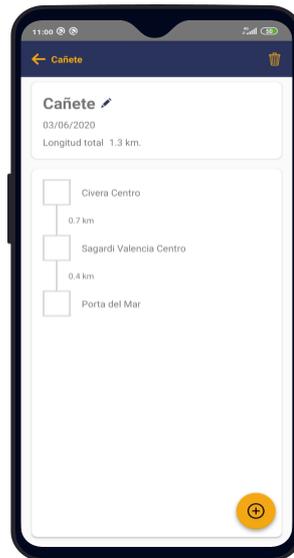


Figura 47 – Interfaz detalle de ruta

A esta pantalla se accede al seleccionar una de las rutas guardadas por el usuario. Aquí se pueden ver y editar algunos datos de esta como el nombre, la fecha en la que se realizó o la distancia total de la misma.

Además, se ofrece una vista detallada de los lugares que la componen, así como de la distancia que hay entre ellos. Desde aquí, también se puede acceder a una vista más detallada de cada lugar.

Una ruta guardada se puede eliminar en cualquier momento usando el botón situado en la parte superior derecha de la pantalla. También existe la posibilidad de repetir la ruta utilizando el botón de la parte inferior.

## 6.1.7 Mis lugares favoritos

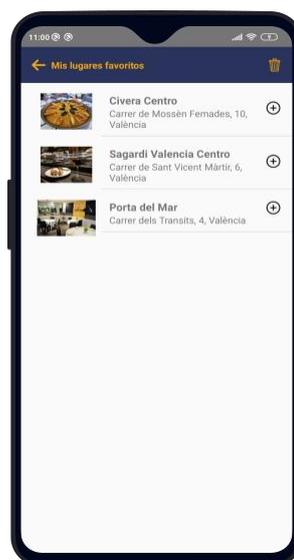


Figura 48 – Interfaz pantalla “Mis lugares favoritos”

## Aplicación móvil para la generación de rutas turísticas basadas en las preferencias del usuario

En esta pantalla se muestra una lista de los lugares que el usuario ha marcado como favoritos. Esta lista es idéntica a la que se muestra en la pestaña “Lugares” del menú principal y la funcionalidad es la misma.

Para eliminar un lugar de favoritos, es necesario entrar a la vista de detalle y desmarcar el botón situado en la parte superior derecha.

### 6.1.8 Mis votos

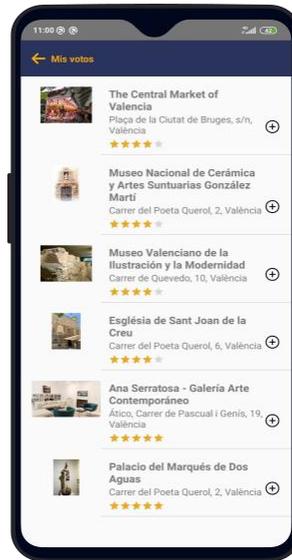


Figura 49 – Interfaz pantalla “Mis votos”

Al igual que en el apartado anterior, se muestra una lista con todos los lugares que el usuario ha valorado, indicando la puntuación. No es posible editar la puntuación de un lugar una vez votado.

### 6.1.9 Mis intereses

En esta pantalla se muestran todos los intereses disponibles en la aplicación. Desde aquí, el usuario puede modificar los intereses que seleccionó al registrarse en la aplicación. No es posible seleccionar menos de tres intereses en ningún caso. La interfaz que se utiliza para esta pestaña se muestra en la figura 37.

## 6.1.10 Cambiar contraseña



*Figura 50 – Interfaz pantalla “Cambiar contraseña”*

En esta pantalla el usuario puede cambiar su contraseña, para ello tiene que introducir su contraseña actual como se puede ver en la figura 50.

## 6.2 Diagrama de navegación

En este apartado se muestra el diagrama de navegación final de la aplicación, este diagrama se ha dividido en dos partes, en la primera se muestra la autenticación de usuarios y la elección de intereses. En la segunda, se muestran el resto de las pantallas de la aplicación.

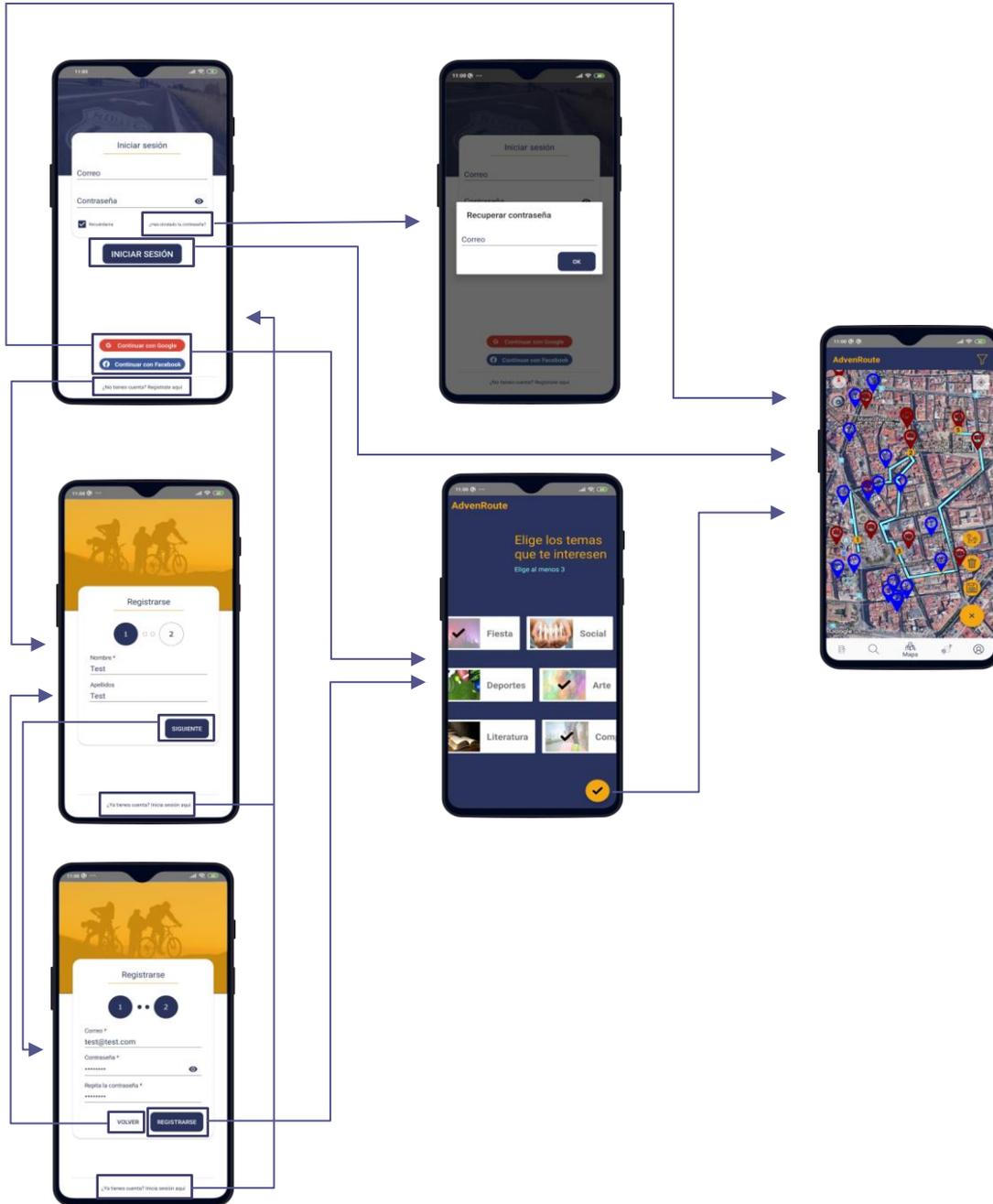


Figura 51 - Diagrama de navegación I



Figura 52 - Diagrama de navegación II



## 6.3 Pruebas

En este apartado se va a explicar el tipo de pruebas realizadas en la aplicación, como se han llevado a cabo y los resultados obtenidos.

Durante el desarrollo de la aplicación se han realizado pruebas de usabilidad probadas en distintos dispositivos, cada vez que se ha finalizado un sprint. Una vez concluidas se han realizado las modificaciones oportunas si se han encontrado fallos.

Para las pruebas finales, se ha compartido la aplicación con un conjunto de usuarios, además se les ha entregado una encuesta de usabilidad que han rellenado después de probar la aplicación.

Para la realización del cuestionario mostrado en el Anexo I se han seguido las directrices propuestas por el SUS “*System Usability Scale*”, estos cuestionarios se caracterizan por ser fáciles de completar y puntuar, permitiendo así que el usuario no tenga que pensar mucho en la respuesta de cada punto. Se realizan después de que el usuario pruebe el software. Este tipo de prueba no tiene en cuenta las respuestas individualmente, sino que busca una puntuación global.

Además, se han añadido algunas preguntas adicionales para obtener información más detallada a cerca de errores más concretos que podrían aparecer al usar la aplicación. También se han solicitado algunos datos personales a los usuarios con el fin de realizar estadísticas.

Los resultados obtenidos en el cuestionario SUS son los siguientes:

Tabla 43 - Resultados

	Usuario 1	Usuario 2	Usuario 3	Usuario 4	Usuario 5	Usuario 6	Usuario 7
Edad	22	24	22	56	60	32	22
Pregunta 1	4	5	3	3	2	5	5
Pregunta 2	1	1	1	4	2	1	1
Pregunta 3	5	5	5	4	3	4	5
Pregunta 4	1	1	1	2	1	1	1
Pregunta 5	4	4	4	4	5	5	4
Pregunta 6	1	1	1	1	1	1	1
Pregunta 7	5	4	5	4	4	5	5
Pregunta 8	1	1	1	2	2	1	1
Pregunta 9	5	5	5	3	3	5	5
Pregunta 10	1	1	1	1	3	1	1

Cada pregunta debe contestarse siguiendo una escala de valores entre uno y cinco, siendo uno muy en desacuerdo y cinco muy de acuerdo.

En la tabla 43 se representan las respuestas de cada usuario antes de hacer los cálculos necesarios para obtener el resultado SUS.

Para calcular los resultados del cuestionario se han seguido las normas indicadas para los cuestionarios SUS. Hay que restar uno al resultado obtenido en las preguntas impares y restarle a cinco el resultado de las preguntas pares. Por ejemplo, para el usuario número uno:

$$(4-1)+(5-1)+(5-1)+(5-1)+(4-1)+(5-1)+(5-1)+(5-1)+(5-1)-(5-1) = 38$$

Este resultado debe multiplicarse por 2,5, obteniendo así, para el usuario número uno, una puntuación SUS de **95**.

En la siguiente tabla, se muestran los resultados con el cálculo aplicado. Como se puede observar, se obtiene un resultado promedio de **88,21**. Según la escala, todo software con un resultado mayor de 68 se puede considerar usable.

Tabla 44 - Cálculos de los resultados

	Usuario 1	Usuario 2	Usuario 3	Usuario 4	Usuario 5	Usuario 6	Usuario 7	
Edad	22	24	22	56	60	32	22	
Pregunta 1	3	4	2	2	1	4	4	
Pregunta 2	4	4	4	1	3	4	4	
Pregunta 3	4	4	4	3	2	3	4	
Pregunta 4	4	4	4	3	4	4	4	
Pregunta 5	3	3	3	3	4	4	3	
Pregunta 6	4	4	4	4	4	4	4	
Pregunta 7	4	3	4	3	3	4	4	
Pregunta 8	4	4	4	3	3	4	4	
Pregunta 9	4	4	4	2	2	4	4	
Pregunta 10	4	4	4	4	2	4	4	
	95	95	92,5	70	70	97,5	97,5	88,21

Como se ha indicado anteriormente, se han pedido a los usuarios algunos datos personales para poder obtener estadísticas. Como se puede observar en la tabla 44, para los usuarios número cuatro y número cinco, los cuales tienen una edad mayor que el resto, se ha obtenido un resultado por debajo de la media, aunque aun así han superado el umbral de usabilidad de 68 que establece la SUS. Por tanto, se llega a la conclusión de que la aplicación es menos usable para usuarios de mayor edad.



Por último, en la tercera sección del cuestionario, donde se realizan preguntas más concretas, se han obtenido los siguientes resultados:

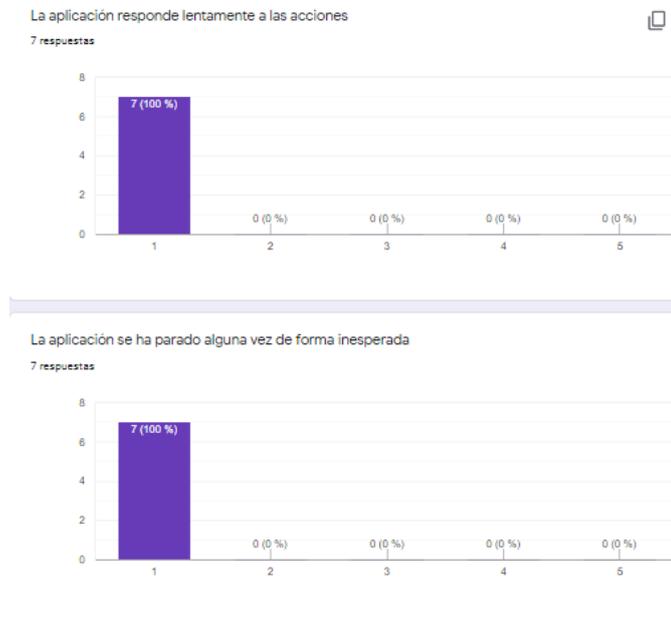


Figura 53 - Resultados cuestionario sección 3.1

Como se puede observar en la figura 53, no ha habido problemas de cierres inesperados de la aplicación. Además, el 100% de los usuarios afirman que no existen tiempos excesivos de carga y que la aplicación responde correctamente a sus acciones.

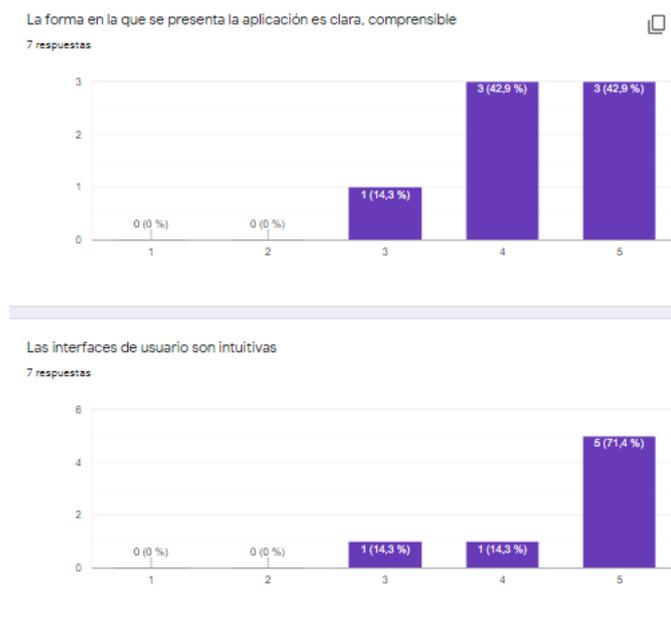


Figura 54 - Resultados cuestionario sección 3.2

En cuanto a las interfaces de usuario, la mayoría de los usuarios coinciden en que la información se presenta de una forma clara y los elementos de la interfaz son comprensibles y fáciles de utilizar.

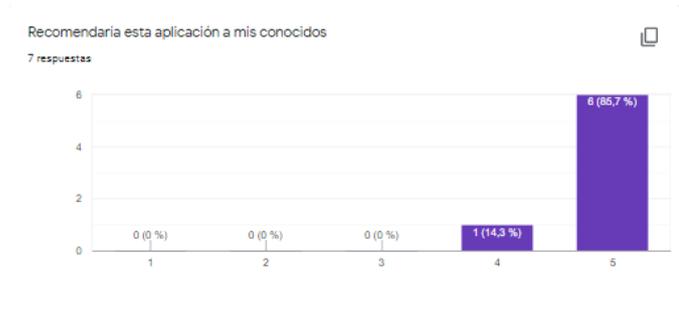


Figura 55 - Resultados cuestionario sección 3.3

Se ha querido saber si los usuarios recomendarían la aplicación, ya que la totalidad de ellos han indicado que sí, se entiende que la aplicación ha resultado útil.

¿Qué te ha resultado más útil de la aplicación y por qué?

7 respuestas

- Poder generar rutas de forma automáticas pulsando únicamente un botón.
- Poder crear una ruta eligiendo los lugares que me gustan, además he podido descubrir nuevos lugares que desconocía de mi ciudad
- He podido generar una ruta por mi localidad
- Poder buscar nuevos lugares que me gustan
- Hacer una ruta para no tener que buscar los lugares que hay
- Poder generar una ruta automáticamente
- Poder buscar lugares de una categoría específica ya que he descubierto algunos lugares de Valencia que no conocía

Figura 56 - Resultados cuestionario sección 3.4

¿Que crees que se debería mejorar en este software y por qué?

7 respuestas

- Me gustaría que se pudieran añadir nuevos lugares que no aparecen en la aplicación, pero que si que creo que pueden ser de interés para otros usuarios.
- Creo que sería muy útil poder compartir mis rutas y lugares favoritos con otros usuarios, ya que normalmente realizo viajes con amigos y nos ayudaría a decidirnos.
- Poder añadir nuevos lugares a la aplicación
- Creo que correcto
- Me gustaría que se pudiera añadir al coche para poder seguir la ruta desde ahí
- Me gustaría poder buscar en una ubicación en la que no estoy actualmente para poder programar una ruta que vaya a realizar posteriormente
- Creo que es correcta

Figura 57 - Resultados cuestionario sección 3.5

Las respuestas obtenidas en las figuras 56 y 57 se han utilizado para realizar un análisis de requisitos futuros, tanto para mejorar las funciones existentes que son más utilizadas como para añadir nuevas funcionalidades que mejoren la aplicación y permitan llegar a más usuarios.

# 7. Conclusión

---

El objetivo principal de este TFG era desarrollar una aplicación que permitiera al usuario poder crear rutas turísticas de una forma sencilla, encontrar nuevos lugares y además ayudar a mejorar la situación en la que se encuentran muchos lugares despoblados y aislados en los que el turismo no está muy desarrollado.

Tras finalizar el desarrollo, concluir y analizar las pruebas finales de los usuarios, se puede afirmar que se han cubierto todos los objetivos marcados inicialmente.

En cuanto a la implementación, el inicio del desarrollo fue muy costoso ya que no se conocían muchas de las tecnologías utilizadas, gracias a la búsqueda de información se han adquirido nuevos conocimientos y se han consolidado las nociones básicas que se tenían de los lenguajes Kotlin y XML.

Para finalizar, se va a hablar del aprendizaje que ha supuesto la realización de este proyecto. Se han adquirido nuevos conocimientos tanto en el ámbito del desarrollo del software como en la gestión de proyectos, ya que ha sido una experiencia nueva tener que gestionar todas las fases de un proyecto de mayor envergadura que los que se han realizado durante el grado.

## 7.1 Trabajo futuro

En este apartado se van a mencionar algunas de las funcionalidades se pretende implementar en futuras versiones de la aplicación, para ello se ha realizado, además de un estudio sobre que opciones sería interesante añadir, una toma de requisitos en base a los resultados obtenidos en las encuestas realizadas por los usuarios finales.

- Compartir rutas con otros usuarios.
- Añadir opiniones sobre los lugares.
- Implementar una versión para Android Auto<sup>20</sup>.
- Implementación de notificaciones cuando el usuario se aproxime a un lugar que pueda ser interesante para él.
- Añadir un apartado de configuración que permita personalizar la aplicación al gusto del usuario.

## 8. Bibliografía

---

- [1] «proyectosagiles.org,» [En línea]. Available: <https://proyectosagiles.org/que-es-scrum/>. [Último acceso: 08 Enero 2020].
- [2] Universidad de Salamanca, «CÁTEDRA VIEWNEXT USAL,» 12 Febrero 2018. [En línea]. Available: <https://viewnext.usal.es/blog/herramienta-%C3%A1gil-taigaio>. [Último acceso: 03 Enero 2020].
- [3] Google, «Google Play,» [En línea]. Available: [https://play.google.com/store/apps/details?id=com.mapamai.maps.batchgeocode&hl=es\\_419](https://play.google.com/store/apps/details?id=com.mapamai.maps.batchgeocode&hl=es_419). [Último acceso: Diciembre 2019].
- [4] Google, «Google Play,» [En línea]. Available: <https://play.google.com/store/apps/details?id=com.calimoto.calimoto&hl=es>. [Último acceso: Diciembre 2019].
- [5] «My Phone Tour,» 2020. [En línea]. Available: <https://myphonetour.com/>. [Último acceso: Diciembre 2019].
- [6] «MyRoute-app,» [En línea]. Available: <https://www.myrouteapp.com/es>. [Último acceso: Diciembre 2019].
- [7] Google, «Google Play,» [En línea]. Available: [https://play.google.com/store/apps/details?id=com.triposo.droidguide.spain&hl=en\\_US](https://play.google.com/store/apps/details?id=com.triposo.droidguide.spain&hl=en_US). [Último acceso: Diciembre 2019].
- [8] Google, «Google Cloud,» 16 Diciembre 2019. [En línea]. Available: <https://cloud.google.com/solutions/mobile/mobile-app-backend-services>. [Último acceso: 21 Diciembre 2019].
- [9] Google, «Google Maps Platform,» 05 Junio 2020. [En línea]. Available: <https://developers.google.com/maps/documentation/directions/intro>. [Último acceso: 12 Junio 2020].
- [10] Google, «Google Maps Platform,» 6 Mayo 2020. [En línea]. Available: <https://developers.google.com/maps/documentation/distance-matrix/intro?hl=es-419>. [Último acceso: 07 Mayo 2020].
- [11] Google, «Google Maps Platform,» 04 Junio 2020. [En línea]. Available: <https://developers.google.com/maps/documentation/android-sdk/intro>. [Último acceso: 12 Junio 2020].
- [12] Google, «Google Maps Platform,» 12 Junio 2020. [En línea]. Available: <https://developers.google.com/places/web->

- service/intro?hl=es&utm\_source=google&utm\_medium=cpc&utm\_campaign=F  
Y18-Q2-global-demandgen-paidsearchonnetworkhouseads-cs-  
maps\_contactsal\_saf&utm\_content=text-ad-none-none-DEV\_c-  
CRE\_436338821424-ADGP\_Hybrid+%7C+AW+. [Último acceso: 12 Junio  
2020].
- [13] «Developers Android,» 22 Mayo 2020. [En línea]. Available:  
<https://developer.android.com/topic/libraries/data-binding?hl=es>. [Último  
acceso: 22 Junio 2020].
- [14] «Material Design,» [En línea]. Available: <https://material.io/>. [Último  
acceso: 05 Junio 2020].
- [15] Universidad Politécnica de Valencia, «Master en Desarrollo de  
Aplicaciones Android,» 2017. [En línea]. Available:  
<http://www.androidcurso.com/index.php/688>. [Último acceso: Febrero 2020].
- [16] «Developers Android,» 01 Mayo 2020. [En línea]. Available:  
<https://developer.android.com/studio/intro>. [Último acceso: 02 Junio 2020].
- [17] J. Torrado, «desarrolloweb.com,» 12 Octubre 2017. [En línea]. Available:  
<https://desarrolloweb.com/articulos/introduccion-gitlab.html>. [Último acceso:  
Enero 2020].
- [18] E. Geek, «if geek then,» 15 octubre 2019. [En línea]. Available:  
[https://ifgeekthen.everis.com/es/que-es-kotlin-y-por-que-deberias-empezar-  
aprenderlo-ya](https://ifgeekthen.everis.com/es/que-es-kotlin-y-por-que-deberias-empezar-aprenderlo-ya). [Último acceso: Febrero 2020].
- [19] «Developers Android,» 18 05 2018. [En línea]. Available:  
<https://developer.android.com/kotlin/style-guide>.
- [20] «SISTEMA ANDROID,» [En línea]. Available:  
<https://scoello12.wordpress.com/caracteristicas/>. [Último acceso: Marzo 2020].
- [21] Universidad Politecnica de Valencia, «Master en Desarrollo de  
Aplicaciones Android,» 2017. [En línea]. Available:  
<http://www.androidcurso.com/index.php/164>. [Último acceso: 15 Junio 2020].
- [22] «android developers,» 07 Mayo 2020. [En línea]. Available:  
<https://developer.android.com/guide/platform?hl=es-419>. [Último acceso: 18  
Mayo 2020].
- [23] V. Giraldo, «blog,» 19 Agosto 2019. [En línea]. Available:  
<https://rockcontent.com/es/blog/que-es-firebase/>. [Último acceso: Diciembre  
2019].

- [24] «Firebase,» 03 Diciembre 2019. [En línea]. Available: <https://firebase.google.com/docs/firestore/rtdb-vs-firestore?hl=es-419>. [Último acceso: 15 Diciembre 2019].
- [25] «IntelliJ IDEA,» [En línea]. Available: <https://www.jetbrains.com/idea/>.

## Glosario

1. **IDE:** Entorno de desarrollo integrado, es una aplicación que proporciona herramientas para facilitar al desarrollador la tarea de programar.
2. **Geolocalización:** Es la capacidad para obtener la ubicación geográfica real de un objeto.
3. **Backend:** Parte de un software que se encarga de toda la lógica, por ejemplo, la comunicación con un servidor.
4. **Frontend:** Parte de un software que interactúa con el usuario.
5. **API:** *Application Programming Interfaces. Interfaz que permite acceder a la funcionalidad de un software.*
6. **HTTP:** *Hypertext Transfer Protocol.* Protocolo de comunicación para transferencia en internet.
7. **SDK:** *Software Development Kit.* Es un paquete de herramientas y datos que permite a los desarrolladores programar en un lenguaje concreto.
8. **REST:** *Representational State Transfer.* Arquitectura utilizada para enviar o recibir datos mediante peticiones HTTP.
9. **JSON:** *JavaScript Object Notation. Es un formato de texto para el intercambio de datos.*
10. **Java:** Lenguaje de programación orientado a objetos cuya sintaxis deriva de C y C++.
11. **Framework:** Conjunto de conceptos y prácticas orientados a solucionar un problema particular.
12. **NoSQL:** Hace referencia a sistemas de bases de datos no relacionales. Es decir, los datos no tienen porqué estar relacionados entre sí ni almacenarse en tablas.
13. **IntelliJ IDEA:** Es un entorno de desarrollo integrado para el desarrollo de programas informáticos.
14. **APK:** *Android Application Package.* Es un archivo ejecutable de aplicación para Android.
15. **SVG:** *Scalable Vector Graphics.* Formato de gráficos vectoriales para representar imágenes.
16. **JVM:** *Java Virtual Machine.* Ejecuta instrucciones en código binario generado por el compilador de Java
17. **C++:** Lenguaje de programación cuyo objetivo es extender el lenguaje C con funciones que permiten la manipulación de objetos.
18. **AOT:** *Ahead-of-time.* Compilación antes de la ejecución de un programa.
19. **JIT:** *Just-in-time.* Compila el código a medida que lo va ejecutando.
20. **Android Auto:** Es una interfaz desarrollada por Google, que adapta la pantalla de nuestro teléfono para poder ser usada en el coche.



# Anexo I

---

En este anexo se muestra el cuestionario realizado para hacer las pruebas finales de usabilidad de la aplicación.

Este cuestionario se ha dividido en tres secciones:

- Primera sección: se piden al usuario algunos datos personales. Estos han sido utilizados para generar estadísticas de uso de la aplicación. También se pide introducir, si se conoce, la versión de Android en que se ha probado para saber hacia que versiones de Android se deberían orientar las futuras versiones de la aplicación. Todos estos datos no son obligatorios.

The image shows a mobile application survey form with three distinct sections, each with a light purple header bar. The first section is titled 'Edad' and contains a text input field labeled 'Tu respuesta'. The second section is titled 'Sexo' and contains four radio button options: 'Mujer', 'Hombre', 'Prefiero no decirlo', and 'Otro:'. The 'Otro:' option is followed by a text input field. The third section is titled 'Versión de Android con la que probaste la aplicación' and contains a text input field labeled 'Tu respuesta'.

Figura 58 - Encuesta de usabilidad sección 1

- Segunda sección: en esta sección se realizan diez preguntas basadas en el sistema SUS para cuestionarios de usabilidad, el funcionamiento del cual se explica con más detalle en el punto 6.3 de este documento.

**Cuestionario de usabilidad**

Creo que usaría esta aplicación frecuentemente \*

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Encuentro esta aplicación innecesariamente compleja \*

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Creo que la aplicación fue fácil de usar \*

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar la aplicación \*

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Figura 59 - Encuesta de usabilidad sección 2.1

Las funciones de esta aplicación están bien integradas \*

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Creo que la aplicación es muy inconsistente \*

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Figura 60 - Encuesta de usabilidad sección 2.2

Imagino que la mayoría de la gente aprendería a utilizar esta aplicación de forma muy rápida

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Creo que la aplicación es muy difícil de usar \*

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Me siento confiado al usar esta aplicación \*

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Necesité aprender muchas cosas antes de usar esta aplicación

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

Figura 61 - Encuesta de usabilidad sección 2.3

- Tercera sección: adicionalmente a la encuesta SUS, se han añadido algunas preguntas para comprobar, más detalladamente, algunos posibles fallos de la aplicación, así como otras orientadas al diseño de la interfaz de usuario. Además, se pide al usuario que haga un comentario indicando que es lo que más le ha gustado y que mejoraría de la aplicación.

La aplicación responde lentamente a las acciones						
	1	2	3	4	5	
Muy en desacuerdo	<input type="radio"/>	Muy de acuerdo				

La aplicación se ha parado alguna vez de forma inesperada						
	1	2	3	4	5	
Muy en desacuerdo	<input type="radio"/>	Muy de acuerdo				

La forma en la que se presenta la aplicación es clara, comprensible						
	1	2	3	4	5	
Muy en desacuerdo	<input type="radio"/>	Muy de acuerdo				

Las interfaces de usuario son intuitivas						
	1	2	3	4	5	
Muy en desacuerdo	<input type="radio"/>	Muy de acuerdo				

Figura 62 - Encuesta de usabilidad sección 3.1

Recomendaría esta aplicación a mis conocidos

1 2 3 4 5

Muy en desacuerdo      Muy de acuerdo

¿Como valorarías tus habilidades y conocimientos en software?

Gran experiencia y conocimientos técnicos

Buenos pero no muy técnicos

Me las arreglo con la mayor parte del software

Casi todos los tipos de software me parecen difíciles de usar

¿Qué te ha resultado más útil de la aplicación y por qué?

Tu respuesta \_\_\_\_\_

¿Que crees que se debería mejorar en este software y por qué?

Tu respuesta \_\_\_\_\_

Figura 63 - Encuesta de usabilidad sección 3.2