



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Aedo, Aplicación de experiencias turísticas entre usuarios:
Frontend Web

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Palacios Martinez, Diego

Tutor/a: Bort Mir, Lorena

CURSO ACADÉMICO: 2022/2023

El proyecto “AedoApp” ha sido realizado en conjunto por cuatro alumnos de la Escuela Técnica Superior de Ingeniería Informática (ETSINF) de la Universidad Politécnica de Valencia en el campus de Vera durante el curso 2022/2023. Este proyecto se ha desarrollado en el marco de Start.inf el espacio de emprendimiento de la ETSINF.

Siguiendo las recomendaciones que ofrece la ETSINF para este tipo de proyectos de emprendimiento, el trabajo se ha realizado de manera conjunta, pero cada alumno presenta de forma separada su TFG asociado a la parte técnica del trabajo en la que ha sido más protagonista. Sin embargo, por ser el mismo proyecto y para que la memoria esté autocontenida, varios apartados del TFG son comunes y han sido elaborados de forma colaborativa. Ellos son: evaluación y desarrollo de la idea de negocio, la sección de “Objetivos de desarrollo sostenibles” y los anexos “Manual de uso de la aplicación”, “Preguntas de la encuesta a los usuarios” y “Experimentos realizados”.

Los alumnos participantes en el proyecto “AedoApp” y las áreas de trabajo que cada uno presenta en su TFG son:

- Hernández Bonet, Álvaro: Inteligencia artificial
- Matarredona Coloma, Joan: Frontend de la aplicación móvil
- Mut Portes, Andreu: Backend de la aplicación
- Palacios Martínez, Diego: Frontend de la aplicación web

Resumen

En la actualidad, el turismo es considerado una de las principales industrias a nivel mundial. Sin embargo, es común observar cómo gran parte del turismo se enfoca únicamente en visitar unos puntos históricos y emblemáticos de una ciudad o región, comprar unos recuerdos y hacer una foto que capture el momento, desperdiciando la oportunidad de sumergirse verdaderamente en la cultura y el estilo de vida de la zona.

En el contexto de España, se observa que muchos turistas se concentran en las grandes ciudades y destinos turísticos populares, dejando de lado la oportunidad de explorar y conocer la España despoblada. Estas son las regiones rurales y menos visitadas, caracterizadas por su belleza natural, su autenticidad y su patrimonio cultural único.

Por lo anterior, en este TFG se establece como objetivo el desarrollo de una idea de negocio asociada a una aplicación móvil que fomente el turismo sostenible y responsable ofreciendo experiencias personalizadas, al tiempo que se promueva la movilidad del usuario. A lo largo del informe, se detallará la evaluación y el desarrollo de la idea de negocio para llevar a cabo la aplicación, así como la implementación de la página web correspondiente.

Palabras clave: turismo, movilidad, página web, España profunda, experiencia personalizada.

Abstract

Currently, tourism is considered one of the main industries worldwide. However, it is common to observe how much of tourism focuses solely on visiting historical and emblematic landmarks in a city or region, buying souvenirs, and taking photos to capture the moment, missing out on the opportunity to truly immerse oneself in the culture and lifestyle of the area.

In the context of Spain, it can be observed that many tourists concentrate on large cities and popular tourist destinations, overlooking the opportunity to explore and get to know the deep rural Spain. These rural and less-visited regions are characterized by their natural beauty, authenticity, and unique cultural heritage.

Therefore, this undergraduate thesis aims to develop a business idea associated with a mobile application that promotes sustainable and responsible tourism while offering personalized experiences and enhancing user mobility. The report will provide detailed assessment and development of the business idea for the application, as well as the implementation of the corresponding website.

Keywords: tourism, mobility, website, deep rural Spain, personalized experience.



Índice general

Contenido

Contenido	4
1. Introducción	9
1.1 Introducción.....	9
1.2 Objetivos	9
1.3 Estructura de la memoria	10
2. Evaluación de la idea de negocio.....	10
2.1 Resumen de Aedo	10
2.2 Idea de negocio	11
2.3 Estudio de mercado	12
2.3.1 Civitatis	12
2.3.2 Guruwalk	14
2.3.3 Freetour.com	14
2.3.4 Identify.....	16
2.3.5 Withlocals	16
2.3.6 Tabla comparativa	17
2.4 Modelo de negocio.....	19
2.5 Proyección económica a 6 años	20
2.6 Análisis DAFO.....	22
2.7 Conclusiones de la evaluación	22
3. Desarrollo de la idea de negocio.....	23
3.1 Desarrollo del primer MVP.....	23
3.1.1 Experimento 1: Feria de proyectos	24
3.2 Desarrollo del segundo MVP	24
3.2.1 Experimento 2: Presentar la aplicación a un trabajador del campo.	25
3.2.2 Experimento 3: Presentar la aplicación a un profesor	25
4. Aspectos técnicos.....	26
4.1 Herramientas utilizadas.....	26
4.2 Entornos de desarrollo.....	27
4.3 Modelo de datos.....	28
4.4 Estructura del proyecto.....	29
4.5 Documentación.....	30
4.6 Estilo artístico.....	31

4.7	Características exclusivas de la aplicación web.....	34
4.8	Buenas prácticas	35
4.8.1	Componentes dumb y Smart.....	36
4.9	Librerías utilizadas.....	37
4.10	Componentes utilizados.....	39
4.10.1	MatToolBar.....	39
4.10.2	MatSideNav	39
4.10.3	MatCard	40
4.10.4	MatButton.....	42
4.10.5	MatFormField y MatInput.....	42
4.10.6	MatTable.....	43
4.8	Desafíos de programación	44
4.8.1	Subida de datos TypeScript a Firebase Firestore	44
4.8.2	Comunicación efectiva entre equipos	46
4.8.3	Uso de servicios personalizados en la base de datos de Firebase	47
5.	Cronología del TFG	49
5.1	Sprint 1 (PIN: Proyecto de Ingeniería del <i>software</i>).....	49
5.2	Sprint 2 (PIN: Proyecto de Ingeniería del <i>software</i>).....	49
5.3	Sprint 3 (PIN: Proyecto de Ingeniería del <i>software</i>).....	50
5.4	Sprint 4.....	50
5.5	Sprint 5.....	51
5.6	Sprint 6.....	51
6.	Conclusiones y trabajo futuro.....	52
6.1	Conclusiones	52
6.2	Trabajo futuro.....	52
7.	Glosario de términos	53
8.	Bibliografía	54
9.	Anexo I: Manual de uso de la aplicación.....	55
9.1	Manual de uso móvil.....	55
	Pantalla de inicio.....	55
	Pantalla de información de una experiencia.....	56
	Pantalla de selección de fecha de reserva.....	57
	Pantalla de buscar:	58
	Pantalla del mapa:.....	59
	Pantalla de tus reservas.....	60
	Pantalla de inicio de sesión.....	61



Pantalla de registro.....	62
Pantalla de perfil de usuario.....	63
Pantalla modificar perfil.....	64
Pantalla crear experiencias.....	65
Pantalla periodicidad.....	66
Pantalla selección de rango de fechas.	67
Pantalla gestión de experiencias.	68
9.2 Manual de uso web	69
Pantalla principal:	69
Pantalla de administración:	69
Pantalla de registro de actividades:.....	70
Pantalla de perfil del usuario.....	71
Pantalla de información de una experiencia.....	71
Pantalla de inicio de sesión	72
Pantalla de registro.....	72
10. Anexo II: Preguntas de la encuesta al usuario	73
11. Anexo III: Experimentos realizados.....	75
12. Anexo IV: Objetivos de desarrollo sostenible	78

Índice de figuras

Figura 1: Lean canvas Aedo App (Fuente: Elaboración propia)	12
Figura 2: Web Civitatis (Fuente: civitatis.com/es/).....	13
Figura 3: Web Civitatis (Fuente: civitatis.com/es/).....	13
Figura 4: Captura de GuruWalk (Fuente: Google play GuruWalk).....	14
Figura 5: Web FreeTour.com (Fuente: https://www.freetour.com/).....	15
Figura 6: Web de FreeTour.com (Fuente: freetour.com/es/).....	15
Figura 7: Web Identify (Fuente: identifytravel.app)	16
Figura 8: Withlocals (Fuente: withlocals.com/es/).....	17
Figura 9: Proyección económica (Fuente: Elaboración propia.).....	21
Figura 10: Resultado semestral acumulado (Fuente: elaboración propia).....	21
Figura 11: Mapa de características inicial (Fuente: Elaboración propia)	23
Figura 12: Fotografía del stand y el equipo de Aedo (Fuente: Fotografía de la Etsinf).....	24
Figura 13: Modelo de datos (Fuente: Elaboración propia)	28
Figura 14: Estructura de carpetas.....	29
Figura 15: Servicio documentado (Fuente: Elaboración propia)	30
Figura 16: Documentación de servicio en Compodoc (Fuente: Elaboración propia)	31
Figura 17: Logotipo de la aplicación (Fuente: Elaboración propia)	32
Figura 18: Paleta de colores de la aplicación (Fuente: Elaboración propia).....	32
Figura 19: Tipografía de la aplicación (Fuente: Elaboración propia).....	33
Figura 20: Interfaz de Firestore (Fuente: https://firebase.google.com)	34
Figura 21: Dashboard de Aedo (Fuente: Elaboración propia).....	35
Figura 22: Representación de flujo de datos entre componentes (Fuente: https://generic-ui.com/blog/enterprise-approach-to-the-smart-and-dumb-components-pattern-in-angular)	36
Figura 23: Ejemplo de uso de un componente dumb (Fuente: Elaboración propia)	37
Figura 24: Representación de un componente dumb (Fuente: Elaboración propia)	37
Figura 25: Lista de dependencias (Fuente: Elaboración propia).....	38
Figura 26: Toolbar de Aedo (Fuente: Elaboración propia).....	39
Figura 27: Sidenav y toolbar de Aedo en conjunto (Fuente: Elaboración propia).....	40
Figura 28: MatCard como contenedor de componentes (Fuente: Elaboración propia).....	41
Figura 29: MatCard como componente de representación (Fuente: Elaboración propia)	41
Figura 30: Tres paletas de color en botones (Fuente: Elaboración propia).....	42
Figura 31: Formulario creación de actividad (Fuente: Elaboración propia)	43
Figura 32: Tabla de gestión de roles de usuario (Fuente: Elaboración propia)	44
Figura 33: Método toJSON de Odisea (Fuente: Elaboración propia)	45
Figura 34: Recuperación de comentario (Fuente: Elaboración propia).....	46
Figura 35: Servicio genérico (Fuente: Elaboración propia).....	47
Figura 36: Servicio específico (Fuente: Elaboración propia).....	48

Índice de tablas

Tabla 1: Tabla comparativa Aedo con otras aplicaciones. (Fuente: Elaboración propia).....	17
Tabla 2: Matriz DAFO (Fuente: Elaboración propia.)	22

1. Introducción

1.1 Introducción

Cuando se piensa en una ciudad o una región, se suelen destacar una serie de lugares o experiencias de mayor calado turístico que la gran mayoría de las personas conoce, pero ¿qué ocurre con todas las tradiciones y la cultura que se esconde tras esos sitios?

Actualmente, existen en el mercado numerosas aplicaciones para ayudar al trotamundos que no sabe qué actividades realizar en la localidad de destino a la que se dirige. Estas aplicaciones ofrecen, por lo general, una serie de tours convencionales que suelen dejar una imagen sesgada de la cultura, las tradiciones o la realidad de la sociedad de una ciudad o una región. Un turista que visita la ciudad de Valencia suele ir a la Ciudad de las Artes y las Ciencias, a la Catedral, a la Plaza del Ayuntamiento, a probar una paella en la playa... pero ¿de dónde viene ese arroz que se ha comido en la paella, qué importancia tiene la preparación de la paella para la cultura valenciana, qué rito gastronómico supone la típica reunión familiar de los domingos para degustar ese plato? O ¿qué supone el “esmorzaret” para un valenciano? ¿Cuál es un típico “esmorzaret”? Todas estas cuestiones son las que al final marcan el carácter y la tradición de una sociedad, y estos elementos no tienen cabida en la típica visita a la ciudad.

Aedo nace de la necesidad y de la importancia de poder ofrecer unas actividades turísticas de calidad a los trotamundos, que permitan poner en valor las tradiciones y la cultura de cualquier lugar, por pequeño o inhóspito que sea. Además, ofrece una plataforma a todas estas localidades para poder publicitar la riqueza de sus tierras, atrayendo a nuevos visitantes con todas las ventajas que esto puede suponer.

1.2 Objetivos

La finalidad de este trabajo es realizar un proyecto de emprendimiento que ofrezca como resultado una aplicación que pueda ser usada por cualquier participante y en la que pueda encontrar actividades (gastronómicas, culturales, al aire libre...) que se ofrezcan en distintas poblaciones de España realizadas por gente local. Para conseguir este producto se han definido una serie de objetivos comunes a todos los miembros del proyecto que habría que alcanzar durante el desarrollo de la aplicación:

- Elaborar, desarrollar y evaluar una idea de negocio teniendo en cuenta el mercado actual.
- Obtener una aplicación que cumpla con las características establecidas por el equipo de desarrollo.
- Comprobar mediante experimentos que el producto software resulta del agrado de los usuarios finales (y sería usado por los mismos).

Y por otra parte están los objetivos propios del autor de este documento que son:

- Creación de una interfaz atractiva, siguiendo el estilo previamente creado en la aplicación móvil.
- Implementar herramientas de administración que optimicen las tareas del equipo administrativo

1.3 Estructura de la memoria

El presente documento expone una idea de negocio y el desarrollo posterior de una aplicación que va a ser el producto software resultante. Los distintos apartados de la memoria van a ser los siguientes:

- Apartado 2: Se realiza el resumen de la aplicación junto a un estudio de la idea de negocio y de mercado para comparar la aplicación con el *software* competidor. Además, se presenta el modelo de negocio, una proyección económica, en análisis de la matriz DAFO y una serie de conclusiones.
- Apartado 3: Se desarrolla la idea de negocio con los dos productos *MVP (Minimum Viable Product)* y sus respectivos experimentos.
- Apartado 4: Explica los distintos aspectos técnicos de la aplicación de una manera general para posteriormente enfocarse en la página web del proyecto
- Apartado 5: Expone la cronología seguida durante los meses empleados para la realización del trabajo.
- Apartado 6: Aparecen las conclusiones del trabajo realizado en base a los objetivos propuestos, así como las limitaciones del mismo y el posible trabajo futuro que se podrá realizar con la aplicación.

El documento finaliza con las referencias bibliográficas consultadas durante el trabajo y un apartado de anexos sobre el funcionamiento general de la aplicación, las preguntas de las encuestas y los experimentos realizados.

2. Evaluación de la idea de negocio

2.1 Resumen de Aedo

Antes de explicar el contenido de la aplicación, es importante explicar la terminología utilizada en el ámbito del proyecto. Empezando por el propio nombre, Aedo proviene del vocablo griego que hacía referencia a los personajes conocidos en la cultura occidental como juglares, y se ha utilizado como un símil a esa tradicional vía oral de transmisión de información, de noticias, tradiciones etc. En este marco histórico, el usuario que utilice la aplicación se convertirá en un “Odiseo”, el cual deberá juntarse con el Aedo para que le transmita ese conocimiento ancestral, y en el momento en el que el Odiseo haya protagonizado esa aventura del saber, podrá convertirse a su vez en un Aedo, al publicar sus propias experiencias.

Aedo es una aplicación tanto móvil como web que permite a cualquier usuario publicar y reservar experiencias turísticas. Estas experiencias pueden ser de cualquier calado, aunque la idea subyacente es que se realicen actividades que pongan en valor las tradiciones, la cultura, la gastronomía, etc. de los lugares en los que se realizan.

El principal atractivo de la aplicación es la posibilidad de realizar actividades ofertadas por personas locales de los lugares que se desean visitar, ayudando a paliar problemáticas como la despoblación, las desigualdades laborales o la pérdida de trabajos artesanales.

2.2 Idea de negocio

La idea de negocio se trata del producto o el servicio que se ofrece al mercado. Dicho producto puede existir en el mercado actual o ser creado. Después se le aporta una propuesta de valor para el mercado al cual se dirige, es por ello que dicha solución puede generar beneficios para el impulsor del producto y otros beneficios para el usuario final de la aplicación.

Para poder elaborar la idea de negocio de una manera más sencilla y visual, se ha utilizado la técnica del *Lean Canvas* (Maurya, 2022) ya que el *Lean Canvas* es una herramienta de visualización de modelos de negocio rápida porque en una única hoja, de un vistazo, se dispone de toda la información importante y necesaria para llevar a cabo la idea de negocio. El tablero *Lean Canvas* se divide en nueve apartados: clientes, problema, proposición de valor, solución, canales, ingresos, costos, métricas y la ventaja competitiva.

Como resultado de la aplicación de esta técnica, se obtiene la **Figura 1** la cual recoge todos los apartados mencionados anteriormente. Del tablero cabría destacar dos puntos que resultan más interesantes. La proposición de valor y la ventaja competitiva ya que tienen un contenido común: las experiencias. La ventaja competitiva de Aedo son las experiencias personales que se transmiten de persona a persona junto a la gamificación y el chat interno mientras que la proposición de valor contempla la realización de experiencias para aprender tradiciones de unos a otros para que estas no se pierdan lo cual fomenta el turismo de zonas despobladas, que es donde suelen estar las tradiciones.

Esta versión del *Lean Canvas* de la **Figura 1** corresponde a la etapa inicial de la idea de negocio de la aplicación y podría sufrir variaciones a lo largo del tiempo ya que la industria del *software* está en constante cambio y es muy posible que una idea que a priori resultaba innovadora después sea implementada por otras aplicaciones competidoras. Entonces la ventaja competitiva planteada inicialmente podría dejar de serlo.

LEAN CANVAS

PROBLEMA	SOLUCIÓN	PROPOSICIÓN DE VALOR	VENTAJA COMPETITIVA	CLIENTES
-Problema de despoblación en España y pérdida de los trabajos populares y la cultura	-Experiencias locales para aprender y no perder la cultura popular	-Eslogan: "Aprende de quien sabe" -Realiza experiencias para aprender tradiciones y nuevas culturas des de un enfoque muy personal.	Experiencias personales (temas de la vida cotidiana), gamificación, chat	-Rango edad: 20 a 60 -Personas con una especialización en un trabajo -Padres/Madres de familia con hijos -Profesores
	MÉTRICAS		CANALES	
	-Número de tours -Número de usuarios -Número de transacciones en los tours		-AppStore/PlayStore -Institutos -Ferias -Empresas locales	
COSTOS		INGRESOS		
Se esperan unos costos para el cuarto año de 157.000€ debido a: -Desarrolladores (junior y senior) -Técnico de soporte -Servidor de base de datos -Compensación teletrabajo -Gastos gestoría -Cuota seguridad social empresa -Campañas de marketing		Se esperan unos ingresos de 137.000€ en el quinto año origen de: -Suscripciones premium de ofertantes -Suscripciones premium de clientes -Packs de personalización -Porcentaje del costo de los tours -Publicidad		

Figura 1: Lean canvas Aedo App (Fuente: Elaboración propia)

2.3 Estudio de mercado

Tras observar las características principales de la aplicación, se realizó un estudio de las posibles alternativas competidoras. Se ha hecho una elección basándose en la similitud del objetivo final de la herramienta, destacando los puntos en común y las diferencias respecto a Aedo.

2.3.1 Civitatis

Esta aplicación está centrada en destinos de viajes a las principales ciudades y destinos de estas. Es una aplicación que tiene más de 500 mil descargas en Google Play; esto la convierte

en uno de los competidores más importantes. Además, también incorpora la opción de buscar tours cerca de la localización actual (ver **Figura 2**).



Figura 2: Web Civitatis (Fuente: civitatis.com/es/)

Además, cuenta con una aplicación específica con guías para varias ciudades europeas para centrarse más en el destino elegido por los usuarios (ver **Figura 3**).



Figura 3: Web Civitatis (Fuente: civitatis.com/es/)

2.3.2 Guruwalk

Guruwalk (ver **Figura 4**) está centrada en *free tours* por las principales ciudades del mundo. Según ellos se encuentran más de **2300 free tours** en español, inglés y otros idiomas. Un *free tour* es un tipo de visita turística en la que el viajero paga lo que quiere al final del recorrido, según su satisfacción y presupuesto.

Establece un sistema de propinas: los/as turistas pueden dar una propina a su guía turístico/a al final del tour si están satisfechos con la experiencia. Además, fomenta el turismo responsable, sostenible y cultural apoyando a los guías locales, promoviendo el contacto directo y contribuyendo a la diversificación y la desestacionalización de la oferta turística, al ofrecer *free tours* en lugares menos masificados y durante todo el año.



Figura 4: Captura de GuruWalk (Fuente: Google play GuruWalk)

2.3.3 Freetour.com

FreeTours.com (ver **Figura 5**) es una plataforma online que ofrece tours gratis o de bajo coste con guías locales en más de 120 países. Tiene un sistema de valoración y comentarios de los usuarios/as que ayuda a elegir el mejor tour y a mejorar la calidad del servicio.

Según SimilarWeb, FreeTour.com tiene un tráfico mensual estimado de 1.3 millones de visitas en los últimos 6 meses, con un promedio de 3.5 páginas vistas por visita y un tiempo de permanencia de 4:28 minutos.

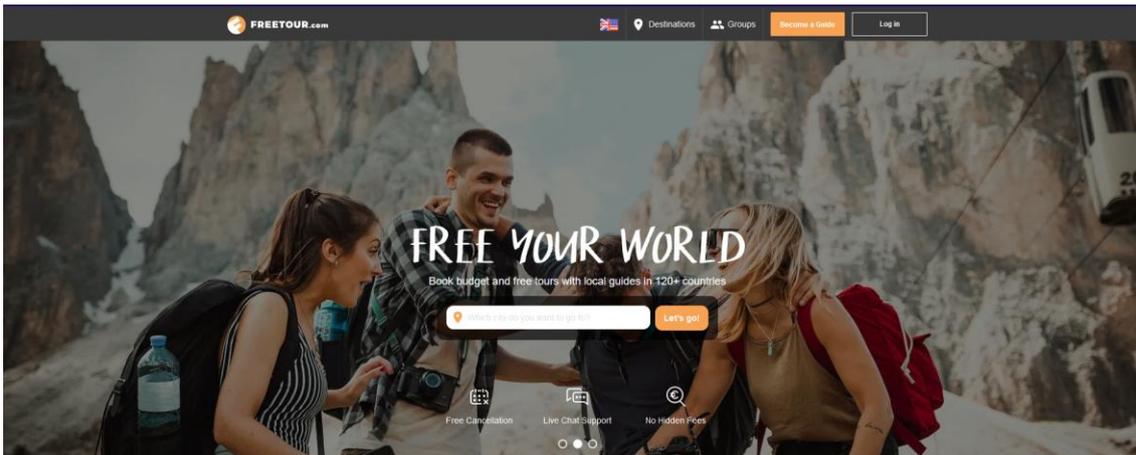


Figura 5: Web FreeTour.com (Fuente: <https://www.freetour.com/>)

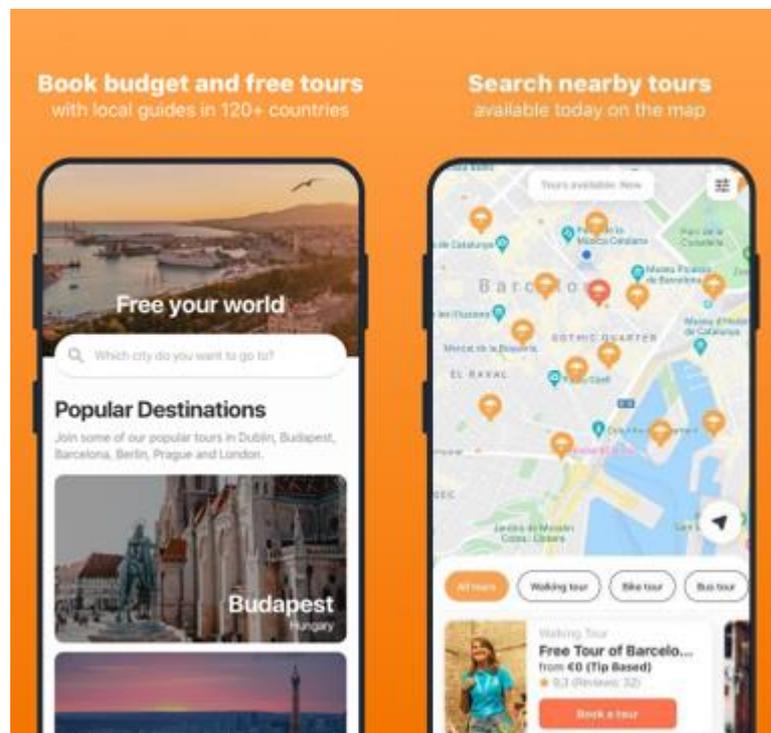


Figura 6: Web de FreeTour.com (Fuente: [freetour.com/es](https://www.freetour.com/es))

Además, también dispone de una versión móvil con más de 100k descargas en Google play (ver **Figura 6**).

2.3.4 Identify

La *app* de turismo cultural Identify (ver **Figura 7**) es una guía digital que te permite descubrir más de 300.000 puntos culturales en todo el mundo. Se puede personalizar la experiencia según los intereses y la geolocalización del usuario/a, así como participar en un juego donde se ganan recompensas y premios por identificar elementos culturales y compartirlos con otros usuarios/as.



Figura 7: Web Identify (Fuente: identifytravel.app)

2.3.5 Withlocals

Withlocals (ver **Figura 8**) es una plataforma que conecta a viajeros con gente local que ofrece comida y experiencias únicas. Puedes disfrutar de una ciudad como un lugareño, conocer su cultura y sus secretos, y apoyar el turismo sostenible. Hay más de 100 destinos disponibles en todo el mundo. Las experiencias son guiadas por expertos locales que conocen la historia, la cultura y las normas sanitarias de cada lugar.



Figura 8: Withlocals (Fuente: withlocals.com/es/)

2.3.6 Tabla comparativa

Tras analizar las distintas aplicaciones competidoras, se ha confeccionado una tabla comparativa (Tabla 1) en la que se comparan una serie de funcionalidades típicas en este tipo de aplicaciones y se tiene un registro de, por cada programa, cual implementa las características analizadas.

Tabla 1: Tabla comparativa Aedo con otras aplicaciones. (Fuente: Elaboración propia)

Característica:	Civitas	Guruwalk	FreeTour.com	Identify	Withlocals	Aedo
Buscador	✓	✓	✓	✓	✓	✓
Usuarios registrados consumidores	✓	✓	✓	✓	✓	✓
Usuarios registrados productores	✓	✓	✓		✓	✓
Filtros	✓	✓		✓	✓	✓
Compras integradas	✓				✓	✓
Soporte técnico	✓	✓	✓	✓	✓	✓
Diferentes idiomas de interfaz	✓			✓	✓	Español, Inglés

Diferentes idiomas del tour		✓	✓		✓	✓
Distintos tipos de tours	Entradas, espectáculos, escapadas gastronómicas, transportes	Alternativo, arte urbano, bici, leyendas..	Tours gratis, diarios, urbanos, bares, bicicleta, gastronómicos	Sólo muestra a puntos de interés	Gastronómicas, bici, arte, música, nocturnas, ...	✓
Sin gastos de reserva	✓	✓	✓		✓	✓
Servicio de la APP gratuito	✓		✓	✓	✓	✓
Ver fotos, reseñas, detalles, descripciones y disponibilidad del tour (quien oferta el tour)	✓	✓	✓	✓	✓	✓
Calificar y comentar el tour/guía	✓	✓	✓		✓	✓
Wishlist de tours	✓		✓			✓
Encontrar tours por cercanía (a la localización actual)	✓		✓	✓		✓
Oferta de entradas a espectáculos	✓					
Recursos para estudiantes y centros educativos				✓		✓
Rutas personalizadas				✓		✓
Verificación de cuenta	✓	✓	✓	✓	✓	✓
Valoración del ofertante sobre los clientes						✓
Chat Bot						✓
Gamificación						✓

Así pues, se puede observar en la Tabla 1 que hay una serie de características comunes a todas las aplicaciones como puedan ser buscador, registro de usuarios, soporte técnico, ver fotos y reseñas y la verificación de cuenta. Después existen algunas características que están en prácticamente todas las aplicaciones como puedan ser los filtros, las compras integradas, uso gratuito de la app, publicar comentarios... Y finalmente están las características propias de la aplicación Aedo, las cuales no están contempladas en la competencia en la actualidad y son: la gamificación, un chat interno, un chat bot para soporte técnico y la valoración por parte de la persona que explica la experiencia sobre los clientes que asisten. Además, Aedo ofrece recursos para estudiantes y centros educativos y rutas personalizadas, característica solo ofrecida por Identify.

2.4 Modelo de negocio

Existen distintos modelos de negocio que se podrían aplicar en Aedo, pero, al tratarse de una aplicación móvil, algunos de ellos no serían bien recibidos. Es por esa razón que, de antemano, el modelo de negocio tradicional, basado en la venta de licencias en la que se adquiere el producto por un precio y se dispone de total libertad para utilizarlo, no encaja con la aplicación. La razón por la que se descarta es que Aedo necesita de una amplia base de datos de usuarios para poder ofertar experiencias por parte de las personas que quieren ofrecerlas y de clientes que asistan a dichas experiencias. Si los/as usuarios/as no utilizan la aplicación, no hay tránsito y por tanto la aplicación no aumentará el número de usuarios activos. Muchos de estos usuarios rehusarían de la aplicación si tienen que pagar por ella, además que probablemente otra empresa podría vislumbrar el potencial de la aplicación y la implementaría con otro modelo de negocio.

Es por esta razón que la opción que se considera acertada para el proyecto es un modelo *freemium*, en el que los usuarios pueden descargar el *software* de manera gratuita y utilizarlo para, posteriormente, adquirir los servicios *premium* si así lo desean. Entre las distintas posibilidades de opciones de compra se han pensado las siguientes:

- **Suscripción *premium* para los ofertantes:** Se trata de un servicio para los/as usuario/as que ofertan experiencias. Mediante este servicio se podrían posicionar las experiencias en el buscador de otra manera. A su vez, permitiría añadir más imágenes a la hora de crear las experiencias para que los usuarios puedan tener más información sobre lo que se van a encontrar. Además, desaparecen los anuncios publicitarios.
- **Suscripción *premium* para los clientes:** Un cliente *premium* no tiene anuncios publicitarios en su cliente móvil y podría tener prioridad a la hora de acceder a una experiencia para un día dado si dicha experiencia tiene el cupo de usuarios completo. Un usuario *premium* se beneficia de prioridad en la cola. También se anularía la publicidad para este tipo de usuarios.
- **Packs de personalización:** Es común en las aplicaciones dar la posibilidad a los/as usuarios/as de adquirir elementos para su perfil de usuario, que son visibles desde la aplicación y los distinguen de otros usuarios que no han adquirido la membresía *premium* como pudiera ser un icono o alguna insignia.

Por otra parte, también se consideraría como fuentes externas de ingresos la implementación de anuncios publicitarios en la aplicación y una pequeña comisión del precio de las experiencias ofertadas en la plataforma cada vez que un cliente se apunta a las mismas.

2.5 Proyección económica a 6 años

Para realizar la estimación de ingresos y gastos durante los seis primeros años, se ha dividido este período de tiempo en doce semestres. Se parte de una pequeña cantidad de usuarios para el primer semestre estimada en 500 hasta llegar a 60.000 usuarios activos en el semestre doceavo.

La **Figura 9**, muestra la proyección económica preestablecida para el proyecto en cuestión, mostrando los diferentes gastos desglosados y los previstos ingresos según la fuente proveniente.

Por una parte, los **ingresos** provienen, como se ha explicado en el apartado de **modelo de negocio** de este documento, de distintas fuentes a ser las suscripciones *premium* para ofertantes y clientes, los packs de personalización, las comisiones de las experiencias y la publicidad. Cabe destacar que no se prevén ingresos publicitarios hasta el semestre décimo momento en el cual ya se dispone de una base de usuarios activos suficiente como para ser rentable que los anunciantes publiquen sus productos. Es por esta razón que resulta tan complicado adquirir ese punto de equilibrio en el que se empiezan a generar beneficios.

Por otra parte, los **gastos** se deben a distintos servicios de *hosting* necesarios para el funcionamiento de la aplicación, a la adquisición de los dispositivos necesarios para realizar el desarrollo del *software* y los salarios de los trabajadores, además de campañas de *marketing*.

Analizando la **Figura 9** se puede apreciar que el primer año los gastos van a ser constantes durante los dos semestres y los ingresos van a ser prácticamente nulos debido a la carencia de usuarios, es por ello por lo que se optará por una campaña de publicidad más agresiva de cara al segundo año.

En el segundo año el número de usuarios se incrementa y con ello los ingresos. Los gastos también aumentarán debido a los costes de gestión y a la campaña publicitaria. Dicha campaña publicitaria se refleja en el tercer año en el que se multiplica por cinco la cantidad de usuarios respecto a inicios del segundo año. Esto permite aumentar aún más los ingresos, pero a su vez se requerirá de la contratación de un desarrollador senior.

A partir de este momento, cuarto año, se prevé que junto a campañas publicitarias y el boca a boca de los usuarios el número de usuarios activos seguirá aumentando y en este momento la empresa empezará a generar beneficios semestrales, pero no será hasta el quinto año donde se equilibren los gastos y beneficios.

Usuarios activos	500	1000	2000	6000	10000	15000	20000	30000	40000	50000	55000	60000
Número de Semestre	1	2	3	4	5	6	7	8	9	10	11	12
Tiempo de suscripciones	Año 1 / S1	Año 1 / S2	Año 2 / S1	Año 2 / S2	Año 3 / S1	Año 3 / S2	Año 4 / S1	Año 4 / S2	Año 5 / S1	Año 5 / S2	Año 6 / S1	Año 6 / S2
Suscripción premium ofertantes	1500	3000	6000	18000	30000	45000	60000	90000	120000	150000	165000	180000
suscripción premium para clientes	200	400	800	2400	4000	6000	8000	12000	16000	20000	22000	24000
Packs de personalización	50	100	200	600	1000	1500	2000	3000	4000	5000	5500	6000
porcentaje odiseas	90	180	360	1080	1800	2700	3600	5400	7200	9000	9900	10800
publicidad										33600	36960	40320
Total Ingresos Semestrales	1840	3680	7360	22080	36800	55200	73600	110400	147200	217600	239360	261120
Gastos Anuales	Año 1 / S1	Año 1 / S2	Año 2 / S1	Año 2 / S2	Año 3 / S1	Año 3 / S2	Año 4 / S1	Año 4 / S2	Año 5 / S1	Año 5 / S2	Año 6 / S1	Año 6 / S2
Servidor BBDD	100 €	200 €	400 €	1.200 €	2.000 €	3.000 €	4.000 €	6.000 €	8.000 €	10.000 €	11.000 €	12.000 €
mobiles IOS / Android	800 €	0 €	0 €	0 €	800 €	0 €						
Compensación teletrabajo	3.600 €	3.600 €	3.600 €	3.600 €	7.200 €	7.200 €	7.200 €	7.200 €	7.200 €	7.200 €	7.200 €	7.200 €
Gestoría	0 €	0 €	1.000 €	1.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €
Desarrollador Senior Android / IOS	0 €	0 €	0 €	0 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €	20.000 €
Desarrollador junior	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €
Desarrollador junior	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €	12.500 €
Tecnico de soporte	0 €	0 €	0 €	0 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €
Cuota seguridad social empresa	300 €	300 €	1.200 €	1.200 €	1.200 €	1.200 €	1.200 €	1.200 €	1.200 €	1.200 €	1.200 €	1.200 €
Campaña marketing	2000	2000	4000	6000	6000	5000	5000	5000	5000	5000	5000	5000
Total Gastos	31.800 €	31.100 €	35.200 €	38.000 €	74.200 €	73400	74400	76400	78400	80400	81400	82400
Resultado Semestral	-29.960 €	-27.420 €	-27.840 €	-15.920 €	-37.400 €	-18.200 €	-800 €	34.000 €	68.800 €	137.200 €	157.960 €	178.720 €
Resultado Semestral Acumulado	-29.960 €	-57.380 €	-85.220 €	-101.140 €	-138.540 €	-156.740 €	-157.540 €	-123.540 €	-54.740 €	82.460 €	240.420 €	419.140 €

Figura 9: Proyección económica (Fuente: Elaboración propia.)

Como se puede apreciar en la **Figura 9**, la previsión establece un *bypass* semestral entre ingresos y gastos a partir del segundo semestre del cuarto año. Esta situación se generalizaría en el acumulado a partir del segundo semestre del quinto año, obteniendo unos beneficios aproximados en dicho plazo de unos 82.000€. En la tabla también aparecen desglosados los diferentes gastos e ingresos proyectados para dicho proyecto y que se han considerado estándares para un trabajo de esta complejidad.

La **Figura 10**, muestra el balance de pérdidas/beneficios acumulados por semestre que se han explicado en el documento. En ella se pueden observar las pérdidas de los primeros semestres hasta llegar al punto de equilibrio en el semestre diez (segundo semestre del quinto año), donde la aplicación empezará a generar beneficios.

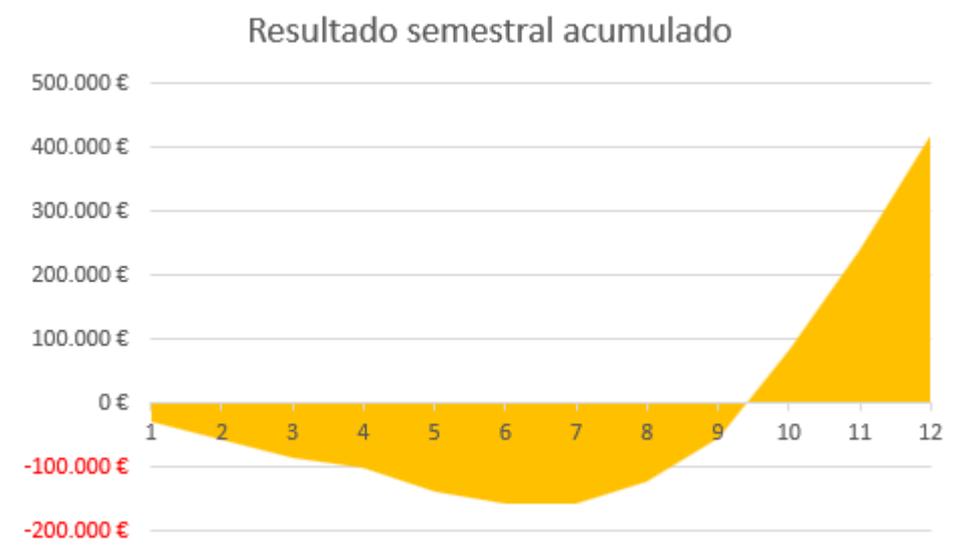


Figura 10: Resultado semestral acumulado (Fuente: elaboración propia)



2.6 Análisis DAFO

La **Tabla 2**, conocida como matriz DAFO, establece las Debilidades, las Amenazas, Fortalezas y Oportunidades que se han observado para el proyecto. En este caso se debe destacar la poca experiencia del equipo, equilibrada por el potencial del proyecto para diferentes sectores de mecenazgo de la sociedad (instituciones, organizaciones etc.) Además, se ha considerado positivo el hecho de contar con un equipo reducido para potenciar la facilidad en la toma de decisiones.

Tabla 2: Matriz DAFO (Fuente: Elaboración propia.)

Debilidades	Amenazas
<ul style="list-style-type: none"> • Equipo junior con poca experiencia. • Poca financiación en el proyecto inicial. • Equipo reducido. • Dirección estratégica inicial poco establecida. 	<ul style="list-style-type: none"> • Entrada de empresas competidoras con mayor financiación. • Cambios demográficos intensos. • Crecimiento lento del mercado. • Futuros cambios en la política económica de la empresa. • Políticas adversas en otros países.
Fortalezas	Oportunidades
<ul style="list-style-type: none"> • Nicho con poca competencia. • Funcionalidad novedosa. • Capacidad de decisión limitada a un equipo pequeño. • Políticas afines. • Basado en ODS con posible financiación europea. • Flexibilidad organizativa. • Equipo joven con ideas modernas y atractivas. 	<ul style="list-style-type: none"> • Afinidad con diferentes sectores de la población. • Afinidades políticas con instituciones. • Aprovechamiento del potencial turístico del país. • Políticas territoriales alineadas con la visión de la empresa. • Financiación a nivel estatal y europea. • Crecimiento rápido de la popularidad de la aplicación.

2.7 Conclusiones de la evaluación

A grandes rasgos, el presente documento ha pretendido mostrar la factibilidad de un proyecto con un gran nicho de mercado, con gran potencial y elementos distintivos, que, a pesar de requerir de un presupuesto inicial elevado, debido a la complejidad del mismo, el riesgo que ello supone es limitado en relación con las posibles oportunidades observadas.

El hecho de aunar en una aplicación diferentes características tan atractivas para diferentes sectores de la sociedad, como actividades de ocio, tiempo libre, medio ambiente, puesta en valor de tradiciones, cultura etc., se ha visto como un potente aliciente por parte del equipo para confiar en la buena ventura del proyecto y el compromiso con el mismo.

3. Desarrollo de la idea de negocio

A continuación, se muestra el mapa de características de la aplicación.

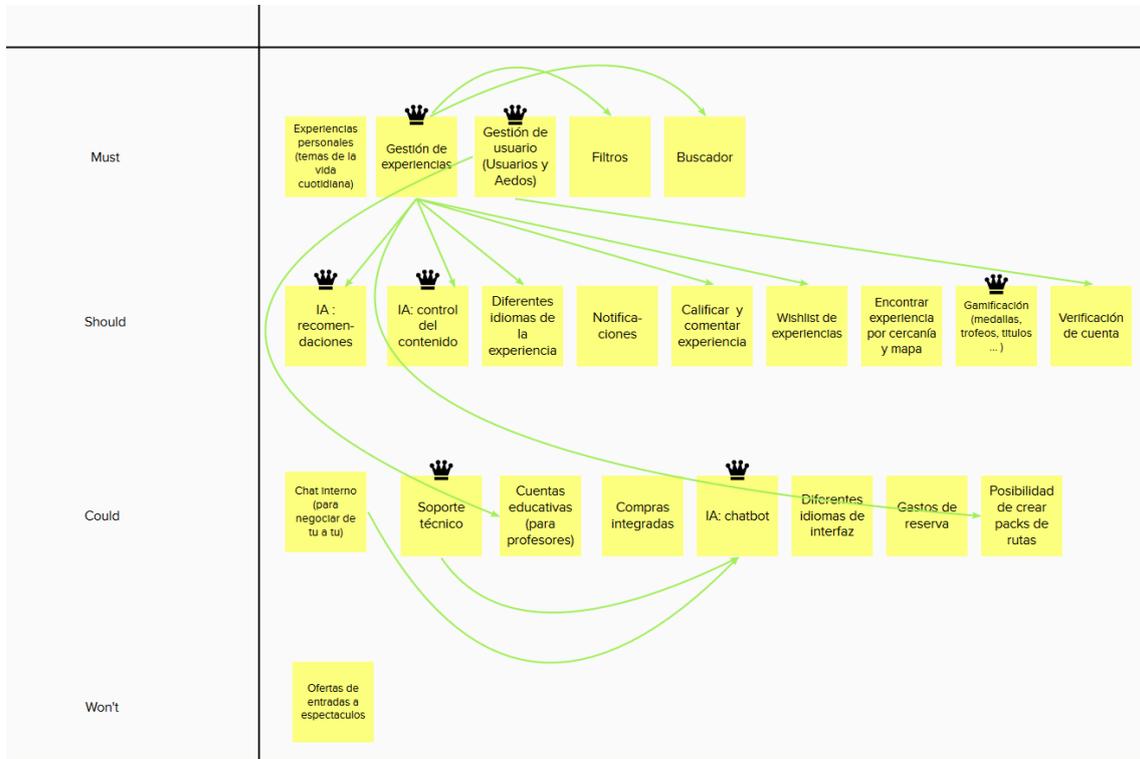


Figura 11: Mapa de características inicial (Fuente: Elaboración propia)

En la **Figura 11** podemos observar los diferentes elementos que se han considerado importantes a implementar en la aplicación durante su desarrollo hasta consolidarse como un producto maduro. También se ha realizado un análisis MOSCOW (Kuhn, 2009) para priorizar las diferentes características, basándose en criterios de funcionalidad y usabilidad.

Destacar funciones básicas como la gestión de experiencias o usuarios, que establecen la base de la aplicación, u otras más avanzadas como la IA de recomendaciones o de control de contenido, que darían un toque distintivo a la misma, mejorando la experiencia de uso de los usuarios.

3.1 Desarrollo del primer MVP

El desarrollo de la aplicación Aedo empezó en el marco de la asignatura de Proyecto de Ingeniería del Software, con una metodología preestablecida por el profesorado de esta. En ese ámbito se utilizó la herramienta Worki para gestionar el esfuerzo dedicado al proyecto y el trabajo individual de cada miembro. Se realizaron 3 Sprints de un mes aproximadamente, en el que se implementaron varias funcionalidades básicas del proyecto para posteriormente presentarlo en la



Feria de Proyectos de la ETSINF. La elección de dichas funcionalidades se basó en un criterio MVP según las características que se consideraron básicas de la aplicación según la **Figura 11**.

El diseño de la aplicación fue desarrollado por el grupo de estudiantes de Bellas Artes integrados en el grupo de desarrolladores de software, técnica novedosa llevada a cabo en dicha asignatura para incrementar la cooperación entre estudiantes de diferentes facultades y crear sinergias de trabajo similares a una experiencia laboral real.

3.1.1 Experimento 1: Feria de proyectos

En la Feria de Proyectos ¹se presentó la aplicación, y se estableció contacto con empresas y otros desarrolladores que la analizaron y opinaron sobre sus funcionalidades. De esta experiencia se extrajo el aprendizaje de la importancia de la viralización en una aplicación de este tipo y de las diferentes posibilidades de realizar experiencias en grupo, por personas desconocidas entre sí, y que decidieran conocerse realizando dicha actividad (ver **Figura 12**).



Figura 12: Fotografía del stand y el equipo de Aedo (Fuente: [Fotografía de la Etsinf](#))

3.2 Desarrollo del segundo MVP

El segundo *MVP* estuvo enfocado a aumentar las funcionalidades de la aplicación, elaborar una inteligencia artificial para las recomendaciones de experiencias, elaborar la página web de la aplicación para las gestiones administrativas y tener una versión básica de la aplicación en una

¹ Feria de Proyectos: Evento promovido anualmente por la UPV donde los estudiantes presentan proyectos propios a profesores, compañeros y profesionales de la industria.

versión de navegador. De entre las funcionalidades implementadas, cabría destacar el apartado educativo de la aplicación. Tras el segundo *MVP*, la aplicación permitía disponer de cuentas educativas y de generar experiencias colectivas para clases de alumnos enteras. También se incluyeron mejoras de interfaz y funcionalidades para conocer, de una manera más clara, cuánta gente estaba apuntada a una experiencia, mejoras en el perfil, categorías extra, etc.

Para el segundo y tercer experimento, tras tener la aplicación con las últimas funcionalidades implementadas, se presentó la aplicación a dos perfiles diferentes, pero a su vez enfocados en la creación de experiencias.

3.2.1 Experimento 2: Presentar la aplicación a un trabajador del campo.

La aplicación se enseñó a un agricultor y quedó bastante encantado con las funcionalidades de la aplicación acorde a las puntuaciones facilitadas en una encuesta de usabilidad que se realizó para medir el experimento. Sí es cierto que nos propuso, como sugerencia, una mejora que se tenía pensado de ser implementada, pero, por falta de tiempo, no pudo ser incluida en el segundo producto viable mínimo: es la adición de un chat persona a persona entre el usuario de la aplicación y el impartidor de la experiencia.

3.2.2 Experimento 3: Presentar la aplicación a un profesor

Ya que uno de los objetivos perseguidos durante la segunda fase del desarrollo de la aplicación fue la implementación de un tipo de cuentas educativas, se consideró oportuno contactar con un profesor de secundaria para conocer su opinión al respecto. La elección de dicha persona vino condicionada por ser excompañero de clase de unos de los miembros del equipo y ser una persona joven con facilidad en el uso de tecnologías.

Se le explicó el proyecto y se le presentaron las funcionalidades de este, destacando el uso de las cuentas educativas y las opciones para reservar actividades para grupos escolares. Ya que el proyecto pretende poner en valor las tradiciones y la cultura, el encuestado recalcó la importancia de este tipo de actividades dentro del ámbito educativo de los centros de secundaria. También sugirió la posibilidad de exportar las descripciones de las actividades a pdf para poder compartirlas con los padres de los alumnos, y firmar los permisos para que los alumnos las realicen. El equipo de desarrollo valoró dicha opción, y se tiene pensado implementar un sistema de exportación a diferentes formatos y de compartir las actividades a través de aplicaciones de terceros en futuras versiones.

Se le facilitó al docente un cuestionario *online* (Ver Anexo 2) para conocer su opinión y registrar sus posibles sugerencias. Las imágenes del experimento pueden verse en el Anexo 3.

4. Aspectos técnicos

Se debe hacer una consideración inicial antes de empezar con la parte específica de la página web del proyecto. Como ya se comentó anteriormente, este proyecto nació en la asignatura de proyecto de ingeniería del software (PIN). En este proyecto en el que se colaboró mano a mano con la escuela de bellas artes.

Fue decidido por el equipo de programadores utilizar una terminología basada en la antigua Grecia para la aplicación de Aedo, es por ello que los usuarios mencionados hasta este momento en el documento, en este documento se les conoce como “odiseos” y las experiencias en el código aparecen comentadas como “odiseas”. Por tanto, siempre que aparezca la palabra “odiseo” en este presente documento, se refiere a los usuarios de la aplicación y cuando aparece “odisea” se refiere a las experiencias a la que los usuarios se inscriben.

Esta discrepancia de nombres tiene un motivo. A la hora de presentar el proyecto en la Feria de Proyectos se mostró la aplicación al grupo de diseño con unas semanas de antelación y se propuso cambiar el nombre de “odisea” por algo más amigable ya que la palabra se relaciona con algo muy complicado o imposible de hacer. Al realizar este cambio no tenía sentido mantener “odiseo” para los usuarios, sin embargo, estos cambios solo se realizaron en las interfaces del usuario e internamente siguen utilizando esta terminología.

4.1 Herramientas utilizadas

Para la realización de la página web de este proyecto se ha optado por la utilización de Angular (*Angular Documentation*, s.f.). Angular es un *framework*² de desarrollo web de código abierto que permite construir aplicaciones de web dinámicas y de una sola página. Angular utiliza *TypeScript* (*TypeScript Documentation*, s.f) como lenguaje de programación y se basa en los conceptos de componentes, directivas y enlaces de datos para mejorar y facilitar el desarrollo y arquitectura de sus aplicaciones.

Para hacer el hosting de la página web, y como base de datos, se ha optado por utilizar los servicios de *Firebase*³ debido a su versatilidad y facilidad de uso. *Firebase* es una plataforma de desarrollo creada por Google que nos ofrece servicios de autenticación, almacenamiento de archivos, análisis y una base de datos escalable.

En cuanto al control de versiones, se ha utilizado *GIT*, en el cual existe una rama *master* donde se encuentra la última versión de la aplicación y la cual es utilizada por *Firebase* para hacer el hosting; existen ramas para cada requisito que se realice.

² Framework: <https://www.edix.com/es/instituto/framework/>

³ Firebase: <https://firebase.google.com>

Para mejorar la calidad y legibilidad del código se ha utilizado en este proyecto un *linter*, en concreto SonarLint⁴. Un *linter* es una herramienta que realiza un análisis estático del código con el fin de detectar errores, vulnerabilidades y malas prácticas para ayudar a los programadores a mantener un código limpio y legible.

Para la documentación se ha utilizado *Compodoc* (ver Sección 4.5 para su descripción completa) para la generación automatizada de la documentación y poder analizar la cobertura de la documentación total del proyecto.

Para la gestión del proyecto y la coordinación del equipo se han utilizado dos herramientas distintas. Para los cuatro primeros *sprints* se utilizó *Worki*⁵, y para los siguientes se ha usado *Jira*⁶. De esta forma se han experimentado diferentes formas de trabajar y organizar los proyectos.

4.2 Entornos de desarrollo

Como entorno de desarrollo para el proyecto se ha utilizado *Visual Studio Code*⁷ de *Microsoft*, debido su interfaz y su soporte para extensiones y complementos que facilitan la colaboración entre pares y la escritura de código.

Los complementos utilizados en este proyecto han sido:

Angular Language Service (*Angular*, 2017): Proporciona herramientas para el desarrollo de proyectos en Angular, ofreciendo sugerencias, autocompletado y verificación de tipos.

Alphabetical sorter (*Ugur Erdal*, 2018): Permite ordenar importaciones y variables por orden alfabético lo que mejórala la legibilidad del código.

Color highlight (*Sergii N*, 2016): Esta extensión permite visualizar los colores aplicados a los estilos en el mismo código lo que facilitó la creación del tema de la aplicación.

Sonarlint (*SonarSource*, 2017): Realiza un análisis estático del código en tiempo real, proporciona sugerencias de cómo mejorar la calidad y seguir buenas prácticas de desarrollo.

Error lens (*Alexander*, 2019): Los errores y advertencias del código aparecen directamente en el código fuente, lo que facilita la corrección de los errores.

Firebase Explorer (*jsayol*, 2018): Permite explorar y administrar los proyectos de *Firebase* desde el mismo *Visual Studio Code*

Prettier (*Prettier*, 2017): *Prettier* (*What is Prettier?*, s.f.) es formateador de código ampliamente utilizado. Se encarga de eliminar cualquier estilo original y se encarga de que todo el código siga un estilo consistente.

⁴ SonarLint: <https://marketplace.visualstudio.com/items?itemName=SonarSource.sonarlint-vscode>

⁵ Worki: <https://cliente.tuneupprocess.com/web>

⁶ Jira: <https://www.atlassian.com/es/software/jira>

⁷ Visual Studio Code: <https://code.visualstudio.com>

GitLens (*GitKraken*, 2016): Este complemento entre sus muchas funcionalidades permite trabajar con Git directamente desde el editor permitiendo hacer operaciones *push* y *pull* desde *Visual Studio Code*

Todo Tree (*Gruntfuggly*, 2017): Facilita realizar el seguimiento de las tareas pendientes dentro del código.

4.3 Modelo de datos

La **Figura 13** representa el modelo de datos del proyecto, protagonizado por las Experiencias como pilar fundamental de la aplicación, con una serie de relaciones con las demás clases de este. Esta estructura está orientada a la optimización en la utilización de la base de datos no relacional *Firestore*⁸, y a las necesidades mismas del correcto funcionamiento de la aplicación. Como se puede observar, todos los elementos propios de una aplicación de este tipo estarían representados, destacando las reservas, los comentarios, los usuarios, los diferentes idiomas disponibles etc.

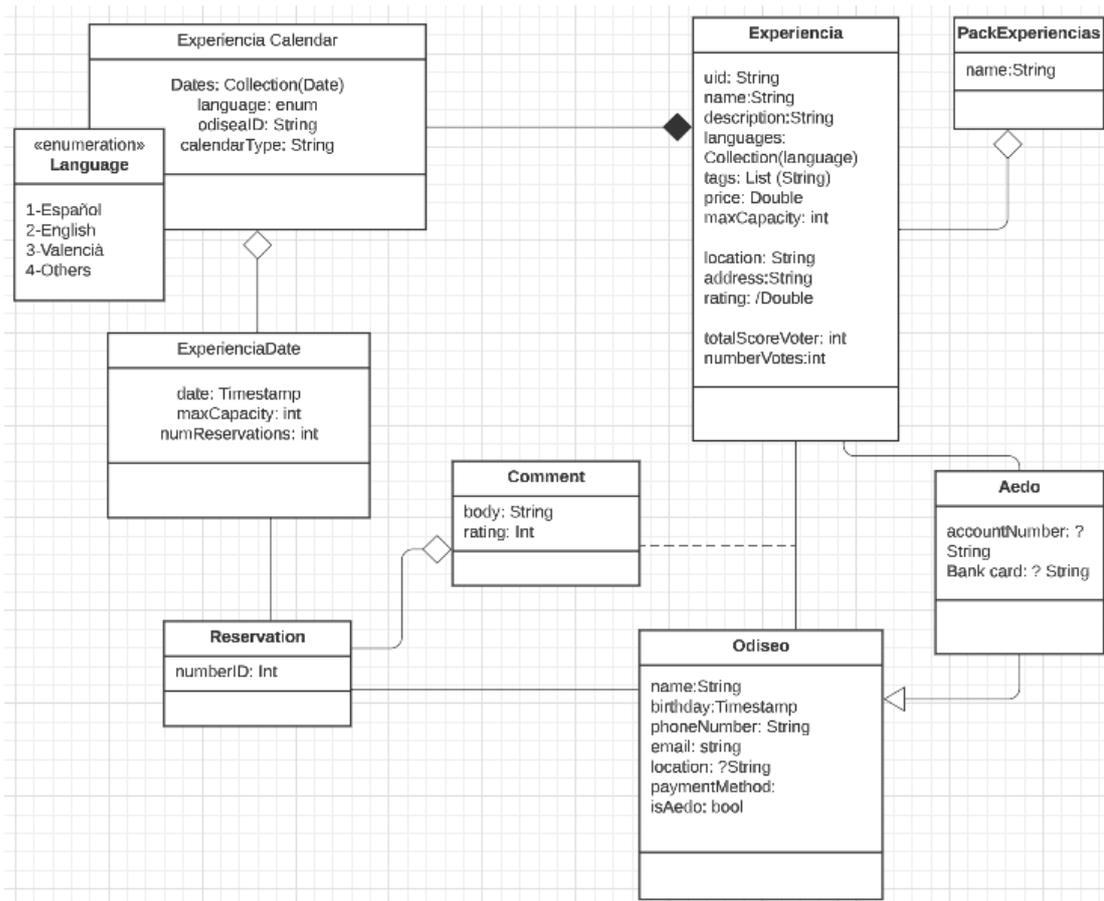


Figura 13: Modelo de datos (Fuente: Elaboración propia)

⁸ Firestore: <https://firebase.google.com/docs/firestore>

4.4 Estructura del proyecto

En este proyecto de *Angular* hemos adoptado una estructura de carpetas organizada y modular para garantizar el código limpio y mantenible del proyecto.

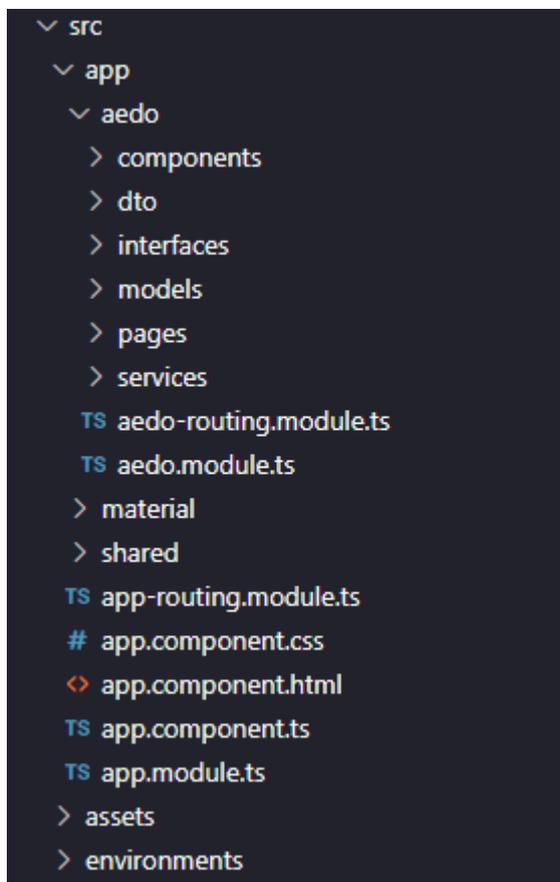


Figura 14: Estructura de carpetas

Como se puede ver en la **Figura 14** se ha seguido una estructura de proyecto basada en componentes en donde se pueden encontrar las siguientes carpetas con sus funciones.

- La carpeta **src**: Es la raíz del proyecto y contiene todos los archivos de código fuente
- La carpeta **app**: Contiene el módulo principal del proyecto y alberga toda la lógica del proyecto
- La carpeta **aedo**: Esta carpeta contiene el módulo que carga el módulo principal y cual tiene las funcionalidades de la aplicación.
- La carpeta **components**: Esta carpeta contiene los componentes reutilizables que se utilizan en el proyecto.
- La carpeta **dto**: Aquí se almacenan los objetos de transferencia de datos utilizados para la comunicación entre el *front-end* y el *back-end*
- La carpeta **interfaces**: Contiene las interfaces utilizadas en el proyecto para definir estructuras de datos
- La carpeta **models**: Se utiliza para definir los modelos de datos utilizados en la aplicación
- La carpeta **pages**: Alberga las diferentes páginas de la aplicación.

- La carpeta **services**: Contiene los servicios utilizados para la lógica de negocio y la comunicación con el *back-end*
- La carpeta **material**: Alberga todos los archivos relacionados con la biblioteca de diseño Angular Material
- La carpeta **shared**: Se dedica a los archivos compartidos entre los diferentes componentes y servicios.
- La carpeta de **assets**: Contiene los colores, paletas, tipografías y temas utilizados en la aplicación para su uso en los distintos módulos de esta.

4.5 Documentación

Para mejorar la comprensión del código y facilitar la colaboración entre los distintos miembros del equipo se ha documentado los distintos servicios, páginas, objetos y componentes que se usan en la aplicación usando *Compodoc*. *Compodoc* es una herramienta de documentación automática de código para proyectos de Angular, nos devuelve información detallada de los componentes que forman nuestro proyecto, sus dependencias y las rutas que existen en él.

Entre en todas las etiquetas que permite utilizar *Compodoc* las utilizadas a la hora de documentar este proyecto han sido:

- **@component**: Documenta un componente de Angular
- **@service**: Documenta un servicio de Angular
- **@input**: Documenta una propiedad de entrada de un componente
- **@output**: Documenta un evento emitido por un componente
- **@param**: Documenta un parámetro de un método o función
- **@returns**: Documenta el valor de retorno de un método o función.

```
/**
 * Service for managing comments.
 */
export class CommentService {
  collection: string = 'comments';

  /**
   * Constructs an instance of CommentService.
   * @param firestoreService - Instance of FirestoreService.
   */
  constructor(private firestoreService: FirestoreService) {}

  /**
   * Creates a new comment.
   * @param comment - Object representing the comment to create.
   * @returns A promise that resolves when the comment is created successfully.
   */
  create(comment: IComment) {
    return this.firestoreService.create(this.collection, comment.toJSON());
  }
}
```

Figura 15: Servicio documentado (Fuente: Elaboración propia)

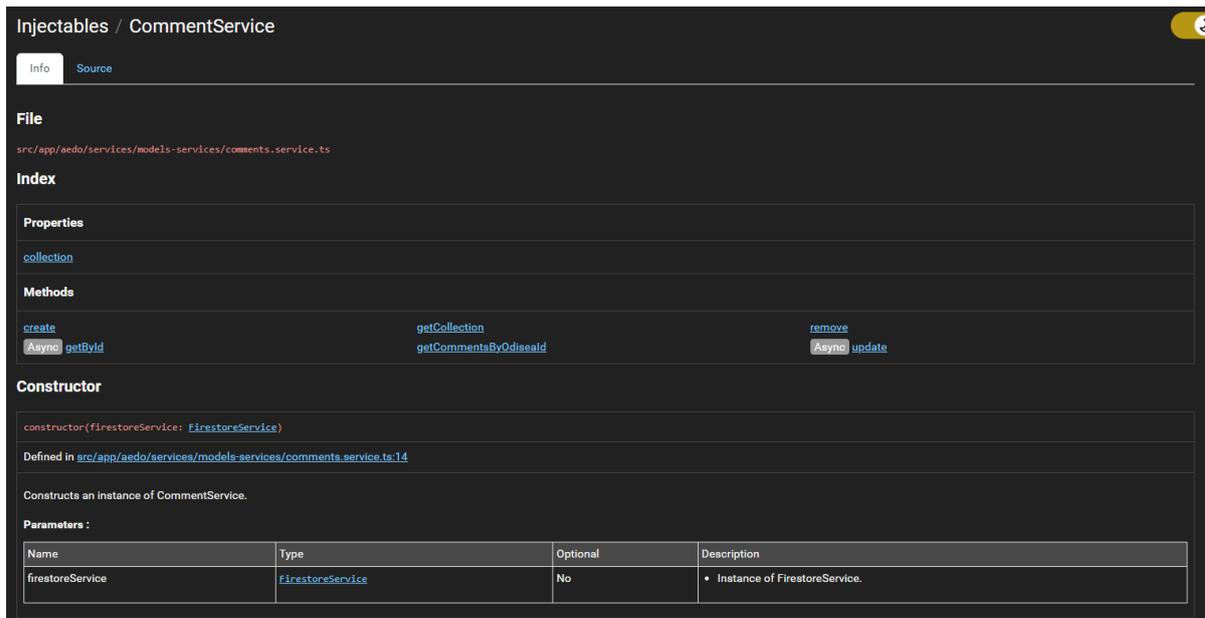


Figura 16: Documentación de servicio en Compodoc (Fuente: Elaboración propia)

En la **Figura 15** y la **Figura 16** se muestra en cómo se implementa la documentación dentro del código y el resultado de este después dentro de *Compodoc*.

4.6 Estilo artístico

El estilo de la aplicación se eligió teniendo en cuenta el objetivo y contexto de esta que era dar visibilidad a los pueblos y cultura de España que de otra forma pasa desapercibida. Para ello, el equipo de diseño creó ilustraciones haciendo referencia a ciertos puntos de la cultura española. Entre aquellas ilustraciones se encontró el icono de la web y la aplicación que fue diseñado teniendo como inspiración al personaje literario Don Quijote (ver **Figura 17**).



Figura 17: Logotipo de la aplicación (Fuente: Elaboración propia)

Para los tonos de la aplicación se han elegido colores de tonos verdosos y blancos para la paleta principal y para la paleta secundaria utilizada para acentuar ciertos aspectos de la aplicación se ha elegido colores rojizos (ver **Figura 18**). Por último, en cuanto a las paletas se decidió usar la paleta por defecto de Google para resaltar los avisos como pueden ser las acciones de borrado en la aplicación ya que estos colores permiten reconocer rápidamente cual es la acción del componente que usan esta paleta.



Figura 18: Paleta de colores de la aplicación (Fuente: Elaboración propia)

En cuanto a la tipografía de la aplicación, el equipo de diseño también seleccionó estilos tipográficos que impactan positivamente en la apariencia general de la aplicación, teniendo en cuenta la paleta de colores elegida. Se eligieron dos tipografías diferentes según si el texto es

primario o secundario, y se crearon estilos específicos de acuerdo con la acción y la importancia del texto a mostrar. En la **Figura 19** se muestran que tipografías se eligieron para el proyecto.



Figura 19: Tipografía de la aplicación (Fuente: Elaboración propia)

Aprovechando que se utiliza *Angular* con toda esta información se creó un tema personalizado que incorpora las paletas de colores y la tipografía previamente mencionadas. Esto se hizo para mejorar la coherencia visual de toda la aplicación y mejorar la experiencia del usuario al hacerla más consistente.

Este estilo se aplicó a todos los componentes de *Angular Material*, además de aquellos componentes que tuvieron que ser importados y aquellos que fueron creados para este proyecto.

Además, el uso del tema personalizado permite la fácil adaptación y modificación de los estilos lo que facilitará en el futuro cambios en la apariencia de la aplicación y añadir nuevas funcionalidades dependientes del estilo como puede ser el modo oscuro de la aplicación.

En resumen, el desarrollo del tema personalizado, aunque consumió tiempo del equipo contribuyó a mejorar la coherencia visual y la experiencia del usuario en la aplicación, al tiempo que proporciona flexibilidad para futuras actualizaciones y personalizaciones. Por lo tanto, esta

inversión de tiempo se ha recuperado con creces con la facilidad con la que ahora se pueden añadir y modificar los estilos.

4.7 Características exclusivas de la aplicación web

La idea de implementar la página web del proyecto venía de la mano de la necesidad de disponer de herramientas con las cuales poder gestionar ciertas estructuras de datos que un usuario corriente no puede editar dentro de la aplicación. Aunque es cierto que dentro de los servicios de *Firebase* estos se pueden editar y cumplen las funcionalidades que el equipo de desarrollo necesitaba, no significaba que fuera de fácil uso (ver **Figura 20**).

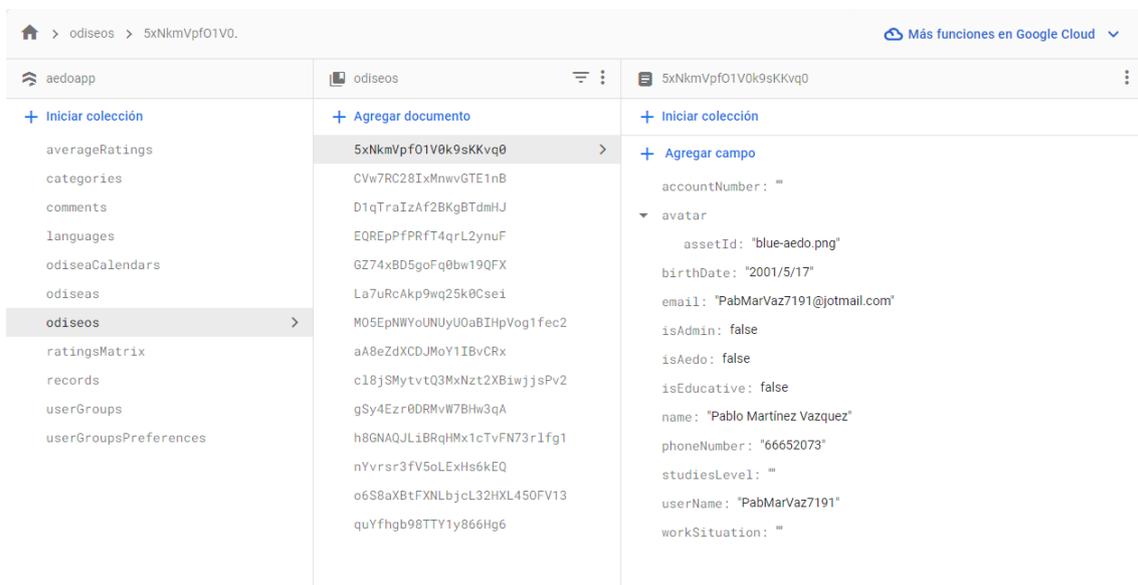


Figura 20: Interfaz de Firestore (Fuente: <https://firebase.google.com>)

Para acabar con este problema, la página web dispone de un *dashboard* de administración, al cual solo pueden acceder los usuarios que disponen del rol de administrador.

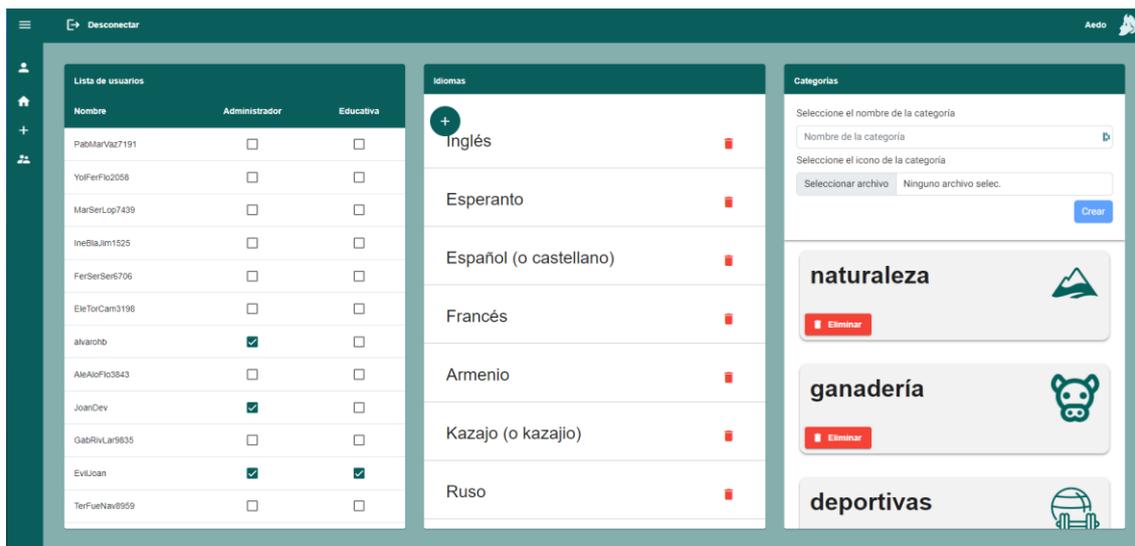


Figura 21: Dashboard de Aedo (Fuente: Elaboración propia)

Cada elemento del *dashboard* (ver **Figura 21**) dispone de su componente y su función, esto permite que sea muy fácil de escalar la funcionalidad del *dashboard*. Actualmente facilita la gestión de los roles de los usuarios, los idiomas de las experiencias y las categorías de las que dispone la aplicación.

4.8 Buenas prácticas

Para asegurar el desarrollo eficiente, mantenible y de alta calidad del código desarrollado durante este proyecto, se han seguido una serie de buenas prácticas:

- Se ha mantenido una estructura de carpetas organizada y modular con el objetivo de separar los distintos módulos, componentes, modelos y otros. Esto facilita la navegación dentro del proyecto.
- Para minimizar la duplicación de código se implementaron los componentes utilizando las directivas de entrada y salida de Angular. Estas directivas son una herramienta de Angular que permite la comunicación entre componentes.
- No sobreutilizar los servicios. Intentar que solo se utilicen en las páginas del proyecto y no los componentes.
- Se usó la programación reactiva que dispone Angular. Esta permite la gestión de eventos y el manejo de datos asíncronos.
- Mantenimiento de la documentación durante todo el curso del proyecto.

4.8.1 Componentes dumb y Smart

Una buena práctica seguida que tiene relación con las prácticas anteriores es la de separar los componentes según su funcionalidad. Esto permite que cada componente tenga una única responsabilidad y se mantenga lo más independiente de otros componentes. Teniendo esto en cuenta se crearon dos clases de componentes distintos:

El primer grupo de componentes son los componentes “dumb” o de presentación, el único objetivo de estos componentes son la representación de los datos que reciben a través de sus directivas de entrada, no tienen conocimiento de cómo se obtienen los datos que representa y no interactúan con la lógica de negocio.

El segundo grupo de componentes son los componentes “smart” o de contenedor, estos componentes se encargan de gestionar la lógica de negocio y hacer llamadas a los distintos servicios del proyecto. También al ser el contenedor de distintos componentes “dumb” son los que obtienen la información que estos componentes representarán y los que gestionarán las directivas de salida que emitirán (ver **Figura 22**).

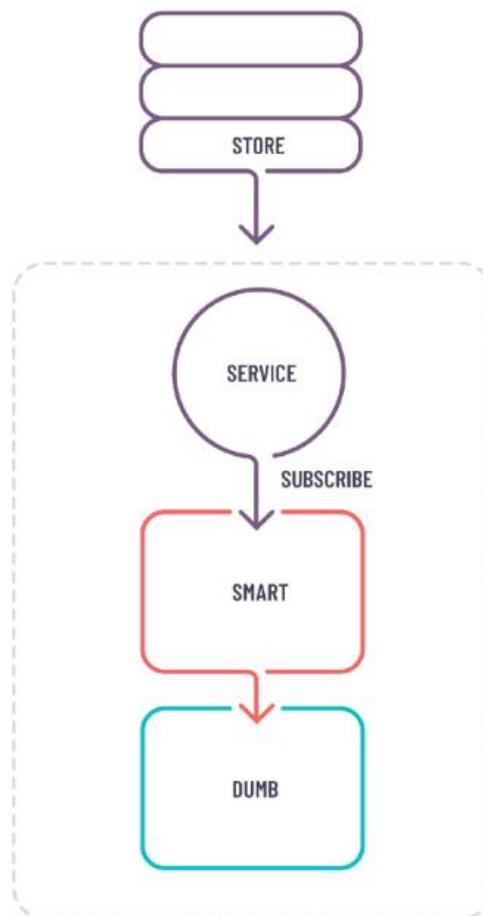


Figura 22: Representación de flujo de datos entre componentes (Fuente: <https://generic-ui.com/blog/enterprise-approach-to-the-smart-and-dumb-components-pattern-in-angular>)

En el proyecto podemos diferenciar estos dos grupos debido a donde se localizan. Los componentes de representación se encuentran dentro de la carpeta **components** mientras que los componentes de presentación se encuentran dentro de la carpeta **pages**. En la **Figura 23** y la **Figura 24** se ven un ejemplo de la implementación de un componente de presentación y el resultado de dicho componente en la interfaz del usuario.

```
<mat-grid-tile *ngFor="let odisea of odiseas; let i = index">
  <app-odisea-item-card
    [odisea]="odisea"
    [imageUrl]="imageUrls[i]"
    [selected]="odisea == selectedOdisea"
    (cardClick)="onCardClick($event)"
  ></app-odisea-item-card>
</mat-grid-tile>
</mat-grid-list>
```

Figura 23: Ejemplo de uso de un componente dumb (Fuente: Elaboración propia)

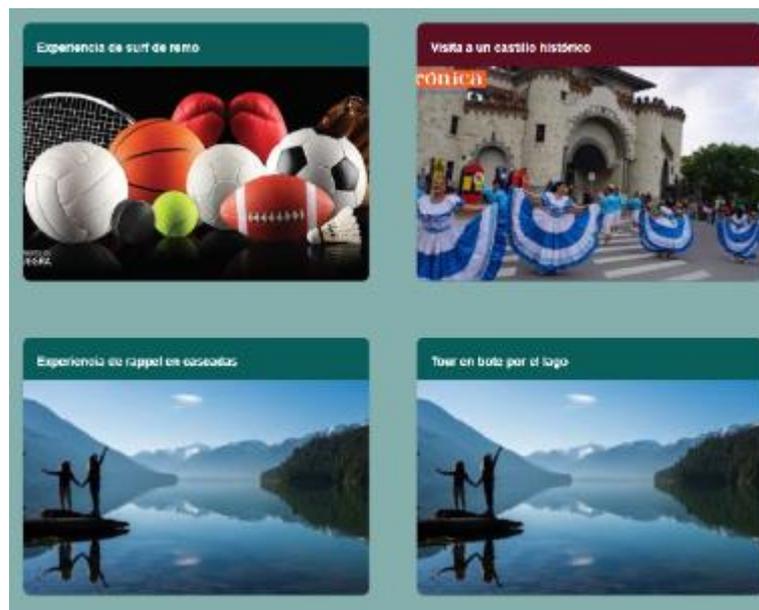


Figura 24: Representación de un componente dumb (Fuente: Elaboración propia)

4.9 Librerías utilizadas

En el marco del proyecto se han utilizado diversas librerías que han sido muy importantes para el desarrollo del proyecto. En la medida de lo posible, se ha intentado que la cantidad de

dependencias del proyecto sea la menor posible. La lista completa de dependencias del proyecto se puede ver de manera más detallada en la **Figura 25**.

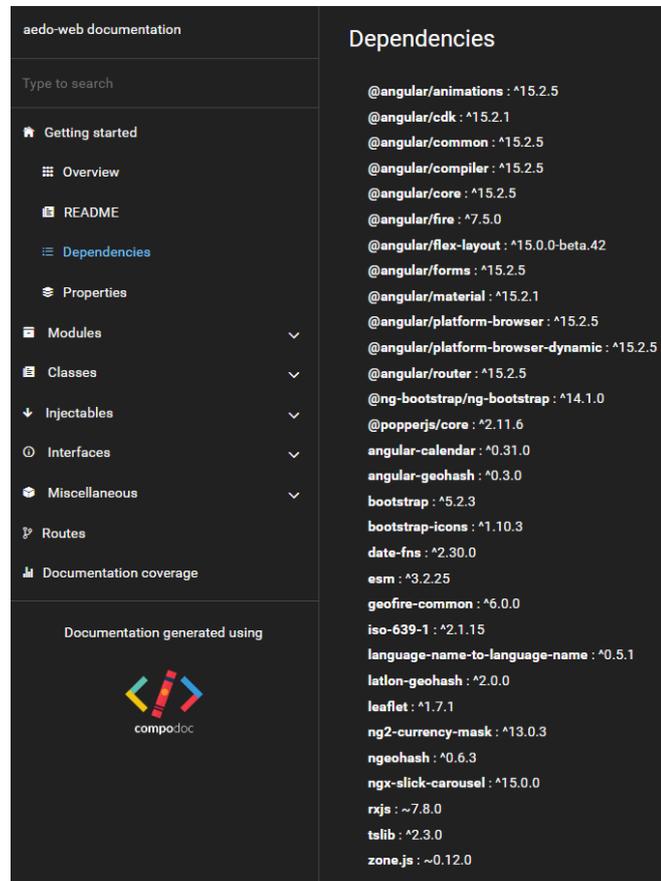


Figura 25: Lista de dependencias (Fuente: Elaboración propia)

De las dependencias mostradas anteriormente las más importantes para el proyecto serían:

- **@angular/fire:** Esta dependencia es parte de *AngularFire*, la cual es la librería oficial de *Firebase* para *Angular*. Es la usada para interactuar con los servicios de *Firebase* desde la aplicación.
- **Rxjs:** Es la librería de programación reactiva utilizada por *Angular*. La cual le permite manejar los eventos y flujos de datos asíncronicos.
- **@angular/material:** Es una biblioteca de componentes de interfaz de usuario basada en *Material Design*⁹. La mayoría de los componentes del proyecto dependen de esta biblioteca.
- **@angular/animations:** Es la dependencia que proporciona la capacidad de añadir los efectos de transición en el proyecto.
- **@angular/cdk:** Ofrece funcionalidades útiles para el desarrollo de aplicaciones web en *Angular*. Es la que permite el manejo de los diseños reactivos.

⁹ Material Design: <https://m3.material.io>

- **@angular/core:** La dependencia fundamental de *Angular*. Proporciona las funcionalidades de gestión de *componentes*¹⁰, *directivas*¹¹ y *servicios*¹².

4.10 Componentes utilizados

En los inicios del proyecto, se propuso seguir los principios de *Material Design*, basado en el compromiso del equipo de desarrollo de crear una interfaz de usuario atractiva, intuitiva y accesible. Para ello, dependimos de la biblioteca de Angular Material, cuyos componentes ya siguen por defecto los principios impuestos por Google.

Dentro de la amplia variedad de componentes de los que dispone esta biblioteca nos enfocamos en siete componentes que no sólo fueron elegidos por su funcionalidad individual, sino también por cómo se alineaban y contribuían conjuntamente al objetivo global de la aplicación. Estos componentes en conjunto formaron un sistema cohesivo que apoya positivamente la funcionalidad de nuestra aplicación.

4.10.1 MatToolBar

La barra de herramientas (**Figura 26**) fue un componente necesario a la hora de poder ofrecer al usuario una cabecera atractiva para la página. Permitió darle al usuario funcionalidades de gestión de su perfil en la aplicación mientras daba un sentido de uniformidad a la hora de navegar por la aplicación.



Figura 26: Toolbar de Aedo (Fuente: Elaboración propia)

4.10.2 MatSideNav

Fue el componente clave para la navegación de la aplicación. El componente por defecto permite ocultarlo completamente y hacerlo aparecer cuando sea necesario, esto permite que la pantalla del usuario esté despejada cuando está realizando una opción que no requiera el uso del componente. El equipo de desarrollo modificó el componente por defecto para evitar que se oculte completamente y que colapse solo mostrando los iconos de los elementos del menú. Este tipo de variante se le llama “*mini variant*” y permite al usuario seguir usando la funcionalidad, incluso cuando el *sidenav* está oculta.

¹⁰ Componentes: <https://docs.angular.lat/guide/architecture-components>

¹¹ Directivas: <https://docs.angular.lat/guide/testing-attribute-directives>

¹² Servicios: <https://docs.angular.lat/guide/singleton-services>

En conjunto con el componente *toolbar* y el tema personalizado del proyecto desempeñan un papel clave a la hora de garantizar que cada página tenga un aspecto uniforme, permitiendo que la aplicación en su totalidad tenga una apariencia coherente (**Figura 27**).

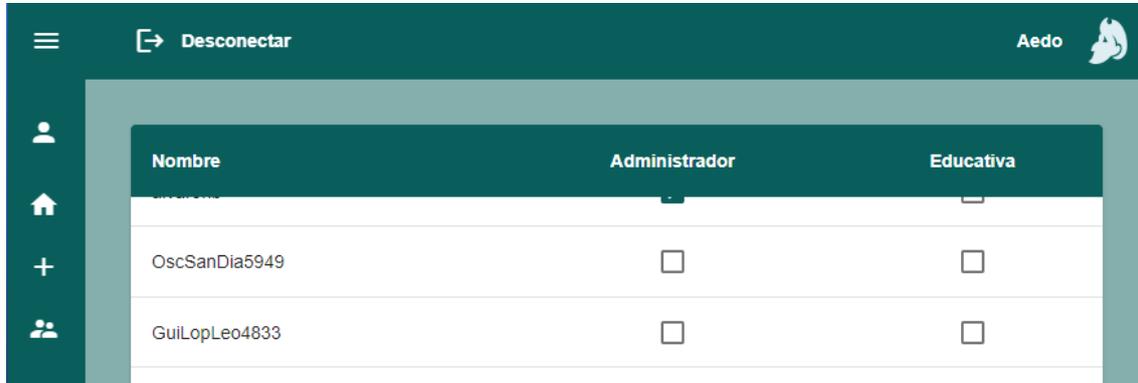


Figura 27: Sidenav y toolbar de Aedo en conjunto (Fuente: Elaboración propia)

4.10.3 MatCard

En la aplicación se utilizó con frecuencia este componente para formar los componentes “dumb” (**Figura 29**) de nuestra aplicación por su utilidad para mostrar información de manera ordenada y concisa. La versatilidad del componente fue un factor decisivo en su implementación ya que necesitábamos que el componente también fuera capaz de ser utilizado como contenedor para otros componentes ya con sus funcionalidades integradas como fue en el caso de la *dashboard* de administración (**Figura 28**).

Lista de usuarios		
Nombre	Administrador	Educativa
NerRomGar5160	<input type="checkbox"/>	<input type="checkbox"/>
BeaGomSer1376	<input type="checkbox"/>	<input type="checkbox"/>
EleGarFer1228	<input type="checkbox"/>	<input type="checkbox"/>
AlvCasAlv6051	<input type="checkbox"/>	<input type="checkbox"/>
alvaro hb	<input checked="" type="checkbox"/>	<input type="checkbox"/>
OscSanDia5949	<input type="checkbox"/>	<input type="checkbox"/>
GuiLopLeo4833	<input type="checkbox"/>	<input type="checkbox"/>
FraLopMar3711	<input type="checkbox"/>	<input type="checkbox"/>
JoanDev	<input checked="" type="checkbox"/>	<input type="checkbox"/>
EvilJoan	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PauJimMor4891	<input type="checkbox"/>	<input type="checkbox"/>
MarLopSer6963	<input type="checkbox"/>	<input type="checkbox"/>

Figura 28: MatCard como contenedor de componentes (Fuente: Elaboración propia)



Figura 29: MatCard como componente de representación (Fuente: Elaboración propia)

4.10.4 MatButton

Todos los botones de nuestra aplicación web fueron proporcionados por Angular Material. Estos botones nos brindaron una forma de controlar las acciones que realiza el usuario en la aplicación.

Gracias al tema personalizado que hemos implementado en nuestra aplicación, la tarea de cambiar el color de estos componentes según su funcionalidad se vuelve mucho más sencilla como se puede ver en la **Figura 30**.



Figura 30: Tres paletas de color en botones (Fuente: Elaboración propia)

4.10.5 MatFormField y MatInput

Estos dos componentes se utilizaron conjuntamente para la implementación de los formularios de la aplicación, desde la creación y modificación de usuarios, hasta la creación de las experiencias subidas por los mismos. Entre las ventajas que nos ofrecieron estos componentes estuvo la de poder gestionar los diferentes tipos de validaciones, lo que permitió poder darle más información al usuario en caso de errores a la hora de rellenar los formularios, mientras que a la vez reducía el tiempo necesario para crear los mismos formularios al equipo de desarrollo. En la **Figura 31** se muestra un ejemplo de ambos componentes.

Registrar actividad

naturaleza	ganaderia	deportivas
artisticas	cultural	acuáticas
adrenalina	gastronomia	agricultura
artesanía	musicales	historia

Inglés	Esperanto
Español	Francés
Armenio	Kazajo
Ruso	Chino

Figura 31: Formulario creación de actividad (Fuente: Elaboración propia)

4.10.6 MatTable

Cuando en la aplicación se necesitó representar datos complejos, en vez de optar por listas de componentes decidimos utilizar las tablas que nos proporciona Angular Material. Estas tablas nos permitieron personalizar completamente las tablas que fuimos implementando en el proyecto. Desde la paginación hasta la filtración y ordenación de los datos. Dado que nuestra aplicación ya contaba con un tema personalizado de Angular, fue sencillo asegurar que el estilo de las tablas se alineara con el diseño general de la aplicación.

Nombre	Administrador	Educativa
NerRomGar5160	<input type="checkbox"/>	<input type="checkbox"/>
BeaGomSer1376	<input type="checkbox"/>	<input type="checkbox"/>
EleGarFer1228	<input type="checkbox"/>	<input type="checkbox"/>
AlvCasAlv6051	<input type="checkbox"/>	<input type="checkbox"/>
alvarohb	<input checked="" type="checkbox"/>	<input type="checkbox"/>
OscSanDia5949	<input type="checkbox"/>	<input type="checkbox"/>
GuiLopLeo4833	<input type="checkbox"/>	<input type="checkbox"/>
FraLopMar3711	<input type="checkbox"/>	<input type="checkbox"/>
JoanDev	<input checked="" type="checkbox"/>	<input type="checkbox"/>
EvilJoan	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PauJimMor4891	<input type="checkbox"/>	<input type="checkbox"/>
MarLopSer6963	<input type="checkbox"/>	<input type="checkbox"/>

Figura 32: Tabla de gestión de roles de usuario (Fuente: Elaboración propia)

La versatilidad de este tipo de tablas no solo fue útil a la hora de representar datos sino también a la hora de tener que gestionar algunas modificaciones en los datos representados. En la **Figura 32** se muestra un caso donde fue necesario la gestión de las modificaciones.

4.8 Desafíos de programación

A lo largo del proyecto surgieron diversos problemas que pusieron a prueba las capacidades de resolución del equipo de desarrollo. Fue necesario tiempo y esfuerzo por parte de todos los individuos del equipo para poder superar las dificultades y mantener el avance constante del proyecto. A continuación, se hablará más en detalle de los problemas principales que se encontraron en la gestión del proyecto.

4.8.1 Subida de datos TypeScript a Firebase Firestore

Cuando iniciamos la subida de datos a la base de datos desde la página web, nos enfrentamos a un desafío inesperado, un problema que no se presenta en la aplicación móvil del proyecto debido a la diferencia en el lenguaje de programación empleado.

Mientras que la aplicación móvil utilizada JavaScript, la página web utiliza *TypeScript*. Como *Firebase* no es capaz de reconocer los tipos que se estaban creando en el proyecto de *Angular* surgía un error al subirlos a la base de datos.

El equipo de desarrollo optó por modificar el método de subida de los objetos a la base de datos. En vez de mandar el objeto estructurado con los tipos creados dentro del proyecto, se añadieron en las clases de objetos métodos *toJSON()* (ver **Figura 33**) que se encargaran de realizar la conversión de los parámetros de las clases a un objeto JSON que *Firebase* pudiera reconocer. A su vez, se realizaron los cambios necesarios en la recogida de estos mismos datos para poder transformar estos objetos en formato *JSON* en objetos de la clase. Con el objetivo de poder utilizar los métodos específicos de la clase (ver **Figura 34**).

```
/**
 * Converts the Odisea object to JSON format.
 * @returns The Odisea object in JSON format.
 */
toJSON() {
  return {
    description: this.description,
    images: this.images,
    languages: this.languages.map((Language) => Language.toJSON()),
    maxCapacity: this.maxCapacity,
    name: this.name,
    numberVotes: this.numberVotes,
    totalScoreVotes: this.totalScoreVotes,
    uid: this.uid,
    tags: this.tags,
    location: this.location,
    price: this.price,
  };
}
```

Figura 33: Método *toJSON* de *Odisea* (Fuente: Elaboración propia)

```

/**
 * Gets a comment by its ID.
 * @param id - Comment ID.
 * @returns A promise that resolves with the corresponding CommentDto object.
 */
async getById(id: string): Promise<CommentDto> {
  return new Promise<CommentDto>((resolve, reject) => {
    this.firestoreService
      .getById<any>(this.collection, id)
      .then((data) => {
        const { body, odiseaId, rating, reservationId, userID, username } =
          data;

        const comment = new CommentDto(
          id,
          body,
          odiseaId,
          rating,
          reservationId,
          userID,
          username
        );
        resolve(comment);
      })
      .catch((error) => {
        reject(error);
      });
  });
}

```

Figura 34: Recuperación de comentario (Fuente: Elaboración propia)

En conclusión, este planteamiento no solo arregló el problema, sino que también siguió las buenas prácticas impuestas por el equipo y que en el futuro en caso de cambios en las clases del proyecto debido a la introducción de nuevas funcionalidades el impacto en el proyecto sea mínimo ya que solo se modificarían los objetos afectados sin necesidad de afectar a los servicios.

4.8.2 Comunicación efectiva entre equipos

En un proyecto como este, en el cual las clases de objetos se comparten y se modifican, es necesario que exista un protocolo de comunicación riguroso. Este protocolo requería que el equipo que realizara el cambio necesitaba comunicar previamente a los otros miembros del proyecto que cambios se iban a realizar y que se debería modificar. Además, durante la implementación del proyecto se realizaron las *weekly scrum*¹³. En estas, cada miembro del equipo podía explicar los progresos que ha realizado durante la semana, los problemas que se han encontrado y si han conseguido o saben cómo solucionarlos o si es necesario ajustar los objetivos que queríamos cumplir durante el *sprint*.

¹³ Weekly scrum: <https://medium.com/agileinsider/the-weekly-scrum-what-is-it-7fdf631aa4eb>

4.8.3 Uso de servicios personalizados en la base de datos de Firebase

Al igual que nuestra aplicación móvil dispone de *APIs* individuales para cada objeto de la base de datos, en la página web optamos por hacer lo mismo, pero aprovechando los servicios que nos ofrece *Angular*. Por lo tanto, creamos unos servicios individuales para cada objeto de la base de datos.

Este enfoque nos proporcionó varias ventajas. La primera fue una mejora de la modularidad y la organización del código, ya que cada servicio se encarga de un conjunto específico de operaciones relacionadas con el objeto al que está relacionado. La siguiente mejora fue referente a la reutilización del código. Dado que cada servicio se encarga de las operaciones CRUD de su objeto, estos servicios se pueden reutilizar en diferentes partes de la aplicación, evitando así que exista código duplicado. En tercer lugar, este enfoque nos ayudó a mantener un alto grado de cohesión y bajo acoplamiento, lo que es fundamental para construir aplicaciones escalables y mantenibles.

Además de las ventajas anteriores, los servicios de *Angular* implementan el patrón *Singleton*¹⁴. Esto hace que *Angular* solo cree una única instancia de cada servicio que se proporciona en la 'raíz'. Esto permite que los componentes que utilizan este servicio solo utilicen una sola instancia, lo que permite que se compartan los datos de forma eficiente en las distintas partes de nuestra aplicación.

```
@Injectable({
  providedIn: 'root',
})
export class FirestoreService {
  /**
   * Constructs an instance of the FirestoreService.
   * @param firestore - Firestore instance provided by Angular.
   */
  constructor(private firestore: Firestore) {}

  /**
   * Retrieves a collection of documents from Firestore.
   * @param coll - Collection name.
   * @returns An observable that emits the collection of documents.
   */
  getCollection = (coll: any) => {
    const collRef = collection(this.firestore, coll);
    return collectionData(collRef, { idField: 'id' });
  };
};
```

Figura 35: Servicio genérico (Fuente: Elaboración propia)

¹⁴ Singleton: <https://refactoring.guru/es/design-patterns/singleton>

```

@Injectable({
  providedIn: 'root',
})
/**
 * Service for managing odyssey entities.
 */
export class OdiseaService {
  collection: string = 'odiseas';

  /**
   * Constructor for OdiseaService.
   * @param firestoreService Instance of FirestoreService.
   */
  constructor(private firestoreService: FirestoreService) {}

  /**
   * Creates a new odyssey entity.
   * @param odisea The odyssey object to create.
   * @returns A promise that resolves when the entity is successfully created.
   */
  create(odisea: IOdisea) {
    return this.firestoreService.create(this.collection, odisea.toJSON());
  }
}

```

Figura 36: Servicio específico (Fuente: Elaboración propia)

La **Figura 35** y la **Figura 36** ilustran la implementación de estos servicios. El servicio general la cual todos los servicios referentes a objetos de la base de datos implementan define las operaciones CRUD básicas de cada servicio. Los servicios específicos adaptan estos métodos a su objeto correspondiente. Al hacerlo, nos garantizamos que todos los servicios específicos mantengan un conjunto coherente y predecible de las operaciones básicas mientras que a la vez nos proporciona flexibilidad para adaptar cada servicio a sus requisitos específicos.

Resumiendo, el desarrollo de la página web no ha estado exento de desafíos, tanto anticipados como inesperados. En cada situación, se propusieron y aplicaron soluciones efectivas para superar dichos obstáculos. Se usaron efectivamente las herramientas proporcionadas por Angular, así como de otras tecnologías integradas en el proyecto. Estos desafíos aportaron experiencia al equipo, que resultaron en una mejor comprensión y aplicación de las prácticas de desarrollo.

5. Cronología del TFG

La cronología de la aplicación se divide en seis *sprints* de programación y un séptimo inicial que se empleó para definir la idea de la aplicación, evaluar la idea de negocio, realizar el estudio de mercado, mapa de características, modelo de negocio y proyección económica. Todo el procedimiento está relacionado con los proyectos de emprendimiento. También se tuvo que elaborar un *pitch* de la idea de negocio, preparar el entorno de desarrollo y formarse los miembros para no afrontar el proyecto sin tener ningún conocimiento.

A continuación, se detallan los seis sprints principales y las áreas en las que el equipo de desarrollo se enfocó durante cada uno de ellos. Es importante destacar que todos los miembros del equipo desempeñaron tareas de desarrollo *full stack*, por lo que las actividades mencionadas abarcaron diferentes aspectos del proyecto.

5.1 Sprint 1 (PIN: Proyecto de Ingeniería del *software*)

Durante el primer *sprint*, se realizaron las implementaciones fundamentales para que la aplicación pudiera admitir usuarios, permitir el registro de experiencias y la capacidad de realizar reservas. Para lograr esto, se desarrollaron diversas pantallas y componentes, como una barra de navegación inferior, una pantalla de inicio que mostraba una lista de todas las experiencias disponibles, y la posibilidad de acceder a cada una de ellas para ver su información y realizar reservas. Además, se incorporaron las funcionalidades de edición y eliminación tanto para las experiencias como para los usuarios.

Estas implementaciones iniciales sentaron las bases funcionales de la aplicación, permitiendo a los usuarios interactuar con las experiencias, realizar cambios en ellas y administrar sus propios perfiles de usuario.

5.2 Sprint 2 (PIN: Proyecto de Ingeniería del *software*)

Durante el segundo sprint, se realizó una adaptación del proyecto al modelo de tres capas (Presentación, Lógica y capa de Datos). También se llevó a cabo la implementación del contexto de usuario, que resultaba esencial para acceder al usuario autenticado siempre que fuera necesario.

Hablando de nuevos requisitos, se introdujo la capacidad de registrar experiencias en varios idiomas y utilizar diferentes calendarios, lo cual implicó una reestructuración del modelo de datos existente. Este cambio permitió brindar soporte para experiencias multilingües y adaptarse a diferentes sistemas de calendario.

Además, se implementaron nuevas características clave, como la integración de un mapa para visualizar la ubicación de las experiencias, la incorporación de un buscador para facilitar la

búsqueda y filtrado de experiencias específicas, la implementación de un control para gestionar el aforo máximo de las experiencias y la funcionalidad de comentar en las mismas.

Estas adiciones en el segundo *sprint* enriquecieron significativamente la experiencia de los usuarios al ampliar las opciones de personalización y brindarles herramientas para interactuar y compartir sus opiniones sobre las experiencias registradas.

5.3 Sprint 3 (PIN: Proyecto de Ingeniería del *software*)

Durante el tercer *sprint*, el enfoque principal del equipo fue refinar y mejorar los aspectos ya implementados de la aplicación, corrigiendo errores y excepciones para asegurar su funcionamiento óptimo de cara a la Feria de Proyectos. Se llevaron a cabo reuniones con los alumnos de bellas artes para colaborar en el diseño de las interfaces gráficas de las distintas pantallas, utilizando el material proporcionado por ellos.

Dado que no se contaba con maquetas para todas las ventanas de la aplicación, se implementaron las restantes siguiendo una estética similar a la proporcionada por los compañeros de bellas artes, con el objetivo de mantener la coherencia visual en toda la aplicación.

En términos de funcionalidades, se incorporó una versión básica de inteligencia artificial, se trató de un algoritmo de recomendaciones basado en las experiencias consultadas hasta el momento. Este agregado permitió ofrecer sugerencias personalizadas a los usuarios en función de sus intereses previos.

Además, se trabajó en la mejora de los filtros de búsqueda de experiencias en la pantalla del buscador, así como en la ventana del mapa. Estas mejoras facilitaron la navegación y la localización de experiencias relevantes para los usuarios.

5.4 Sprint 4

Durante el cuarto *sprint*, se inició el desarrollo de la página web y con ello inició otro periodo de formación necesario para el equipo de desarrollo para entender las nuevas herramientas necesarias para poder implementar dicha página web. Se configuró el servicio de hosting, se creó las funcionalidades básicas de autenticación y se agregó la capacidad de eliminar y crear categorías e idiomas desde la página de administración.

A su vez comenzó la investigación de algoritmos y técnicas necesarias para desarrollar la inteligencia artificial con el fin de mejorar las funcionalidades de recomendación de experiencias basadas en los algoritmos utilizados hasta el momento.

5.5 Sprint 5

Durante el quinto *sprint* se continuó agregando funcionalidades básicas similares a las disponibles en la aplicación móvil en la página web. Esto implicó desarrollar y adaptar características como la visualización de experiencias, la reserva de actividades y la gestión de perfiles desde la plataforma web.

En el apartado de la inteligencia artificial, se implementaron algoritmos y técnicas adecuadas para mejorar las recomendaciones de experiencias y personalizar aún más la interacción de los usuarios con la aplicación.

En el apartado móvil se introdujeron las cuentas educativas para los profesores, brindándoles un acceso especializado y permitiéndoles gestionar y supervisar las actividades de sus estudiantes dentro del sistema.

Un aspecto importante abordado durante esta iteración fue el desarrollo y mejora del script de generación de datos en Firebase. Este script era necesario para proporcionar datos de muestra y entrenar la inteligencia artificial de manera efectiva, lo que contribuyó a mejorar la calidad de las recomendaciones realizadas.

5.6 Sprint 6

Durante el sexto y último *sprint*, el equipo se enfocó en agregar nuevas funcionalidades a la página web y finalizar la programación y pulido de la inteligencia artificial.

En el apartado web, se implementó el tema personalizado en la totalidad de la aplicación aumentando la atraktividad de la página y se creó la documentación del proyecto.

En el apartado móvil se implementó la capacidad de realizar múltiples reservas desde una misma cuenta de usuario, lo que brindó mayor flexibilidad y comodidad a los usuarios al interactuar con la aplicación.

Además, se generó la APK de la aplicación para dispositivos móviles y se procedió a publicarla en la tienda de Google, para así realizar los experimentos con los usuarios.



6. Conclusiones y trabajo futuro

6.1 Conclusiones

Durante el desarrollo de este proyecto se ha realizado un gran esfuerzo para obtener un producto mínimo viable capaz de cumplir con la mayoría de los requisitos impuestos por el equipo de desarrollo. Dicho esto, aún existe un margen para el desarrollo de la aplicación tanto por la parte web como en la aplicación móvil.

Durante los sprints 4, 5 y 6, el equipo de desarrollo se encargó de implementar la parte web de la aplicación incluyendo la documentación de esta, obteniendo así un producto que funciona en sintonía con la aplicación móvil, parte principal de la idea de negocio.

En resumen, este proyecto ha logrado satisfacer las metas establecidas, proporcionando un producto tanto operativo como visualmente atractivo. Al mismo tiempo, se han establecido cimientos sólidos para futuras iteraciones y mejoras, abarcando tanto la calidad del código como la experiencia del usuario.

6.2 Trabajo futuro

Aunque se han implementado numerosas funcionalidades impuestas por el equipo de desarrollo, hubo algunas que quedaron sin implementarse debido a la falta de tiempo. En el futuro, estas funcionalidades serán añadidas.

- Integrar el sistema de chat interno, lo cual permitirá la comunicación entre usuarios y anfitriones sin necesidad de aplicaciones externas.
- Implementar el sistema de pago con el fin de aumentar la comodidad de los usuarios a la hora de realizar las transacciones.
- Incorporar un sistema de notificaciones, lo que permitirá avisar de los cambios realizados tanto a la hora de realizar reservas en la aplicación como de notificar cambios realizados por el usuario a través de la página web.
- Seguir añadiendo funcionalidades de administración con el objetivo de aumentar la comodidad del equipo de administración.
- Creación de pruebas automáticas de las funcionalidades principales de la página web, aumentando la seguridad y consistencia de esta.
- Seguir incluyendo funcionalidades exclusivas de la versión móvil con el fin de mejorar la experiencia de los usuarios en distintos dispositivos.

7. Glosario de términos

MVP: Minimum Viable Product. (Gasbarrino s.f.) Producto viable mínimo. En términos de desarrollo de aplicaciones, se trata de un producto con un número de características suficientes como para satisfacer las necesidades iniciales de los clientes.

Back-end: Complementario al *front-end*. El *back-end* (Machuca, 2022) está formado por todos aquellos elementos de la página web o de la aplicación los cuales no son accesibles para los usuarios finales de la aplicación.

Framework: Un *framework* (Framework, 2022), o marco de trabajo, tiene como objetivo facilitar las tareas de desarrollo que surgen a la hora de programar, estos aceleran el proceso de programar gracias a que facilitan las tareas propias por tanto su objetivo al final es facilitar las tareas a los programadores.

Front-end: Complementario al *back-end*. El *front-end* (Chapaval, 2018) está formado por la parte visible de la aplicación, sea una interfaz web, una aplicación móvil o de escritorio. Es la parte de la aplicación con la que los usuarios interactúan. Las letras, los colores, elementos... que los usuarios pueden visualizar.

Full stack: Programador que realiza las tareas tanto de *front-end* como de *back-end*.

API: Application Programming Interface. Actúa como intermediario entre dos partes. Pueden ser dos aplicaciones distintas o diferentes capas de una misma aplicación.

8. Bibliografía

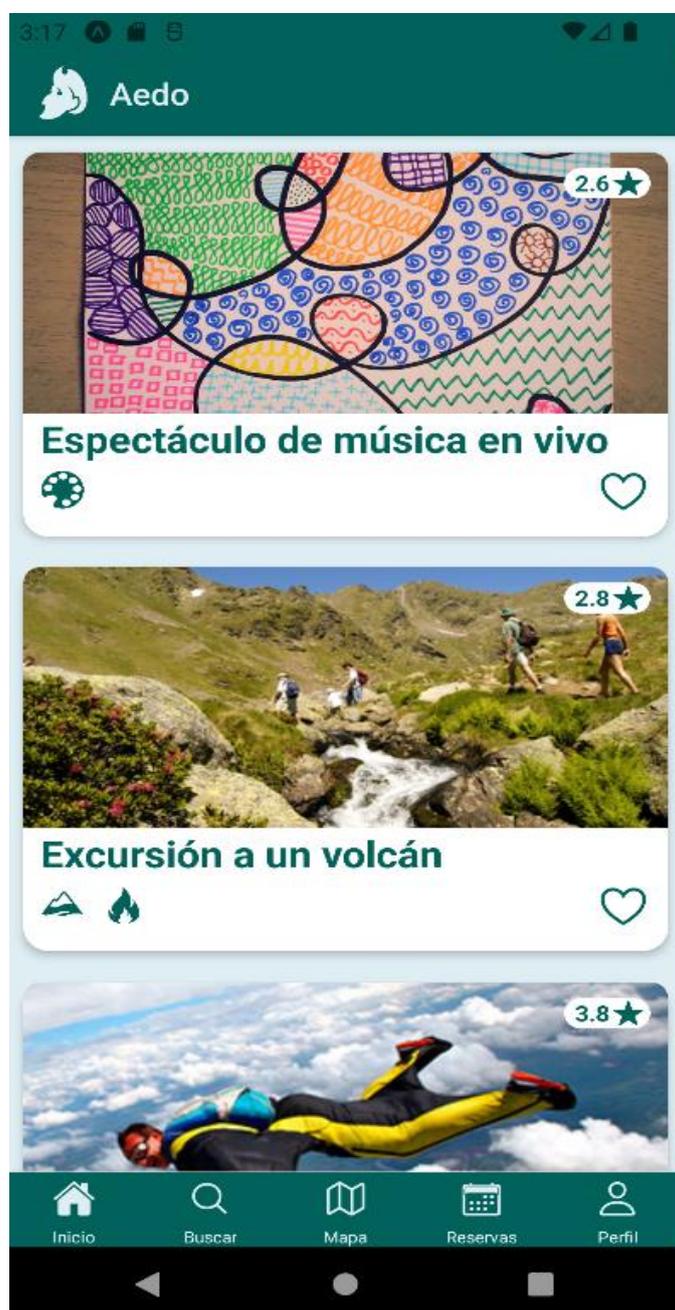
- *TypeScript Documentation* (s.f). TypeScript. <https://www.typescriptlang.org/docs/handbook/>
- *Angular Documentation*. (s.f). Angular. <https://angular.io/docs>
- *Material Design*. (s.f). Material Design. <https://m3.material.io>
- *What is Prettier?*. (s.f). Prettier. <https://prettier.io/docs/en/index.html>
- Angular. (2017, 4 de abril). *Angular Language Service*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=Angular.ng-template>
- Ugur Erdal. (2018, 4 de marzo). *Alphaetical Sorter*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=ue.alphabetical-sorter>
- Sergii N. (2016, 27 de enero). *Color Highlight*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=naumovs.color-highlight>
- SonarSource. (2017, 31 de mayo). *SonarLint*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=SonarSource.sonarlint-vscode>
- Alexander. (2019, 18 de mayo). *Error Lens*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=usernamehw.errorlens>
- jsayol. (2018, 9 de diciembre). *Firebase Explorer*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=jsayol.firebase-explorer>
- Prettier. (2017, 10 de enero). *Prettier*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=esbenp.prettier-vscode>
- GitKraken. (2016, 6 de septiembre). *GitLens – Git suprecharged*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=eamodio.gitlens>
- Gruntfuggly. (2017, 23 de noviembre) *Todo Tree*. Visual Studio Marketplace. <https://marketplace.visualstudio.com/items?itemName=Gruntfuggly.todo-tree>
- Kuhn, J. (2009). Decrypting the MoSCoW analysis. *The workable, practical guide to Do IT Yourself*, 5.
- Maurya, A. (2022). *Running lean*. O'Reilly Media, Inc.
- Objetivos de desarrollo sostenible. (25 de Setiembre, 2015). *Objetivos de desarrollo sostenible*. ONU. <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/> Consultado en Abril 2023
- Gasbarrino, S. (s.f.). MVP: qué es el producto mínimo viable, cómo hacerlo y ejemplos. *Hubspot*. <https://blog.hubspot.es/sales/producto-minimo-viable> Consultado en Mayo 2023.
- Machuca, F. (22 de Mayo, 2022). Backend: ¿qué es y para qué sirve este tipo de programación?. *Crehana*. <https://www.crehana.com/blog/transformacion-digital/que-es>
- Framework. (26 de Julio, 2022). Framework. *Edix*. <https://www.edix.com/es/instituto/framework/> Consultado en Mayo 2023.
- Chapaval, N. (2018). Qué es Frontend y Backend: diferencias y características. *Platzi*. <https://platzi.com/blog/que-es-frontend-y-backend/> Consultado en Mayo 2023.

9. Anexo I: Manual de uso de la aplicación

9.1 Manual de uso móvil

Pantalla de inicio.

En esta pantalla aparecen todas las experiencias propuestas por los distintos usuarios en la aplicación. En la parte inferior aparecen recomendaciones personalizadas por una inteligencia artificial.



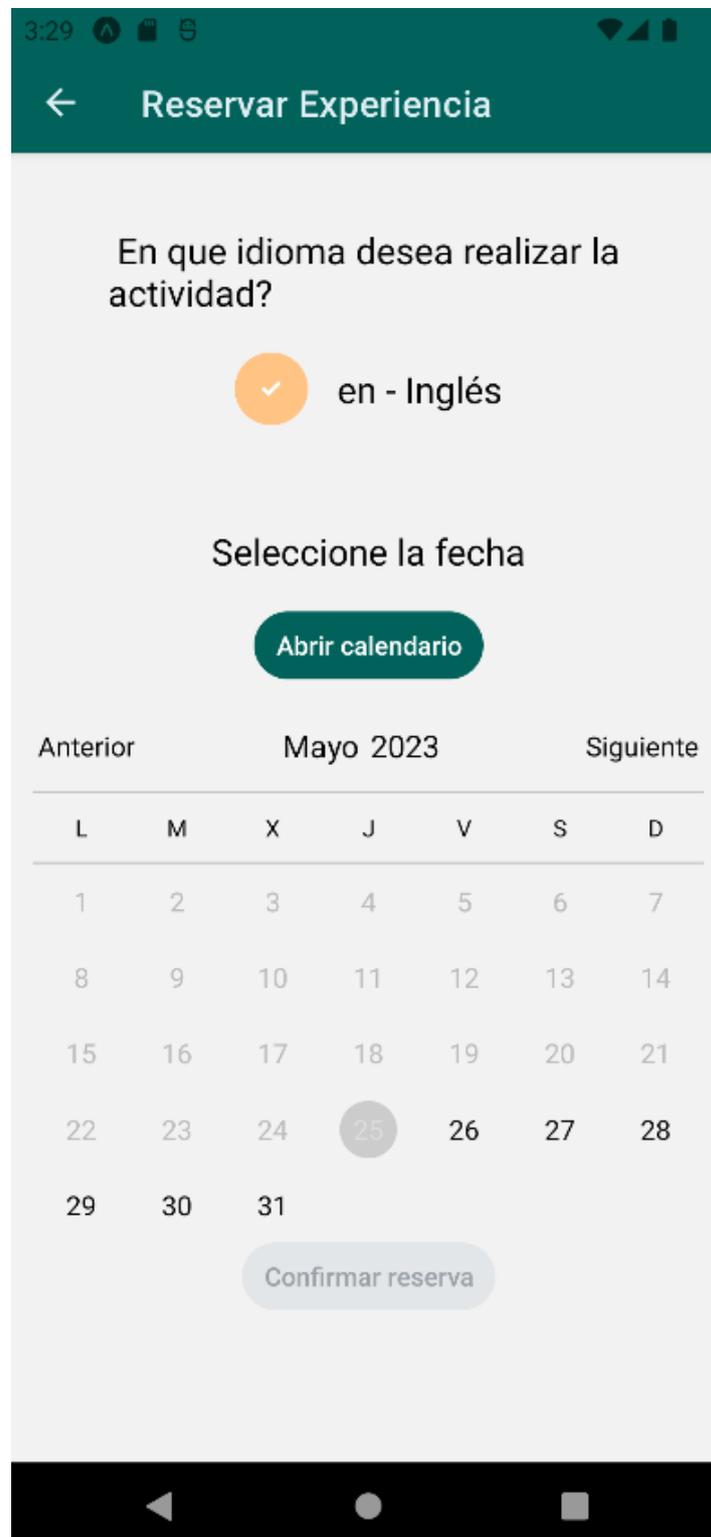
Pantalla de información de una experiencia.

Aparece la información de la experiencia, así como los comentarios y valoraciones dados por otros usuarios de la aplicación. Además, se puede reservar la experiencia.



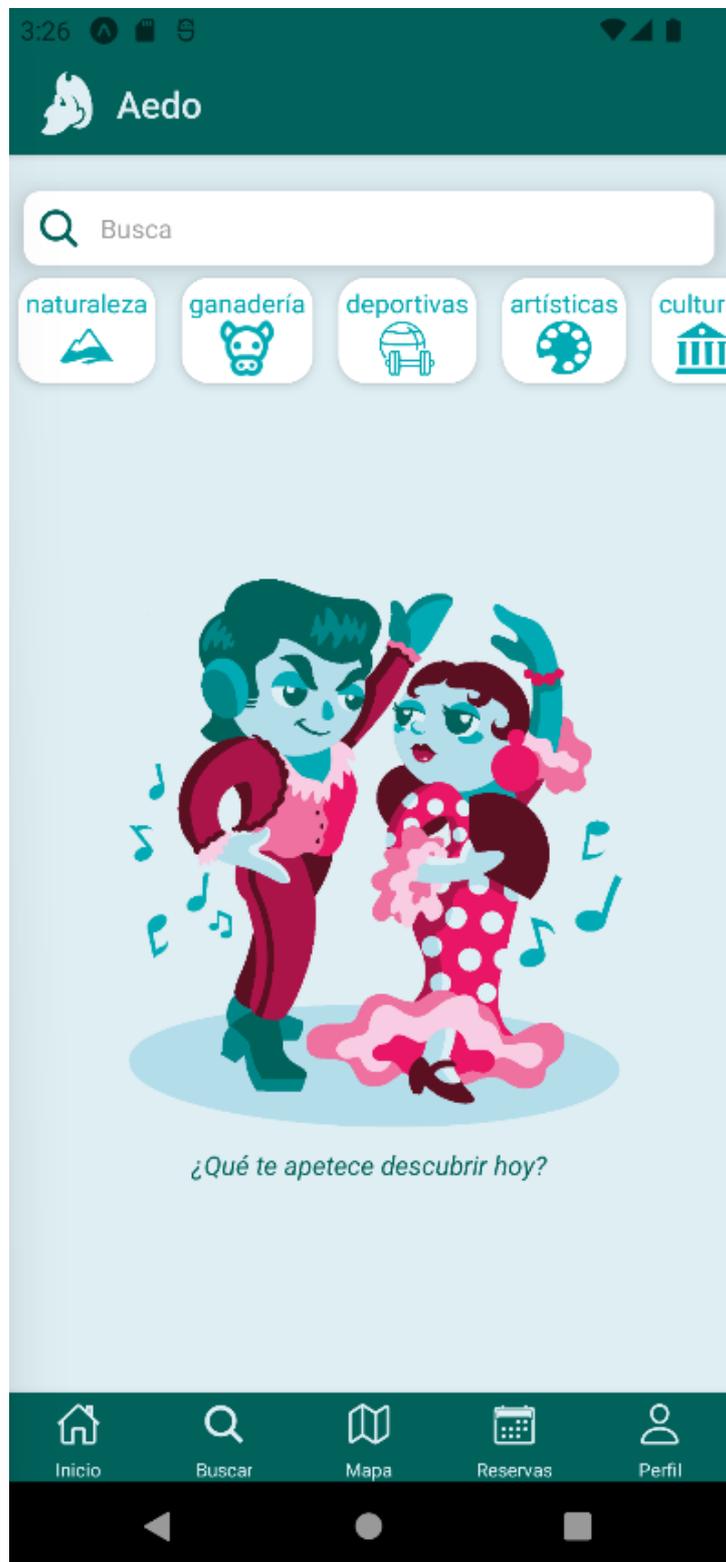
Pantalla de selección de fecha de reserva.

Al intentar reservar una experiencia, aparecen los distintos idiomas disponibles y un calendario para poder escoger la fecha de la reserva.



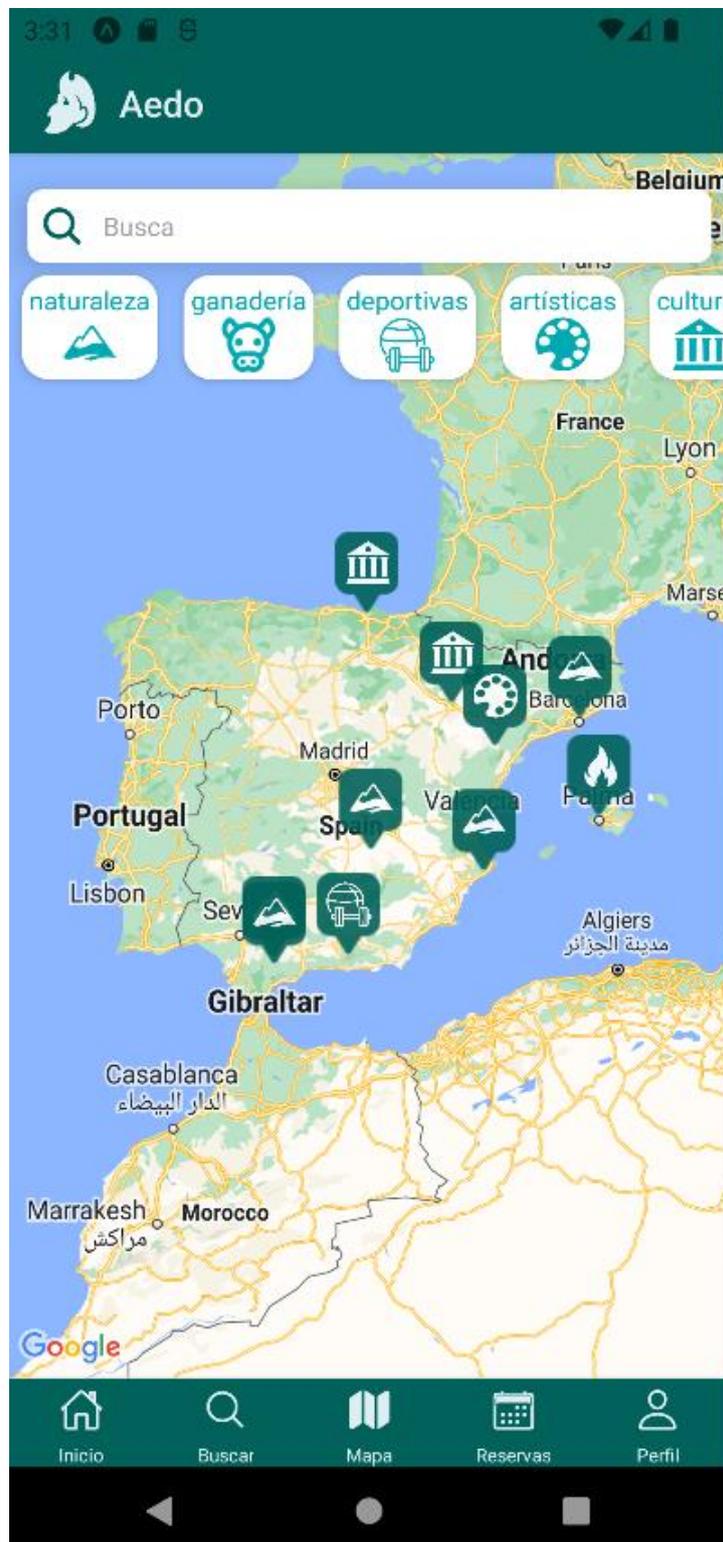
Pantalla de buscar:

Esta pantalla muestra experiencias próximas a la ubicación del dispositivo. Además, ofrece un buscador para buscar por localidad y un sistema de filtros por categorías de experiencias.



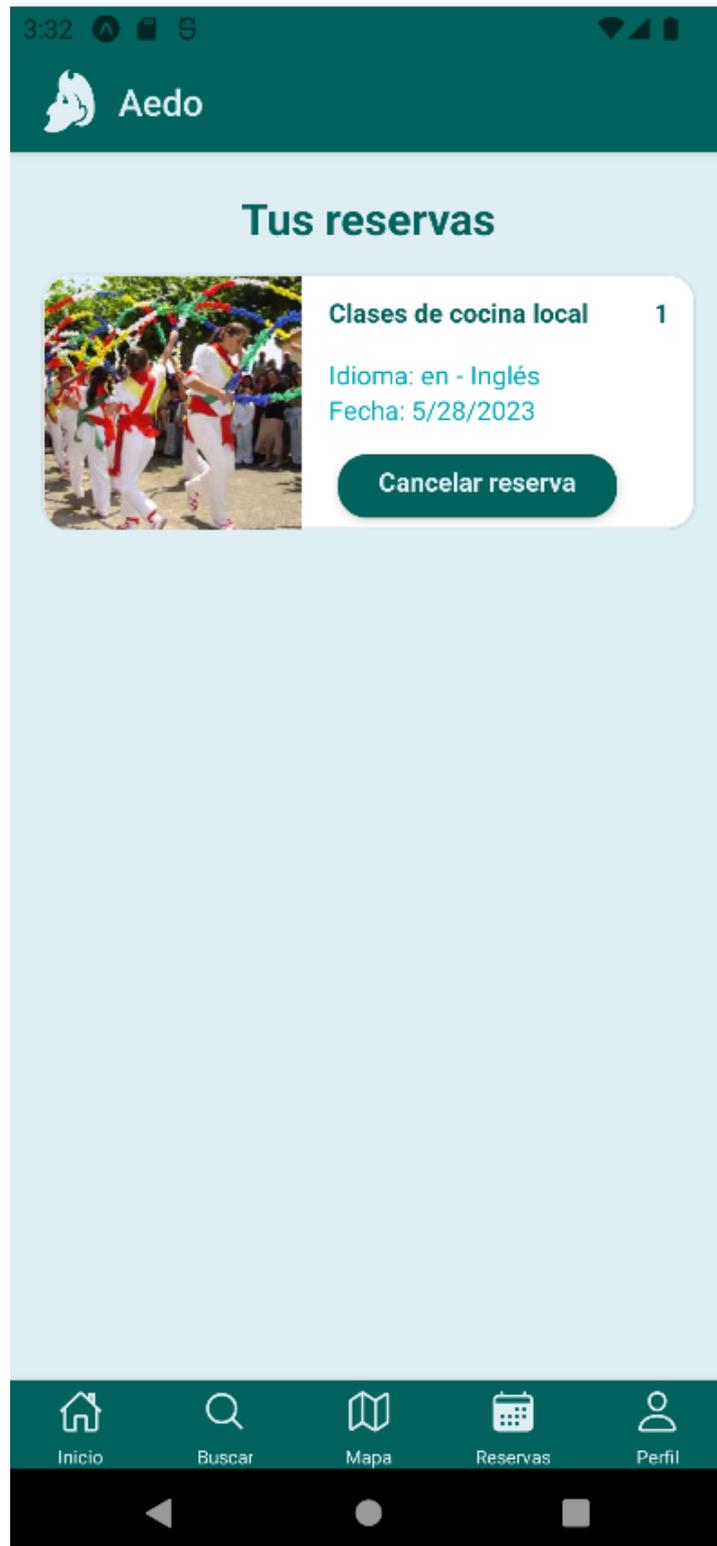
Pantalla del mapa:

En este elemento aparece el mapa y todas las experiencias clasificadas con sus categorías y la ubicación. Se puede acceder a la “pantalla de información de una experiencia” tocando el icono y el nombre en el mapa.



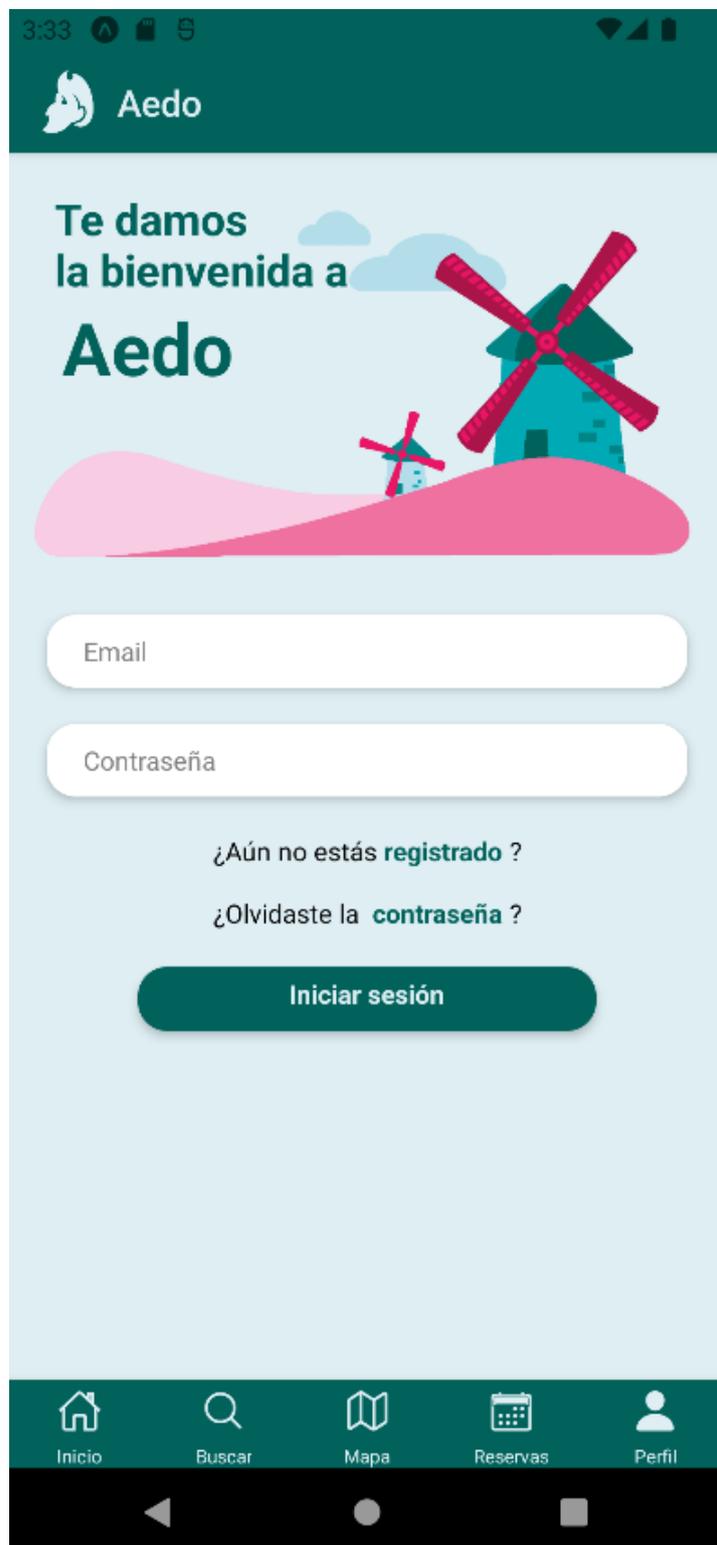
Pantalla de tus reservas.

En esta interfaz aparecen las reservas realizadas por el usuario que ha iniciado sesión y la información de estas como la fecha, el idioma y el número de reservas realizadas. Permite cancelar la reserva también.



Pantalla de inicio de sesión.

Interfaz para iniciar sesión con el correo electrónico y la contraseña. Además, permite registrarse o restablecer la contraseña.



Pantalla de registro.

Incluye el formulario con los datos necesarios para realizar el registro en la aplicación. Los campos marcados con un asterisco son obligatorios. Al registrarse se debe confirmar el correo electrónico.

3:34

← Registrarse

Te damos la bienvenida a
Aedo

*Nombre de Usuario

Nombre

Número de teléfono

*Email

*Contraseña

*Repite la contraseña

*Fecha de nacimiento

Terminar

Pantalla de perfil de usuario.

La pestaña del perfil con el usuario autenticado se convierte en la pantalla de perfil, aquí se pueden crear experiencias o gestionaras, además aparecen las experiencias favoritas del usuario y permite acceder a la modificación de datos personales, correo y contraseña.



Pantalla modificar perfil.

Esta interfaz se utiliza para modificar los datos del usuario facilitados durante el registro en la aplicación. Permite además modificar el correo electrónico y la contraseña.

3:48

← Modifica tu perfil

 **andrac275**
andrac275@gmail.com

Nombre

AndreuMP

Teléfono

666123456

Fecha de nacimiento

1991/06/14

Situación laboral

Trabajando

Nivel de Estudios

Educación Superior

Modificar email

Modificar contraseña



Pantalla crear experiencias.

Interfaz para crear experiencias en la aplicación. Permite introducir un título, una descripción, el aforo máximo, las imágenes, la categoría, el/los idioma/s y la localización. La pantalla de creación de experiencias lleva como paso final a la selección de fechas en las que se ofrece la experiencia.

3:36

← Crear Experiencia

Título*

Llegar a Itaca ...

Descripción

Vuelve a tu hogar y descansa ...

Aforo máximo

Nº máximo de personas (10 por defecto)

Seleccionar imágenes

Seleccionar imágenes

Seleccionador de idiomas*

Select option

Seleccionador de categorías*

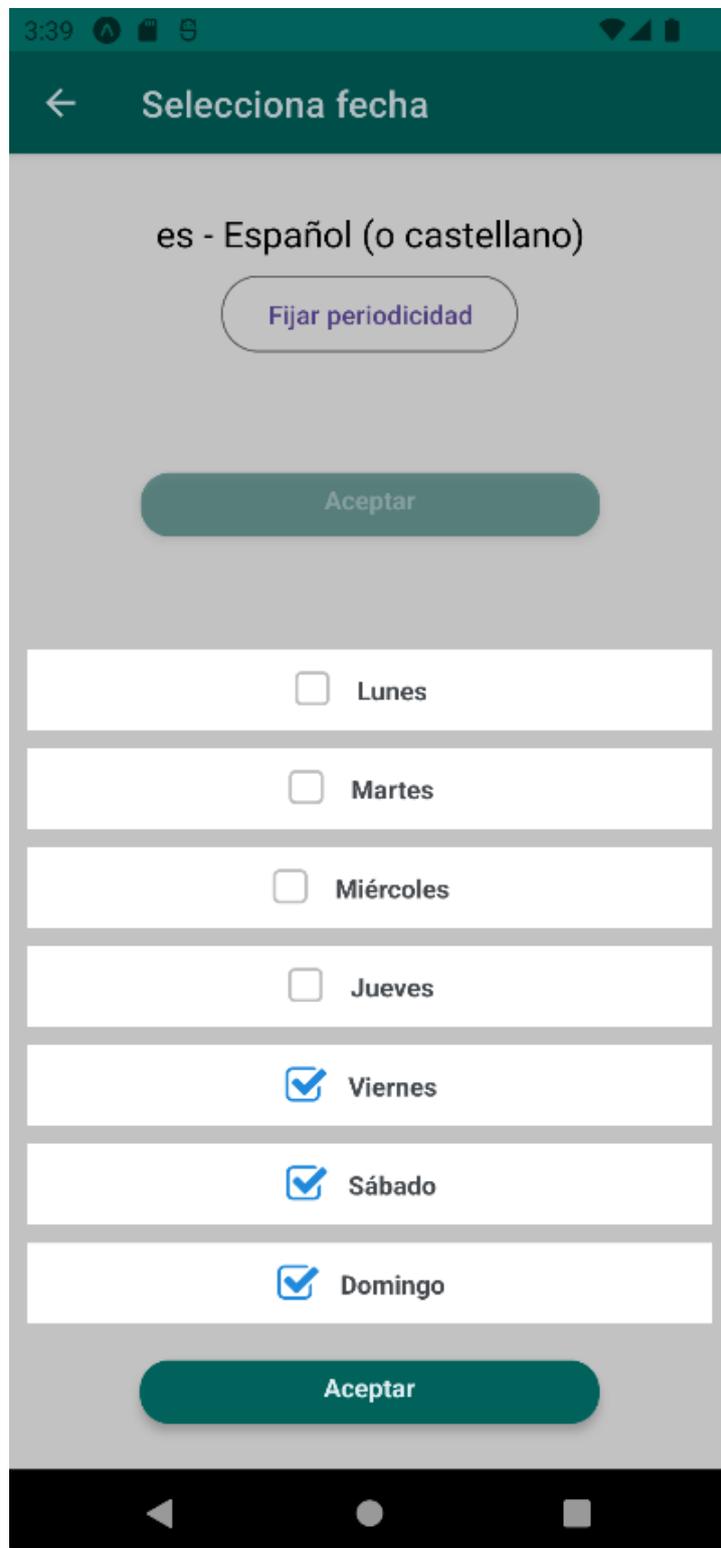
naturaleza ganadería deportivas artísticas

cultural acuáticas adrenalina gastronomía

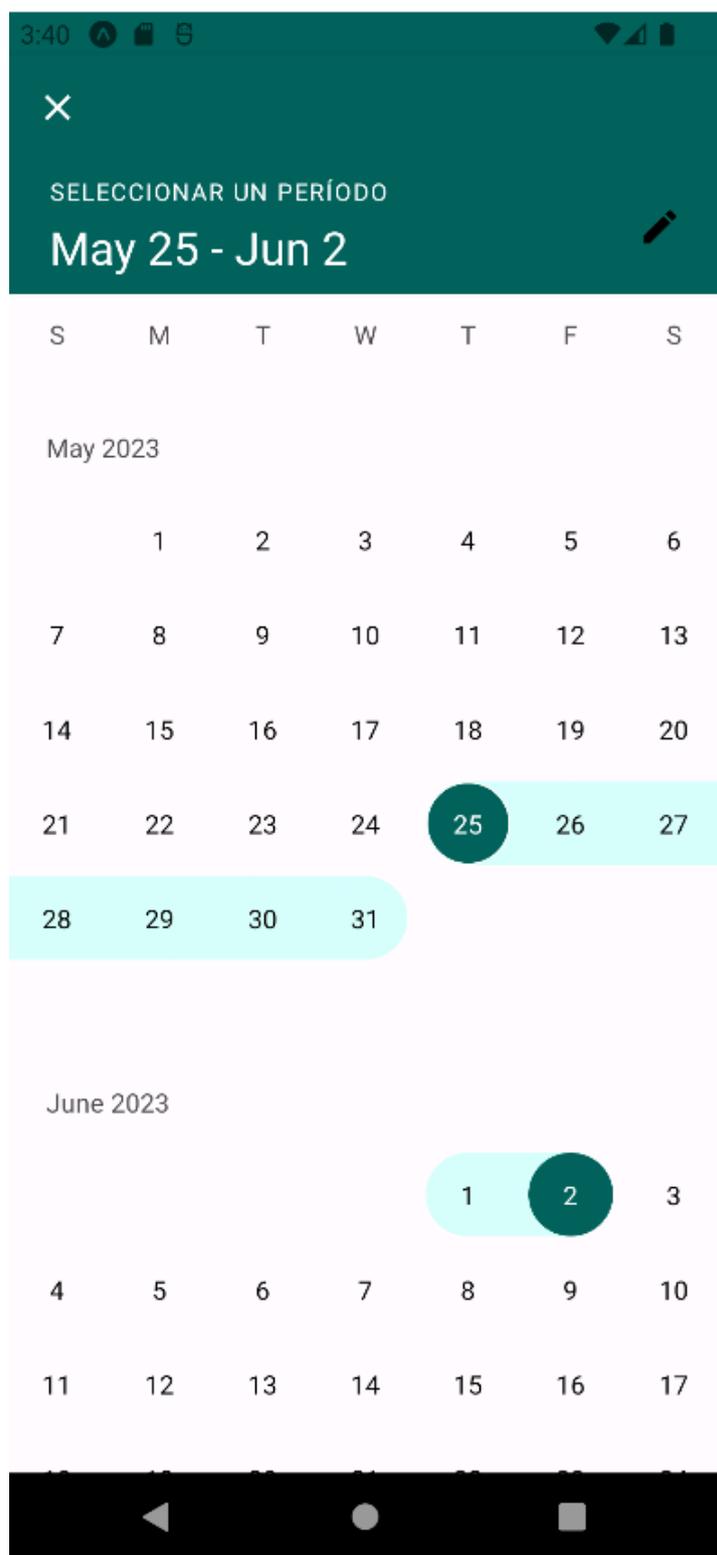
SIGUIENTE

Pantalla periodicidad.

La ventana anterior lleva a la selección de fechas o periodicidad de la experiencia por cada idioma.

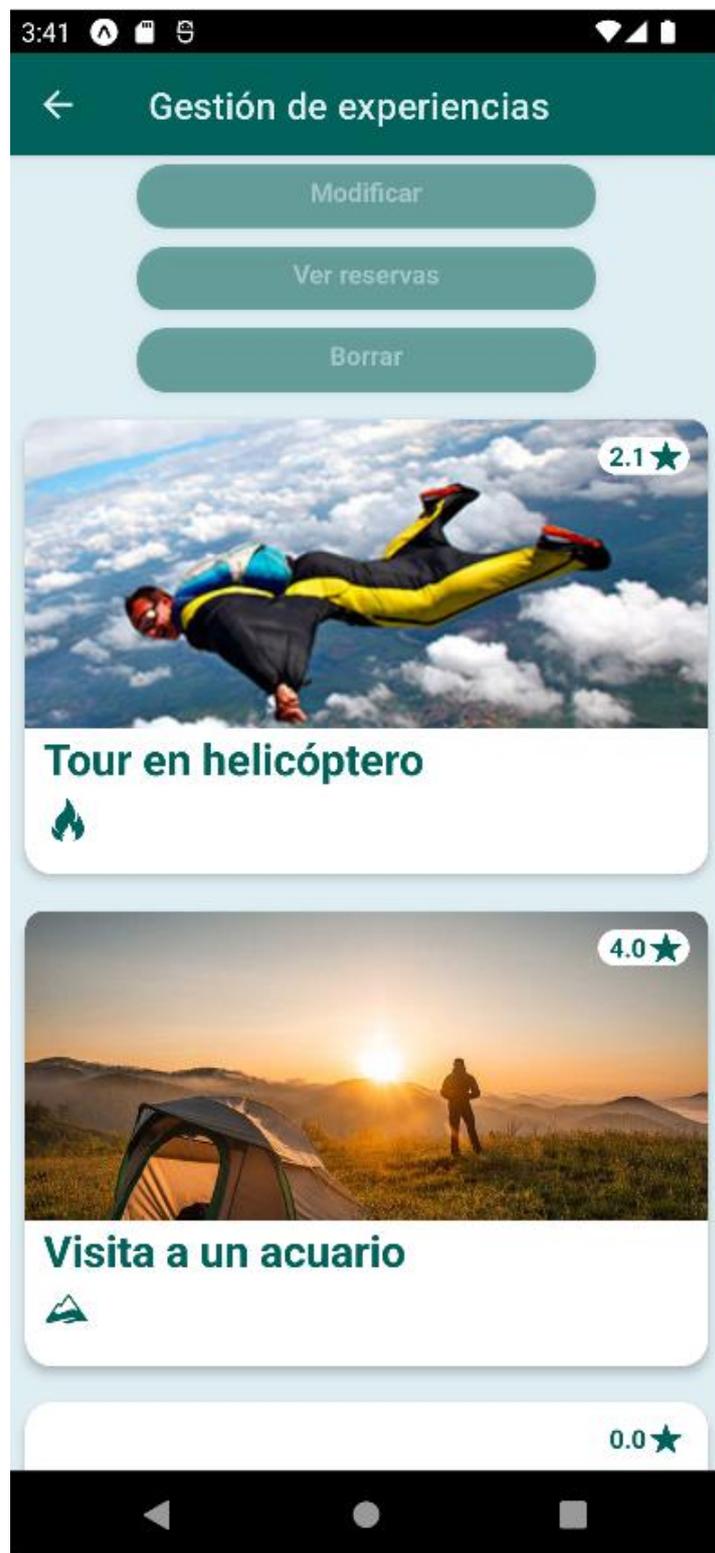


Pantalla selección de rango de fechas.



Pantalla gestión de experiencias.

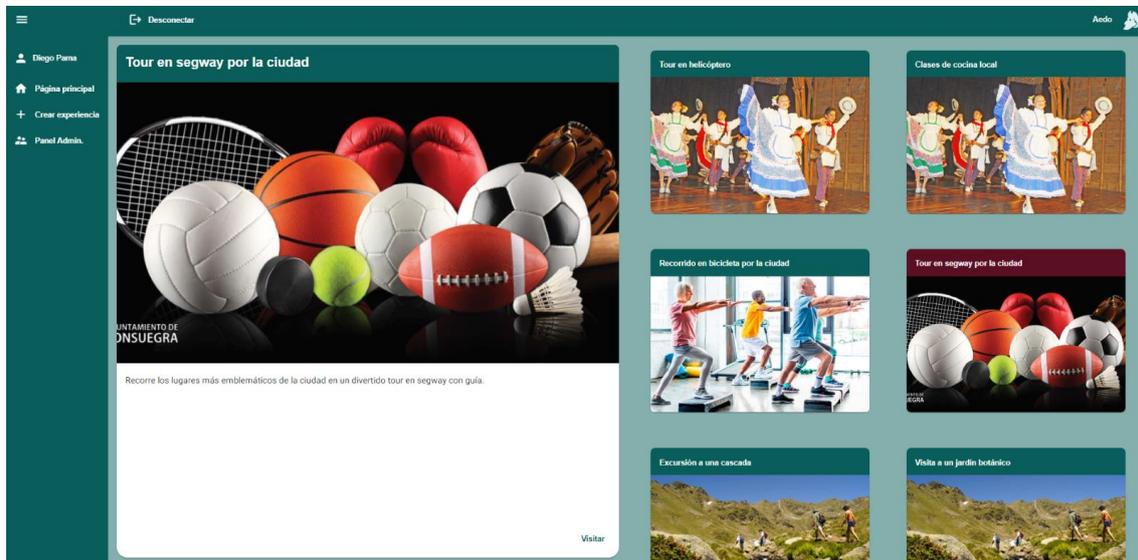
Aquí se presentan las distintas experiencias ofrecidas por el usuario registrado. Permite modificar la información, borrarlas o ver otros usuarios que se han apuntado y para qué fecha lo han hecho.



9.2 Manual de uso web

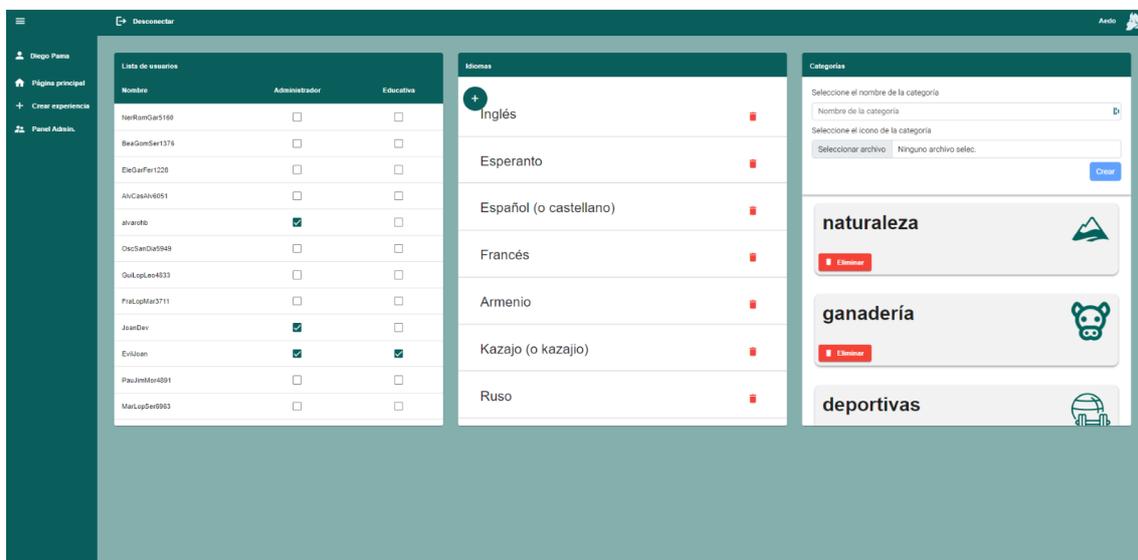
Pantalla principal:

Primera pantalla que ve el usuario al entrar a la web. En ella se muestran las distintas actividades que están disponibles y permite seleccionarlas para ver más información sobre ellas e ir a la página de reserva.



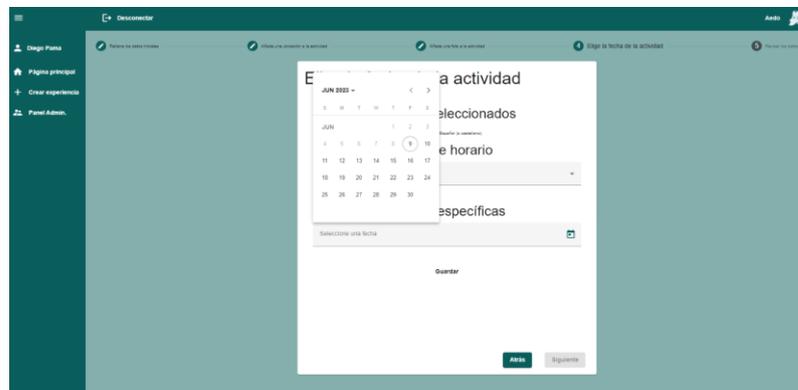
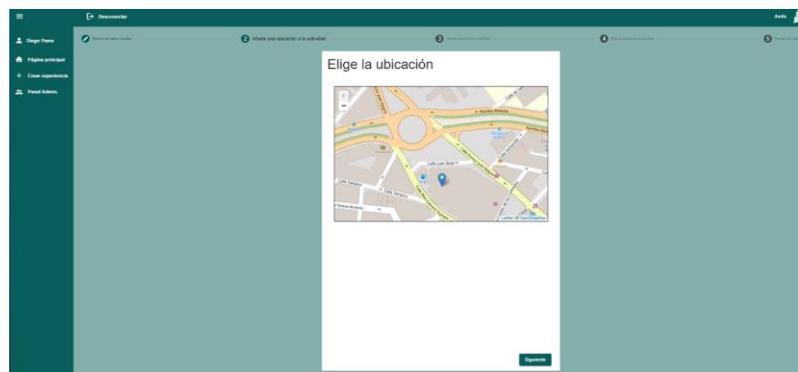
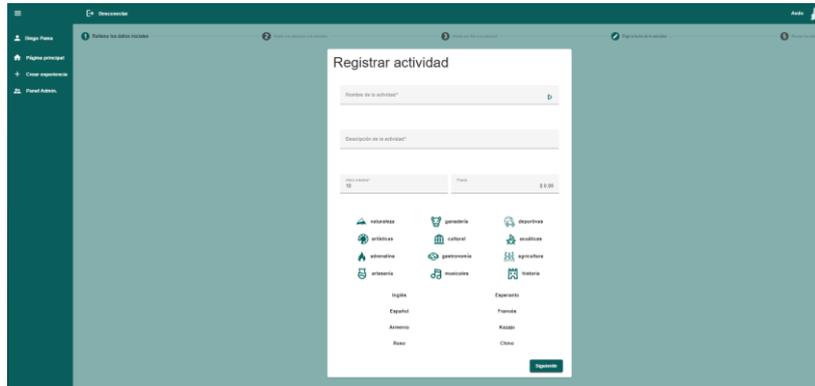
Pantalla de administración:

Pantalla disponible para los usuarios con el rol de administrador, en ella se muestran en distintas funcionalidades de gestión. No es accesible para otros tipos de usuarios.



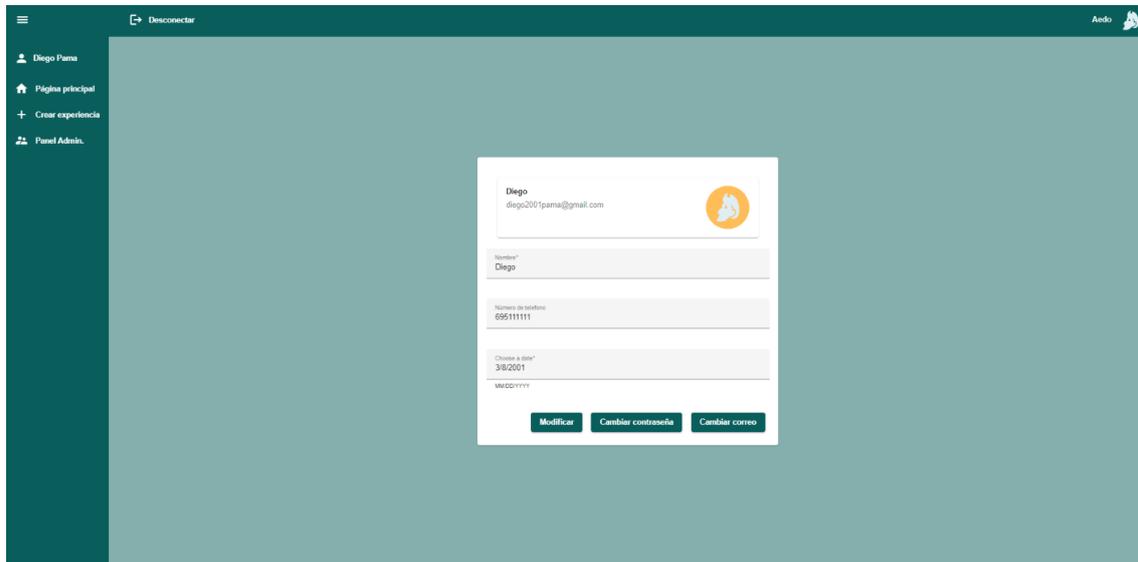
Pantalla de registro de actividades:

Pantalla disponible para los usuarios registrados. Permite introducir un título, una descripción, el aforo máximo, las imágenes, la categoría, el/los idioma/s y la localización.



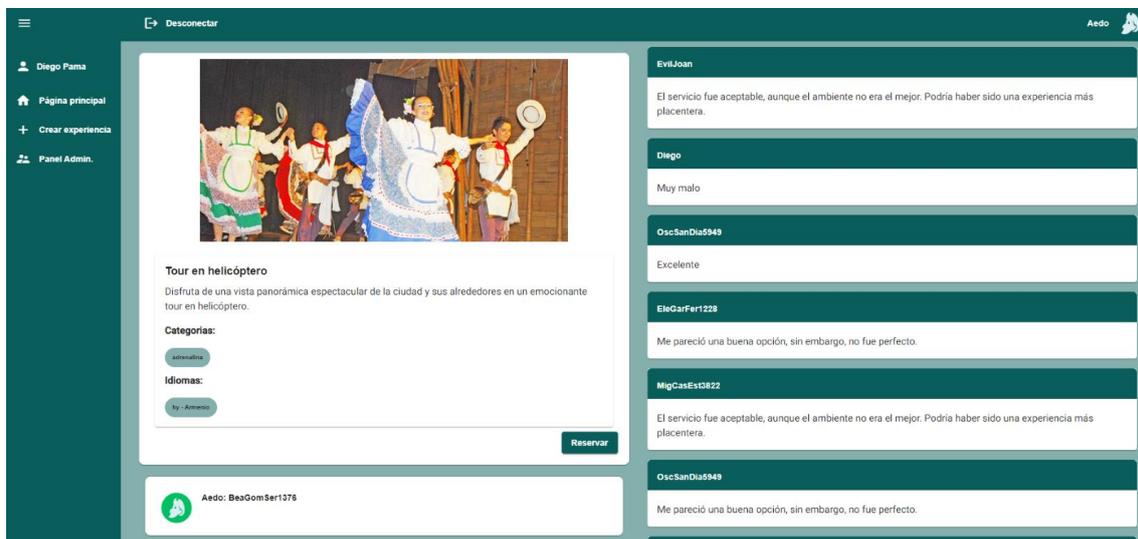
Pantalla de perfil del usuario.

Muestra la información del usuario que está conectado y a su vez permite la modificación de nombre del usuario, su número de teléfono, su fecha de nacimiento, contraseña y correo.



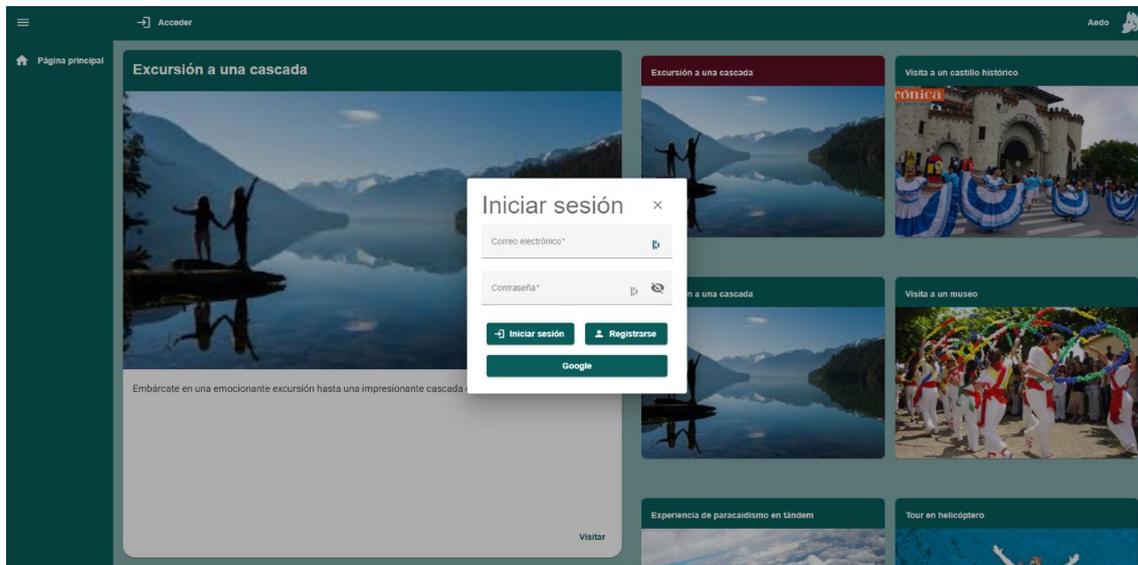
Pantalla de información de una experiencia.

Aparece la información de la experiencia, así como los comentarios dados por otros usuarios de la aplicación. Además, permite al usuario crear una reserva para dicha experiencia.



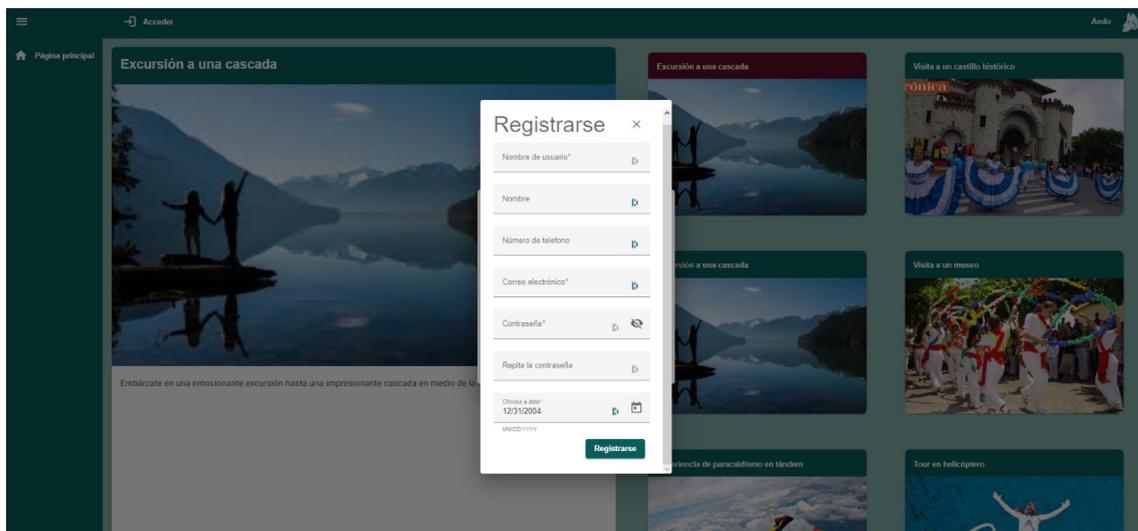
Pantalla de inicio de sesión.

Permite el inicio de sesión de usando la información de un usuario registrado y también da acceso a la pantalla de registro de un usuario.



Pantalla de registro.

Pide al usuario un nombre de usuario, su nombre real, su número de teléfono, correo electrónico, fecha de nacimiento y contraseña con el fin de registrar el usuario en la aplicación.



10. Anexo II: Preguntas de la encuesta al usuario

Se realizaron dos encuestas distintas. Una encuesta era para un usuario/a que fuera a utilizar la aplicación y a realizar una experiencia simple (Experimento 2, ver sección “3.2.1 Experimento 2: Presentar la aplicación a un potencial creador de contenido”); la otra encuesta (Experimento 3, sección “3.2.3 Experimento 3: Presentar la aplicación a un profesor”) iba enfocada al ámbito educativo, permitiendo reservar una experiencia para una clase entera de alumnos. A continuación, se exponen las preguntas generales para un usuario/a estándar que desee realizar una experiencia. La encuesta “educativa” incluye las preguntas generales y dos preguntas específicas.

Preguntas generales:

Pregunta	Tipo de pregunta
Grado de satisfacción con la aplicación	Escala lineal de 1 a 5
¿Qué posibilidades hay de que recomiende Aedo a un conocido?	Escala lineal de 1 a 5
¿Considera que fue sencillo aprender a utilizar la aplicación?	Escala lineal de 1 a 5
Considero que la aplicación me informa en cómo proceder cuando existe algún problema.	Escala lineal de 1 a 5
¿Considera que siempre ha sabido como proceder en lo que deseaba hacer?	Escala lineal de 1 a 5
¿Considera que la interfaz de la aplicación es placentera?	Escala lineal de 1 a 5
¿La navegación de la aplicación le pareció intuitiva?	Escala lineal de 1 a 5
¿Considera que la aplicación cumple con las características esperadas de una aplicación para tal fin?	Escala lineal de 1 a 5
¿La velocidad de respuesta de la aplicación le parece adecuada?	Escala lineal de 1 a 5
¿Qué le ha parecido el proceso de creación de experiencias?	Escala lineal de 1 a 5
¿Qué le ha parecido la administración de experiencias y de usuarios apuntados a las mismas? ¿Considera que la información es suficiente?	Escala lineal de 1 a 5



11. Anexo III: Experimentos realizados



Se realizaron tres experimentos con la aplicación. El primer experimento se realizó durante la Feria de Proyectos de la ETSInf. Se expuso el primer *MVP* de la aplicación. Durante la experiencia en la feria se extrajo mucha información útil y feedback por parte de los asistentes; ya fueran alumnos, profesores o empresas.

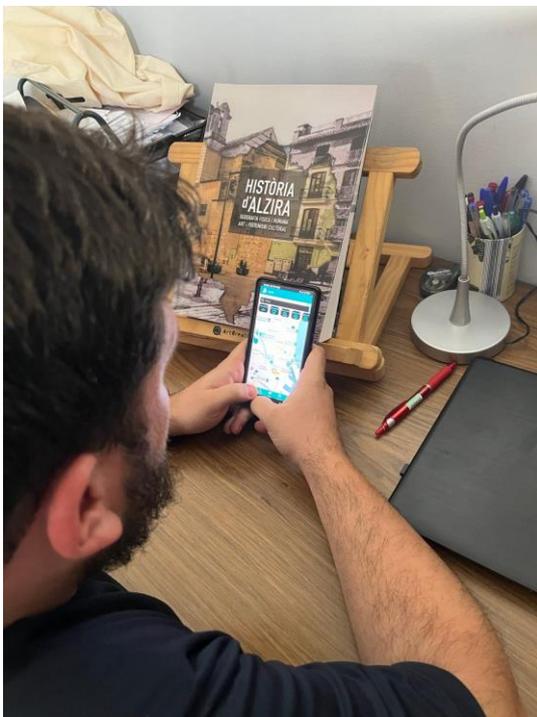




El segundo experimento se realizó fuera del recinto universitario, directamente en Játiva con un agricultor que mostró interés en la aplicación. Se visitaron sus parcelas de campo y el agricultor explicó que la aplicación le parecería útil para enseñar a los usuarios/as de qué manera se trabajan las hortalizas.



Tras la visita del campo, se nos acompañó a la tienda del barrio donde el agricultor comercializa con productos de kilómetro cero.



Para el tercer, y último, experimento se contactó con un profesor de historia y se le planteó la posibilidad de utilizar la aplicación para poder inscribir a su clase de alumnos a las experiencias ofrecidas en la plataforma.

12. Anexo IV: Objetivos de desarrollo sostenible

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

Objetivos de Desarrollo Sostenibles	Alto	Medio	Bajo	No Procede
1. Fin de la pobreza.				X
2. Hambre cero.				X
3. Salud y bienestar.	X			
4. Educación de calidad.		X		
5. Igualdad de género.	X			
6. Agua limpia y saneamiento.				X
7. Energía asequible y no contaminante.				X
8. Trabajo decente y crecimiento económico.				X
9. Industria, innovación e infraestructuras.				X
10. Reducción de las desigualdades.			X	
11. Ciudades y comunidades sostenibles.	X			
12. Producción y consumo responsables.				X
13. Acción por el clima.			X	
14. Vida submarina.			X	
15. Vida de ecosistemas terrestres.			X	
16. Paz, justicia e instituciones sólidas.				X
17. Alianzas para lograr objetivos.				X

Reflexión sobre la relación del TFG/TFM con los ODS y con el/los ODS más relacionados.

La aplicación Aedo está ampliamente alineada con tres importantes Objetivos de Desarrollo Sostenible de las Naciones Unidas: Salud y bienestar (Objetivo 3), Igualdad de género (Objetivo 5) y Ciudades y comunidades sostenibles (Objetivo 11).

- Objetivo 3: Salud y bienestar: Aedo fomenta el compartir experiencias entre usuarios/as, muchas de las experiencias son al aire libre y repercuten en un incremento de la salud, debido a la motivación por parte del equipo de Aedo en incentivar el movimiento de personas a sitios completamente distintos a los que uno puede estar acostumbrado.

- Objetivo 5: Igualdad de género: Aedo no solo permite que cualquier usuario/as, sin importar su género, se convierta en mentor de experiencias, sino que también promueve activamente la igualdad de género en todas las áreas. La aplicación destaca y promociona experiencias y logros

realizados tanto por hombres como por mujeres, brindando visibilidad a personas de todos los géneros. Es por lo que Aedo favorece la igualdad de género. Gracias a la aplicación se pueden compartir, por ejemplo, trabajos realizados tanto por hombres como por mujeres. Además, asociaciones de mujeres o relacionados con la igualdad de género de los diferentes pueblos pueden hacer uso de Aedo para potenciar y publicitar sus actividades culturales.

- Objetivo 11: Ciudades y comunidades sostenibles: La aplicación Aedo tiene como objetivo primordial fomentar la descentralización del turismo en las grandes ciudades y trasladarlo a zonas menos conocidas, pero con la misma capacidad de generar interés en los turistas. Teniendo en cuenta esa premisa, Aedo “obliga” a los usuarios/as consumidores de experiencias a desplazarse. Muchas veces la ubicación de las experiencias serán localidades de la España vacía, ayudando a la economía y las personas locales de dichas poblaciones. Aedo se esfuerza por no solo descentralizar el turismo, sino también por promover la sostenibilidad y el desarrollo equilibrado de las comunidades locales. Además, Aedo fomenta la colaboración con organizaciones locales y pequeñas empresas para ofrecer experiencias auténticas y sostenibles a los usuarios/as. De esta manera, Aedo ayuda a evitar la sobrecarga turística en áreas urbanas y fomenta un turismo más responsable y consciente, que a su vez contribuye a la preservación del patrimonio cultural y natural de las comunidades visitadas.

La aplicación puede estar en menor medida relacionada con los objetivos 4 (Educación de calidad), 10 (reducción de las desigualdades), 13 (Acción por el clima), 14 (Vida submarina) y 15 (Vida de ecosistemas terrestres). La razón de este resquicio reside en que se cumplan o no los objetivos recae en el uso que dan los usuarios/as a la aplicación; por ejemplo: si un usuario/a promueve acciones por el clima, entonces la aplicación se relaciona con el ODS número 13.

En conclusión, la aplicación Aedo está alineada con los Objetivos de Desarrollo Sostenible de las Naciones Unidas al promover la salud y el bienestar a través del intercambio de experiencias al aire libre, fomentar la igualdad de género al permitir que todos los usuarios/as compartan sus trabajos y logros, y contribuir a ciudades y comunidades sostenibles al descentralizar el turismo y apoyar el desarrollo equilibrado y sostenible en áreas menos conocidas. Además, aunque en menor medida, la aplicación Aedo puede tener relación con otros Objetivos de Desarrollo Sostenible, como la educación de calidad, la reducción de las desigualdades, la acción por el clima, la protección de la vida submarina y la conservación de los ecosistemas terrestres, dependiendo del uso que los usuarios/as hagan de la plataforma y de las acciones que promuevan a través de ella.