

## ***ArteFatto***

**Autor:** Daniel Navarro Vázquez

**Tutor:** Javier Martín Rivero

**Fecha de entrega:** 21/05/2025

**Convocatoria:** 2024 2025

## Índice

Índice.....	1
Introducción.....	2
Motivación.....	3
Abstract.....	3
Objetivos Propuestos.....	4
Generales.....	4
Específicos.....	6
Metodología.....	8
Tecnologías usadas.....	9
Estimación de recursos y planificación.....	11
Análisis.....	12
Requisitos Funcionales(RF).....	12
Requisitos No Funcionales (RNF).....	14
Diseño.....	17
Diagrama de Clases.....	21
Diagrama de Casos de Uso.....	22
Valores Límite.....	25
Pruebas de Caja Negra.....	26
Conclusiones.....	26
Vías Futuras.....	27
Glosario.....	27
Bibliografía.....	30

## Introducción

En el mundo actual, los artistas y manufactureros enfrentan un reto importante: conectar con su público en un mercado saturado por grandes plataformas como Temu, Shein o Aliexpress. Estas plataformas, aunque accesibles, tienden a eclipsar el trabajo artesanal y único, dificultando a los creadores emergentes destacar y llegar a una audiencia más amplia.

Este trabajo final de grado tiene como objetivo desarrollar una tienda en línea que actúe como un puente entre los artistas –tanto experimentados como novatos– y su público. La aplicación busca facilitar a los creadores mostrar y poner a la venta sus productos, promoviendo una experiencia accesible y enriquecedora que resalta la autenticidad y el valor de lo hecho a mano.

Para lograr este objetivo, adopté una metodología dividida en tres etapas fundamentales: **"Lo entiendo, lo veo y funciona"**.

1. **Lo entiendo:** En esta primera fase, se define qué se quiere lograr y cómo llevarlo a cabo, a través de un análisis profundo y la recopilación de información clave.
2. **Lo veo:** Una vez clara la visión, se desarrolla un modelo visual de la aplicación que, aunque no funcional, permite organizar las ideas y establecer una base sólida para avanzar.
3. **Funciona:** Finalmente, se implementa la funcionalidad del sistema, resolviendo los problemas técnicos que surjan para garantizar una experiencia fluida y efectiva.

Si bien el desarrollo de esta plataforma enfrenta desafíos, como la competencia con grandes marcas y los problemas técnicos inherentes al diseño y despliegue de aplicaciones, este trabajo explora soluciones viables para crear un espacio digital que celebre el arte y la creatividad.

Este trabajo se desarrolla en torno a tres etapas principales que guían el diseño y la construcción de la aplicación. En primer lugar, se incluye una sección dedicada a la recopilación de información y la creación de esquemas que reflejan el muestreo inicial de la aplicación.

A continuación, se presenta una fase enfocada en el diseño visual. En esta etapa, se elaboran MockUps que ilustran el aspecto y la experiencia de usuario deseados para la aplicación. Estos diseños permiten transmitir una visión clara del resultado final y facilitan el proceso de desarrollo.

Finalmente, el trabajo concluye con una parte centrada en la implementación de las funcionalidades del código. Aquí se analizan los desafíos técnicos encontrados durante el desarrollo, junto con las soluciones aplicadas para resolverlos.

## Motivación

Lo que me impulsó para decidir este proyecto fueron mis amigos y mis propios proyectos personales. Soy un coleccionista de barajas de cartas con diseños exóticos y decorativos y decidí empezar a crear mis propios diseños. Diseño barajas, las imprimo y las recorto para crear barajas únicas con mi marca.

Junto con dos buenos amigos, los cuales también usan algo de su tiempo para crear diseños y productos manufacturados, decidimos pagar por un espacio en una página para subir y vender nuestros productos.

Entonces pensé *“¿Y si diseñamos nuestra propia plataforma donde vender nuestros trabajos?”*, y de ahí surgió esta idea.

## Abstract

This final degree project aims to develop an online store that connects artists and emerging creators with their audience, offering an accessible and ethical platform for selling handmade products. In a digital market dominated by large-scale platforms such as Temu, Shein, and Aliexpress, this initiative seeks to provide an alternative space that values authenticity, creativity, and the uniqueness of handcrafted goods. The platform's mission is to create a community-driven environment that supports local talent and promotes responsible consumption, offering users a more personal and meaningful shopping experience.

The development of the application is structured into three key phases: the first involves analysis and information gathering ("I understand"), where market needs and

user expectations are identified. The second focuses on designing an initial visual model ("I see"), establishing the foundation for an intuitive and aesthetically appealing interface. The third and final phase centers on the implementation and testing of the system's core functionalities ("It works"), ensuring a stable and user-friendly platform.

This project also addresses technical challenges such as scalability, performance optimization, and user security. Furthermore, it proposes strategic solutions to compete in a saturated market, positioning itself as a digital space that celebrates individuality, craftsmanship, and artistic expression.

## Objetivos Propuestos

### Generales

#### **Registro e Inicio de sesión para el cliente:**

- **Objetivo:** Permitir a los usuarios crear cuentas personales en la plataforma y acceder a sus perfiles de manera segura mediante autenticación.
- **Descripción:** Este objetivo implica la implementación de un sistema de registro que permita a los usuarios introducir su información personal (nombre, correo electrónico, dirección, etc.) y contraseñas seguras. El sistema debe ofrecer un mecanismo de inicio de sesión mediante autenticación de usuario y contraseña, asegurando la privacidad y seguridad de los datos. Asimismo, se implementarán validaciones para garantizar que el proceso de registro cumpla con los estándares de seguridad, como la protección contra ataques de inyección SQL y contraseñas débiles.

#### **Guardar y retirar artículos del carrito:**

- **Objetivo:** Facilitar a los usuarios la capacidad de añadir, retirar o modificar artículos dentro de su carrito de compras virtual.
- **Descripción:** Los usuarios podrán navegar por la tienda, seleccionar productos, y agregarlos a su carrito. Se deberá permitir también modificar la cantidad de los artículos seleccionados, eliminar productos del carrito y ver un resumen del contenido antes de proceder al pago. Además, se deberá calcular

automáticamente el precio total de la compra, incluyendo impuestos y posibles descuentos. El sistema también debería ser capaz de guardar los elementos del carrito para que el usuario pueda continuar su compra en futuras sesiones.

**Navegar por la tienda para buscar productos:**

- **Objetivo:** Implementar un sistema de navegación intuitivo y eficiente para que los usuarios puedan explorar y buscar productos según sus preferencias (categorías, filtros, búsqueda por nombre).
- **Descripción:** El diseño de la tienda debe permitir a los usuarios navegar fácilmente entre diferentes categorías de productos, con la opción de aplicar filtros como precio, popularidad, o disponibilidad. Además, se implementará una barra de búsqueda avanzada para que los usuarios puedan encontrar productos específicos por nombre, descripción o características clave. La experiencia de usuario debe ser fluida, permitiendo búsquedas rápidas y eficientes.

**Subir productos:**

- **Objetivo:** Desarrollar un sistema que permita a los artistas o vendedores subir sus productos a la plataforma de manera sencilla y rápida.
- **Descripción:** Este objetivo implica crear una interfaz de usuario donde los artistas o administradores puedan subir imágenes, descripciones y precios de sus productos, asignarlos a una categoría específica, y definir su disponibilidad. También se incluirá una opción para que los productos puedan ser modificados o eliminados, permitiendo a los vendedores gestionar su inventario de manera eficiente.

## Específicos

- **Comprobación de caracteres especiales:**
  - **Objetivo:** Implementar un sistema de validación para asegurar que los usuarios ingresen datos válidos, especialmente para campos como nombre, correo electrónico y contraseñas.
  - **Descripción:** Para garantizar la seguridad y el correcto funcionamiento del sistema, se deberán implementar validaciones para los campos de entrada, prohibiendo caracteres especiales no permitidos. Por ejemplo, el nombre del producto, del usuario y otros campos no deberían contener caracteres especiales como "<", ">", o "&", que podrían ser utilizados en ataques de inyección. Además, se debe verificar que las contraseñas sigan ciertas reglas de seguridad, como un mínimo de caracteres y la inclusión de mayúsculas, minúsculas, números y caracteres especiales.
- **Confirmación de contraseña:**
  - **Objetivo:** Asegurar que el usuario ingrese correctamente su contraseña durante el proceso de registro o cambio de contraseña.
  - **Descripción:** Para evitar errores al ingresar la contraseña, se implementará un sistema que permita a los usuarios confirmar su contraseña durante el proceso de registro o cuando decidan cambiarla. Esta confirmación debe ser obligatoria para evitar que el usuario se registre con contraseñas erróneas o inintencionadas. Se aplicarán reglas de seguridad que aseguren que la contraseña es suficientemente fuerte y se almacenará de manera segura, utilizando técnicas como el hashing.

- **Validación de campos obligatorios y formatos:**
  - **Objetivo:** Garantizar que todos los campos obligatorios durante el registro y la subida de productos sean completados correctamente, y en el formato adecuado.
  - **Descripción:** Durante el registro del usuario y el proceso de carga de productos, se debe validar que se completen todos los campos obligatorios como el nombre, el correo electrónico, la descripción del producto y otros datos clave. Además, se comprobará que los formatos de los campos, como el correo electrónico, sean correctos. Los productos también deben tener imágenes de un tamaño y resolución adecuados para garantizar una experiencia visual atractiva y funcional en la tienda online.
- **Implementación de un sistema de notificación de errores y confirmaciones:**
  - **Objetivo:** Proveer de un sistema de retroalimentación al usuario para indicarle si los datos han sido correctamente ingresados o si ha habido algún error.
  - **Descripción:** Es esencial que el sistema muestre mensajes claros y comprensibles cuando un usuario haya cometido un error en la introducción de datos, como contraseñas no coincidentes, campos vacíos o con formato incorrecto, o productos sin categoría asignada. Estos mensajes de error deben ser específicos y fáciles de entender, mientras que las confirmaciones de éxito (como "Producto subido correctamente") también deben ser claras y visibles para garantizar una experiencia de usuario fluida.



- **Integración de métodos de pago seguros:**
  - **Objetivo:** Asegurar que el sistema permite realizar pagos de manera segura y eficiente, protegiendo la información del usuario.
  - **Descripción:** El sistema de pago debe permitir la integración de métodos como tarjetas de crédito, transferencias bancarias o servicios de pago como PayPal. Se implementará el cifrado de datos y el uso de conexiones HTTPS para proteger las transacciones. Además, se debe validar que los pagos sean procesados correctamente y mostrar un mensaje de confirmación al usuario una vez completada la transacción.

## Metodología

Uso la metodología SCRUM para gestionar el desarrollo de mi tienda online de manera ágil y organizada, adaptándola a mi trabajo individual en mi Trabajo de Fin de Grado (TFG). Aunque SCRUM suele aplicarse en equipos, ajusto sus principios para optimizar mi flujo de trabajo y mejorar la productividad.

Como único responsable del proyecto, asumo todos los roles:

- **Product Owner:** Defino las funcionalidades clave, como el catálogo de productos, el carrito de compras y el sistema de pago.
- **Scrum Master:** Me aseguro de seguir la metodología y evitar bloqueos.
- **Desarrollador:** Diseño, programo y pruebo cada funcionalidad de la tienda.

Para organizarme, creo un **Backlog** del Producto, donde priorizo tareas como el diseño de la interfaz, la integración de pasarelas de pago y la optimización SEO. Antes de cada sprint (que dura entre 1 y 2 semanas), hago una **Sprint Planning** para definir los objetivos. Diariamente, reviso mi progreso y ajusto tareas según sea necesario. Al final de cada sprint, realizo una **Sprint Review** para evaluar lo desarrollado y una Retrospectiva para identificar mejoras en mi proceso de trabajo.

Gracias a esta adaptación de SCRUM, puedo mantener un ritmo constante, estructurar mejor mis tareas y desarrollar mi tienda online de manera eficiente y organizada.

## Tecnologías usadas

En el desarrollo de este proyecto, he empleado diversas tecnologías y herramientas que me han permitido construir una tienda online eficiente y estructurada. A continuación, detallo cada una de ellas junto con las razones que motivaron su elección:

- **IntelliJ IDEA:** He utilizado este entorno de desarrollo integrado (IDE) por su compatibilidad con Java, su potente sistema de autocompletado y depuración, además de las herramientas integradas que facilitan la gestión de proyectos con Maven y bases de datos. También se destaca su integración con sistemas de control de versiones como Git, lo que facilita la gestión del código fuente a lo largo del desarrollo y permite un mejor seguimiento de los cambios.
- **JavaFX:** Para la interfaz gráfica, he elegido JavaFX debido a su flexibilidad y capacidades avanzadas en el desarrollo de aplicaciones de escritorio. Permite crear interfaces modernas con CSS y FXML, lo que mejora la organización y el diseño del código. Además, su compatibilidad multiplataforma asegura que la aplicación sea accesible en diferentes sistemas operativos (Windows, macOS y Linux), garantizando una experiencia de usuario consistente y accesible.
- **Hibernate:** Como framework de persistencia, Hibernate simplifica la gestión de la base de datos en Java mediante el uso de ORM (Object-Relational Mapping). Su capacidad para manejar consultas de manera eficiente y evitar la escritura manual de SQL lo hace ideal para este proyecto. Hibernate también incluye funcionalidades de caching, que optimizan el rendimiento al reducir la carga de la base de datos, y ofrece soporte para transacciones, lo que garantiza la integridad y seguridad de las operaciones de datos.
- **MySQL:** He optado por MySQL como sistema de gestión de bases de datos relacional (RDBMS) por su estabilidad, seguridad y facilidad de integración con Hibernate. Además, su amplio soporte y documentación permiten una administración eficiente de los datos de la tienda. MySQL también es ampliamente conocido por su capacidad para manejar grandes volúmenes de datos, lo que asegura que el sistema pueda escalar sin comprometer el

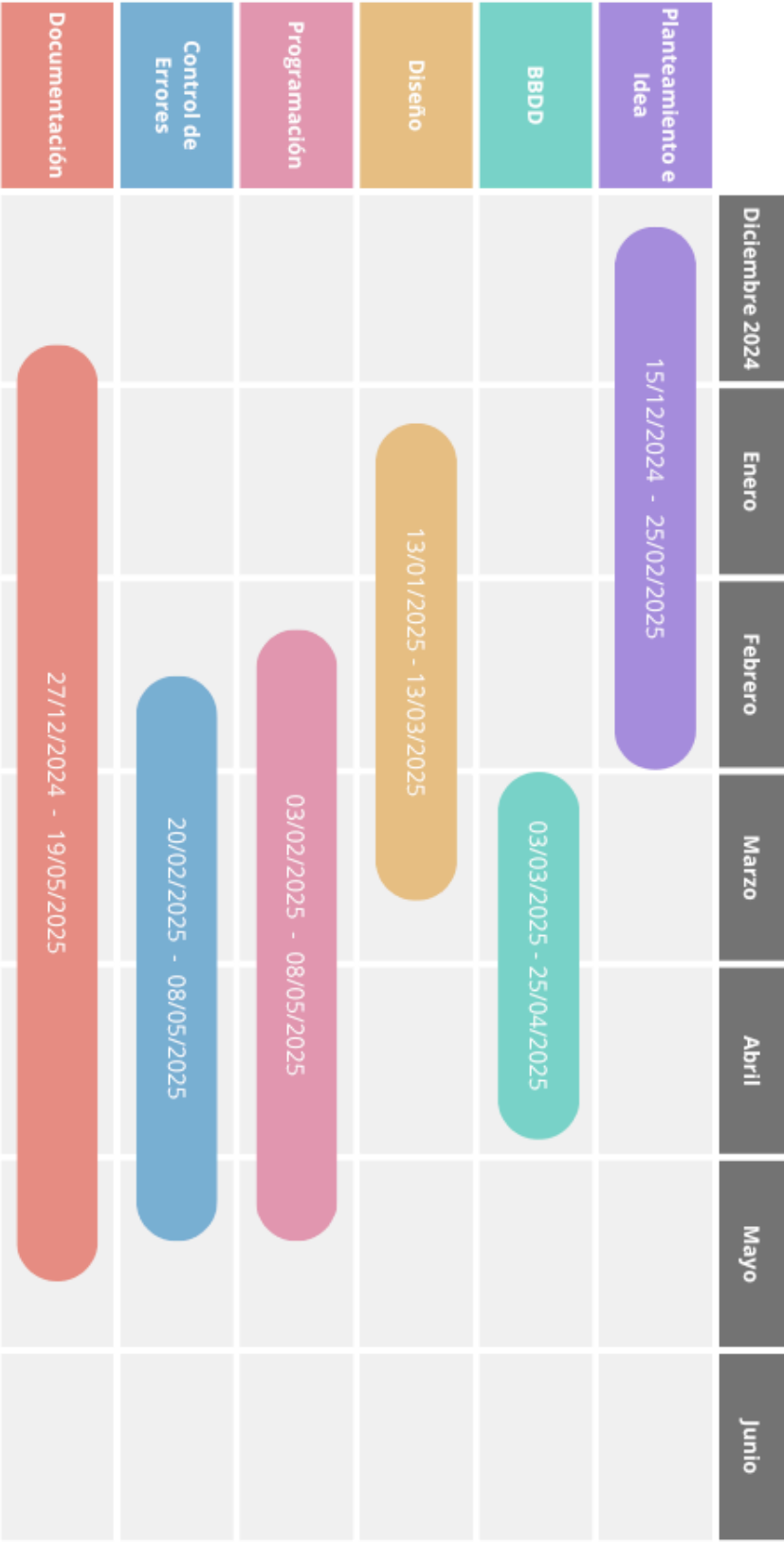
rendimiento.

- **Otras tecnologías y herramientas:**

- **Git:** Para la gestión de versiones del código y el seguimiento de cambios, utilicé **Git**. Esto me permitió mantener un registro detallado de las modificaciones, revertir cambios cuando fuera necesario y colaborar con otros miembros si se hubiera trabajado en equipo.
- **JUnit:** Para asegurar la calidad del código y garantizar que las funcionalidades de la tienda operaran correctamente, implementé pruebas unitarias utilizando **JUnit**, permitiendo detectar errores antes de que el sistema se desplegara.

La elección de estas tecnologías se basa en su robustez, eficiencia y compatibilidad entre sí, lo que facilita el desarrollo de una tienda online funcional, segura y escalable. Cada herramienta ha sido seleccionada no solo por su fiabilidad, sino también por la posibilidad de mejorar la eficiencia del desarrollo y la calidad del producto final.

Estimación de recursos y planificación



## Análisis

### Requisitos Funcionales(RF)

- **RF01 - Gestión de usuarios**
  - RF01.1: Permitir el registro de nuevos usuarios mediante la introducción de datos personales (nombre, email, contraseña, etc.).
  - RF01.2: Validar que las contraseñas ingresadas coincidan antes de registrar al usuario.
  - RF01.3: Comprobar la existencia previa del usuario para evitar duplicados.
  - RF01.4: Permitir iniciar sesión mediante un sistema de autenticación (usuario/contraseña).
  - RF01.5: Posibilidad de iniciar sesión como invitado para navegación básica sin registro.
  - RF01.6: Permitir a los usuarios editar su información personal (nombre, contraseña, email).
  - RF01.7: Permitir a los usuarios eliminar su cuenta de forma permanente.
  - RF01.8: Habilitar la activación o desactivación de cuentas de usuario por parte del sistema o un administrador.
- **RF02 - Gestión de productos**
  - RF02.1: Permitir a usuarios autorizados subir nuevos productos, incluyendo nombre, descripción, imagen y categoría.
  - RF02.2: Permitir la edición de información de productos existentes.

- RF02.3: Permitir la desactivación de productos (sin eliminar definitivamente).
- RF02.4: Visualización de un catálogo de productos con cards informativas (imagen, nombre, precio).
- RF02.5: Filtrar productos por categorías.
- RF02.6: Buscar productos por nombre o palabra clave.
- **RF03 - Carrito de compras**
  - RF03.1: Permitir a los usuarios agregar productos al carrito desde el catálogo.
  - RF03.2: Permitir eliminar productos del carrito.
  - RF03.3: Mostrar al usuario un resumen de su carrito, incluyendo nombre de producto, cantidad, precio unitario y total.
  - RF03.4: Calcular y mostrar automáticamente el total de la compra antes de proceder al pago.
- **RF04 - Gestión de pedidos**
  - RF04.1: Permitir realizar una compra (checkout) desde el carrito.
  - RF04.2: Registrar el pedido en la base de datos, incluyendo datos del usuario, productos y total pagado.
  - RF04.3: Enviar una notificación de confirmación del pedido al correo electrónico del usuario.

## Requisitos No Funcionales (RNF)

- **RNF01 - Rendimiento**

- RNF01.1: La página principal debe cargar en menos de 3 segundos en condiciones normales.
- RNF01.2: Las búsquedas y filtros deben tener una latencia inferior a 2 segundos.
- RNF01.3: El sistema debe soportar al menos 100 usuarios concurrentes sin degradación significativa del rendimiento.

- **RNF02 - Seguridad**

- RNF02.1: Las contraseñas deben almacenarse en la base de datos con cifrado seguro (por ejemplo, hashing con sal).
- RNF02.2: Toda comunicación entre cliente y servidor debe realizarse mediante HTTPS.
- RNF02.3: Se debe implementar validación en ambos lados (cliente y servidor) para evitar inyecciones o accesos indebidos.

- **RNF03 - Usabilidad**

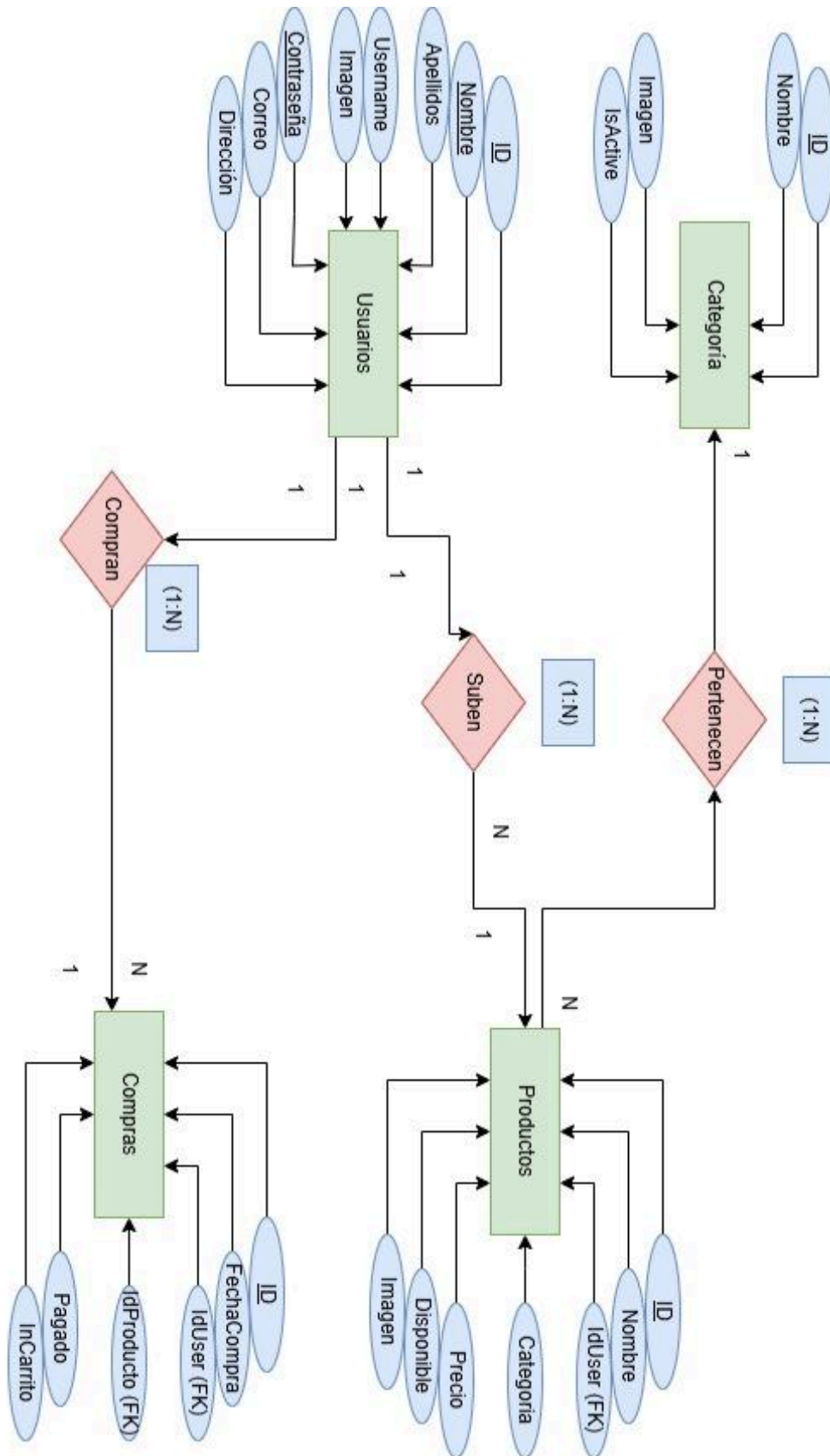
- RNF03.1: La interfaz debe ser clara e intuitiva, accesible para usuarios sin conocimientos técnicos.
- RNF03.2: Debe haber coherencia visual en botones, formularios y navegación.
- RNF03.3: El sistema debe ser responsive y funcionar correctamente en móviles, tablets y PC.

- **RNF04 - Mantenibilidad y escalabilidad**

- RNF04.1: El código fuente debe estar comentado y organizado para facilitar la comprensión y futuras modificaciones.
- RNF04.2: La arquitectura del sistema debe permitir la adición de nuevas categorías, productos o funcionalidades sin afectar el rendimiento.
- RNF04.3: La base de datos debe estar normalizada y optimizada para consultas frecuentes (como búsquedas y filtros).



• Entidad-Relación:



## Diseño

### Login Page



The login page features a dark purple header with a close button (X) on the left and the brand name 'Arte Fatto' in a rounded purple box on the right. The main content area is white with a light purple background. On the left, there is a pink paintbrush icon. On the right, there is a teal paint palette icon. The central form is titled 'Bienvenido' and contains two input fields: 'Nombre de Usuario' (Username) and 'Contraseña' (Password). Below these fields is a purple 'Entrar' (Enter) button. At the bottom of the form, there are two links: 'Registrate' and 'Invitado'.

### Register Page



The register page features a dark purple header with a back button (<) on the left and the brand name 'Arte Fatto' in a rounded purple box on the right. The main content area is white with a light purple background. On the left, there is a pink paintbrush icon. On the right, there is a teal paint palette icon. The central form is titled 'Registro' and contains six input fields arranged in two columns: 'Nombre' (Name), 'Apellidos' (Surname), 'Nombre de Usuario' (Username), 'Dirección' (Address), 'Contraseña' (Password), and 'Repetir Contraseña' (Repeat Password). Below these fields is a purple 'Registrarte' (Register) button. At the bottom of the form, there is a link: 'Correo' (Email).

Arte Fatto

Username

ID: 7126378216GHA  
unetearepo@gmail.com

Dirección:

Nombre:

Apellidos:

Correo:

Apodo:

Guardar Cambios

Descartar Cambios

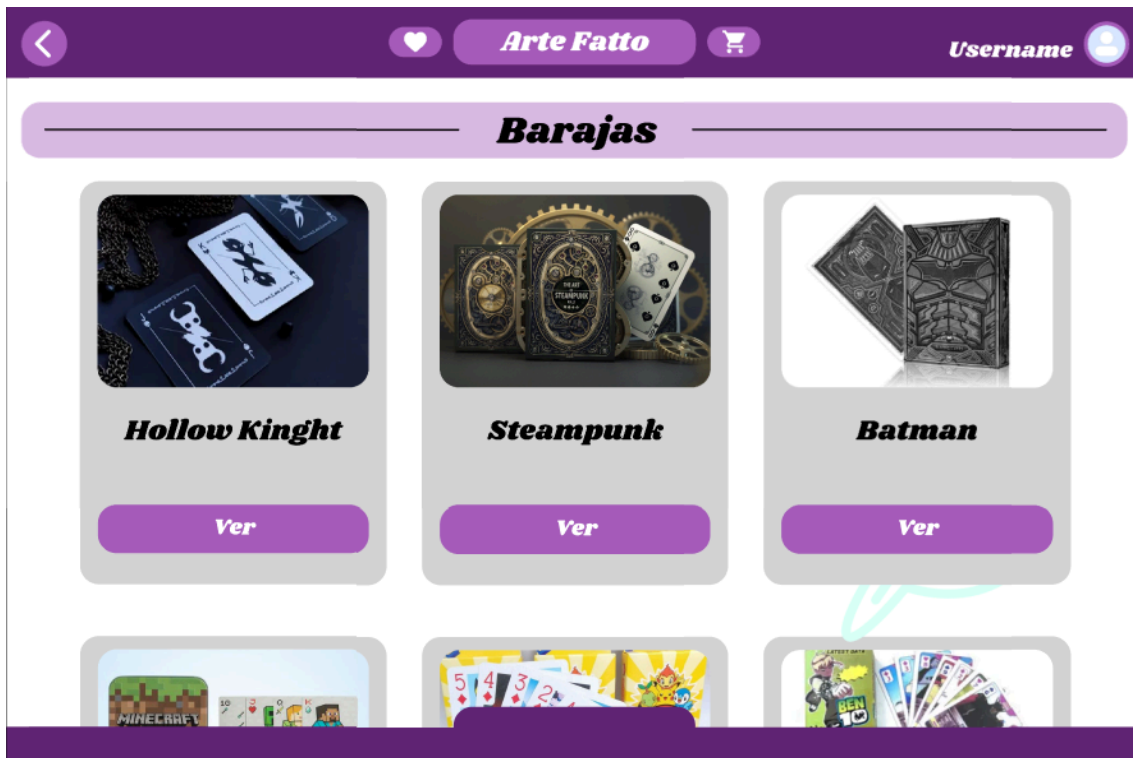
Cerrar Sesión

Eliminar Cuenta

## Main Shop Page



## Category Shop Page



## Products Page Shop



## Cart Shop Page

<     **Arte Fatto**     Username

<b>Baraja Hollow Knight - Edición coleccionista</b>	<b>x1</b>
<b>Ilustración Cuerpo Completo + Fondo</b>	<b>x24</b>

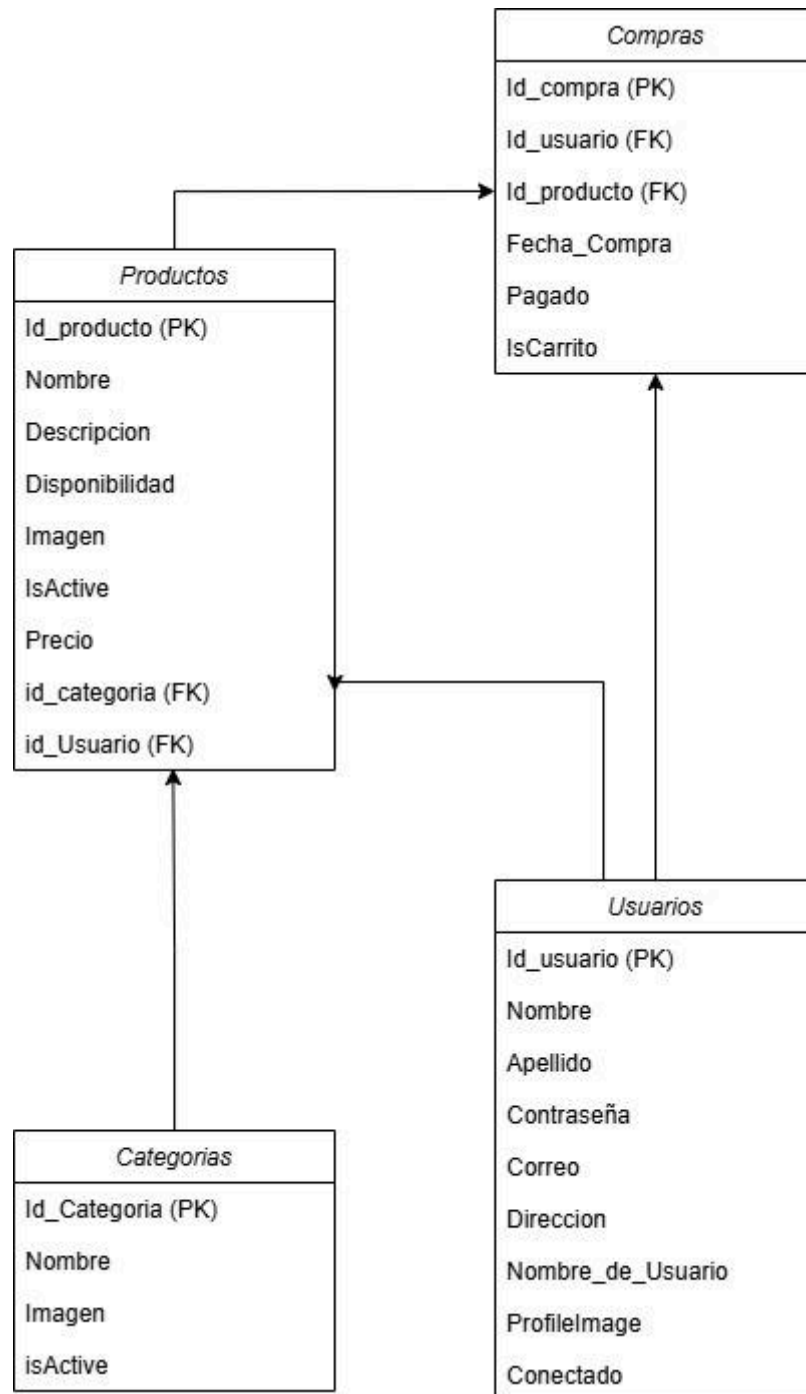
**Método de Pago**  
 \*\*\*\*\*7833

**Dirección de Cobro**  
 **C/ Ciencias 2090**

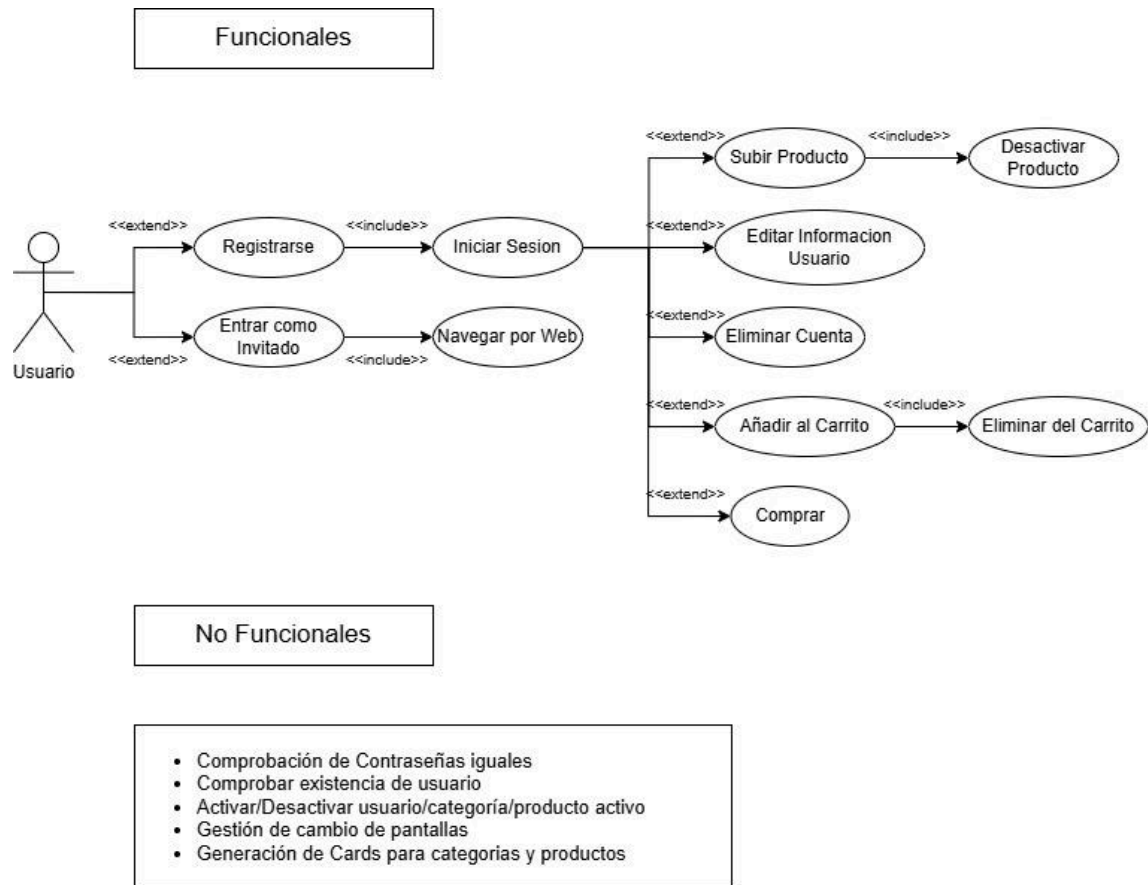
**Total:**  
**659,99€**

**Pagar**

## Diagrama de Clases



## Diagrama de Casos de Uso



Campo	Descripción
Caso de uso	Registrarse
Alias	-
Actores	User
Requisito Funcional	El usuario puede añadir sus datos para registrarse
Descripción	Permite al usuario crear una cuenta en la aplicación para poder acceder a todas las funcionalidades de la tienda.
Referencias	-
Comentarios	Sin comentarios

Campo	Descripción
Caso de uso	Iniciar Sesión
Alias	-
Actores	User
Requisito Funcional	El usuario registrado puede iniciar sesión con sus credenciales
Descripción	Permite al usuario acceder con su cuenta ya creada y tener acceso completo a la tienda.
Referencias	-
Comentarios	Sin comentarios

Campo	Descripción
Caso de uso	Entrar como Invitado
Alias	-
Actores	User
Requisito Funcional	El usuario puede navegar sin estar registrado
Descripción	Permite al usuario explorar la tienda sin necesidad de iniciar sesión o registrarse.
Referencias	-
Comentarios	Acceso limitado a funcionalidades.

Campo	Descripción
Caso de uso	Navegar por Web
Alias	-
Actores	User
Requisito Funcional	Permite al usuario explorar productos y categorías
Descripción	El usuario puede moverse por la tienda viendo productos, categorías y detalles.
Referencias	-
Comentarios	Disponible para usuarios registrados e invitados.

Campo	Descripción
Caso de uso	Subir Producto
Alias	-
Actores	User
Requisito Funcional	Permite añadir productos nuevos
Descripción	El usuario puede agregar un producto a la tienda proporcionando los datos requeridos.
Referencias	-
Comentarios	Puede requerir permisos especiales.



Campo	Descripción
Caso de uso	Desactivar Producto
Alias	-
Actores	User
Requisito Funcional	Permite ocultar productos de la tienda
Descripción	El usuario puede desactivar productos para que no aparezcan públicamente.
Referencias	-
Comentarios	Puede ser usado para productos sin stock.

Campo	Descripción
Caso de uso	Editar Información Usuario
Alias	-
Actores	User
Requisito Funcional	Permite modificar los datos del perfil
Descripción	El usuario puede actualizar sus datos personales, como nombre, correo, etc.
Referencias	-
Comentarios	Sin comentarios

Campo	Descripción
Caso de uso	Eliminar Cuenta
Alias	-
Actores	User
Requisito Funcional	Permite eliminar permanentemente la cuenta
Descripción	El usuario puede borrar su cuenta del sistema junto con sus datos asociados.
Referencias	-
Comentarios	Acción irreversible.

Campo	Descripción
Caso de uso	Añadir al Carrito
Alias	-
Actores	User
Requisito Funcional	Permite seleccionar productos para comprar más tarde
Descripción	El usuario puede añadir productos al carrito antes de hacer la compra.
Referencias	-
Comentarios	Sin comentarios

Campo	Descripción
Caso de uso	Eliminar del Carrito
Alias	-
Actores	User
Requisito Funcional	Permite quitar productos del carrito
Descripción	El usuario puede eliminar productos seleccionados antes de proceder con la compra.
Referencias	-
Comentarios	Sin comentarios

Campo	Descripción
Caso de uso	Comprar
Alias	-
Actores	User
Requisito Funcional	El usuario puede finalizar la compra
Descripción	Permite al usuario realizar el pago y completar la adquisición de los productos seleccionados.
Referencias	-
Comentarios	Puede incluir múltiples métodos de pago.

## Valores Límite

Clase	Campo	Valor mínimo válido	Valor máximo válido	Valor inválido ejemplo
Usuario	nombre	""	50 caracteres	>50 caracteres
	apellido	""	50 caracteres	>50 caracteres
	nombreUsuario	"abc"	20 caracteres	"us@" / "a"
	contrasena	8 caracteres	100+ (as needed)	"short"
	correo	"a@b.co"	50 caracteres	"noemail@"
	direccion	""	100 caracteres	>100 caracteres
	profileImage	null / ""	cualquier string	no aplica
	conectado	FALSE	TRUE	null (si no inicializa)

Clase	Campo	Valor mínimo válido	Valor máximo válido	Valor inválido ejemplo
Producto	nombre	"abc"	50 caracteres	"##nombre"
	descripcion	""	largo texto	null (si requerido)
	precio	0.01	Double.MAX_VALUE	null / -10.0
	disponible	FALSE	TRUE	null
	imagen	""	largo texto (ej. base64 img)	no aplica
	isActive	FALSE	TRUE	null
	usuario	objeto válido	objeto válido	null
	categoria	objeto válido	objeto válido	null

Clase	Campo	Valor mínimo válido	Valor máximo válido	Valor inválido ejemplo
Compras	id_usuario	usuario válido	usuario válido	null
	id_producto	producto válido	producto válido	null
	isPagado	FALSE	TRUE	null
	fecha_compra	fecha válida (ej. 1970-01-01)	fecha actual / futura	null / fecha no parseable

Clase	Campo	Valor mínimo válido	Valor máximo válido	Valor inválido ejemplo
Categoria	nombre	"abc"	50 caracteres	"##cat!" / 2 caracteres / >50
	imagen	null / ""	ruta larga o base64	no aplica
	isActive	FALSE	TRUE	null

## Pruebas de Caja Negra

Caso de Uso	Prueba de Caja Negra
Registrarse	Introducir contraseñas que no coinciden → Verificar error.
Registrarse	Introducir un correo ya registrado → Verificar mensaje de "usuario ya existe".
Iniciar sesión	Usuario correcto y contraseña correcta → Debe iniciar sesión exitosamente.
Iniciar sesión	Usuario incorrecto → Mostrar error de "usuario no encontrado".
Añadir al carrito	Añadir producto sin stock → Mostrar error de "producto no disponible".
Comprar	Comprar con carrito vacío → Bloquear compra y mostrar advertencia.
Subir producto	Subir producto sin imagen → Mostrar advertencia de campo requerido.
Editar información de usuario	Dejar campos obligatorios vacíos → Mostrar advertencia.

## Conclusiones

Después de meses de trabajo y dedicación, he llegado a una conclusión importante: independientemente del tipo de proyecto que se quiera realizar, si se aborda con ilusión y esfuerzo, es posible llevarlo a cabo. Este proyecto de desarrollo de una tienda local me ha permitido comprender la complejidad real que implica crear una aplicación completa, con muchos más retos de los que inicialmente fui capaz de anticipar.

A lo largo del proceso, he tenido que enfrentarme a numerosos desafíos, realizar cambios constantes y resolver errores persistentes. Sin embargo, todo ese esfuerzo ha valido la pena al ver el sistema en funcionamiento. Aunque soy consciente de que el proyecto aún no está finalizado tal y como me gustaría, me siento satisfecho con el trabajo realizado.

Considero que he aprendido mucho durante este camino, y también soy consciente de que aún tengo mucho por descubrir y mejorar. A pesar de ello, me siento orgulloso tanto del esfuerzo invertido como del resultado alcanzado.

## Vías Futuras

Este proyecto aún presenta múltiples áreas de mejora que podrían abordarse en el futuro. Entre ellas destacan el diseño de interfaces más limpias e intuitivas, la optimización de los tiempos de respuesta del sistema, la implementación de funcionalidades para la gestión de productos (como la creación y carga de nuevos artículos), así como una página dedicada a productos favoritos.

Existen muchas funcionalidades que inicialmente había planteado desarrollar, pero que, debido a la proximidad de la fecha de entrega y a la carga de trabajo, no pudieron materializarse. No obstante, como ya he mencionado en el apartado anterior, me siento satisfecho y orgulloso del resultado alcanzado hasta este punto.

## Glosario

### **Abstract**

Resumen breve en inglés que presenta los objetivos, metodología y resultados del proyecto de forma concisa.

### **Backlog del Producto**

Lista priorizada de tareas, funcionalidades o requisitos pendientes por desarrollar en un proyecto Scrum.

### **Base de Datos**

Sistema estructurado para almacenar, gestionar y recuperar datos de forma eficiente.

### **Carrito de Compras**

Funcionalidad de la tienda que permite al usuario seleccionar productos y ver un resumen antes de realizar el pago.

### **CRUD**

Acrónimo de Create, Read, Update y Delete; operaciones básicas de gestión de datos en una base de datos.

**CSS (Cascading Style Sheets)**

Lenguaje utilizado para dar estilo y formato visual a interfaces de usuario web o gráficas.

**Entity-Relationship (E-R)**

Modelo gráfico que representa las entidades de una base de datos y las relaciones entre ellas.

**FXML**

Lenguaje basado en XML utilizado con JavaFX para definir interfaces gráficas de usuario de forma declarativa.

**Hibernate**

Framework de Java para el mapeo objeto-relacional (ORM), que facilita la conexión entre clases Java y tablas de bases de datos.

**HTTP/HTTPS**

Protocolos de comunicación utilizados para enviar y recibir datos en la web. HTTPS incluye cifrado para mayor seguridad.

**IntelliJ IDEA**

Entorno de desarrollo integrado (IDE) especializado en lenguajes como Java, que facilita la escritura y depuración de código.

**Interfaz de Usuario (UI)**

Parte visual de la aplicación que permite la interacción entre el usuario y el sistema.

**JavaFX**

Librería de Java para desarrollar interfaces gráficas de usuario en aplicaciones de escritorio.

**Mantenibilidad**

Facilidad con la que una aplicación puede ser modificada o mejorada en el futuro.

**Metodología SCRUM**

Marco de trabajo ágil utilizado para organizar y gestionar el desarrollo de software de forma iterativa.

**Mockup**

Representación visual estática de una interfaz de usuario, utilizada para mostrar cómo se verá una aplicación antes de implementarla.

**ORM (Object-Relational Mapping)**

Técnica que conecta objetos en código (como clases Java) con tablas en bases de datos relacionales.

**Producto**

Entidad del sistema que representa un artículo disponible para la venta dentro de la tienda.

**Refactorización**

Proceso de reestructuración del código para mejorar su calidad sin modificar su comportamiento externo.

**Requisitos Funcionales (RF)**

Características que describen lo que el sistema debe hacer (como registro de usuarios o gestión del carrito).

**Requisitos No Funcionales (RNF)**

Criterios que definen cómo debe comportarse el sistema (rendimiento, seguridad, usabilidad, etc.).

**Sprint**

Periodo de tiempo determinado durante el cual se desarrollan tareas específicas dentro de la metodología SCRUM.

**Sprint Review / Retrospectiva**

Revisión del trabajo realizado en un sprint y análisis para mejorar el proceso de desarrollo.

**Temu, Shein, Aliexpress**

Grandes plataformas de comercio electrónico globales que sirven como referencia y contraste para el desarrollo de una tienda artesanal local.

**Tienda Online**

Aplicación web o de escritorio diseñada para vender productos a través de Internet.

## Usuario

Persona que interactúa con la aplicación, ya sea como comprador, administrador o creador de contenido.

## Bibliografía

Oracle. (n.d.). *JavaFX overview*. Oracle Help Center.  
<https://docs.oracle.com/javase/8/javase-clienttechnologies.htm>

Hibernate. (n.d.). *Hibernate ORM documentation*.  
<https://hibernate.org/orm/documentation/>

JetBrains. (n.d.). *IntelliJ IDEA: The smartest Java IDE*. <https://www.jetbrains.com/idea/>

MySQL. (n.d.). *MySQL documentation*. Oracle. <https://dev.mysql.com/doc/>

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to Scrum: The rules of the game*. <https://scrumguides.org/>

Villalobos, C. A., & Solís, A. (2019). *Introducción a la ingeniería del software*. Alfaomega.

Pressman, R. S., & Maxim, B. R. (2020). *Ingeniería del software: Un enfoque práctico* (8.ª ed.). McGraw-Hill.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.