



UNIVERSITAT DE
BARCELONA

Proyecto final de carrera

GRADO EN INGENIERÍA
INFORMÁTICA

Facultad de Matemáticas e Informática
Universidad de Barcelona

DESARROLLO DE UNA
PÁGINA WEB CENTRADA EN
LA MODA

Autor: Sara San José Gómez

Tutor: Dr. Xavier Baró
Realizado en: Departamento de
Matemáticas e Informática

Barcelona, Junio 2025

Abstract

This work consists of the development of a website called *Serendipia*, whose objective is to offer users a web platform where they can consult all the news about the products of the physical fashion shop. The idea of creating this application has arisen through the need to update, as well as to take the step and be present online.

A whole development process is executed, from the analysis and definition of the data model to be implemented, through the implementation of the API to the development of the interface and the deployment in production. This will involve using the knowledge acquired during the degree, drawing on existing technology, but also learning new tools and emerging technologies.

The implementation objectives include: to be able to offer an intuitive interface that adapts to any screen size and to be able to ensure scalability using Docker containers. The project also foresees a strategy of unit, integration and usability testing.

In short, *Serendipia* aims to become a robust and attractive digital showcase that will boost the brand's competitiveness, but at the same time serve as an exercise to demonstrate competencies and encourage active learning.

Resum

Aquest treball consisteix en el desenvolupament d'una pàgina web anomenada *Serendipia*, l'objectiu de la qual és oferir als usuaris una plataforma web en la qual consultar totes les novetats sobre els productes de la botiga de moda física. La idea de la creació d'aquesta aplicació ha anat sorgint a través de la necessitat d'actualitzar-se, així com de fer el pas i ser present de manera en línia.

S'executa tot un procés de desenvolupament, des de l'anàlisi i la definició del model de dades que s'implementarà, passant per la implementació de la API per a anar a parar en el desenvolupament de la interfície i el desplegament en producció. Per a això s'empraran els coneixements adquirits durant la carrera, recorrent a tecnologia existents, però també aprenent noves eines i tecnologies emergents.

Entre els objectius d'aplicació es troben: poder oferir una interfície intuïtiva i que s'adapti a qualsevol grandària de pantalla i poder garantir l'escalabilitat usant contenidors Docker. El projecte també preveu una estratègia de proves unitàries, d'integració i d'usabilitat.

En definitiva, Serendipia aspira a convertir-se en una vitrina digital robusta i atractiva que impulsi la competitivitat de la marca, però al mateix temps, serveixi d'exercici per a demostrar competències i incitar un aprenentatge actiu.

Resumen

Este trabajo consiste en el desarrollo de una página web llamada *Serendipia*, cuyo objetivo es ofrecer a los usuarios una plataforma web en la que consultar todas las novedades sobre los productos de la tienda de moda física. La idea de la creación de esta aplicación ha ido surgiendo a través de la necesidad de actualizarse, así como de dar el paso y estar presente de manera online.

Se ejecuta todo un proceso de desarrollo, desde el análisis y la definición del modelo de datos que se va a implementar, pasando por la implementación de la API para ir a parar en el desarrollo de la interfaz y el despliegue en producción. Para ello se emplearán los conocimientos adquiridos durante la carrera, recurriendo a tecnología existentes, pero también aprendiendo nuevas herramientas y tecnologías emergentes.

Entre los objetivos de aplicación se encuentran: poder ofrecer una interfaz intuitiva y que se adapte a cualquier tamaño de pantalla y poder garantizar la escalabilidad usando contenedores Docker. El proyecto también prevé una estrategia de pruebas unitarias, de integración y de usabilidad.

En definitiva, Serendipia aspira a convertirse en una vitrina digital robusta y atractiva que impulse la competitividad de la marca, pero al mismo tiempo, sirva de ejercicio para demostrar competencias e incitar un aprendizaje activo.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a todos aquellos que me han acompañado y apoyado a lo largo del desarrollo de este Trabajo de Fin de Grado.

En primer lugar, agradezco profundamente a mi tutor, Xavier, por su dedicación, guía y por todos los valiosos consejos brindados durante el desarrollo de este trabajo. Su experiencia ha sido fundamental para sacarlo adelante.

También quiero agradecer a mis padres, por su apoyo constante, por aportarme tantas ideas y, sobre todo, por su infinita paciencia. Gracias por creer en mí incluso en los momentos más difíciles.

Finalmente, quiero agradecer a mi amigo David, por su ayuda y sus consejos al inicio del proyecto, así como a Óscar, por saber tranquilizarme y hacerme creer que realmente era capaz de lograrlo.

Índice

1. Introducción	1
1.1. Motivación	2
1.2. Objetivos	2
2. Planificación	3
2.1. Organización	3
2.2. Requisitos	9
2.3. Historias de usuario	14
3. Diseño	29
3.1. Diagrama del diseño de navegación	29
3.2. Diseño de la base de datos	30
3.3. Diseño de los mockups	32
4. Implementación	37
4.1. Selección de Tecnologías	37
4.2. Desarrollo del Backend	38
4.3. Desarrollo del Frontend	40
5. Testeo	53
5.1. Testeo de experiencia de usuarios	54
5.2. Testeo de accesibilidad	57

6. Deploy	58
6.1. Deploy Front-end	58
6.2. Deploy Back-end	60
6.3. Deploy Base de Datos	60
7. Conclusiones	61
8. Trabajo futuro	62
9. Bibliografía	64
A. Anexos	70
A.1. Tablas base de datos detalladas	70

1. Introducción

La idea de este proyecto es el desarrollo de una página web centrada en una tienda de ropa y complementos llamada *Serendipia*. La idea de esta aplicación se concibió para responder a las demandas actuales de los consumidores, de manera que se les pueda brindar un espacio donde consultar todas las novedades sin tener que moverse de su casa y poder hacer las compras a través de la página. Durante el desarrollo se van a integrar todos los conocimientos adquiridos a lo largo de la carrera de ingeniería informática.

A lo largo del proyecto se ha profundizado en cada tarea, dedicándole el tiempo necesario a cada una y aprovechando la ocasión para experimentar con nuevas metodologías y tecnologías. Los objetivos principales de este proyecto son participar en el ciclo completo de desarrollo de una página web, abarcando todas sus etapas, y profundizar en el uso de nuevas tecnologías como el lenguaje de programación Dart y su framework Flutter.

En primer lugar, una de las tareas más importantes es hacer una estimación del tiempo dedicado a cada etapa o tarea de desarrollo, así que lo primero que encontraremos será la organización de las tareas y el tiempo dedicado a ellas. Daremos un vistazo a la organización inicial con el tiempo estimado para cada tarea y la organización final con los tiempos reales que se han cumplido. Seguidamente, es muy importante plasmar los requisitos la aplicación debe cumplir, tanto funcionales, como no funcionales. Finalmente, se presentan las historias de usuario que nos permiten saber cómo debe actuar nuestra página web frente a ciertas situaciones y qué criterios de aceptación tenemos.

Una vez visto el bloque de planificación, nos encontraríamos con el bloque de diseño. Donde podremos echar un vistazo al diagrama de navegación de la página web, al diseño de la base de datos y por último a los mock ups *high-fidelity* en los cuales se inspirará el diseño del frontend.

Abordadas estas etapas iniciales, ya podemos pasamos a la fase clave del desarrollo. La implementación. Durante esta parte se detallará el desarrollo seguido para llevar a cabo los diferentes bloques, la base de datos, el backend y el frontend, y como la explicación del despliegue. Una vez definido todo lo que se ha desarrollado, tendremos las pruebas y el testeos hechos sobre los diferentes bloques para comprobar su correcto funcionamiento.

Por último, las conclusiones y trabajo futuro, donde se recoge todo lo que se ha hecho y lo que está previsto para realizar en el futuro.

1.1. Motivación

La idea de crear la web nació con un doble objetivo. Por una parte, ofrecer a todos los clientes la posibilidad de ver los productos sin tener que desplazarse, facilitando un espacio donde ellos puedan consultar información sobre los productos y novedades. Y, por otra, dar a Eva, la dueña del negocio, la posibilidad de mostrar su tienda online de manera fácil y accesible.

Conozco a Eva desde hace mucho tiempo y he tenido la suerte de ver cómo con esfuerzo y dedicación ha acabado montando su propio negocio. Con el tiempo no solo he visto crecer su tienda, sino también su pasión por la moda y el trato cercano con sus clientes. Con la web he querido aportar mi granito de arena, ayudarla con su proyecto y ofrecerle una manera para que su trabajo pueda llegar a más gente.

Además, este proyecto a nivel personal tiene algo muy especial para mí. Tanto el desarrollo móvil como el desarrollo web son algo que realmente me apasiona. Me gusta disfrutar de cada paso, desde el planteamiento del proyecto hasta verlo en pantalla. El desarrollo de la página web ha sido una forma de ayudar a Eva, pero también ha significado para mí la oportunidad para aprender, seguir desarrollando mis habilidades y hacer algo que realmente me gusta y disfruto, en un entorno real.

1.2. Objetivos

El objetivo principal de este proyecto es demostrar y consolidar todo lo aprendido durante el grado. Además de que este proyecto también va a servir de herramienta de aprendizaje.

Los objetivos que me gustaría alcanzar son los siguientes:

1. Aprender nuevas técnicas y enfoques haciendo uso de tecnologías actuales y novedosas, como Dart y su framework Flutter y profundizar en ellos.
2. Ser capaz de cumplir todo el ciclo de desarrollo de un proyecto real, abordando todas sus fases: planificación, diseño, implementación, pruebas y despliegue. Esto supone un reto para mí y a su vez, también, una oportunidad para aplicar todos los conocimientos adquiridos.
3. Mejorar mis habilidades en diseño y usabilidad, siendo capaz de aportar una experiencia de usuario agradable e intuitiva.
4. Ganar experiencia práctica en el desarrollo de aplicaciones web.
5. Ayudar en un proyecto local y significativo, aportando valor a alguien cercano, como en este caso es Eva, mediante mis habilidades técnicas.

Estos objetivos reflejan el enfoque personal del trabajo y rigen el desarrollo del proyecto en todas sus fases. En los siguientes apartados se detallan los requisitos funcionales y no funcionales de la página web, que complementan estos objetivos generales con criterios técnicos.

2. Planificación

En esta sección vamos a ver la planificación general del proyecto. Esta etapa es fundamental para facilitar el proceso de creación y hacerlo de una manera estructurada sin necesidad de malgastar tiempo.

Para empezar encontramos el apartado de organización, donde se detallan las tareas y se muestra tanto la organización inicial planteada como la real.

Luego se van a desglosar los requisitos funcionales y no funcionales de la página web. Estos requisitos servirán de guía para el desarrollo, asegurando que la página web que estamos creando responda tanto a las necesidades del usuario como a los estándares de calidad esperados.

El último paso serán las historias de usuario que sirven para entender los requisitos desde el punto de vista del usuario final. Estas historias son muy útiles para conocer cómo se debe comportar la aplicación en diferentes situaciones y qué funcionalidades son críticas.

2.1. Organización

Para poder desarrollar el proyecto de una manera cómoda y estructurada, utilizamos diagramas de Gantt. De esta manera, se puede organizar el tiempo y se puede plasmar la estimación del tiempo dedicado a cada tarea visualmente.

A continuación se detallan las tareas a realizar, las cuales están mencionadas en ambos diagramas, el de las estimaciones y el final:

- **Desarrollo del modelo entidad-relación:** diseño lógico de la base de datos, donde se incluyen entidades, sus atributos y claves primarias o foráneas y relaciones que reflejen las necesidades del sistema.
- **Desarrollo mock-up/sketch página web:** creación de los diseños de la interfaz para definir la estructura de la página web y tener un esquema que sirva de guía para el desarrollo del frontend.

- **Desarrollo diagrama de navegación:** representación visual del flujo de navegación entre las distintas páginas y secciones del sitio web.
- **Creación de la base de datos:** implementación de la base de datos, creando las tablas definidas en el modelo entidad-relación.
- **Creación de las peticiones a la base de datos:** desarrollo de los endpoints que permiten insertar, consultar, actualizar y eliminar datos en la base de datos, gestionados a través de funciones en Python.
- **Desarrollo y creación del frontend:** implementación de la interfaz de usuario utilizando el Framework de Flutter intentando respetar el diseño definido previamente.
- **Testeo de la página web:** verificación del correcto funcionamiento de todas las funcionalidades implementadas para la página web, intentando detectar errores o comportamientos inesperados.
- **Arreglar errores surgidos durante el testeo:** corrección de bugs o mejoras necesarias identificadas durante la fase de testeo previa.
- **Deploy de la página web:** publicación del sitio web en un servidor o plataforma online para su acceso público o privado.
- **Redacción de la memoria:** redacción del documento que detalla el desarrollo del proyecto, incluyendo decisiones técnicas, diseño, implementación y conclusiones.
- **Revisión de la memoria:** lectura crítica y corrección del documento final para asegurar su calidad, coherencia y claridad antes de la entrega.

Para la organización del proyecto, se ha adoptado un enfoque basado en tareas, estructurado en forma de sprints por tarea, mientras que la estimación del trabajo por tarea se ha realizado en días.

Se ha considerado cada tarea como un sprint independiente con objetivos concretos. Esto ha permitido centrarse en un solo componente a la vez (por ejemplo, el modelo entidad-relación, el desarrollo del frontend o el testeo), facilitando un desarrollo ordenado.

Para hacer la estimación se ha seguido la técnica de estimación *three-point estimation* que se basa en hacer diferentes estimaciones, una estimación optimista, una pesimista y la más probable que realmente suceda. De manera que con estas tres estimaciones se hace un cálculo para estimar el tiempo que una tarea puede llevar.

La estimación se calcula a partir de la siguiente fórmula:

$$E = \frac{O + 4M + P}{6}$$

Donde:

- *O*: Estimación optimista
- *M*: Estimación más probable
- *P*: Estimación pesimista
- *E*: Estimación esperada

A continuación, en el cuadro 1, se puede visualizar la tabla con las estimaciones. La más probable, la pesimista, la optimista y la estimación esperada.

Tarea	M (Días)	O (Días)	P (Días)	E (Días)
Desarrollo del modelo entidad-relación	4	2	6	4.0
Mock-up/sketch página web	7	4	10	6.8
Diagrama de navegación	4	2	6	4.0
Creación de la base de datos	4	2	6	4.0
Peticiones a la base de datos	10	6	14	10.0
Desarrollo y creación del frontend	28	21	35	27.3
Testeo de la página web	4	2	6	4.0
Arreglo de errores tras testeo	7	3	10	6.3
Deploy de la página web	14	5	20	12.2
Redacción de la memoria	10	6	14	10.0
Revisión de la memoria	4	2	6	4.0

Cuadro 1: Estimaciones aplicando la técnica de los Tres Puntos

A continuación, se presentan los diagramas de Gantt correspondientes a la planificación del proyecto. En ellos se reflejan las distintas tareas planificadas, que ya hemos mencionado previamente, su duración estimada en días y el orden en que se deben realizar. Permiten visualizar de forma clara las fases del desarrollo, así como la relación y posible solapamiento entre algunas actividades.

La elaboración del diagrama se ha basado en las estimaciones obtenidas mediante la técnica *Three-Point Estimation*, permitiendo definir un marco temporal

realista para cada tarea. Podemos observar dicho diagrama en la figura 1. Pese a haber obtenido valores decimales en los cálculos de las estimaciones, estos se han redondeado al alza.

En la figura 2 podemos ver tanto la estimación como el desarrollo real. En general se ha mantenido un buen seguimiento, sobre todo en las fases iniciales donde algunas de las tareas se han completado en el periodo previsto. Sin embargo, algunas tareas posteriores han experimentado retrasos, como el desarrollo del backend, donde el retraso realmente no ha sido tan grande, sino que durante el tiempo de desarrollo del frontend se han ido haciendo cambios mínimos en el backend que eran necesarios para el correcto funcionamiento. En el caso del frontend, me ha llevado más tiempo del esperado realizarlo; había mucho trabajo y creo que mi estimación del tiempo más probable era algo optimista. Debido a estos dos retrasos principales, el diagrama está desplazado hacia la derecha de manera que todas las tareas posteriores en términos de fechas se han demorado.

A continuación una breve explicación de la leyenda de los diagramas de Gantt.

- Las **zonas a rayas lilas** corresponden a la estimación inicial planificada para cada tarea, en días.
- Las **zonas en color lila sólido** corresponden a las tareas que han sido completadas dentro del plazo estimado
- Las **zonas en color naranja** corresponden a las tareas que se han completado fuera del plazo.

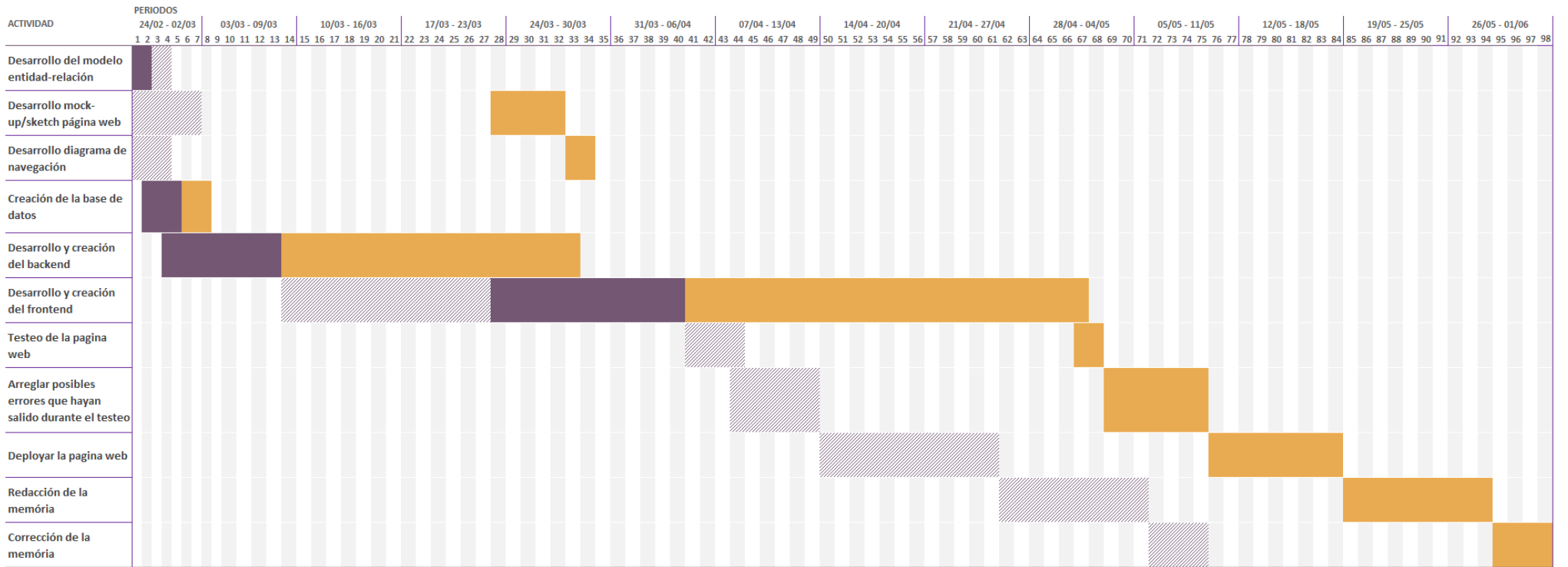


Figura 2: Diagrama de Gantt de la planificación final

2.2. Requisitos

Requisitos funcionales

- Registro de usuarios

- El sistema debe permitir a los usuarios registrarse con nombre, apellidos, fecha de nacimiento, correo electrónico y contraseña.

- Inicio de sesión de usuarios

- Los usuarios deben poder iniciar sesión utilizando sus credenciales.
- En caso de no poder acceder a la cuenta, el sistema debe permitir a los usuarios recuperar o cambiar la contraseña mediante un enlace enviado al correo electrónico del usuario.

- Perfil de usuario

- Los usuarios registrados deben poder acceder a su perfil e información personal.
- Los usuarios registrados deben poder gestionar su perfil, editando su información personal (nombre, apellidos, correo electrónico, foto de perfil y contraseña).
- Los usuarios registrados deben poder acceder a su historial de pedidos.
- Los usuarios registrados pueden acceder a los productos que han sido guardados previamente (favoritos).

- Visualización del catálogo de productos

- Los usuarios podrán acceder a un catálogo de productos, ya filtrado por categoría y tipo.
- Los usuarios podrán visualizar el panel de filtros y utilizarlos.
- Los usuarios podrán visualizar los productos y acceder al detalle de cada uno.

- Visualización de productos

- Los usuarios podrán acceder a una pantalla con los detalles del producto, donde podrán ver los datos del producto (nombre, referencia, descripción - si tiene -, precio y tallas disponibles).
- Los usuarios deben poder elegir la talla del producto para añadir al carrito.

- Gestión del catálogo de productos

- El sistema debe permitir al administrador y al gestor dar de alta productos con detalles. Será necesario proporcionar el nombre del producto, el código o número de referencia, una imagen, la categoría del producto, el tipo de producto, la visibilidad del producto (oculto, visible), el precio, el color y las tallas con su número de existencias.
- El sistema debe permitir al administrador y gestor modificar la visibilidad de un producto; este puede estar oculto o visible.
- Debe haber una sección para cada categoría de productos. Tendremos las siguientes categorías:
 - **Ropa:** camisetas, camisa, blusa, top, polo, falda, falda-pantalón, pantalones, vaqueros, pantalones cortos, vestido, mono, sudadera, suéter, chaqueta, abrigo, blazer, chaleco, bañador.
 - **Zapatos:** botas, botines, zapatillas, tacones, sandalias, bailarinas, bambas, manolequinas, zuecos.
 - **Accesorios:** bolso, mochila, gorro, bufanda, pañuelo, bandana, diadema, guantes, sombrero, cinturones, gafas de sol.
 - **Joyería:** collar, pulsera, brazalete, tobillera, anillo, pendientes, piercings.

- **Búsqueda y filtrado de productos**

- Los usuarios registrados y los no registrados deben poder hacer búsquedas de productos en el catálogo utilizando palabras clave, en nuestro caso el nombre del producto.
- El sistema permitirá utilizar los filtros conjuntamente con la búsqueda o individualmente.
- El sistema ofrecerá opciones para ordenar por precio (ordenar de menor a mayor precio), popularidad (ordenar por los más recomendados), fecha de creación (productos que han sido creados más recientemente, o a la inversa, los que son más antiguos).

- **Carrito de compras**

- Los usuarios deben poder añadir productos al carrito.
- Los productos del carrito deben de ser editables, los usuarios pueden modificar la cantidad de los productos añadidos, eliminar elementos y actualizar el total.
- El sistema debe mostrar el resumen del carrito con el total de la compra, más impuestos.
- El sistema tiene que dar la opción de proceder con la compra de los productos del carrito.
- Antes de iniciar el proceso a través de la pasarela de pago el sistema debe mostrar una ventana emergente que permita al comprador leer las condiciones de compra.

- Debe haber una casilla de verificación, desmarcada por defecto, que indique *He leído y acepto las condiciones de compra* antes de proceder con el pago.
- El sistema no permitirá seguir con el pago si no se han aceptado las condiciones.

- Roles de usuario

Administrador:

- Puede gestionar el catálogo completo de productos: crear, editar y eliminar productos.
- Tiene acceso completo al panel de administración (gestión de pedidos, usuarios, comentarios y valoraciones).
- Tiene permisos para manejar cupones de descuento y promociones.
- Puede gestionar la configuración general de la tienda, como los ajustes, las opciones de pago y las políticas.
- Puede establecer roles y permisos de otros usuarios.
- Tiene permiso para crear o invitar a nuevos gestores.

Gestor:

- Puede gestionar el catálogo completo de productos: crear, editar y eliminar productos.
- Tiene permisos para manejar cupones de descuento y promociones.
- Tiene acceso para gestionar pedidos y visualizar el estado.
- Puede crear, editar y eliminar conjuntos de prendas y accesorios.
- Puede moderar comentarios y valoraciones de los usuarios.

Usuario registrado:

- Puede registrarse e iniciar sesión.
- Puede tener una cuenta con información personal.
- Puede ver su historial de pedidos y el estado de estos.
- Puede guardar productos para visualizarlos posteriormente.
- Puede dejar comentarios y valoraciones en los productos que ha comprado.
- Puede visualizar conjuntos.
- Puede comprar productos individuales y/o conjuntos de productos (prendas y accesorios).

Usuario no registrado:

- Puede navegar por el catálogo de productos y de conjuntos.

- Puede buscar y filtrar productos.
- Puede añadir productos al carrito, pero para finalizar la compra debe registrarse o iniciar sesión.

- **Creación y visualización de conjuntos**

- El administrador y el gestor deben tener la capacidad de crear conjuntos. Para crear un conjunto, tendrán que introducir el nombre, la descripción (opcional), la visibilidad del conjunto, el precio y una imagen del producto. En la misma pantalla se mostrarán los productos ya existentes y habrá que seleccionar los productos que componen el conjunto.
- El conjunto tendrá un nombre, un precio y una descripción, esta última es opcional.
- El administrador y el gestor deben poder crear, editar o eliminar los conjuntos.
- Los usuarios deben poder ver los conjuntos completos en una sección de la página web dedicada a ello.
- Cada conjunto debe mostrar la combinación de prendas y accesorios con imágenes de cada artículo.
- Los usuarios deben tener la opción de añadir el conjunto completo al carrito de compra y el sistema comprobará el stock de cada producto, en caso de no tener stock, se comunicará al usuario con un mensaje pop up.

- **Sistema de descuentos y promociones**

- Un usuario puede identificarse como VIP (descuento de un 20%).
- El administrador y gestor deben poder crear y gestionar cupones de descuento o promociones especiales dirigidas a los grupos de usuarios.
- Los usuarios registrados pueden aplicar códigos de descuento durante el proceso de pago.

- **Pasarela de pago**

- El sistema debe integrar una pasarela de pago para realizar compras seguras.
- Debe haber diferentes opciones de pago (tarjeta de crédito/débito, PayPal, transferencia bancaria).
- El sistema debe verificar la disponibilidad de stock antes de confirmar la compra.
- El sistema debe enviar una confirmación del pedido al correo electrónico del usuario registrado indicando los productos comprados, el precio de cada uno, y el subtotal de la compra.

- **Soporte al cliente**

- Debe haber una sección de *Contacto* o *Ayuda*, donde los usuarios puedan enviar consultas y/o dudas.
- **Publicación de contenido, comentarios y reacciones**
- Los usuarios registrados deben poder crear, y eliminar comentarios en los productos.
 - Los usuarios registrados deben poder dar una calificación de 1 a 5 estrellas sobre los productos, una única vez.

Requisitos no funcionales

- **Multilinguaje**
- La página debe estar disponible en diferentes lenguajes:
 - Castellano
 - Catalán
 - Inglés
- **Diseño responsive**
- La página debe adaptarse a diferentes dispositivos, móviles, tablets, ordenadores.
- **Seguridad**
- La página debe ser segura.
 - HTTPS.
- **Rendimiento**
- La página debe ofrecer búsquedas rápidas.
- **Compatibilidad con navegadores y dispositivos**
- La página debe ser compatible con diferentes navegadores como Firefox, Chrome, Safari, Opera...
 - La página debe ser compatible con todo tipo de dispositivos, independientemente del sistema operativo o las tecnologías que utilicen.

2.3. Histórias de usuario

A continuación tenemos el diagrama de casos de uso, donde podemos tener una visión general de las interacciones entre los diferentes actores. El diagrama permite identificar las funcionalidades que ofrece la aplicación y los distintos roles que intervienen. Los casos de uso que están representados sirven como base para el diseño funcional del sistema.

Seguidamente, están detalladas las historias de usuario correspondientes a cada uno de estos casos de uso. Las historias de usuario permiten describir, desde la perspectiva del actor, los objetivos concretos que se pretenden alcanzar mediante el uso del sistema.

En la figura, 3, se presenta el diagrama de casos de uso de la página web. En él se identifican tres actores principales:

- **Usuario no registrado:** puede acceder a funcionalidades básicas como visualizar el catálogo, buscar y filtrar productos, ver detalles de productos y conjuntos, añadir productos al carrito, registrarse o iniciar sesión.
- **Usuario registrado:** incluye tanto a clientes normales como a clientes VIP. Además de las funcionalidades anteriores, puede acceder a su perfil, consultar el historial de pedidos, realizar compras, guardar productos como favoritos y visualizarlos posteriormente, entre otras funcionalidades.
- **Administrador:** tiene acceso a funcionalidades exclusivas de gestión, como la creación y edición de productos y conjuntos, la visualización y gestión del estado de los pedidos, la moderación de comentarios y la administración de todos los usuarios del sistema.

La figura 3, también permite visualizar claramente la separación de responsabilidades y accesos en función del rol del actor. También refleja cómo cada tipo de usuario interactúa con el sistema, lo que facilita la validación del cumplimiento de los requisitos funcionales definidos en las etapas iniciales del proyecto.

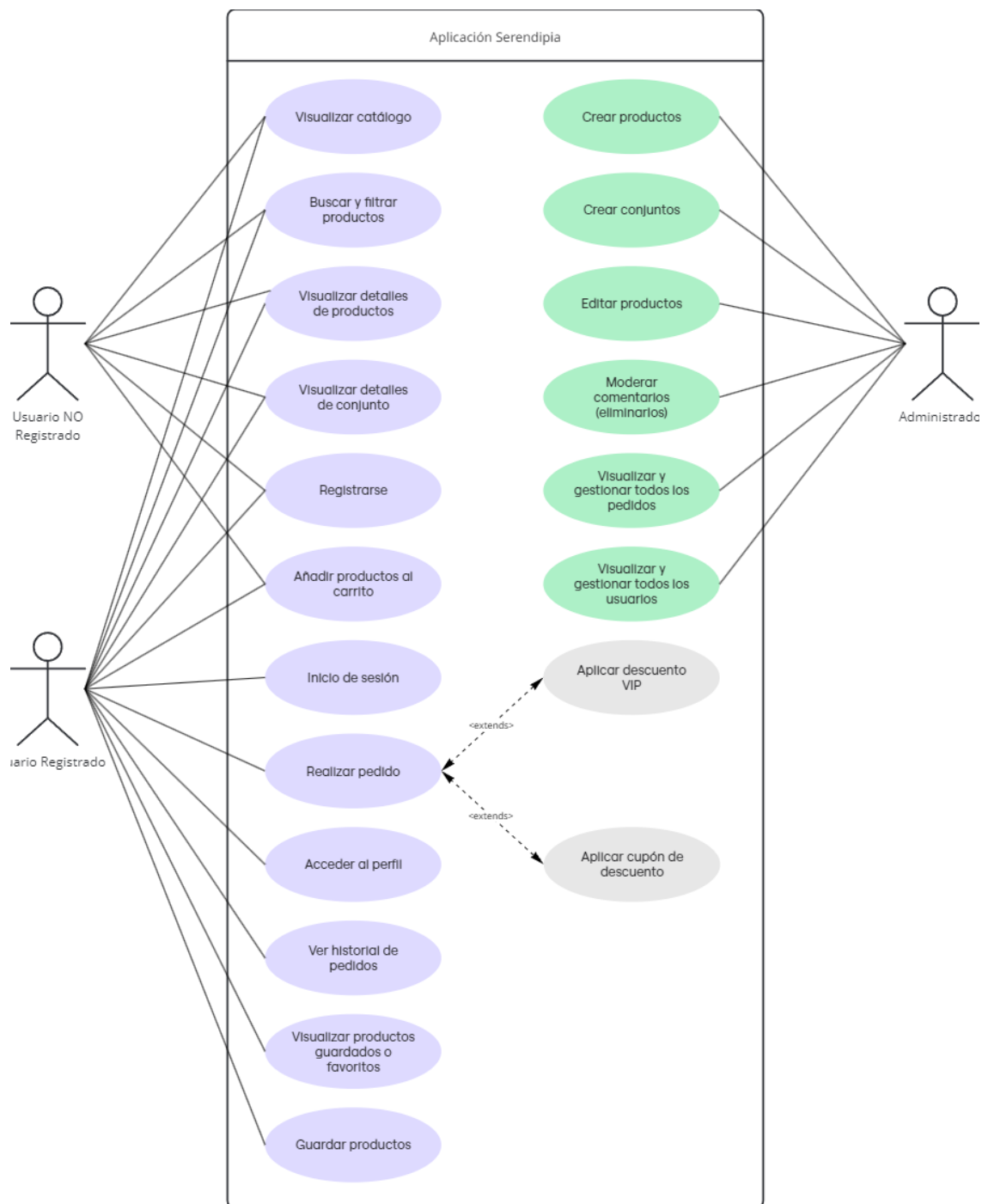


Figura 3: Diagrama de casos de uso

En el diagrama se ha representado la funcionalidad exclusiva de los clientes VIP mediante un caso de uso extendido denominado *Aplicar descuento VIP*. Este no aparece vinculado directamente a un actor específico, ya que todos los usuarios registrados comparten el mismo actor en el modelo. Sin embargo, solo aquellos que cuentan con ser clientes VIP pueden ejecutar esta funcionalidad durante el proceso de compra.

Por este motivo, se ha utilizado una relación de tipo «*extend*» desde el caso de uso Realizar pedido”, indicando que el descuento VIP es una funcionalidad opcional, condicionada a una propiedad específica del usuario.

Una vez visto el diagrama para tener una visión general de las interacciones, tenemos las historias de usuario redactadas con su descripción y criterios de aceptación.

UC01 Visualizar la pantalla de inicio	
Descripción	Como usuario sin registrar quiero acceder a la pantalla de inicio.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de inicio. - El usuario puede visualizar los productos recomendados. - El usuario puede visualizar los datos de contacto y ubicación.

Cuadro 2: Historia de usuario UC01

UC02 Crear una cuenta de usuario	
Descripción	Como nuevo usuario, quiero crear una cuenta en la aplicación introduciendo mi información personal (nombre, apellidos, fecha de nacimiento, correo electrónico y contraseña), para poder acceder a todas las funcionalidades de la web.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de registro. - El usuario rellena todos los campos, ya que son obligatorios. - El usuario pulsa el botón '<i>Registrarse</i>'. - Si los datos son válidos (el correo tiene un formato adecuado, no está registrado en la base de datos, la fecha de nacimiento indica una edad válida y la contraseña cumple con la longitud requerida), la cuenta se crea con éxito, se informa al usuario y se le redirige a la página de inicio de sesión. - Si algún dato es inválido o algún campo está sin rellenar, se muestra un mensaje informando del error ocurrido.

Cuadro 3: Historia de usuario UC02

UC03 Iniciar sesión	
Descripción	Como usuario registrado, quiero iniciar sesión en la aplicación utilizando mis credenciales, para acceder tanto a mis datos como a las diferentes funcionalidades que proporciona la página web.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de inicio de sesión. - Introduce el correo electrónico y la contraseña asociados a su cuenta. - Pulsa el botón '<i>Iniciar sesión</i>'. - Si las credenciales son correctas, el usuario es redirigido a la pantalla de inicio. - Si alguna de las credenciales es incorrecta, o algún campo está sin rellenar, se muestra un mensaje informando del error.

Cuadro 4: Historia de usuario UC03

UC04 Restaurar contraseña	
Descripción	Como usuario registrado, quiero restaurar mi contraseña porque la he olvidado, para poder recuperar el acceso a mi cuenta.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de inicio de sesión. - Pulsa el botón '<i>¿Has olvidado la contraseña?</i>'. - Se muestra un mensaje que solicita el correo electrónico asociado a la cuenta, con el fin de enviar un enlace de restauración. - El usuario introduce el correo con el que está registrado y recibe un mensaje de recuperación en su bandeja de entrada.

Cuadro 5: Historia de usuario UC04

UC05 Perfil de usuario	
Descripción	Como usuario registrado, quiero poder acceder a mi perfil para consultar y editar mi información personal, con el fin de mantener mis datos actualizados.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de perfil. - Puede consultar sus datos personales. - Puede modificar su nombre, apellidos, fecha de nacimiento, teléfono, dirección, foto de perfil, correo electrónico y contraseña.

Cuadro 6: Historia de usuario UC05

UC06 Productos guardados	
Descripción	Como usuario registrado, quiero poder acceder a los productos que he guardado desde mi perfil, para poder consultarlos más adelante.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de perfil. - Accede al apartado de productos guardados. - Puede visualizar los productos que ha guardado previamente. - En caso de no haber ningún producto guardado, se muestra un mensaje informativo indicando que no hay productos disponibles.

Cuadro 7: Historia de usuario UC06

UC07 Historial de pedidos	
Descripción	Como usuario registrado, quiero poder consultar el historial de pedidos para revisar compras anteriores y conocer su estado.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de perfil. - Accede al apartado de historial de pedidos. - Puede visualizar los pedidos realizados previamente y el estado de cada uno. - En caso de no haber ningún pedido previo, se muestra un mensaje informativo. - Si el estado del pedido es <i>Creado</i> o <i>Procesándolo</i>, el usuario puede cancelar el pedido.

Cuadro 8: Historia de usuario UC07

UC08 Visualización del catálogo de productos	
Descripción	Como usuario (registrado o no registrado), quiero poder visualizar el catálogo de productos para explorar las prendas disponibles por categoría y tipo.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página del catálogo. - Puede navegar por las diferentes categorías y tipos de productos. - Puede aplicar filtros según sus preferencias. - El sistema muestra los productos disponibles según los criterios seleccionados. - Si no hay productos coincidentes o se produce un error al realizar la petición, se muestra un mensaje informativo.

Cuadro 9: Historia de usuario UC08

UC09 Visualización del catálogo de conjuntos	
Descripción	Como usuario (registrado o no registrado), quiero poder visualizar el catálogo de conjuntos para explorar los conjuntos disponibles.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la sección de conjuntos. - Visualiza una lista de conjuntos. - Si algún producto del conjunto no tiene stock, el sistema informa al usuario mediante un mensaje emergente.

Cuadro 10: Historia de usuario UC09

UC10 Visualización de detalles del producto	
Descripción	Como usuario, quiero poder visualizar productos en los que estoy interesado.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la sección detalles del producto. - Visualiza el producto con su imagen, nombre, valoración y descripción (en el caso de estas últimas, si las tiene). - Puede seleccionar la talla, de las que hay disponibles. - Puede visualizar los comentarios y la valoración que ha dado (en caso de que lo haya valorado). - Puede añadir el producto al carrito. - En caso de no haber seleccionado talla y querer añadir el producto al carrito, se le informa con un mensaje emergente.

Cuadro 11: Historia de usuario UC10

UC11 Visualización de detalles del conjuntos	
Descripción	Como usuario, quiero poder visualizar conjuntos de productos recomendados para añadir varios artículos al carrito de forma rápida y cómoda.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la sección de conjuntos. - Visualiza cada conjunto con su imagen, nombre y descripción (si la tiene). - Puede acceder al detalle de cada prenda individual incluida en el conjunto. - Puede añadir el conjunto completo al carrito con un solo clic. - Si algún producto del conjunto no tiene stock, el sistema informa al usuario mediante un mensaje emergente.

Cuadro 12: Historia de usuario UC11

UC12 Buscar productos	
Descripción	Como usuario, quiero buscar productos por nombre o palabra clave, para encontrar rápidamente lo que necesito.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario escribe una palabra clave en el campo de búsqueda. - El sistema muestra los productos cuyo nombre o descripción coinciden con la búsqueda. - Si no hay coincidencias, se muestra un mensaje informativo.

Cuadro 13: Historia de usuario UC12

UC13 Añadir un producto o conjunto al carrito	
Descripción	Como usuario, quiero poder añadir productos o conjuntos al carrito de compra, seleccionando la talla y la cantidad deseada, para prepararme para finalizar la compra.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de detalle del producto o conjunto. - Selecciona la talla, el color (si aplica) y la cantidad. - Pulsa el botón '<i>Añadir al carrito</i>'. - El producto o conjunto se añade al carrito y se muestra una notificación de confirmación. - Si falta alguna selección obligatoria (como la talla), se muestra un mensaje de error.

Cuadro 14: Historia de usuario UC13

UC14 Carrito de compras	
Descripción	Como usuario registrado, quiero ver y gestionar el contenido de mi carrito, para revisar mi compra antes de finalizarla.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página del carrito. - Puede ver los productos añadidos, junto con su cantidad, talla y precio. - Puede modificar la cantidad de cada producto o eliminarlo del carrito. - El sistema actualiza automáticamente el total de la compra. - Si el usuario es cliente VIP, se le aplica un 20% de descuento. - El usuario puede continuar al proceso de pago si ha aceptado las condiciones y ha iniciado sesión.

Cuadro 15: Historia de usuario UC14

UC15 Realizar un pedido	
Descripción	Como usuario registrado, quiero poder finalizar la compra de los productos que tengo en el carrito, para completar el pedido y recibir una confirmación.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la pantalla de resumen del pedido. - El sistema muestra el total de la compra y los impuestos aplicables. - El usuario acepta las condiciones de compra marcando una casilla obligatoria. - Selecciona un método de pago y confirma la compra. - El sistema verifica el stock de los productos y procesa el pago. - El usuario recibe una confirmación en pantalla y por correo electrónico.

Cuadro 16: Historia de usuario UC15

UC16 Crear o editar productos	
Descripción	Como gestor o administrador, quiero poder crear y editar productos, para mantener el catálogo actualizado y ofrecer información precisa a los usuarios.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador o gestor accede al panel de gestión de productos. - Puede crear un nuevo producto introduciendo: nombre, referencia, imagen, categoría, tipo, precio, visibilidad, color y tallas con su stock. - También puede seleccionar un producto existente y modificar cualquiera de sus campos. - Al guardar los cambios, el sistema actualiza la base de datos y refleja el producto en el catálogo (si está visible).

Cuadro 17: Historia de usuario UC16

UC17 Crear conjuntos de productos	
Descripción	Como gestor o administrador, quiero poder crear conjuntos de productos a partir de artículos ya existentes, para ofrecer combinaciones recomendadas a los usuarios.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador o gestor accede a la sección de gestión de conjuntos. - Introduce el nombre del conjunto, imagen, descripción (opcional), visibilidad y precio. - Selecciona desde el catálogo los productos que formarán parte del conjunto. - Guarda el conjunto, que estará disponible según la visibilidad configurada.

Cuadro 18: Historia de usuario UC17

UC18 Gestionar visibilidad de productos	
Descripción	Como gestor o administrador, quiero poder cambiar la visibilidad de un producto para mostrarlo u ocultarlo del catálogo según sea necesario.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador o gestor accede a la ficha del producto. - Selecciona el estado de visibilidad: oculto, visible o visible solo para usuarios VIP (si aplica). - El sistema actualiza la visibilidad del producto y lo muestra únicamente a los usuarios correspondientes.

Cuadro 19: Historia de usuario UC18

UC19 Ordenar productos	
Descripción	Como usuario, quiero poder ordenar los productos del catálogo por diferentes criterios, para facilitar mi decisión de compra.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario selecciona una opción de ordenación: por precio (ascendente o descendente), por novedades o por popularidad. - El sistema reorganiza el catálogo automáticamente según el criterio seleccionado. - El usuario puede cambiar el criterio de ordenación en cualquier momento.

Cuadro 20: Historia de usuario UC19

UC20 Filtrar productos	
Descripción	Como usuario, quiero poder aplicar filtros en el catálogo para visualizar únicamente los productos que se ajustan a mis preferencias.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede al panel de filtros. - Puede aplicar uno o más filtros: categoría, precio, talla, color, entre otros. - El catálogo se actualiza automáticamente en función de los filtros seleccionados. - El usuario puede restablecer los filtros para volver a visualizar todos los productos.

Cuadro 21: Historia de usuario UC20

UC21 Visualizar la lista de usuarios	
Descripción	Como administrador, quiero poder ver un listado de todos los usuarios registrados, para tener un control general sobre quién utiliza la plataforma.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador accede al panel de gestión de usuarios. - El sistema muestra un listado con información básica: nombre, correo electrónico, fecha de registro y rol. - El administrador puede buscar usuarios por nombre o correo electrónico. - El administrador puede ordenar la lista por fecha de registro o nombre.

Cuadro 22: Historia de usuario UC21

UC22 Ver detalles de un usuario	
Descripción	Como administrador, quiero poder acceder a la información completa de un usuario específico, para revisar o verificar sus datos.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador selecciona un usuario de la lista. - El sistema muestra su perfil completo: datos personales, historial de pedidos, productos guardados y rol asignado.

Cuadro 23: Historia de usuario UC22

UC23 Editar información de un usuario	
Descripción	Como administrador, quiero poder modificar los datos personales o el rol de un usuario, para corregir errores o cambiar permisos.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador accede al perfil de un usuario. - Puede editar campos como nombre, correo electrónico, dirección, teléfono o rol (usuario/gestor/VIP). - El sistema guarda los cambios y actualiza la información del usuario.

Cuadro 24: Historia de usuario UC23

UC24 Eliminar un usuario	
Descripción	Como administrador, quiero poder eliminar un usuario del sistema si ya no debe tener acceso a la plataforma.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador accede al perfil de un usuario. - Pulsa el botón de eliminación. - El sistema solicita confirmación antes de proceder. - Si se confirma, el sistema elimina la cuenta y los datos asociados (según la política de datos).

Cuadro 25: Historia de usuario UC24

UC25 Asignar o modificar roles	
Descripción	Como administrador, quiero poder asignar o modificar el rol de un usuario, para controlar sus permisos dentro de la plataforma.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador accede a la información del usuario. - Selecciona el nuevo rol (usuario, gestor o administrador). - El sistema actualiza los permisos automáticamente según el rol asignado.

Cuadro 26: Historia de usuario UC25

UC26 Visualización de pedidos	
Descripción	Como administrador o gestor, quiero poder visualizar todos los pedidos realizados por los usuarios, para controlar y hacer seguimiento del estado de las compras.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador o gestor accede al panel de gestión de pedidos. - El sistema muestra un listado con los pedidos realizados: identificador, usuario asociado, fecha, estado, importe total y productos incluidos. - Puede filtrar los pedidos por estado (pendiente, en preparación, completado, cancelado), por fecha o por usuario. - Puede ordenar los pedidos por fecha, importe o estado. - Al hacer clic en un pedido, se accede a una vista detallada con toda la información del mismo.

Cuadro 27: Historia de usuario UC26

UC27 Gestión de pedidos	
Descripción	Como administrador o gestor, quiero poder consultar y gestionar los pedidos de todos los usuarios, para controlar el flujo de compras y su estado.
Criterios de aceptación	<ul style="list-style-type: none"> - El administrador accede al panel de gestión de pedidos. - Puede visualizar todos los pedidos realizados, junto con la información del usuario, el estado, los productos incluidos y la fecha. - Puede cambiar el estado del pedido (por ejemplo: pendiente, en preparación, completado, cancelado). - Puede aplicar filtros para buscar pedidos por estado, fecha o usuario.

Cuadro 28: Historia de usuario UC27

UC28 Aplicar un cupón de descuento	
Descripción	Como usuario registrado, quiero poder aplicar un código de descuento durante el proceso de compra, para reducir el coste total del pedido.
Criterios de aceptación	<ul style="list-style-type: none"> - El usuario accede a la página de resumen del pedido. - Introduce un código de descuento en el campo correspondiente. - El sistema verifica si el código es válido y está activo. - Si es válido, se aplica el descuento y se actualiza el importe total. - Si no es válido, se muestra un mensaje informativo con el motivo del error.



Cuadro 29: Historia de usuario UC28

3. Diseño

El apartado de diseño tiene como objetivo describir la estructura visual y funcional de la aplicación web. Se detallan los aspectos relacionados con la navegación, la jerarquía de páginas, la disposición de la interfaz y las decisiones tomadas para garantizar una experiencia de usuario clara y accesible. A través de diagramas y explicaciones, se justifica la organización del contenido y la lógica de acceso a las diferentes secciones de la aplicación.

3.1. Diagrama del diseño de navegación

El diseño de navegación representa la estructura jerárquica de nuestra aplicación web, facilitando la comprensión del flujo de pantallas mediante una representación visual clara. Este tipo de diagrama muestra cómo están organizadas las diferentes secciones del sitio.

A continuación, en la figura 4, se presenta el diagrama de navegación desarrollado para mi página web. En él se puede observar que algunas rutas están marcadas con un icono de candado () , lo cual indica que el acceso a dichas vistas está restringido únicamente a usuarios que han iniciado sesión, es decir, que están autenticados. Además, otras rutas están marcadas con un icono de administrador () , representando funcionalidades exclusivas para usuarios con rol de administrador o

gestor.

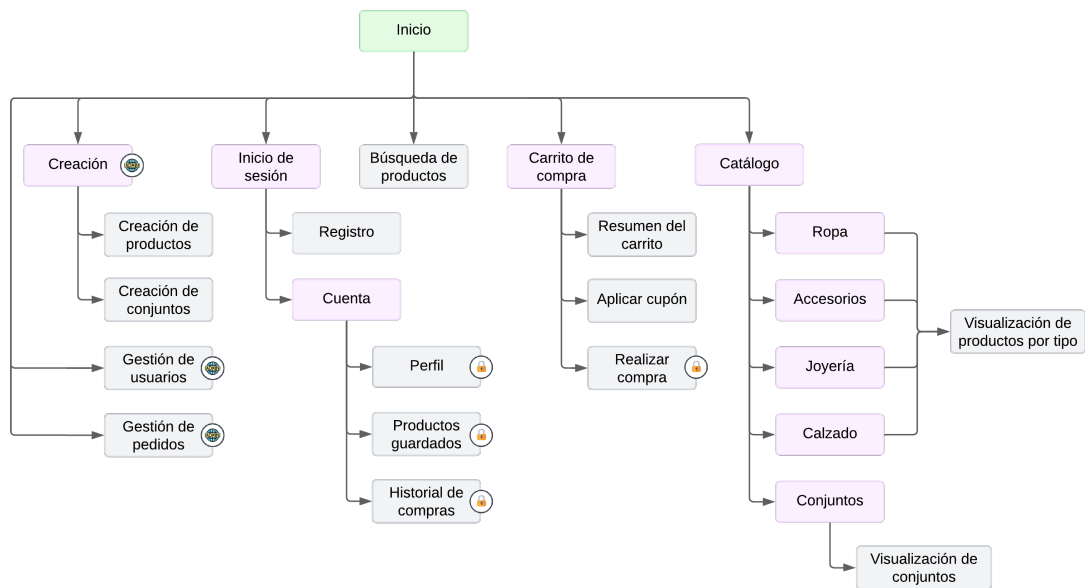


Figura 4: Mapa de navegación

Tal y como se muestra en la figura 4, desde la página de inicio es posible acceder al resto de las secciones principales, como el inicio de sesión, el catálogo, el carrito de la compra, la cuenta de usuario o las funcionalidades administrativas.

Es importante destacar que, una vez dentro de la aplicación, la navegación es flexible, lo que permite al usuario moverse libremente entre las distintas vistas sin necesidad de siempre regresar a la pantalla de inicio.

En el mapa de navegación también se ha procurado identificar claramente la separación entre las funcionalidades públicas, privadas y administrativas.

3.2. Diseño de la base de datos

Respecto a la base de datos, se han definido diferentes tablas que se encargarán de gestionar la información relacionada con los usuarios y los productos, como los comentarios, valoraciones, los detalles de los productos y más. En la figura 5 se muestra el modelo entidad-relación.

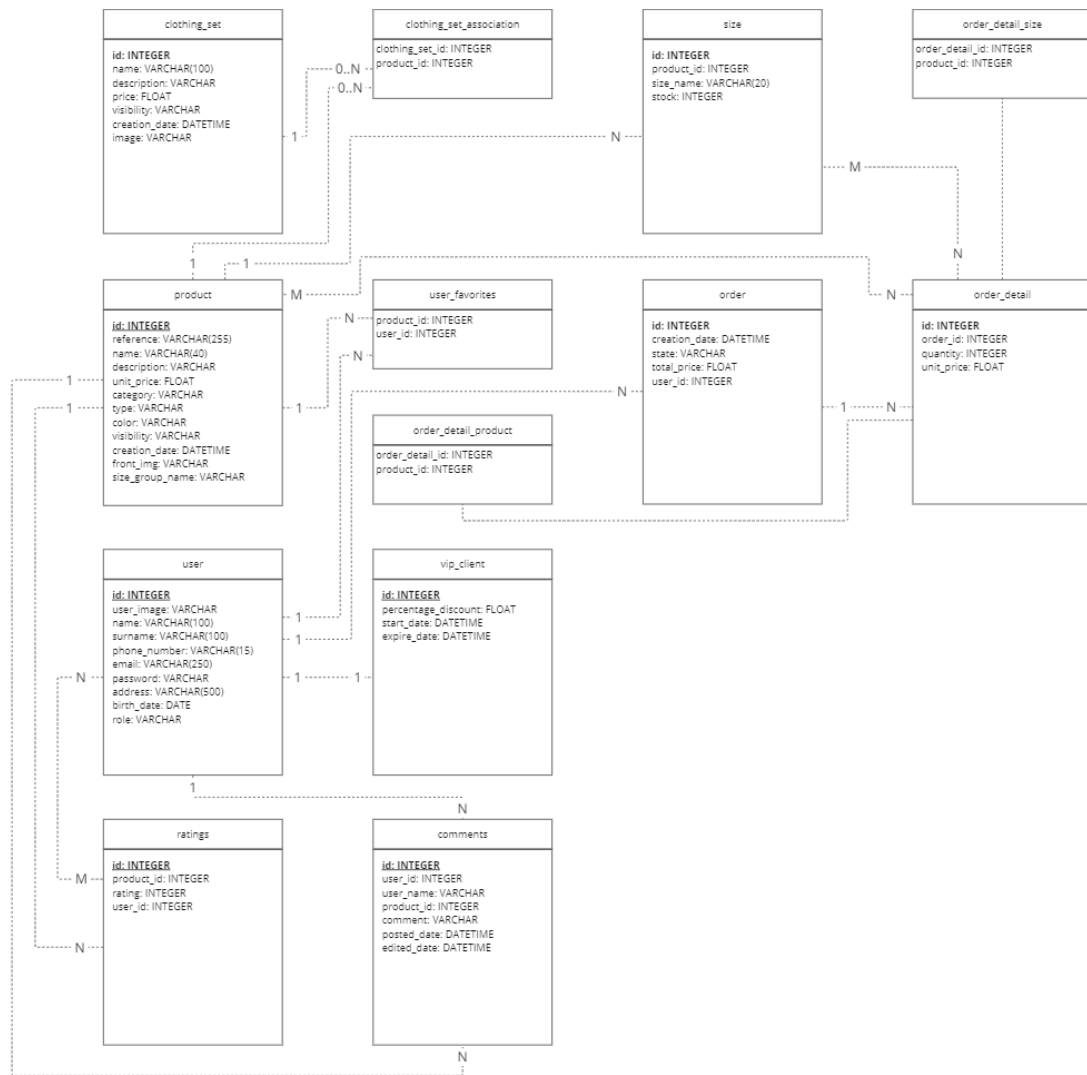


Figura 5: Modelo Entidad-Relación de la base de datos de Serendipia

A continuación, se detallan las tablas principales por escrito:

- User: Guarda los datos de los usuarios registrados.
- Vip Client Data: Guarda los datos de los clientes VIP.
- Product: Guarda los datos de los productos.
- Clothing Set: Guarda los datos de los conjuntos.
- Size: Guarda las tallas de cada producto y el stock de cada una.
- Order: Guarda los pedidos que hacen los usuarios.
- Order Detail: Guarda los detalles de los pedidos

- Comments: Guarda los comentarios que los usuarios hacen sobre los productos.
- Ratings: Guarda las valoraciones que los usuarios hacen sobre los productos.

A parte de estas tablas principales, que utilizamos y tienen funcionalidades asociadas. Existen algunas tablas que han sido creadas pero no tiene funcionalidades asociadas, como la tabla de cupones. En el apartado de extensiones se entrará más en detalle sobre ellas.

En los anexos están definidas las tablas con los detalles y los atributos de cada tabla, concretamente en anexos A.1.

3.3. Diseño de los mockups

El diseño de unos buenos *mockups* es un trabajo muy importante y además una de las tareas que más disfruto. Por lo que a continuación muestro el diseño inicial que realicé para que me sirviese de guía a la hora de desarrollar la *User Interface*. El diseño no está completo, ya que lo realicé únicamente para las pantallas más principales de la web.

También comentar que hay cambios sustanciales y que mi idea principal no era seguir el diseño al pie de la letra, sino que fuese una orientación a la hora de programarlo.

En la figura 6 podemos dar respuesta a la UC01 Visualización de la pantalla de inicio cuadro 2. Podemos visualizar la pantalla de inicio con los productos destacados y en el footer tenemos el Instagram de la tienda (dato de contacto) y, además, podemos acceder a la sección de contacto.

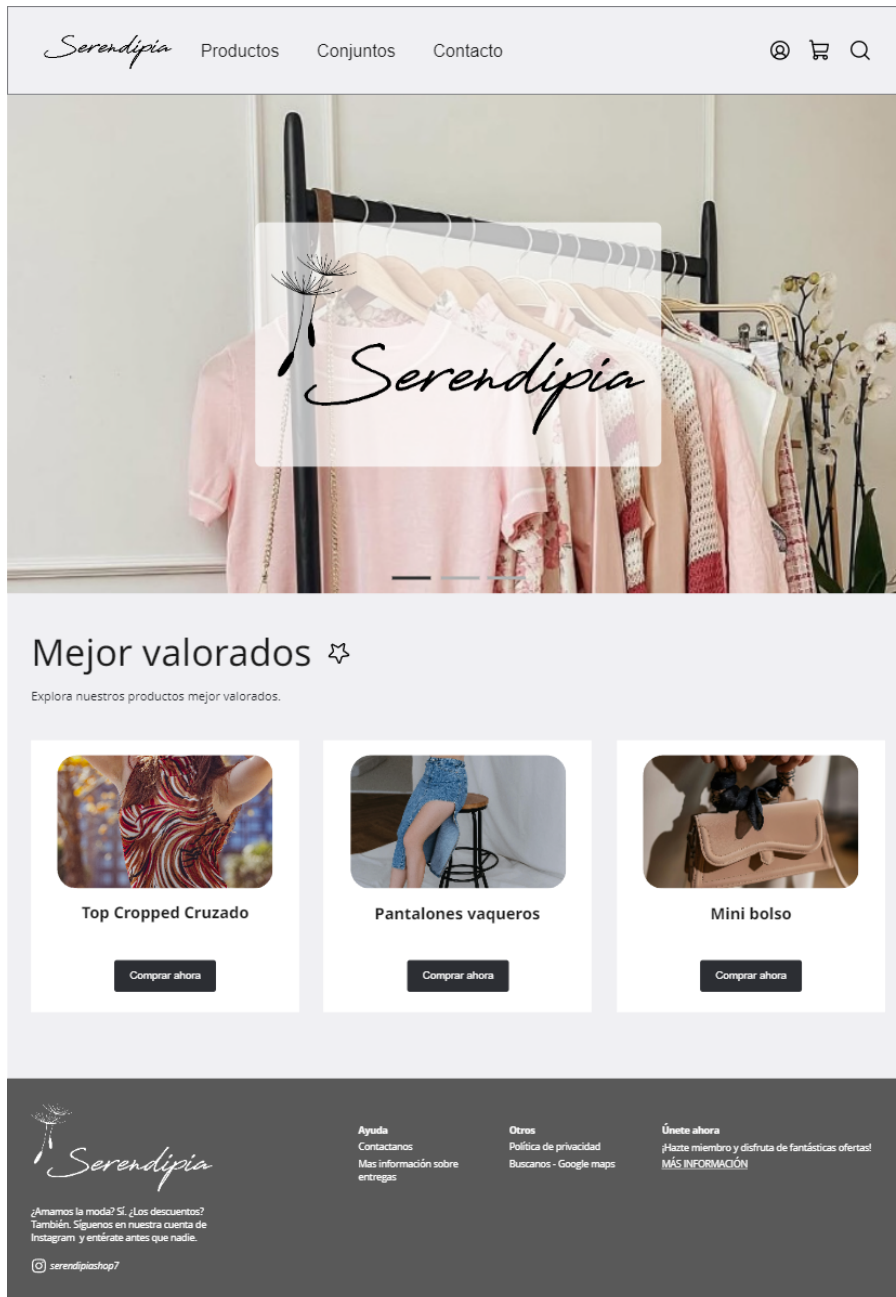


Figura 6: Página de inicio

En la figura 7 damos respuesta al caso de uso UC08 Visualización del catálogo de productos, cuadro 9. Se puede navegar entre las categorías y tipos de los productos y se pueden aplicar filtros de ordenación.

En la figura 8 damos respuesta al caso de uso UC09 Visualización del catálogo de conjuntos Cuadro 10. Se visualizan los conjuntos en las tarjetas con el precio.

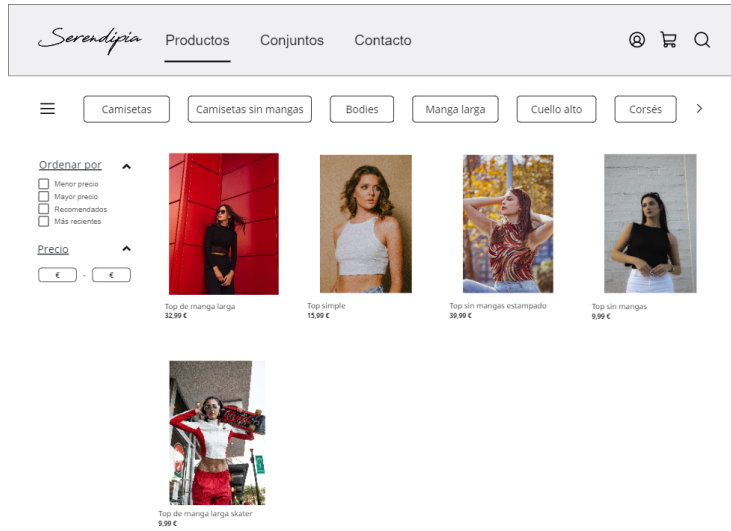


Figura 7: Catalogo de productos

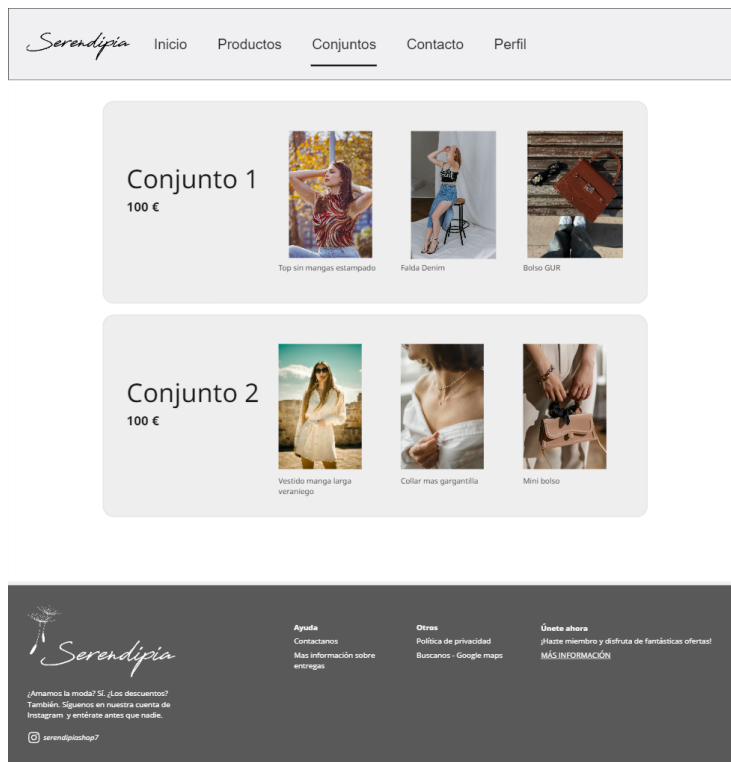


Figura 8: Catalogo de conjuntos

En la figura 9 damos respuesta al caso de uso UC10 Visualización de detalles del producto cuadro 11. Se puede visualizar el producto con sus datos. También se puede acceder a la sección de comentarios y valoraciones sobre el producto.

Serendipia Productos Conjuntos Contacto

Productos > Superior > Top_Cropped_Cruzado

Top sin mangas estampado
39,99 € | ★★★★★ 18 Valoraciones

SELECCIONAR TALLA

XS	S	M	L	XL	XXL
----	---	---	---	----	-----

Añadir

Opiniones y valoraciones

4.1 ★★★★★ 18 Valoraciones

Comentarios

Ana María Jiménez 23 Marzo 2025
★★★★★
Está súper bien y tiene muy buena calidad, lo recomiendo mucho

Juan José Chavez 03 Febrero 2025
★★★★☆
A mi mujer le encantó!

Añade tu comentario.

Serendipia

¿Amamos la moda? Sí. ¿Los descuentos? También. Siguenos en nuestra cuenta de Instagram y entérate antes que nadie.
@serendipiahop7

Ayuda
Contactanos
Mas información sobre entregas

Otros
Política de privacidad
Buscamos - Google maps

Únete ahora
¡Hazte miembro y disfruta de fantásticas ofertas!
MÁS INFORMACIÓN

Figura 9: Detalle de conjunto

En la figura 10 damos respuesta al caso de uso UC11 Visualización de detalles del conjunto cuadro 12. Se puede visualizar el detalle del conjunto y los productos que lo conforman.

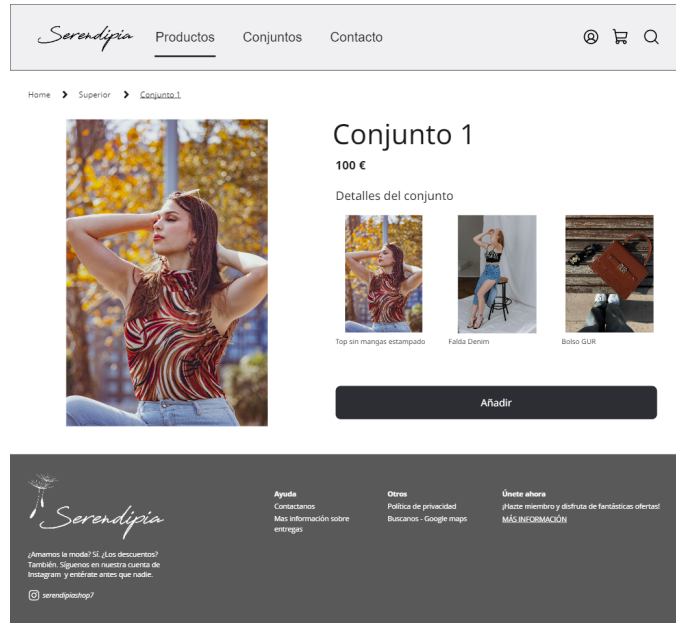


Figura 10: Detalle de producto

En la figura 11 damos respuesta al caso de uso UC14 Carrito de compras, cuadro 15. Se pueden visualizar los productos añadidos al carrito, el precio unitario, el subtotal de la compra y también se pueden añadir cupones de descuento.

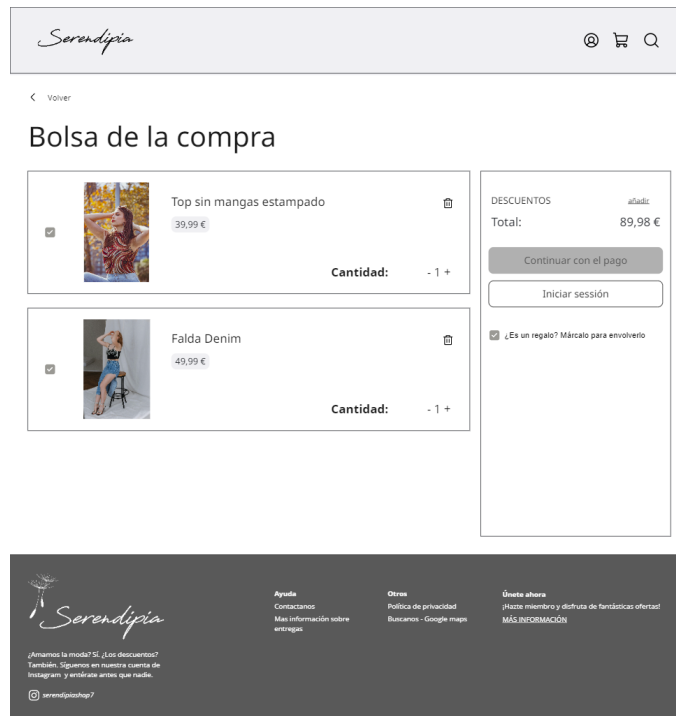


Figura 11: Carrito de compra

4. Implementación

Durante la implementación, definiremos tres espacios de trabajo bien separados. Tendremos la base de datos, el backend y el frontend. Inicialmente, se van a especificar las tecnologías utilizadas durante la fase de desarrollo y la fase de despliegue.

4.1. Selección de Tecnologías

A la hora de desarrollar un proyecto, es muy importante el momento de escoger las tecnologías con las que vamos a trabajar.

- Base de datos **SQLite** y **PostgreSQL**

- **SQLite producción:** es una base de datos relacional, perfecta para almacenar datos de forma local durante la fase de desarrollo. Es simple, portátil y fácil de integrar con Python.
- **PostgreSQL producción:** es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto. Ofrece mayor robustez, rendimiento y capacidad de gestión que SQLite.

- **Backend**

- **FastAPI:** Framework web moderno para la creación de APIs con Python. Permite definir las rutas (endpoints) de manera declarativa y rápida, con validación automática de datos mediante `pydantic`.
- **Uvicorn:** Servidor ASGI utilizado para ejecutar la aplicación FastAPI de forma eficiente y asíncrona.
- **SQLAlchemy:** Librería de acceso a la base de datos desde Python. Permite ejecutar consultas SQL para operaciones CRUD (Create, Read, Update, Delete).

- **Conexión entre frontend y backend**

- **Dio:** Es una librería de Flutter que se utiliza para realizar peticiones HTTP. He utilizado esta librería para hacer llamadas a los endpoints de la API del backend. Permite enviar y recibir datos mediante los métodos GET, POST, PUT y DELETE.

- Frontend **Framework Flutter**

- Flutter está basado en Dart. Dart es un lenguaje de programación moderno, orientado a objetos y *open source*, desarrollado por Google. Utilizado principalmente junto a Flutter, permite escribir código limpio,

mantenible y eficiente. Flutter permite desarrollar aplicaciones multi-plataforma desde un único código base, lo que reduce los tiempos de desarrollo y facilita la coherencia visual entre plataformas. Ofrece una gran flexibilidad a nivel de diseño, gracias a su sistema de widgets personalizables.

4.2. Desarrollo del Backend

El desarrollo del backend es una tarea crucial, ya que es el encargado de gestionar y procesar las solicitudes de los clientes y ofrecerles un buen servicio sin errores inesperados. En mi caso he utilizado Python con el framework de FastAPI y el servidor Uvicorn para la creación de las peticiones del backend. A continuación, podemos ver el proceso empleado para el desarrollo.

Antes de ponernos a trabajar tenemos que hacer la configuración inicial e instalar todas las librerías necesarias para poder trabajar con las herramientas comentadas, las librerías más destacadas que hemos utilizado son:

- **Autenticación y seguridad** (passlib, bcrypt, python-jose): Bcrypt se ha utilizado para hacer un textit hashing seguro de las contraseñas, es decir, encriptarlas. Se ha utilizado tanto para la creación de contraseñas como para la comparación. Passlib permite el uso del algoritmo bcrypt. Por último Python sirve para la creación, verificación y firma de tokens JWT, para gestionar el acceso y la autenticación.
- **CORE ASGI y manejo de peticiones** (fastapi, uvicorn, python-multipart): FastAPI permite definir los endpoints y validar los datos y con uvicorn estos datos se sirven bajo ASGI. Por último python-multipart permite procesar solicitudes multipart/form-data (subida de imágenes y archivos).
- **Validación y configuración de datos** (pydantic, pydantic-settings, email_validator): pydantic asegura que las entradas y salidas cumplen tipos estrictos, pydantic-settings centraliza la configuración a partir de variables de entorno. Email_validator comprueba que una cadena cumple la sintaxis y, opcionalmente, que el dominio existe.
- **Persistencia y migraciones** (SQLAlchemy, alembic): SQLAlchemy principalmente facilita las interacciones con las bases de datos. Alembic se utiliza para hacer migraciones de bases de datos para SQLAlchemy. Esta herramienta permite crear, hacer versiones y aplicar scripts que modifican el esquema/las tablas de la base de datos sin perder información ni rehacer la base de datos de cero.

- **Carga de variables de entorno** (python-dotenv - (load_dotenv)): Permite cargar las variables de entorno del archivo .env para tener más seguridad y mantener secretos fuera del código fuente.

Una vez tenemos las librerías importadas, se procede con la inicialización de FastAPI y los middlewares. Donde añadimos CORS para que nuestro frontend pueda acceder a la API e invocarla sin restricciones.

Seguidamente, se establece la conexión entre la aplicación y la base de datos, y las tablas son definidas mediante modelos de Python utilizando SQLAlchemy. Las tablas se representan como clases y se definen los atributos. Una vez definidos los modelos, se pueden crear las migraciones y aplicarlas con Alembic para sincronizar la estructura de la base de datos en el código.

Como se utilizan imágenes en la base de datos, es necesario el montaje de archivos para poder servir imágenes de productos, ya que se tienen que mostrar en la página web, por lo que necesitamos una manera de proporcionar estas imágenes. Lo que hacemos es guardar nuestra imagen en un directorio llamado static de manera que es accesible desde la URL */static/*. Para ello se utiliza el componente StaticFiles.

Los siguientes pasos son los de desarrollar. Hay que definir las rutas o endpoints a los que se accede para hacer las peticiones junto con el código de estos. Para todas las tablas hay métodos para crear entradas en la base de datos, para obtener los datos, ya sea por algún tipo de parámetro o extraerlos todos, para actualizar los datos de la entrada de la base de datos y para eliminar entradas en la base de datos, tanto en grupo, individualmente o todos ellos.

Durante el desarrollo de los endpoints se ha utilizado POSTMAN para ir comprobando el funcionamiento de las funciones.

Para que los endpoints estén protegidos se ha añadido autenticación mediante JWT, de manera que hay tareas que solo pueden realizar usuarios registrados o los administradores.

Finalmente, se ha hecho testing del backend, tests unitarios para comprobar el correcto funcionamiento de los métodos. Ver que lo que se espera es realmente eso y las funciones no tienen un comportamiento inesperado o fuera de lo común, es decir, que los casos de error posibles están gestionados.

4.3. Desarrollo del Frontend

El desarrollo del frontend se ha hecho completamente con Dart y el framework Flutter. Flutter dispone de muchas librerías que proporcionan infinidad de opciones en cuanto a diseño, y en su mayoría los componentes base son editables en la gran parte de sus características.

El código del proyecto, del frontend en concreto sigue una estructura de Clean Architecture, en gran medida y un poco simplificado. Donde separamos los archivos en diferentes directorios. Este tipo de estructura facilita la organización del código y la escalabilidad del proyecto.

- *data*: contiene todos lo relacionado con acceso a datos como peticiones HTTP a la API o modelos de respuesta. Aquí podremos encontrar los *repositories* que es donde se hacen las peticiones a la API y se transforma la información a modelos que hemos creado en el dominio.
- *domain*: en esta capa contamos con los modelos y los contratos que utilizan los repositorios, principalmente estos contratos son abstract classes.
- *presentation/ui*: como bien indica su nombre es la interfaz de usuario, tenemos todas las pantallas que se desarrollan con widgets de Flutter e incluyen los controladores de estado, en mi caso concreto BLoC, y más específicamente Cubit, que es el controlador de estados de las pantallas, estos se encargan de traducir el contenido que recibimos de los repositorios.

Para mejorar la gestión del estado y tener una definición más clara de los modelos de datos, se ha utilizado la librería *Freezed*, es una herramienta que facilita la creación de clases inmutables y que son comparables. Permite crear automáticamente los constructores, los operadores de igualdad y las copias modificables de los modelos y los gestores de estado. En el caso de los modelos también permite sobrescribir los métodos *fromJson* y *toJson* para leer datos que vienen con una estructura de mapa.

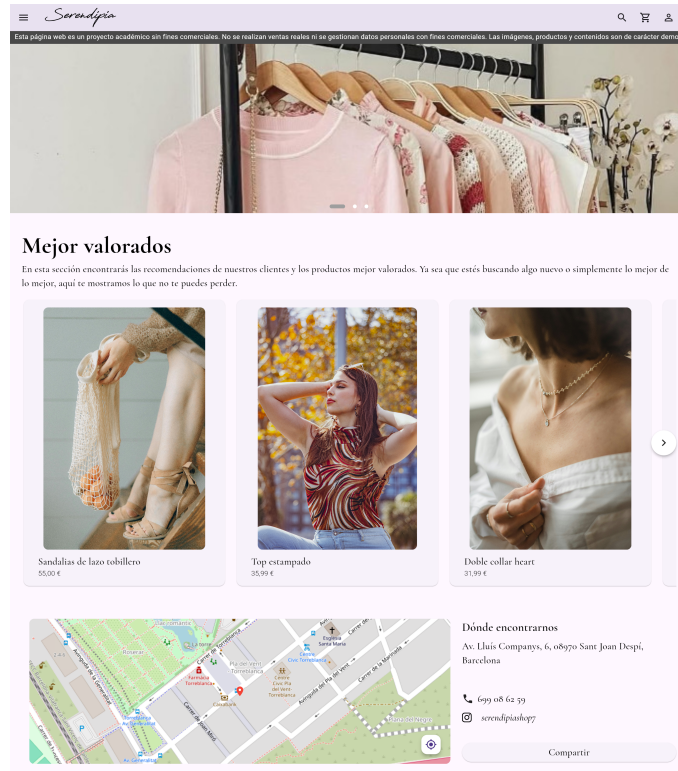
Para el desarrollo de las pantallas se ha hecho todo desde cero, utilizando combinaciones de widgets para crear componentes adaptados a lo que realmente se requiere. No se ha partido de ninguna plantilla predefinida, sino que cada ruta de pantalla era un folio en blanco donde se ha creado de forma manual la interfaz. Se ha cuidado tanto la funcionalidad como la estética, con atención a adaptabilidad a diferentes pantallas.

A continuación, se presentan capturas de las principales vistas del frontend, acompañadas de una breve explicación del propósito de cada una, los elementos que la componen y las decisiones de diseño tomadas durante su desarrollo.

Estas imágenes sirven como evidencia visual del resultado final y reflejan la implementación real de los componentes definidos previamente en los mockups de alta fidelidad. Además, se indicarán los puntos clave en los que se ha buscado mejorar la usabilidad, accesibilidad y consistencia de la interfaz.

A continuación, en la figura 12, podemos ver la pantalla de inicio de la página web. Se pueden observar diferentes espacios; tenemos la *AppBar* o *Header* donde tenemos los iconos que nos permiten navegar a través de las diferentes pantallas y el logo de la empresa que siempre nos lleva a la pantalla de inicio. A continuación, tenemos un carrusel de imágenes; después, la sección de mejor valorados donde se muestran las prendas que tienen una mejor valoración; se muestran las 12 mejor valoradas. La siguiente sección es la de contacto, donde se pueden encontrar los datos de contacto y un botón para compartir la ubicación por WhatsApp. Finalmente, tenemos el *footer* con un pequeño eslogan el Instagram de la empresa y documentos legales que de momento no han sido redactados.

Es importante destacar que en la pantalla de inicio se ha incorporado un mensaje informativo que indica que la página web corresponde a un proyecto académico y que, por tanto, no permite realizar compras reales. Esta aclaración se ha incluido como medida preventiva, ya que, aunque la página está desplegada y no es accesible públicamente sin el enlace directo, se ha considerado conveniente advertirlo explícitamente para evitar cualquier posible malentendido por parte de usuarios externos. Esta decisión responde a una cuestión de responsabilidad y transparencia, dado que la web simula un entorno de tienda real, pero no tiene fines comerciales de momento, es una primera versión.



¿Amamos la moda? Sí. ¿Los descuentos? También.
 Síguenos en nuestra cuenta de Instagram y entréate antes que nadie.




@serendipiahop77

[Contáctanos](#)
[Política de privacidad y cookies](#)
[Condiciones de compra](#)

Figura 12: Pantalla inicial de la web

Centrandonos momentáneamente en la barra de navegación superior, en la figura 13 podemos ver la barra de navegación de un usuario sin registrar o un usuario registrado y que ha iniciado sesión, en cambio, en la figura 14 podemos ver la barra de navegación de un administrador únicamente.

A continuación vamos a ver los componentes que la conforman. Inicialmente, tenemos los componentes comunes y, seguidamente, los que son de administrador únicamente.

-  El icono de hamburguesa permite abrir un drawer lateral que te muestra el menú desplegado con las categorías. Se puede observar en la figura 15.
-  El icono de la lupa nos redirige a la pantalla de búsqueda por palabra clave, es decir por el nombre del producto. Podemos observar dicha pantalla en la figura.
-  El icono del carrito nos redirige a la pantalla del carrito.





-  El icono del usuario al pulsarlo hace la navegación tanto al login, registro como al perfil del usuario. Si se ha iniciado sesión se navega a la pantalla de iniciar sesión desde donde se puede acceder al registro y en caso de tener la sesión iniciada, navega directamente a la pantalla del perfil. En las figuras 16, 17 y 18 se pueden ver las pantallas mencionadas.
-  El icono de la percha con un más, permite a los administradores navegar a la pantalla de creación donde se puede elegir entre crear un producto o crear un conjunto de ropa.
-  El icono de las cuentas de usuario permite al administrador navegar a la pantalla de gestión de usuarios. Como veremos en la figura x más adelante.
-  El icono de la bolsa de compra permite al administrador navegar a la pantalla de gestión de pedidos. En la figura X se muestra dicha pantalla.



Figura 13: App bar de la página web para un usuario con sesión iniciada o sin ella



Figura 14: App bar de la página web para un administrador

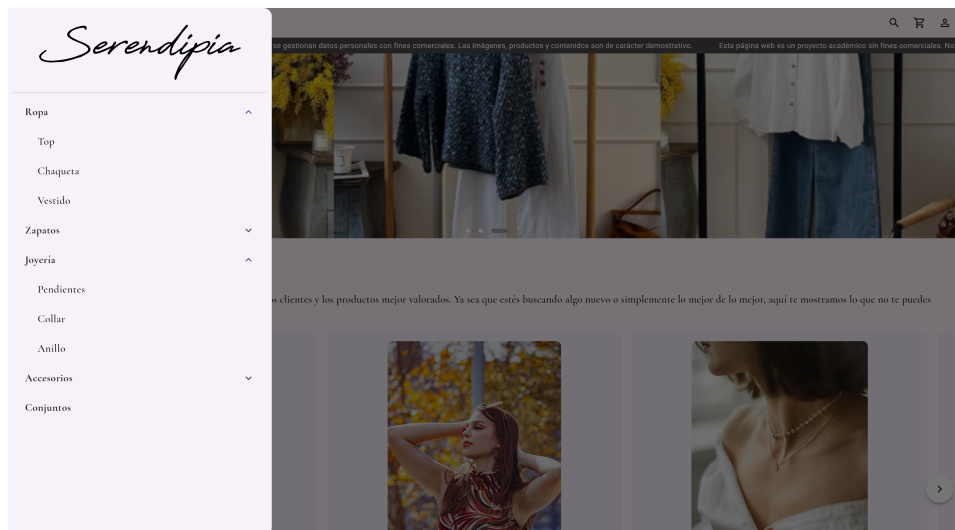


Figura 15: Drawer lateral para la navegación

A continuación pasamos a la pantalla de inicio de sesión y la de registro. Podemos navegar de una u otra sin problema. En la pantalla de inicio de sesión, figura 16,

tenemos que introducir el correo electrónico y la contraseña asociada a ese correo para poder iniciar sesión. Adicionalmente también tenemos el botón de texto de '¿Has olvidado la contraseña?' el cual no está implementado, pero está presente para trabajo futuro.

En la pantalla de registro, figura 17, hay que introducir algunos datos más, como el nombre, los apellidos, la fecha de nacimiento, el correo electrónico y la contraseña; además, hay que aceptar los términos y condiciones. Cabe destacar que el documento de texto legal de términos y condiciones no está redactado, por lo que temporalmente el enlace de ese apartado redirige a Google a la espera de desarrollar dicho documento.

En el formulario de registro hay una validación, en caso de que no se haya introducido ningún valor en los campos se muestran en rojo con un mensaje informativo advirtiendo de que son unos campos obligatorios. El campo del correo electrónico también es comprobado con una expresión regular para verificar que tiene un formato de correo válido y el número de teléfono también, validando la longitud.

Cuando el usuario ha completado el registro se le redirige a la pantalla de inicio de sesión para que introduzca sus datos. Esta decisión ha sido tomada para poder hacer el proceso de confirmación por correo; trabajo futuro, se explicará en otro apartado.

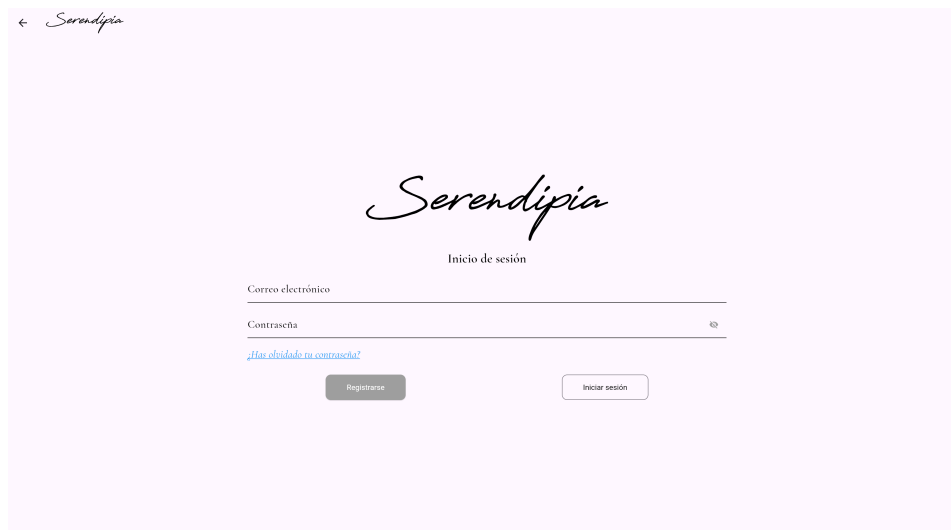


Figura 16: Pantalla de inicio de sesión

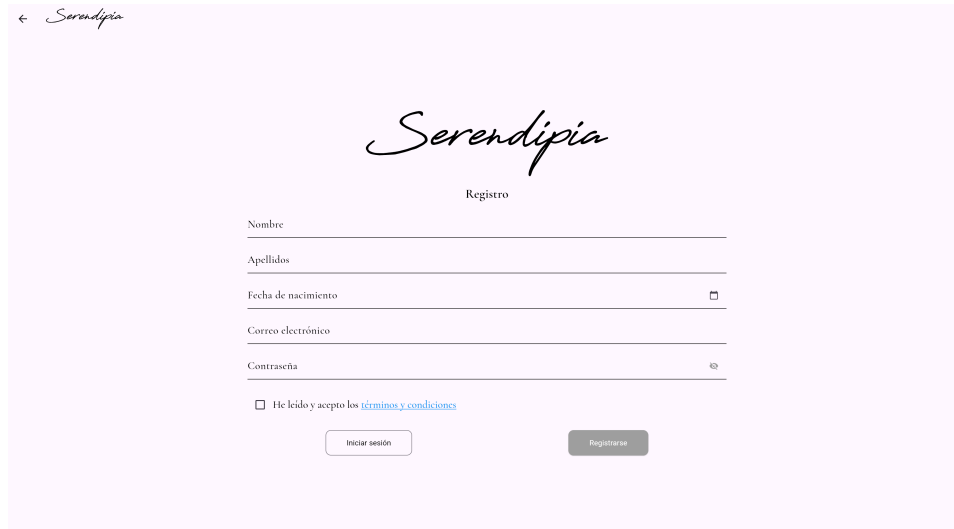


Figura 17: Pantalla de registro

Una vez el usuario ha iniciado sesión, es redirigido a la pantalla de perfil, figura 18. En esta pantalla tenemos diferentes secciones. La principal es la parte superior donde podemos ver nuestros datos principales, imagen de perfil, nombre y correo. Justo debajo tenemos acceso a los botones: favoritos, pedidos y cerrar sesión. Los dos primeros redirigen a las pantallas de productos guardados/favoritos, y la pantalla de tus pedidos respectivamente y el último te permite cerrar la sesión rediriéndote a la pantalla de inicio.

Pulsando sobre el icono (✎) podemos editar nuestros datos personales, que solo se actualizarán si presionamos sobre el botón de actualizar.

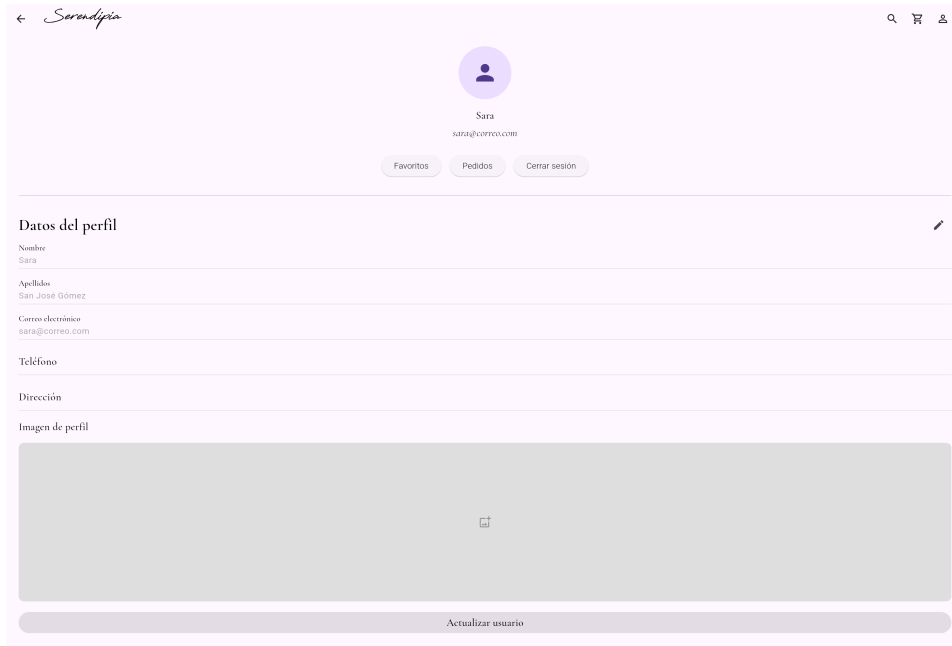


Figura 18: Pantalla de perfil

Una vez visto lo relacionado con autenticación y con los datos de perfil pasamos a los catálogos de productos y el catálogo de conjuntos.

En la figura 19 podemos ver el catálogo de productos, escogiendo la categoría de **ropa** y el tipo de ropa **top**. De manera que el filtraje se hace previamente para que en el supuesto de haber muchos productos la carga de ellos se minimizase cargando únicamente los datos necesarios. Al acceder a esta pantalla se hace una petición al backend para cargar los datos de los productos, y hasta que no se cargan no se muestran.

En esta pantalla vemos los métodos de ordenación de los que disponemos, y en caso de querer acceder al detalle del producto concreto únicamente tenemos que presionar sobre el producto.

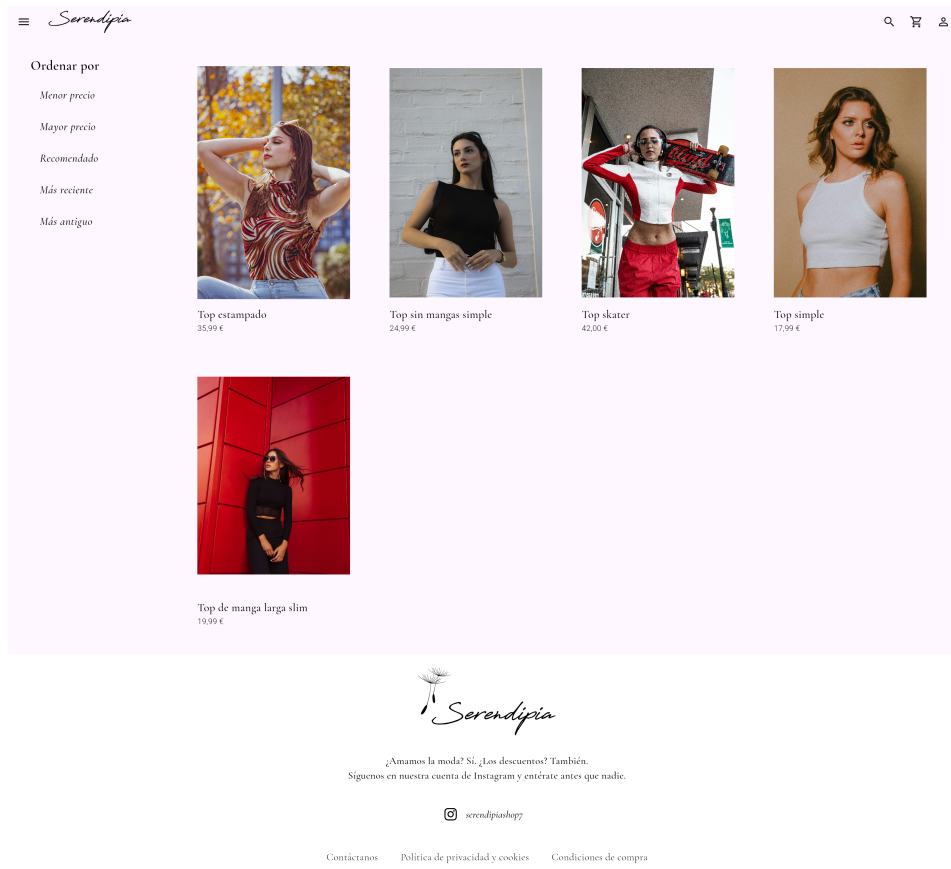


Figura 19: Catálogo de productos

En la pantalla de detalle del producto, figura 20, observamos toda la información referente al producto, la valoración media del producto, las tallas y en el apartado inferior tenemos los comentarios y valoraciones.

En el apartado de información sobre el producto tenemos diferentes interacciones. Podemos presionar el icono (🔖) para guardar el producto en guardados/-favoritos, para poder acceder a él en otro momento más cómodamente. También tenemos el apartado de las tallas que muestra las tallas disponibles y, en caso de que no tengan stock, no son seleccionables. Para poder añadir un producto a la cesta, hay que seleccionar una talla primero, en caso de no seleccionarla, se muestra un mensaje informando de ello. Cuando se añade al carrito el producto, también se muestra un mensaje informando al respecto.

Los productos se pueden valorar independientemente de si los has comprado o no, pero solo se pueden valorar una vez, es decir, un producto solo puede ser valorado una vez por el mismo usuario. De manera que, si queremos dar otra valoración, se modificará la que habíamos dado inicialmente.

En el apartado de comentarios se pueden poner todos los comentarios que se

quieran, el propietario de los mensajes puede tanto editarlos como eliminarlos.

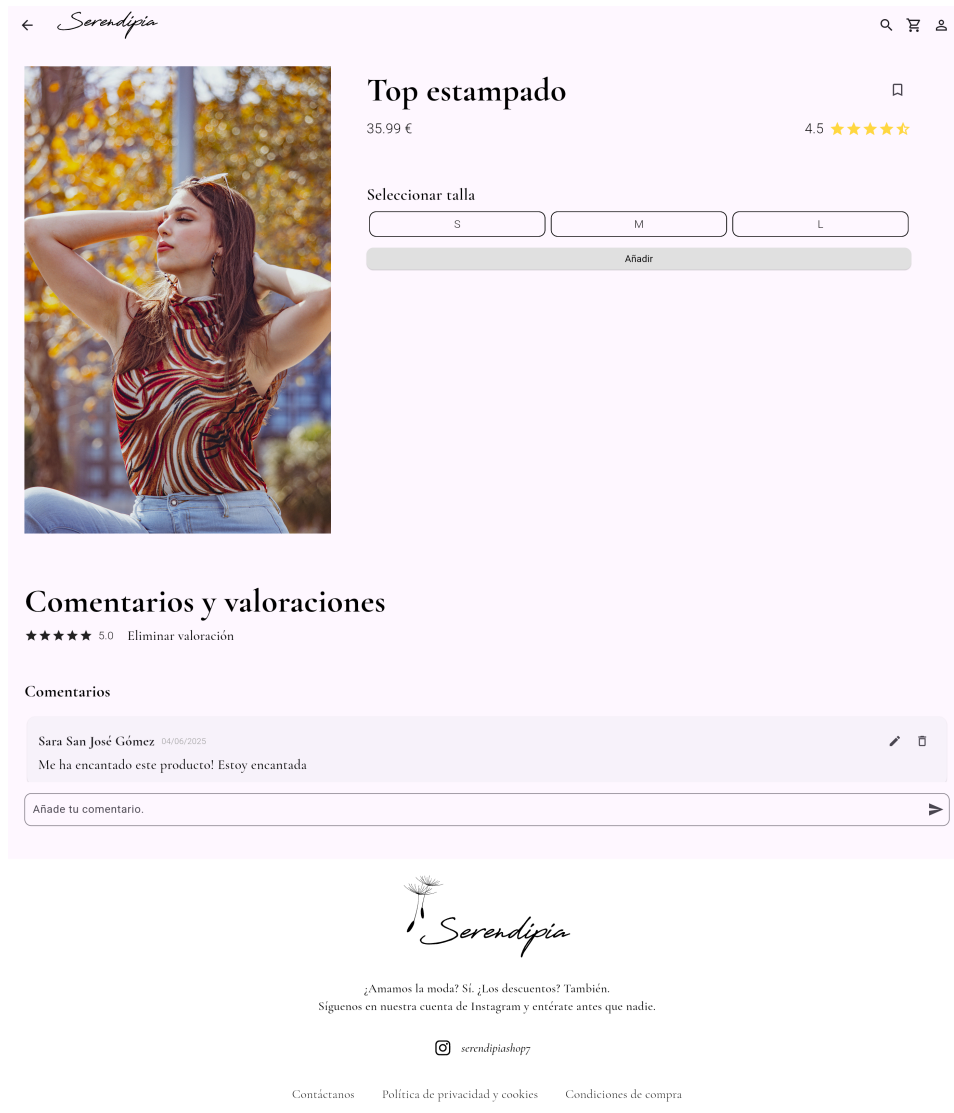


Figura 20: Pantalla de detalle del producto

A continuación tenemos el catálogo de conjuntos, figura 21 es similar al anterior, pero simplificado, no tenemos ordenación ya que los conjuntos se actualizan cada jueves en un directo que se hace en instagram donde se muestran dichos productos. Todos los conjuntos tienen el mismo precio.

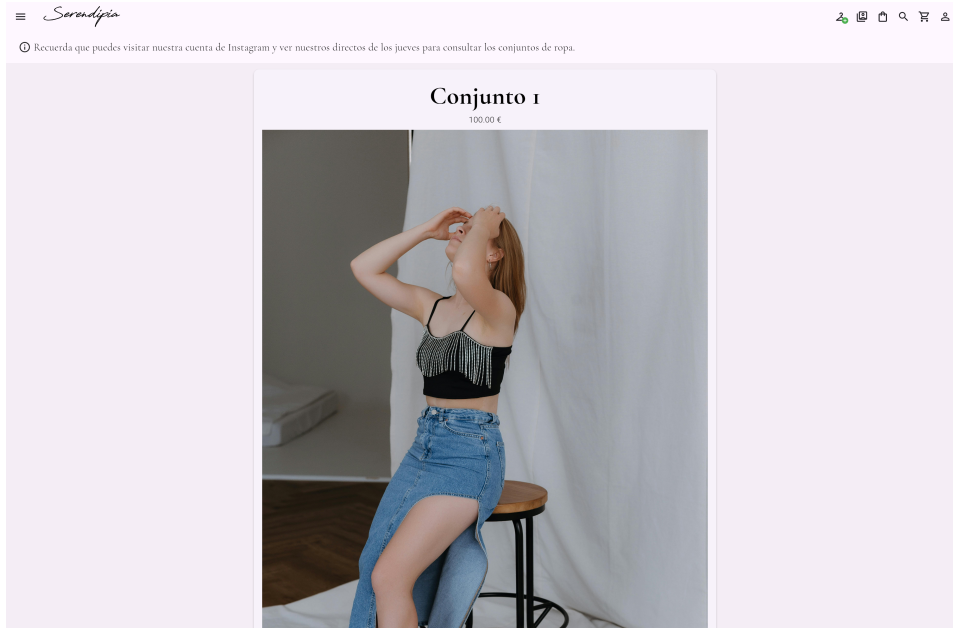


Figura 21: Catálogo de conjuntos

En el caso de los detalles de conjuntos, figura 22, es una pantalla muy simple. En esta no tenemos comentarios ni valoraciones debido al tiempo que dichos conjuntos van a estar publicados, como máximo una semana. En esta pantalla, la única interacción que tenemos es elegir las tallas de los productos y añadir el conjunto al carrito. Al igual que en el caso de los productos, si las tallas no están añadidas no se puede añadir el producto al carrito y el sistema informa de dicha falta.

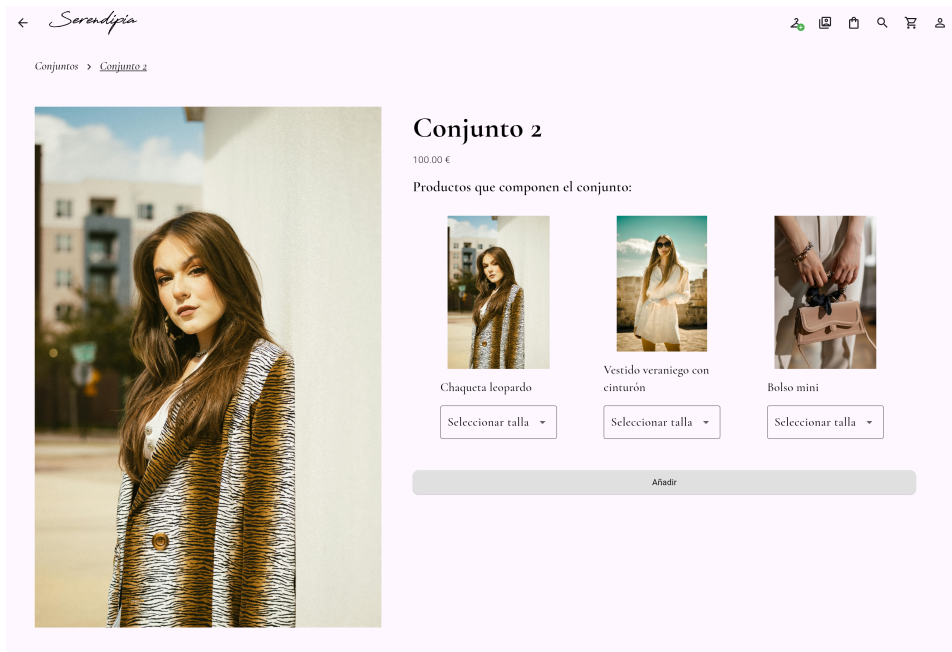


Figura 22: Pantalla de detalle de conjunto

Para la creación de productos y conjuntos primero nos encontramos con una pantalla en la que te da a elegir el tipo de creación que quieres hacer. Podemos ver dicha pantalla en la figura 23.

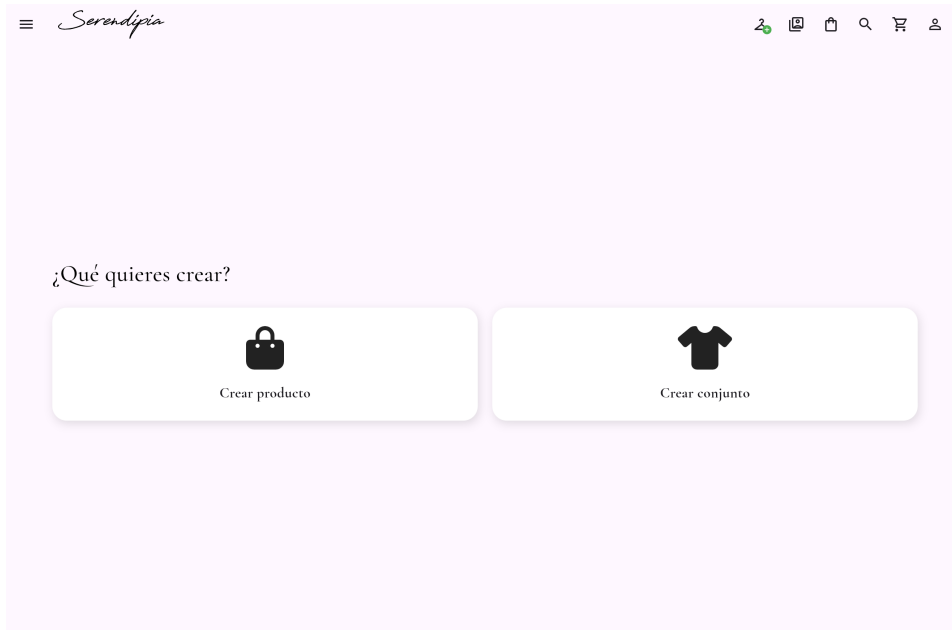


Figura 23: Pantalla de detalle de conjunto

Seguidamente, tenemos la creación de productos y la creación de conjuntos, en las figuras 24 y 25 respectivamente. En estas pantallas cabe destacar que en ambas pantallas es obligatorio rellenar todos los datos si no se puede crear el producto o conjunto, y en el caso de los productos hay que ponerles un número de referencia único.

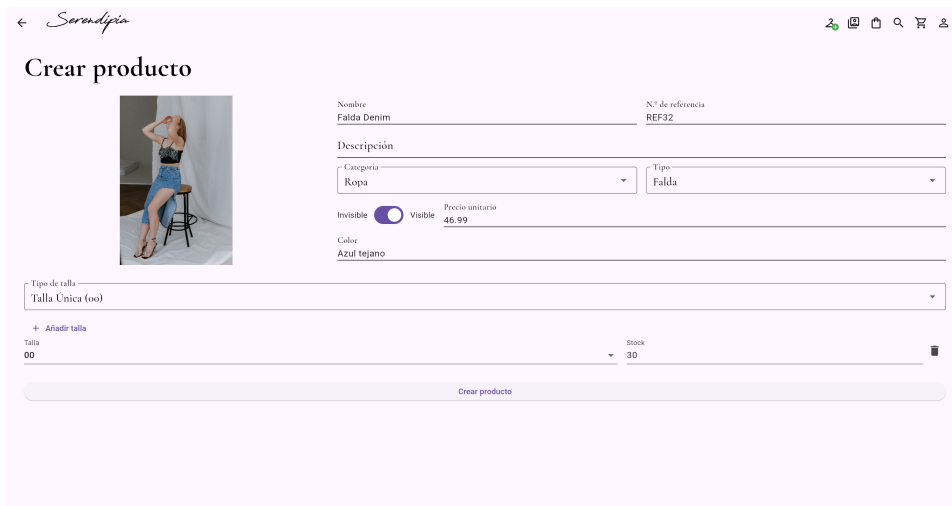



Figura 24: Pantalla de creación de producto

← Serendipia 🏠 📄 🔍 🛒 🧑

Crear conjunto



Nombre
Conjunto 2


Descripción

Invisible Visible Precio unitario
100,00


Selecciona los productos

A continuación selecciona los productos que conforman el conjunto.


Página anterior Página siguiente




Pendientes de oro




Doble collar santo




Conjunto de anillos




Doble collar heart




Pendientes desiguales




Conjunto de anillos simples



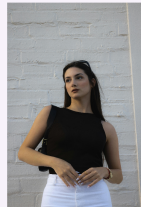
Set de collares




Set de collares planetario




Top estampado




Top sin mangas simple




Top skater




Top simple




Top de manga larga slim




Chaqueta leopardo




Vestido veraniego con cinturón




Vestido holgado




Bolso mini



Bolsa de hombro de piel



Bolso de mano



Bolso redondeado

Crear conjunto

Figura 25: Pantalla de creación de conjunto

Finalmente tenemos la pantalla del carrito, desde donde se pueden hacer los

pedidos, internamente se comprueba que no se supere el stock a la hora de hacer un pedido. No hay proceso de pago, debido a que esto se implementa con pasarelas de pago, y no suelen ser gratuitas, así que por el momento no tenemos pasarela de pago. En el lado derecho podemos observar el subtotal de la compra y un apartado para añadir cupones, cuya implementación queda como trabajo futuro.

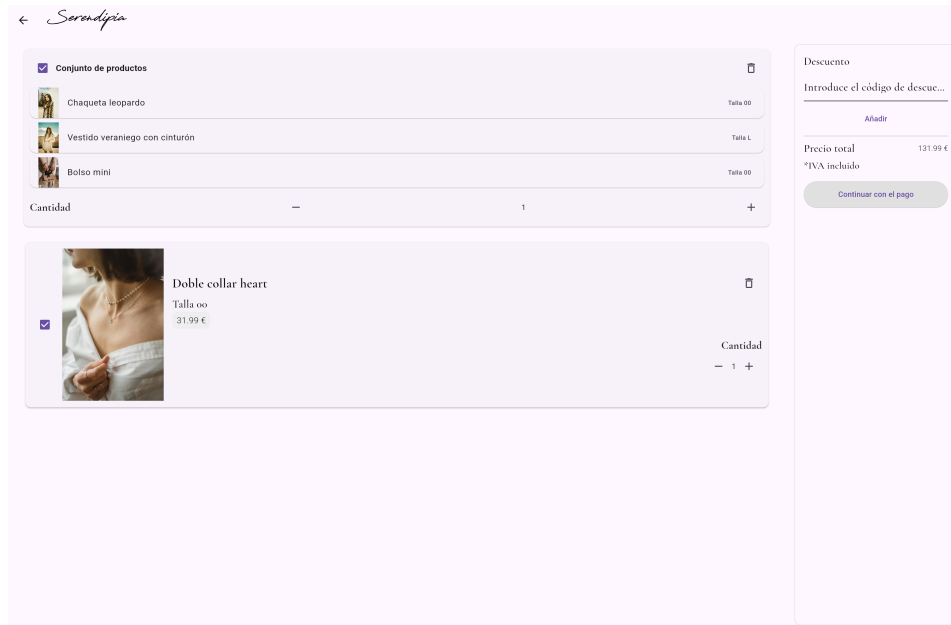


Figura 26: Pantalla de creación de conjunto

5. Testeo

Una parte fundamental del desarrollo es la etapa del testeo porque permite verificar que todo funciona correctamente, que cumple con los requisitos establecidos y ofrece una experiencia de usuario satisfactoria. En este proyecto se han llevado a cabo diferentes tipos de testeo, podemos diferenciar las pruebas en dos grupos, las que están orientadas a comprobar el sistema interno del sistema y las que están orientadas a la interacción con los usuarios finales.

En primer lugar, se han realizado tests unitarios sobre el backend utilizando *pytest*, con la finalidad de comprobar el correcto funcionamiento de los endpoints, tanto cuando funcionan como cuando no, es decir, el correcto funcionamiento para cuando tienen que mostrar errores concretos.

Por otro lado, en el frontend se ha llevado a cabo un testeo con usuarios reales, con el objetivo de analizar la usabilidad de la página web y de la navegación de esta. Gracias a esta prueba hemos recabado mucha información observando a los usuarios y del *feedback* que han proporcionado. En esta etapa se han detectado

algunos comportamientos inesperados y posibles mejoras.

Como último testeo se ha hecho un análisis de accesibilidad, centrado en garantizar que la aplicación sea utilizable por personas con distintas capacidades. Esto incluye la verificación del contraste de colores, la navegación mediante teclado y el correcto etiquetado de los elementos visuales.

La realización de estas pruebas nos ofrece una visión del funcionamiento del sistema y garantiza un nivel mínimo de calidad.

5.1. Testeo de experiencia de usuarios

Para comprobar la usabilidad, funcionalidad y experiencia real del usuario en la página web Serendipia, se ha realizado un testeo de usuarios. El objetivo principal es comprobar que todo funciona de manera correcta. Esto se ha llevado a cabo mediante tareas clave planteadas a los usuarios, las cuales nos permitirán detectar oportunidades de mejora en la comprensión de la interfaz, la navegación y en las interacciones con la web.

Además de este objetivo general, este testeo también está pensado para evaluar si la interfaz es intuitiva, para comprobar que se realizan bien los flujos, identificar posibles errores y recoger impresiones cualitativas de usuarios sobre la experiencia de uso.

Para hacer estas pruebas hemos seleccionado diferentes perfiles tecnológicos para tener más variedad y simular un entorno realista. Hemos tenido cuatro participantes, dos de ellos usuarios que habitualmente utilizan aplicaciones y webs y otros dos usuarios con un perfil tecnológico más bajo, que ocasionalmente utilizan aplicaciones o visitan páginas web.

Las tareas planteadas a los usuarios son las siguientes:

- Registrarse en la página web como un usuario nuevo.
- Iniciar sesión con la cuenta creada.
- Navegar por el catálogo y aplicar algún filtro en el caso del catálogo de productos.
- Hacer una búsqueda.
- Visualizar el detalle de un producto o conjunto.
- Guardar un producto como favorito.

- Añadir uno o varios productos al carrito.
- Acceder al carrito y modificar la cantidad del producto añadido o eliminarlo.
- Volver a la pantalla de inicio.
- Consultar el perfil.
- Acceder a la sección de favoritos y ver los productos guardados previamente.
- Cerrar sesión.

Durante la prueba se ha llevado un registro del tiempo estimado para cada tarea, las dificultades que los usuarios se han encontrado o los errores, algunos comentarios que el usuario ha hecho espontáneamente, así como confusiones, clics erróneos o retrocesos en la página, entre otros.

Los resultados generales obtenidos son que los usuarios han completado las tareas con éxito, con velocidades diferentes. Se ha obtenido una valoración positiva respecto al diseño visual y la claridad general, y también se han detectado algunos pequeños problemas. En el cuadro 30, se puede observar una tabla con las observaciones detalladas para cada tarea.

Tarea realizada	Observaciones
Registrarse en la página web como un nuevo usuario	El formulario funciona correctamente, con validaciones adecuadas. La cuenta se crea y permite el acceso posterior.
Iniciar sesión con la cuenta creada	Inicio de sesión exitoso con redirección automática a la zona autenticada.
Navegar por el catálogo y aplicar algún filtro	La navegación es fluida y los filtros funcionan, aunque son básicos.
Hacer una búsqueda	Se muestran resultados relevantes al término de búsqueda introducido, pero no busca entre los conjuntos.
Visualizar el detalle de un producto o conjunto	Se carga toda la información del producto correctamente. Buen diseño visual.
Guardar un producto como favorito	El producto se marca correctamente y aparece luego en la sección de favoritos.
Añadir uno o varios productos al carrito	El producto se añade correctamente aunque falta un aviso visual más claro de confirmación. Para algunos es bastante el mensaje inferior que se muestra, pero para otros no.
Acceder al carrito y modificar la cantidad del producto añadido o eliminarlo con observaciones	Funciona correctamente, aunque se echa en falta una retroalimentación visual más clara al modificar elementos.
Volver a la pantalla de inicio	Navegación correcta, pero se ha comentado que al pasar por encima del icono de Serendipia, el cursor no cambia, por lo que da la sensación de que no es un botón interactivo.
Consultar el perfil	El usuario puede acceder a su perfil y visualizar sus datos, se permite editarlos, pero se ha encontrado un error al querer subir la imagen de perfil.
Acceder a la sección de favoritos y ver los productos guardados previamente	Los productos marcados aparecen correctamente y se puede acceder a ellos, pero en algunos casos se ha comentado que visualmente no da esa sensación porque el foco no cambia.
Cerrar sesión	Cierre de sesión funcional con redirección limpia a la pantalla de inicio.

Cuadro 30: Resumen del testeo de usuarios sobre tareas funcionales

El testeo ha sido útil para validar que la aplicación es funcional, comprensible y suficientemente robusta para un uso real. Se han extraído varias mejoras concretas que podrían implementarse en futuras versiones para reforzar aún más la experiencia del usuario.

5.2. Testeo de accesibilidad

Para el testeo de accesibilidad, mi tutor me había propuesto utilizar alguna página donde, sirviendo la URL de mi página web, esta hiciera un análisis completo de las pantallas. Pero Flutter Web no utiliza HTML semántico tradicional, sino que la interfaz se renderiza principalmente en un lienzo *canvas*. Se han realizado varias acciones para mejorar y evaluar la accesibilidad de la aplicación. Se ha utilizado Lighthouse, una herramienta de Google Chrome que permite generar informes centrados en aspectos clave de accesibilidad como el contraste de color, el tamaño del texto, la existencia de títulos y descripciones en la página... Este análisis sirve como punto de partida para detectar posibles carencias en lo referente a accesibilidad.

A continuación, añado información extraída de los tests hechos con Lighthouse de Google Chrome. En el caso de la home screen, añado una captura del puntaje extraído, la figura 27.

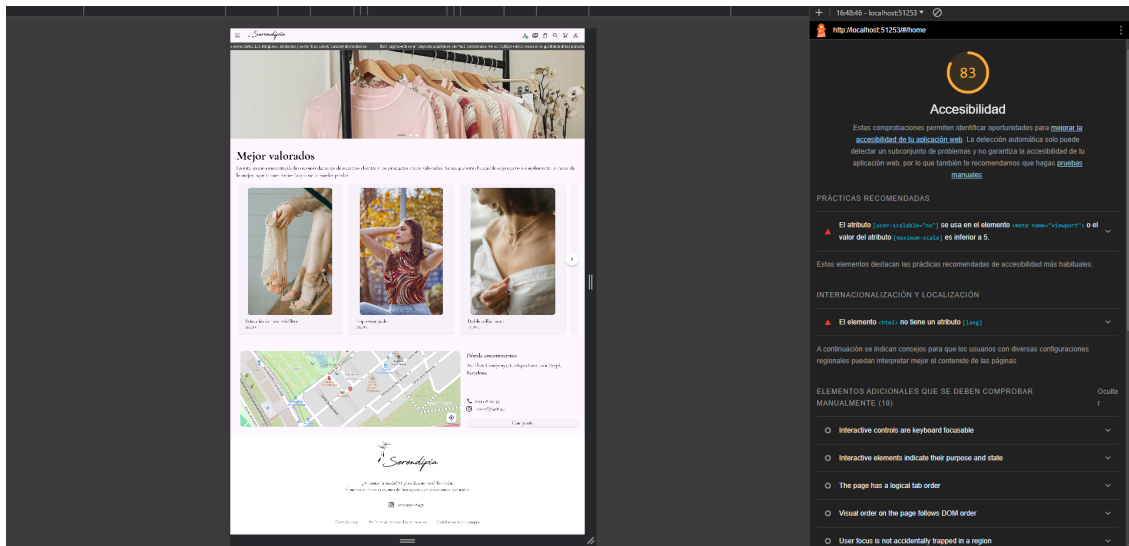


Figura 27: Testeo de accesibilidad con Lighthouse

En todas las ventanas me da la misma puntuación de 83, y las mejoras que se proponen son, en su mayoría, relacionadas con elementos que Flutter no genera automáticamente debido a su sistema de renderizado sobre *canvas*.

- Evitar restringir la escalabilidad del contenido. Esta configuración puede dificultar la ampliación de contenido para personas con dificultades visuales, por lo que se recomienda permitir el escalado del contenido en dispositivos móviles.
- Incluir el atributo `lang` en el elemento `<html>`: la ausencia de este atributo impide a los lectores de pantalla identificar correctamente el idioma del contenido.

También propone revisar manualmente la navegación, es decir, que todos los elementos interactivos sean accesibles por teclado, que haya suficiente contraste y que existan descripciones adecuadas mediante **Semantics**.

Se han llevado a cabo pruebas manuales de navegación con teclado, verificando que los elementos interactivos puedan enfocarse correctamente mediante la tecla Tab, y que exista una respuesta visual del foco en cada componente. La mayoría de los elementos interactivos son accesibles a partir del teclado. En el caso del detalle del producto, en el apartado de comentarios y valoraciones, para dar una valoración a los productos, las estrellas no son accesibles mediante teclado, algo que se debería arreglar de cara al futuro.

Respecto a las **Semantics**, la mayoría de los botones o títulos no tienen un widget **Semantics**, cosa que debo hacer y que realmente tenía muy presente, pero que por temas de tiempo me ha resultado complicado de añadir.

En resumen, aunque Flutter Web presenta ciertas limitaciones en cuanto a accesibilidad debido a su arquitectura basada en canvas, se han realizado esfuerzos conscientes para evaluar y mejorar la experiencia del usuario a través de herramientas automáticas y pruebas manuales. El uso de Lighthouse ha permitido identificar áreas de mejora concretas, y la incorporación progresiva del widget **Semantics** representa un paso importante hacia una aplicación más inclusiva. En futuras versiones, se priorizará la accesibilidad como un aspecto esencial del desarrollo, reforzando así el compromiso con una experiencia de usuario universal.

6. Deploy

6.1. Deploy Front-end

Para desplegar el front-end de la aplicación se ha utilizado Google Cloud, en particular los servicios de **Cloud Run** y **Artifact Registry**, así como **Cloud Build**, que permite verificar los *outputs* y errores de compilación durante el despliegue.

A continuación, se detallan los pasos seguidos para realizar correctamente el despliegue:

- Se crea un nuevo proyecto en Google Cloud.
- Se habilitan los siguientes servicios necesarios:
 - Cloud Run Admin API
 - Cloud Build
 - Artifact Registry
- Se crea un repositorio de imágenes en Artifact Registry para alojar las imágenes Docker.
- Se configura el entorno local con las credenciales y permisos adecuados.
- Se compila el proyecto Flutter Web:
 - `flutter clean`
 - `flutter build web --base-href=/`
 - `mv build/web web_dist`
- Se crea el archivo `Dockerfile` con la configuración necesaria para servir la aplicación mediante Nginx.
- Se crea el archivo de configuración `nginx.conf` para gestionar correctamente las rutas.
- Se construye la imagen Docker y se sube al repositorio:

```
gcloud builds submit . --tag europe-southwest1-docker
.pkg.dev/serendipia-460114/serendipia-deploy/
serendipia-app
```

- Finalmente, se despliega el contenedor en Cloud Run:

```
gcloud run deploy serendipia-app --image europe-
southwest1-docker.pkg.dev/serendipia-460114/
serendipia-deploy/serendipia-app --platform
managed --region europe-southwest1 --allow-
unauthenticated
```

6.2. Deploy Back-end

Para desplegar el back-end, al igual que para desplegar el front-end, se han utilizado los servicios de Google Cloud .

A continuación se detalla el proceso que se ha llevado a cabo:

- Se accede al proyecto creado previamente en Google Cloud.
- Se crea un repositorio de imágenes en Artifact Registry para alojar las imágenes Docker.
- Se configura el entorno local con las credenciales y permisos adecuados.
- Se construye la imagen Docker y se sube al repositorio:

```
gcloud builds submit --tag europe-west1-docker.pkg.dev/serendipia-460114/serendipia-backend/serendipia-backend
```

- Finalmente, se despliega el contenedor en Cloud Run:

```
gcloud run deploy serendipia-backend --image europe-west1-docker.pkg.dev/serendipia-460114/serendipia-backend/serendipia-backend --platform managed --region europe-west1 --allow-unauthenticated --set-env-vars=ENV=prod,DATABASE_URL=postgresql://postgresssl_serendipia_db_user:LHf5IF2pTG3bbdQm631kjKs3LWSrTEyy@dpg-d0pftc6mcj7s73e3fktg-a.frankfurt-postgres.render.com/postgresssql_serendipia_db
```

El proceso es el mismo, pero cambiando el nombre de los contenedores y en este caso le añadimos las variables de entorno necesarias, el environment y la URL de la base de datos desplegada.

6.3. Deploy Base de Datos

Para el despliegue de la base de datos se ha utilizado Render.com, una plataforma que permite crear y administrar bases de datos PostgreSQL en la nube de manera sencilla, y con un plan gratuito.

Para hacer el deploy inicialmente se ha creado la base de datos en Render creando una instancia de base de datos PostgreSQL. Una vez terminado este proceso, la

plataforma proporciona una URL de conexión. Esta URL se integra en el backend de manera que ahora nuestra base de datos es la que esta desplegada. Es decir, se configura como una variable de entorno. Cuando ya hemos configurado la conexión se utiliza Alembic para aplicar las migraciones, es decir, aplicar la estructura que tenemos en local a la base de datos desplegada. Garantizando que las tablas y las relaciones son las mismas que hemos definido.

Además, encontré que el dominio que Render proporciona para acceder a la base de datos no se resolvía correctamente en mi sistema local. Para solucionar este problema, configuré manualmente los servidores DNS de mi equipo, utilizando los servidores públicos de Google (8.8.8.8). Esta modificación permitió que el dominio se resolviera de forma adecuada y, por tanto, que la conexión con la base de datos funcionara sin inconvenientes.

Con estas configuraciones realizadas, pude conectarme desde local a la base de datos alojada en Render y aplicar todas las migraciones necesarias utilizando Alembic, asegurando así que el esquema de la base de datos estuviera actualizado.

7. Conclusiones

El desarrollo de Serendipia me ha permitido cumplir los objetivos planteados al inicio del proyecto y, al mismo tiempo, crecer tanto en el ámbito técnico como personal. El proyecto me ha permitido comprobar que soy capaz de llevar a cabo todo el proceso de desarrollo de un proyecto de inicio a fin. Me ha dado la confianza de saber que soy capaz de idear, construir y mantener una web completa por mí misma. Desde la planificación, teniendo en cuenta lo importante que es hacerlo con criterio, y la estimación de tiempos, pasando por el diseño visual y de datos, la implementación del frontend y del backend, manteniendo un control limpio del código y probando cada cambio antes de aprobarlo, hasta las fases de testeo y despliegue.

Uno de los retos que me propuse era trabajar con Flutter Web y Dart, tecnologías con las que tenía poca experiencia y quería ganar más conocimiento. Integrarlas con un backend en FastAPI y una base de datos relacional me ha dado una visión de las arquitecturas full-stack y me ha obligado a aplicar principios de Clean Architecture y buenas prácticas de control de versiones.

El proyecto también ha reforzado mi sensibilidad hacia el diseño de interfaces y la usabilidad. Las pruebas con usuarios han confirmado que la aplicación resulta intuitiva y visualmente atractiva, en general, aunque han expuesto algunos errores y posibles mejoras de accesibilidad y retroalimentación visual que ya figuran en el trabajo futuro. Asimismo, he interiorizado la importancia de planificar sprints

realistas y validar cada funcionalidad mediante pruebas.

Además de los logros técnicos, Serendipia tiene un componente humano especial: ofrecer una plataforma que acerque los productos de Serendipia a un público más amplio. Haber podido aportar valor a un proyecto local refuerza mi convicción de que la tecnología puede y debe resolver necesidades concretas de la gente.

Aunque la aplicación funciona de forma estable, todavía quedan pasos para convertirla en un producto listo para el mercado: integrar una pasarela de pago real, ampliar la accesibilidad, optimizar el rendimiento y añadir más capacidades administrativas. Estas mejoras, descritas en el apartado de trabajo futuro, son el siguiente paso del proyecto.

En definitiva, este trabajo de fin de grado ha demostrado que soy capaz de concebir, diseñar, desarrollar y mantener una aplicación web completa. Me llevo una experiencia práctica valiosa, una base tecnológica sólida y, sobre todo, la confianza para afrontar los desafíos profesionales que me esperan en el mundo del desarrollo de software.

8. Trabajo futuro

Para que esta página fuese una aplicación que saliese al mercado, completamente funcional, habría que hacer unas cuantas mejoras y algunas implementaciones nuevas. Voy a enumerar este trabajo futuro:

- Añadir **pasarela de pago**: Al ser un trabajo universitario no hemos añadido pasarela de pago porque no se sabe si la página verá la luz finalmente y hasta el momento que se despliegue hay muchas otras tareas que se pueden hacer antes de añadir una pasarela de pago.
- Redacción de **documentos legales**: Hay que redactar los textos legales de política de privacidad y cookies y condiciones legales.
- Funcionalidad de *¿Has olvidado la contraseña?*: Desarrollar la funcionalidad para los usuarios que han olvidado la contraseña y quieren restaurarla.
- **Confirmación de creación de cuenta** vía email: Desarrollar la funcionalidad para enviar un email de confirmación al crear la cuenta, y no permitir el acceso hasta que se haya confirmado la cuenta vía email.
- **Persistencia de datos en el carrito**: Ahora mismo si se cierra la app no se guardan los datos del carrito, por lo que me gustaría mantener estos

datos, almacenarlos para que si se cierra la página y se vuelve a abrir, estén guardados.

- **Redirección a detalles del producto** en favoritos y mejor valorados: Poder acceder a detalles del producto desde favoritos y desde mejor valorados.
- Añadir **semantics**: Una de las cosas que realmente quería añadir desde el inicio del desarrollo era las semantics, para que si tienes un lector de pantalla para una persona con visión reducida, pueda saber a qué está accediendo y pueda interpretar la pantalla mediante lo que narra el lector de pantalla.
- Permitir al administrador añadir **categorías y tipos**: Ahora mismo las categorías y los tipos de los productos están seteados a mano, intentando tener en cuenta todas las posibilidades que hay, pero me gustaría poder hacerlo de manera que el administrador pueda editar estas etiquetas, borrarlas o añadir nuevas.
- Funcionalidad **editar producto**: No he llegado a implementar la funcionalidad de editar producto en el frontend, el endpoint está desarrollado pero me gustaría poder desarrollar esta funcionalidad y editar tanto los datos del producto como las tallas de este.
- Funcionalidades de los **cupones de descuento**: Los cupones de descuento
- **Conversión de usuarios**: Aplicar la funcionalidad de administrador para convertir usuarios, tanto usuarios normales a VIP, como VIPs a usuarios normales.
- **Internacionalización**: Aplicar traducciones a la página web. Lo he dejado todo preparado, todos los textos tienen traducciones, únicamente hay que programar el botón para el cambio de idioma.
- **Gestión de stock avanzada y alertas**: Incluir lógica para mostrar al usuario avisos cuando el stock esté bajo, permitir activar alertas para reposiciones y bloquear la compra cuando el producto esté agotado.

En conclusión, el proyecto ya ofrece una base sólida y funcional para su uso como plataforma de comercio electrónico, pero aún existen múltiples aspectos que pueden ser ampliados, optimizados o mejorados. Estas tareas futuras no solo permitirían acercar la aplicación a un entorno real de producción y comercialización, sino que también representan oportunidades para seguir aplicando y profundizando en conocimientos técnicos, buenas prácticas de desarrollo y criterios de accesibilidad y usabilidad.

9. Bibliografía

Referencias

- [1] AI Adex. Database Migrations with Alembic and FastAPI: A Comprehensive Guide Using Poetry. Consulta: 20 de Marzo de 2025. <https://adex.ltd/database-migrations-with-alembic-and-fastapi-a-comprehensive-guide-using-poetry>.
- [2] Atlassian. Historias de usuario con ejemplos y plantilla. Consulta: 10 de Marzo de 2025. <https://www.atlassian.com/es/agile/project-management/user-stories>.
- [3] OpenAI. ChatGPT-Language Model by OpenAI [página web]. Múltiples consultas durante el desarrollo del proyecto. <https://chat.openai.com/>.
- [4] Universitat Oberta de Catalunya. Diagrama de flujo de navegación UOC. Consulta: 27 de Marzo de 2025. https://cv.uoc.edu/UOC/a/moduls/90/90_574b/web/main/m5/c4/3.html.
- [5] Documentación de Flutter (docs.flutter.dev). Implementación de la internacionalización. Consulta: 25 de Marzo de 2025. <https://docs.flutter.dev/ui/accessibility-and-internationalization/internationalization>.
- [6] Google Fonts. Iconos para la memoria. Consulta: 30 de Mayo de 2025. <https://fonts.google.com/icons?selected=Material+Symbols+Outlined:bookmark:FILL@0;wght@400;GRAD@0;opsz@24&icon.size=24&icon.color=%23000000&icon.query=book>.
- [7] Just in Mind. Diseño de navegación: Casi todo lo que necesitas saber. Consulta: 27 de Marzo de 2025. <https://www.justinmind.com/es/blog/navegacion-diseno-casi-todo-lo-que-necesitas-saber/>.
- [8] Medium. Agile Testing: de historias de usuario, criterios de aceptación y cómo escribir una historia testeable. Consulta: 10 de Marzo de 2025. <https://medium.com/@wc.testing.qa/agile-testing-de-historias-de-usuario-criterios-de-aceptaci%C3%B3n-y-c%C3%B3mo-escribir-una-historia-2a94149ecec7>.
- [9] Medium. Deploying Flutter Web with Docker. Consulta: 10 de Mayo de 2025. <https://medium.com/@vosarat1995/deploying-flutter-web-with-docker-50ae07d9d7e3>.
- [10] Medium. Flutter web with Docker Compose. Consulta: 10 de Mayo de 2025. <https://medium.com/@codemax120/flutter-web-with-docker-06cee1839adb>.

- [11] Medium. Getting Started with Flutter Map. Consulta: 16 de Abril de 2025. <https://raphaeldelio.medium.com/getting-started-with-flutter-map-9cf4113f22e9>.
- [12] Medium. How to deploy Flutter WebApp using Google AppEngine. Consulta: 11 de Mayo de 2025. <https://teresa-wu.medium.com/host-flutter-webapp-with-google-appengine-bc3de7e56ba2>.
- [13] Medium. How to deploy Flutter WebApp using Google Cloud Run. Consulta: 11 de Mayo de 2025. <https://medium.com/tide-engineering-team/deploy-flutter-webapp-using-gcp-cloud-run-4511c3d87e92>.
- [14] Medium. How To Host Flutter Web In Linux Using Nginx. Consulta: 11 de Mayo de 2025. <https://medium.com/flutter-community/how-to-host-flutter-using-nginx-a71bcb11d96>.
- [15] Medium. How to use alembic for your database migrations? Consulta: 20 de Marzo de 2025. <https://simplyprashant.medium.com/how-to-use-alembic-for-your-database-migrations-d3e93cacf9e8>.
- [16] Medium. Map in Dart/Flutter. Consulta: 16 de Abril de 2025. <https://medium.com/@emanyaqoob/map-in-dart-flutter-adf80a0299b5>.
- [17] Medium. Unlocking the Power of Flutter Maps: A Beginner's Roadmap. Consulta: 16 de Abril de 2025. <https://medium.com/@moinkhan603/unlocking-the-power-of-flutter-maps-a-beginners-roadmap-d728fd6e9c5f>.
- [18] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Clean Architecture. Consulta: 23 de Marzo de 2025. <https://medium.com/@yamen.abd98/clean-architecture-in-flutter-mvvm-bloc-dio-79b1615530e1>.
- [19] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Bloc Test. Consulta: 25 de Marzo de 2025. https://pub.dev/packages/bloc_test.
- [20] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Build Runner. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/build_runner.
- [21] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Build Web Compilers. Consulta: 17 de Abril de 2025. https://pub.dev/packages/build_web_compilers.
- [22] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Dio. Consulta: 25 de Marzo de 2025. <https://pub.dev/packages/dio>.
- [23] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Easy Localizations. Consulta: 28 de Marzo de 2025. https://pub.dev/packages/easy_localization.

- [24] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Equatable. Consulta: 1 de Abril de 2025. <https://pub.dev/packages/equatable>.
- [25] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete File Picker. Consulta: 10 de Abril de 2025. https://pub.dev/packages/file_picker.
- [26] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Flutter Bloc. https://pub.dev/packages/flutter_bloc. Consulta: 25 de Marzo de 2025. https://pub.dev/packages/flutter_bloc.
- [27] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Flutter HTML. Consulta: 15 de Abril de 2025. https://pub.dev/packages/flutter_html.
- [28] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Flutter Lints. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/flutter_lints.
- [29] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Flutter Localizations. Consulta: 25 de Marzo de 2025. https://pub.dev/packages/flutter_localization.
- [30] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Flutter Map. Consulta: 15 de Abril de 2025. https://pub.dev/packages/flutter_map.
- [31] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Flutter Map Cancellable Tile Provider. Consulta: 16 de Abril de 2025. https://pub.dev/packages/flutter_map_cancellable_tile_provider.
- [32] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Flutter SVG. Consulta: 25 de Marzo de 2025. https://pub.dev/packages/flutter_svg.
- [33] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Font Awesome Flutter. Consulta: 30 de Marzo de 2025. https://pub.dev/packages/font_awesome_flutter.
- [34] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Freezed. Consulta: 24 de Marzo de 2025. <https://pub.dev/packages/freezed>.
- [35] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Freezed Annotation. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/freezed_annotation.
- [36] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Go Router. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/go_router.

- [37] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete HTTP Parser. Consulta: 15 de Abril de 2025. https://pub.dev/packages/http_parser.
- [38] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Intl. Consulta: 24 de Marzo de 2025. <https://pub.dev/packages/intl>.
- [39] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete intl utils. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/intl_utils.
- [40] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete JSON Annotation. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/json_annotation.
- [41] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete JSON serializable. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/json_serializable.
- [42] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Lat Long 2. Consulta: 15 de Abril de 2025. <https://pub.dev/packages/latlong2>.
- [43] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Marquee. Consulta: 27 de Mayo de 2025. <https://pub.dev/packages/marquee>.
- [44] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Shared Preferences. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/shared_preferences.
- [45] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete Smooth Page Indicator. Consulta: 24 de Marzo de 2025. https://pub.dev/packages/smooth_page_indicator.
- [46] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete URL Launcher. Consulta: 20 de Mayo de 2025. https://pub.dev/packages/url_launcher.
- [47] Repositorio oficial de paquetes para aplicaciones Dart y Flutter (pub.dev). Paquete URL Strategy. Consulta: 20 de Mayo de 2025. https://pub.dev/packages/url_strategy.
- [48] Stack Overflow. Flutter open WhatsApp with DrawerListTile. Consulta: 20 de Mayo de 2025. [https://es.stackoverflow.com/questions/498053/como-agregar-un-bot%
c3%
b3n-de-whatsapp-en-flutter-con-drawerlisttile](https://es.stackoverflow.com/questions/498053/como-agregar-un-bot%c3%b3n-de-whatsapp-en-flutter-con-drawerlisttile).

- [49] Stack Overflow. Flutter open WhatsApp with text message. Consulta: 20 de Mayo de 2025. <https://stackoverflow.com/questions/60947937/flutter-open-whatsapp-with-text-message>.
- [50] Stack Overflow. Flutter web built release but NGINX route 404. Consulta: 11 de Mayo de 2025. <https://stackoverflow.com/questions/67621319/flutter-web-built-release-but-nginx-route-404>.
- [51] Pexels. Chic Brown Leather Handbag on Wooden Bench. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/chic-brown-leather-handbag-on-wooden-bench-32283144/>.
- [52] Pexels. Imagen Ropa. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/blonde-woman-posing-in-black-crop-top-with-rhinestone-fringes-and-blue-denim-maxi-skirt-18193926/>.
- [53] Pexels. Imagen Ropa. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/slim-young-woman-in-crop-top-posing-in-studio-18490987/>.
- [54] Pexels. Imagen Ropa. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/es-es/foto/mujer-modelo-maqueta-en-pie-21637707/>.
- [55] Pexels. Imagen Ropa. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/es-es/foto/madera-ciudad-paisaje-naturaleza-17245485/>.
- [56] Pexels. Imagen Ropa. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/es-es/foto/madera-ciudad-paisaje-naturaleza-17243573/>.
- [57] Pexels. Imagen Ropa. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/es-es/foto/madera-ciudad-paisaje-naturaleza-15647539/>.
- [58] Pexels. Imagen Ropa. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/es-es/foto/madera-ciudad-paisaje-naturaleza-15647590/>.
- [59] Pexels. Person Carrying Black Leather Handbag. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/person-carrying-black-leather-handbag-1374910/>.
- [60] Pexels. Person Wearing Chain Bracelet Holding Bag. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/person-wearing-chain-bracelet-holding-bag-8989582/>.
- [61] Pexels. Shoes. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/elegant-woman-wearing-black-high-heel-sandals-32285300/>.
- [62] Pexels. Shoes. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/a-woman-wearing-animal-print-stiletto-heels-8980188/>.
- [63] Pexels. Shoes. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/kickin-27505763/>.

- [64] Pexels. Shoes. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/person-with-high-heels-and-wine-bottle-11991550/>.
- [65] Pexels. Shoes. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/women-cowboy-boots-in-sunflowers-17965011/>.
- [66] Pexels. Shoes. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/close-up-on-womans-legs-on-car-park-10084175/>.
- [67] Pexels. Shoes. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/food-stairs-light-fashion-6669413/>.
- [68] Pexels. Shoes. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/photo-of-a-person-putting-on-black-boots-6862124/>.
- [69] Pexels. Woman Holding Brown Leather Bag. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/woman-holding-brown-leather-bag-26316180/>.
- [70] Pexels. Woman Wearing Black Open-toe Stilettos. Consulta: 1 de Mayo de 2025. <https://www.pexels.com/photo/woman-wearing-black-open-toe-stilettos-2090785/>.
- [71] Python poetry. Poetry. Consulta: 20 de Marzo de 2025. <https://python-poetry.org/>.
- [72] Unplash. Earrings. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/woman-in-white-shirt-wearing-gold-necklace-UBeNYvk6ED0>.
- [73] Unplash. Earrings. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/woman-wearing-silver-and-white-pearl-earring-P6NiFTyI294>.
- [74] Unplash. Imagen Ropa. Consulta: 1 de Mayo de 2025. https://unsplash.com/photos/woman-in-white-long-sleeve-dress-wearing-black-sunglasses-_y6hMz1pJC0.
- [75] Unplash. Imagen Ropa. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/woman-in-black-long-sleeve-shirt-and-black-pants-standing-beside-red-wall-nSmkZ4AfL2M>.
- [76] Unplash. Necklace. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/woman-in-white-off-shoulder-dress-wearing-silver-necklace-WKD2vIe8Rb0>.
- [77] Unplash. Necklace. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/woman-in-white-off-shoulder-shirt-wearing-silver-necklace-ceSCZzjTReg>.
- [78] Unplash. Necklace. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/woman-in-gold-sleeveless-dress-wearing-gold-necklace-vbnxgj6cJg>.

- [79] Unplash. Necklace. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/woman-wearing-gold-necklace-and-white-shirt-6uv43p1ePrk>.
- [80] Unplash. Rings. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/three-gold-rings-in-a-white-box-wewzzx11ShI>.
- [81] Unplash. Rings. Consulta: 1 de Mayo de 2025. <https://unsplash.com/photos/person-wearing-gold-ring-and-gold-ring-pjrPWwwYx1I>.
- [82] ViewNext. Escritura de criterios de aceptación para una historia de usuario. Consulta: 10 de Marzo de 2025. <https://viewnext.usal.es/blog/escritura-de-criterios-de-aceptaci%C3%B3n-para-una-historia-de-usuario>.

A. Anexos

A.1. Tablas base de datos detalladas

A continuación, se detallan las tablas principales por escrito:

Tabla User:

- **id**: Identificador del usuario. Número entero que se autoincrementa.
- **user_image**: Imagen de perfil del usuario.
- **name**: Nombre del usuario. Cadena de caracteres de longitud máxima 100.
- **surname**: Apellidos del usuario. Cadena de caracteres de longitud máxima 100.
- **phone_number**: Número de teléfono del usuario. Cadena de caracteres de longitud máxima 15.
- **email**: Dirección de correo electrónico del usuario. Cadena de caracteres de longitud máxima 250.
- **password**: Contraseña cifrada del usuario. Cadena de caracteres sin longitud predefinida.
- **address**: Dirección física del usuario. Cadena de caracteres de longitud máxima 500.
- **birth_date**: Fecha de nacimiento del usuario. Dato con formato de fecha.

- **role**: Rol del usuario dentro del sistema (cliente, cliente_vip, gestor o administrador). Cadena de caracteres sin longitud predefinida.

Tabla Vip Client Data:

- **id**: Identificador de los datos del cliente VIP. Número entero que se autoincrementa.
- **percentage_discount**: Porcentaje de descuento aplicado al cliente VIP. Valor decimal, por defecto 20.0 %.
- **start_date**: Fecha de inicio del estado VIP. Dato con formato de fecha. Por defecto, el día en que se crea el cliente VIP.
- **expire_date**: Fecha de expiración del estado VIP. Dato con formato de fecha. Por defecto, un año después de la creación del cliente VIP.

Tabla Product:

- **id**: Identificador del producto. Número entero que se autoincrementa.
- **reference**: Referencia única del producto. Cadena alfanumérica de hasta 255 caracteres.
- **name**: Nombre del producto. Cadena de caracteres de longitud máxima 40.
- **description**: Descripción detallada del producto. Cadena de caracteres sin longitud fija.
- **unit_price**: Precio unitario del producto. Valor decimal.
- **category**: Categoría a la que pertenece el producto. Cadena de caracteres.
- **type**: Tipo específico del producto. Cadena de caracteres.
- **color**: Color predominante del producto. Cadena de caracteres.
- **visibility**: Estado de visibilidad del producto (visible/no visible). Cadena de caracteres.
- **creation_date**: Fecha de creación del producto en el sistema. Dato con formato de fecha. Por defecto, el día en que se crea el producto.
- **front_img**: Imagen principal del producto.
- **size_group_name**: Nombre del grupo de tallas asociado al producto. Cadena de caracteres.

Tabla Clothing Set:

- **id**: Identificador del conjunto. Número entero que se autoincrementa.
- **name**: Nombre del conjunto de prendas. Cadena de caracteres de longitud máxima 100.
- **description**: Descripción del conjunto. Cadena de caracteres sin longitud fija.
- **price**: Precio total del conjunto. Valor decimal.
- **visibility**: Estado de visibilidad del conjunto. Cadena de caracteres.
- **creation_date**: Fecha de creación del conjunto. Dato con formato de fecha. Por defecto, el día en que se crea el conjunto.
- **image**: Imagen representativa del conjunto.

Tabla Size:

- **id**: Identificador de la talla. Número entero que se autoincrementa.
- **product_id**: Identificador del producto al que pertenece la talla. Número entero.
- **size_name**: Nombre o código de la talla (por ejemplo, S, M, L). Cadena de caracteres de hasta 20 caracteres.
- **stock**: Cantidad disponible de esa talla en inventario. Número entero.

Tabla Order:

- **id**: Identificador del pedido. Número entero que se autoincrementa.
- **user_id**: Identificador del usuario que realizó el pedido. Número entero.
- **total_price**: Precio total del pedido. Valor decimal.
- **creation_date**: Fecha en la que se creó el pedido. Dato con formato de fecha. Por defecto, el día en que se crea el pedido.
- **state**: Estado actual del pedido. Cadena de caracteres.

Tabla Order Detail:

- **id**: Identificador del detalle del pedido. Número entero que se autoincrementa.
- **order_id**: Identificador del pedido asociado. Número entero.
- **quantity**: Cantidad de unidades del producto que se han comprado. Número entero.
- **unit_price**: Precio unitario del producto en el momento del pedido. Valor decimal.

Tabla Comments:

- **id**: Identificador del comentario. Número entero que se autoincrementa.
- **user_id**: Identificador del usuario que hizo el comentario. Número entero.
- **user_name**: Nombre del usuario que hizo el comentario. Cadena de caracteres.
- **product_id**: Identificador del producto comentado. Número entero.
- **comment**: Texto del comentario realizado. Cadena de caracteres.
- **posted_date**: Fecha en la que se publicó el comentario. Dato con formato de fecha. Por defecto, el día en que se publica.
- **edited_date**: Fecha de la última edición del comentario, si se ha editado. Dato con formato de fecha.

Tabla Ratings:

- **id**: Identificador de la valoración. Número entero que se autoincrementa.
- **user_id**: Identificador del usuario que realizó la valoración. Número entero.
- **product_id**: Identificador del producto valorado. Número entero.
- **rating**: Valor de la puntuación otorgada al producto. Número entero entre 1 y 5.

En el diagrama se pueden observar tablas intermedias. A continuación, se mencionan junto con sus respectivas funcionalidades:

- **Clothing Set Association**: Esta tabla intermedia tiene el propósito de almacenar la lista de productos que pertenecen a un mismo conjunto de ropa.

- **User Favorites:** Esta tabla permite registrar qué productos están en la lista de favoritos de un usuario en particular.
- **Order Detail Product** y **Order Detail Size:** Estas dos tablas se utilizan para almacenar los productos y las tallas asociadas a cada detalle de pedido.