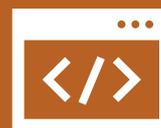




Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo



Escuela de
Ingeniería
Informática
Universidad de Oviedo



Trabajo de
Desarrollo

DISEÑO Y DESARROLLO DE UNA APLICACIÓN PARA LA PLANIFICACIÓN EFICIENTE DE RUTAS Y VIAJES(II): APP MÓVIL

**GRADO EN INGENIERÍA INFORMÁTICA DEL
SOFTWARE
TRABAJO DE FIN DE GRADO**

AUTOR

Francisco Manuel Fernández Suárez

TUTOR

José Manuel Redondo López

Julio 2023

Este documento ha sido creado basándose en la plantilla elaborada por JOSÉ MANUEL REDONDO LÓPEZ.

Declaración Responsable

El alumno: Francisco Manuel Fernández Suárez

Con DNI:

Y UO:

DECLARA

Que esta obra es completamente original y se han citado debidamente las fuentes utilizadas durante la realización de esta.

Y para que conste, lo firma en Oviedo, a 29 de junio de 2023

Firmado:

Índice de contenido

Capítulo 1	¿Qué es este Trabajo?	13
1.1	Resumen	13
1.2	Palabras clave	13
Capítulo 2	Planificación del Sistema de Información	14
2.1	PSI 1: Inicio del Plan de Sistemas de Información	15
2.1.1	PSI 1.1: Análisis de la Necesidad del PSI	15
2.1.2	PSI 1.2: Identificación del Alcance del PSI	15
2.1.3	PSI 1.3: Determinación de Responsables	15
2.2	PSI 2: Definición y Organización del PSI	16
2.2.1	PSI 2.1: Especificación del Ámbito y Alcance	16
2.2.2	PSI 2.2: Organización del PSI	18
Capítulo 3	PSI 7: Definición de la Arquitectura Tecnológica	21
3.1	PSI 7.1: Identificación de las Necesidades de Infraestructura Tecnológica	22
3.2	PSI 7.2: Selección de la Arquitectura Tecnológica	24
Capítulo 4	Estudio de Viabilidad del Sistema	27
4.1	EVS 4, 5 y 6: Estudio y Valoración de Alternativas de Solución y Selección de Alternativa Final 28	
4.1.1	ViaMichelin	28
4.1.2	Guía Repsol	28
4.1.3	51Trips	30
4.1.4	Google Travel	31
Capítulo 5	Planificación y Gestión del TFG	33
5.1	Planificación del proyecto	34
5.1.1	Identificación de Interesados	34
5.1.2	Planificación Inicial. WBS	34
5.1.3	PBS	34
5.1.4	Riesgos	36
5.1.5	Presupuesto Inicial	38
5.2	Cierre del Proyecto	40

5.2.1	Planificación Final	40
5.2.2	Presupuesto Final de Costes	40
5.2.3	Informe de Lecciones Aprendidas	40
Capítulo 6	Análisis del Sistema de Información	43
6.1	ASI 1: Especificación de Requisitos	44
6.2	ASI 3: Identificación de Subsistemas de Análisis.....	44
6.2.1	Descripción de los Subsistemas	44
6.2.2	Descripción de las Interfaces entre Subsistemas	45
6.3	ASI 5: Análisis de Clases.....	46
6.3.1	Diagrama de Clases	46
6.3.2	Descripción de las Clases	46
6.4	ASI 8: Definición de Interfaces de Usuario.....	54
6.4.1	Diseño inicial de la Interfaz	54
6.4.2	Descripción de la Interfaz	64
6.4.3	Descripción del Comportamiento de la Interfaz.....	71
6.4.4	Diagrama de Navegabilidad	71
6.5	ASI 10: Especificación del Plan de Pruebas	73
Capítulo 7	Diseño del Sistema de Información	75
7.1	DSI 3: Diseño de Casos de Uso Reales.....	76
7.1.1	Diagrama de Interacción Web	76
7.1.2	Diagrama de Interacción App	77
7.2	DSI 4: Diseño de Clases	77
7.2.1	Diagrama de Clases.....	77
7.3	DSI 5: Diseño de la Arquitectura de Módulos del Sistema	78
7.3.1	DSI 5.1 Diseño de Módulos del Sistema	78
7.3.2	DSI 5.2 Diseño de Comunicaciones entre Módulos.....	79
7.4	DSI 10: Especificación Técnica del Plan de Pruebas.....	79
7.4.1	Pruebas Unitarias.....	79
7.4.2	Pruebas de Integración y del Sistema.....	79
7.4.3	Pruebas de Usabilidad	80
7.4.4	Pruebas de Accesibilidad	82
7.4.5	Pruebas de Rendimiento.....	82

Capítulo 8	Construcción del Sistema de Información	83
8.1	CSI 1: Preparación del Entorno de Generación y Construcción	84
8.1.1	Herramientas y programas usados para el desarrollo	84
8.2	CSI 2: Generación del Código de los Componentes y Procedimientos	86
8.3	CSI 4: Ejecución de las Pruebas de Integración	86
8.4	CSI 5: Ejecución de las Pruebas del Sistema	89
8.4.1	Prueba de Usabilidad	89
8.4.2	Pruebas de Accesibilidad	91
8.4.3	Pruebas de Rendimiento	95
8.5	CSI 6: Elaboración de los Manuales de Usuario	96
8.5.1	Manual de Usuario	96
Capítulo 9	Conclusiones y Ampliaciones	109
9.1	Conclusiones	110
9.2	Ampliaciones	110
Apéndices		113
	Especificación de Requisitos	114
	Introducción	114
	Descripción general	115
	Requisitos específicos	116
	Diagramas UML	121
	Escenarios	124
	Plan de gestión de riesgos	126
	Metodología	126
	Categorías de Riesgo	128
	Definiciones de riesgo	128
	Matriz de probabilidad e impacto	129
	Referencias Bibliográficas	129
	Contenido entregado en los anexos	133

Índice de Figuras

ILUSTRACIÓN 1. COMPARACIÓN ENTRE BASES DE DATOS.....	24
ILUSTRACIÓN 2. CAPTURA DEL SISTEMA VIAMICHELIN.....	28
ILUSTRACIÓN 3. INTERFAZ VISUAL DE WEB DE GUÍA REPSOL.....	29
ILUSTRACIÓN 4. INTERFAZ VISUAL DE APP GUÍA REPSOL.....	29
ILUSTRACIÓN 5. CAPTURA DE UN LIBRO DE VIAJES.....	30
ILUSTRACIÓN 6. GOOGLE TRAVEL.....	31
ILUSTRACIÓN 7. PBS.....	35
ILUSTRACIÓN 8. COSTES DE PERSONAL.....	38
ILUSTRACIÓN 9. COSTES INDIRECTOS.....	39
ILUSTRACIÓN 10. COSTES DE PRODUCCIÓN.....	39
ILUSTRACIÓN 11. INTERFACES ENTRE SUBSISTEMAS.....	45
ILUSTRACIÓN 12. HOMEPAGE.....	55
ILUSTRACIÓN 13. RUTAS DEL USUARIO Y DETALLE DE LA RUTA.....	56
ILUSTRACIÓN 14. EDICIÓN Y CREACIÓN DE RUTAS.....	57
ILUSTRACIÓN 15. PERFILES DE USUARIO.....	58
ILUSTRACIÓN 16. COLECCIONES Y LISTA DE AMIGOS.....	59
ILUSTRACIÓN 17. PANTALLAS MÓVIL.....	60
ILUSTRACIÓN 18. PANTALLAS MÓVIL.....	61
ILUSTRACIÓN 19. PANTALLAS MÓVIL.....	62
ILUSTRACIÓN 20. PANTALLAS MÓVIL.....	63
ILUSTRACIÓN 21. PANTALLAS MÓVIL.....	63
ILUSTRACIÓN 22. SPLASHACTIVITY.....	64
ILUSTRACIÓN 23. LOGINACTIVITY.....	65
ILUSTRACIÓN 24. MAINACTIVITY.....	65
ILUSTRACIÓN 25. MENÚ PRINCIPAL.....	66
ILUSTRACIÓN 26. EDICIÓN DE PERFIL.....	66
ILUSTRACIÓN 27. LISTADO DE COLECCIONES.....	67
ILUSTRACIÓN 28. LISTADO DE RUTAS.....	67
ILUSTRACIÓN 29. LISTA DE AMIGOS.....	68
ILUSTRACIÓN 30. LISTADO DE SOLICITUDES.....	68
ILUSTRACIÓN 31. DETALLE DE RUTA.....	69
ILUSTRACIÓN 32. COMPARTIR RUTAS.....	69
ILUSTRACIÓN 33. MAPA INTERACTIVO.....	70
ILUSTRACIÓN 34. AYUDA.....	70
ILUSTRACIÓN 35. DIAGRAMA DE NAVEGABILIDAD WEB.....	72
ILUSTRACIÓN 36. DIAGRAMA DE NAVEGABILIDAD MÓVIL.....	72
ILUSTRACIÓN 37. DIAGRAMA DE SECUENCIA. CREAR RUTA EN WEB.....	76
ILUSTRACIÓN 38. DIAGRAMA DE INTERACCIÓN DE CREACIÓN DE RUTA.....	77
ILUSTRACIÓN 39. DIAGRAMA DE PAQUETES MÓVIL.....	78
ILUSTRACIÓN 40. DIAGRAMA DE DESPLIEGUE.....	79
ILUSTRACIÓN 41. PRUEBAS DE ACCESIBILIDAD CON TESTLAB.....	92
ILUSTRACIÓN 42. PROBLEMA MENOR 1.....	93
ILUSTRACIÓN 43. PROBLEMA MENOR 2.....	94
ILUSTRACIÓN 44. PROBLEMA MENOR 3.....	95
ILUSTRACIÓN 45. PRUEBAS DE RENDIMIENTO DE FIREBASE.....	95
ILUSTRACIÓN 46. RESULTADOS PRUEBAS RENDIMIENTO TESTLAB.....	96

ILUSTRACIÓN 47. LOGIN 1	97
ILUSTRACIÓN 48. LOGIN 2	97
ILUSTRACIÓN 49. MENÚ	98
ILUSTRACIÓN 50. PERFIL	99
ILUSTRACIÓN 51. COLECCIONES	100
ILUSTRACIÓN 52. RUTAS	101
ILUSTRACIÓN 53. CREAR RUTA 1	102
ILUSTRACIÓN 54. CREAR RUTA 2	103
ILUSTRACIÓN 55. COMPARTIR	104
ILUSTRACIÓN 56. MAPA	105
ILUSTRACIÓN 57. AMIGOS	106
ILUSTRACIÓN 58. SOLICITUDES	107
ILUSTRACIÓN 59. AYUDA	108
ILUSTRACIÓN 60. DIAGRAMA DE CONTEXTO	121
ILUSTRACIÓN 61. CASO DE USO ACCESO AL FEED GLOBAL	122
ILUSTRACIÓN 62. CASO DE USO GESTIÓN DE RUTAS PROPIAS	122
ILUSTRACIÓN 63. CASO DE USO GESTIÓN DE COLECCIONES	123
ILUSTRACIÓN 64. DIAGRAMA DE ESTADOS DE RUTA	124
ILUSTRACIÓN 65. DIAGRAMA METODOLOGÍA DE BOEHM	127
ILUSTRACIÓN 66. ESTRUCTURA DE DESGLOSE DE RIESGOS	128
ILUSTRACIÓN 67. MATRIZ DE PROBABILIDAD E IMPACTO	129

Índice de Tablas

TABLA 1. ORGANIZACIÓN DEL PSI	18
TABLA 2. RIESGOS DEL PROYECTO	36
TABLA 3. IMAGEPAGERADAPTER	46
TABLA 4. LISTAMIGOSADAPTER	46
TABLA 5. LISTCOLECCIONESADAPTER	46
TABLA 6. LISTRUTASADAPTER	47
TABLA 7. LISTSHAREADAPTER	47
TABLA 8. LISTSOLICITUDESADAPTER	47
TABLA 9. AMIGO	48
TABLA 10. COLECCION	48
TABLA 11. RUTA	48
TABLA 12. SOLICITUD	49
TABLA 13. USER	49
TABLA 14. AMIGOSFRAGMENT	49
TABLA 15. AYUDAFRAGMENT	50
TABLA 16. COLECCIONESFRAGMENT	50
TABLA 17. INICIOFRAGMENT	51
TABLA 18. PERFILFRAGMENT	51
TABLA 19. SOLICITUDESFRAGMENT	51
TABLA 20. LOGINACTIVITY	52
TABLA 21. MAINACTIVITY	52
TABLA 22. RUTAACTIVITY	53
TABLA 23. SHAREACTIVITY	54

TABLA 24. SPLASHACTIVITY	54
TABLA 25. PREGUNTAS DE CARÁCTER GENERAL	80
TABLA 26. PREGUNTAS CORTAS DE USABILIDAD	81
TABLA 27. CASO DE USO 1	86
TABLA 28. CASO DE USO 2	86
TABLA 29. CASO DE USO 3	87
TABLA 30. CASO DE USO 4	87
TABLA 31. CASO DE USO 5	87
TABLA 32. CASO DE USO 6	87
TABLA 33. CASO DE USO 7	88
TABLA 34. CASO DE USO 8	88
TABLA 35. CASO DE USO 9	88
TABLA 36. CASO DE USO 10	88
TABLA 37. CASO DE USO 11	88
TABLA 38. CASO DE USO 12	89
TABLA 39. CASO DE USO 13	89
TABLA 40. CASO DE USO 14	89
TABLA 41. ENCUESTA DE USABILIDAD	90
TABLA 42. PREGUNTAS CORTAS DE USABILIDAD	90
TABLA 43. FUNCIONALIDADES PRINCIPALES	116
TABLA 44. ESCENARIO MÓVIL 1	124
TABLA 45. ESCENARIO MÓVIL 2	125
TABLA 46. ESCENARIO WEB 1	125
TABLA 47. ESCENARIO WEB 2	125
TABLA 48. ESCENARIO WEB 3	126
TABLA 49. TABLA DE PROBABILIDAD DE RIESGOS	128

Capítulo 1 ¿QUÉ ES ESTE TRABAJO?

1.1 RESUMEN

Este Trabajo Fin de Grado, al que se ha denominado FavoritePath, nace de la idea de que las aplicaciones actuales de mapeo y manejo de rutas son, utilizadas en conjunto, muy completas, pero por separado todas carecen de algunas funcionalidades interesantes como la posibilidad de tener la parte web y una aplicación móvil, todas las funcionalidades gratuitas y sin anuncios o pasando a mayores, tener la certeza de que tus datos de localización no son usados para otros fines que no sean los tuyos propios. Es un sistema que pretende aunar varias de estas características y funcionalidades interesantes en una sola aplicación gratuita.

El sistema está conformado por un sitio web y una aplicación móvil. Su finalidad es la preparación de viajes y rutas, para lo cual los usuarios podrán seleccionar distintos puntos de interés, y se creará automáticamente el recorrido óptimo entre ellos. Este podrá ser compartido con amigos. La ruta se mostrará en un mapa interactivo con indicaciones para su fácil realización.

Debido a que el desarrollo de un sistema de tal complejidad se escapa del alcance, en horas, de un trabajo fin de grado, se ha llevado a cabo por dos alumnos, repartiendo el trabajo entre sitio web y aplicación móvil. Por lo tanto, a estos trabajos, además de la complejidad propia del sistema en sí, hay que añadirles la dificultad derivada de la comunicación y compenetración necesaria entre las dos partes para llevarlo a cabo de forma homogénea.

En este trabajo se abordará, como en su título indica, la aplicación móvil de este sistema.

1.2 PALABRAS CLAVE

Aplicación móvil, Android, gestión de rutas, colecciones.



Capítulo 2 PLANIFICACIÓN DEL SISTEMA DE INFORMACIÓN

FASE DE PLANIFICACIÓN

PSI



2.1 PSI 1: INICIO DEL PLAN DE SISTEMAS DE INFORMACIÓN

2.1.1 PSI 1.1: Análisis de la Necesidad del PSI

Nos encontramos ante la necesidad de desarrollar un sistema informático que facilite a los usuarios la preparación de viajes y rutas de manera eficiente junto a otras personas de forma gratuita. Este sistema deberá ser accesible a todo el público en general, y tendrá que ser portable para que los usuarios puedan utilizarlo en cualquier lugar, a fin de que sea útil durante la realización de las rutas. También deberá ofrecer a los usuarios la posibilidad de contactar entre ellos a través del sistema. El software debe planificar, seleccionar y compartir rutas entre sus usuarios, así como proporcionar ayudas e indicaciones necesarias para que varias personas las completen con éxito.

2.1.2 PSI 1.2: Identificación del Alcance del PSI

Actualmente, el sistema de referencia de casi todos los usuarios para realizar rutas eficientes es el que proporciona Google, aunque existen otros con las mismas funcionalidades, o incluso otras más innovadoras. Sin embargo, si recogemos estos sistemas y los estudiamos, podemos observar que en todos existe alguna carencia para los usuarios. Un punto importante es el bloqueo de funcionalidades debido a las compras dentro de la aplicación. Otro, que el sistema contenga funcionalidades muy potentes y eficientes, pero también muy acotadas. Con esto en mente, el alcance de nuestro proyecto es la creación de un sistema completo y portable totalmente gratuito. Esto conlleva la creación de un portal web para las funcionalidades más complejas, y de una aplicación móvil, más enfocada a un uso cooperativo del sistema.

Por tanto, los objetivos estratégicos a lograr para que el proyecto sea un éxito son:

- Simplificar la labor de planificación de rutas y viajes.
- Gestionar las rutas creadas por los usuarios.
- Gestionar amigos para un uso cooperativo del sistema.
- Facilitar la información necesaria para completar en grupo las rutas y viajes anteriormente planificados.
- Integrar de forma intuitiva las funcionalidades básicas de sistemas competidores.

2.1.3 PSI 1.3: Determinación de Responsables

Este es un proyecto realizado en pareja. A continuación, delimitaremos las responsabilidades de los dos proyectantes en su desarrollo:

- Por un lado, Sara Rubín Estrada-Nora es la encargada de la construcción del portal web. Esto incluye el diseño, el desarrollo y la realización de las correspondientes pruebas.



- Por otro lado, Francisco Manuel Fernández Suárez se encarga de la construcción de la aplicación móvil, incluyendo el diseño, desarrollo y realización de las correspondientes pruebas.
- Ambos **proyectantes** colaborarán para obtener homogeneidad visual y funcional entre ambas partes del sistema. Además, negociarán el modelo de datos compartido, y las interfaces que se suministrarán entre ellos.
- El **director de proyecto** supervisará la consecución de los objetivos intermedios durante todo el proceso. No obstante, serán los **proyectantes** los responsables del cumplimiento de estas metas durante el desarrollo del proyecto hasta su finalización.

2.2 PSI 2: DEFINICIÓN Y ORGANIZACIÓN DEL PSI

2.2.1 PSI 2.1: Especificación del Ámbito y Alcance

A raíz de los objetivos estratégicos determinados en el punto 2.1.2, el proyecto se dividirá en las siguientes fases con sus correspondientes objetivos:

Fase 1: Planificación de recorridos

Se desarrollará un sistema que permita a los usuarios realizar la labor de planificación de sus rutas y viajes. Se podrán crear recorridos a partir de una lista de puntos de interés. Los usuarios seleccionarán estos puntos sobre un mapa, y decidirán si quieren que la ruta sea lineal o circular. El sistema se encargará de calcular el recorrido óptimo que pase por todos los puntos de interés. Una vez calculado el recorrido, el usuario podrá guardarlo con un nombre. Opcionalmente, podrá añadir otros datos como una descripción y fotografías. La ruta podrá ser privada o pública, de manera que otros usuarios puedan interactuar con ella. Existirá la posibilidad de editar a posteriori de la publicación el nombre, la descripción y las fotografías. El creador de una ruta será el único usuario con la opción de eliminarla.

Objetivos de la fase:

- Que los usuarios puedan crear sus propias rutas marcando los puntos que quieren visitar en un mapa, y decidiendo si el recorrido será lineal o circular.
- Que los usuarios puedan añadir una descripción y fotografías a sus rutas.
- Que los usuarios puedan publicar sus recorridos para que otros puedan interactuar con ellos.
- Que los usuarios puedan editar las rutas creadas por ellos mismos.
- Que los usuarios puedan eliminar las rutas creadas por ellos mismos.

Fase 2: Gestión de rutas

Los usuarios podrán acceder a un listado de rutas públicas. Podrán visualizar los detalles de estas rutas. También podrán realizar un filtrado o una ordenación de las rutas. Cada usuario podrá crear sus propias colecciones, donde guardar los recorridos que gusten. Se podrán guardar tanto rutas propias como de otros usuarios. Todos los usuarios contarán con una colección creada

automáticamente en la que se guardarán las rutas creadas por ellos mismos. A su vez, existirá otra colección en la que se guardarán aquellas rutas que los usuarios marquen como favoritas, tanto si son propias como si no. Se podrán crear otras colecciones, a las que se podrá dar el nombre que se considere. Se podrá añadir cualquier ruta a una colección, siempre y cuando no esté ya en ella. Esas colecciones podrán ser eliminadas por el usuario, al contrario que las dos anteriores.

Objetivos de la fase:

- Proporcionar un listado con las rutas públicas.
- Proporcionar filtros y ordenación para los listados.
- Que los usuarios puedan crear colecciones donde añadir las rutas que deseen.
- Que los usuarios puedan eliminar las colecciones creadas por ellos mismos.

Fase 3: Gestión de amigos

Los usuarios podrán añadir a otros como amigos. Para ello, podrán enviarles una solicitud, especificando el correo electrónico del amigo o desde el perfil de este. El amigo recibirá un aviso, en el que podrá aceptar o rechazar la solicitud. Una vez aceptada, ambos usuarios verán al otro en su correspondiente lista de amigos. En el momento de realización de una ruta, los usuarios podrán conformar un grupo con varios amigos.

Objetivos de la fase:

- Que los usuarios puedan enviar solicitudes de amistad a otros.
- Que los usuarios puedan ver las solicitudes de amistad que se les ha enviado, y aceptarlas o rechazarlas.
- Que los usuarios puedan visualizar su lista de amigos.
- Que los usuarios puedan eliminar amigos de su lista.
- Que los usuarios puedan añadir a un grupo a varios amigos al momento de comenzar una ruta.

Fase 4: Facilitación de información para completar los recorridos

Cuando el usuario lo especifique, se dará comienzo a una ruta. El sistema se valdrá de la geolocalización del usuario para mostrar, en tiempo real, las indicaciones necesarias para llegar al siguiente punto de interés del recorrido. Además, se revelará a cada usuario la localización de los amigos que estén realizando la ruta en grupo con él en el mapa.

Objetivos de la fase:

- Mostrar las indicaciones pertinentes hasta el próximo punto de interés de la ruta que el usuario haya comenzado.
- Mostrar en un mapa a los amigos del usuario que estén realizando la ruta en conjunto con él.

Fase 5: Estandarización con sistemas competidores

Se diseñará la interfaz gráfica del sistema y sus funcionalidades de tal forma que, aunque un usuario nunca haya utilizado el sistema, le resulte familiar si está acostumbrado a algún sistema competidor. Además, existirá una página de ayuda a la cual se podrá acceder en cualquier momento para consultar dudas sobre el uso del sistema.

Objetivos de la fase:

- Obtener un sistema cuya interfaz gráfica sea intuitiva y similar a la de otros sistemas competidores.
- Mostrar en todo momento un acceso a la página de ayuda, donde se podrá consultar cualquier duda sobre el sistema.

2.2.2 PSI 2.2: Organización del PSI

Tras explicar los objetivos de cada fase del proyecto y el objetivo general de este, se han creado los siguientes equipos de trabajo:

Tabla 1. Organización del PSI

USUARIO	PERFIL	FUNCIÓN
EQUIPO DE SUPERVISIÓN		
<i>Director del proyecto</i>	<i>Informático</i>	<i>Supervisión de la consecución de los objetivos intermedios de todas las fases.</i>
EQUIPO DE DESARROLLO		
<i>Sara Rubín Estrada-Nora</i>	<i>Informático</i>	<i>Diseño y desarrollo de los módulos software del portal web.</i>
<i>Francisco Manuel Fernández Suárez</i>	<i>Informático</i>	<i>Diseño y desarrollo de los módulos software de la aplicación móvil. Construcción de la base de datos.</i>
EQUIPO DE PRUEBAS		
<i>Sara Rubín Estrada-Nora</i>	<i>Informático</i>	<i>Pruebas de integración, rendimiento, usabilidad y accesibilidad del software en el portal web. Comprobación de que las funcionalidades previstas se están cumpliendo correctamente.</i>
<i>Francisco Manuel Fernández Suárez</i>	<i>Informático</i>	<i>Pruebas de integración, rendimiento, usabilidad y accesibilidad del software en la aplicación móvil. Comprobación de que las funcionalidades previstas se están cumpliendo correctamente.</i>
<i>Otros usuarios seleccionados</i>	<i>No informático</i>	<i>Pruebas de usabilidad y accesibilidad del software completo.</i>



EQUIPO DE DISEÑO

<i>Sara Rubín Estrada-Nora</i>	<i>Informático</i>	<i>Diseño del logotipo del sistema para la web</i>
<i>Francisco Manuel Fernández Suárez</i>	<i>Informático</i>	<i>Diseño del logotipo del sistema para la aplicación móvil</i>

Los materiales que los alumnos utilizarán para la realización del proyecto serán dos ordenadores portátiles para el desarrollo y pruebas del proyecto y dos dispositivos móviles para probar la aplicación móvil.

Capítulo 3 PSI 7: DEFINICIÓN DE LA ARQUITECTURA TECNOLÓGICA

FASE DE PLANIFICACIÓN

PSI



3.1 PSI 7.1: IDENTIFICACIÓN DE LAS NECESIDADES DE INFRAESTRUCTURA TECNOLÓGICA

El sistema se divide en un portal web y una aplicación móvil. Cada una de estas partes tiene unas necesidades muy distintas en cuanto a tecnologías, pero, dado que los datos utilizados por ambas serán los mismos, tiene sentido que se comparta la base de datos.

Por tanto, se hace necesaria una base de datos que se pueda compartir entre el portal web y la aplicación móvil. Debe poder sincronizar sus datos a tiempo real con todos los clientes. Es importante que estos datos se mantengan disponibles aun cuando el sistema no lo esté. También hay que considerar el alojamiento de esta base de datos. Dado que no contamos con un servidor propio, una opción es alojar nuestra base de datos en una nube gratuita, y otra, elegir una base de datos alojada en la nube. No es fundamental elegir entre una base de datos relacional o no; ambas servirían para nuestro propósito, ya que el modelo de datos no es muy complejo. Teniendo en cuenta todo lo anteriormente mencionado, y priorizando las opciones que ya hemos manejado con anterioridad, tenemos estas posibilidades:

- MongoDB Atlas.
- Firebase Realtime Database.
- Cloud Firestore.

Además, necesitaremos una forma de almacenar las imágenes, para guardarlas y mostrarlas, por lo que, teniendo en cuenta lo anterior y añadiendo opciones conocidas y que ya hemos manejado, existen estas posibilidades:

- Firebase Storage.
- Amazon S3.
- Google Cloud Storage.

El sistema requiere de alguna forma de autenticación, para poder manejar los datos generados para cada usuario. Implementar nosotros mismos esta funcionalidad tan importante para la seguridad del usuario en una aplicación tan pequeña y sin un equipo especializado queda descartado, para evitar problemas innecesarios. Así pues, delegaremos en un tercero, teniendo la ventaja de no tener que guardar las contraseñas, facilitando la gestión de la información del usuario y así cumpliendo con la LOPD (Ley Orgánica de Protección de Datos) (Ley Orgánica de Protección de Datos), teniendo siempre en cuenta que se debe compartir esta funcionalidad entre el portal web y la aplicación móvil. Por tanto, y priorizando aquellas que ya conocemos, las opciones para la autenticación son:

- MongoDB Atlas Authentication.
- Firebase Authentication.



Siguiendo la misma filosofía que con la base de datos, el portal web también tendrá que alojarse en un servidor gratis y sin anuncios. Algunas opciones de hosting que cumplen con estos requisitos y con las que estamos familiarizados son:

- Github Pages.
- Firebase Hosting.

Para el desarrollo del servicio web se han de tener en cuenta:

- El lenguaje de programación y framework con el que se construirá.
 - El lenguaje de programación tendrá que ser uno que sirva para desarrollo web y que conozcamos mínimamente. Nuestras opciones son JavaScript, Python y Java.
 - No es necesario utilizar un framework, pero puede ser útil. Para elegirlo se ha de tener en cuenta el lenguaje de programación elegido. Nuestras opciones de frameworks son React, Angular, Django y Spring.
- Las librerías externas que se usarán. Para su elección tendremos en cuenta las necesidades a la hora de desarrollar el sistema. Una opción útil podría ser jQuery.
- APIs. Necesitaremos una interfaz de programación con algún sistema proveedor de mapas. Nuestras opciones, basándonos en nuestro conocimiento sobre ellas, son Google Maps,
- OpenStreetMap, Leaflet y Mapbox.
- El IDE que se utilizará para el desarrollo. Esta es una cuestión de comodidad del programador. Las opciones barajadas son Visual Studio Code, Eclipse, Webstorm e IntelliJ IDEA.

Para el desarrollo de la aplicación móvil se han de tener en cuenta:

- El lenguaje de programación y el framework con el que se construirá.
 - Las opciones de lenguaje de programación para el desarrollo de una aplicación móvil son Java, Kotlin y Swift.
 - No es necesario un framework como se ha comentado anteriormente, pero algunas opciones son React Native, Flutter y Cordova.
- Las librerías externas que se usarán. Aquí nos centraremos en las necesidades del desarrollo del sistema para poder elegir las. Algunas opciones son Picasso, Glide o Coil para la visualización de imágenes.
- APIs. Al igual que se ha comentado antes, las opciones serían Google Maps, OpenStreetMap, Leaflet y Mapbox.
- El IDE que se utilizará para el desarrollo. Las diferentes opciones para llevar a cabo el proyecto serían Visual Studio Code, Eclipse, Webstorm y Android Studio.

3.2 PSI 7.2: SELECCIÓN DE LA ARQUITECTURA TECNOLÓGICA

Una vez acotadas las opciones de arquitectura en el apartado anterior, se ha hecho un estudio más profundo de cada una de ellas para determinar cuáles serán las más beneficiosas para el proyecto.

Para la base de datos, MongoDB ha sido desestimado porque la integración de la API es más complicada. Se ha hecho un estudio para ver cuál de las bases de datos ofrecidas por Google es mejor, teniendo en cuenta sus propias recomendaciones. Se ha llegado a la conclusión de que, para este proyecto, ninguna es mejor que la otra:

Consideraciones clave

Más allá de las excelentes funciones principales comunes de ambas bases de datos, piensa en cómo algunas de las siguientes consideraciones (o todas) influirán en el éxito de tus apps.

Función de la base de datos	Mi app usa una base de datos para... Si no necesitas realizar consultas, transacciones ni ordenamientos avanzados, te recomendamos Realtime Database .
Operaciones con datos	El uso de la base de datos de mi aplicación tiene... Si tu app envía un flujo de pequeñas actualizaciones, como en una app de pizarra digital, recomendamos utilizar Realtime Database .
Modelo de datos	Prefiero estructurar mis datos como... Para los documentos y colecciones estructurados, recomendamos Cloud Firestore .
Disponibilidad	Mis necesidades de disponibilidad son... Cuando se acepta una disponibilidad muy alta, pero que no llega a ser crítica, recomendamos Cloud Firestore o Realtime Database .
Consultas sin conexión en datos locales	Mi app deberá realizar consultas en dispositivos con conectividad limitada o sin conexión... Para capacidades de consulta sofisticadas en datos locales cuando el usuario no tiene conexión, recomendamos Cloud Firestore .
Cantidad de instancias de base de datos	En mis proyectos individuales, necesito usar... Si necesitas una sola base de datos, te recomendamos Cloud Firestore o Realtime Database .

Ilustración 1. Comparación entre bases de datos

Finalmente se ha decidido utilizar Cloud Firestore, por ser la única que no habíamos utilizado todavía, su sencillez y las opciones que nos ofrece. Proporciona una base de datos no relacional, protegida y fiable para acceder desde la web y desde la app a la vez.

En cuanto al proceso de autenticación, se ha decidido delegar en Firebase. Esto es debido principalmente al deseo de aunar todos los servicios externos en un solo proveedor, en este caso Google. También ofrece tanto una autenticación email/contraseña como la posibilidad de integrar las credenciales de Google, entre otras opciones.



También hemos decidido usar Firebase Storage, debido a su sencillez, como forma de unión de servicios en un proveedor y como forma de aprendizaje de otra opción del manejo de imágenes en la nube, debido a que no la habíamos utilizado con anterioridad.

El servidor que se ha elegido para hostear la página web es Firebase, ya que la opción de Github Pages no proporcionaba ninguna ventaja, mientras que Firebase ya se utiliza en otras partes del sistema.

El lenguaje de programación escogido para la implementación de la página web es JavaScript, por su versatilidad y su uso extendido en desarrollo web, y también por preferencia personal de la desarrolladora. El lenguaje se utiliza sin ayuda de ningún framework ni librería en el lado del frontend. Para el backend se hace uso de Node.js.

La aplicación móvil se desarrolla en Java, debido a la familiaridad del desarrollador con el lenguaje a la hora de la realización de aplicaciones móviles. Tampoco se utiliza ningún framework. Las librerías utilizadas son Picasso, para el uso de imágenes, debido a la familiaridad y a su sencillez.

Para la visualización y manejo de los mapas del sistema, la elección final es Mapbox. No hemos utilizado Google Maps debido a la necesidad de añadir una tarjeta para el cobro de las funcionalidades que excedan sus límites gratuitos, ya que queremos una API totalmente gratis. Mapbox es la elección por ser gratuita, así como por el amplio abanico de opciones que ofrece.

Por último, los IDEs elegidos para el desarrollo son Visual Studio Code para la parte web, y Android Studio para la aplicación móvil. Esta elección se basa en la comodidad y gusto personal de cada desarrollador.

Capítulo 4 ESTUDIO DE VIABILIDAD DEL SISTEMA

FASE DE DESARROLLO

EVS

4.1 EVS 4, 5 Y 6: ESTUDIO Y VALORACIÓN DE ALTERNATIVAS DE SOLUCIÓN Y SELECCIÓN DE ALTERNATIVA FINAL

4.1.1 ViaMichelin

El primer sistema que valoramos es [ViaMichelin](#), bastante conocido. Muestra en un mapa diferentes puntos de interés como hoteles, restaurantes o zonas de turismo a un nivel bastante amplio. Se puede escoger entre una vista general de mapa y vistas detalladas de ciudades de interés. Además, te permite calcular diferentes rutas, dependiendo de los puntos que hayas seleccionado, y añadirlas como rutas favoritas. En la Ilustración 2 se muestra una vista general del sistema.



Ilustración 2. Captura del sistema ViaMichelin

La principal ventaja es que es muy completo y ofrece gran cantidad de información al usuario. Cabe resaltar que cuenta con un apartado de opciones donde te permite calcular, en el caso de viajar usando un medio de transporte, el coste total de los kilómetros que vas a realizar, una funcionalidad pocas veces vista en sistemas parecidos y muy útil.

Por otro lado, como inconveniente y ligado a una de sus ventajas, al ofrecer tanta información de golpe puede llegar a saturar a usuarios que no estén familiarizados con el sistema. Tampoco existe un apartado de ayuda. Esto puede llevar a la elección de otro sistema más fácil de usar, dificultando la captación de nuevos usuarios.

4.1.2 Guía Repsol

Como segundo sistema a valorar, elegimos [Guía Repsol](#). Este sistema cuenta tanto con sitio web como con aplicación móvil. El sitio web está más enfocado como blog. En él se puede encontrar todo

tipo de información útil para preparar viajes, pero hace hincapié en la gastronomía (reviews de lugares para comer y recetas). Existe una sección de viajes, con un buscador donde el usuario introduce una localización (existe también la opción de búsqueda por geolocalización del usuario) y el sistema devuelve lugares de interés en la zona, representados en un mapa, como se muestra en la Ilustración 3 y 4.

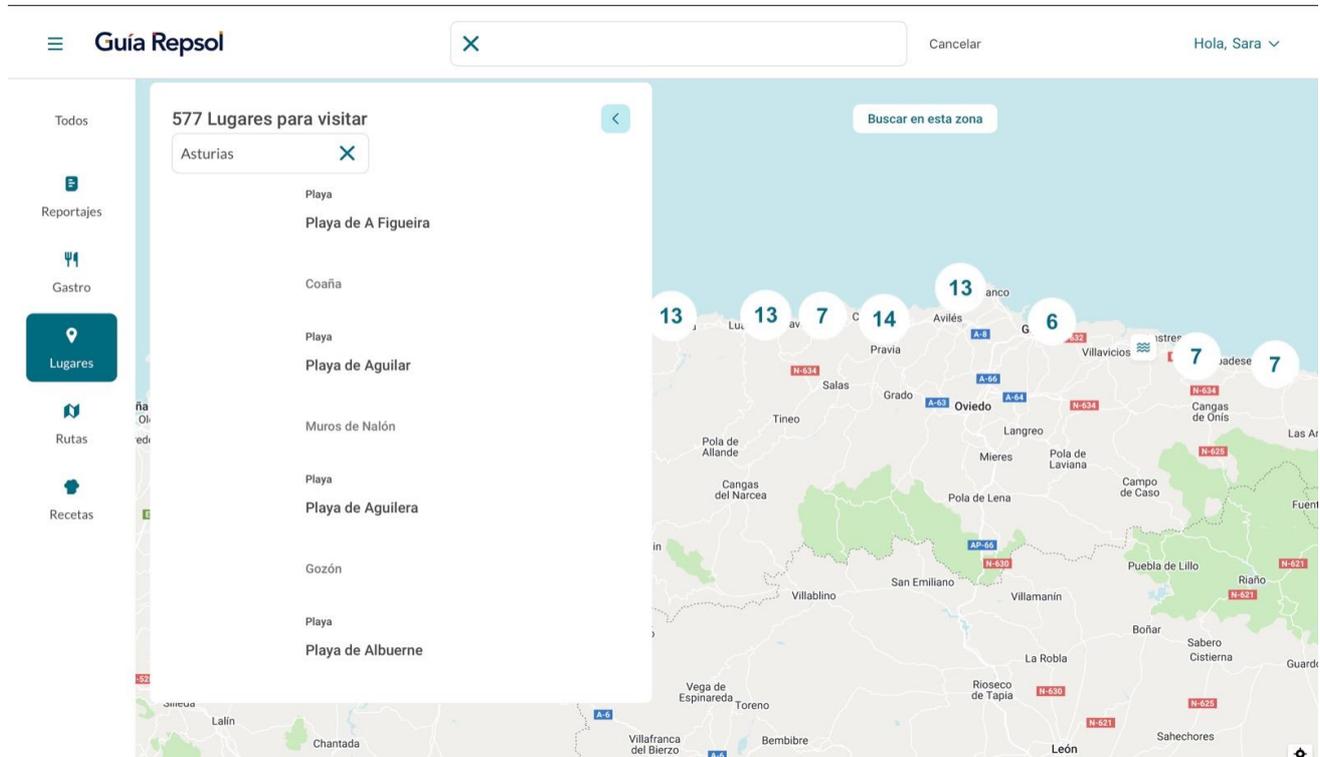


Ilustración 3. Interfaz visual de web de Guía Repsol



Ilustración 4. Interfaz visual de App Guía Repsol

En la sección de rutas encontramos un artículo para cada ruta, con una descripción de esta, un mapa donde aparece marcado el recorrido, y una galería de fotos. Debajo aparecen enlaces a otras entradas sobre dónde comer, dormir y hacer compras cerca de la ruta, así como reportajes sobre sitios cercanos de interés y rutas relacionadas. Las rutas largas aparecen divididas por días, donde se va explicando qué hacer en cada momento.

En cuanto a la aplicación móvil, esta cuenta con una función para organizar viajes, en la cual podemos editar el nombre del viaje, las fechas de ida y vuelta, y las localidades por las que tenemos pensado pasar. El recorrido se muestra en un mapa, donde también se refleja el tiempo atmosférico. También se puede abrir con Google Maps, de manera que nos dará las indicaciones a seguir. Se pueden añadir amigos al viaje, pero es necesario haber iniciado sesión para ello. La Ilustración 3 muestra una vista de la aplicación en su modo de creación de rutas.

La ventaja principal de este sistema es la cantidad y buena organización de la información para la preparación de viajes y rutas. Sin embargo, la página web es bastante lenta, y la mayoría de su contenido es estático. No cuenta con una opción para organizar las rutas en la web, y la opción para móvil no cubre todas las necesidades de planificación (dónde comer, dónde dormir, etc).

4.1.3 51Trips

El tercer sistema valorado se llama [51Trips](#). Se trata de una aplicación móvil donde podrás crear tu propio libro de viajes, eligiendo los diferentes países y regiones que deseas visitar y añadiendo fotos y experiencias de este, disfrutándolo por tu cuenta o añadiendo amigos al viaje.

La Ilustración 5 muestra un ejemplo de un libro de viaje creado por el sistema.

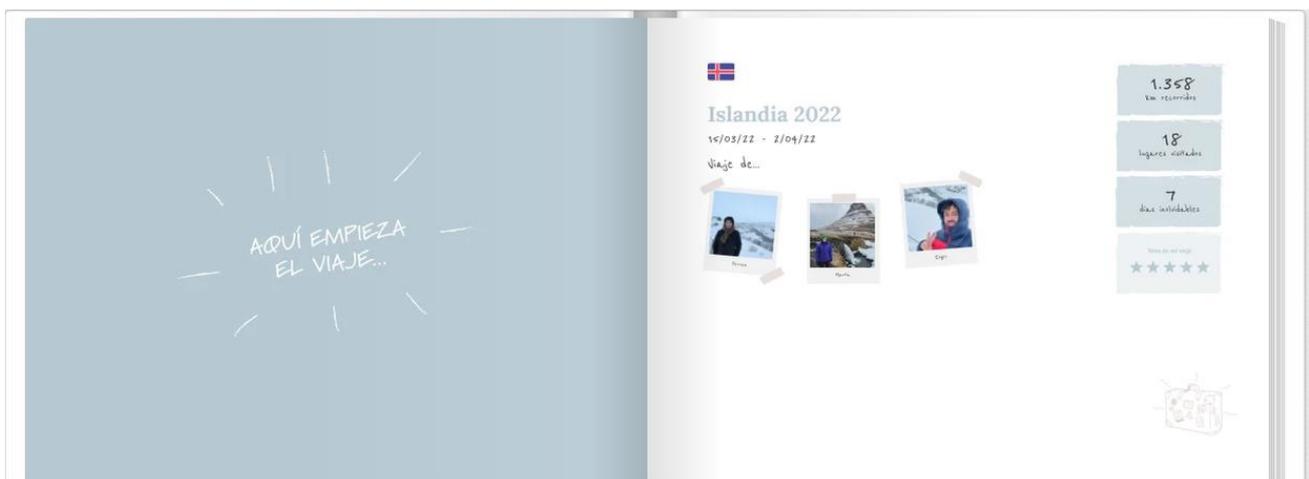


Ilustración 5. Captura de un libro de viajes

Una de las ventajas que ofrece este sistema es que es muy fácil de usar y la aplicación te guía para la realización de tu primer viaje. Además, su forma de unir a otros usuarios a tu viaje es muy intuitiva y sencilla, sólo compartiendo un enlace o usando un código QR. Otro punto a favor de este sistema es que tiene un apartado con porcentajes de países y maravillas del mundo que hayas visitado con la

app, dándole al usuario un toque competitivo para poder alcanzar el 100%. Además, cuenta con un apartado de preguntas frecuentes para aquellos usuarios menos experimentados para mejorar la satisfacción de este con el sistema.

Por otro lado, encuentro como inconveniente que este sistema no te ayuda en ningún momento a realizar tu viaje correctamente; debe ser el propio usuario quien explique su día a día, además de tener que avanzar él mismo los días en la aplicación. Otro inconveniente es que no está muy internacionalizado, ya que te muestra regiones de España, pero si quieres buscar otras fuera de la península, sólo muestra el país en cuestión como conjunto.

4.1.4 Google Travel

El último sistema que hemos valorado es [Google Travel](#). Este servicio de Google permite organizar viajes fácilmente. Lo primero que nos encontramos es un buscador donde el usuario puede introducir cualquier destino o lugar de interés. Además, existen varias opciones de búsqueda: qué hacer, vuelos, hoteles y alquileres. También hay una sección con los destinos más populares, en la que se muestra un precio estimado de vuelos y hoteles para ciertas fechas. Existe una sección donde se pueden ir guardando los lugares que interesen al usuario.

Como principales ventajas, contiene mucha información en un formato amigable, y permite la comparación y reserva de vuelos y hoteles. Al ser un servicio de Google, cada lugar de interés consta de sus reseñas y puntuación de Google, así como de las funciones de Google Maps. Esto lo hace muy completo. Como inconveniente, no permite crear un recorrido ni que este aparezca reflejado en un mapa o línea temporal. Además, no permite compartir los lugares guardados con otras personas.

La Ilustración 6 muestra la página principal de este sistema.

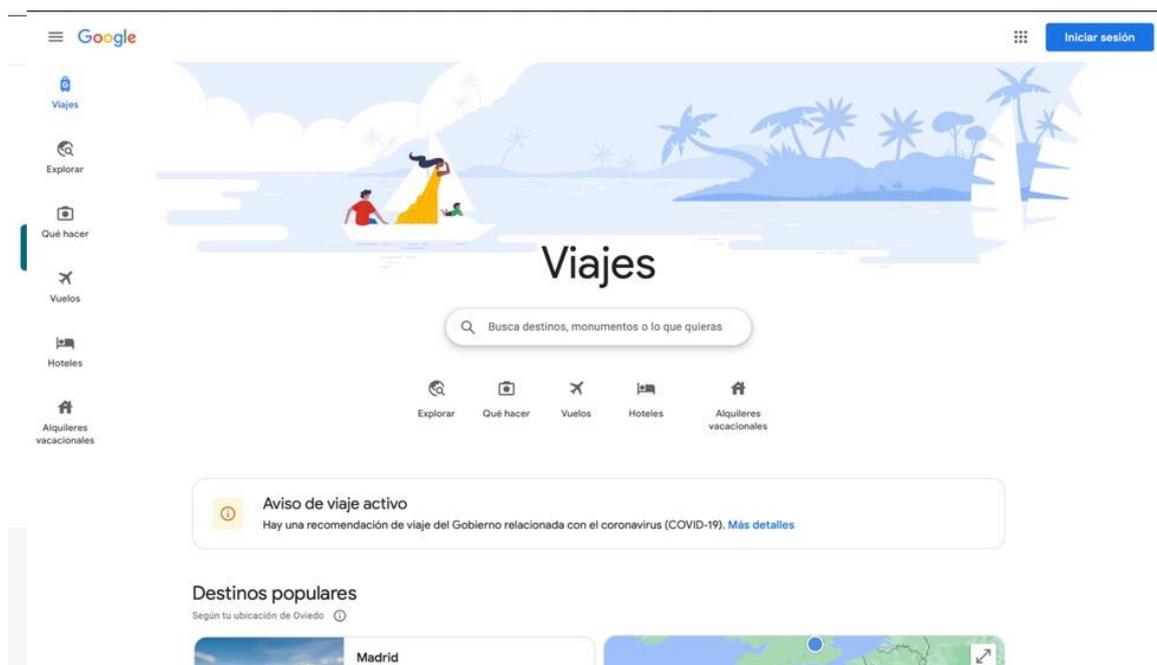


Ilustración 6. Google Travel

Capítulo 5 PLANIFICACIÓN Y GESTIÓN DEL TFG

FASE DE DESARROLLO



5.1 PLANIFICACIÓN DEL PROYECTO

5.1.1 Identificación de Interesados

A continuación, se enumeran los interesados del proyecto:

- Director del proyecto.
- Diseñadores, desarrolladores y testers.
- Usuarios finales del sistema.
- Google Firebase.
- Google Play Store.
- Mapbox.
- Navegadores web.
- Android.
- Empresas competidoras.

5.1.2 Planificación Inicial. WBS

Antes de entrar en la planificación inicial, se ha de tener en cuenta que los realizadores del proyecto estuvieron trabajando a tiempo completo y asistiendo a clases simultáneamente durante la mayor parte del trabajo. Con esto en mente, se ha calculado que se han podido dedicar aproximadamente dos horas diarias a la realización del proyecto, siendo tres los fines de semana. Se han tomado como descansos las vacaciones de Navidad y Semana Santa.

Al final, se ha estimado una duración de 292 días para considerar el proyecto terminado, empezando a finales de agosto de 2022, finalizando por tanto a finales de mayo de 2023.

La planificación del proyecto y el diagrama de Gantt se adjuntan como Anexos, ya que contienen mucha información y no sería posible mostrarlos con suficiente calidad dentro de este documento.

5.1.3 PBS

Para poder cumplir con los entregables descritos en el WBS, se ha realizado un estudio de los productos que se deben obtener, plasmados en el diagrama que se muestra a continuación.

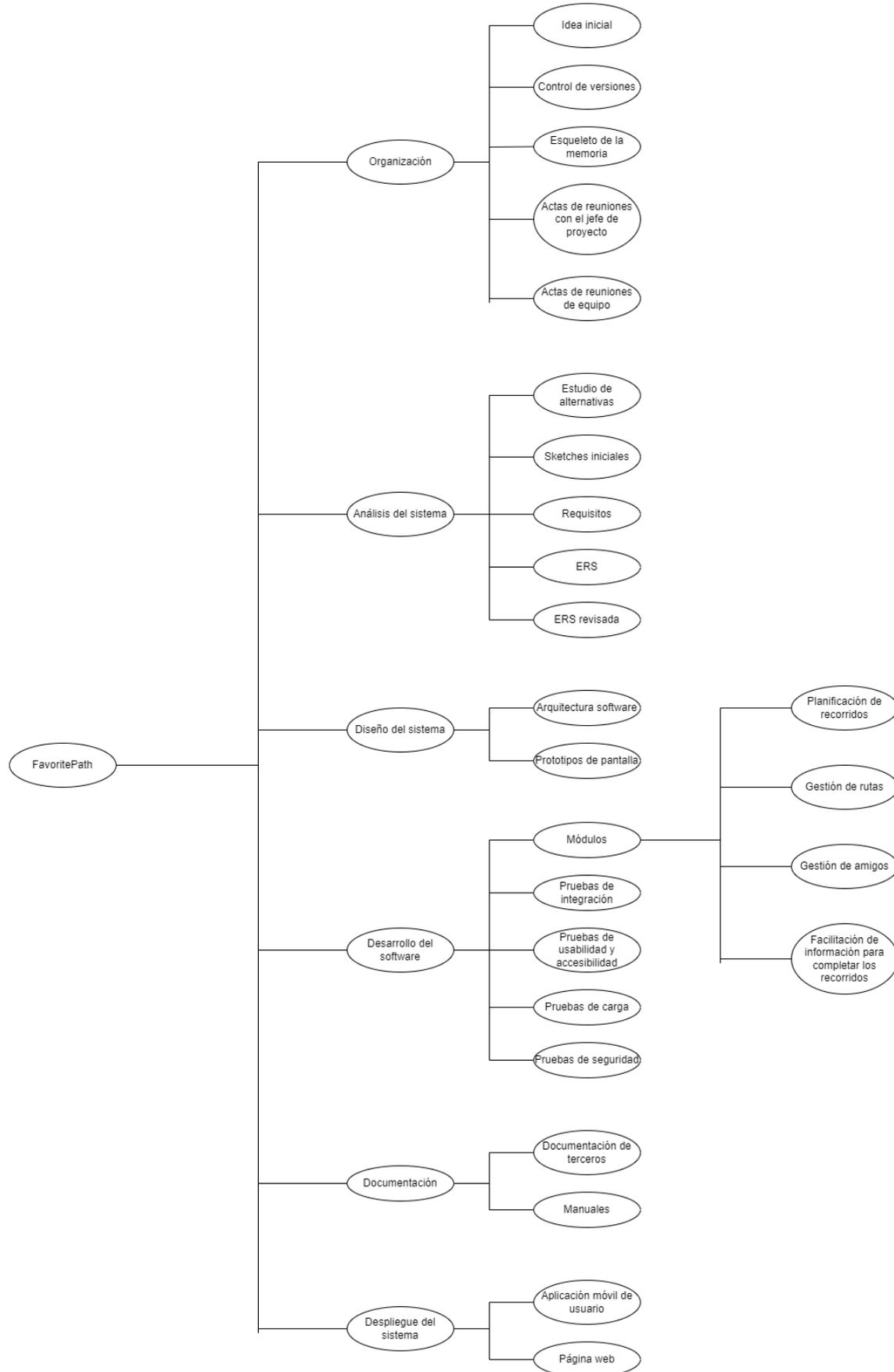


Ilustración 7. PBS

5.1.4 Riesgos

5.1.4.1 Plan de Gestión de Riesgos

El plan de gestión de riesgos ha sido especificado en los [Apéndices](#) por lo que no será explicado aquí.

5.1.4.2 Identificación de Riesgos

A continuación, se listan los diez riesgos identificados para el proyecto. La matriz de riesgos se puede encontrar como Anexo en un documento aparte.

Tabla 2. Riesgos del proyecto

<i>Nombre del riesgo</i>	<i>Descripción</i>	<i>Categoría</i>	<i>Probabilidad</i>	<i>Respuesta</i>
CAÍDA INDEFINIDA DE LAS APIS	<i>Alguna de las APIs utilizadas deja de funcionar una vez finalizado el proyecto</i>	<i>Externo</i>	<i>Muy baja</i>	<i>Asumir el riesgo. Como su probabilidad es muy baja, se esperaría a que vuelvan a estar operativas.</i>
CAMBIOS SIGNIFICATIVOS EN LAS APIS	<i>Alguna de las APIs cambia su interfaz una vez finalizado el proyecto</i>	<i>Externo</i>	<i>Muy baja</i>	<i>Mitigar el riesgo. Habría que actualizar el sistema de manera que sea compatible.</i>
ELIMINACIÓN DE UNA API	<i>Alguna de las APIs utilizadas es eliminada una vez finalizado el proyecto</i>	<i>Externo</i>	<i>Muy baja</i>	<i>Mitigar el riesgo. Se buscaría una API alternativa.</i>
RUPTURA DEL EQUIPO	<i>El equipo de desarrollo se disuelve</i>	<i>Organizacional</i>	<i>Baja</i>	<i>Asumir el riesgo. Se extinguiría el proyecto sin llegar a su finalización.</i>



CREACIÓN DE UN SISTEMA MÁS AVANZADO POR PARTE DE LA COMPETENCIA	<i>Un competidor saca al mercado un sistema igual que este, pero más potente</i>	<i>Externo</i>	<i>Media</i>	<i>Asumir el riesgo. El sistema aún podría tener usuarios. Se podrían idear mejoras para atraer otros usuarios.</i>
CAÍDA INDEFINIDA DE LOS SERVICIOS EN LA NUBE	<i>Alguno de los servicios en la nube utilizados deja de funcionar durante el desarrollo</i>	<i>Externo</i>	<i>Muy baja</i>	<i>Asumir el riesgo. Debido a la alta fiabilidad de los servicios en la nube, se esperaría a que volvieran a estar operativos.</i>
MALA PLANIFICACIÓN DEL PROYECTO	<i>En fases finales del proyecto, se hace evidente que no se ha realizado una buena estimación de tiempo y/o dinero</i>	<i>Gestión del proyecto</i>	<i>Baja</i>	<i>Mitigar el riesgo. Habría que reconducir el proyecto corrigiendo la planificación actual con ayuda de expertos.</i>
APARICIÓN DE UNA NUEVA TECNOLOGÍA MÁS ADECUADA	<i>En fases finales del desarrollo, aparece una tecnología que podría haber acelerado el proyecto</i>	<i>Tecnología</i>	<i>Baja</i>	<i>Asumir el riesgo. Ya sería tarde para cambiar de tecnología sin que afectase negativamente a la planificación.</i>
APARICIÓN DE INVERSORES	<i>En fases iniciales del proyecto, alguien se interesa</i>	<i>Organizacional</i>	<i>Muy baja</i>	<i>Asumir el riesgo. Se darían cambios que</i>

	<i>por él y contribuye económicamente</i>			<i>afectarían a todo el triángulo del proyecto.</i>
INVIABILIDAD DEL PROYECTO	<i>Durante la fase de desarrollo, se demuestra que el proyecto es inviable aun habiendo aprobado su viabilidad previamente</i>	<i>Gestión del proyecto</i>	<i>Muy baja</i>	<i>Asumir el riesgo. Si se diese este caso habría que detener el proyecto.</i>

5.1.5 Presupuesto Inicial

La hoja de cálculo con los presupuestos se puede encontrar como Anexo. Aquí se mostrará un resumen con la información más destacable. Se calcula únicamente el presupuesto de costes, necesario para viabilizar el proyecto.

5.1.5.1 Presupuesto de Costes

Para la realización del presupuesto de costes se han tenido en cuenta los costes de personal, costes indirectos y costes de producción. El coste total es de 57002€.

Para los **costes de personal** se han tenido en cuenta todos los perfiles necesarios para la realización del proyecto. Al ser dos los proyectantes y repartirse el trabajo de forma equitativa, el número siempre será de dos para cada perfil. Los sueldos se han decidido tras realizar un estudio de varias ofertas de trabajo de cada uno de los perfiles. El número de meses se obtuvo de la planificación del proyecto, considerando las actividades en las que participa cada perfil.

Personal	Número	Sueldo / Mes	Meses	Total
Analista de Sistemas	2	2.000,00 €	1	4.000,00 €
Experto en usabilidad	2	1.500,00 €	3	9.000,00 €
Programador	2	1.500,00 €	5	15.000,00 €
Tester	2	1.500,00 €	4	12.000,00 €
Arquitecto de software	2	2.500,00 €	3	15.000,00 €
Total				55.000,00 €

Ilustración 8. Costes de personal

En cuanto a los costes indirectos, se han considerado todos los gastos que no producen valor, necesarios para el desarrollo del proyecto.



Servicio	Coste mes	Meses	Total
Consumos de electricidad	80,00 €	9	720,00 €
Gastos de viaje, desplazamiento y manutención	50,00 €	9	450,00 €
Gastos en material de oficina	10,00 €	9	90,00 €
Gastos de Internet	40,00 €	9	360,00 €
Total			1.620,00 €

Ilustración 9. Costes indirectos

Por último, para los **costes de producción** se han recopilado todos los materiales utilizados, y se ha calculado su amortización. El plazo de amortización se calcula según la vida útil estimada del producto, en años. Al ser este un presupuesto sencillo, no se tiene en cuenta la tasa de depreciación anual (el cambio de valor del producto cada año). Se divide el coste del activo (el precio al momento de su compra) entre sus años de vida. Como este proyecto tiene una duración de un año aproximadamente, solo contamos con los costes para un año.

Equipo/Licencia	Unid.	Precio	Coste Total	Coste año	Tipo	Plazo
Ordenador portátil	2	700 €	1.400 €	280 €	Amortizacion	5
Teléfono móvil	2	250 €	500 €	100 €	Amortizacion	5
Ratón	2	10 €	20 €	2 €	Amortizacion	10
Total				382,00 €		

Ilustración 10. Costes de producción



5.2 CIERRE DEL PROYECTO

5.2.1 Planificación Final

Se han realizado ciertos cambios en la planificación respecto a la inicial. En primer lugar, el comienzo del proyecto se ha pospuesto hasta el inicio del curso lectivo. Además, se ha reducido el tiempo dedicado al desarrollo, debido a que habíamos tenido en cuenta por separado una fase que resultó ser completada mientras se realizaba una de las anteriores.

Al final, se ha estimado una duración de 263 días para considerar el proyecto terminado, empezando a mediados de septiembre de 2022, finalizando por tanto a finales de mayo de 2023.

La planificación del proyecto y el diagrama de Gantt finales se adjuntan como Anexos, ya que contienen mucha información y no sería posible mostrarlos con suficiente calidad dentro de este documento.

5.2.2 Presupuesto Final de Costes

A pesar de que la planificación ha sufrido cambios, el presupuesto se mantiene tal y como se calculó inicialmente en el punto [5.1.5 Presupuesto Inicial](#), ya que los únicos cambios serían en los recursos humanos, y estos habrían sido contratados por meses, de manera que su remuneración sería la misma.

5.2.3 Informe de Lecciones Aprendidas

Durante el desarrollo del proyecto nos hemos ido encontrando con diversos obstáculos que nos han dejado lecciones valiosas. Se detallan a continuación:

La importancia de la planificación previa: la planificación es una de las partes menos valoradas en cualquier tipo de proyecto. Siempre buscamos resultados, y para ello a veces se cree más conveniente comenzar cuanto antes a producir que reservar un tiempo suficiente para el análisis cuidadoso de las necesidades del proyecto. Durante este trabajo nos hemos fijado en la importancia de la planificación, ya que sin ella el desarrollo nos hubiese llevado mucho más tiempo, y habríamos realizado muchas tareas que no hubiesen sido necesarias. Así mismo, la elección previa de las herramientas adecuadas nos ha permitido trabajar desde el primer momento de la mejor forma, evitando tener que rehacer trabajo una vez finalizado por causa de incompatibilidades.

Las dos caras del trabajo en equipo: realizar un proyecto en equipo puede parecer, en un primer momento, más sencillo que trabajar solo. Sin embargo, hay muchos factores que se han de tener en cuenta cuando se involucra más de una persona, dado que hay que saber cómo repartir el trabajo de forma eficiente, aprender a delegar, llevar un ritmo adecuado para no hacer esperar a los demás



integrantes, y mantener una comunicación fluida y eficiente para que el proyecto salga adelante. En definitiva, cuantas más personas formen parte del equipo, mayor tiempo habrá que dedicar exclusivamente a la comunicación, tiempo que, si se gestiona de forma errónea, puede ser tiempo perdido; pero, por otro lado, no dedicar tiempo a esta labor haría fracasar el proyecto.

Capítulo 6 ANÁLISIS DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO

ASI



6.1 ASI 1: ESPECIFICACIÓN DE REQUISITOS

La especificación de requisitos ha sido adjuntada en los [Apéndices](#), junto al diagrama de contexto del sistema, por lo que no será explicada aquí.

6.2 ASI 3: IDENTIFICACIÓN DE SUBSISTEMAS DE ANÁLISIS

6.2.1 Descripción de los Subsistemas

FavoritePath se compone de diferentes subsistemas que trabajan en conjunto. Estos subsistemas no se estructuran de igual forma en la aplicación móvil como en la página web, pero se pueden distinguir fácilmente. A continuación, se detallarán cada uno de ellos.

6.2.1.1 *Subsistema de acceso a datos*

Aquí se concentran todas las llamadas a la API de la base de datos. En la web, estas se encuentran separadas en su fichero JavaScript, mientras que en la app móvil se encuentran repartidas por las diferentes clases según se vayan necesitando.

6.2.1.2 *Subsistema de acceso a mapas*

En este subsistema se agrupan las llamadas a la API del proveedor de mapas. Tanto en la web como en la app, al igual que en el apartado anterior, se encontrarán separadas en su propio fichero y repartidas por las clases respectivamente.

6.2.1.3 *Subsistema de autenticación*

Un pequeño subsistema se encarga de las llamadas al servicio de autenticación, que es externo al sistema.

6.2.1.4 *Subsistema de negocio*

Este subsistema es el encargado de todas las operaciones internas del sistema. Se vale de los datos recogidos de la base de datos y de los introducidos por los usuarios. Contiene todas las operaciones realizadas sobre las entidades (amigos, rutas y colecciones).

6.2.1.5 Subsistema de vistas

Este subsistema recoge la parte visual tanto de la aplicación móvil como del portal web. Es la encargada de mostrar al usuario los datos recopilados por los demás subsistemas.

6.2.2 Descripción de las Interfaces entre Subsistemas

El siguiente diagrama representa las conexiones entre los diferentes subsistemas de FavoritePath.

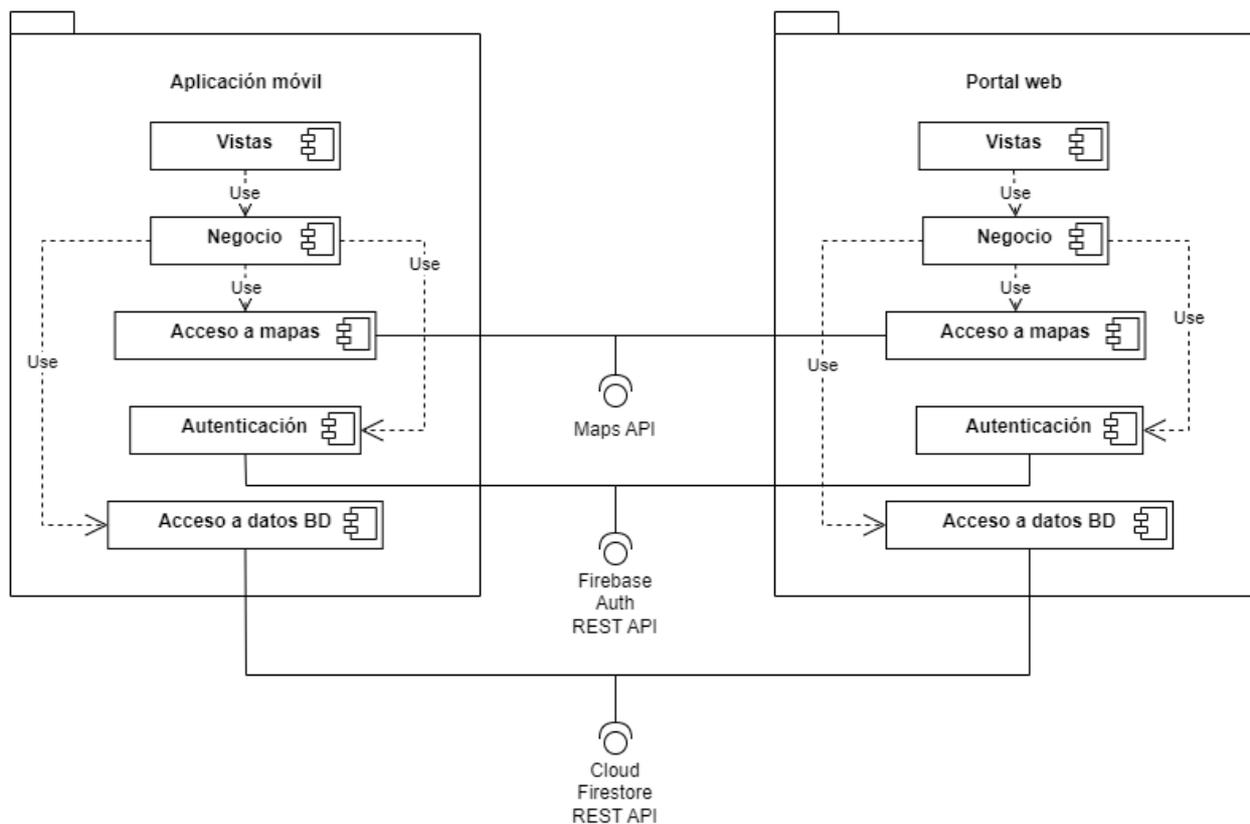


Ilustración 11. Interfaces entre subsistemas

En primer lugar, se puede contemplar cómo el portal web y la aplicación móvil están distribuidos de igual manera, puesto que ambos ofrecen los mismos servicios y deben acceder a los mismos datos. Las diferentes pantallas (subsistema de vistas) reciben de la lógica de negocio los datos que deben mostrar. A su vez, el negocio se vale de los subsistemas de acceso a mapas, autenticación y acceso a la base de datos para obtener toda la información que necesita gestionar. Cada uno de estos tres últimos subsistemas hace uso de una API externa para acceder a los datos.

6.3 ASI 5: ANÁLISIS DE CLASES

En este apartado solo se hará referencia a la aplicación móvil. Esto es porque el portal web se ha programado sin seguir la Orientación a Objetos y no aplica porque carece de clases.

6.3.1 Diagrama de Clases

Debido a que en otro apartado posterior mostraremos un diagrama de paquetes del sistema, ahí se podrá visualizar el diagrama de clases de una forma general, mientras que, en el siguiente punto, desglosaremos cada clase con una descripción de esta. La visión general del diagrama se encuentra en el apartado [7.3.1](#).

6.3.2 Descripción de las Clases

6.3.2.1 Clases Adapter

Tabla 3. ImagePagerAdapter

Nombre de la Clase: ImagePagerAdapter
Descripción: Clase para gestionar las imágenes de las rutas
-mImages: List<String>
+getCount(): int
+ isViewFromObject(View, Object): boolean
+instantiateltem(ViewGroup, int): Object
+destroyItem(ViewGroup, int, Object): void

Tabla 4. ListAmigosAdapter

Nombre de la Clase: ListAmigosAdapter
Descripción: Clase para gestionar los amigos de los usuarios
-amigos: List<Amigo>
-listener: OnItemClickListener
+onCreateViewHolder(ViewGroup, int): AmigoViewHolder
+onBindViewHolder(AmigoViewHolder, int): void
+getItemCount(): int

Tabla 5. ListColeccionesAdapter

Nombre de la Clase: ListColeccionesAdapter
Descripción: Clase para gestionar las colecciones de los usuarios

-colecciones: List<Coleccion>
-listener: OnItemClickListener

+onCreateViewHolder(ViewGroup, int): ColeccionViewHolder
+onBindViewHolder(ColeccionViewHolder, int): void
+getItemCount(): int

Tabla 6. ListRutasAdapter

Nombre de la Clase: ListRutasAdapter
Descripción: Clase para gestionar las rutas de los usuarios
-rutas: List<Ruta> -listener: OnItemClickListener +onCreateViewHolder(ViewGroup, int): RutaViewHolder +onBindViewHolder(RutaViewHolder, int): void +getItemCount(): int

Tabla 7. ListShareAdapter

Nombre de la Clase: ListShareAdapter
Descripción: Clase para gestionar los amigos de los usuarios para compartir la ruta
-amigos: List<Amigo> -listener: OnItemClickListener +onCreateViewHolder(ViewGroup, int): AmigoViewHolder +onBindViewHolder(AmigoViewHolder, int): void +getItemCount(): int

Tabla 8. ListSolicitudesAdapter

Nombre de la Clase: ListSolicitudesAdapter
Descripción: Clase para gestionar las solicitudes de los usuarios
-solicitudes: List<Solicitud> -listener: OnItemClickListener +onCreateViewHolder(ViewGroup, int): SolicitudViewHolder +onBindViewHolder(SolicitudViewHolder, int): void +getItemCount(): int



6.3.2.2 Clases Modelo

Tabla 9. Amigo

Nombre de la Clase: Amigo
Descripción: Clase para gestionar los datos de un amigo
-nombre: String -descripcion: String +setNombre(String): void +getNombre(): String +getDescripcion(): String

Tabla 10. Coleccion

Nombre de la Clase: Coleccion
Descripción: Clase para gestionar los datos de una colección
-titulo: String +getTitulo(): String +writeToParcel(Parcel, int): void +describeContents(): int

Tabla 11. Ruta

Nombre de la Clase: Ruta
Descripción: Clase para gestionar los datos de una ruta
#id: String #titulo: String #origen: String #descripcion: String #puntos: List<GeoPoint> #imagenes: List<String> #circular: boolean #publica: boolean +getId(): String +setId(String): void +getTitulo(): String +setTitulo(String): void +getOrigen(): String +setOrigen(String): void +getDescripcion(): String +setDescripcion(String): void +getImagenes(): List<String> +setImagenes(List<String>): void +writeToParcel(Parcel, int): void +describeContents(): int



Tabla 12. Solicitud

Nombre de la Clase: Solicitud
Descripción: Clase para gestionar los datos de una solicitud
-titulo: String -rutald: String +getTitulo(): String +getRutald(): String +writeToParcel(Parcel, int): void +describeContents(): int

Tabla 13. User

Nombre de la Clase: User
Descripción: Clase para gestionar los datos de un usuario
-nombre: String -descripcion: String +setNombre(String): void +getNombre(): String +getDescripcion(): String

6.3.2.3 Clases Adapter

Tabla 14. AmigosFragment

Nombre de la Clase: AmigosFragment
Descripción: Clase para mostrar el fragmento de los amigos en la pantalla principal y la gestión de ellos.
#solicitud: EditText #solicitudBtn: Button #mAuth: FirebaseAuth #db: FirebaseFirestore #reference: DocumentReference #amigoListView: RecyclerView #amigos: ArrayList<Amigo> +onCreate(Bundle): void +onCreateView(LayoutInflater, ViewGroup, Bundle): View -mandarSolicitud(): void -existeUsuarioSolicitud(String): void -borrarAmigo(Amigo): void



```
-init(View): void  
-configureView(): void  
-addAdapter(ArrayList<Amigo>): void  
-cargarDatos(): void  
+updateAmigos(Amigo): void  
+initializeAmigos(): void
```

Tabla 15. AyudaFragment

Nombre de la Clase: AyudaFragment
Descripción: Clase para mostrar el fragmento de la ayuda en la pantalla principal.
+onCreate(Bundle): void
+onCreateView(LayoutInflater, ViewGroup, Bundle): View

Tabla 16. ColeccionesFragment

Nombre de la Clase: ColeccionesFragment
Descripción: Clase para mostrar el fragmento de las colecciones y rutas en la pantalla principal, además de administrar la gestión de las colecciones y rutas.
#vista: View #coleccion: EditText #coleccionBtn: Button #rutaBtn: Button #titulo: String #mAuth: FirebaseAuth #db: FirebaseFirestore #reference: DocumentReference #recyclerView: RecyclerView #rutas: ArrayList<Ruta> #colecciones: ArrayList<Colección> +RUTA: int +onActivityResult(int, int, Intent): void +onCreate(Bundle): void +onCreateView(LayoutInflater, ViewGroup, Bundle): View -crearRuta(): void -crearColeccion(): void -borrarColeccion(Coleccion): void -abrirColeccion(Coleccion): void -init(View): void -configureView(): void -addColeccionAdapter(ArrayList<Coleccion>): void -cargarDatosColecciones(): void +updateColecciones(Coleccion): void +initializeColecciones(): void -borrarRuta(Ruta): void -abrirRuta(Ruta): void +updateRutas(Ruta): void -addRutaAdapter(ArrayList<Ruta>): void +initializeRutas(): void



+onBackPressed(): boolean

Tabla 17. InicioFragment

Nombre de la Clase: InicioFragment
Descripción: Clase para mostrar el fragmento de inicio en la pantalla principal.
+onCreate(Bundle): void
+onCreateView(LayoutInflater, ViewGroup, Bundle): View

Tabla 18. PerfilFragment

Nombre de la Clase: PerfilFragment
Descripción: Clase para mostrar el fragmento del perfil del usuario en la pantalla principal.
#nombre: EditText
#descripcion: EditText
#guardarBtn: FloatingActionButton
-mAtuh: FirebaseAuth
#db: FirebaseFirestore
#reference: DocumentReference
#navView: ImageView
#perfilView: ImageView
#nombreView: TextView
#descripcionView: TextView
+onCreate(Bundle): void
+onCreateView(LayoutInflater, ViewGroup, Bundle): View
-actualizarPerfil(): void
-updateNavHeader(String, String): void
-updateUI(): void
+init(View): void

Tabla 19. SolicitudesFragment

Nombre de la Clase: SolicitudesFragment
Descripción: Clase para mostrar el fragmento de las solicitudes en la pantalla principal y de la gestión de ellas.
#mAtuh: FirebaseAuth
#db: FirebaseFirestore
#reference: DocumentReference
#solicitudListView: RecyclerView
#solicitudes: ArrayList<Solicitud>
+onCreate(Bundle): void
+onCreateView(LayoutInflater, ViewGroup, Bundle): View
-init(View): void
-aceptarSolicitud(Solicitud): void
-borrarSolicitud(Solicitud): void



```
-configureView(): void
-addAdapter(ArrayList<Solicitud>): void
-cargarDatos(): void
+updateSolicitudes(Solicitud): void
+initializeSolicitudes(): void
```

6.3.2.4 Clases Activity

Tabla 20. LoginActivity

Nombre de la Clase: LoginActivity
Descripción: Clase para gestionar la aparición de la pantalla de inicio de sesión como la gestión de registro e inicio de sesión de la aplicación.
<pre>-RC_SIGN_IN: int -PASSWORD_MIN: int -emailText: EditText -passwordText: EditText -loginButton: Button -registerButton: Button -googleButton: SignInButton -mAuth: FirebaseAuth -db: FirebaseFirestore #mGoogleSignInClient: GoogleSignInClient #onCreate(Bundle): void +onActivityResult(int, int, Intent): void -firebaseAuthWithGoogle(String): void -loginUser(): void -registerUser(): void +createUserInFirestore(String): void -signInWithGoogle(): void +init(): void</pre>

Tabla 21. MainActivity

Nombre de la Clase: MainActivity
Descripción: Clase para gestionar la aparición de la pantalla principal como la gestión de cambio de opciones en el menú y de cerrar sesión.
<pre>-REQUEST_STORAGE_PERMISSION: int #drawerLayout: DrawerLayout #navigationView: NavigationView #toolbar: Toolbar -mAuth: FirebaseAuth -db: FirebaseFirestore #nombreView: TextView #descripcionView: TextView +onRequestPermissionsResult(int, String[], int[]): void</pre>



```
+onBackPressed(): void
#onCreate(Bundle): void
-navigation(): void
+logout(View): void
+cargarFragment(Fragment): void
-initialize(): void
+actualizarNavHeader(): void
+abrirColecciones(): void
-checkPermisos(): void
```

Tabla 22. RutaActivity

Nombre de la Clase: RutaActivity
Descripción: Clase para gestionar la aparición de la pantalla de las rutas como la gestión de creación y actualización de ellas.
<pre>-REQUEST_CODE_SELECT_IMAGE: int #mAuth: FirebaseAuth #db: FirebaseFirestore #storage: FirebaseStorage #reference: DocumentReference #saveBtn: FloatingActionButton #rutald: String #idAuxiliar: String #origen: String #nombreColeccion: String #titulo: EditText #descripcion: EditText #publico: CheckBox #circular: CheckBox #viewPager: ViewPager #imageBtn: Button #initRutaBtn: Button #mImageUris: ArrayList<String> #imagenes: List<String> #mapView: MapView #mapboxMap: MapboxMap #puntos: ArrayList<GeoPoint> +onBackPressed(): void #onActivityResult(int, int, Intent): void #onCreate(Bundle): void -iniciarRuta(): void -guardarImagenes(): void -guardarRuta(): void -crearRuta(): void -actualizarRuta(boolean): void +initializeRuta(Bundle): void -initMapa(Bundle): void -init(): void</pre>



Tabla 23. ShareActivity

Nombre de la Clase: ShareActivity
Descripción: Clase para gestionar la aparición de la pantalla para compartir rutas como la aparición del mapa para realizar la ruta.
#mAuth: FirebaseAuth #db: FirebaseFirestore #reference: DocumentReference #amigoListView: RecyclerView #amigos: ArrayList<Amigo> #iniciarRutaBtn: Button #rutald: String #circular: Boolean #puntos: ArrayList<GeoPoint> #onCreate(Bundle): void -iniciarRuta(): void -shareRuta(Amigo): void -init(): void -configureView(): void -addAdapter(ArrayList<Amigo>): void -cargarDatos(): void -updateAmigos(Amigo): void -getDatosRuta(): void -initializeAmigos(): void -navigationRoute(): void

Tabla 24. SplashActivity

Nombre de la Clase: SplashActivity
Descripción: Clase para mostrar al arranque de la aplicación una pantalla con el logotipo de FavoritePath.
#onCreate(Bundle): void

6.4 ASI 8: DEFINICIÓN DE INTERFACES DE USUARIO

6.4.1 Diseño inicial de la Interfaz

En esta sección mostraremos los bocetos iniciales que habíamos diseñado para el sistema, que no necesariamente se adecuan al diseño final de las aplicaciones. Los bocetos iniciales se hicieron antes de tomar requisitos, como punto de partida de la idea inicial. Posteriormente se fueron refinando en

varias iteraciones, hasta alcanzar el diseño final de las vistas, que se puede consultar en el siguiente apartado ([6.4.2 Descripción de la Interfaz](#)).

6.4.1.1 Diseño inicial de la Interfaz Web

A continuación, se muestran los bocetos iniciales para el portal web:

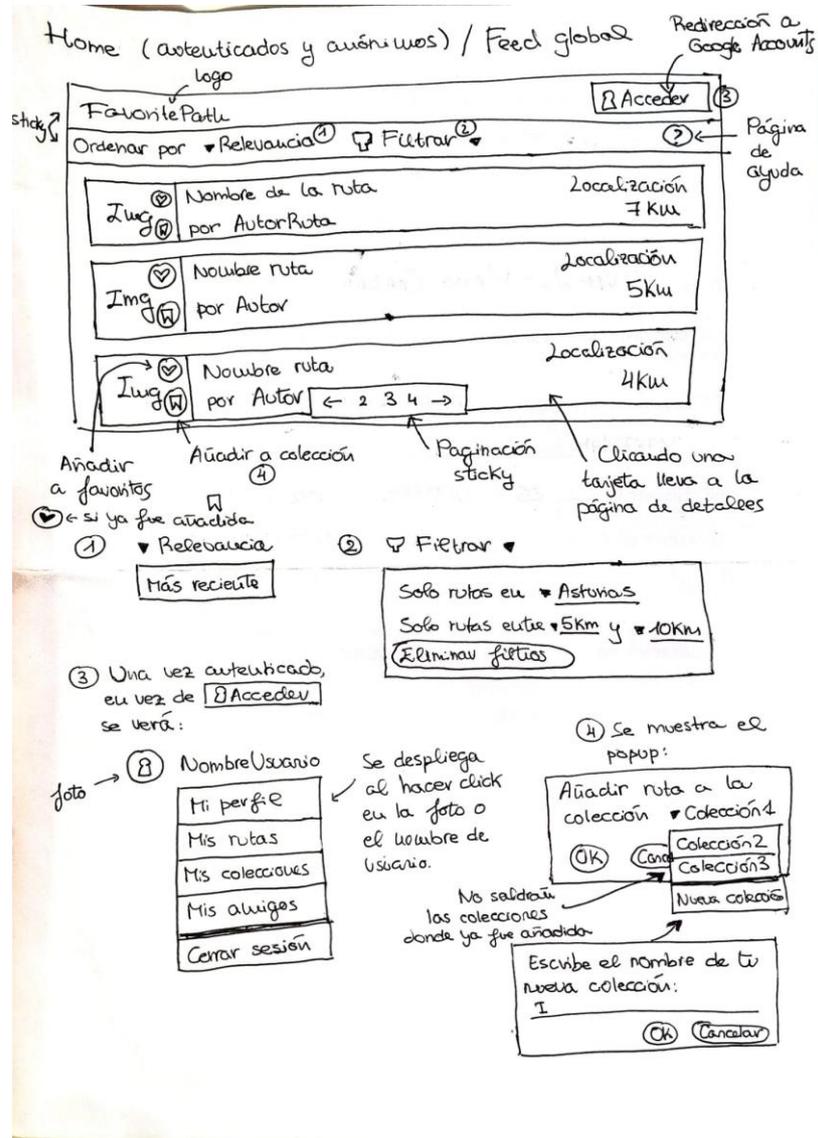
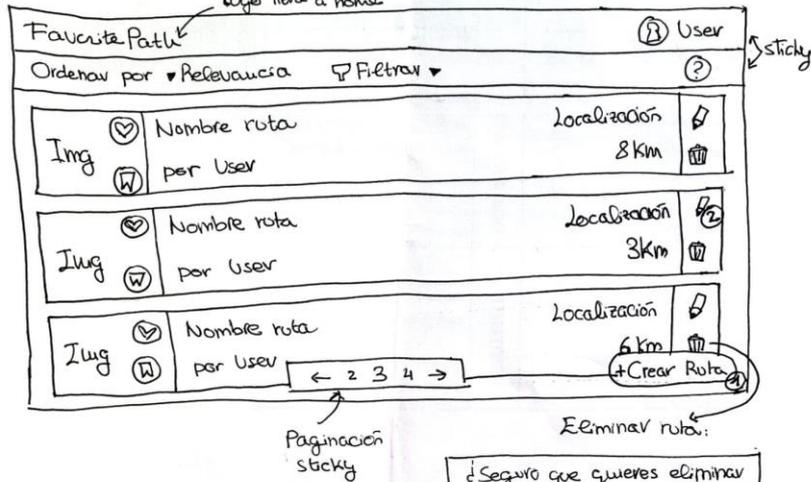
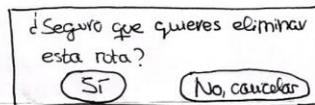


Ilustración 12. Homepage

Mis Rutas (autenticados) / Feed privado
logo lleva a home



Página muy similar a House, pero con rutas creadas por el usuario
 Opciones Editar, Eliminar y + Crear Ruta

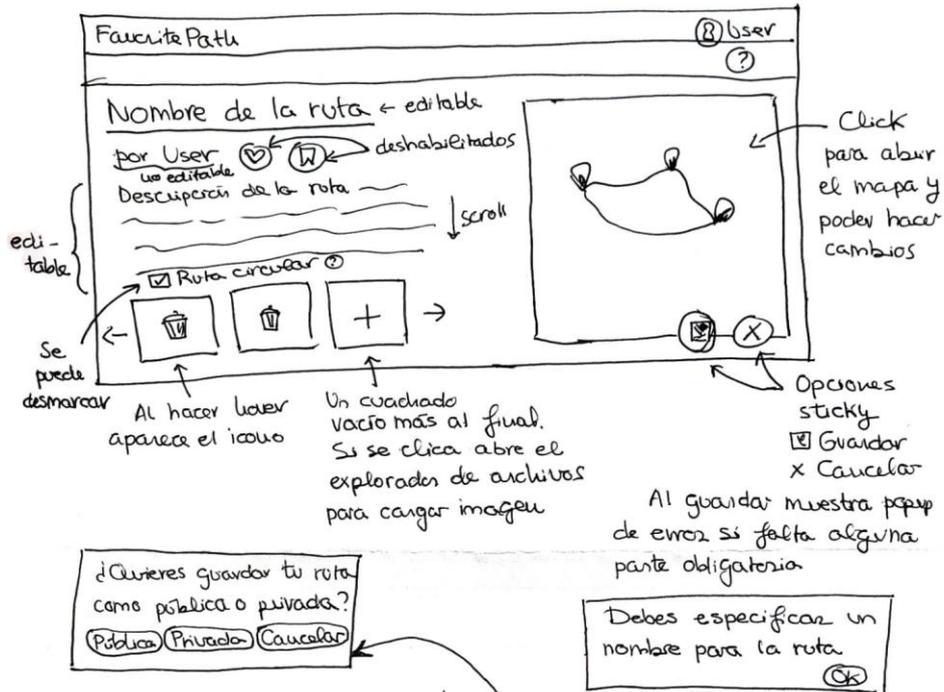


Detalles Ruta (autenticados y anónimos)



Ilustración 13. Rutas del usuario y Detalle de la ruta

Editar Ruta (autenticados)



Crear Ruta (autenticados)

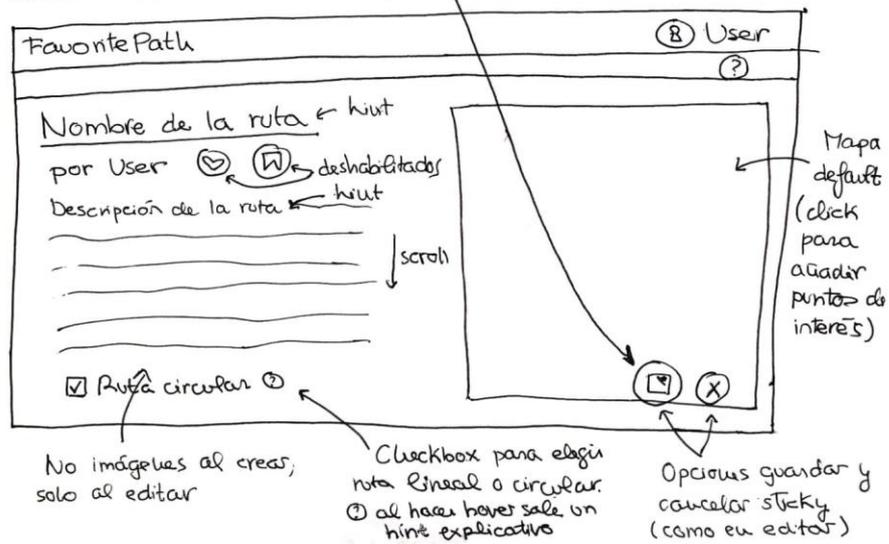
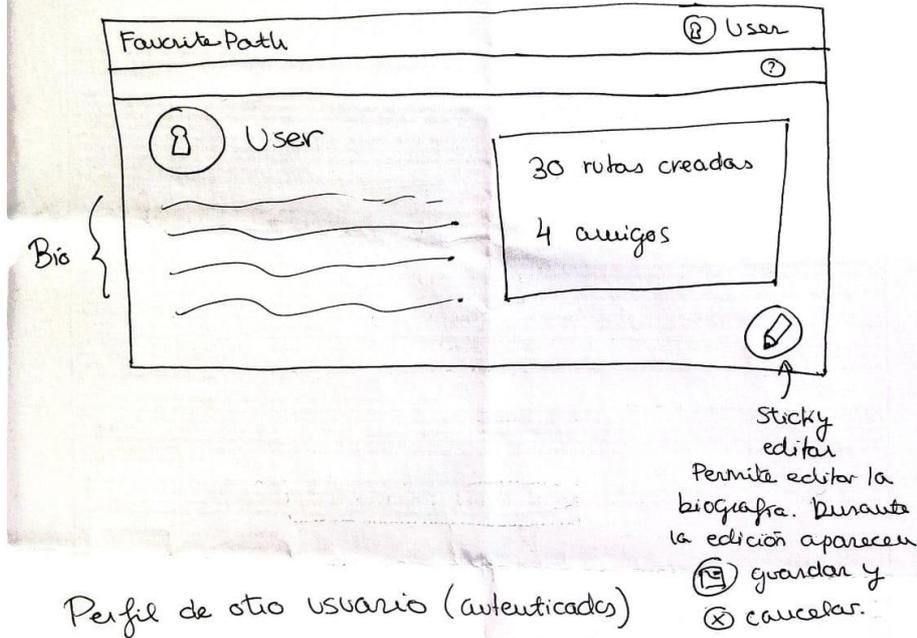


Ilustración 14. Edición y creación de rutas

Perfil propio (autenticados)



Perfil de otro usuario (autenticados)

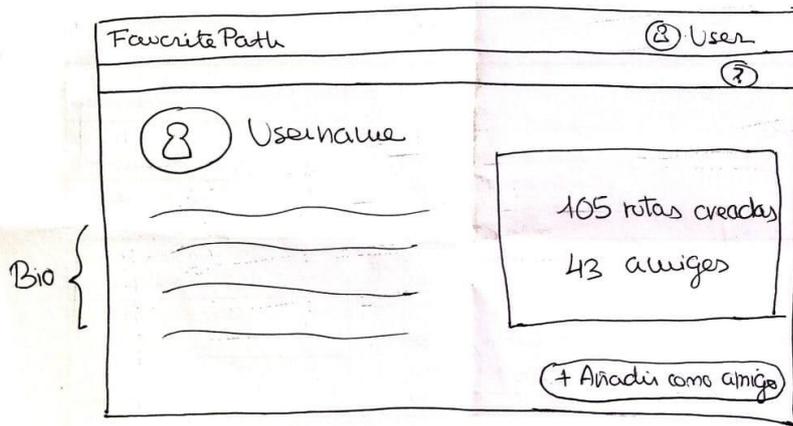
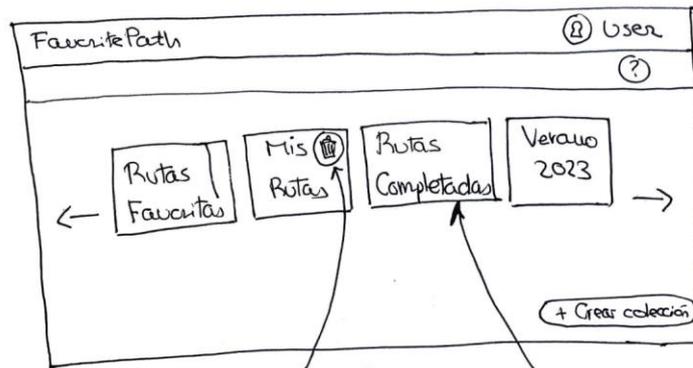


Ilustración 15. Perfiles de usuario

Colecciones (autenticadas)



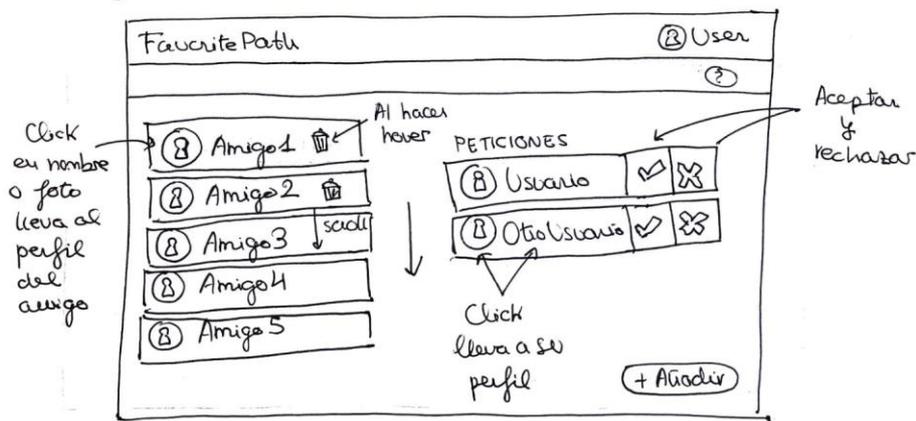
Opción de eliminar al hacer hover

¿Seguro que quieres eliminar esta colección?

Sí No, cancelar

El click en una colección lleva a una lista con feeds como el resto de feeds. Se podrán editar y eliminar solo las rutas propias, y no aparecerá la opción de crear ruta. En su lugar habrá una opción Eliminar Colección

Amigos (autenticados)



Click en nombre o foto lleva al perfil del amigo

Al hacer hover

Aceptar y rechazar

Click lleva a su perfil

Ilustración 16. Colecciones y lista de amigos

6.4.1.2 Diseño inicial de la Interfaz de la App

A continuación, se muestran los bocetos iniciales para la aplicación móvil:

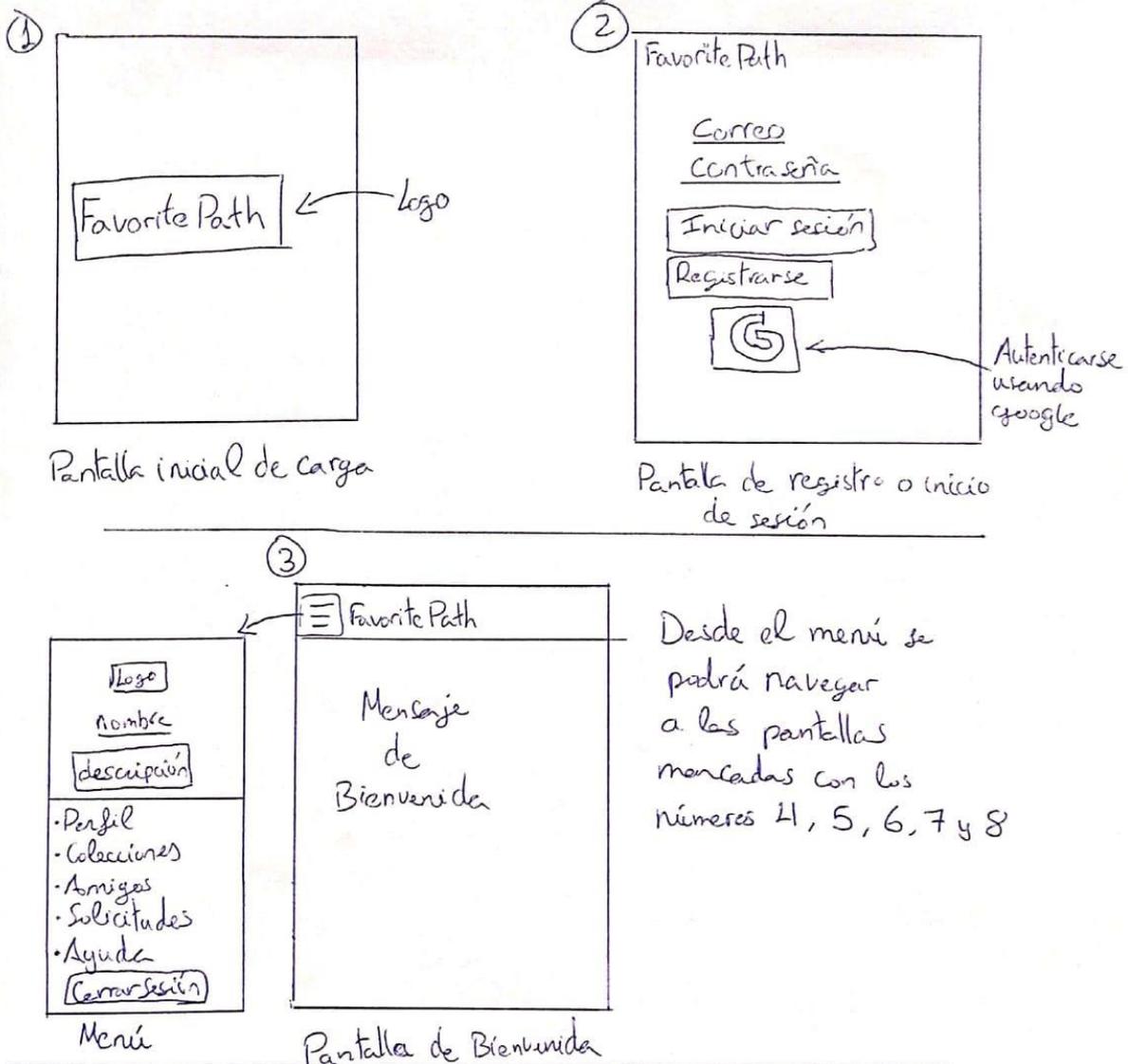
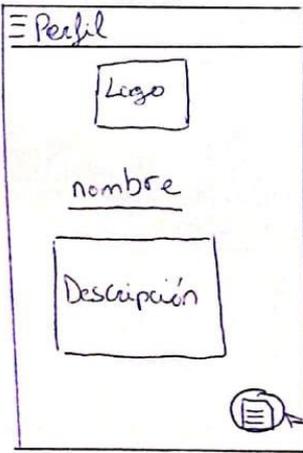


Ilustración 17. Pantallas móvil

④



Pantalla del perfil

Botón
para guardar
cambios

⑤



Pantalla de ayuda

Ilustración 18. Pantallas móvil

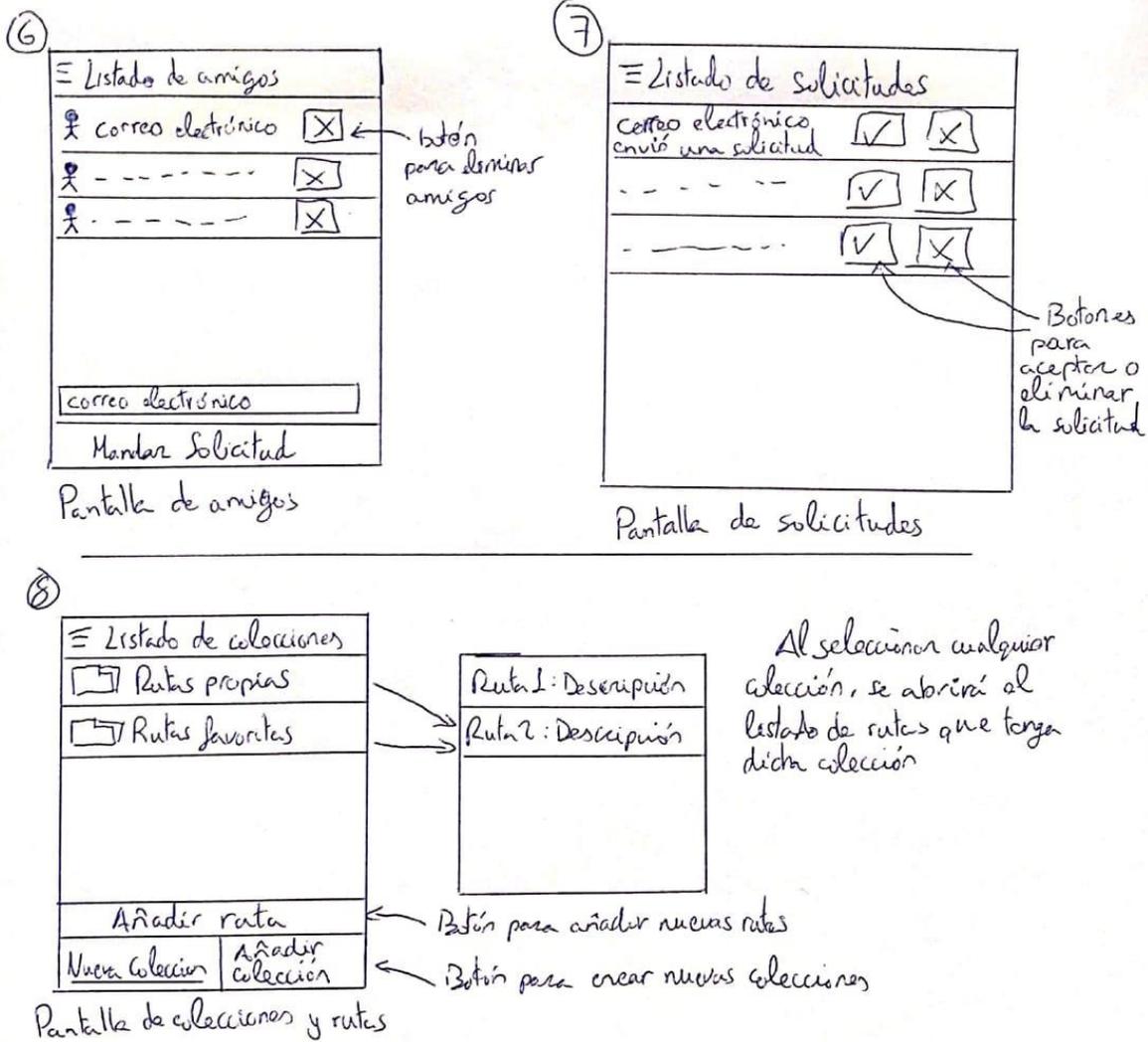


Ilustración 19. Pantallas móvil

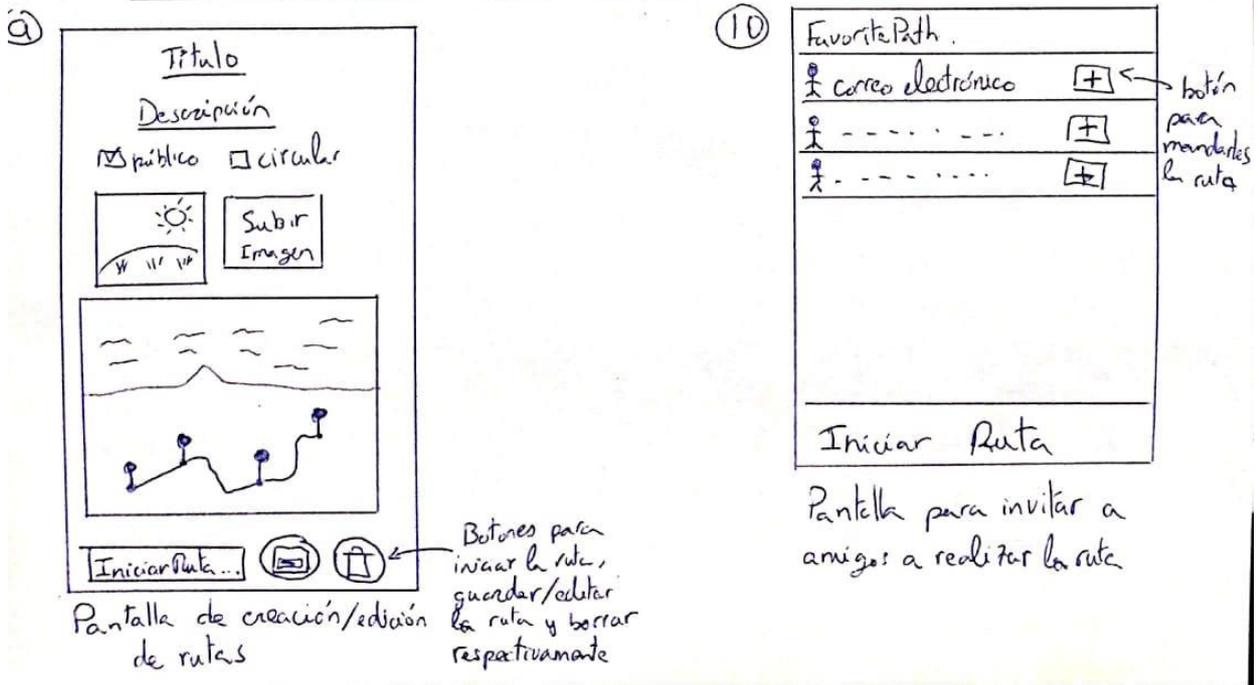


Ilustración 20. Pantallas móvil

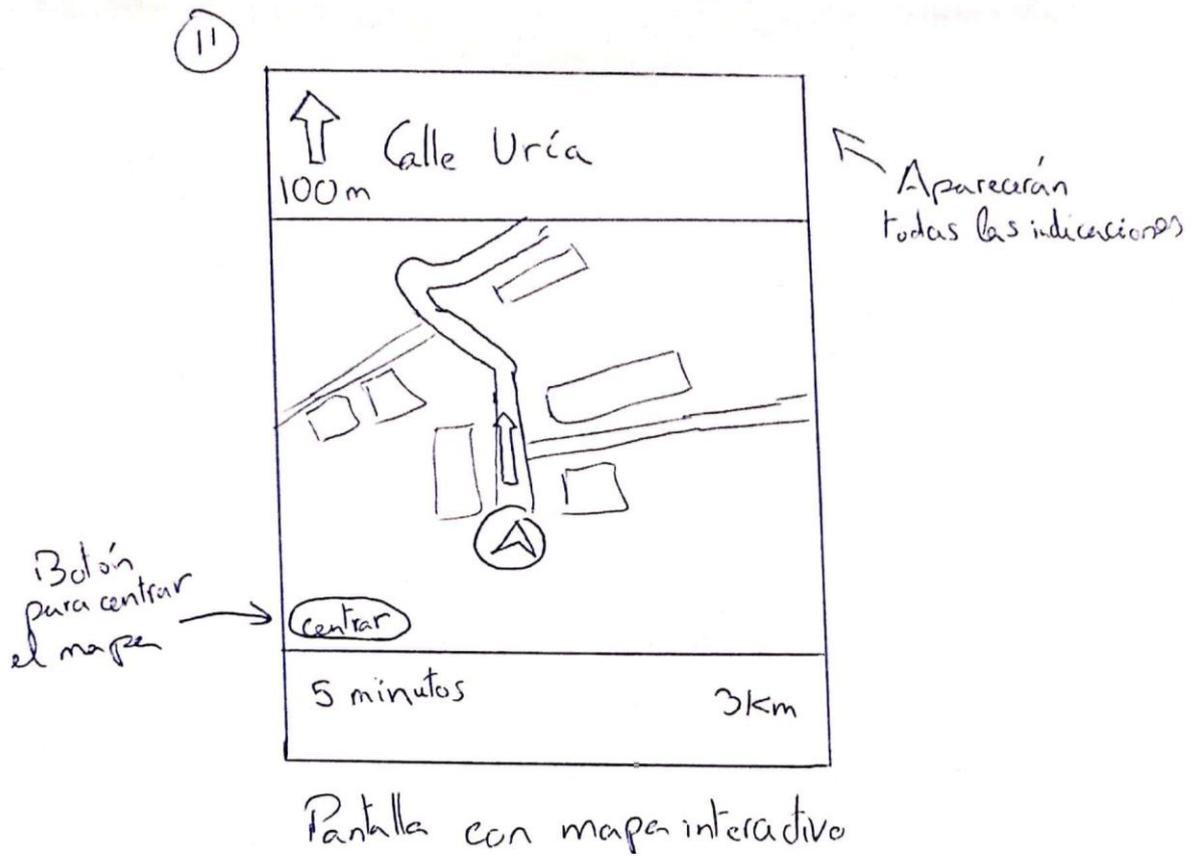


Ilustración 21. Pantallas móvil

6.4.2 Descripción de la Interfaz

En este apartado procederemos a mostrar el resultado final de las interfaces de usuario, separándolo por la parte web y la aplicación móvil, dentro del apartado [8.6.1 Manual de usuario](#), puede encontrar una explicación más detallada de lo que ofrece cada pantalla separada en funcionalidades.

Dentro de la aplicación móvil:



Ilustración 22. SplashActivity



Ilustración 23. LoginActivity



Ilustración 24. MainActivity

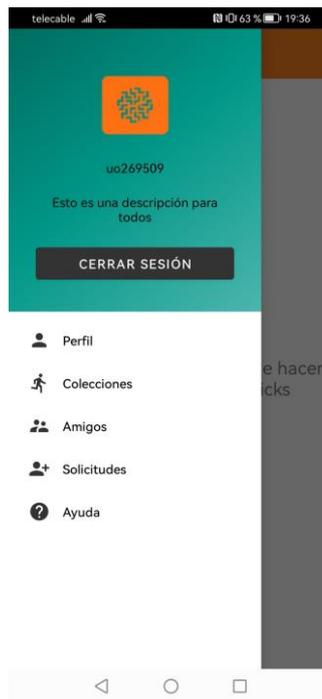


Ilustración 25. Menú Principal



Ilustración 26. Edición de perfil



Ilustración 27. Listado de colecciones

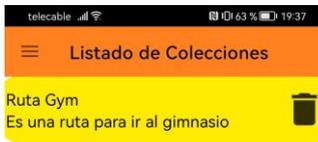


Ilustración 28. Listado de rutas



Ilustración 29. Lista de amigos



Ilustración 30. Listado de solicitudes



Ilustración 31. Detalle de ruta



Ilustración 32. Compartir rutas



Ilustración 33. Mapa interactivo

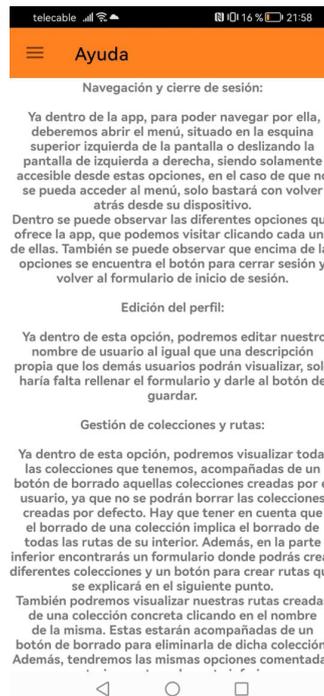


Ilustración 34. Ayuda



6.4.3 Descripción del Comportamiento de la Interfaz

Las interfaces de usuario de la página web y la aplicación móvil tratarán de ser lo más intuitivas posible para los usuarios. Para ello, se aplicarán las convenciones seguidas por otros sistemas similares. Además, las pantallas mostrarán la mínima cantidad de información en cada momento, de manera que el usuario no se pierda entre demasiadas opciones. La limpieza y claridad son claves en el diseño.

Siempre que el usuario deba introducir información, se mostrarán tooltips y se validarán los campos para evitar confusiones y errores. Cualquier problema referente a las acciones realizadas por el usuario será mostrado en pantalla, para que este entienda lo que sucede. Siempre aparecerá accesible un enlace a la página de ayuda, donde se explica paso a paso cualquier funcionalidad del sistema.

6.4.4 Diagrama de Navegabilidad

El siguiente diagrama muestra la navegabilidad entre pantallas en la página web. El acceso a la página de inicio está siempre disponible a través del logo desde cualquier pantalla. Lo mismo sucede con la página de ayuda. El número de pantallas a las que se puede acceder cambia según si el usuario se ha autenticado o no. Las pantallas de libre acceso aparecen sin relleno, mientras que las que requieren de autenticación se muestran en un color rojizo. Cuando un usuario se encuentra autenticado, desde todas las pantallas podrá acceder a un desplegable con accesos a sus páginas privadas (“Mi perfil”, “Feed privado”, “Mis colecciones” y “Mis amigos”). En el diagrama solo se muestran los accesos existentes en el menú para la página de inicio, para hacer más entendibles el resto de las conexiones.

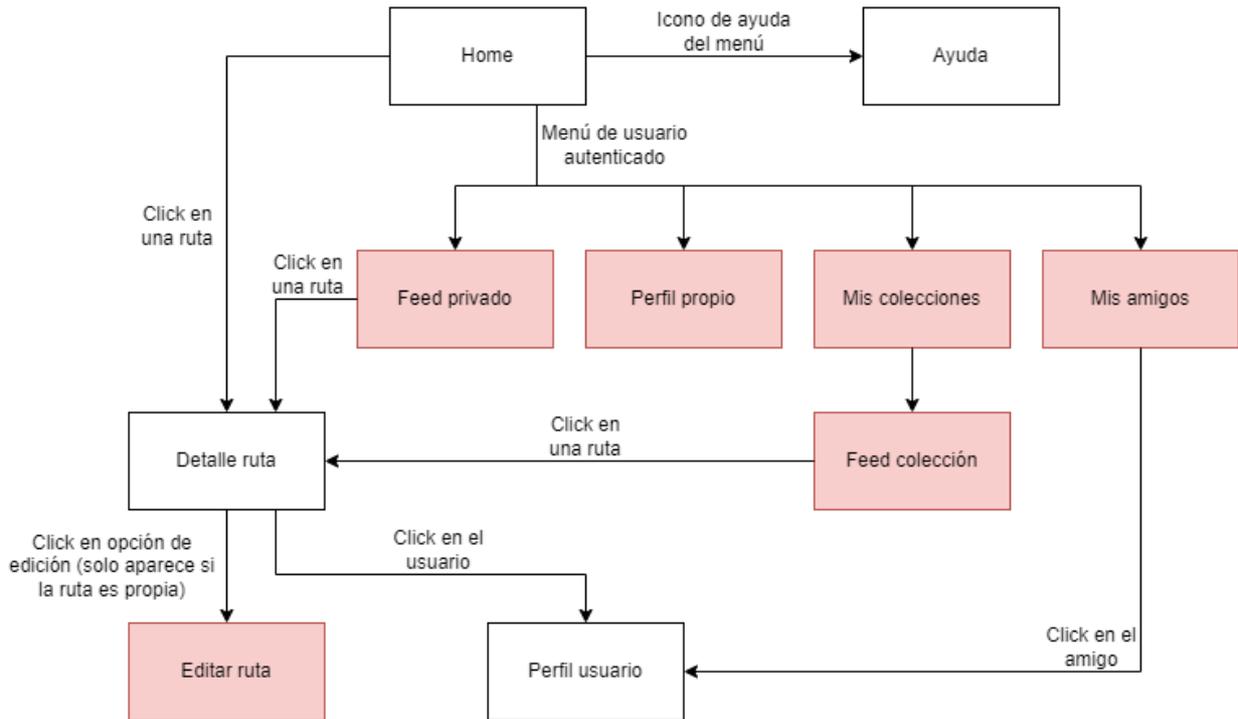


Ilustración 35. Diagrama de navegabilidad web

A continuación, se muestra el diagrama de navegabilidad para la aplicación móvil. En este caso, el usuario deberá autenticarse para poder usar la aplicación y navegar por todas las pantallas, resaltando que deberá aceptar permisos al entrar en las colecciones para seguir navegando por ellas o se le denegará el acceso.

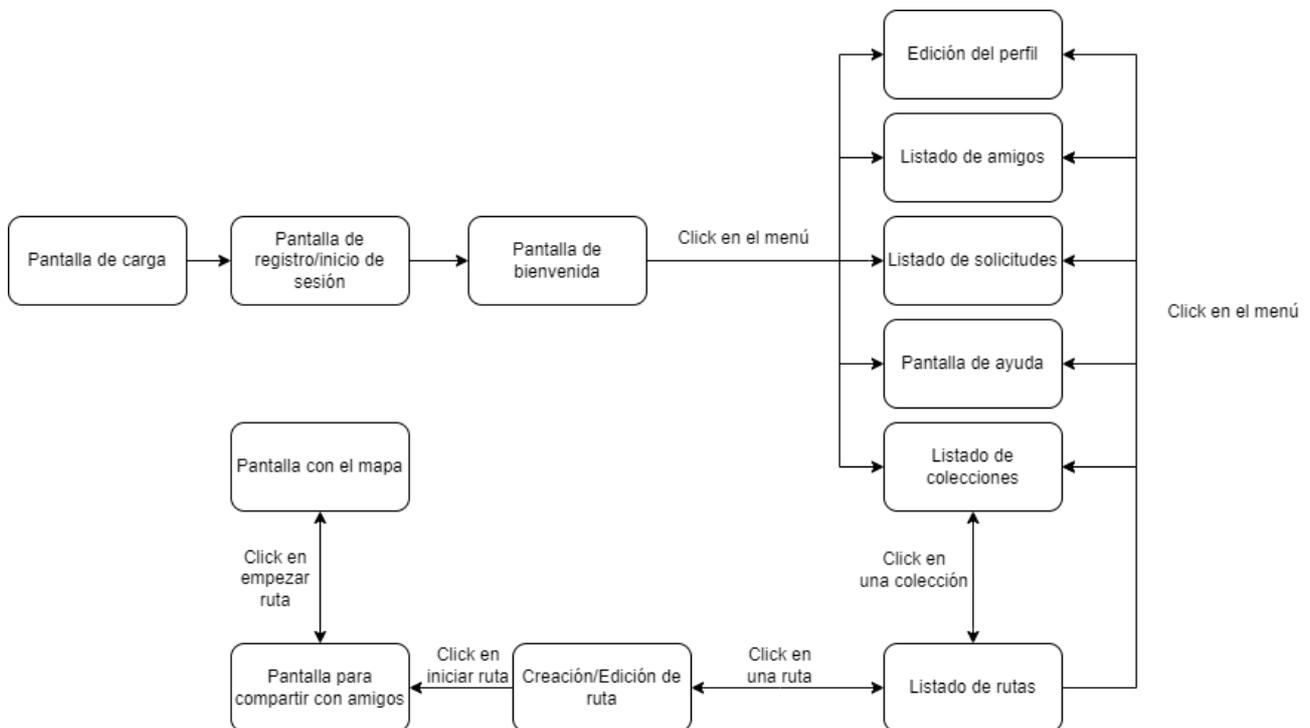


Ilustración 36. Diagrama de navegabilidad móvil



6.5 ASI 10: ESPECIFICACIÓN DEL PLAN DE PRUEBAS

En este apartado se expondrá el plan de pruebas que se llevará a cabo durante el desarrollo del sistema, englobando las dos aplicaciones.

En la fase de desarrollo, se podrán realizar estas pruebas, pero siempre se harán después de desarrollar el código, con pruebas de caja blanca.

Primero, aislaremos el fallo para saber su procedencia, ya que puede provenir de la aplicación móvil o de la aplicación web, después analizaremos el código para detectar el origen del fallo y analizar posibles alternativas para así, por último, implementar ese cambio y comprobar que las pruebas funcionan correctamente.

Primero de todo, hay que comentar que no realizaremos unas **pruebas unitarias** formalizadas, para poner a prueba funcionalidades muy concretas de las aplicaciones que tengan posibilidad de fallo, ya que, debido a que las realizaciones de las pruebas son posteriores a la implementación, y ya que durante el desarrollo se irán probando cada funcionalidad nueva que se vaya realizando, hemos visto conveniente no añadir este apartado, pero sin pasarlo por alto.

Realizaremos unas **pruebas de integración** que se limitarán a realizar los casos de uso y escenarios definidos en el apartado [7.1](#), donde tendremos en cuenta valores incorrectos y casos límite. No pondremos tanto énfasis en los valores correctos ya que se fue probando durante el desarrollo.

Para garantizar que la aplicación es accesible correctamente, se llevarán a cabo **pruebas de accesibilidad**. Estas pruebas se realizarán utilizando herramientas especializadas de accesibilidad y se evaluarán diferentes elementos, como el contraste de colores, el tamaño de fuente y los elementos interactivos, entre otros. El objetivo de estas pruebas es garantizar que la aplicación sea accesible para todas las personas, independientemente de su capacidad visual, motora o cognitiva.

También realizaremos unas **pruebas de usabilidad** donde utilizaremos diferentes usuarios para la realización de ciertas tareas en la aplicación para después hacerles cumplimentar un cuestionario con diferentes preguntas sobre la usabilidad durante el uso de la aplicación que será descrito en el apartado [7.4.3](#).

Capítulo 7 DISEÑO DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO

DSI

7.1 DSI 3: DISEÑO DE CASOS DE USO REALES

Varios casos de uso están desarrollados en el Apéndice [Diagramas UML](#) para reflejar algunas de las situaciones más típicas que podemos encontrar en este sistema, seguido también de [Escenarios](#). En este punto describiremos un caso de uso un poco más complejo mediante diagramas de interacción para la parte web y la parte móvil. Este caso de uso será el más usado en las aplicaciones y sería el de la creación de una ruta siempre y cuando el usuario esté autenticado.

7.1.1 Diagrama de Interacción Web

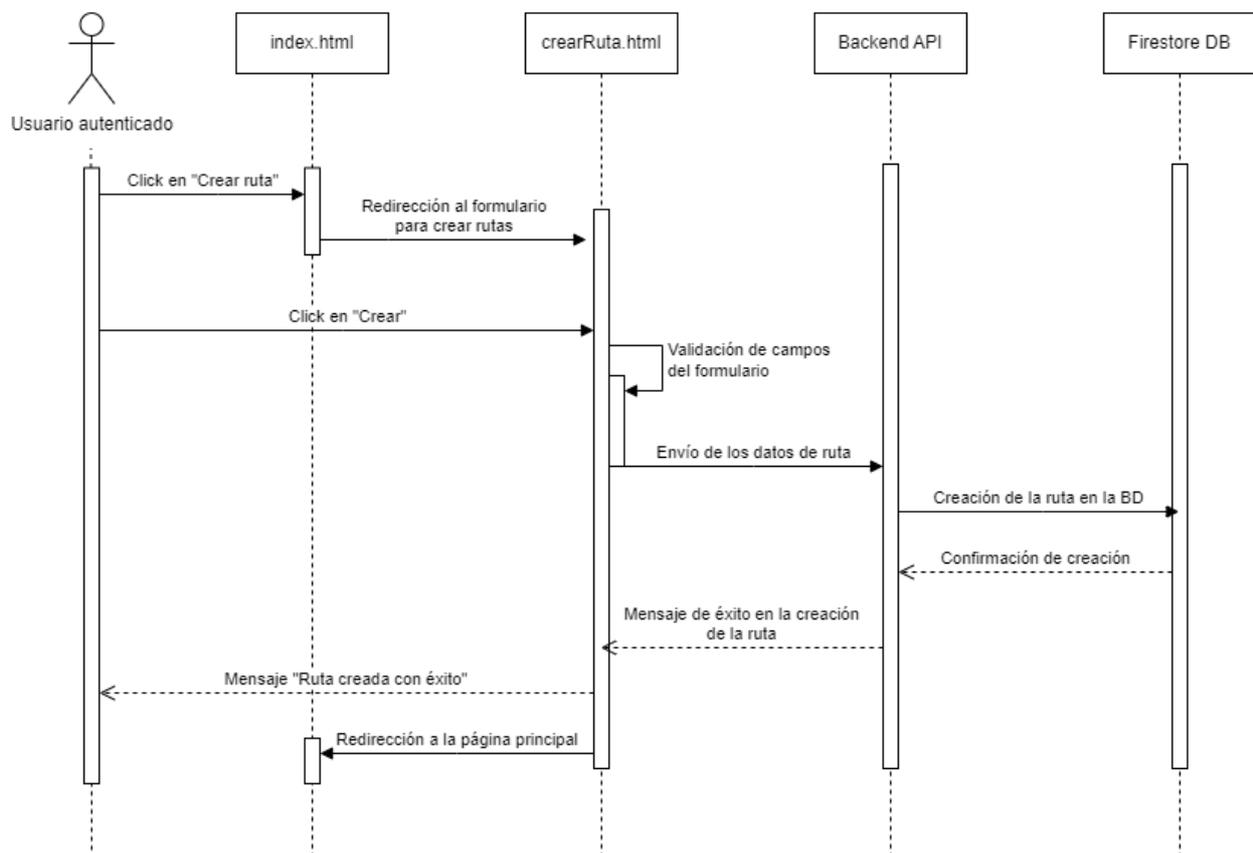


Ilustración 37. Diagrama de secuencia. Crear ruta en web

7.1.2 Diagrama de Interacción App

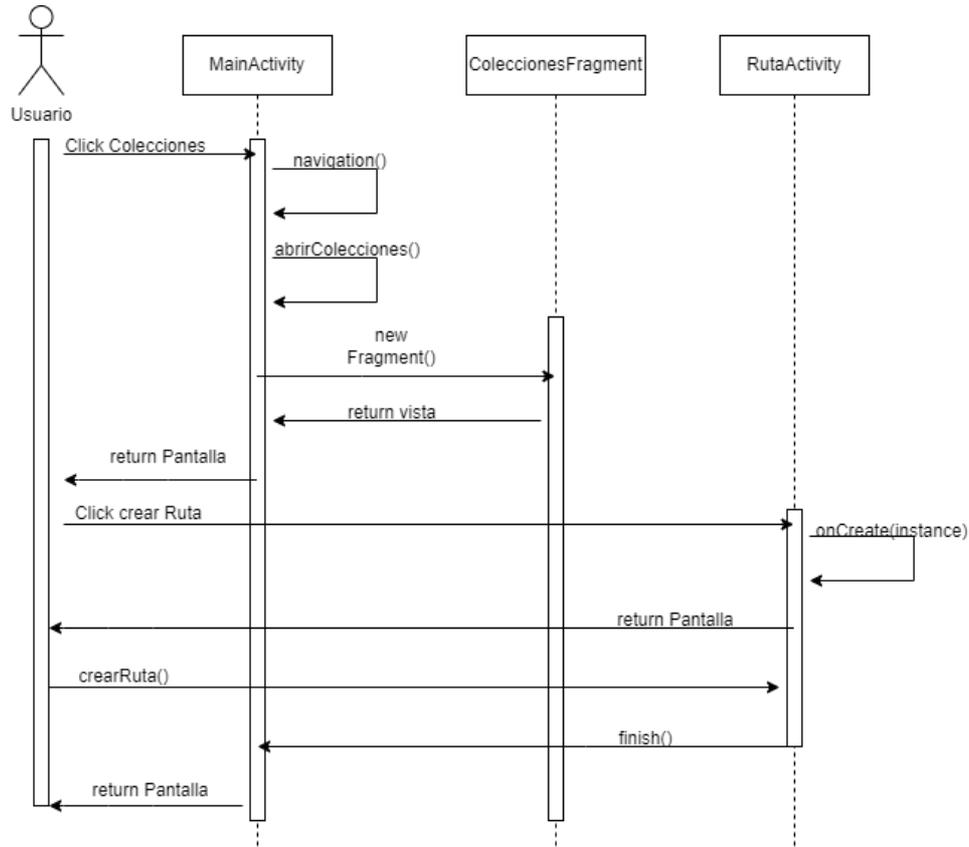


Ilustración 38. Diagrama de interacción de creación de ruta

7.2 DSI 4: DISEÑO DE CLASES

7.2.1 Diagrama de Clases

El diagrama de clases utilizado es el especificado en el punto [6.3.1 Diagrama de Clases](#).

7.3 DSI 5: DISEÑO DE LA ARQUITECTURA DE MÓDULOS DEL SISTEMA

7.3.1 DSI 5.1 Diseño de Módulos del Sistema

En este apartado, mostraremos una vista del diagrama de paquetes del sistema, por una parte, el de la web y por el otro el de la aplicación móvil y se explicará su contenido de una manera breve y concisa.

Este sería el diagrama de la aplicación móvil, donde se pueden observar 4 paquetes diferentes:

- **Adapter:** En este paquete se encuentran las clases usadas para todos los contenedores usados en las diferentes pantallas que muestran listas de objetos.
- **Modelos:** En este paquete se encuentran las clases usadas en la aplicación, siendo aquellos objetos usados para el paso de información y forma de mostrar la información entre pantallas.
- **UI:** En este paquete están las clases encargadas de cargar en pantalla principal las opciones del menú.
- **Activity:** En este paquete se encuentran las clases principales que se usan en la aplicación debido a que son las encargadas de mostrar las diferentes pantallas.

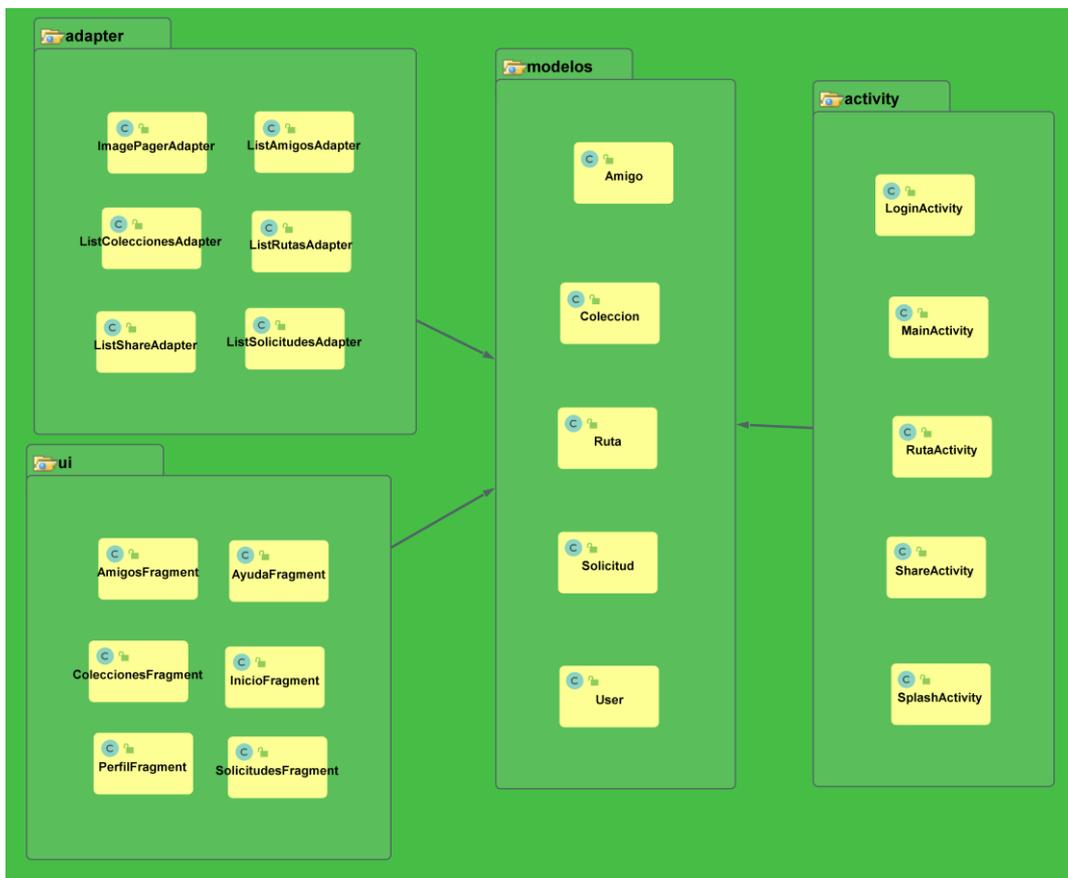


Ilustración 39. Diagrama de paquetes móvil

7.3.2 DSI 5.2 Diseño de Comunicaciones entre Módulos

A continuación, para conocer las comunicaciones entre los módulos y también los usuarios del sistema, mostraremos un diagrama de despliegue:

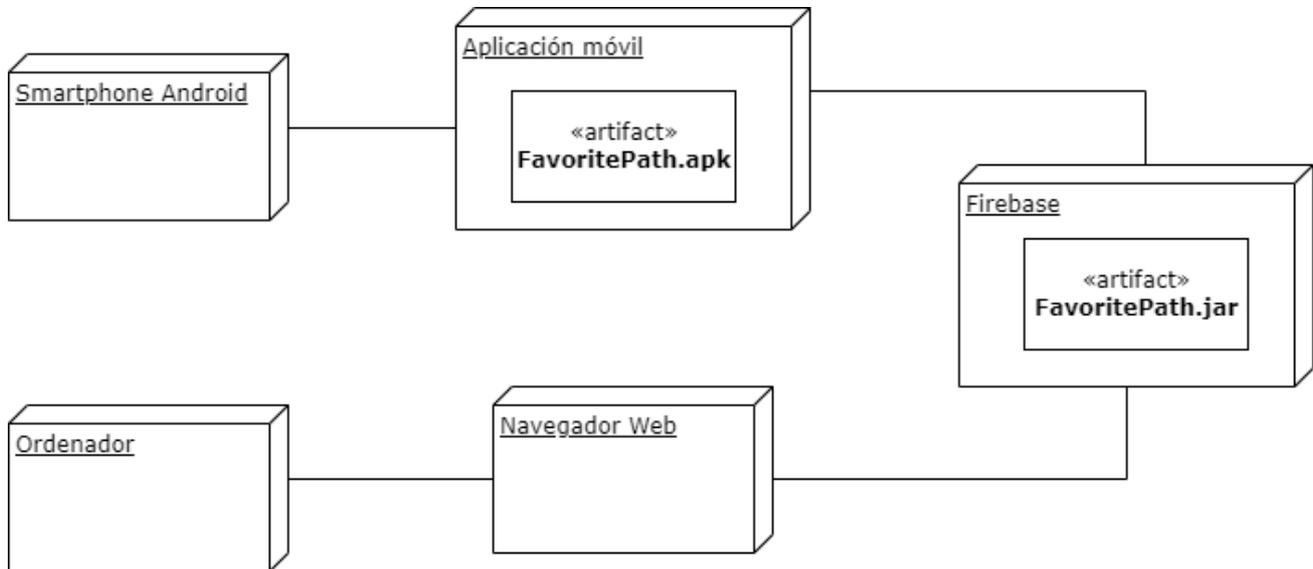


Ilustración 40. Diagrama de despliegue

Se puede observar que los puntos de acceso al sistema se encuentran, por un lado, en el dispositivo Android y por el otro en un ordenador. Mientras que el móvil se conecta con la aplicación móvil, el ordenador debe entrar por un navegador web al dominio que ofrecerá Firebase, donde alojará el sistema. Además, los dos puntos de acceso se conectan con Firebase para poder usar el servicio de autenticación y guardado de datos del sistema.

7.4 DSI 10: ESPECIFICACIÓN TÉCNICA DEL PLAN DE PRUEBAS

7.4.1 Pruebas Unitarias

Como hemos comentado anteriormente, no se van a realizar unas pruebas unitarias formalizadas, por lo que no tendremos en el siguiente capítulo ningún apartado referenciándolas ya que nos centraremos más en las demás pruebas, sobre todo en las de integración ya que son aquellas que probarán toda la aplicación en su conjunto.

7.4.2 Pruebas de Integración y del Sistema

Las pruebas de integración se basarán en comprobar valores límite como erróneos. Además de que el desarrollador las irá realizando a medida que implemente el sistema, pero además se utilizará

Selenium en la parte web. Estas pruebas incluirán pruebas de campos vacíos al igual que repeticiones y datos erróneos, en los siguientes apartados:

- Probar el registro e inicio de sesión
- Inicio de sesión con Google.
- Edición del perfil de usuario
- Mandar solicitudes de amistad
- Creación de colecciones
- Creación y edición de rutas

7.4.3 Pruebas de Usabilidad

Las pruebas de usabilidad se basarán en un cuestionario que se entregará a los usuarios que elegiremos según su destreza informática y su edad, buscando un grupo heterogéneo para alcanzar al mayor número de colectivos potenciales de usar la aplicación. También valoraremos cualquier otro tipo de información que nos aporten vía mensaje o vía oral.

7.4.3.1 Diseño del cuestionario de las Pruebas de Usabilidad

Preguntas de carácter general

Tabla 25. Preguntas de carácter general

¿Usa un ordenador frecuentemente?
<ol style="list-style-type: none">1. Todos los días2. Varias veces a la semana3. Ocasionalmente4. Nunca o casi nunca
¿Qué tipo de actividades realiza con el ordenador?
<ol style="list-style-type: none">1. Es parte de mi trabajo o profesión2. Lo uso básicamente para ocio3. Solo empleo aplicaciones estilo Office4. Únicamente leo el correo y navego ocasionalmente
¿Ha usado alguna vez software como el de esta prueba?
<ol style="list-style-type: none">1. Sí, he empleado software similar2. No, aunque si empleo otros programas que me ayudan a realizar tareas similares3. No, nunca
¿Qué busca Vd. Principalmente en un programa?
<ol style="list-style-type: none">1. Que sea fácil de usar2. Que sea intuitivo3. Que sea rápido

4. Que tenga todas las funciones necesarias

Actividades guiadas

- Autentícate.
- Cierra tu sesión y vuelve a entrar.
- Accede a la ventana de ayuda para informarte.
- Edita la biografía de tu perfil.
- Envía una solicitud de amistad al email yakilandia15@gmail.com.
- Crea una colección.

MÓVIL

- Crea una ruta dentro de esa colección y añádele una imagen.
- Edita la ruta para que no sea lineal, en caso de que lo sea.
- Empieza la ruta.

WEB

- Crea una ruta y añádela a esa colección.
- Edita la ruta para añadirle una imagen.
- Añade una ruta a favoritas.
- Busca una ruta de menos de 10km.
- Entra al perfil de otra persona.

Preguntas Cortas sobre la Aplicación y Observaciones

Tabla 26. Preguntas cortas de Usabilidad

Preguntas Cortas de Usabilidad				
Funciones de la Aplicación				
Facilidad de Uso	Siempre	Frecuentemente	Ocasionalmente	Nunca
¿Sabe dónde está dentro de la aplicación?				
¿Existe ayuda para las funciones en caso de que tenga dudas?				
¿Le resulta sencillo el uso de la aplicación?				
Funcionalidad	Siempre	Frecuentemente	Ocasionalmente	Nunca
¿Funciona cada tarea como Vd. Espera?				
¿El tiempo de respuesta de la aplicación es adecuado?				
Calidad del Interfaz				
Aspectos gráficos	Muy Adecuado	Adecuado	Poco Adecuado	Nada Adecuado
El tipo y tamaño de letra es				



Los iconos e imágenes usados son				
Los colores empleados son				
Diseño de la Interfaz		Si	No	A veces
¿Le resulta fácil de usar?				
¿El diseño de las pantallas es claro y atractivo?				
¿Cree que el programa está bien estructurado?				
Observaciones				
Cualquier comentario del usuario				

7.4.4 Pruebas de Accesibilidad

Algunas de las áreas clave que se evaluarán durante estas pruebas incluyen:

1. Navegación y operación: Se verifica que todas las funciones y acciones del sistema sean accesibles a través de diferentes métodos de interacción.
2. Percepción de la información: Se evalúa la presentación de la información en términos de contraste, tamaño de fuente, legibilidad, uso de colores, etc.
3. Etiquetado y descripción: Se verifica que los elementos interactivos, imágenes, enlaces, formularios y otros elementos estén debidamente etiquetados y descritos para permitir una comprensión clara y precisa.
4. Cumplimiento de estándares: Se evalúa si el sistema cumple con los estándares y pautas de accesibilidad reconocidos internacionalmente, como las Directrices de Accesibilidad para el Contenido Web (WCAG) del W3C.

El objetivo de las pruebas de accesibilidad será asegurar que el software sea accesible para todos los usuarios, sin importar sus capacidades o limitaciones. Estas pruebas ayudan a identificar posibles barreras de acceso y permiten tomar medidas correctivas para mejorar la experiencia del usuario y cumplir con los requisitos de accesibilidad.

7.4.5 Pruebas de Rendimiento

No teníamos pensado realizar unas pruebas de rendimiento debido a que nuestro sistema está centralizado con Firebase y siendo un proveedor tan conocido y fiable, no debería dar problemas de rendimiento, pero ya que nos ofrece un producto llamado TestLab, donde podemos hacer varias pruebas dentro de nuestra aplicación, las realizaremos para ver si la conexión con Firebase es rápida, además de analizar el rendimiento del mapa sobre todo en esas acciones que involucren a los mismos.

Capítulo 8 CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO

CSI



8.1 CSI 1: PREPARACIÓN DEL ENTORNO DE GENERACIÓN Y CONSTRUCCIÓN

8.1.1 Herramientas y programas usados para el desarrollo

A continuación, se enumeran las herramientas utilizadas tanto para la parte web como móvil. En los siguientes subapartados se especifican aquellas que solo se han utilizado en uno u otro de los subsistemas.

Microsoft Word: Es un procesador de textos que nos ofrece Microsoft, permitiendo realizar cualquier tipo de documentos de texto de una manera sencilla y accesible para cualquier usuario. Hemos utilizado Word para la redacción de este documento y utilizando la versión que ofrece el paquete de Office 365. La URL de acceso es <https://microsoft-word.softonic.com>.

Microsoft Project: Es una herramienta para planificar y administrar proyectos, permitiendo planificar tareas, gestionar recursos y sus costes, hacer seguimientos del avance del proyecto, generar gráficas y muchas cosas más. Hemos utilizado Project para la planificación y seguimiento del proyecto, utilizando la versión que ofrece el paquete de Office 365. La URL de acceso es <https://microsoft-project-2010.softonic.com>.

Google Firebase: esta es una plataforma para el desarrollo de aplicaciones web y apps móviles. Se ha utilizado para alojar el portal web, para almacenar imágenes y base de datos y para usar su sistema de autenticación. La URL de acceso es <https://firebase.google.com/>.

Looka: esta es una página web que contiene un generador de logos para todo tipo de empresas. Respondiendo a varias preguntas, ofrece distintos diseños que luego se pueden modificar a gusto del usuario. La URL de acceso es <https://looka.com/>.

Mapbox: este es un proveedor de mapas en línea. Se hace uso de su API para mostrar los mapas del sistema. La URL de acceso es <https://www.mapbox.com/>.

Github: este es el portal de alojamiento de código donde están subidos los repositorios del portal web y la aplicación móvil. La URL de acceso es <https://github.com/>. El enlace al repositorio de la aplicación móvil es <https://github.com/UO269509/FavoritePath>.

8.1.1.1 Portal web

JavaScript: este es un lenguaje de programación interpretado, muy utilizado en el desarrollo web. Se trata de un dialecto del estándar ECMAScript. Se usa mucho en el lado del cliente. Su documentación se puede encontrar en <https://developer.mozilla.org/en-US/docs/Web/JavaScript>.



Node.js: se trata de un entorno de ejecución para servidor en JavaScript. Es multiplataforma y asíncrono, con una arquitectura orientada a eventos. Se puede descargar en <https://nodejs.org/es/download>.

Visual Studio Code: un editor de código muy completo y sencillo de utilizar. Contiene un gran repertorio de herramientas útiles, como el resaltado de sintaxis, soporte para la depuración del código, refactorización o el control integrado de Git, entre otros. Se puede descargar en <https://code.visualstudio.com/download>.

CSSMatic Box Shadow generator: esta página ofrece una forma intuitiva de crear sombras en CSS para los elementos del frontend. La URL de acceso es <https://www.cssmatic.com/box-shadow>.

Google Fonts Material Symbols: esta es la librería de símbolos de Google, utilizada en Material Design (un sistema de guías, componentes y herramientas cuyo uso es buena práctica en el diseño de interfaces de usuario). La URL de acceso es <https://fonts.google.com/icons>.

XAMPP: este es un paquete de software libre, que contiene un sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes PHP y Perl. Solo se usa el servidor Apache para alojar la web en local durante su desarrollo. La versión utilizada es la 3.3.0. El lugar de descarga es <https://www.apachefriends.org/es/download.html>.

Postman: esta es una herramienta para el diseño, construcción y pruebas de APIs. Se ha utilizado para facilitar las pruebas durante el desarrollo de la API que conecta la web con la base de datos de Firestore. La versión utilizada es la 10.14.8. Se puede descargar en <https://www.postman.com/downloads/>.

8.1.1.2 Aplicación móvil

Java: Es un lenguaje de programación de propósito general y orientado a objetos, que se caracteriza por su portabilidad, seguridad y amplia biblioteca estándar. Su portabilidad, el rasgo más importante, permite que pueda ejecutarse en diferentes plataformas sin necesidad de realizar modificaciones significativas, siendo posible gracias a la Máquina Virtual de Java (JVM), interpretando el código Java y ejecutándolo en diferentes sistemas operativos. Hemos utilizado Java para programar la aplicación móvil y se ha usado Java 1.8.0_371. Se puede descargar en https://www.java.com/es/download/ie_manual.jsp.

Gradle: Es una herramienta de construcción de proyectos, proporcionando un enfoque declarativo para definir la estructura del proyecto, las dependencias y las tareas de construcción. Permite automatizar y gestionar eficientemente la compilación, prueba, empaquetado y despliegue del proyecto. Utiliza un lenguaje de dominio específico (DSL) basado en Groovy o Kotlin para describir la configuración del proyecto y las tareas de construcción. se caracteriza por su flexibilidad, escalabilidad y capacidad para gestionar proyectos grandes con múltiples módulos y dependencias.



Hemos utilizado Gradle para la construcción del proyecto móvil siendo la versión 7.0.2. Viene ya descargado dentro de Android Studio.

Android Studio: Es un entorno de desarrollo integrado (IDE) utilizado para la creación de aplicaciones móviles en el sistema operativo Android. Proporciona una plataforma completa y robusta para el desarrollo de aplicaciones, incluyendo herramientas de edición de código, depuración, emulación de dispositivos, compilación y gestión de dependencias. Con una interfaz intuitiva y orientada al desarrollo de Android, facilita la creación de aplicaciones modernas y eficientes. Además, ofrece una amplia gama de características, como el diseño de interfaces de usuario con el editor de diseño y la compatibilidad con múltiples lenguajes de programación como Java y Kotlin. Hemos utilizado Android Studio para implementar la aplicación móvil siendo la versión Artic Fox | 2020.3.1 para Windows. Se puede descargar en <https://developer.android.com/studio/>.

8.2 CSI 2: GENERACIÓN DEL CÓDIGO DE LOS COMPONENTES Y PROCEDIMIENTOS

Debido al alto contenido de clases, la generación de código se encuentra en el siguiente enlace:
<https://uo269509.github.io>

8.3 CSI 4: EJECUCIÓN DE LAS PRUEBAS DE INTEGRACIÓN

Como habíamos comentado en el punto [7.4.2](#), aquí incluiremos las pruebas de campos vacíos al igual que repeticiones y datos erróneos dentro de los campos para ver su correcta interacción. Al final, hablaremos de los fallos detectados y su corrección:

Tabla 27. Caso de Uso 1

Caso de Uso 1: Registro fallido sin campos	
Prueba	Resultado Esperado
Registramos un usuario con datos vacíos	El sistema informará al usuario que el registro no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Tabla 28. Caso de Uso 2

Caso de Uso 2: Registro fallido con un email mal escrito	
Prueba	Resultado Esperado
Registramos un usuario con datos, pero el campo email no sigue el patrón correcto	El sistema informará al usuario que el registro no se ha podido realizar



Resultado Obtenido

El sistema ha informado con un mensaje al usuario de su acción
--

Tabla 29. Caso de Uso 3

Caso de Uso 3: Registro fallido con contraseña menor a 6 caracteres	
Prueba	Resultado Esperado
Registramos un usuario con datos, pero el campo contraseña tiene 5 caracteres	El sistema informará al usuario que el registro no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Tabla 30. Caso de Uso 4

Caso de Uso 4: Inicio de sesión fallido sin campos	
Prueba	Resultado Esperado
Iniciamos sesión con datos vacíos	El sistema informará al usuario que el inicio de sesión no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Tabla 31. Caso de Uso 5

Caso de Uso 5: Inicio de sesión con usuario inexistente	
Prueba	Resultado Esperado
Iniciamos sesión con datos, pero el usuario no existe en la base de datos	El sistema informará al usuario que el inicio de sesión no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Tabla 32. Caso de Uso 6

Caso de Uso 6: Inicio de sesión con Google	
Prueba	Resultado Esperado
Iniciamos sesión usando el apartado de Google	El sistema cambiará a la pantalla principal con el usuario seleccionado
Resultado Obtenido	
El sistema muestra la pantalla principal con el perfil del usuario seleccionado	



Tabla 33. Caso de Uso 7

Caso de Uso 7: Edición de perfil fallido sin nombre	
Prueba	Resultado Esperado
Editar un perfil con el nombre vacío	El sistema informará al usuario que la edición no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Tabla 34. Caso de Uso 8

Caso de Uso 8: Mandar solicitud de amistad vacía	
Prueba	Resultado Esperado
Mandar una solicitud de amistad a ningún email	El sistema informará al usuario que la acción no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Tabla 35. Caso de Uso 9

Caso de Uso 9: Mandar solicitud de amistad repetida	
Prueba	Resultado Esperado
Mandar una solicitud de amistad a un usuario que ya se le ha mandado una	El sistema informará al usuario que la acción no se ha podido realizar
Resultado Obtenido (ERROR)	
El sistema ha dejado que el usuario mandase otra solicitud al mismo amigo	

Tabla 36. Caso de Uso 10

Caso de Uso 10: Mandar solicitud de amistad a un amigo	
Prueba	Resultado Esperado
Mandar una solicitud de amistad a un usuario que ya es amigo	El sistema informará al usuario que la acción no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Tabla 37. Caso de Uso 11

Caso de Uso 11: Creación de colección fallido sin nombre	
Prueba	Resultado Esperado
Crear una colección con el nombre vacío	El sistema informará al usuario que la edición no se ha podido realizar
Resultado Obtenido	

El sistema ha informado con un mensaje al usuario de su acción

Tabla 38. Caso de Uso 12

Caso de Uso 12: Creación de colección con nombre existente	
Prueba	Resultado Esperado
Crear una colección con el nombre igual a otra colección	El sistema informará al usuario que la creación no se ha podido realizar
Resultado Obtenido (ERROR)	
El sistema ha dejado que el usuario cree la colección, eliminando la anterior.	

Tabla 39. Caso de Uso 13

Caso de Uso 13: Crear ruta sin campos	
Prueba	Resultado Esperado
Crear una ruta con datos vacíos	El sistema informará al usuario que la acción no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Tabla 40. Caso de Uso 14

Caso de Uso 14: Actualizar ruta sin campos	
Prueba	Resultado Esperado
Actualizar una ruta con datos vacíos	El sistema informará al usuario que la acción no se ha podido realizar
Resultado Obtenido	
El sistema ha informado con un mensaje al usuario de su acción	

Como podemos observar, hemos tenido dos pruebas que han dado un error no esperado, que han sido el caso de uso 9 y el 12, relativos al uso de campos duplicados. Finalmente, se ha arreglado validando que dicho campo no exista en la base de datos, ya que anteriormente, no se estaba validando.

8.4 CSI 5: EJECUCIÓN DE LAS PRUEBAS DEL SISTEMA

8.4.1 Prueba de Usabilidad

En primer lugar, hablaremos un resumen de los encuestados en la aplicación móvil, siendo en total 10 personas que cumplimentaron el cuestionario y de los cuáles, estamos agradecidos por su colaboración.



Hemos obtenido los siguientes resultados:

Tabla 41. Encuesta de Usabilidad

¿Usa un ordenador frecuentemente?
6 - Todos los días
3 - Varias veces a la semana
1 - Ocasionalmente
0 - Nunca o casi nunca
¿Qué tipo de actividades realiza con el ordenador?
5 - Es parte de mi trabajo o profesión
4 - Lo uso básicamente para ocio
0 - Solo empleo aplicaciones estilo Office
1 - Únicamente leo el correo y navego ocasionalmente
¿Ha usado alguna vez software como el de esta prueba?
4 - Sí, he empleado software similar
2 - No, aunque si empleo otros programas que me ayudan a realizar tareas similares
4 - No, nunca
¿Qué busca Vd. Principalmente en un programa?
3 - Que sea fácil de usar
3 - Que sea intuitivo
1 - Que sea rápido
3 - Que tenga todas las funciones necesarias

Tabla 42. Preguntas Cortas de Usabilidad

Preguntas Cortas de Usabilidad				
Funciones de la Aplicación				
Facilidad de Uso	Siempre	Frecuentemente	Ocasionalmente	Nunca
<i>¿Sabe dónde está dentro de la aplicación?</i>	6	3	-	-
<i>¿Existe ayuda para las funciones en caso de que tenga dudas?</i>	10	-	-	-
<i>¿Le resulta sencillo el uso de la aplicación?</i>	6	4	-	-
Funcionalidad	Siempre	Frecuentemente	Ocasionalmente	Nunca

<i>¿Funciona cada tarea como Vd. Espera?</i>	7	3	-	-
<i>¿El tiempo de respuesta de la aplicación es adecuado?</i>	8	2	-	-
Calidad del Interfaz				
Aspectos gráficos	Muy Adecuado	Adecuado	Poco Adecuado	Nada Adecuado
<i>El tipo y tamaño de letra es</i>	6	4	-	-
<i>Los iconos e imágenes usados son</i>	5	5	-	-
<i>Los colores empleados son</i>	4	6	-	-
Diseño de la Interfaz		Si	No	A veces
<i>¿Le resulta fácil de usar?</i>		10	-	-
<i>¿El diseño de las pantallas es claro y atractivo?</i>		7	-	3
<i>¿Cree que el programa está bien estructurado?</i>		9	-	1

En conclusión, podemos decir que las pruebas han tenido un resultado satisfactorio, además de que, gracias a las observaciones de los usuarios, podremos mejorar la aplicación para tener una segunda versión sin defectos vistos por los usuarios.

A continuación, expondremos algunas observaciones que nos han puesto, las cuáles, serían una obligación para una segunda iteración del proyecto, entrando al apartado de mejoras:

“El texto de la sección de ayuda se ve muy junto y parece mucho texto.”

“En las rutas, si el título tiene más de 18 caracteres, hay que desplazarlo para verlo entero, igual quedaría mejor si apareciese en dos líneas.”

“Resulta un poco equívoca la denominación del apartado solicitudes, quizá le pondría solicitudes recibidas, y el apartado amigos, envío de solicitudes.”

8.4.2 Pruebas de Accesibilidad

En la parte móvil nos hemos valido de TestLab para pasar las pruebas de accesibilidad en Android, las cuales, son las siguientes:

Descripción general	Tipos de problemas	Advertencias ⓘ	Problemas menores	ⓘ Sugerencias ⓘ	^
Se encontraron 31 problemas ⓘ		0	15	16	
	Tamaño de objetivos táctiles ⓘ	0	1	0	
	Contraste bajo ⓘ	0	13	0	
	Etiquetas de contenido ⓘ	0	0	0	
	Implementación ⓘ	0	1	16	

Ilustración 41. Pruebas de Accesibilidad con TestLab

En primera medida podemos observar que no tenemos ninguna advertencia, dando un síntoma positivo de las pruebas y por otro lado tenemos 16 sugerencias, todas ellas de implementación, donde se centran en decir que utilizo elementos fijos en las pantallas, los cuales deben aparecer así para el correcto funcionamiento de la aplicación y que ninguno influye en la accesibilidad, por lo que no haremos mucho caso a las mismas.

Lo que comentaremos ahora son esos 15 problemas menores, que a simple vista suena a que pueden influir en la accesibilidad del sistema, pero desglosaremos los 3 tipos de problemas que existen y daremos una conclusión razonable:

Tamaño de los objetivos táctiles (1)

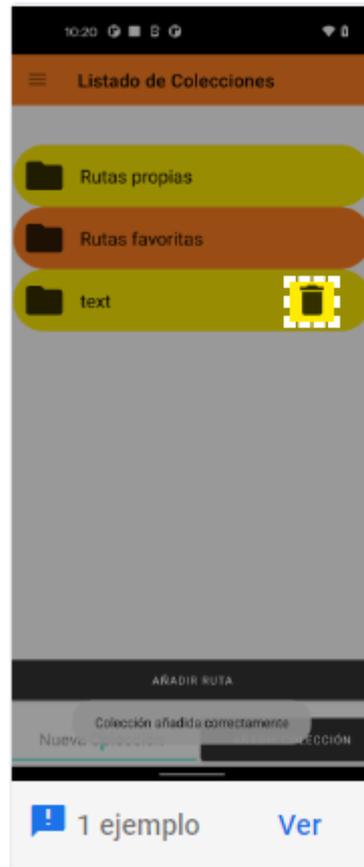


Ilustración 42. Problema menor 1

Como podemos observar, el primer problema viene asociado a que supuestamente el botón de borrado es demasiado pequeño y podría dar problemas. Se ha pensado en hacer más grande el botón, pero al final rompía mucho los diferentes elementos debido al gran espacio que ocasionaban, por lo que finalmente no se ha cambiado.

Ahora pasamos al mayor número de fallos, que serían esos 13 problemas de contraste bajo, los cuales, podemos resumir con la siguiente foto de dos ejemplos:



Ilustración 43. Problema menor 2

Estos problemas vienen derivados del uso de imágenes con fondo blanco sobre blanco, lo cual nos viene genial para que no resalte el borde de la foto con el fondo, quedando así, y los otros problemas vienen del uso de una pista en los formularios grisáceo, pero tampoco se solucionaron por la familiaridad de muchos usuarios con otras aplicaciones.

El último problema de implementación es parecido a lo que comentamos antes, por lo que solo mostraré donde da ese problema, pero volviendo a recordar que no hace falta ninguna modificación.

Implementación (17)



Ilustración 44. Problema menor 3

8.4.3 Pruebas de Rendimiento

Dentro de Firebase, hemos podido utilizar su parte de pruebas de aplicaciones y gracias a ello, para la realización de estas pruebas, hemos dejado que este proveedor sea el encargado de realizarlas, cuyos resultados pasaremos a comentar ahora:

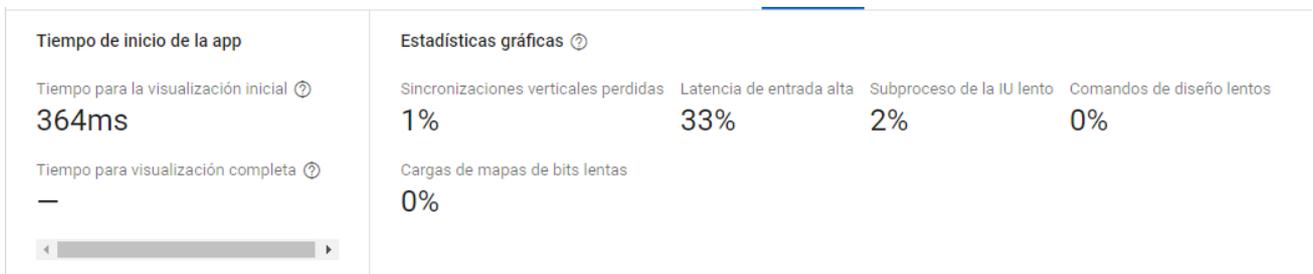


Ilustración 45. Pruebas de Rendimiento de Firebase

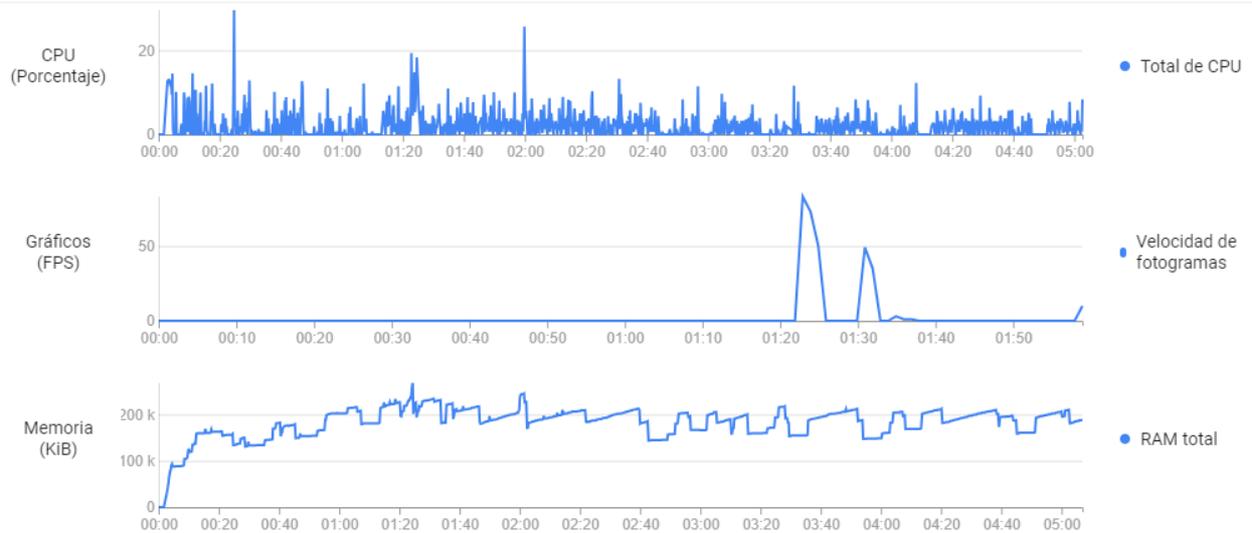


Ilustración 46. Resultados pruebas Rendimiento TestLab

Estos fueron los resultados arrojados por TestLab, donde nos dibuja en diferentes gráficas los usos diversos de la aplicación seccionado segundo a segundo junto a un videoclip de las diferentes acciones que ha realizado, donde los únicos picos de las gráficas se deben a temas de carga de datos o mismamente el renderizado del mapa, pero sin exceder mucho el rendimiento de la aplicación, por ende, dando por buenas estas pruebas de rendimiento.

8.5 CSI 6: ELABORACIÓN DE LOS MANUALES DE USUARIO

8.5.1 Manual de Usuario

FavoritePath busca que cualquier usuario pueda realizar una ruta y compartirla con amigos sin importar el nivel técnico del usuario ya que usamos una interfaz intuitiva y sencilla. En estos puntos enumeramos todas las acciones que alguien puede realizar con una explicación detallada si hay dudas.

Registro e inicio de sesión: para poder ingresar a la app, tenemos 2 formas, una sería el previo registro en el formulario, con un correo electrónico que no esté en uso y una contraseña de mínimo 6 cifras y clicando en el botón de registro, con estas credenciales podrás acceder con el botón de inicio de sesión.

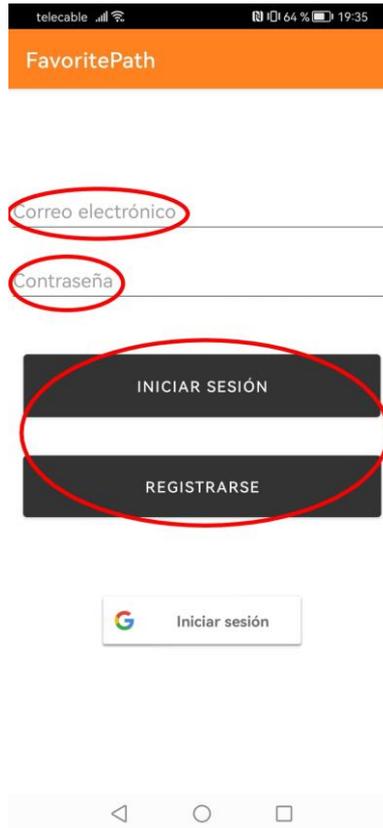


Ilustración 47. Login 1

El otro método sería utilizar el inicio de sesión de Google y elegir una cuenta propia y con eso accederemos directamente.

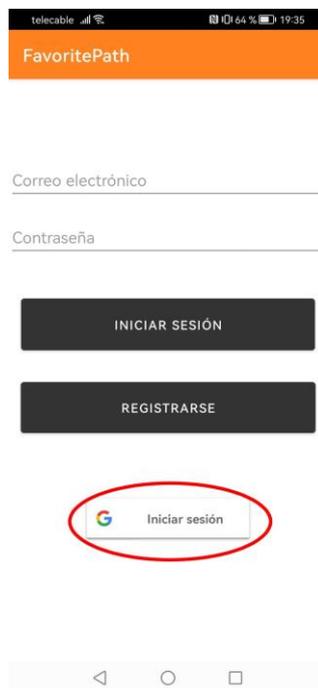


Ilustración 48. Login 2

Navegación y cierre de sesión: ya dentro de la app, para poder navegar por ella, deberemos abrir el menú, situado en la esquina superior izquierda de la pantalla o deslizando la pantalla de izquierda a derecha, siendo solamente accesible desde estas opciones, en el caso de que no se pueda acceder al menú, solo bastará con volver atrás desde su dispositivo.

En él se pueden observar las opciones de la app, que podemos visitar clicando cada una. También se observa que encima de las opciones está el botón para cerrar sesión y volver al formulario de inicio de sesión.



Ilustración 49. Menú

Edición del perfil: Ya dentro de esta opción, podremos editar nuestro nombre de usuario al igual que una descripción propia que los demás usuarios podrán visualizar, solo haría falta rellenar el formulario y darle al botón de guardar.



Ilustración 50. Perfil

Gestión de colecciones y rutas: Ya dentro de esta opción, podremos visualizar todas las colecciones que tenemos, acompañadas de un botón de borrado aquellas colecciones creadas por el usuario, ya que no se podrán borrar las colecciones creadas por defecto. Hay que tener en cuenta que el borrado de una colección implica el borrado de todas las rutas de su interior. Además, en la parte inferior encontrarás un formulario donde podrás crear diferentes colecciones y un botón para crear rutas que se explicará en el siguiente punto.



Ilustración 51. Colecciones

También podremos visualizar nuestras rutas creadas de una colección concreta clicando en el nombre de esta. Estas estarán acompañadas de un botón de borrado para eliminarla de dicha colección. Además, tendremos las mismas opciones comentadas anteriormente en la parte inferior.



Ilustración 52. Rutas

Creación y edición de rutas: Este apartado ya no cuenta con el menú lateral a su disposición, para volver a él, deberá clicar el botón de retroceder de su dispositivo. Para poder crear una ruta, deberá clicar el botón comentado en el anterior punto, el cual, si se presiona desde el listado general de colecciones creará la ruta en la colección de rutas propias por defecto, pero si se clica desde una colección concreta también aparecerá la ruta creada dentro, mientras que, para editar una ruta, tendrá que seleccionar una ruta ya creada dentro de una colección.



Ilustración 53. Crear ruta 1

Estas funciones comparten el mismo formulario donde podrá escribir un título y descripción, elegir si quiere que la ruta sea pública a todos los usuarios o en cambio, privada, al igual que elegir si quiere que la ruta sea circular y te lleve al mismo punto de partida al acabar o que sea lineal, la posibilidad de añadir imágenes para recordar momentos del viaje y un mapa donde poder elegir que sitios visitar, siempre situado en el centro de su ubicación. En el caso de querer editar los puntos en el mapa, estos solo se podrán borrar desde el último seleccionado hacía el más antiguo. Como final, en la parte inferior se puede observar el botón de guardado y el botón de iniciar ruta que se explica en el siguiente punto.



Ilustración 54. Crear ruta 2

Compartir rutas e iniciarlas: Este apartado ya no cuenta con el menú lateral a su disposición, para volver a él, deberá clicar el botón de retroceder de su dispositivo. Para acceder a este apartado, deberá clicar el botón comentado en el anterior punto, donde podremos visualizar todos los amigos que tenemos, acompañados de un botón para poder compartirles la ruta y en la parte inferior hay un botón que da por comenzada la ruta.



Ilustración 55. Compartir

Después de clicar, abrirá un mapa interactivo donde se visualizará el trayecto más eficiente para realizar el viaje con todas las indicaciones necesarias.



Ilustración 56. Mapa

Gestión de amigos: Ya dentro de esta opción, podremos visualizar todos los amigos que tenemos, acompañados de un botón de borrado en el caso de que quieras eliminar a algún usuario de tu lista. Además, en la parte inferior encontrarás un formulario donde podrás enviar peticiones de amistad a otros usuarios existentes de la app, solo tendrás que escribir su correo electrónico y esperar a que el otro usuario acepte la invitación.

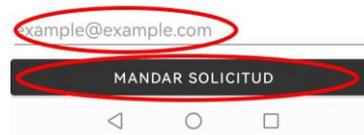


Ilustración 57. Amigos

Gestión de solicitudes: Ya dentro de esta opción, podremos visualizar todas las solicitudes que tenemos, contado con dos tipos de solicitudes, las primeras serían las de amistad, que son aquellas que otros usuarios te han enviado para pasar a ser amigos en la app, las otras serían solicitudes donde un amigo te ha compartido una ruta suya. Todas ellas vienen acompañadas de dos botones, siendo uno para aceptar y otro para denegar respectivamente.



Ilustración 58. Solicitudes

Ayuda: Ya dentro de esta opción, podremos visualizar un pequeño resumen de todas las funcionalidades que ofrece la app, lleno de explicaciones para que, en cualquier momento, el usuario pueda entrar y buscar cualquier inquietud que tenga sobre el funcionamiento de una parte del sistema.

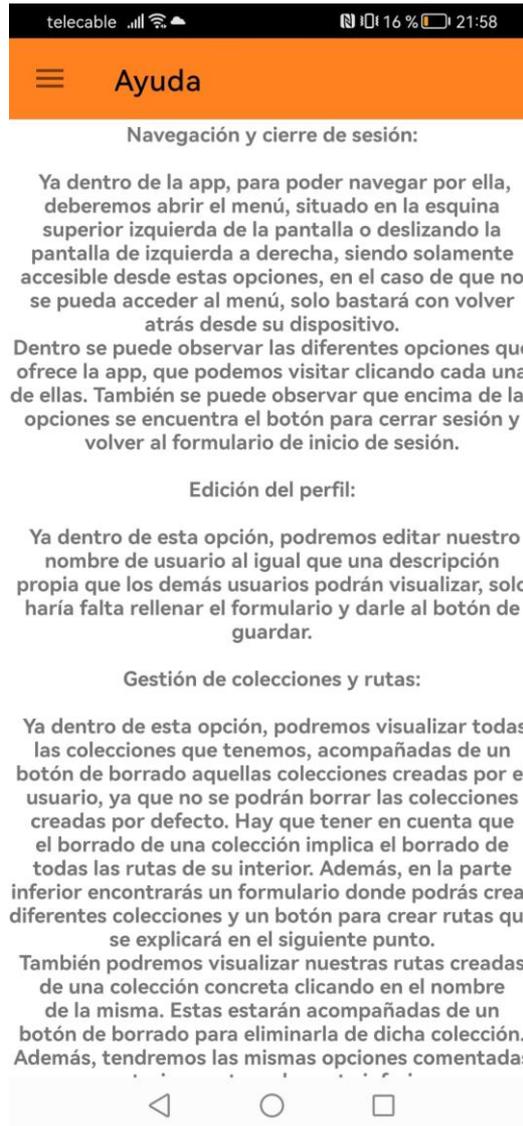
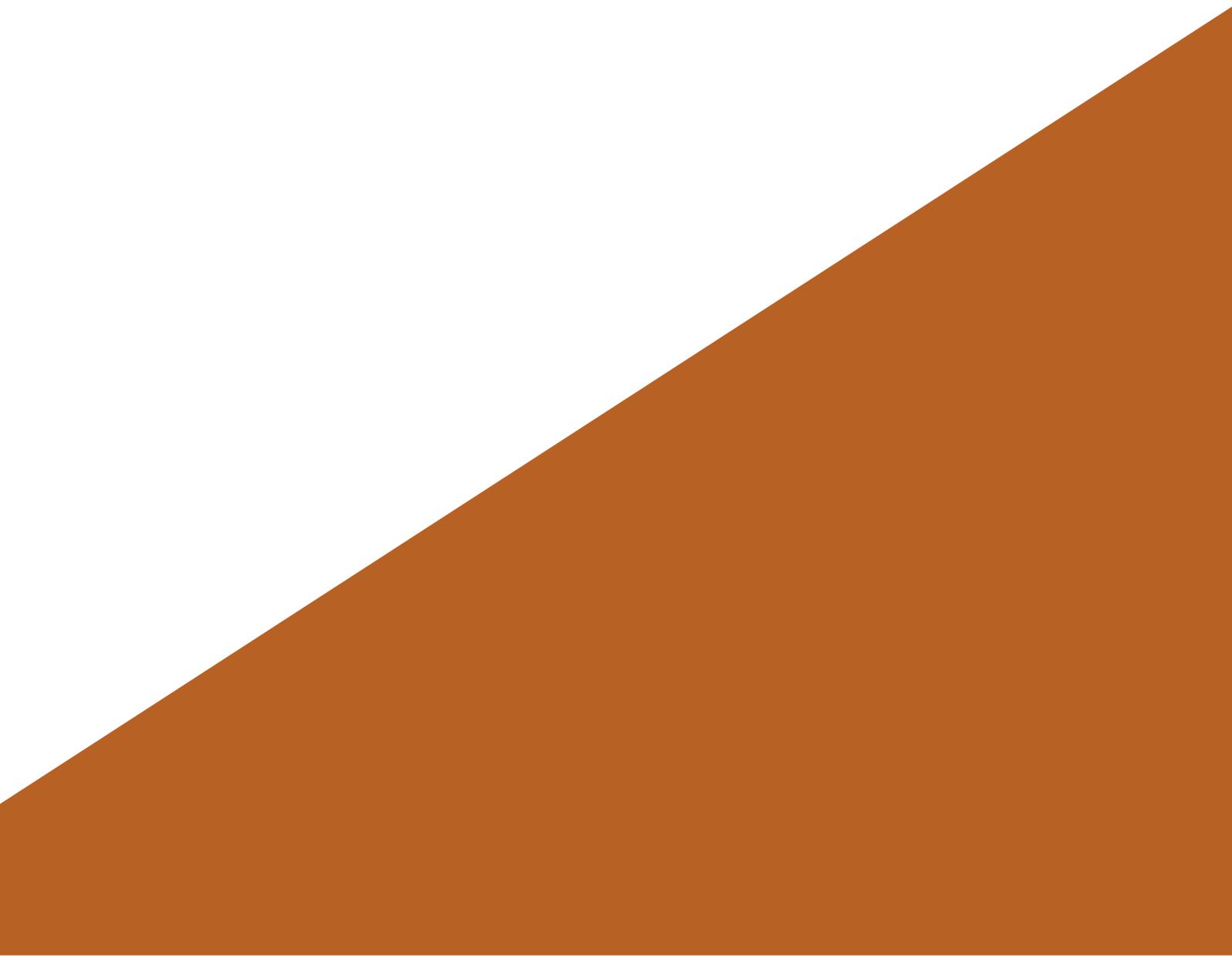


Ilustración 59. Ayuda

Capítulo 9 CONCLUSIONES Y AMPLIACIONES





9.1 CONCLUSIONES

La realización de un trabajo de este calibre ha puesto a prueba los conocimientos adquiridos en prácticamente todas las asignaturas de la carrera. A su vez, ha sido una excelente manera de comprobar hasta qué punto el esfuerzo dedicado durante estos cuatro años ha merecido la pena, al ser conscientes de cuánto hemos aprendido y de la utilidad de este aprendizaje en un entorno práctico.

Tenemos en nuestras manos un sistema complejo, realizado desde cero entre dos personas. Podría parecer que el hecho de ser dos facilitaría el trabajo. Aunque tener a un compañero de apoyo es algo que se agradece en cualquier proyecto, también conlleva una dificultad añadida. Aunque es cierto que “dos cabezas piensan más que una” y que se puede dividir el trabajo para desarrollarlo en paralelo, existe un importante factor de toma de decisiones conjuntas y comunicación que no existiría en un proyecto individual. Al partir ambos del mismo punto, ha sido fundamental comunicar, compartir y debatir ideas desde el primer día hasta el último, para garantizar la coherencia del proyecto. El tiempo dedicado a esto son horas en las que no se observan avances en el proyecto, y, sin embargo, no se puede recortar por este lado si queremos asegurar la llegada a buen puerto de este.

Por tanto, la realización de este Trabajo de Fin de Grado nos ha enseñado que, en primer lugar, todo el esfuerzo invertido en la carrera, por obstáculos que hayan surgido en el camino, no ha sido en vano, ya que ahora somos capaces de desenvolvernos en el entorno informático y llevar a cabo nuestros proyectos valiéndonos de las herramientas adecuadas; y, en segundo lugar, la importancia de reservar tiempo y energía de sobra a la comunicación entre todas y cada una de las partes que tengan algo que aportar en los proyectos desde el primer momento, dado que es una de las inversiones más fructíferas que se pueden realizar.

9.2 AMPLIACIONES

Como ampliaciones para el portal web se han valorado:

- **Integración con servidores de correo electrónico:** cuando un usuario se registre, se le enviará un correo electrónico al email especificado. También se enviará un email cuando otro usuario le envíe una solicitud de amistad. Por otro lado, podría existir algún tipo de newsletter periódica.
- **Integración con API de restaurantes:** se añadirá una funcionalidad que permita que la aplicación sugiera lugares donde parar a comer, teniendo en cuenta las características geográficas y temporales de la ruta. Los usuarios podrán dar cierta información para mejorar las sugerencias, como especificar alergias alimentarias, vegetarianismo, veganismo, etc.



- **Integración con API de hoteles:** se añadirá una funcionalidad que permita que la aplicación sugiera lugares donde pernoctar, teniendo en cuenta las características geográficas y temporales de la ruta.
- **Integración con API de transportes:** se añadirá una funcionalidad que permita a la aplicación sugerir medios de transporte para llegar al punto inicial de la ruta desde la localización actual. Esta función aparecería al acceder al detalle de una ruta.

Como ampliaciones en la aplicación móvil se han pensado las siguientes:

- **Abrir la aplicación al mercado:** subir la aplicación a la tienda de Google Play. Para ello hay que hacerse desarrollador y realizar un pago de veinte euros, por lo que habría que buscar una forma de minimizar el gasto. Una idea sería añadir publicidad en la aplicación en la parte inferior, de manera que no influya en el uso de esta, y así conseguir beneficios con los usos del sistema, pudiendo dejar la aplicación gratis para todo el público.
- **Aplicación multiplataforma:** otra idea, relacionada con el punto anterior, sería desarrollar la misma aplicación para usuarios de iOS. Esto implicaría el desarrollo de una aplicación desde cero, y queda fuera de la planificación de este trabajo.
- **Mejora de la adaptabilidad de las vistas:** aunque se ha procurado que las pantallas sean accesibles desde cualquier dispositivo móvil, existe gran variedad de tamaños de pantalla y resoluciones que se deberían tener en cuenta. El tiempo que se ha podido dedicar al testeo de las vistas es limitado. Es posible que, a la larga, se necesitare alguna actualización para evitar problemas en este aspecto.
- **Mejora en el mapa y envío de notificaciones:** esta ampliación contempla el hacer posible la visualización en el mapa de la localización de los diferentes amigos con los que se ha compartido la ruta. Además, añadir un pequeño chat de manera que siempre que escriba alguien del grupo y un usuario no esté dentro de la aplicación, le llegase una notificación.



APÉNDICES





ESPECIFICACIÓN DE REQUISITOS

Introducción

Propósito

En este documento se podrá observar la especificación de requisitos de un producto de software. Este sistema permitirá a los usuarios organizar y gestionar rutas desde cualquier dispositivo conectado a Internet.

El documento está dirigido al cliente, al equipo de desarrollo del sistema y al equipo de implementación de pruebas. El equipo de mantenimiento puede utilizarlo en el futuro.

Ámbito y Alcance

El sistema se llamará FavoritePath y así será referido a lo largo del documento.

FavoritePath consistirá en un sitio web y una aplicación móvil para la planificación de viajes y rutas indistintamente. Tanto el sitio web como la aplicación móvil deberán permitir a los usuarios registrarse. Una vez registrado, el usuario podrá iniciar sesión para gestionar sus propias rutas y su perfil. Los usuarios no autenticados solo podrán ver las rutas creadas por otros usuarios desde la web.

Tanto el sitio web como la aplicación permitirán la creación, eliminación y edición de rutas. Para su creación, se permitirá al usuario añadir tantos puntos de interés como quiera visitar a lo largo de su viaje, y especificar si quiere que la ruta generada sea circular o no. El sistema calculará entonces el camino óptimo entre todos los puntos elegidos y se lo mostrará al usuario. Se podrán editar otros parámetros, como el nombre y la descripción, así como añadir imágenes.

El sitio web también permitirá crear colecciones de rutas. En estas colecciones se podrán almacenar tanto las rutas creadas por el propio usuario como otras publicadas por usuarios diferentes. Las colecciones podrán ser privadas o públicas, en cuyo caso se mostrarán en el perfil del usuario. Los usuarios podrán marcar como favoritas rutas que no se hayan creado ellos mismos. Estas se guardarán automáticamente en una colección predefinida. Se podrán eliminar las colecciones creadas por el usuario, así como sacar de una colección las rutas que ya no se quieran. Desde la aplicación móvil también se podrán crear colecciones, pero cualquier ruta creada por otro usuario que haya sido compartida con este pasará a formar parte de sus rutas propias.

Los usuarios podrán gestionar su perfil. Aquí podrán personalizar su biografía y su lista de amigos.

Los usuarios podrán enviar peticiones de amistad a otros usuarios. Estas podrán ser aceptadas o rechazadas por el usuario que las reciba. También se podrá eliminar amigos.

El sitio web constará de una sección global, donde se mostrarán las rutas públicas, ordenadas según su popularidad (el número de veces que hayan sido marcadas como favoritas por los usuarios).



Existirá la opción de ordenarlas según su fecha de creación, apareciendo en primer lugar la ruta más reciente. También se podrá filtrar por la zona en la que se encuentre, y por los kilómetros de distancia de la ruta. Se podrá visualizar el detalle de cada ruta.

Existirá una ventana de ayuda donde aparecerá explicada paso a paso toda la funcionalidad de la aplicación.

La aplicación móvil permitirá elegir una de las rutas que el usuario tenga guardada, y ver sus detalles para realizarla. Existirá una opción para dar por comenzada la ruta, que mostrará el camino a seguir para completarla, así como el tiempo estimado para alcanzar el próximo punto de interés.

En la aplicación móvil, existirá la posibilidad de compartir con un grupo de usuarios cada ruta a la hora de realizarla. Para añadir un usuario, este deberá ser amigo del usuario que lo añada.

Además, el usuario podrá denegar el permiso para acceder a su ubicación, pero para poder usar las funcionalidades de la aplicación deberá aceptarlo.

El beneficio de este sistema es poder disfrutar de un mapa, una guía de viajes y la posibilidad de compartir la ruta con amigos, de forma que los viajes sean más fáciles de organizar y llevar a cabo.

Visión global

A continuación, se detallarán de forma exhaustiva los requisitos de FavoritePath.

En el siguiente apartado (2. Descripción general) se describirá en profundidad el sistema: se establecerán las relaciones del sistema con otros sistemas externos, las principales funciones que va a desempeñar el sistema, los usuarios finales del sistema y sus características, restricciones generales que debe seguir el sistema, suposiciones y dependencias, así como los requisitos pendientes de desarrollar en futuras versiones del documento.

En el tercer apartado del documento (3. Requisitos específicos) se detallan los requisitos del sistema, agrupados por categorías: interfaces externas, funcionales, de rendimiento, de base de datos, de desarrollo, restricciones de diseño y otras restricciones, atributos del sistema y otros requisitos.

Finalmente, en los apéndices (Apéndices) se describirán normas de validación para diferentes elementos como el email del usuario, diagramas UML y escenarios del sistema.

Descripción general

Perspectiva del producto

El sistema estará relacionado con:

- Servicio de almacenamiento de datos.
- Proveedor de mapas.



Funciones del producto

Las funcionalidades principales de FavoritePath serán:

Tabla 43. Funcionalidades principales

Usuarios no autenticados	Registrarse en la aplicación. Consultar la página de ayuda.
Usuarios autenticados	Identificarse en la aplicación. Gestionar su perfil. Crear, editar y eliminar rutas propias. Crear y eliminar colecciones de rutas. Añadir rutas a las colecciones. Enviar y recibir peticiones de amistad. Eliminar amigos. Comenzar rutas. Añadir usuarios a rutas. Consultar la página de ayuda.

Además, el usuario autenticado puede llevar a cabo las mismas funcionalidades que los usuarios no autenticados (a excepción de registrarse).

Características de los usuarios

- Usuario no autenticado: usuario estándar, con nociones básicas de navegación.
- Usuario autenticado: usuario con cierta soltura en el sistema.

Restricciones generales

- La aplicación por desarrollar constará de un portal Web y una aplicación móvil.
- La web debe ser compatible con los principales navegadores web.
- La aplicación móvil debe ser compatible con Android.
- La información debe encriptarse con HTTPS.
- El sistema deberá cumplir el Reglamento General de Protección de Datos (RGPD).
- La aplicación debe cumplir el nivel AA de las WCAG.

Requisitos específicos

Requisitos de interfaces externas

Interfaces de usuario

R-IU-1. El portal web deberá ser accesible a los usuarios mediante los principales navegadores web.



R-IU-1.1. Google Chrome.

R-IU-1.2. Mozilla Firefox.

R-IU-1.3. Microsoft Edge.

R-IU-1.4. Safari.

R-IU-2. La aplicación móvil deberá ser compatible con el sistema operativo Android.

Interfaces hardware

N/A.

Interfaces software

R-IS-1. El sistema se deberá comunicar con el servicio de almacenamiento de datos.

R-IS-1.2. Los datos que se guardarán serán:

R-IS-1.2.1. Cuentas de usuario.

R-IS-1.2.2. Amigos.

R-IS-1.2.3. Solicitudes.

R-IS-1.2.4. Colecciones.

R-IS-1.2.5. Rutas.

Interfaces de comunicaciones

R-C-1. El sistema se deberá comunicar con servicios externos.

R-C-1.1. Servicio de almacenamiento de datos.

R-C-1.1.1. El sistema redireccionará a este sistema para la autenticación del usuario.

R-C-1.1.2. Una vez autenticado, se aterrizará en la página principal de FavoritePath.

R-C-1.2. Proveedor de mapas.

R-C-1.2.1. El sistema enviará al proveedor de mapas los datos de geolocalización del usuario.

R-C-1.2.1.1. Se enviarán estos datos siempre y cuando el usuario haya dado su consentimiento.

R-C-1.2.2. El sistema interactuará con el proveedor de mapas para la representación de la información relativa a estos.

R-C-1.2.2.1. Representación del mapa de la zona relativa a la ruta elegida.

R-C-1.2.2.2. Colocación de marcadores en los puntos de interés definidos por el usuario.



R-C-1.2.2.3. Representación del itinerario a seguir en el mapa.

Requisitos funcionales

R1. El sistema proporcionará un método de autenticación para los usuarios no autenticados.

R1.1. El sistema delegará en un servicio externo el proceso de autenticación.

R1.2. Si el proceso finaliza con éxito,

R1.2.1. Se redirigirá al sistema, pidiendo el permiso de geolocalización al usuario.

R1.2.1.1. Si el usuario no acepta el permiso, se le avisará de que no tendrá los permisos necesarios para el correcto funcionamiento de la aplicación.

R1.3. Si es la primera vez que el usuario se autentica:

R1.3.1. El sistema le enviará un correo electrónico confirmando su registro.

R1.3.1.1. El mensaje se enviará al correo utilizado para la autenticación.

R1.3.2. El sistema asociará las colecciones predefinidas a los usuarios autenticados por primera vez.

R1.3.2.1. Colección “Rutas propias” (R4.1.1).

R1.3.2.2. Colección “Rutas favoritas” (R4.2.1).

R1.4. El sistema permitirá al usuario autenticado cerrar sesión.

R2. El sistema proporcionará a los usuarios opciones para ver rutas.

R2.1. Rutas propias o feed privado (R4.1).

R2.1.1. El feed privado solo existirá para los usuarios autenticados.

R2.1.2. Cada usuario podrá acceder únicamente a su feed privado, y no al de otros usuarios.

R2.2. En la aplicación móvil se cargarán todas las rutas al acceder al feed.

R2.3. Se podrá añadir cualquier ruta a una de las colecciones del usuario (R4.5).

R3. El sistema permitirá a los usuarios autenticados y no autenticados ver el detalle de una ruta.

R3.1. Nombre de la ruta.

R3.2. Recorrido en el mapa.

R3.4. Fotos adjuntas.

R3.5. Descripción.

R4. El sistema permitirá a los usuarios autenticados gestionar sus colecciones.

R4.1. Colección “Rutas propias” (o feed privado) (R.2.1).

R4.1.1. Esta colección se creará automáticamente cuando el usuario se autentique por primera vez en el sistema (R1.3.2.1.).

R4.1.2. En ella se guardarán automáticamente todas las rutas que el usuario cree.

R4.2. Colección “Rutas favoritas”.

R4.2.1. Esta colección se creará automáticamente cuando el usuario se autentique por primera vez en el sistema (R1.3.2.2).

R4.3. El usuario podrá crear una colección.

R4.3.1. Se requerirá que el usuario especifique el nombre de la nueva colección.

- R4.4. El usuario podrá eliminar una de sus colecciones.
 - R4.4.1. No se podrán eliminar las colecciones creadas automáticamente.
 - R4.5. El usuario podrá añadir una ruta a una de sus colecciones.
 - R4.5.1. El sistema pedirá al usuario que especifique en cuál de sus colecciones quiere añadir la ruta.
 - R4.5.2. Se podrá añadir la misma ruta a diferentes colecciones.
 - R4.6. El usuario podrá eliminar una ruta de una colección.
- R5. El sistema permitirá a los usuarios autenticados crear rutas.
- R5.1. El sistema requerirá que el usuario especifique:
 - R5.1.1. Datos obligatorios.
 - R5.1.1.1. Nombre de la ruta.
 - R5.1.1.2. Punto de inicio.
 - R5.1.1.3. NUM_PUNTOS puntos de interés.
 - R5.1.1.3.1. NUM_PUNTOS deberá tomar un valor mayor que 0.
 - R5.1.1.4. El usuario deberá elegir entre una ruta lineal o circular.
 - R5.1.1.4.1. La ruta lineal será el camino más corto que atraviese todos los puntos, siendo el punto inicial el primero.
 - R5.1.1.4.2. La ruta circular será el camino más corto que atraviese todos los puntos, siendo el punto inicial el primero y el último.
 - R5.1.1.5. El usuario deberá elegir si quiere que la ruta sea pública o privada.
 - R5.1.2. Datos opcionales.
 - R5.1.2.1. Descripción.
 - R5.1.2.2. Fotografías.
 - R5.2. El sistema guardará:
 - R5.2.1. Todos los datos obligatorios especificados por el usuario.
 - R5.2.2. Todos los datos opcionales que hayan sido rellenados por el usuario.
 - R5.2.3. La longitud en kilómetros de la ruta.
 - R5.2.3.1. Este dato será calculado automáticamente.
 - R5.3. Una vez creada, la ruta aparecerá en la colección “Rutas propias” del usuario (R4.1.2.).
 - R5.4. Una vez creada, si es pública, la ruta aparecerá en el feed global para todos los usuarios (R2.1.1.).
- R6. El sistema permitirá a los usuarios eliminar rutas creadas por ellos mismos.
- R7. El sistema permitirá a los usuarios editar rutas que hayan sido creadas por ellos mismos.
- R8. Los usuarios autenticados podrán gestionar su perfil.
- R8.1. Los usuarios podrán visualizar su perfil.
 - R8.1.1. Nombre de usuario.
 - R8.1.2. Imagen de perfil.
 - R8.1.3. Biografía.
 - R8.2. Los usuarios podrán editar su perfil.
 - R8.2.1. Biografía.



R9. Los usuarios autenticados podrán gestionar sus amigos.

R9.1. El usuario podrá visualizar su lista de amigos.

R9.2. El usuario podrá enviar una petición de amistad a otro usuario.

R9.2.1. El sistema requerirá que el usuario indique el correo electrónico del amigo al que quiere enviar la petición.

R9.2.2. El sistema avisará al usuario de la nueva solicitud de amistad mostrando el remitente de esta.

R9.3. El usuario podrá recibir peticiones de amistad de otros usuarios.

R9.3.1. El usuario podrá aceptar la petición.

R9.3.1.1. El amigo se añadirá a la lista de amigos del usuario.

R9.3.1.2. El usuario se añadirá a la lista de amigos del amigo.

R9.3.2. El usuario podrá rechazar la petición.

R9.4. El usuario podrá eliminar un amigo de su lista de amigos.

R10. El sistema contará con una ventana de ayuda.

R10.1. Esta ventana contendrá las FAQ y otra información relevante para el usuario.

R11. La aplicación móvil permitirá al usuario iniciar una ruta dentro de sus colecciones.

R11.1. El sistema mostrará una iteración de indicaciones para llegar al siguiente punto de interés no alcanzado de la ruta.

R11.2. El sistema permitirá al usuario finalizar una ruta iniciada.

R12. La aplicación móvil permitirá al usuario compartir una ruta iniciada con cualquier amigo.

R12.1. Si se acepta la invitación, el sistema añadirá la ruta a su colección "Rutas compartidas".

R12.1.1. El sistema iniciará la ruta como en R11.

R12.2 Si no se acepta la invitación, el sistema mostrará una notificación del fallo de invitación al usuario.

Requisitos de rendimiento

R-Rend-1. Ninguna pantalla tardará más de 5 segundos en cargar.

Atributos del sistema

AS-1 El sistema debe cargar por completo en menos de 5 segundos.

AS-2 El sistema debe responder a las acciones del usuario en menos de 5 segundos.

AS-3 El sistema debe estar disponible el 99.9% del tiempo.

Diagramas UML

Diagrama de contexto

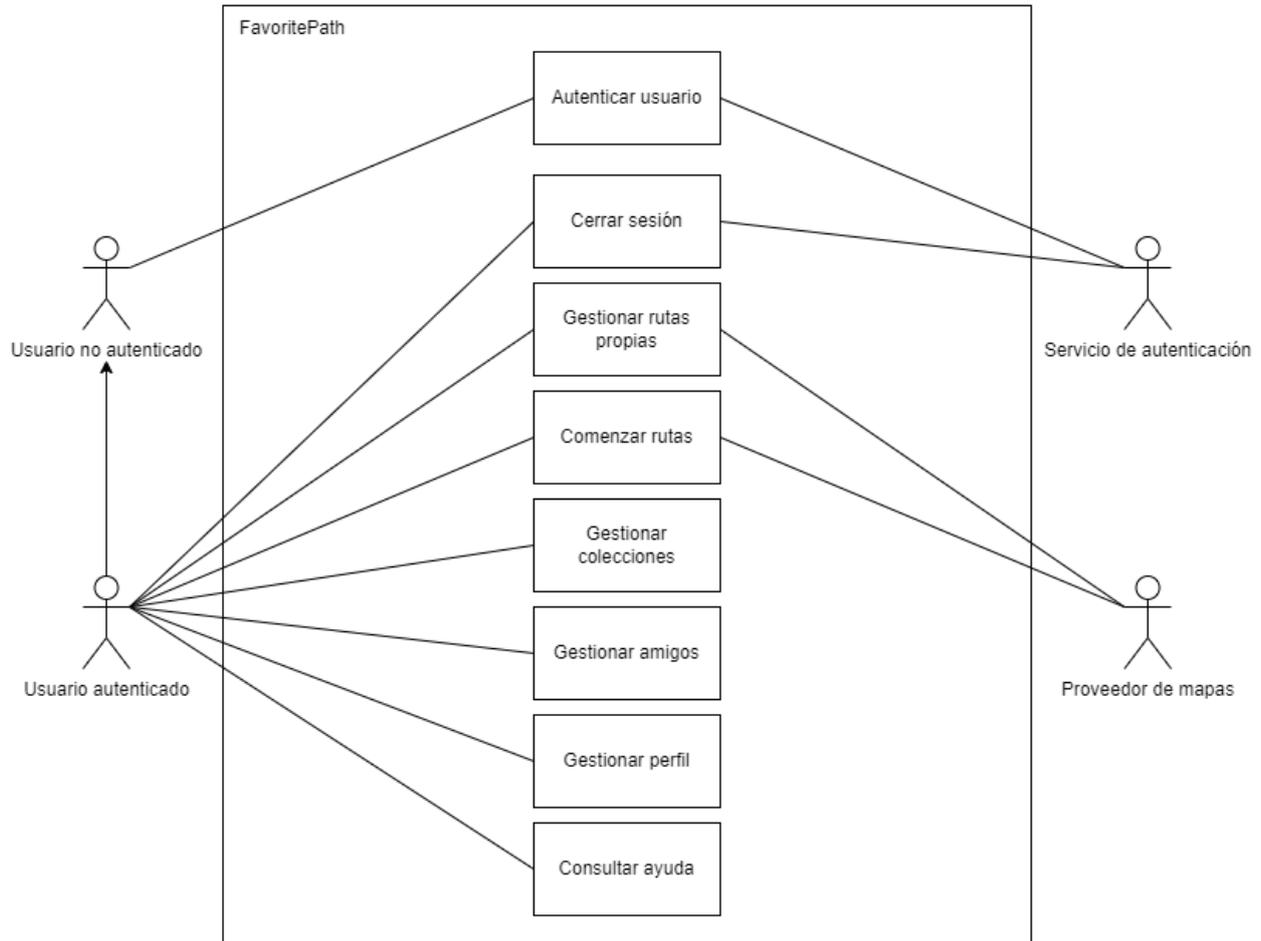


Ilustración 60. Diagrama de contexto

Diagramas de casos de uso

Acceso al feed global

Acceso al feed global

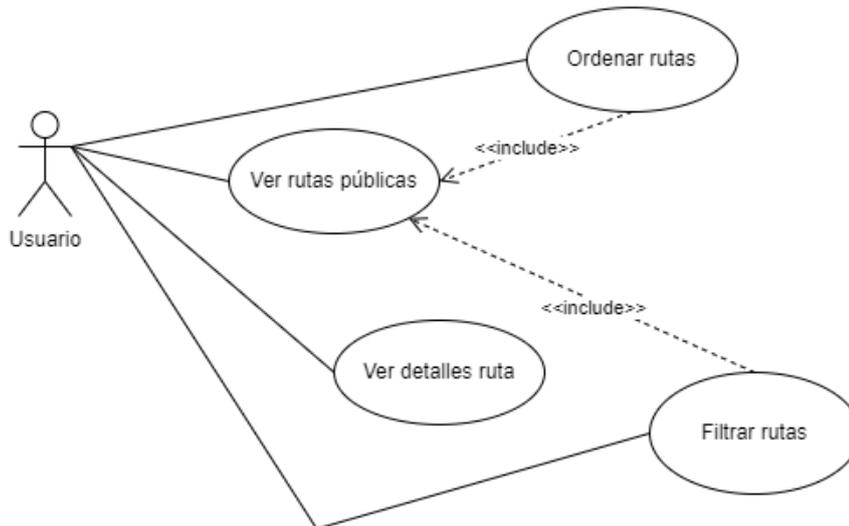


Ilustración 61. Caso de Uso Acceso al Feed Global

Gestión de rutas propias

Gestionar rutas

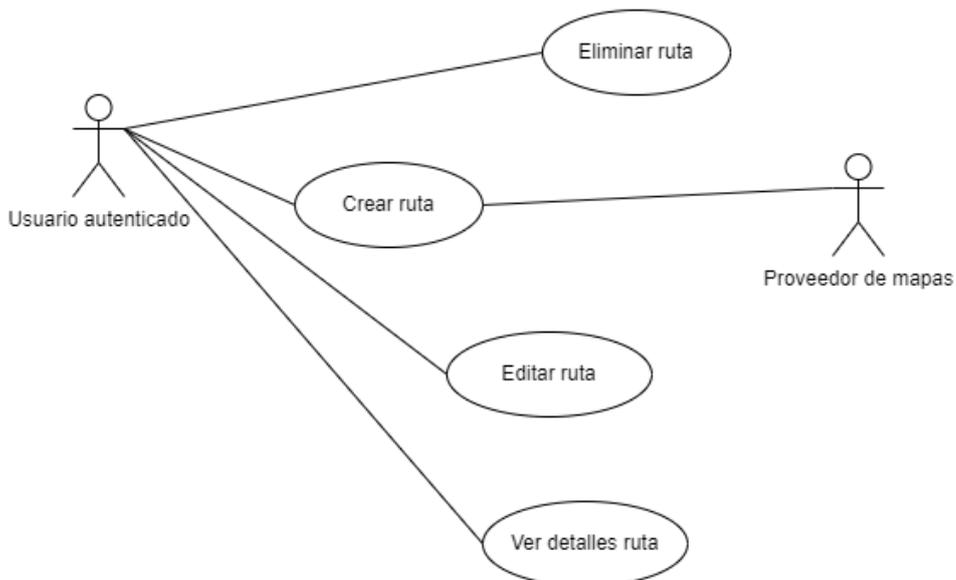


Ilustración 62. Caso de Uso Gestión de Rutas Propias

Gestión de colecciones

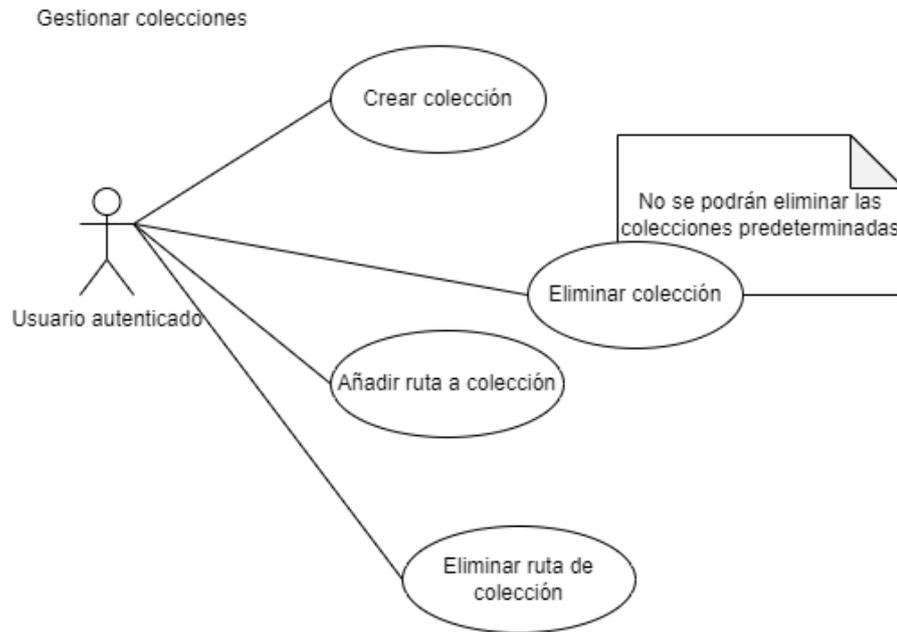


Ilustración 63. Caso de Uso Gestión de Colecciones

Diagrama de estados

Ruta

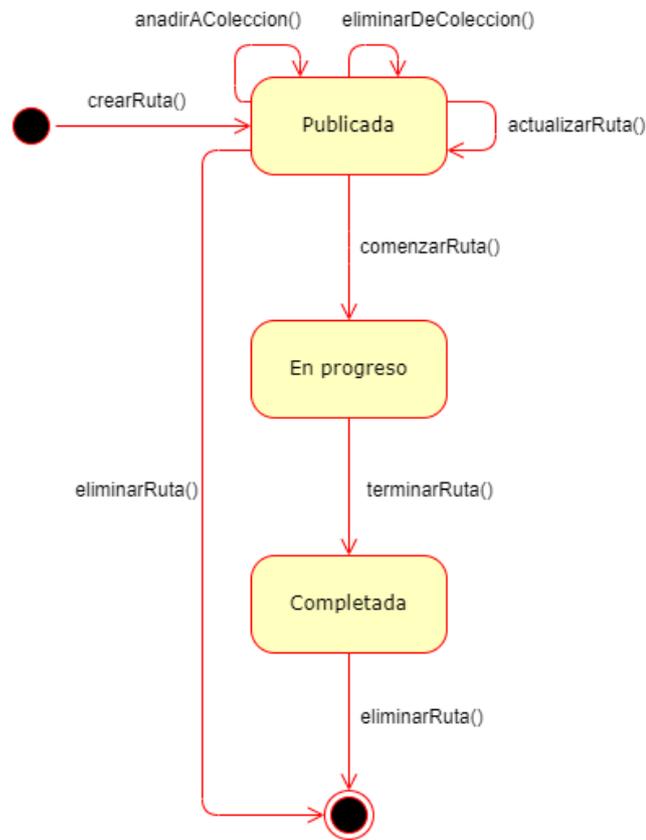


Ilustración 64. Diagrama de Estados de Ruta

Escenarios

Tabla 44. Escenario móvil 1

Escenario móvil "Eliminar una colección de rutas con éxito"	
Numeración:	1
Autor:	Francisco Manuel Fernández Suárez
Precondiciones:	Debe existir una colección diferente a las predeterminadas
Postcondiciones:	-
Quién lo comienza:	Usuario registrado
Quién lo finaliza:	El sistema
Excepciones:	-
Descripción:	El usuario accede a su listado de colecciones de rutas. El usuario elige una colección creada manualmente por él con anterioridad.

El sistema borrará la colección y las rutas incluidas en ella.

Tabla 45. Escenario móvil 2

Escenario móvil "Editar el nombre de una ruta con éxito"
Numeración: 2
Autor: Francisco Manuel Fernández Suárez
Precondiciones: Debe existir una ruta creada por el usuario
Postcondiciones: -
Quién lo comienza: Usuario registrado
Quién lo finaliza: El sistema
Excepciones: El usuario no podrá dejar el nombre vacío
Descripción: El usuario accede a su listado de rutas. El usuario elige una ruta creada manualmente y accede a la pantalla de edición de ruta. El usuario elimina el nombre de la ruta y escribe "Nombre actualizado". El usuario aceptará la edición y el sistema actualizará la ruta en la base de datos.

Tabla 46. Escenario web 1

Escenario web "Guardar en favoritos una ruta de entre 4km y 6km de longitud"
Numeración: 3
Autor: Sara Rubín Estrada-Nora
Precondiciones: -
Postcondiciones: -
Quién lo comienza: Usuario
Quién lo finaliza: El sistema
Excepciones: -
Descripción: El usuario accede al feed global. El sistema muestra 15 rutas ordenadas por popularidad. El usuario utiliza el filtro, eligiendo un rango entre 4km y 6km. El sistema muestra 15 rutas comprendidas entre esas longitudes, ordenadas por popularidad. El usuario pulsa el icono de "favorita" de una de las rutas mostradas. El sistema guarda la ruta en la colección "Rutas favoritas" del usuario.

Tabla 47. Escenario web 2

Escenario web "Añadir un usuario como amigo desde su perfil"
Numeración: 4
Autor: Sara Rubín Estrada-Nora



Precondiciones: -
Postcondiciones: El usuario acepta la solicitud
Quién lo comienza: Usuario registrado
Quién lo finaliza: El sistema
Excepciones: -
Descripción: El usuario accede al feed global. El sistema muestra 15 rutas ordenadas por popularidad. El usuario elige una ruta y pulsa en ella. El sistema muestra los detalles de la ruta seleccionada. El usuario pulsa sobre el usuario creador de la ruta. El sistema muestra el perfil del usuario creador de la ruta. El usuario selecciona la opción "Añadir como amigo". El sistema envía una solicitud al otro usuario. El sistema notifica al otro usuario.

Tabla 48. Escenario web 3

Escenario web "Crear una ruta con un nombre ya existente"
Numeración: 5
Autor: Sara Rubín Estrada-Nora
Precondiciones: Existe una ruta de mismo nombre que la que se va a crear
Postcondiciones: -
Quién lo comienza: Usuario registrado
Quién lo finaliza: El sistema
Excepciones: -
Descripción: El usuario accede a su listado privado de rutas. El sistema muestra 15 de sus rutas, ordenadas por popularidad. El usuario elige la opción de creación de rutas. El sistema accede a la página de creación de rutas. El usuario elige varios puntos de interés en el mapa, elige que la ruta sea circular y pública, y escribe un nombre igual al de otra ruta que ya existe en el sistema. A continuación, pulsa la opción de guardar. El sistema crea la ruta y la añade a la base de datos.

PLAN DE GESTIÓN DE RIESGOS

Metodología

Para realizar de forma eficiente una gestión de riesgos es necesario seguir una metodología, cuyo resultado pueda servir para identificar los riesgos del proyecto y definir estrategias para evaluarlos y

monitorizarlos. Para este proyecto se seguirá la metodología de Boehm, por ser ya conocida y aportar buenos resultados.

Esta metodología divide la gestión de riesgos en dos fases principales:

1. Valoración de riesgos: fase que trata con los trabajos relacionados con la identificación, análisis y priorización de los riesgos.
2. Control de riesgos: fase que define las estrategias a aplicar el riesgo, la manera de identificar los indicadores y la monitorización.

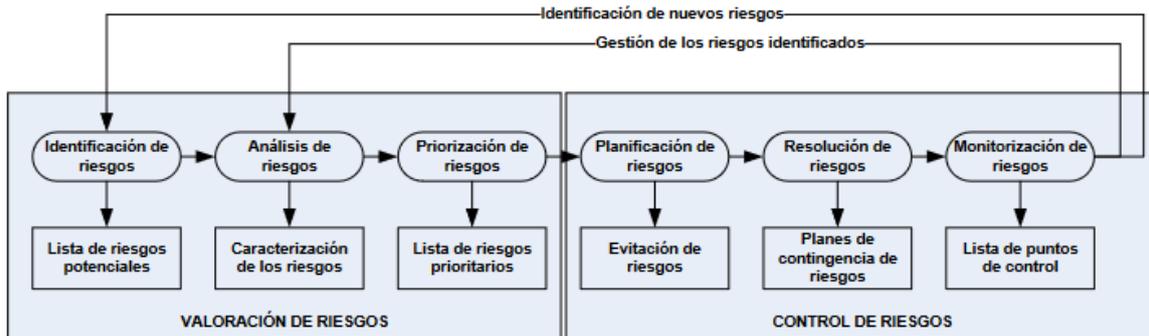


Ilustración 65. Diagrama Metodología de Boehm

Atendiendo al esquema de la metodología de Boehm, se pueden identificar seis pasos desde la identificación de riesgos hasta la monitorización:

- **Identificación de Riesgos:** en este proceso se genera una lista de riesgos potenciales para el proyecto.
- **Análisis de Riesgos:** en este proceso se mide el impacto, la probabilidad de cada riesgo y los niveles de riesgo de los métodos alternativos
- **Priorización de los Riesgos:** se genera una lista de riesgos ordenados por su importancia en el proyecto de manera que se prioricen unos riesgos frente a otros. De esta lista se extraen los riesgos para el Control de riesgos.
- **Planificación de la gestión de Riesgos:** se genera un plan para cada riesgo significativo. Un riesgo será significativo en función del umbral de riesgo, del apetito por el riesgo, etc. en función del resultado de la priorización.
- **Resolución de Riesgos:** es el proceso de ejecución del Plan de Riesgos para resolver cada uno de los riesgos significativos.
- **Monitorización de Riesgos:** es la actividad de seguimiento de los riesgos, usualmente evaluando los indicadores de los factores de riesgo. También incluye tareas como la eliminación de algunos riesgos, la identificación de otros nuevos o la reevaluación de riesgos que no fueron priorizados en su momento

Categorías de Riesgo

Las categorías de Riesgo usadas son las pertenecientes al PMBOK y proveen de una estructura conocida como RBS (Risk Breakdown Structure) que facilita la implementación y el manejo de los riesgos.

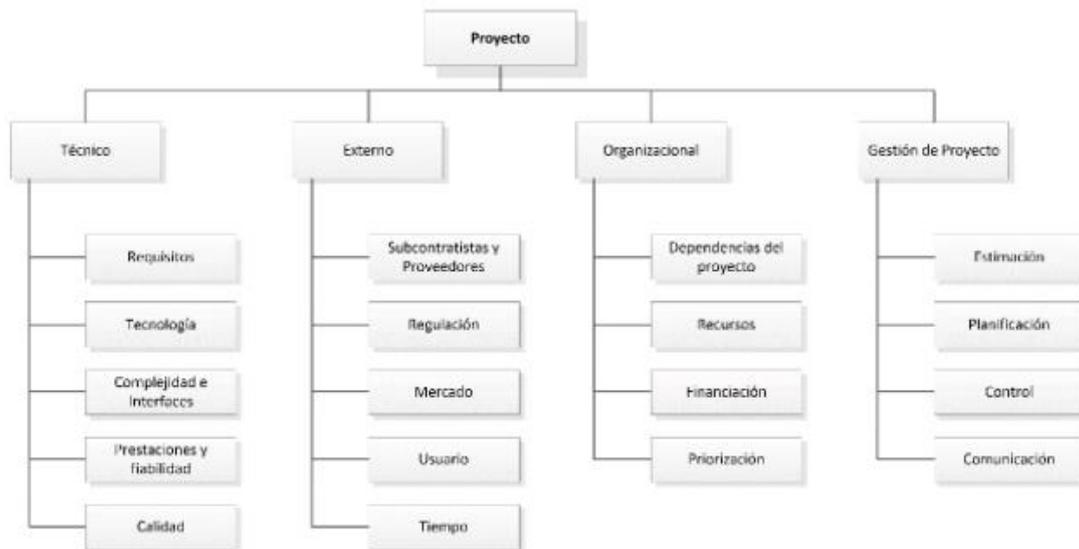


Ilustración 66. Estructura de Desglose de Riesgos

Definiciones de riesgo

En los proyectos se debe medir la probabilidad de que un riesgo tenga lugar y ejerza sus efectos, tanto dañinos como beneficiosos.

Como no se pueden seguir todos los riesgos del proyecto, se debe priorizar atendiendo a su probabilidad de ocurrencia y a su respectivo impacto. Para ello, se define la tabla de probabilidad.

Tabla 49. Tabla de probabilidad de riesgos

Etiqueta	Rango	Valor usado para los cálculos
Muy bajo	[0%...20%]	10%
Bajo	(20%...40%]	30%
Medio	(40%...60%]	50%
Alto	(60%...80%]	70%
Muy alto	(80%...100%]	90%

El impacto del riesgo sobre el proyecto debe definirse para cada uno de los objetivos de este. Se usarán escalas de impacto para los riesgos negativos y otras diferentes para los riesgos positivos.

Matriz de probabilidad e impacto

La Matriz de Probabilidad e Impacto se utiliza para definir los valores que se utilizan a la hora de priorizar los riesgos basándose en la probabilidad de que un riesgo ocurra y a su impacto en el proyecto.

En esta matriz es donde se ven reflejadas las políticas y actitudes de la organización frente al riesgo. Además, la matriz debe definirse tanto para los riesgos negativos como para los positivos.

Matriz de Probabilidad e Impacto										
Probabilidad	Amenazas					Oportunidades				
0,90	0,05	0,09	0,18	0,36	0,72	0,72	0,36	0,18	0,09	0,05
0,70	0,04	0,07	0,14	0,28	0,56	0,56	0,28	0,14	0,07	0,04
0,50	0,03	0,05	0,10	0,20	0,40	0,40	0,20	0,10	0,05	0,03
0,30	0,02	0,03	0,06	0,12	0,24	0,24	0,12	0,06	0,03	0,02
0,10	0,01	0,01	0,02	0,04	0,08	0,08	0,04	0,02	0,01	0,01
	0,05 Muy bajo	0,10 Bajo	0,20 Moderado	0,40 Alto	0,80 Muy Alto	0,80 Muy Alto	0,40 Alto	0,20 Moderado	0,10 Bajo	0,05 Muy Bajo
	Impacto Negativo					Impacto Positivo				

Cada riesgo es evaluado de acuerdo a la probabilidad de que ocurra y al impacto en algún objetivo si ocurriera. Los umbrales de tolerancia de cada organización se trasladan a la matriz, de manera que las áreas verde, amarilla y roja indiquen estos umbrales para la priorización de riesgos.

Ilustración 67. Matriz de Probabilidad e Impacto

REFERENCIAS BIBLIOGRÁFICAS

J. M. Redondo, «Documentos-modelo para Trabajos de Fin de Grado/Master de la Escuela de Informática de Oviedo,» 17 6 2019. [En línea]. Available: https://www.researchgate.net/publication/327882831_Plantilla_de_Proyectos_de_Fin_de_Carrera_de_la_Escuela_de_Informatica_de_Oviedo.

J. Redondo, «Creación y evaluación de plantillas para trabajos de fin de grado como buena práctica docente.,» Revista de Innovación y Buenas Prácticas Docentes, p. pp, 2020.

Krug, Steve, "No me hagas pensar", 2015.

Mattish. (2021, 1 de junio). How Do the Top Ecommerce Sites Display Filters on Their Listings Pages? Recuperado de <https://mattish.com/blog/post/how-do-the-top-ecommerce-sites-display-filters-on-their-listings-pages>

Aukera. (s.f.). Bases de datos relacionales vs no relacionales. Recuperado de <https://aukera.es/blog/bases-de-datos-relacionales-vs-no-relacionales/>

OpenWebinars. (s.f.). ¿Qué es Serverless? Ventajas y servicios. Recuperado de <https://openwebinars.net/blog/que-es-serverless-ventajas-y-servicios/>



FreeCodeCamp. (s.f.). Ventajas y desventajas de JavaScript. Recuperado de <https://www.freecodecamp.org/espanol/news/ventajas-y-desventajas-de-javascript/>

MongoDB. (s.f.). MongoDB Manual. Recuperado de <https://www.mongodb.com/docs/manual/>

StackShare. (s.f.). MongoDB - StackShare. Recuperado de <https://stackshare.io/mongodb>

Last Week in AWS. (s.f.). 10 Free Cloud Databases You Should Consider (and 1 You Shouldn't) - Last Week in AWS. Recuperado de <https://www.lastweekinaws.com/blog/10-free-cloud-databases-you-should-consider-and-1-you-shouldnt/>

Firebase. (s.f.). Firebase. Recuperado de <https://firebase.google.com/>

StackShare. (s.f.). Firebase Authentication Alternatives. Recuperado de <https://stackshare.io/firebase-authentication/alternatives>

Auth0. (s.f.). Introduction to Web Authentication. Recuperado de <https://auth0.com/blog/introduction-to-web-authentication/>

WebHostingSecretRevealed. (s.f.). 9 Mejores Sitios de Hosting Gratuito (2022). Recuperado de <https://www.webhostingsecretrevealed.net/es/blog/web-hosting-guides/free-hosting-sites/>

HubSpot. (s.f.). ¿Qué es un Framework de Desarrollo Web? [Blog post]. Recuperado de <https://blog.hubspot.es/website/framework-desarrollo-web>

Aglowid IT Solutions. (s.f.). Firebase vs MongoDB: ¿Cuál es la mejor opción para tu aplicación? Recuperado de <https://aglowiditsolutions.com/blog/firebase-vs-mongodb/>

StackShare. (s.f.). Firebase Authentication vs MongoDB. Recuperado de <https://stackshare.io/stackups/firebase-authentication-vs-mongodb>

StackShare. (s.f.). Firebase vs GitHub Pages. Recuperado de <https://stackshare.io/stackups/firebase-vs-github-pages#:~:text=%22Realtime%20backend%20made%20easy%22%2C,why%20GitHub%20Pages%20is%20favored.>

Back4App. (s.f.). Firebase Pricing - Hosting. Recuperado de https://blog.back4app.com/firebase-pricing/#Firebase_Hosting%20Pricing.

Holded. (s.f.). Amortización de equipos informáticos. Recuperado de <https://www.holded.com/es/blog/amortizacion-de-equipos-informaticos>

DiagramasUML.com. (s.f.). Diagramas UML - Herramienta Online para Crear Componentes. Recuperado de https://diagramasuml.com/componentes/?utm_content=cmp-true.

IONOS Digital Guide. (s.f.). Diagrama de componentes: ¿Qué es y cómo se utiliza? Recuperado de <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/diagrama-de-componentes/>



Node.js. (s.f.). API documentation - Packages. Recuperado de <https://nodejs.org/api/packages.html#exports>

Geekflare. (s.f.). Cómo construir una API con Firebase. Recuperado de <https://geekflare.com/es/build-api-with-firebase/>

freeCodeCamp. (s.f.). Module.exports: How to Export in Node.js and JavaScript. Recuperado de <https://www.freecodecamp.org/news/module-exports-how-to-export-in-node-js-and-javascript/>

Google Cloud. (s.f.). Firestore. Recuperado de <https://cloud.google.com/firestore>

Mozilla. (s.f.). Cross-Origin-Opener-Policy - HTTP. Recuperado de <https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cross-Origin-Opener-Policy>

Mozilla. (s.f.). Cross-Origin-Resource-Policy - HTTP. Recuperado de <https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cross-Origin-Resource-Policy>

Mozilla. (s.f.). JavaScript. Recuperado de <https://developer.mozilla.org/en-US/docs/Web/JavaScript>

International Organization for Standardization (ISO). (2015). ISO 9000:2015. Sistemas de gestión de la calidad - Fundamentos y vocabulario. Recuperado de <https://www.iso.org/obp/ui/es/#iso:std:iso:9000:ed-4:v1:es>

Boletín Oficial del Estado (BOE). (2018). Real Decreto 951/2018, de 27 de julio, por el que se desarrolla la estructura orgánica básica del Ministerio de Sanidad, Consumo y Bienestar Social y se modifica el Real Decreto 595/2018, de 22 de junio, por el que se establece la estructura orgánica básica de los departamentos ministeriales. Recuperado de <https://www.boe.es/buscar/pdf/2018/BOE-A-2018-16673-consolidado.pdf>

Boletín Oficial del Estado (BOE). (2016). Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo, de 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos. Recuperado de <https://www.boe.es/buscar/doc.php?id=DOUE-L-2016-80807>

Jooble. (s.f.). Salario de Analista de Sistemas. Recuperado de <https://es.jooble.org/salary/analista-de-sistemas>

Jooble. (s.f.). Salario de Junior UI/UX Designer en Madrid. Recuperado de <https://es.jooble.org/salary/junior-ui-ux-designer/Madrid>

Formación y Estudios. (s.f.). ¿A qué se dedica un arquitecto de software? Recuperado de <https://www.formacionyestudios.com/a-que-se-dedica-un-arquitecto-de-software.html>

Jobted. (s.f.). Salario de un programador. Recuperado de <https://www.jobted.es/salario/programador>

Glassdoor. (s.f.). Sueldo de un Junior QA Tester. Recuperado de https://www.glassdoor.es/Sueldos/junior-qa-tester-sueldo-SRCH_K00,16.htm



Project Management Institute, 2013. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) Fifth Edition. Recuperado de <https://archive.org/details/pmbok-guide-5th-edition-pmi>

Juan Fuente, Aquilino A. y López Pérez, Benjamín. 2021. Guía de Aprendizaje de la asignatura de Dirección y Planificación de Proyectos Informáticos. Versión 0.093. Dirección y Planificación de Proyectos Informáticos. Departamento de Informática. Universidad de Oviedo.



CONTENIDO ENTREGADO EN LOS ANEXOS

Directorio	Contenido
./	Directorio raíz donde se encuentran los siguientes directorios y un fichero llamado "Leeme"
./apk	Contiene un código QR con el archivo apk del proyecto para su descarga en un dispositivo móvil
./doc	Contiene los ficheros generados de Javadoc, solo haría falta abrir el fichero "index"
./src	Ficheros de código fuente de la aplicación
./documentacion	Contiene la documentación asociada al proyecto como la lista de riesgos, la planificación tanto inicial como final y el presupuesto.