

26 DE JUNIO DE 2023



# APLICACIÓN WEB PARA LA GESTIÓN Y VENTA DE PRODUCTOS 3D

ÁREA DE DESARROLLO WEB

ELOHIM CABEZAS SANCHEZ  
GRADO DE INGENIERÍA INFORMÁTICA  
ECABEZASSA@UOC.EDU



Esta obra está sujeta a una licencia de Reconocimiento-  
NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

**FICHA DEL TRABAJO FINAL**

<b>Título del trabajo:</b>	Aplicación web para la gestión y venta de productos 3D
<b>Nombre del autor:</b>	Elohim Cabezas Sánchez
<b>Nombre del consultor:</b>	Vicenç Font Sagrista
<b>Fecha de entrega (mm/aaaa):</b>	06/2023
<b>Área del Trabajo Final:</b>	Desarrollo Web
<b>Titulación:</b>	<i>Ingeniería Informática</i>
<b>Resumen del Trabajo (máximo 250 palabras):</b>	
<p>El propósito de mi proyecto es ofrecer un servicio web para que diferentes empresas y/u organizaciones puedan gestionar sus productos de una manera fácil, cómoda y eficaz. Además, este servicio ofrecerá a dichas entidades, la posibilidad de vender sus productos a los distintos usuarios interesados en estos. Por otro lado, la misma plataforma web brinda la posibilidad de contratar usuarios con perfiles de diseñador con el fin de que estos realicen los distintos diseños que requieren las diferentes organizaciones.</p> <p>Dejando a un lado la gestión de los productos de las distintas empresas/organizaciones, este servicio web también ofrecerá un portal de compras para los usuarios interesados. Un usuario podrá, desde buscar un producto filtrando por distintos criterios (precio, valoraciones, colores, etc.) hasta comprar un producto realizando el pago online. Además, los usuarios podrán dejar valoraciones de los distintos productos y, en caso de duda, contarán con un chat interno para poder ser atendidos por asistentes.</p> <p>Finalmente, el servicio web ofrecerá una plataforma 3D para que, los usuarios con perfiles de diseño puedan realizar de una forma fácil e intuitiva los distintos bocetos solicitados por las entidades.</p>	
<b>Abstract (in English, 250 words or less):</b>	
<p>This project aims to provide a comprehensive web service that enables companies and organizations to manage their products effectively and conveniently. Additionally, the service offers the opportunity for these entities to sell their products to interested users. The web platform also facilitates the hiring of designers who can fulfill various design requirements for different organizations. Apart from product management, the service includes a user-friendly online shopping portal where users can search for products based on various criteria, make online payments, leave product reviews, and access an internal chat for assistance. Furthermore, the platform offers a 3D design platform for designers to create requested sketches easily and</p>	

intuitively. The project offers an integrated approach to product management and sales, catering to the needs of both businesses and interested users. However, several critical aspects should be carefully considered. These include evaluating the technical and economic feasibility of developing such a comprehensive platform, ensuring an intuitive user interface, guaranteeing secure transactions and data privacy, implementing effective selection and evaluation processes for hiring designers, and establishing a unique value proposition to stand out in the competitive market. This ambitious platform aims to address various needs in product management and e-commerce. However, a critical analysis of challenges and considerations is necessary to ensure its viability, usability, and competitiveness in the market.

**Palabras clave (entre 4 y 8):**

Gestión de productos

Comercio electrónico

Servicio web

Venta en línea

Diseño de productos

Experiencia de usuario

Contratación de diseñadores

Competitividad en el mercado

## índice

<b>1. Introducción</b> .....	8
<b>1.1. Contexto</b> .....	8
<b>1.2. Descripción / Definición</b> .....	9
<b>1.3. Objetivos</b> .....	11
<b>1.3.1. Objetivos principales</b> .....	11
<b>1.3.2. Objetivos secundarios</b> .....	12
<b>1.4. Metodología y procedimiento de trabajo</b> .....	12
<b>1.5 Planificación</b> .....	14
<b>1.6 Presupuesto</b> .....	16
<b>2. Fundamentos teóricos</b> .....	16
<b>2.1 Desarrollo web: conceptos básicos</b> .....	16
<b>2.1.1 Cliente y servidor</b> .....	16
<b>2.1.2 Lenguajes de programación web</b> .....	17
<b>2.1.3 Tecnologías de front-end y back-end</b> .....	17
<b>2.1.4 Bases de datos</b> .....	17
<b>2.1.5 Hosting y dominios</b> .....	17
<b>2.2 Lenguajes de programación web</b> .....	17
<b>2.2.1 HTML (<i>HyperText Markup Language</i>)</b> .....	17
<b>2.2.2 CSS (<i>Cascading Style Sheets</i>)</b> .....	18
<b>2.2.3 JavaScript</b> .....	18
<b>2.2.4 PHP</b> .....	18
<b>2.2.5 Python</b> .....	18
<b>2.2.6 Ruby</b> .....	18
<b>2.3 Frameworks y herramientas</b> .....	18
<b>2.3.1 React.js</b> .....	18
<b>2.3.2 Angular</b> .....	18
<b>2.3.3 Django</b> .....	19
<b>2.3.4 Ruby on Rails</b> .....	19
<b>2.3.5 Flask</b> .....	19
<b>2.3.6 Visual Studio Code</b> .....	19
<b>3. Estado del arte</b> .....	19
<b>3.1 Tendencias actuales en desarrollo web</b> .....	19
<b>3.1.1 Desarrollo orientado a dispositivos móviles</b> .....	19
<b>3.1.2 Aplicaciones web progresivas (PWA)</b> .....	19

3.1.3 Single-Page Aplicativos (SPA) .....	20
3.1.4 Progressive Web Apps (PWA).....	20
3.1.5 API RESTful y microservicios .....	20
3.1.6 Inteligencia Artificial y Machine Learning.....	20
3.1.7 Realidad virtual y realidad aumentada.....	20
3.2 Estudio de casos relevantes .....	20
3.2.1 Caso 1 .....	20
3.2.2 Caso 2 .....	21
3.2.3 Caso 3 .....	21
3.2.4 Caso 4 .....	21
4. Análisis y diseño del sistema.....	21
4.1 Requisitos del sistema .....	21
4.1.1. Fichas y modelo de casos de uso y diagrama de actores .....	21
4.1.2. Diagrama de actores.....	35
4.1.3. Diseño relacional de la base de datos.....	35
4.1.4. Diagrama de clases principales .....	36
4.2 Arquitectura del sistema .....	37
4.3 Diseño de la interfaz de usuario.....	38
5. Implementación y desarrollo .....	39
5.1 Elección de tecnologías y herramientas.....	39
5.2 Desarrollo de los módulos principales.....	43
5.3 Pruebas y depuración .....	45
6. Evaluación y resultados .....	47
6.1 Validación del sistema .....	47
6.2 Análisis de resultados obtenidos .....	50
7. Conclusiones .....	51
7.1 Cumplimiento de los objetivos.....	51
7.2 Aportaciones y limitaciones .....	52
7.3 Trabajos futuros .....	54
8. Glosario .....	56
9. Bibliografía.....	63
9.1 Libros.....	63
9.2 Sitos web.....	63
9.3 Artículos consultados .....	64

## Índice de figuras

Ilustración 1. Gráfica crecimiento de la WWW.....	9
Ilustración 2. Plataforma 3D de diseño.....	11
Ilustración 3. Ciclo de vida metodología agile.....	13
Ilustración 4. Interfaz de JetBrains YouTrack.....	14
Ilustración 5. Diagrama de Gantt.....	16
Ilustración 6 Pantalla "Localizador de envío".....	23
Ilustración 7 Pantalla "formulario de contacto".....	24
Ilustración 8 Pantalla "Detalle del producto".....	25
Ilustración 9 Pantalla "Aplicar descuento".....	26
Ilustración 10 Pantalla "Creación/Edición de un producto".....	27
Ilustración 11 Pantalla "Carrito de la compra".....	29
Ilustración 12 Pantalla "Resumen del pedido".....	30
Ilustración 13 Pantalla "Tus productos".....	31
Ilustración 14 Pantalla "Catálogo de productos".....	32
Ilustración 15 Pantalla "Inicio Sesión".....	34
Ilustración 16 Diagrama de actores.....	35
Ilustración 17 Diseño relacional de la base de datos.....	36
Ilustración 18 Diagrama de clases.....	36
Ilustración 19 Diagrama de la estructura hexagonal.....	37

## Índice de tablas

Tabla 1. Planificación por fases.....	15
Tabla 2. Presupuesto del proyecto.....	17
Tabla 3. CU-01 Seguimiento de pedido.....	22
Tabla 4. CU-02 Contactar.....	23
Tabla 5. CU-03 Añadir un producto al carrito.....	24
Tabla 6. CU-04 Aplicar cupón.....	25
Tabla 7. CU-05 Añadir imagen.....	26
Tabla 8. CU-06 Comprar producto.....	28
Tabla 9. CU-07 Gestionar productos.....	30
Tabla 10. CU-08 Filtrar por precio.....	31
Tabla 11. CU-09 Iniciar sesión.....	33



## 1. Introducción

### 1.1. Contexto

*Amazon, Facebook, Wikipedia, Netflix...* únicamente nombrando estas palabras, ya sabemos a qué nos estamos refiriendo. Y es que hoy en día, Internet y las redes sociales, han transformado por completo nuestra forma de trabajar, aprender e interactuar. Estamos hablando de la famosa Era Digital, donde el acceso a la información y las diferentes formas de comunicación que existen, están cada vez más democratizadas. Poder acceder a nuestra red social favorita, consulta el tiempo que hará mañana, ver un capítulo de nuestra serie favorita... es tan fácil como abrir un navegador web e introducir la URL a la cual queremos acceder. Si bien es cierto que existen diferentes formas de acceder a un mismo servicio (por ejemplo, podemos acceder a *Netflix* mediante un navegador web, dispositivo inteligente, TV, etc.) vamos a enfocar este trabajo al acceso de la información mediante páginas web.

Las páginas web es un instrumento que utilizamos todos y cada uno de nosotros prácticamente cada día: instituciones, organizaciones, empresas... con el objetivo de comunicarnos, publicitar nuestro trabajo, expresarnos e incluso, interactuar entre nosotros. Y la verdad sea dicha. Hoy en día podemos crear nuestra página web sin tener ningún conocimiento previo. Basta con hacer unos cuantos clics y un asistente web realizará el trabajo por nosotros.

Podemos ver como la creación y el uso de las páginas web ha ido aumentando exponencialmente a lo largo de los años. Desde el lanzamiento del *World Wide Web* en 1991 y hasta el día de hoy, existen más de 1880 millones de páginas web.

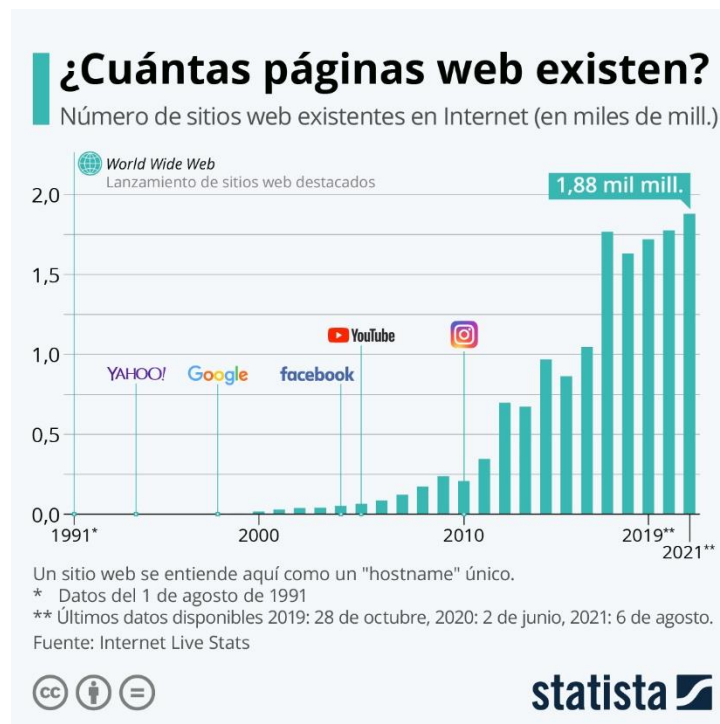


Ilustración 1. Gráfica crecimiento de la WWW

Eso sí, no toda la *World Wide Web* es accesible. La mayor parte, está inactiva. Se estima que aproximadamente menos de 200 millones de páginas web están activas. El resto, por una razón u otra, no se han ido actualizando y finalmente acaban inactivas o desaparecen para siempre.

En resumen, podemos afirmar que estamos viviendo una cuarta revolución industrial protagonizada por la tecnología. Y es que a pesar de que la democratización de Internet sea un concepto relativamente nuevo, en los últimos años ha evolucionado a un ritmo exponencial. Y es que vivimos en la Era Digital, donde obtener una respuesta inmediata a nuestra necesidad, hace que todo lo busquemos en Internet. Esa red, donde tenemos a nuestra disposición millones de páginas web dispuestas a ofrecernos sus servicios para poder satisfacer nuestras necesidades de la mejor forma posible. Así pues, si hay una herramienta que va evolucionando sin cesar y está al alcance de todos esa es, la página web, utilizada a diario por millones de usuarios de todo el mundo.

## 1.2. Descripción / Definición

El propósito de este proyecto es ofrecer un servicio web para que diferentes empresas y/u organizaciones puedan gestionar sus productos de una manera fácil, cómoda y eficaz. Además, este servicio ofrecerá a dichas entidades, la posibilidad de vender sus productos a los distintos usuarios interesados en estos. Por otro lado, la misma plataforma web brinda la posibilidad de contratar usuarios con perfiles de diseñador (previamente registrados en el sistema) con el fin de que estos realicen los distintos diseños que requieren las diferentes entidades.

Dejando a un lado la gestión de los productos de las distintas empresas/organizaciones, este servicio web también ofrecerá un portal de compras para los usuarios interesados. Un usuario podrá, desde buscar un producto filtrando por distintos criterios (precio, valoraciones, colores, etc.) hasta comprar un producto realizando el pago online. Además, los usuarios podrán dejar valoraciones de los distintos productos y, en caso de duda, contarán con un chat interno para poder ser atendidos por asistentes.

Finalmente, el servicio web ofrecerá una plataforma 3D para que, los usuarios con perfiles de diseño puedan realizar de una forma fácil e intuitiva los distintos bocetos solicitados por las entidades.

Así pues, podemos ver claramente diferenciadas 3 partes de las cuales constará este proyecto. Por un lado, tendremos toda la gestión íntegra (CRUD) de las diferentes entidades que existirán en el sistema (usuarios, empresas, organizaciones, productos, roles...). Por otro lado, tendremos la tienda online, donde los usuarios podrán filtrar los productos ofrecidos por las distintas entidades, valorarlos e incluso, comprarlos. Y finalmente contaremos con una plataforma 3D (de uso exclusivo para los perfiles de diseño) para poder trabajar con ella y diseñar lo que se les ha requerido.

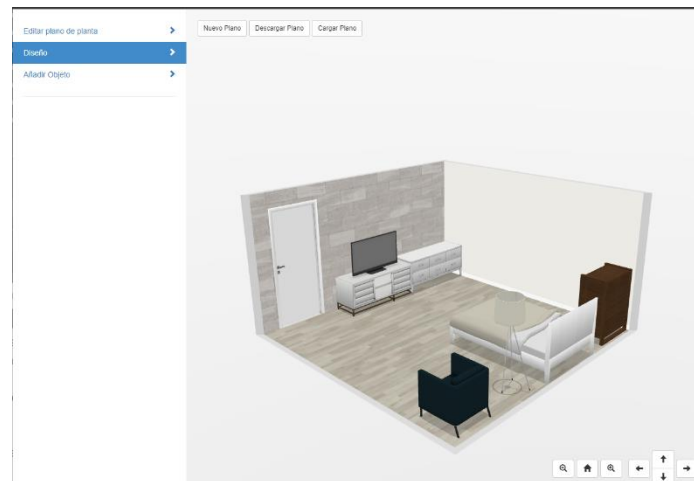


Ilustración 2. Plataforma 3D de diseño

Después de esta breve introducción, queda claro que depende del rol que juguemos en este servicio web, podremos realizar unas acciones en concreto u otras. Lo que si podremos notar es que ambas 3 partes deben estar muy bien relacionadas entre sí, ya que cada una de ellas depende de la otra. Por ejemplo, si una empresa necesita el boceto del diseño de un comedor, ésta deberá previamente contratar a un diseñador para que este pueda realizar el boceto. Una vez entregado dicho boceto, la empresa podrá vender al público este nuevo producto y para ello deberá gestionarlo previamente, dándolo de alta en el sistema, agregando una descripción, imágenes, precio, fecha de inicio de venta, etc. Una vez publicado, un usuario podrá acceder a la tienda online para comprar dicho diseño.

Así pues, este proyecto quiere poner en práctica los distintos conocimientos que se han ido aprendiendo a lo largo de la carrera universitaria, cuidando desde el más mínimo detalles todos los aspectos que se puedan encontrar. Además, con la realización de este proyecto se espera adquirir nuevos conocimientos y experimentar en primera persona los retos que supone gestionar una plataforma web de este calibre. Se espera obtener competencias hasta la fecha no estudiadas que ayuden a crecer tanto personalmente como profesionalmente. El objetivo será obtener un producto muy atractivo visualmente que cumpla con sus expectativas, ofreciendo sus servicios a distintos usuarios con distintos intereses, pero todos ellos utilizando el mismo servicio web.

### 1.3. Objetivos

A continuación, se muestran los objetivos que se proponen lograr a lo largo del recorrido de este proyecto. Para ello, hemos separado los objetos en objetivos principales y objetivos secundarios. Con los objetivos principales se quiere lograr obtener un producto final 100% funcional, con todas las funcionalidades anteriormente comentadas. Los objetivos secundarios, no los debemos entender como objetivos menos importantes, sino que estos nos van a aportar pequeños detalles extra a nuestro producto final que lograrán que este destaque más. Y, sobre todo, los objetivos secundarios servirán para que nuestro crecimiento y aprendizaje personal sea más enriquecedor y de provecho para nuestro futuro.

#### 1.3.1. Objetivos principales

Los objetivos de la aplicación/servicio/web son:

- Ofrecer un sistema robusto de gestión completa (CRUD) de entidades (gestión de usuarios, roles, productos, empresas, perfiles...)
- Ofrecer un sistema de *login* seguro (doble autenticación, mail...)
- Ofrecer un canal de comunicación asíncrono (notificación) y/o síncrono (chat) entre las diferentes entidades: usuarios con administradores, empresas con diseñadores, etc.
- Ofrecer un diseño atractivo y *responsive*.
- Ofrecer un espacio personalizado para cada uno de los distintos roles del sistema.
- Permitir la búsqueda de productos aplicando diferentes filtros.
- Simular el proceso de compra de un producto hasta llegar al pago de este.
- Gestionar el carrito de la compra.
- Ofrecer una plataforma web para poder crear bocetos en 3D y poderlos descargar, cargar, etc.

Los objetivos del usuario/cliente/empresa/organización:

- Poder gestionar sus productos cómodamente (interfaz intuitiva).
- Poder comunicarse con otros usuarios de una manera fluida y rápida.
- Poder buscar productos de interés aplicando criterios de búsqueda, así como poder valorarlos y comprarlos.
- Contar con una herramienta 3D para agilizar el diseño de bocetos.

Los objetivos personales son:

- Ganar experiencia con tecnologías ya conocidas.
- Aprender el uso de nuevas tecnologías.
- Poder introducirse en el mundo de los despliegues automatizados.
- Ganar experiencia al gestionar un proyecto real.

### 1.3.2. Objetivos secundarios

Los objetivos secundarios son:

- Generar documentación útil
- Conseguir un sistema lo más automatizado posible referente a los despliegues.
- Tener la aplicación testeada.
- Tener un sistema de *backup* de datos.
- Personalización de la aplicación al gusto del usuario (poder configurar colores, menús, etc)

### 1.4. Metodología y procedimiento de trabajo

Elegir la metodología de trabajo es una parte muy importante (para no decir la que más) del proyecto que vamos a comenzar. Saber decidir correctamente la metodología y procedimiento de trabajo puede ser el elemento que, o bien nos ayudará a cumplir todos nuestros objetivos previstos o bien fracasaremos en el intento. Para la elección de la metodología de trabajo que vamos a seguir en este proyecto, debemos tener presentes los siguientes aspectos:

- El proyecto va a ser desarrollado por una única persona.
- El desarrollador tiene que ir aprendiendo a medida que avanza el proyecto, hecho que puede provocar que las estimaciones de las entregas no sean con exactitud.
- El proyecto contiene entregas (PECs) con unos plazos ya definidos.
- Imaginaremos que nuestro cliente nos ha pedido que le demos un *feedback* constante. Así pues, durante la realización del proyecto se va a ir dando *feedback* al tutor del TFG (de hecho, esto queda contemplado en el diagrama de *Gantt*).
- Junto al punto anterior imaginaremos que este proyecto ha sido encargado por un cliente.

Por todos estos motivos y teniendo en cuenta que queremos trabajar con una metodología sólida, la opción más viable es utilizar la metodología *agile*.

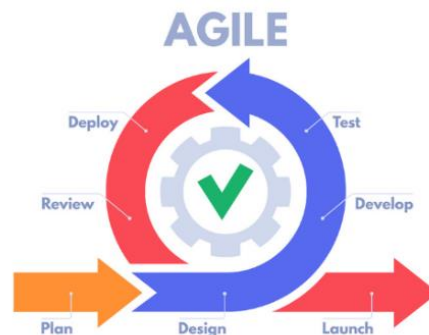


Ilustración 3. Ciclo de vida metodología agile

Si trabajamos con esta metodología, vamos a poder superar cada uno de nuestros ciclos de producción con éxito. Tendremos un entorno de trabajo iterativo, donde podremos separarlo en ciclos que nos van a permitir realizar test de nuestro trabajo, poder desplegarlo y, lo más importante, revisar que el trabajo realizado es correcto.

Para poder llevar a cabo el testeo de nuestra aplicación, vamos a utilizar la metodología conocida como TDD o *Test-Driven-Development*, que se basa principalmente en transformar los requisitos del *software* en casos de prueba para poderlos testear primero.

Para el despliegue automatizado de la aplicación se utilizará la herramienta *Jenkins*. Podremos automatizar nuestros despliegues de forma que cuando realicemos un *push* en nuestro repositorio, Jenkins desplegará la aplicación de forma automática y controlada, de forma que, si hubiera algún tipo de problema, este pararía y no afectaríamos a nuestro entorno productivo.

Así pues, al utilizar estas tecnologías nos ayuda a confirmar que la metodología *agile* es la más adecuada para este proyecto, obteniendo en cada uno de nuestros ciclos de trabajo, una aplicación testeada y desplegada funcionando correctamente donde nuestro cliente puede ir observando la evolución de esta y analizar los resultados esperados.

Para organizar el flujo de trabajo, utilizaremos *JetBrains YouTrack*, que nos ofrecerá (dentro de la metodología *agile*) su variante *Kanban*. Aquí ya podríamos crear nuestras primeras tareas como *Walking Skeleton*, Configurar el servidor...

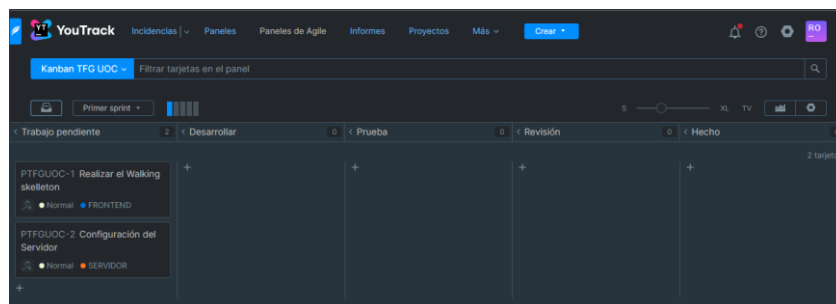


Ilustración 4. Interfaz de JetBrains YouTrack

Por último, las tecnologías con las que vamos a trabajar van a ser las siguientes:

- Cliente
  - Bootstrap
  - Javascript, JQuery, Ajax
  - ThreeJS
- Servidor
  - PHP
  - Framework Slim (PHP)
  - Python
- Base de datos
  - Mysql

## 1.5 Planificación

El proyecto se ha dividido en 5 fases, cada una de ellas corresponde a las 4 PECs + Defensa del proyecto. Así pues, estas fases son:

- **Fase 1.** Plan de trabajo → Se define el problema que pretende resolver el proyecto, el trabajo concreto que se llevará a cabo y su descomposición en tareas e hitos temporales.
- **Fase 2.** Análisis y Diseño → Se definen los requerimientos, el análisis y los diseños. Por otro lado, también se estudia la tecnología escogida, así como el *software* a utilizar y se realiza su instalación y pruebas de funcionamiento.
- **Fase 3.** Implementación → Se ha de desarrollar la aplicación y desplegarla.
- **Fase 4.** Memoria y presentación → Se redacta la memoria final y se realiza una presentación virtual del trabajo.
- **Fase 5.** Defensa → El tribunal evaluará cada trabajo (i.e. presentación, memoria y el producto) y a continuación hará las preguntas que considere oportunas.

En la siguiente tabla se muestra la planificación de distintas fases, mostrando la fecha de inicio y fecha fin de los distintos ítems:

Fase	Tarea	Inicio	Fin	Días
1	Selección del tema	01/03/23	02/03/23	2
1	Contexto y Descripción de objetivos	03/03/23	05/03/23	3
1	Metodología y procedimiento de trabajo	06/03/23	09/03/23	4
1	Planificación y presupuesto	10/03/23	13/03/23	4
1	Entrega y finalización PEC 1	14/03/23	14/03/23	1
2	Estudio de los requerimientos	15/03/23	21/03/23	7
2	Estudio de Análisis y Diseño	22/03/23	29/03/23	8
2	Estudio de las tecnologías escogidas	30/03/23	04/04/23	6
2	Configuración del servidor y pruebas del entorno	05/04/23	11/04/23	7
2	Entrega y finalización PEC 2	12/04/23	14/04/23	3
3	Creación de la Base de Datos	15/04/23	16/04/23	2
3	Frontend tienda virtual	17/04/23	21/04/23	5
3	Frontend zona privada de organizaciones	22/04/23	26/04/23	5
3	Frontend zona privada de administradores	23/04/23	01/05/23	9
3	Backend zona privada de administradores	02/05/23	10/05/23	9
3	Backend zona privada de organizaciones	11/05/23	20/05/23	10
3	Backend tienda virtual	21/05/23	29/05/23	9
3	Configuración plataforma 3D	30/05/23	31/05/23	2
3	Realización pruebas de integración y unitarias	01/06/23	03/06/23	3
3	Despliegue de la aplicación	04/06/23	05/06/23	2
3	Entrega y finalización PEC 3	05/06/23	05/06/23	1
4	Realización de Memoria y entrega	06/06/23	16/06/23	11
4	Realización de la presentación virtual y entrega	17/06/23	23/06/23	7
4	Realización del informe de autoevaluación y entrega	24/06/23	26/06/23	3
5	Defensa del proyecto	03/07/23	07/07/23	5

Tabla 1. Planificación por fases

A continuación, se muestra el diagrama de Gantt. En este se podrá observar que la fase 3 contiene *feedbacks* al usuario (revisión con tutor) cada 15 días para poder ir revisando el trabajo realizado. El siguiente diagrama complementa nuestro tablero Kaban que hemos creado, ayudándonos así a contextualizar la consecución de las diferentes entregas.

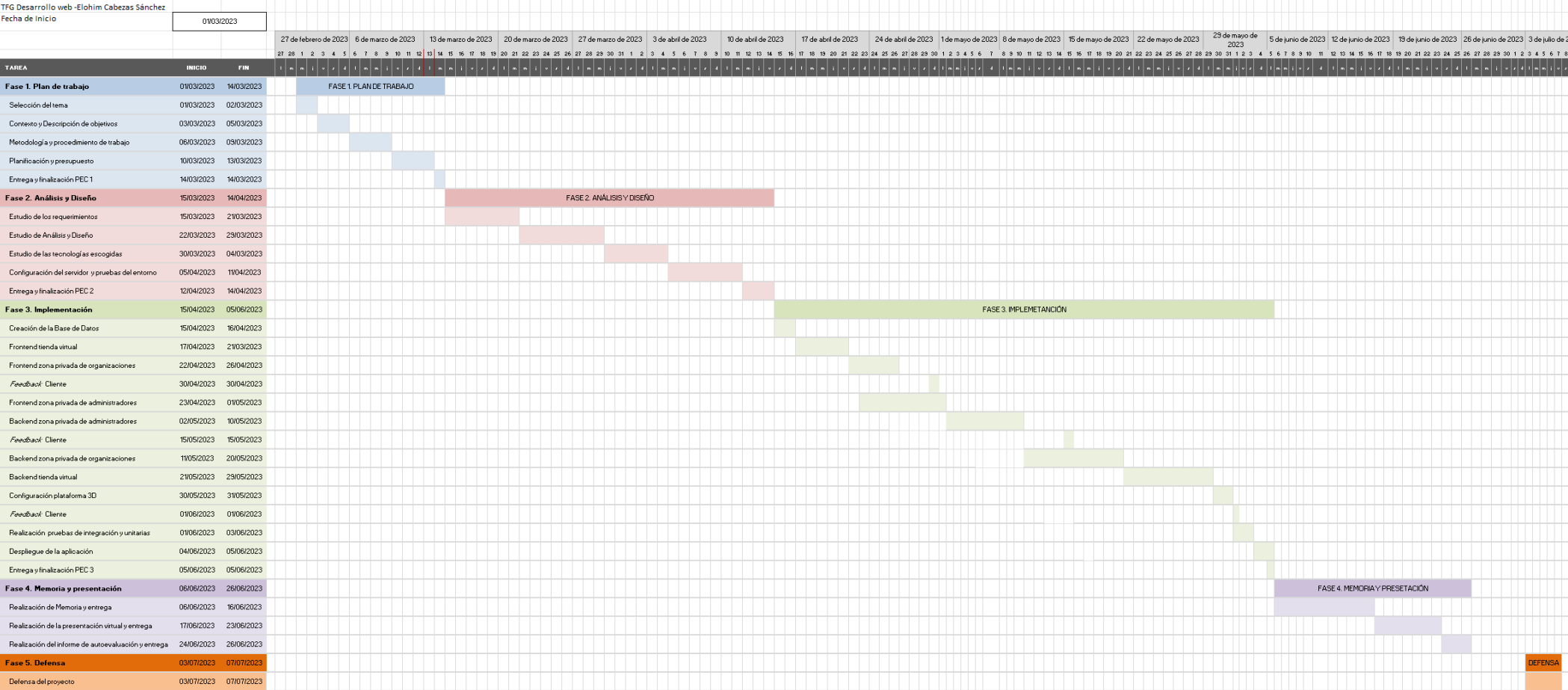


Ilustración 5. Diagrama de Gantt



## 1.6 Presupuesto

El proyecto que nos ocupa está desarrollado por una única persona, con lo cual todos los distintos equipos que podrías encontrar en un entorno real (equipo de *testing*, equipo de desarrollo, equipo funcional, etc.) recaen sobre la misma persona.

A continuación, se muestra el plan económico del proyecto, teniendo en cuenta los diferentes elementos que intervienen en este.

Nota: Para los elementos que requieran pago mensual, el precio total será multiplicado por 6 meses (semestre completo).

<b>Fecha del presupuesto</b>	09/03/2023
------------------------------	------------

DESCRIPCIÓN	UNIDADES	PRECIO	TOTAL
<i>Dominio elohimcabezas.com</i>	1	12,00 €	12,00 €
<i>Cuota mensual servidor OVH Cloud</i>	6	3,50 €	42,00 €
<i>Ordenador personal para entorno de trabajo</i>	1	1050,00 €	1050,00 €
<i>Desarrollador del proyecto<sup>1</sup></i>	153	25,00/hora €	3825,00 €
<i>Visual Studio Code</i>	1	0 €	0 €
Jenkins	1	0 €	0 €
JetBrains YouTrack	1	0 €	0 €
Repositorio GitHub	1	0 €	0 €

	<b>SUB-TOTAL</b>	4929,00 €
	<b>IVA %</b>	21,00%
	<b>TOTAL PRESUPUESTADO</b>	<b>5964,09 €</b>

Tabla 2. Presupuesto del proyecto

## 2. Fundamentos teóricos

### 2.1 Desarrollo web: conceptos básicos

El desarrollo web es el proceso de crear aplicaciones, sitios web y otros recursos en la ya conocida *World Wide Web*. Implica una combinación de diseño visual, programación y gestión de bases de datos para construir y mantener soluciones web funcionales. A continuación, vamos a presentar algunos conceptos básicos en el desarrollo web:

**2.1.1 Cliente y servidor:** En el desarrollo web, se utiliza la arquitectura cliente-servidor. El cliente se refiere al software o dispositivo que solicita y consume la información, como un navegador web en una computadora (sobremesa o portátil) o dispositivo móvil. El servidor, por otro lado, es el software o el hardware que proporciona los recursos solicitados por el cliente, como archivos HTML, imágenes, datos de la base de datos, etc.

2.1.2 **Lenguajes de programación web:** Hay varios lenguajes de programación utilizados en el desarrollo web. Algunos de los más comunes son:

- **HTML** (*HyperText Markup Language*): este lenguaje de marcas se utiliza para estructurar el contenido y la información en una página web.
- **CSS** (*Cascading Style Sheets*): se utiliza para definir la presentación y el diseño visual de una página web.
- **JavaScript**: es un lenguaje de programación del lado del cliente que permite agregar interactividad y dinamismo a una página web.
- **PHP, Python, Ruby, entre otros**: son lenguajes de programación del lado del servidor utilizados para generar contenido dinámico y trabajar con bases de datos.

2.1.3 **Tecnologías de front-end y back-end:** En el desarrollo web, se distingue entre *front-end* y *back-end*. El *front-end* se refiere a todo lo relacionado con la interfaz y la interacción del usuario. Se utilizan tecnologías como HTML, CSS y JavaScript para crear una experiencia visual y funcional para el usuario. Por otro lado, el *back-end* se ocupa de la lógica y la gestión de datos en el servidor. Se utilizan lenguajes de programación del lado del servidor, como PHP o Python, junto con bases de datos, para manejar las solicitudes del cliente y proporcionar la funcionalidad necesaria.

2.1.4 **Bases de datos:** Las bases de datos son componentes esenciales en muchas aplicaciones web. Se utilizan para almacenar, gestionar y acceder a grandes volúmenes de información. Algunas bases de datos comunes son MySQL, PostgreSQL y MongoDB. Los lenguajes de consulta como SQL (*Structured Query Language*) se utilizan para interactuar con las bases de datos y realizar operaciones como la inserción, consulta, actualización y eliminación de datos.

2.1.5 **Hosting y dominios:** Para que un sitio web esté accesible en Internet, se necesita un servicio de alojamiento web (*hosting*) donde se almacenen los archivos y recursos del sitio. Además, se requiere un nombre de dominio único (por ejemplo, **elohimcabezas.com**) para identificar y acceder al sitio web.

## 2.2 Lenguajes de programación web

En el desarrollo web, existen varios lenguajes de programación que se utilizan para crear aplicaciones y sitios web interactivos. A continuación, presentaremos algunos de los lenguajes de programación web más populares:

2.2.1 **HTML** (*HyperText Markup Language*): HTML es el lenguaje de marcado estándar utilizado para estructurar y presentar contenido en la web. Con HTML, se definen los elementos y las etiquetas que representan los diferentes componentes de una página web, como encabezados, párrafos, imágenes, enlaces, formularios, entre otros.

**2.2.2 CSS (*Cascading Style Sheets*):** CSS es un lenguaje utilizado para definir la apariencia y el diseño visual de una página web. Con CSS, se pueden aplicar estilos, como colores, fuentes, márgenes, tamaños y efectos, a los distintos elementos que hemos definido con HTML. Esto permite personalizar la presentación y la estética de una página web.

**2.2.3 JavaScript:** JavaScript es un lenguaje de programación del lado del cliente que permite agregar interactividad y funcionalidad dinámica a una página web. Con JavaScript, se pueden realizar acciones como validación de formularios, manipulación del DOM (*Document Object Model*), animaciones, llamadas a servicios web, entre otros. Es uno de los lenguajes más utilizados en el desarrollo web.

**2.2.4 PHP:** PHP es un lenguaje de programación del lado del servidor que se utiliza para crear aplicaciones web dinámicas. Con PHP, se pueden generar páginas HTML dinámicamente, interactuar con bases de datos y realizar diversas operaciones en el servidor. PHP es especialmente popular para la creación de aplicaciones basadas en el popular CMS (*Content Management System*) como WordPress.

**2.2.5 Python:** Python es un lenguaje de programación versátil que se utiliza en una amplia gama de aplicaciones, incluido el desarrollo web. Con Python, se pueden construir sitios web y aplicaciones web utilizando *frameworks* populares como Django y Flask. Python se destaca por su sintaxis clara y legible, lo que facilita el desarrollo rápido y la mantenibilidad del código.

**2.2.6 Ruby:** Ruby es otro lenguaje de programación utilizado en el desarrollo web, especialmente con el *framework* **Ruby on Rails**. Ruby se enfoca en la simplicidad y la productividad del programador, lo que lo convierte en una elección popular para el desarrollo web ágil y eficiente.

## 2.3 Frameworks y herramientas

En el desarrollo web, los *frameworks* y herramientas desempeñan un papel crucial al proporcionar una estructura, funcionalidades predefinidas y herramientas que agilizan el proceso de desarrollo. Estas herramientas permiten a los desarrolladores enfocarse en la lógica de negocio y acelerar la creación de aplicaciones web. Vamos a enumerar algunos *frameworks* y herramientas populares en el desarrollo web:

**2.3.1 React.js:** React.js es una biblioteca JavaScript de código abierto desarrollada por Facebook. Se utiliza para construir interfaces de usuario interactivas y reactivas. React.js permite la creación de componentes reutilizables, facilita la manipulación del DOM virtual y ofrece un rendimiento eficiente al actualizar solo los componentes necesarios. Es ampliamente utilizado para construir aplicaciones de una sola página (SPA) y aplicaciones web dinámicas.

**2.3.2 Angular:** Angular es un *framework* de desarrollo web de código abierto desarrollado por Google. Utiliza TypeScript, un lenguaje de programación tipado y basado en JavaScript. Angular facilita la creación de aplicaciones web complejas y de gran escala al proporcionar una arquitectura robusta, un enlace de datos bidireccional y una amplia gama de características y herramientas.

**2.3.3 Django:** Django es un *framework* de desarrollo web de alto nivel y de código abierto, basado en Python. Django se centra en la simplicidad, la eficiencia y la reutilización del código. Proporciona un conjunto completo de herramientas y características para desarrollar aplicaciones web seguras y escalables, incluyendo un ORM (*Object-Relational Mapping*) para trabajar con bases de datos, autenticación de usuarios, administración del sistema y enrutamiento.

**2.3.4 Ruby on Rails:** Ruby on Rails, también conocido como Rails, es un *framework* de desarrollo web escrito en el lenguaje de programación Ruby. Rails sigue el principio de convención sobre configuración, lo que significa que proporciona una estructura y una configuración predeterminadas, lo que facilita el desarrollo rápido y la codificación eficiente. Rails ofrece una arquitectura MVC (*Modelo-Vista-Controlador*), un ORM llamado *ActiveRecord* y una amplia gama de gemas (paquetes) para agregar funcionalidades adicionales.

**2.3.5 Flask:** Flask es un *microframework* de Python para el desarrollo web. Aunque es minimalista en comparación con otros *frameworks*, Flask es extremadamente flexible y fácil de aprender. Proporciona una base sólida para construir aplicaciones web simples y rápidas, y permite la extensión con otras bibliotecas y herramientas según sea necesario.

**2.3.6 Visual Studio Code:** Visual Studio Code (VS Code) es un editor de código fuente desarrollado por Microsoft. Es ampliamente utilizado en el desarrollo web debido a su interfaz intuitiva, su amplia compatibilidad con lenguajes de programación y su capacidad para integrarse con herramientas y extensiones populares. VS Code ofrece funciones avanzadas como resaltado de sintaxis, depuración integrada, control de versiones y administración de paquetes.

### 3. Estado del arte

#### 3.1 Tendencias actuales en desarrollo web

El campo del desarrollo web está en constante evolución, impulsado por avances tecnológicos y demandas cambiantes de los usuarios. A continuación, vamos a presentar algunas de las tendencias más destacadas en el desarrollo web en la actualidad:

##### 3.1.1 Desarrollo orientado a dispositivos móviles

Con el aumento del uso de dispositivos móviles, el desarrollo web ha adoptado un enfoque centrado en la experiencia del usuario móvil. Las técnicas de diseño y desarrollo responsivo permiten crear sitios web que se adaptan y funcionan de manera óptima en diferentes dispositivos y tamaños de pantalla.

##### 3.1.2 Aplicaciones web progresivas (PWA)

Las Aplicaciones Web Progresivas (PWA) son sitios web que ofrecen funcionalidades similares a las aplicaciones nativas, como acceso *offline*, notificaciones *push* y acceso al *hardware* del dispositivo. Las PWA brindan una experiencia de usuario más fluida y permiten a los desarrolladores llegar a una audiencia más amplia sin tener que crear aplicaciones nativas para diferentes plataformas.

### 3.1.3 Single-Page Aplicativos (SPA)

Las *Single-Page Applications* (SPA) son aplicaciones web que cargan una única página HTML y, a medida que los usuarios interactúan con la aplicación, se actualizan dinámicamente sin tener que cargar páginas adicionales. Esto mejora la velocidad y la experiencia del usuario al eliminar los tiempos de espera entre las transiciones de página.

### 3.1.4 Progressive Web Apps (PWA)

Las *Progressive Web Apps* (PWA) son aplicaciones web que utilizan tecnologías web modernas para brindar una experiencia de usuario similar a la de una aplicación nativa. Las PWA se pueden instalar en el dispositivo del usuario, funcionan *offline* y pueden enviar notificaciones *push*, lo que las convierte en una opción atractiva para desarrolladores y usuarios.

### 3.1.5 API RESTful y microservicios

Las API RESTful se han vuelto fundamentales para el desarrollo web, permitiendo la comunicación entre diferentes sistemas y aplicaciones. Los microservicios, que son pequeñas unidades de funcionalidad independientes y escalables, se han convertido en una arquitectura popular para el desarrollo de aplicaciones web, permitiendo la construcción de sistemas modulares y flexibles.

### 3.1.6 Inteligencia Artificial y Machine Learning

La integración de la inteligencia artificial (IA) y el *machine learning* en el desarrollo web está en constante crecimiento. Estas tecnologías se utilizan para mejorar la personalización, la recomendación de contenidos y la automatización de tareas. Los *chatbots* y los asistentes virtuales basados en IA también se están volviendo comunes en los sitios web.

### 3.1.7 Realidad virtual y realidad aumentada

El desarrollo web también ha incorporado la realidad virtual (VR) y la realidad aumentada (AR). Las aplicaciones web basadas en VR y AR permiten experiencias inmersivas y enriquecidas, desde la visualización de productos en 3D hasta la integración de información virtual en el entorno real.

## 3.2 Estudio de casos relevantes

En esta sección, veremos algunos casos relevantes que demuestran la aplicación de las tendencias actuales en el desarrollo web. Estos casos destacados muestran cómo diversas empresas o proyectos han utilizado tecnologías y enfoques innovadores para mejorar la experiencia del usuario y alcanzar sus objetivos comerciales. A continuación, se mencionan cuatro casos de ejemplo:

**3.2.1 Caso 1: Airbnb** Airbnb es una plataforma líder en el sector de la industria hotelera que ha adoptado una arquitectura basada en microservicios y una estrategia de desarrollo orientada a dispositivos móviles. Su enfoque centrado en el usuario móvil, combinado con una interfaz

intuitiva y funcionalidades de geolocalización, ha permitido a los usuarios encontrar alojamientos de manera sencilla y rápida, brindando una experiencia fluida y adaptada a dispositivos móviles.

**3.2.2 Caso 2: Spotify** Spotify, uno de los principales servicios de transmisión de música, ha implementado una aplicación web progresiva (PWA) que permite a los usuarios acceder a su música favorita incluso sin conexión a internet. La PWA de Spotify ofrece una experiencia de usuario similar a la de una aplicación nativa, brindando un rendimiento rápido y notificaciones *push* para mantener a los usuarios comprometidos y actualizados con sus artistas favoritos.

**3.2.3 Caso 3: Slack** Slack es una plataforma de comunicación empresarial que ha apostado por el desarrollo de una *Single-Page Application* (SPA). Mediante el uso de tecnologías como React.js y WebSocket, Slack proporciona una interfaz de usuario dinámica y fluida, permitiendo a los usuarios comunicarse en tiempo real y colaborar de manera eficiente, sin la necesidad de recargar la página constantemente.

**3.2.4 Caso 4: Google Maps** Google Maps ha integrado la realidad aumentada (AR) en su aplicación web móvil. Los usuarios pueden ver superpuestas las indicaciones de navegación en tiempo real en el entorno físico a través de la cámara de su dispositivo. Esta aplicación de AR proporciona una experiencia inmersiva y enriquecida, facilitando la navegación y la orientación en el mundo real.

Como acabamos de ver, cada caso destaca el enfoque utilizado y las tecnologías empleadas, mostrando el impacto positivo que pueden tener estas tendencias en el ámbito del desarrollo web. Así pues, podemos decir que estos casos demuestran cómo diversas empresas han implementado tendencias actuales en el desarrollo web para mejorar la experiencia del usuario y lograr sus objetivos comerciales.

## 4. Análisis y diseño del sistema


### 4.1 Requisitos del sistema

#### 4.1.1. Fichas y modelo de casos de uso y diagrama de actores

A continuación, se muestran los distintos casos de uso con su ficha técnica y prototipo de pantalla propuesto en cada caso. La imagen del prototipo se encuentra debajo de cada tabla.

CU-01	Seguimiento de pedido
Versión	1.0 (20/04/23)
Dependencias	Ninguna
Precondición	Ninguna
Descripción	Un usuario, identificado o no, deberá poder revisar el estado de un pedido a partir de un identificador y correo electrónico.

	Paso	Acción
Secuencia normal	1	El usuario pulsará al enlace "Pedidos" de la barra de navegación.
	2	El sistema mostrará la vista "Seguimiento pedido".
	3	El usuario introducirá los datos que requiere la vista y pulsará al botón "Localizar".
	4	El sistema mostrará una vista con la información y el estado del pedido introducido.
Postcondición	Se muestra por pantalla la vista con la información y el estado del pedido.	
Excepciones		E1 No existe ningún pedido con ese identificador y correo electrónico.
	4	E1.1 El sistema muestra un mensaje diciendo que no existe ningún pedido vinculado a ese localizar y correo electrónico.
		E1.2 Se cancela el caso de uso.
Comentarios	Ninguno	



TELF: +34 973451332 | EMAIL: ecabezas3@hotmail.com

Seguimiento pedido | Contacta con nosotros

LOGOTIPO INICIO TIENDA PEDIDOS CONTACTO

**Localizador de envío**  
Localice su pedido de forma rápida

Para rastrear su pedido, ingrese su número de (ID de pedido) en el cuadro a continuación, juntamente con su correo electrónico y presione el botón "Rastrear".  
Podrá encontrar el ID del pedido en el correo electrónico de confirmación que debería haber recibido en la compra del producto.

Número del pedido

correo electrónico

Localizar

Ilustración 6 Pantalla "Localizador de envío"

CU-02	Contactar
Versión	1.0 (20/04/23)
Dependencias	Ninguna
Precondición	Ninguna

Descripción	Un usuario, identificado o no, deberá poder contactar con la administración.		
Secuencia normal	Paso	Acción	
	1	El usuario pulsará al enlace "Contacto" de la barra de navegación.	
	2	El sistema mostrará la vista "Contacto".	
	3	El usuario introducirá los datos que requiere la vista y pulsará al botón "Enviar mensaje".	
4	El sistema indica que se ha enviado el formulario correctamente.		
Postcondición	Se ha enviado un correo con los datos introducidos en el formulario.		
Excepciones	Paso	Acción	
	4	E1	Alguno de los campos no tiene el formato indicado.
		E1.1	El sistema muestra un mensaje diciendo que alguno de los campos no sigue el formato correcto.
		E1.2	Se cancela el caso de uso.
Comentarios	El mensaje se envía como un correo a la administración con los datos introducidos.		

Ilustración 7 Pantalla "formulario de contacto"

CU-03	Añadir un producto al carrito
Versión	1.0 (20/04/23)
Dependencias	Ninguna



Precondición	Existe como mínimo un producto dentro del sistema.	
Descripción	Un usuario, identificado o no, deberá poder añadir un producto al carrito.	
Secuencia normal	Paso	Acción
	1	El usuario pulsará al enlace “Tienda” de la barra de navegación.
	2	El sistema mostrará la vista “Catálogo de productos”.
	3	El usuario pulsa en uno de los productos de la lista.
	4	El sistema mostrará la vista “Detalle de un producto” con la información del producto seleccionado.
	5	El usuario introducirá la cantidad de unidades que quiere comprar y pulsará al botón “Añadir al carrito”.
6	El sistema indicará que el producto ha sido añadido al carrito.	
Postcondición	El número de productos en el carrito del usuario será igual que antes más uno con los datos del producto seleccionado junto a su cantidad.	
Excepciones	Ninguna	
Comentarios	Ninguno	

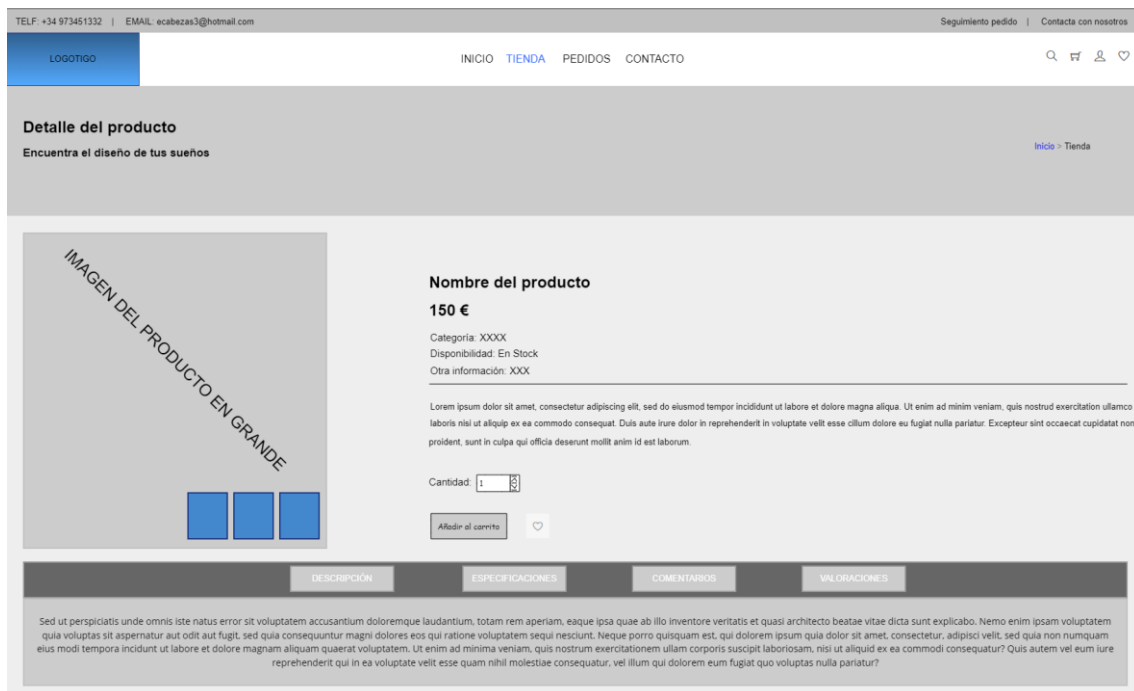


Ilustración 8 Pantalla "Detalle del producto"

<b>CU-04</b>	<b>Aplicar cupón</b>						
Versión	1.0 (20/04/23)						
Dependencias	Ninguna						
Precondición	El usuario está identificado con su usuario y tiene productos en el carrito.						
Descripción	Un usuario deberá poder aplicar un código de descuento a una compra.						
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario introduce el código de cupón y pulsa en el botón "Aplicar".</td> </tr> <tr> <td>2</td> <td>El sistema indica que el cupón se ha aplicado correctamente.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario introduce el código de cupón y pulsa en el botón "Aplicar".	2	El sistema indica que el cupón se ha aplicado correctamente.
	Paso	Acción					
	1	El usuario introduce el código de cupón y pulsa en el botón "Aplicar".					
2	El sistema indica que el cupón se ha aplicado correctamente.						
Postcondición	El usuario habrá aplicado un cupón sobre una compra.						
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>E1 Si el código de cupón no existe o ya ha sido usado.</td> </tr> </tbody> </table>	Paso	Acción	2	E1 Si el código de cupón no existe o ya ha sido usado.		
	Paso	Acción					
2	E1 Si el código de cupón no existe o ya ha sido usado.						

	E1.1	El sistema indica que ese código de cupón no es válido.
	E1.2	Se cancela el caso de uso.
	E2	Si la compra ya tiene un cupón aplicado.
2	E2.1	El sistema indica que esta compra ya tiene un cupón aplicado.
	E2.2	Se cancela el caso de uso.
Comentarios Ninguno.		

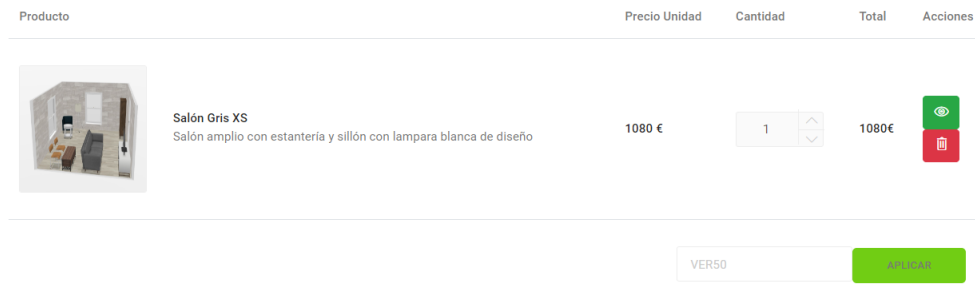


Ilustración 9 Pantalla "Aplicar descuento"

CU-05	Añadir imagen										
Versión	1.0 (20/04/23)										
Dependencias	Ninguna										
Precondición	El usuario está identificado con un usuario de tipo organización y está en la vista de "Edición/Creación producto".										
Descripción	Un usuario organización deberá poder añadir imágenes vinculadas a un producto.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario pulsa al enlace "Imágenes".</td> </tr> <tr> <td>2</td> <td>El sistema muestra una vista para poder añadir imágenes.</td> </tr> <tr> <td>3</td> <td>El usuario añade una o varias imágenes y pulsa al botón "Añadir".</td> </tr> <tr> <td>4</td> <td>El sistema indica que se han añadido las imágenes correctamente y vuelve a la vista "Edición/Creación producto".</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario pulsa al enlace "Imágenes".	2	El sistema muestra una vista para poder añadir imágenes.	3	El usuario añade una o varias imágenes y pulsa al botón "Añadir".	4	El sistema indica que se han añadido las imágenes correctamente y vuelve a la vista "Edición/Creación producto".
	Paso	Acción									
	1	El usuario pulsa al enlace "Imágenes".									
	2	El sistema muestra una vista para poder añadir imágenes.									
3	El usuario añade una o varias imágenes y pulsa al botón "Añadir".										
4	El sistema indica que se han añadido las imágenes correctamente y vuelve a la vista "Edición/Creación producto".										
Postcondición	El usuario habrá añadido imágenes vinculadas a un producto.										

	Paso	Acción
Excepciones		E1 Si alguna imagen no tiene el formato correcto.
	4	E1.1 El sistema indica que una de las imágenes no tiene el formato correcto.
		E1.2 Se cancela el caso de uso.
		E2 Si alguna imagen pesa demasiado.
	4	E2.1 El sistema indica que una de las imágenes pesa demasiado.
		E2.2 Se cancela el caso de uso.
Comentarios	Ninguno.	

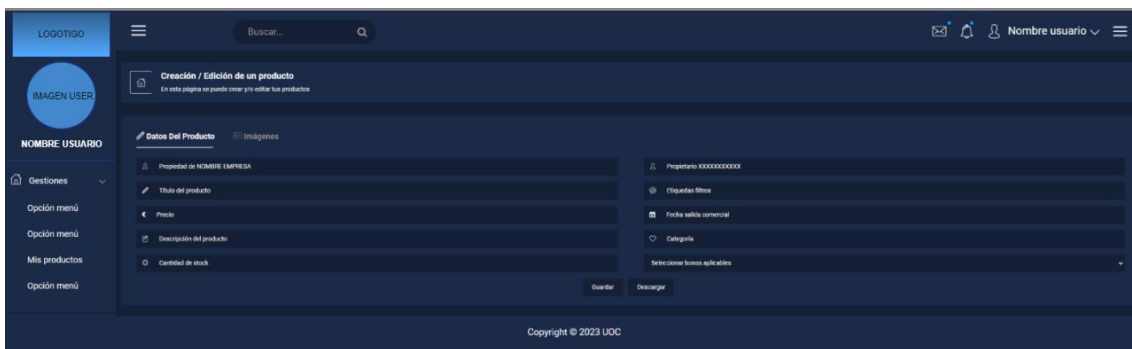


Ilustración 10 Pantalla "Creación/Edición de un producto"

CU-06 Comprar producto																																			
Versión	1.0 (20/04/23)																																		
Dependencias	<ul style="list-style-type: none"> <li>• Iniciar sesión</li> <li>• Aplicar cupón</li> <li>• Añadir un producto al carrito</li> </ul>																																		
Precondición	Ninguna																																		
Descripción	Un usuario, identificado o no, deberá poder comprar productos que se venden en la plataforma.																																		
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Se hace el caso de uso "Añadir un producto al carrito".</td> </tr> <tr> <td>2</td> <td>Si el usuario quiere seguir comprando</td> </tr> <tr> <td>2.1</td> <td>Se vuelve al paso 1.</td> </tr> <tr> <td>3</td> <td>El usuario pulsa en el botón con icono de carrito.</td> </tr> <tr> <td>4</td> <td>El sistema muestra la vista "Carrito de la compra".</td> </tr> <tr> <td>5</td> <td>Si el usuario quiere introducir un código cupón.</td> </tr> <tr> <td>5.1</td> <td>Se hace el caso de uso "Aplicar cupón".</td> </tr> <tr> <td>6</td> <td>El usuario selecciona el tipo de envío y pulsa en el botón "Continuar con el pago".</td> </tr> <tr> <td>7</td> <td>El sistema muestra la vista "Facturación".</td> </tr> <tr> <td>8</td> <td>Si el usuario quiere iniciar sesión.</td> </tr> <tr> <td>8.1</td> <td>Se hace el caso de uso "Iniciar sesión".</td> </tr> <tr> <td>9</td> <td>Si el usuario quiere introducir un código cupón.</td> </tr> <tr> <td>9.1</td> <td>Se hace el caso de uso "Aplicar cupón".</td> </tr> <tr> <td>10</td> <td>El usuario introduce los datos que solicita la vista y pulsa al botón "Proceder con el pago".</td> </tr> <tr> <td>11</td> <td>Si los datos introducidos no son válidos.</td> </tr> <tr> <td>11.1</td> <td>El sistema informa que uno de los campos no es válido.</td> </tr> </tbody> </table>	Paso	Acción	1	Se hace el caso de uso "Añadir un producto al carrito".	2	Si el usuario quiere seguir comprando	2.1	Se vuelve al paso 1.	3	El usuario pulsa en el botón con icono de carrito.	4	El sistema muestra la vista "Carrito de la compra".	5	Si el usuario quiere introducir un código cupón.	5.1	Se hace el caso de uso "Aplicar cupón".	6	El usuario selecciona el tipo de envío y pulsa en el botón "Continuar con el pago".	7	El sistema muestra la vista "Facturación".	8	Si el usuario quiere iniciar sesión.	8.1	Se hace el caso de uso "Iniciar sesión".	9	Si el usuario quiere introducir un código cupón.	9.1	Se hace el caso de uso "Aplicar cupón".	10	El usuario introduce los datos que solicita la vista y pulsa al botón "Proceder con el pago".	11	Si los datos introducidos no son válidos.	11.1	El sistema informa que uno de los campos no es válido.
	Paso	Acción																																	
	1	Se hace el caso de uso "Añadir un producto al carrito".																																	
	2	Si el usuario quiere seguir comprando																																	
	2.1	Se vuelve al paso 1.																																	
	3	El usuario pulsa en el botón con icono de carrito.																																	
	4	El sistema muestra la vista "Carrito de la compra".																																	
	5	Si el usuario quiere introducir un código cupón.																																	
	5.1	Se hace el caso de uso "Aplicar cupón".																																	
	6	El usuario selecciona el tipo de envío y pulsa en el botón "Continuar con el pago".																																	
	7	El sistema muestra la vista "Facturación".																																	
8	Si el usuario quiere iniciar sesión.																																		
8.1	Se hace el caso de uso "Iniciar sesión".																																		
9	Si el usuario quiere introducir un código cupón.																																		
9.1	Se hace el caso de uso "Aplicar cupón".																																		
10	El usuario introduce los datos que solicita la vista y pulsa al botón "Proceder con el pago".																																		
11	Si los datos introducidos no son válidos.																																		
11.1	El sistema informa que uno de los campos no es válido.																																		




	11.2	Se vuelve al paso 10.	
	12	El sistema redirecciona al usuario a la pasarela de pagos externa.	
	13	El usuario introduce los datos del método de pago elegido y finaliza el pago.	
	14	El sistema informa que el pago se ha finalizado correctamente.	
Postcondición	El número de pedidos del usuario es el mismo de antes más uno con los datos del pedido que se acaba de realizar.		
Excepciones	Paso	Acción	
	12	E1	Si la pasarela de pago no está disponible.
		E1.1	El sistema informa que el sistema de pago no está disponible en ese momento.
		E1.2	Se cancela el caso de uso.
		E2	Si durante el proceso de pago el producto ha quedado agotado o no tiene existencias suficientes para el pedido.
		E2.1	El sistema informa que no hay existencias suficientes para uno de los productos.
		E2.2	Se cancela el caso de uso.
Comentarios	La pasarela de pagos simulará ser externa administrada por un tercero.		

TELF. +34 973451332 | EMAIL: ecabezas3@hotmail.com

Seguimiento pedido | Contacta con nosotros

LOGOTIPO INICIO TIENDA PEDIDOS CONTACTO

**Carrito de la compra**  
Detalle de los productos de tu carrito [Inicio - Carrito](#)

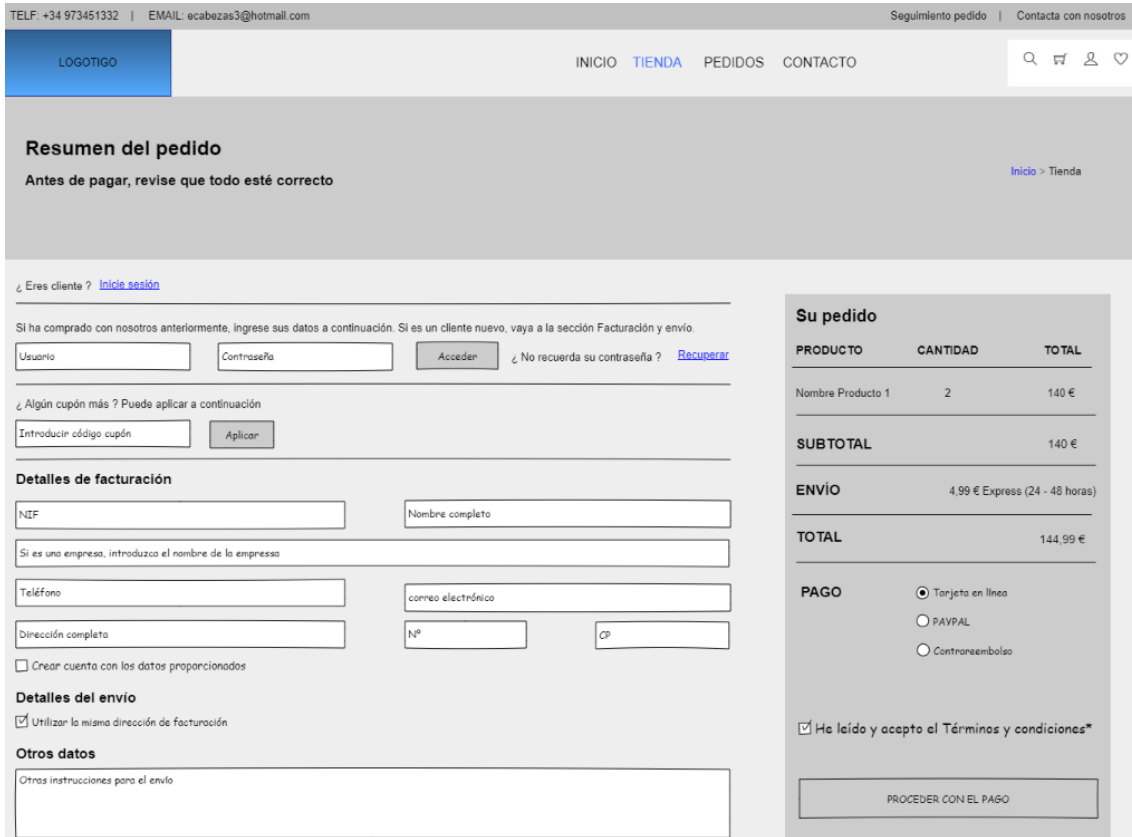
PRODUCTO	PRECIO / UNIDAD	CANTIDAD	TOTAL	ACCIONES
 <b>Titulo del producto</b> Subtítulo de producto	140 €	2	240 €	 

**Subtotal** 240 €

**Envío**  0 € - Estándar (3-4 días)  4,99 € Express (24-48 horas)  0 € - Contrarembolso

**Total** 244,99 €

Ilustración 11 Pantalla "Carrito de la compra"



TEL: +34 973451332 | EMAIL: ecabezas3@hotmail.com Seguimiento pedido | Contacta con nosotros

LOGOTIGO INICIO TIENDA PEDIDOS CONTACTO 🔍 🛒 👤 ❤️

### Resumen del pedido

Antes de pagar, revise que todo esté correcto Inicio > Tienda

¿ Eres cliente ? [Inicio sesión](#)

Si ha comprado con nosotros anteriormente, ingrese sus datos a continuación. Si es un cliente nuevo, vaya a la sección Facturación y envío.

Usuario  Contraseña   [¿ No recuerda su contraseña ? Recuperar](#)

¿ Algún cupón más ? Puede aplicar a continuación

Introducir código cupón

#### Detalles de facturación

NIF  Nombre completo

Si es una empresa, introduzca el nombre de la empresa

Teléfono  correo electrónico

Dirección completa  Nº  CP

Crear cuenta con los datos proporcionados

#### Detalles del envío

Utilizar la misma dirección de facturación

#### Otros datos

Otras instrucciones para el envío

#### Su pedido

PRODUCTO	CANTIDAD	TOTAL
Nombre Producto 1	2	140 €
<b>SUBTOTAL</b>		140 €
<b>ENVÍO</b>	4,99 € Express (24 - 48 horas)	
<b>TOTAL</b>	144,99 €	

**PAGO**

Tarjeta en línea

PAYPAL

Contrareembolso

He leído y acepto el **Términos y condiciones\***

Ilustración 12 Pantalla "Resumen del pedido"

El caso de uso CU-07 representa las acciones de añadir, eliminar y modificar un producto.

CU-07	Gestionar productos										
Versión	1.0 (20/04/23)										
Dependencias	<ul style="list-style-type: none"> <li>Añadir imagen</li> </ul>										
Precondición	Existe un usuario de tipo organización en el sistema y el usuario se ha identificado satisfactoriamente en él.										
Descripción	Un usuario de tipo organización deberá poder gestionar los productos que vende.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario pulsará al botón "Mis productos" de la barra lateral.</td> </tr> <tr> <td>2</td> <td>El sistema mostrará la vista "Lista de productos".</td> </tr> <tr> <td>3</td> <td>El usuario pulsa en el botón añadir producto.</td> </tr> <tr> <td>4</td> <td>El sistema mostrará la vista "Edición/Creación producto".</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario pulsará al botón "Mis productos" de la barra lateral.	2	El sistema mostrará la vista "Lista de productos".	3	El usuario pulsa en el botón añadir producto.	4	El sistema mostrará la vista "Edición/Creación producto".
	Paso	Acción									
	1	El usuario pulsará al botón "Mis productos" de la barra lateral.									
	2	El sistema mostrará la vista "Lista de productos".									
3	El usuario pulsa en el botón añadir producto.										
4	El sistema mostrará la vista "Edición/Creación producto".										

	Si el usuario quiere insertar una imagen al producto.
5	5.1 Se hace el caso de uso "Añadir imagen".
	5.2 Se vuelve al paso 5.
6	El usuario introducirá los datos del producto y pulsará al botón "Guardar".
	Si alguno de los datos introducidos contiene algún valor inválido.
7	7.1 El sistema informa que alguno de los campos introducidos contiene algún valor inválido.
	7.2 Se vuelve al paso 6.
8	El sistema indica que el producto se ha añadido correctamente.
Postcondición	El número de productos es el mismo de antes más uno y existe un producto con los datos introducidos.
Excepciones	Ninguna
Comentarios	El usuario debe introducir como mínimo una imagen del producto.

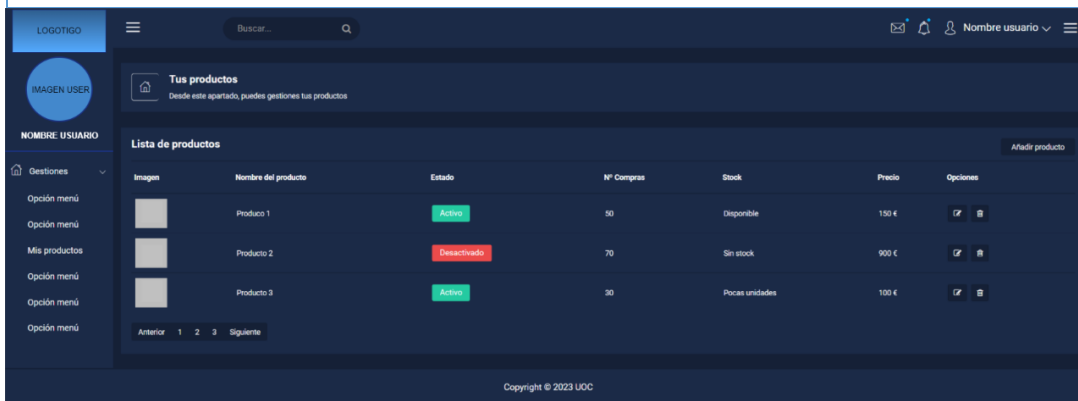


Ilustración 13 Pantalla "Tus productos"

<b>CU-08</b>	<b>Filtrar por precio</b>
Versión	1.0 (20/04/23)
Dependencias	Ninguna
Precondición	Ninguna
Descripción	Un usuario, identificado o no, deberá poder filtrar los productos por precio.



	Paso	Acción
Secuencia normal	1	El usuario pulsará al enlace "Tienda" de la barra de navegación.
	2	El sistema mostrará la vista "Catálogo de productos".
	3	El usuario asignará un rango de precios en la barra lateral izquierda.
	4	El sistema mostrará la vista "Catálogo de productos" con aquellos productos que están dentro del rango que ha asignado.
Postcondición	El número de productos en el carrito del usuario será igual que antes más uno con los datos del producto seleccionado junto a su cantidad.	
Excepciones	E1	Si no existe ningún producto dentro del rango de precios especificado.
	4	E1.1 El sistema muestra un mensaje diciendo que no existe ningún producto dentro de ese rango de precios.
	E1.2	Se cancela el caso de uso.
Comentarios	Ninguno	

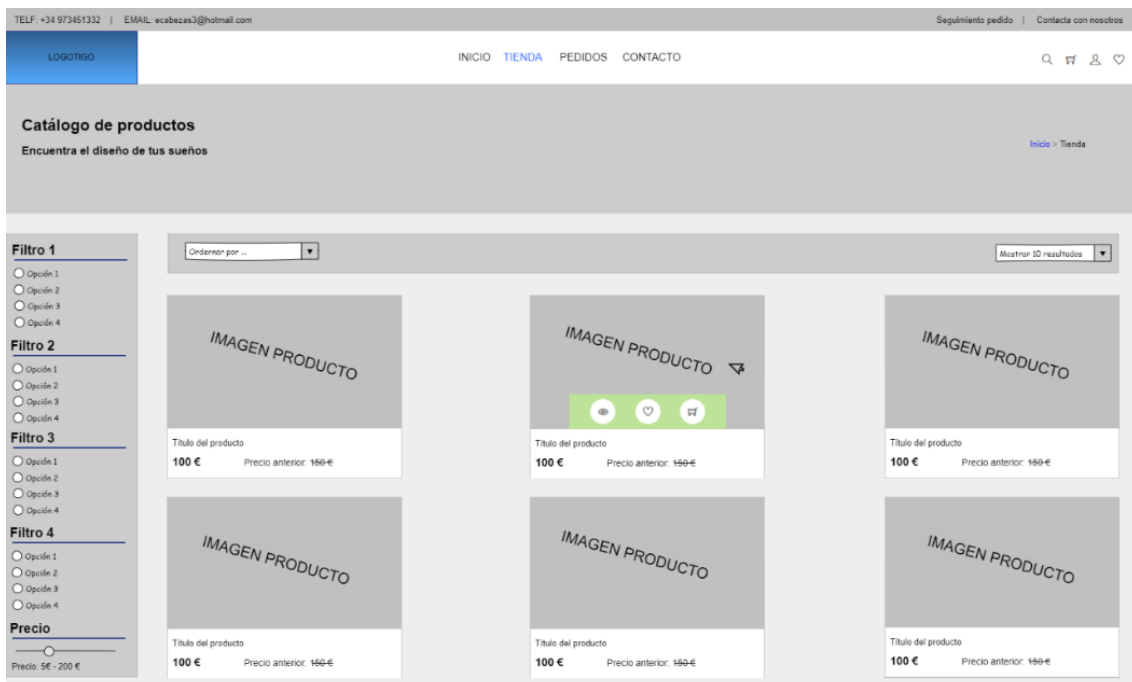


Ilustración 14 Pantalla "Catálogo de productos"

CU-09		Iniciar sesión	
Versión	1.0 (20/04/23)		
Dependencias	Ninguna		
Precondición	Existe un o más usuarios en el sistema y el usuario se encuentra en una vista que permite conectarse.		
Descripción	Un usuario deberá poder iniciar sesión en el sistema.		
Secuencia normal	Paso	Acción	
		Si el usuario quiere recuperar su contraseña.	
	1	1.1	El sistema muestra una vista para introducir su usuario.
		1.2	El usuario introduce su id de usuario y pulsa en “Recuperar”.
		1.3	El sistema indica que se ha enviado un correo para restablecer la contraseña.
	2	El usuario introduce su usuario y contraseña y pulsa en el botón “Acceder”.	
3	El sistema indica que el usuario se ha identificado correctamente y vuelve a la vista en la que se encontraba.		
Postcondición	El usuario estará conectado con su id de usuario.		
Excepciones	Paso	Acción	
		E1	Si el usuario y la contraseña no coinciden con ningún usuario de la base de datos.
	3	E1.1	El sistema indica que la contraseña introducida no es correcta.
		E1.2	Se cancela el caso de uso.
Comentarios	Ninguno		

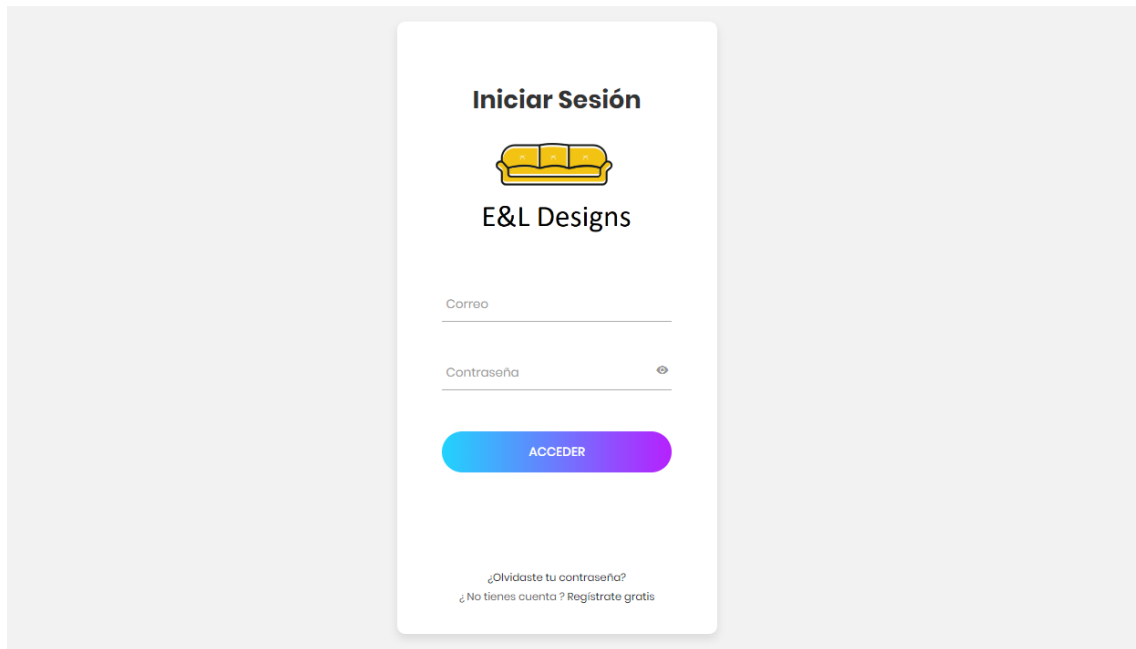


Ilustración 15 Pantalla "Inicio Sesión"

Se pueden visualizar las pantallas en mejor resolución, visitando:

<https://elohimcabezas.com/pantallas/>

#### 4.1.2. Diagrama de actores

El diagrama de actores resultante de los distintos casos de usos descritos en el apartado anterior es:

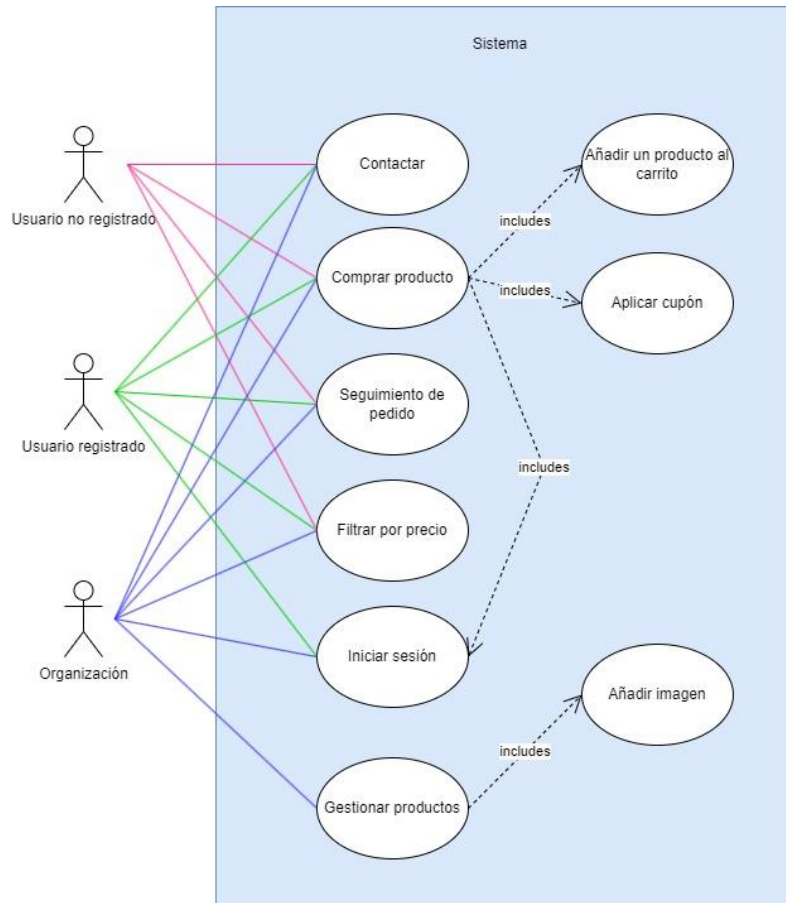


Ilustración 16 Diagrama de actores

#### 4.1.3. Diseño relacional de la base de datos

La estructura de la base de datos definida para este proyecto tiene una cardinalidad 3FN y cumple los principios de buenas prácticas definidos por Boyce-Cood, que establece un protocolo de regulación para que, en toda relación existente entre tablas, no exista dependencias terceras y por tanto el mantenimiento garantice la independencia de las partes.

Aseguramos así que ningún atributo que no forme parte de la clave primaria requiera ser mantenido por los cambios que se sucedan en otras, lo que se denomina "dependencia funcional". De esta forma sólo, y únicamente sólo, existen dependencias en atributos determinantes, es decir, en aquellos que forman parte de la clave de identificación y cuya finalidad única es la del mantenimiento de la integridad de los datos.



## 4.2 Arquitectura del sistema

La arquitectura de la solución está basada en el Modelo de Software Hexagonal. La selección sigue la tendencia del desarrollo enfocado al desacoplamiento total de las partes principales, sus tres capas (infraestructura, aplicación y dominio), permitiendo la independencia total de la codificación, tal como se observa en la siguiente imagen:

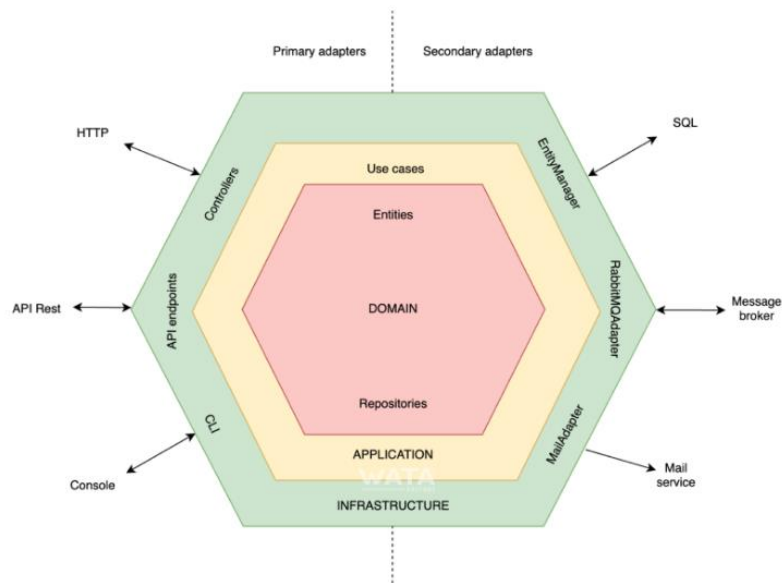


Ilustración 191 Diagrama de la estructura hexagonal

Con esta definición se va a conseguir, por un lado, un aislamiento lógico que se corresponde con la parte “interna” correspondiente a las capas de aplicación y dominio, donde se ve a definir los casos prácticos del negocio y describir las entidades. Y por otro, un desprendimiento de la responsabilidad de persistencia, o parte “externa”.

Históricamente, los datos, los repositorios, los controladores... se concebían como la parte interior, oculta y muy dependiente, donde se accedía de afuera a adentro. Con el modelo hexagonal, la infraestructura se encuentra en la capa más exterior y se ofrece a la aplicación con un sistema de interconexión mediante puertos.

La interconectividad mediante los puertos es la magia de la arquitectura. Se puede entender como una analogía del “*plug and play*” llevado al desarrollo estructural, cuya metodología permite cambiar objetos, relaciones, ... con mucha facilidad y sin afectar a otras capas. El desarrollador debe enfocarse solo en el protocolo de intercambio y olvidarse del resto, haciéndolo muy versátil y dinámico. Con este enfoque se facilita la integración de servicios externos y permite un desarrollo aislado, así como la simplicidad del testeo gracias a la liberación de dependencias.

Por debajo de la capa más exterior (capa de infraestructura), se encuentra la capa de aplicación, que es donde se concentran los casos de uso o servicios. La capa de infraestructura solo interactuará consigo misma enviándole los *inputs* y esta a su vez, hará lo mismo hacia la capa núcleo (capa de dominio), donde se encuentra la lógica de negocio.

La peculiaridad de esta estructura respecto a otras por capas es que esta solo se comunica con el mismo nivel y con su nivel inferior. Es decir, de las capas externas hacia las capas internas, donde cada capa tiene una única responsabilidad direccional y toda comunicación se realizará mediante puertos, que normalmente se conocen como puntos de acceso API, lo que permite al máximo la reutilización por terceros.

### 4.3 Diseño de la interfaz de usuario

El diseño de la interfaz de usuario es una parte esencial de nuestro servicio web, ya que buscamos brindar una experiencia visualmente atractiva, intuitiva y cómoda para todos los usuarios. El objetivo principal es crear una interfaz que refleje la identidad de nuestra plataforma y permita a los usuarios acceder y utilizar todas las funcionalidades de manera eficiente y sin problemas.

En términos de diseño visual, nos enfocaremos en un enfoque moderno y limpio. Utilizaremos un esquema de colores atractivo y coherente en todas las páginas para mantener una apariencia visualmente agradable y profesional. La tipografía y los elementos gráficos se seleccionarán cuidadosamente para garantizar una legibilidad óptima y una coherencia visual en toda la plataforma.

Un aspecto crucial será la adaptabilidad de la interfaz a diferentes dispositivos y tamaños de pantalla. Implementaremos un diseño responsivo que se ajuste automáticamente a las dimensiones de la pantalla del usuario, ya sea que estén accediendo desde un ordenador de escritorio, una tableta o un teléfono móvil. Esto asegurará que los usuarios tengan una experiencia consistente y fácil de usar sin importar el dispositivo que utilicen.

La página de inicio será una vitrina principal de nuestra plataforma, presentando de manera clara y llamativa las funcionalidades principales. Utilizaremos secciones destacadas para resaltar aspectos como la búsqueda de productos, las valoraciones de los usuarios y las últimas novedades. También mostraremos información relevante sobre las empresas y organizaciones asociadas, así como los diseñadores disponibles para contratar. Esto permitirá a los usuarios obtener una visión general de lo que nuestra plataforma ofrece y les ayudará a navegar rápidamente a las áreas de su interés.

La navegación se organizará de manera intuitiva y sencilla. Implementaremos un menú principal que permitirá a los usuarios acceder fácilmente a las diferentes secciones del servicio web, como la gestión de productos, la tienda online, el portal de contratación de diseñadores y la plataforma 3D. Además del menú principal, incluiremos un menú desplegable o una barra lateral para acceder a opciones adicionales y configuraciones de cuenta. Esto facilitará la navegación y garantizará que los usuarios puedan acceder a todas las áreas relevantes de la plataforma sin dificultad.

Cada página contendrá elementos visuales claros y llamativos. Los botones y enlaces estarán bien diferenciados y diseñados de manera que destaquen y sean fácilmente identificables. Utilizaremos íconos representativos y gráficos visuales para mejorar la comprensión de la información y guiar a los usuarios a través del flujo de acciones. Al utilizar un diseño limpio y espacioso, aseguraremos que los elementos en pantalla estén organizados de manera ordenada y sean fáciles de leer y seleccionar.

En las páginas de gestión de productos, nos enfocaremos en proporcionar una interfaz intuitiva y práctica para que las empresas y organizaciones puedan agregar, editar y eliminar sus productos. Utilizaremos formularios bien estructurados y amigables para recopilar la información necesaria, como descripciones, imágenes, precios y fechas de inicio de venta. Además, proporcionaremos herramientas visuales para cargar imágenes y previsualizar los diseños, lo que facilitará la creación y gestión de los productos.

En la tienda online, implementaremos opciones de filtrado avanzadas para permitir a los usuarios buscar productos según diferentes criterios, como precio, valoraciones, colores, etc. Mostraremos miniaturas claras de los productos junto con información relevante, como precios, descripciones y valoraciones. Destacaremos los botones de compra para que sean fácilmente accesibles y visibles, y proporcionaremos un proceso de pago online seguro y sencillo. Los usuarios podrán seleccionar métodos de pago y completar la transacción de manera fluida, garantizando una experiencia de compra conveniente y sin complicaciones.

El portal de contratación de diseñadores y la plataforma 3D contarán con interfaces diseñadas específicamente para cada funcionalidad. Los diseñadores podrán acceder a una interfaz 3D interactiva y fácil de usar, donde podrán crear y modificar los bocetos según las necesidades de las empresas y organizaciones.

**Portal gestión de productos** → <https://elohimcabezas.com/login>  
 Correo: ecabezas3@hotmail.com; Contraseña: 123

**Plataforma 3D** → <http://elohimcabezas.com/editor3D/>

**Tienda online** → <http://elohimcabezas.com/tienda/>

## 5. Implementación y desarrollo

### 5.1 Elección de tecnologías y herramientas

He elegido las tecnologías PHP, MySQL, Bootstrap y Ajax para mi trabajo final debido a las ventajas y capacidades que ofrecen en el desarrollo web. Estas tecnologías combinadas proporcionan un conjunto completo de herramientas y características que me permiten construir un sitio web dinámico, funcional y atractivo. A continuación, vamos a presentar los argumentos clave para respaldar mi elección:

1. **PHP:** PHP es un lenguaje de programación del lado del servidor ampliamente utilizado en el desarrollo web. Su popularidad se debe a su facilidad de uso, su amplia comunidad de desarrolladores y su capacidad para interactuar con diferentes sistemas y bases de datos. Al elegir PHP, puedo aprovechar su sintaxis sencilla y su amplia disponibilidad de recursos y documentación, lo que facilita el desarrollo de aplicaciones web robustas y escalables.



2. **MySQL:** MySQL es un sistema de gestión de bases de datos relacional que se integra perfectamente con PHP. Al elegir MySQL, puedo almacenar y administrar eficientemente grandes volúmenes de datos en mi sitio web. MySQL es conocido por su rendimiento, seguridad y escalabilidad, lo que garantiza un acceso rápido y confiable a la información. Además, la compatibilidad entre PHP y MySQL es excelente, lo que facilita la interacción y manipulación de datos en el desarrollo de mi trabajo final.
3. **Bootstrap:** Bootstrap es un *framework* de desarrollo *front-end* que proporciona una amplia variedad de componentes y estilos predefinidos. Al elegir Bootstrap, puedo acelerar el proceso de diseño y desarrollo de mi sitio web, ya que ofrece una estructura de diseño receptivo y flexible. Con Bootstrap, puedo crear rápidamente una interfaz de usuario atractiva y responsiva, adaptada a diferentes dispositivos y tamaños de pantalla. Además, la comunidad activa de Bootstrap ofrece una gran cantidad de documentación y recursos, lo que facilita la personalización y la mejora de la apariencia visual de mi trabajo final.
4. **Ajax:** Ajax (*Asynchronous JavaScript and XML*) es una técnica que permite realizar solicitudes asíncronas al servidor sin tener que recargar toda la página. Al elegir Ajax, puedo mejorar la experiencia del usuario al brindar una navegación más fluida y respuestas en tiempo real. Con Ajax, puedo actualizar dinámicamente ciertas partes de mi sitio web sin interrumpir la interacción del usuario, lo que proporciona una sensación de velocidad y fluidez. Además, Ajax se integra fácilmente con PHP y otras tecnologías, lo que me permite realizar operaciones y actualizaciones de forma eficiente y sin problemas en mi trabajo final.

Así pues, la elección de las tecnologías PHP, MySQL, Bootstrap y Ajax para mi trabajo final se basa en su facilidad de uso, su amplia compatibilidad, su escalabilidad y su capacidad para brindar una experiencia de usuario mejorada.

A continuación, vamos a describir las herramientas (*software*) que hemos utilizado.

### Visual Studio Code

En primer lugar, Visual Studio Code es conocido por su eficiencia y rendimiento. Es un IDE ligero y rápido, lo que significa que se inicia rápidamente y no consume demasiados recursos del sistema. Esto es especialmente beneficioso al trabajar en proyectos grandes o en computadoras con capacidades limitadas.

Además, Visual Studio Code se destaca por su amplia gama de extensiones. Estas extensiones permiten personalizar y adaptar el entorno de desarrollo según las necesidades específicas de cada programador. Ya sea que estés trabajando en JavaScript, Python, Java u otros lenguajes de programación, vamos a encontrar extensiones que brindan soporte y mejoras para facilitar nuestros desarrollos.

Otra ventaja importante es la compatibilidad multiplataforma de Visual Studio Code. Podemos utilizarlo en sistemas operativos como Windows, macOS y Linux, lo que facilita el trabajo en diferentes entornos y equipos de desarrollo con configuraciones diversas.

Visual Studio Code también cuenta con herramientas de depuración integradas que simplifican la identificación y corrección de errores en el código. Podemos establecer puntos de interrupción, inspeccionar variables y seguir el flujo de ejecución para resolver problemas de manera efectiva.

Por último, Visual Studio Code cuenta con una comunidad activa de usuarios. Esta comunidad comparte conocimientos, extensiones y soluciones a problemas comunes.

### **XAMPP**

XAMPP proporciona una solución todo en uno para el desarrollo local de aplicaciones web. Incluye los componentes esenciales para ejecutar un servidor web en tu propio entorno de desarrollo, como Apache, MySQL, PHP y Perl. Esto permite crear y probar aplicaciones web sin necesidad de configurar cada componente individualmente.

Además, XAMPP es fácil de instalar y usar. Su proceso de instalación es rápido y sencillo, lo que facilita su configuración en diferentes sistemas operativos. Además, viene con una interfaz intuitiva y amigable que permite administrar y controlar los servicios del servidor de manera conveniente.

XAMPP también es altamente personalizable y flexible. Podemos habilitar o deshabilitar los servicios según nuestras necesidades específicas, lo que nos permite adaptar el entorno de desarrollo a nuestros distintos proyectos. Además, XAMPP es compatible con una amplia gama de aplicaciones y *frameworks* populares, lo que nos brinda la libertad de utilizar las herramientas que mejor se ajusten a nuestros requerimientos.

Otra ventaja de XAMPP es su portabilidad. Podemos ejecutar nuestro servidor web localmente en nuestra máquina, lo que nos brinda la flexibilidad de desarrollar y probar aplicaciones sin conexión a internet. Esto es especialmente útil cuando trabajamos en entornos donde la conectividad puede ser limitada o inestable.

### **OVH**

OVH proporciona una amplia gama de servicios de alojamiento web, que incluyen opciones compartidas, servidores dedicados, servidores virtuales privados (VPS) y alojamiento en la nube. Esto permite a los usuarios elegir la solución que mejor se adapte a sus necesidades y escalar según el crecimiento de su proyecto.

Además, OVH se destaca por su enfoque en la seguridad. Sus centros de datos están equipados con medidas de seguridad física avanzadas, como vigilancia las 24 horas, sistemas de detección de intrusiones y acceso restringido. También ofrecen opciones de seguridad adicionales, como certificados SSL y firewalls, para proteger los datos y las transacciones en línea.

OVH también se caracteriza por su rendimiento y confiabilidad. Sus infraestructuras están diseñadas para garantizar tiempos de actividad elevados y velocidades de carga rápidas. Utilizan

tecnologías avanzadas, como almacenamiento en SSD, balanceo de carga y redes de alta capacidad, para ofrecer un rendimiento óptimo a los sitios web y aplicaciones alojadas.

Otra ventaja de OVH es su enfoque en la flexibilidad y la escalabilidad. Sus servicios permiten a los usuarios ajustar los recursos asignados a medida que sus necesidades cambian, lo que les brinda la capacidad de escalar vertical u horizontalmente según sea necesario. Esto es especialmente útil para proyectos que experimentan fluctuaciones en el tráfico o que requieren recursos adicionales en momentos específicos.

OVH también ofrece una amplia gama de servicios complementarios, como registro de dominios, soluciones de correo electrónico, backups automáticos y herramientas de administración de servidores. Estos servicios adicionales permiten a los usuarios tener un enfoque integral en la gestión de sus sitios web y aplicaciones en línea.

### **JetBrains YouTrack**

YouTrack destaca por su interfaz intuitiva y fácil de usar, lo que facilita la creación y seguimiento de problemas. Permite a los equipos organizar y priorizar tareas, asignar responsabilidades y realizar un seguimiento del progreso de los proyectos de manera eficiente.

Además, YouTrack ofrece potentes capacidades de personalización. Los usuarios pueden adaptar la herramienta a sus procesos de desarrollo específicos, definir flujos de trabajo personalizados, crear campos personalizados y generar informes personalizados. Esto permite una mayor flexibilidad y adaptabilidad a las necesidades de cada equipo.

YouTrack también brinda una visión general clara y completa de los proyectos. Permite realizar un seguimiento del tiempo invertido en tareas, gestionar sprints y realizar análisis detallados del estado y el progreso de los proyectos. Esto ayuda a los equipos a tener una mejor comprensión de su rendimiento y a tomar decisiones informadas.

Una ventaja adicional de YouTrack es su integración con otras herramientas populares de JetBrains, como IntelliJ IDEA y PyCharm. Esto facilita la colaboración y mejora la productividad al permitir a los desarrolladores acceder y administrar los problemas directamente desde su entorno de desarrollo.

### **HeidiSQL**

HeidiSQL es una opción destacada como cliente de base de datos para administrar y gestionar bases de datos MySQL, PostgreSQL y Microsoft SQL Server. Ofrece una serie de características que lo convierten en una elección popular para desarrolladores y administradores de bases de datos.

HeidiSQL se destaca por su interfaz intuitiva y fácil de usar. Permite a los usuarios navegar y administrar bases de datos de manera eficiente, realizar consultas SQL, importar y exportar datos, y realizar tareas de administración como copias de seguridad y restauración.

Además, HeidiSQL ofrece una amplia gama de herramientas y funciones que mejoran la productividad del desarrollador. Estas incluyen resaltado de sintaxis, autocompletado de código, generador de consultas visuales, editor de tablas y visor de datos. Estas características facilitan el desarrollo y la manipulación de datos de manera efectiva.

HeidiSQL también proporciona opciones avanzadas de administración de bases de datos. Permite crear y gestionar usuarios y permisos, realizar optimizaciones de rendimiento, monitorear la actividad de la base de datos y ejecutar scripts SQL de forma automatizada. Esto ayuda a los administradores de bases de datos a mantener y optimizar el rendimiento de sus sistemas.

Una ventaja adicional de HeidiSQL es su compatibilidad multiplataforma. Está disponible para sistemas operativos Windows y Linux, lo que brinda flexibilidad a los usuarios que trabajan en diferentes entornos.

## 5.2 Desarrollo de los módulos principales

A continuación, vamos a explicar cómo se han desarrollado los distintos 4 módulos principales de la aplicación. Veremos las características más destacadas de estos y sus funcionalidades.

### Módulo de gestión de datos

- **Creación de usuarios:** Se ha implementado un formulario de registro donde las empresas y organizaciones puedan proporcionar información relevante, como nombre, dirección de correo electrónico y contraseña. Una vez que se envía el formulario, se validan los datos y se almacenan de forma segura en la base de datos. Se han utilizado técnicas de encriptación para proteger las contraseñas de los usuarios.

- **Gestión de productos:** Se ha desarrollado una interfaz administrativa en la que las empresas pueden agregar nuevos productos proporcionando detalles como nombre, descripción, precio, imágenes y categoría. También se les permite editar y eliminar productos existentes. Estos datos se almacenan en la base de datos.

- **Edición de datos maestros:** Se ha definido una interfaz para que los administradores del sistema puedan editar y actualizar los datos maestros de toda la aplicación, como categorías, métodos de pago, estados, roles, detalles de envío y opciones de personalización. Estos datos se almacenan en la base de datos y se utilizan para completar automáticamente los formularios y los procesos relacionados con la gestión de productos.

- **Creación de datos maestros:** Para permitir a los administradores crear datos maestros personalizados, se ha implementado una funcionalidad que les permite definir nuevas categorías de productos, campos adicionales o cualquier otro atributo que sea relevante para el negocio.

### Módulo de login

- **Inicio de sesión:** Se ha implementado una página de inicio de sesión donde los usuarios pueden ingresar su nombre de usuario y contraseña. Los datos se verificarían en la base de datos y, si son correctos, se establece una sesión activa para el usuario. Se utilizan técnicas de autenticación segura, como el uso de tokens de sesión y el cifrado de las comunicaciones.

- **Registro de usuario:** Se ha implementado una página de registro donde los usuarios pueden ingresar su nombre completo, correo y contraseña deseada. Si los datos son correctos, se envía un correo de verificación al usuario con un pin de 4 dígitos que deberá verificar para poder registrar su correo correctamente.

- **Recuperación de contraseña:** Si un usuario olvida su contraseña, tiene a su disposición la funcionalidad de "olvidé mi contraseña" que permite al usuario proporcionar su dirección de correo electrónico. Se envía un correo electrónico con un enlace seguro para restablecer la contraseña. Al hacer clic en el enlace, el usuario será redirigido a una página donde puede ingresar una nueva contraseña.

### Módulo de tienda virtual

- **Búsqueda y filtrado de productos:** Se ha implementado una función de búsqueda que permita a los usuarios buscar productos por nombre, categoría, precio y otros criterios. Estos criterios se utilizan para consultar la base de datos y mostrar los resultados correspondientes en la interfaz de la tienda.

- **Carrito de compras:** Se ha desarrollado un carrito de compras que permite a los usuarios agregar productos seleccionados mientras navega por la tienda. El estado del carrito se almacena en la sesión del usuario (si ha iniciado sesión) o en la sesión del navegador (temporal), lo que permite a los usuarios ver y modificar el contenido de su carrito antes de proceder al pago.

- **Proceso de pago en línea:** Se ha integrado una pasarela de pago simulada, como PayPal, para permitir a los usuarios realizar pagos en línea de forma segura. Se proporciona una interfaz de pago donde los usuarios ingresan su información financiera y se comunican ficticiamente con la pasarela de pago para procesar la transacción (en nuestro caso, simulamos dicha transacción). Una vez completado el pago, se registran los detalles de la transacción y se notifica al usuario sobre el éxito de la compra, generando en su cuenta un registro de pedido nuevo, donde podrá realizar el seguimiento de este, así como ver el estado y la fecha estimada de entrega.

- **Valoraciones y comentarios:** Se ha implementado una funcionalidad que permite a los usuarios dejar valoraciones y comentarios sobre los productos que han comprado. Estos comentarios son visibles para otros usuarios interesados en el producto, brindando información adicional y fomentando la confianza en la compra.

### Módulo de chat

- **Comunicación entre usuarios:** Se ha desarrollado un sistema de chat interno que permite a los usuarios comunicarse entre sí. Incluye una interfaz de chat en tiempo real donde los usuarios

puedan enviar y recibir mensajes. Se utilizan tecnologías como *WebSockets* para lograr una comunicación bidireccional entre el cliente y el servidor, lo que permite una experiencia de chat en tiempo real.

- **Soporte de asistentes:** Los usuarios pueden ser atendidos por asistentes. Estos asistentes podrían ser empleados de servicio al cliente o representantes de la plataforma que brinden soporte personalizado a los usuarios. Los asistentes recibirían notificaciones sobre nuevos mensajes y podrían responder a las consultas, resolver problemas o brindar asesoramiento según sea necesario.

### 5.3 Pruebas y depuración

En este apartado se han definido 4 pruebas para el proyecto *E&L Designs*. Estas pruebas son de funcionalidad, rendimiento y escalabilidad, seguridad, usabilidad y, por último, definiremos la depuración y seguimiento de errores del proyecto.

#### Pruebas de funcionalidad

- **Pruebas de registro y autenticación:** Se verifica que los usuarios puedan registrarse correctamente proporcionando la información requerida, como nombre, dirección de correo electrónico y contraseña. Se comprueba que los datos se almacenen correctamente en la base de datos y que los usuarios puedan iniciar sesión con las credenciales registradas. También se prueba el cierre de sesión para asegurarse de que los usuarios puedan salir de sus cuentas de manera adecuada.

- **Pruebas de gestión de datos:** Se analiza la funcionalidad de creación, edición y eliminación de usuarios, productos y datos maestros. Por ejemplo, se verifica que las empresas puedan agregar nuevos productos proporcionando detalles como nombre, descripción, precio e imágenes. También se comprueba que los cambios realizados en los datos maestros se reflejen correctamente en el sistema.

- **Pruebas de búsqueda y filtrado:** Se realizan pruebas exhaustivas de la función de búsqueda y los filtros de productos. Se prueba la capacidad de búsqueda por nombre, categoría, precio u otros criterios relevantes. Además, se verifica que los resultados de búsqueda sean precisos y relevantes, y que los filtros se apliquen correctamente.

- **Pruebas de carrito de compras:** Se prueba la funcionalidad del carrito de compras, asegurándose de que los usuarios puedan agregar y eliminar productos del carrito. Se verifica que el precio total y la cantidad de productos en el carrito se actualicen correctamente al realizar cambios. También se prueba la finalización del pedido para garantizar que los usuarios puedan confirmar su compra con éxito.

- **Pruebas de proceso de pago:** Se realizan pruebas exhaustivas del proceso de pago en línea. Se prueba la integración con una pasarela de pago, como PayPal, para asegurarse de que las transacciones se procesen correctamente y de forma segura. Se verifica que los usuarios puedan ingresar su información financiera y que se registren los detalles de la transacción de manera adecuada.

### Pruebas de rendimiento y escalabilidad

- **Pruebas de carga:** Se simulan cargas de usuarios simultáneos en el sistema para evaluar su rendimiento bajo condiciones de alta demanda. Estas pruebas ayudan a identificar posibles cuellos de botella y asegurarse de que el sistema pueda manejar la carga sin degradar su rendimiento.

- **Pruebas de tiempo de carga:** Se mide y se optimiza el tiempo de carga de las páginas y funcionalidades clave. Esto implica identificar y solucionar cualquier demora o retraso en la carga de contenido para garantizar una experiencia de usuario fluida y rápida.

- **Pruebas de escalabilidad:** Se evalúa la capacidad del sistema para escalar y crecer de manera efectiva. Se realizan pruebas para determinar cómo responde el sistema al aumentar el número de usuarios, transacciones y productos. Esto nos ayuda a garantizar que el sistema pueda manejar el crecimiento futuro sin comprometer su rendimiento.

### Pruebas de seguridad

- **Pruebas de autenticación y autorización:** Se verifica que el sistema implemente correctamente la autenticación y autorización de usuarios. Se prueba que solo los usuarios autenticados y autorizados puedan acceder a las funcionalidades correspondientes y que los permisos se apliquen correctamente según el rol de usuario.

- **Pruebas de protección de datos:** Se asegura que la información confidencial, como contraseñas y datos financieros, esté protegida adecuadamente. Se implementan técnicas de cifrado y se siguen buenas prácticas de seguridad para evitar brechas de seguridad y proteger la privacidad de los usuarios.

- **Pruebas de vulnerabilidad:** Se realizan pruebas de seguridad para identificar posibles vulnerabilidades en el sistema, como ataques de inyección SQL, cross-site scripting (XSS) y otros ataques comunes. Se implementan medidas de seguridad adecuadas para mitigar estos riesgos y garantizar la integridad del sistema.

### Pruebas de usabilidad

- **Pruebas de navegación y flujo de usuario:** Se evalúa la facilidad de uso y la intuitividad de la interfaz de usuario. Se realizan pruebas para asegurarse de que los usuarios puedan navegar de manera fluida y realizar las acciones deseadas sin confusiones. Se busca optimizar la disposición y el diseño de los elementos de la interfaz para que sean intuitivos y fáciles de entender.

- **Pruebas de responsive design:** Se prueba la capacidad de la aplicación web para adaptarse y funcionar correctamente en diferentes dispositivos y tamaños de pantalla. Se verifica que la interfaz se vea bien y sea totalmente funcional en ordenadores de escritorio, dispositivos móviles y tabletas. Esto implica realizar pruebas en diferentes dispositivos y navegadores para asegurarse de que la experiencia del usuario sea consistente en todos ellos.

- **Pruebas de accesibilidad:** Se asegura que la aplicación cumpla con los estándares de accesibilidad web. Esto incluye proporcionar alternativas para usuarios con discapacidades visuales o de otro tipo, como texto alternativo para imágenes, soporte para lectores de pantalla y navegación por teclado. Se realizan pruebas para garantizar que la aplicación sea accesible para todos los usuarios, sin importar sus limitaciones.

### Depuración y seguimiento de errores

- **Identificación y corrección de errores:** Durante todo el proceso de desarrollo y pruebas, se ha realizado un seguimiento de los errores y se han ido identificando las causas subyacentes. Se han depurado los problemas encontrados y se han corregido los errores para garantizar el correcto funcionamiento del sistema.

- **Registro y seguimiento de errores:** Como mejora del sistema, se implementaría un sistema de registro de errores para capturar y registrar información detallada sobre los errores encontrados. Esto ayudaría a rastrear y analizar los problemas, lo que facilitaría su resolución más eficiente y rápida. Además, se podría utilizar un sistema de seguimiento de errores, como Jira o Bugzilla, para organizar y gestionar eficazmente los problemas identificados.

## 6. Evaluación y resultados

### 6.1 Validación del sistema

La validación del sistema es un paso crucial en el desarrollo de nuestro proyecto. Consiste en asegurarse de que el sistema cumpla con los requisitos y expectativas establecidos por los usuarios y las partes interesadas. La validación del sistema implica diferentes actividades para evaluar y confirmar que el sistema cumple con los criterios de aceptación definidos. A continuación, explicaremos algunas consideraciones importantes sobre la validación del sistema de nuestro proyecto **E&L Designs**:

1. **Requisitos de validación:** Antes de comenzar la validación del sistema, es necesario definir claramente los requisitos de validación. Estos requisitos deben ser específicos, medibles y verificables. Pueden incluir aspectos funcionales, de rendimiento, de seguridad y de usabilidad que son críticos para el éxito de nuestro proyecto.

2. **Diseño de casos de prueba:** Para validar el sistema, es necesario diseñar casos de prueba que cubran todos los aspectos relevantes del sistema. Estos casos de prueba deben estar alineados con los requisitos de validación establecidos previamente. Los casos de prueba deben incluir escenarios típicos, casos límite y situaciones de error para garantizar una cobertura adecuada.

3. **Ejecución de pruebas:** Una vez que los casos de prueba estén diseñados, se procederá a ejecutarlos en el sistema. Durante la ejecución de las pruebas, se registrarán los resultados y se verificará si el sistema cumple con los requisitos y expectativas establecidos. Se deben documentar los problemas encontrados y realizar un seguimiento adecuado para su resolución.



4. **Verificación de requisitos funcionales:** En la validación del sistema, se verificará si las funciones principales y los procesos del sistema están operando correctamente. Se comprobará que los usuarios puedan realizar las acciones previstas, como registrar usuarios, gestionar productos, realizar compras y utilizar el chat interno. Además, se revisarán las funcionalidades específicas relacionadas con los perfiles de diseñador y la plataforma 3D.

5. **Evaluación de rendimiento y escalabilidad:** También es importante evaluar el rendimiento del sistema durante la validación. Se realizarán pruebas de carga para asegurarse de que el sistema pueda manejar una gran cantidad de usuarios y transacciones sin degradar su rendimiento. Esto incluye verificar el tiempo de respuesta del sistema, la capacidad de procesamiento y la escalabilidad del sistema ante picos de demanda.

6. **Verificación de seguridad:** La validación del sistema también incluye la verificación de las medidas de seguridad implementadas. Se realizarán pruebas para identificar posibles vulnerabilidades y asegurarse de que los datos de los usuarios estén protegidos adecuadamente. Esto puede incluir pruebas de inyecciones y verificación del cumplimiento de estándares de seguridad.

7. **Evaluación de usabilidad:** La validación del sistema también implica evaluar la usabilidad de la aplicación. Se realizarán pruebas de usabilidad para asegurarse de que la interfaz de usuario sea intuitiva, fácil de usar y cumpla con las expectativas de los usuarios. Se verificará la accesibilidad en diferentes dispositivos y se recopilarán comentarios de los usuarios para realizar mejoras.

8. **Aprobación del sistema:** Una vez completada la validación y resueltos los problemas identificados, el sistema estará listo para su aprobación. Esto implica obtener la aceptación formal de las partes interesadas y los usuarios para confirmar que el sistema cumple con los requisitos establecidos y está listo para su implementación y uso en producción.

Dicho todo esto, podemos definir las siguientes validaciones que nos permitirán asegurarnos de que nuestro servicio web cumpla con los estándares de calidad, brindando una experiencia óptima a los usuarios, garantizando el éxito del proyecto:

#### 1. Pruebas de funcionalidad:

- Verificar que todas las funcionalidades principales de la aplicación web, como la gestión de productos, el portal de compras, las valoraciones de productos, el chat interno y la plataforma 3D, estén implementadas correctamente y sean accesibles para los usuarios.
- Realizar pruebas exhaustivas de cada función para asegurar de que se comporten según lo esperado y no presenten errores o comportamientos inesperados.
- Comprobar que los procesos de agregar, editar, eliminar y buscar productos funcionen correctamente y actualicen la información en la base de datos de manera adecuada.

## 2. Pruebas de usabilidad:

- Observar cómo los usuarios interactúan con la aplicación web y evaluar su facilidad de uso y comprensión de las características.
- Realizar pruebas con usuarios reales para obtener retroalimentación sobre la navegación, el diseño de la interfaz de usuario y la disposición de los elementos.
- Verificar que las acciones más comunes, como buscar productos, agregar elementos al carrito de compras o dejar valoraciones, sean intuitivas y fáciles de realizar.

## 3. Pruebas de rendimiento:

- Evaluar la velocidad de carga de la aplicación web y asegurarnos de que las páginas se carguen con rapidez, especialmente aquellas con mayor demanda.
- Realizar pruebas de carga para verificar cómo se comporta el sistema bajo condiciones de alto tráfico, simulando un gran número de usuarios accediendo simultáneamente.
- Monitorear y medir el tiempo de respuesta del servidor y la capacidad de escalabilidad para garantizar que el sistema pueda manejar la demanda sin problemas de rendimiento.

## 4. Pruebas de seguridad:

- Realizar análisis de seguridad para identificar posibles vulnerabilidades en la aplicación web y en la infraestructura subyacente.
- Realizar pruebas de inyección para intentar acceder a datos o funcionalidades sin autorización y asegurarnos de que el sistema resista ataques.
- Implementar medidas de seguridad como encriptación de datos, autenticación segura y control de acceso adecuado para proteger la información de los usuarios y prevenir amenazas de seguridad.

## 5. Pruebas de integración:

- Verificar que todas las integraciones con sistemas externos, como pasarelas de pago o herramientas de diseño, funcionen correctamente.
- Realizar pruebas para garantizar la transferencia correcta de datos entre la aplicación web y los sistemas externos, asegurando que la comunicación sea fluida y sin errores.
- Asegurarte de que los datos se sincronicen adecuadamente y que no haya pérdida o corrupción de información durante el proceso de integración.

## 6. Pruebas de compatibilidad:

- Probar la aplicación web en diferentes navegadores web populares, como Chrome, Firefox, Safari, para asegurarte de que se vea y funcione correctamente en cada uno de ellos.
- Realizar pruebas en dispositivos móviles, como teléfonos y tabletas, para verificar que la plataforma sea compatible y se adapte correctamente a diferentes tamaños de pantalla y resoluciones.
- Asegurarte de que la experiencia de usuario sea consistente y de calidad en todas las plataformas y dispositivos utilizados por los usuarios.

## 7. Pruebas de carga:

- Realizar pruebas de carga simulando un gran número de usuarios accediendo a la aplicación web simultáneamente.
- Observar cómo se comporta el sistema bajo esta carga de trabajo y verificar si el rendimiento se mantiene estable.
- Identificar posibles cuellos de botella, tiempos de respuesta lentos o fallas en el proceso y solucionarlos para garantizar que el sistema pueda manejar la demanda esperada.

### 6.2 Análisis de resultados obtenidos

El análisis de los resultados obtenidos en el proyecto es crucial para evaluar el rendimiento, la eficacia y el impacto de nuestro servicio web. Para ellos, podemos llevar a cabo un análisis exhaustivo de los resultados obtenidos. Este análisis nos brinda información valiosa para identificar áreas de mejora, tomar decisiones estratégicas y planificar acciones futuras. Utilizaremos estos análisis para optimizar nuestro servicio web, ajustar nuestras estrategias y maximizar el valor que brinda a las empresas, organizaciones y usuarios. A continuación, enumeraremos algunas consideraciones importantes para poder llevar a cabo dicho análisis:

**1. Evaluación de la funcionalidad:** Analizar si todas las funcionalidades de la aplicación web están cumpliendo con los objetivos establecidos. Revisar si la gestión de productos, el portal de compras, las valoraciones de productos, el chat interno y la plataforma 3D están funcionando correctamente y si están brindando los resultados esperados.

**2. Medición de la usabilidad:** Examinar la retroalimentación de los usuarios y las pruebas de usabilidad realizadas para evaluar la facilidad de uso de la plataforma web. Analizar si los usuarios encuentran intuitivas las acciones y los flujos de trabajo, si la navegación es clara y si la interfaz de usuario es atractiva y accesible.

**3. Rendimiento y escalabilidad:** Analizar los resultados de las pruebas de rendimiento y carga para evaluar cómo responde nuestro servicio web bajo diferentes niveles de tráfico. Observar si

se cumplen los tiempos de respuesta aceptables, si hay problemas de rendimiento o cuellos de botella identificados y si la plataforma web es capaz de escalar correctamente para manejar la demanda.

**4. Seguridad y protección de datos:** Evaluar si se han cumplido los objetivos de seguridad establecidos. Analizar los resultados de las pruebas de seguridad y la efectividad de las medidas implementadas para proteger la información de los usuarios. Verificar si se han identificado y solucionado posibles vulnerabilidades y si se han implementado protocolos de seguridad adecuados.

**5. Retroalimentación de los usuarios:** Analizar las opiniones, comentarios y sugerencias de los usuarios sobre el servicio web. Prestar atención a las valoraciones, los comentarios en el chat interno y cualquier otro canal de comunicación con los usuarios. Identificar patrones y tendencias en la retroalimentación para identificar áreas de mejora y oportunidades de crecimiento.

**6. Cumplimiento de objetivos y logros:** Evaluar si los resultados obtenidos cumplen con los objetivos y logros que nos planteamos al inicio de este proyecto. Comparar los resultados con los indicadores clave de desempeño establecidos y determina si se han alcanzado los hitos definidos.

**7. Impacto en el negocio:** Analizar cómo el servicio web ha impactado a las empresas, organizaciones y usuarios que lo utilizan. Evaluar si se han logrado mejoras en la gestión de productos, en las ventas de las entidades y en la experiencia de compra de los usuarios.

## 7. Conclusiones

### 7.1 Cumplimiento de los objetivos

En cuanto a la aplicación web, he desarrollado un sistema completo de gestión que permite a las empresas y organizaciones administrar de manera eficiente sus productos. Esto lo he logrado implementando funcionalidades de CRUD (Crear, Leer, Actualizar, Eliminar) para entidades como usuarios, roles, productos, empresas y perfiles. La aplicación también cuenta con un sistema de inicio de sesión seguro que incluye medidas adicionales de autenticación, como la doble autenticación y el envío de correos electrónicos para verificar la identidad de los usuarios.

Para facilitar la comunicación entre las diferentes entidades, he creado canales de comunicación asíncronos y síncronos. Los usuarios pueden recibir notificaciones y también participar en chats en tiempo real con otros usuarios y administradores. Esto ha mejorado la interacción y la fluidez de la comunicación dentro del sistema.

En términos de diseño, he logrado crear una interfaz atractiva y responsive. He utilizado tecnologías como Bootstrap, jQuery y Ajax para desarrollar una interfaz que se adapta a diferentes dispositivos y tamaños de pantalla. Esto ha mejorado la experiencia del usuario y ha hecho que la aplicación sea fácil de usar y visualmente agradable.

Cada uno de los roles en el sistema tiene su propio espacio personalizado, lo que permite una experiencia adaptada a las necesidades específicas de cada usuario o entidad. Esto ha mejorado

la comodidad y la eficiencia en la gestión de productos y la comunicación entre las diferentes partes involucradas.

Para facilitar la búsqueda de productos, he implementado filtros que permiten a los usuarios aplicar diferentes criterios de búsqueda, como precio, categorías, fecha de publicación, etc. Esto ha mejorado la experiencia de búsqueda y ha ayudado a los usuarios a encontrar los productos que les interesan más rápidamente.

En cuanto al proceso de compra, he simulado todo el flujo, desde la selección de productos hasta el pago en línea. Los usuarios pueden agregar productos a su carrito de compra, gestionarlo y realizar pagos de manera segura. Esto ha proporcionado una experiencia completa de compra en línea y ha facilitado el proceso para los usuarios.

Además, he desarrollado una plataforma web que permite a los usuarios con perfiles de diseño crear bocetos en 3D de manera fácil e intuitiva. Estos bocetos se pueden descargar, cargar y gestionar dentro de la plataforma. Esto ha agilizado el proceso de diseño y ha brindado a los usuarios una herramienta poderosa para materializar sus ideas.

En resumen, he logrado alcanzar todos los objetivos establecidos en mi proyecto. He desarrollado una aplicación web funcional y completa que brinda una gestión eficiente de productos, una comunicación fluida, un diseño atractivo y responsive, así como una experiencia personalizada para los diferentes roles del sistema. Además, he implementado funcionalidades de búsqueda, compra y una plataforma 3D para potenciar aún más la experiencia de los usuarios.

## 7.2 Aportaciones y limitaciones

Respecto a las aportaciones, destacaríamos las siguientes:

**1. Simplificación de la gestión de productos:** El servicio web ofrece una solución conveniente y eficiente para que empresas y organizaciones puedan gestionar sus productos de manera más sencilla. Al proporcionar una plataforma centralizada, brinda herramientas y funcionalidades para administrar inventarios, precios, descripciones de productos, actualizaciones y otras tareas relacionadas. Esto les permite optimizar sus procesos internos, ahorrar tiempo y recursos, y mantener un mejor control sobre sus productos.

**2. Facilita la venta de productos:** Al permitir que las empresas y organizaciones vendan sus productos a través de la plataforma, ésta les ofrece nuevas oportunidades de negocio para ellas. Muchas empresas pueden no tener los recursos o conocimientos técnicos para desarrollar su propia plataforma de comercio electrónico, por lo que *E&L Designs* les brinda una alternativa fácil de usar. Esto les permite llegar a un público más amplio y aprovechar el creciente mercado del comercio electrónico. Además, al proporcionar una plataforma centralizada, también les ayuda a consolidar y administrar sus ventas en un solo lugar.

**3. Portal de compras para usuarios:** La plataforma web ofrece a los usuarios un portal de compras donde pueden explorar y adquirir productos de diferentes empresas y organizaciones. Esto les brinda una amplia gama de opciones en un solo lugar, lo que facilita su experiencia de compra. Los usuarios pueden realizar búsquedas, filtrar productos según sus preferencias,

comparar precios y leer valoraciones de otros usuarios antes de realizar una compra. Además, al ofrecer la posibilidad de realizar pagos en línea, simplifica el proceso de compra y brinda conveniencia a los usuarios.

**4. Contratación de diseñadores:** Una ventaja adicional de la plataforma es la capacidad de contratar usuarios con perfiles de diseñador para realizar los diseños requeridos por las empresas y organizaciones. Esto ofrece a las empresas acceso a talento creativo y especializado sin tener que buscar diseñadores externos por separado. Los diseñadores contratados pueden trabajar en los diferentes diseños necesarios, como interiores, habitaciones, despachos y más. Esto ayuda a las empresas a obtener diseños personalizados y de calidad para sus productos, lo que puede marcar la diferencia en términos de atractivo y posicionamiento en el mercado.

Respecto a las limitaciones de E&L Designs, podemos destacar:

**1. Competencia en el mercado:** El mercado de plataformas y servicios web para la gestión de productos y el comercio electrónico es altamente competitivo. Existen numerosas alternativas disponibles, incluidas grandes plataformas establecidas y otros startups emergentes. Para destacar y atraer a empresas y usuarios, necesitamos diferenciarnos de la competencia. Esto implica ofrecer características únicas, brindar un valor agregado, enfocándonos en un punto específico o proporcionar una experiencia de usuario excepcional.

**2. Necesidad de adopción por parte de las empresas:** Convencer a las empresas y organizaciones de que adopten y utilicen nuestra plataforma como su sistema de gestión de productos puede ser un desafío. Algunas empresas pueden estar acostumbradas a sus métodos existentes o pueden tener preocupaciones sobre la transición a una nueva plataforma. Deberíamos de ser capaces de demostrarles los beneficios y las ventajas de utilizar nuestro servicio web, como la eficiencia, la facilidad de uso, la capacidad de llegar a nuevos clientes y el potencial de aumentar sus ventas.

**3. Seguridad y confianza:** La seguridad es una preocupación crítica cuando se trata de transacciones financieras en línea y el manejo de datos de los usuarios. Deberíamos asegurarnos de implementar medidas de seguridad sólidas para proteger la información financiera y personal de los usuarios. Esto incluye la implementación de encriptación de datos, el cumplimiento de los estándares de seguridad establecidos, como el cumplimiento del PCI DSS (*Payment Card Industry Data Security Standard*), y la selección de proveedores confiables de servicios de pago en línea.

**4. Experiencia de usuario:** La experiencia del usuario desempeña un papel crucial en la aceptación y adopción del servicio web. Es importante garantizar que la navegación sea intuitiva y fácil de usar para los usuarios. La plataforma debe ser accesible desde diferentes dispositivos y navegadores, y debe ofrecer una interfaz atractiva y amigable. Además, la plataforma debe ser rápida y receptiva, evitando retrasos o tiempos de carga prolongados que puedan frustrar a los usuarios. La atención al detalle en el diseño de la interfaz de usuario y la experiencia de navegación ayudará a mejorar la satisfacción de los usuarios y a fomentar su lealtad hacia nuestra plataforma.

En resumen, E&L Designs ofrece una solución completa para la gestión de productos y el comercio electrónico, facilitando tanto a las empresas como a los usuarios la realización de transacciones y la compra de productos.

### 7.3 Trabajos futuros

Se han definido una serie de futuras integraciones y/o mejoras para nuestra plataforma web. A continuación, explicaremos detalladamente cinco posibles trabajos futuros que podríamos realizar en E&L Designs para seguir mejorando nuestro sistema:

**1. Integrar sistemas de pago y seguridad:** En nuestro servicio web, es fundamental garantizar transacciones seguras para proteger la información financiera de los usuarios. Para lograrlo, deberíamos investigar y seleccionar proveedores confiables de servicios de pago en línea, como PayPal o Stripe, que cuenten con una sólida reputación en términos de seguridad. Estos proveedores ofrecen soluciones de pago encriptadas que protegen los datos sensibles durante las transacciones.

Además, deberíamos asegurarnos de cumplir con los estándares de seguridad establecidos para la protección de datos, como el cumplimiento del Reglamento General de Protección de Datos (GDPR). Esto implica implementar medidas de encriptación, autenticación y acceso seguro a la información sensible.

**2. Realizar pruebas y depuración:** A medida que desarrollamos nuestro servicio web, es esencial llevar a cabo pruebas rigurosas para garantizar su estabilidad y rendimiento óptimo. Las pruebas de funcionalidad son necesarias para verificar que todas las características de la plataforma funcionen como se espera. Esto implica probar diferentes escenarios y casos de uso para identificar posibles errores y corregirlos. Las pruebas de rendimiento son cruciales para evaluar cómo responde nuestra plataforma ante cargas de usuarios y transacciones simultáneas. Deberíamos simular situaciones de alta demanda y monitorear el rendimiento del sistema, asegurándonos de que pueda manejar la carga de trabajo sin problemas. Además, las pruebas de seguridad son necesarias para identificar vulnerabilidades y proteger la información de los usuarios. Estas pruebas incluyen el análisis de posibles puntos de acceso no autorizados y la implementación de medidas de seguridad adicionales.

**3. Implementar mejoras y retroalimentación:** Una vez que el servicio web esté en funcionamiento, es importante recopilar los comentarios y sugerencias de los usuarios para mejorar continuamente la plataforma. Podríamos utilizar encuestas, formularios de comentarios y análisis de métricas para recopilar esta información. Utilizar la retroalimentación para implementar mejoras en la plataforma. Esto puede implicar optimizar la interfaz de usuario para que sea más intuitiva y fácil de usar, agregar nuevas características solicitadas por los usuarios o corregir cualquier problema o error identificado. Es decir, deberíamos mantener una comunicación abierta con los usuarios a través de un sistema de soporte o chat interno para resolver sus inquietudes y necesidades.

**4. Implementar un sistema de análisis de datos:** Para mejorar la toma de decisiones y optimizar la experiencia de los usuarios, podríamos incorporar un sistema de análisis de datos en nuestro

servicio web. Esto nos permitiría recopilar y analizar información relevante sobre el comportamiento de los usuarios, las preferencias de compra y otros datos relacionados. Al implementar un sistema de análisis, podríamos obtener información valiosa, como el número de visitas, el tiempo de permanencia en la plataforma, los productos más populares, las conversiones de ventas y las tasas de abandono del carrito de compras. Esta información nos ayudaría a comprender mejor el rendimiento de nuestro servicio web y a identificar oportunidades de mejora. Podríamos utilizar herramientas de análisis web, como *Google Analytics*, para recopilar datos y generar informes detallados. Estos informes permitirían tomar decisiones sobre cómo optimizar la navegación, la presentación de productos, las estrategias de marketing y otras áreas clave de E&L Designs.

**5. Incorporar un sistema de atención al cliente:** Para brindar un servicio completo y satisfactorio, podríamos considerar la implementación de un sistema de atención al cliente en nuestra plataforma web. Esto permitiría a los usuarios y a las empresas/organizaciones obtener soporte y resolver sus dudas o problemas de manera eficiente. Podríamos ofrecer un chat en vivo o un sistema de *tickets* de soporte, donde los usuarios puedan comunicarse directamente con el equipo de atención al cliente. Además, podríamos establecer una base de conocimientos o sección de preguntas frecuentes (FAQ) donde los usuarios puedan encontrar respuestas a las preguntas comunes. Esto ayudaría a reducir la carga de consultas repetitivas y permitiría a los usuarios resolver problemas por sí mismos de manera rápida.



## 8. Glosario

**Amazon:** Una de las mayores empresas de comercio electrónico y servicios en línea a nivel mundial, fundada por Jeff Bezos en 1994. Amazon ofrece una amplia gama de productos y servicios, incluyendo ventas minoristas en línea, servicios de computación en la nube y contenido de transmisión de video.

**Facebook:** Una plataforma de redes sociales fundada por Mark Zuckerberg en 2004. Facebook permite a los usuarios conectarse con amigos y familiares, compartir contenido, publicar actualizaciones y participar en grupos y eventos.

**Wikipedia:** Una enciclopedia en línea gratuita y colaborativa que permite a los usuarios crear y editar artículos. Wikipedia abarca una amplia variedad de temas y proporciona información detallada sobre personas, lugares, eventos históricos, conceptos, entre otros.

**Netflix:** Un servicio de transmisión de contenido en línea que ofrece una amplia variedad de películas, series de televisión y documentales. Los usuarios pueden ver el contenido en sus dispositivos conectados a Internet, como televisores, computadoras y teléfonos inteligentes.

**Era Digital:** Se refiere al período en el que la tecnología digital, como Internet y las redes sociales, ha tenido un impacto significativo en la sociedad y ha transformado la forma en que trabajamos, aprendemos e interactuamos.

**Internet:** Una red global de computadoras interconectadas que permite la comunicación y el intercambio de información a nivel mundial.

**Redes sociales:** Plataformas en línea que permiten a las personas crear perfiles, conectarse con otros usuarios y compartir contenido, ideas e intereses comunes.

**Democratización:** El proceso de hacer que algo esté disponible y accesible para un mayor número de personas. En este contexto, se refiere al acceso generalizado a Internet y las diversas formas de comunicación que se han vuelto más accesibles para la sociedad en general.

**URL:** Uniform Resource Locator (Localizador Uniforme de Recursos) es la dirección que se utiliza para acceder a un recurso en Internet, como una página web específica. Una URL generalmente comienza con "http://" o "https://" seguido de un dominio y una ruta.

**Navegador web:** Una aplicación o programa informático que permite a los usuarios acceder y visualizar páginas web en Internet. Algunos navegadores populares incluyen Google Chrome, Mozilla Firefox, Safari y Microsoft Edge.

**Páginas web:** Documentos electrónicos que se muestran en un navegador web y contienen información, imágenes, enlaces y otros elementos multimedia. Las páginas web son creadas y mantenidas por individuos, organizaciones y empresas para comunicar, promocionar y compartir información en línea.

**World Wide Web:** También conocida como la Web, es un sistema de información en línea que permite acceder y explorar páginas web a través de Internet. Fue desarrollada por Tim Berners-Lee en 1991 y se basa en el uso de hipervínculos para conectar y navegar entre diferentes páginas web.

**Revolución Industrial:** Un período histórico caracterizado por cambios significativos en la producción industrial, la tecnología y la organización laboral. La primera revolución industrial ocurrió en el siglo XVIII con la introducción de la maquinaria y la mecanización. La "cuarta revolución industrial" se refiere a la integración de tecnologías digitales avanzadas, como la inteligencia artificial, la robótica y el Internet de las cosas, en los procesos de producción y la vida cotidiana.

**Tecnología:** El conjunto de conocimientos, herramientas, procesos y sistemas utilizados para crear bienes o servicios, así como para facilitar diversas actividades humanas.

**World Wide Web inactiva:** Se refiere a las páginas web que ya no están en funcionamiento o que no han sido actualizadas durante un período prolongado de tiempo. Estas páginas pueden estar fuera de línea, desactualizadas o eliminadas permanentemente.

**Usuarios:** Personas que utilizan y acceden a Internet y sus servicios, como páginas web, aplicaciones y plataformas en línea.

**Servicio web:** Una aplicación o plataforma en línea que ofrece funciones o recursos a los usuarios a través de Internet.

**Empresas y/u organizaciones:** Entidades comerciales o instituciones que pueden utilizar el servicio web para gestionar y vender sus productos.

**Portal de compras:** Una sección o área dentro del servicio web donde los usuarios pueden buscar, filtrar y comprar productos ofrecidos por las diferentes empresas u organizaciones.

**Pago online:** La opción de realizar transacciones monetarias a través de Internet utilizando métodos de pago electrónicos.

**Valoraciones:** Opiniones o calificaciones que los usuarios pueden dejar sobre los productos adquiridos, brindando retroalimentación y recomendaciones a otros usuarios.

**Chat interno:** Un sistema de mensajería dentro del servicio web que permite a los usuarios comunicarse entre sí, en este caso, para obtener asistencia o soporte.

**Plataforma 3D:** Una herramienta o entorno virtual que permite a los usuarios con perfiles de diseño crear bocetos y diseños en tres dimensiones.

**CRUD:** Acrónimo de Create, Read, Update y Delete (Crear, Leer, Actualizar y Borrar), se refiere a las operaciones básicas de gestión de datos en una base de datos o sistema, que en este caso se aplicarían a la gestión de las diferentes entidades y productos dentro del servicio web.

**Roles:** Funciones o categorías asignadas a los usuarios dentro del servicio web, como usuarios normales, diseñadores, administradores, etc.

**Competencias:** Habilidades, conocimientos y capacidades que se adquieren y desarrollan a través de la educación y la experiencia.

**Metodología de trabajo:** Un enfoque o conjunto de procesos y prácticas utilizados para planificar, organizar y ejecutar un proyecto.

**Procedimiento de trabajo:** Las acciones y pasos específicos que se siguen dentro de una metodología para completar un proyecto o tarea.

**Estimaciones de las entregas:** La predicción o cálculo del tiempo necesario para completar y entregar las diferentes partes o fases del proyecto.

**Feedback:** Retroalimentación o comentarios recibidos sobre el progreso o resultado del trabajo realizado.

**Metodología Agile:** Un enfoque de gestión de proyectos que se centra en la adaptabilidad, la colaboración y la entrega iterativa y rápida de resultados.

**Ciclos de producción:** Divisiones o etapas del proyecto donde se realizan actividades específicas y se obtienen resultados parciales.

**Test-Driven Development (TDD):** Metodología de desarrollo de software que se basa en escribir pruebas antes de escribir el código, para garantizar un enfoque más orientado a la calidad y la prueba del software.

**Despliegue automatizado:** Proceso de implementación y distribución de la aplicación de forma automática, generalmente utilizando herramientas y scripts.

**Jenkins:** Una herramienta de automatización de código abierto utilizada para facilitar y controlar los procesos de construcción, prueba y despliegue de software.

**Entorno productivo:** El entorno en el que se ejecuta y utiliza la aplicación de forma activa por los usuarios finales.

**JetBrains YouTrack:** Una herramienta de gestión de proyectos y seguimiento de problemas que ofrece una variante Kanban dentro de la metodología Agile.

**Kanban:** Un enfoque visual y de flujo de trabajo para la gestión de proyectos que utiliza tarjetas y columnas para representar las tareas y su progreso.

**Bootstrap:** Un framework de diseño web front-end que proporciona componentes y estilos predefinidos para facilitar el desarrollo rápido y responsivo de sitios web.

**JavaScript, jQuery, Ajax:** Tecnologías utilizadas para agregar interactividad y dinamismo a las páginas web, realizar solicitudes asíncronas al servidor y manipular el contenido dinámicamente.

**ThreeJS:** Una biblioteca de JavaScript utilizada para crear y renderizar gráficos y animaciones en 3D en el navegador web.

**PHP:** Un lenguaje de programación utilizado principalmente para el desarrollo de aplicaciones web del lado del servidor.

**Framework Slim (PHP):** Un framework ligero y minimalista de PHP utilizado para construir aplicaciones web y servicios API de manera eficiente.

**Python:** Un lenguaje de programación versátil utilizado en diversos campos, incluido el desarrollo web y la automatización de tareas.

**MySQL:** Un sistema de gestión de bases de datos relacional utilizado para almacenar y administrar datos.

**Requerimientos:** Son las especificaciones y características que debe cumplir el proyecto para satisfacer las necesidades del cliente o usuario.

**Análisis y Diseño:** Fase del proyecto en la que se realiza un estudio detallado de los requerimientos, se analizan las posibles soluciones y se diseñan las especificaciones técnicas y funcionales del proyecto.

**Instalación y pruebas de funcionamiento:** Consiste en la configuración e implementación de los software y herramientas necesarias para el proyecto, así como en la realización de pruebas para comprobar su correcto funcionamiento.

**Desarrollo y despliegue de la aplicación:** Fase en la que se lleva a cabo la programación y construcción del proyecto, así como su posterior implementación y puesta en funcionamiento.

**Memoria final:** Es un documento en el que se describe de manera detallada todo el proceso seguido en el proyecto, desde la planificación hasta la implementación, incluyendo los resultados obtenidos y las conclusiones.

**Diagrama de Gantt:** Es una herramienta visual utilizada para planificar y programar tareas en un proyecto. Muestra las fechas de inicio y finalización de cada tarea y la dependencia entre ellas.

**Tablero Kanban:** Es un tablero visual que se utiliza para gestionar tareas y su flujo de trabajo. Permite visualizar el progreso de las tareas, identificar cuellos de botella y asignar prioridades.

**Desarrollo web:** El proceso de crear aplicaciones, sitios web y recursos en la World Wide Web, que implica diseño visual, programación y gestión de bases de datos.

**Arquitectura cliente-servidor:** Modelo utilizado en el desarrollo web donde el cliente (software o dispositivo) solicita y consume información, y el servidor (software o hardware) proporciona los recursos solicitados.

**Front-end:** Se refiere a la interfaz y la interacción del usuario en una aplicación o sitio web, utilizando tecnologías como HTML, CSS y JavaScript.

**Back-end:** Se ocupa de la lógica y la gestión de datos en el servidor, utilizando lenguajes de programación del lado del servidor y bases de datos.

**Bases de datos:** Componentes esenciales para almacenar, gestionar y acceder a grandes volúmenes de información. Algunas bases de datos comunes son MySQL, PostgreSQL y MongoDB.

**Hosting:** Servicio de alojamiento web donde se almacenan los archivos y recursos de un sitio web para que esté accesible en Internet.

**Dominio:** Nombre único utilizado para identificar y acceder a un sitio web en Internet, como "elohimcabezas.com".

**Desarrollo orientado a dispositivos móviles:** Enfoque en el diseño y desarrollo de sitios web que se adaptan y funcionan de manera óptima en dispositivos móviles, teniendo en cuenta las diferentes pantallas y experiencias de usuario.

**Aplicaciones Web Progresivas (PWA):** Sitios web que ofrecen funcionalidades similares a las aplicaciones nativas, como acceso offline, notificaciones push y acceso al hardware del dispositivo. Proporcionan una experiencia de usuario fluida sin la necesidad de crear aplicaciones nativas para cada plataforma.

**Single-Page Applications (SPA):** Aplicaciones web que cargan una sola página HTML y actualizan el contenido dinámicamente a medida que los usuarios interactúan, evitando cargar páginas adicionales. Esto mejora la velocidad y la experiencia del usuario al eliminar los tiempos de espera entre transiciones de página.

**Progressive Web Apps (PWA):** Aplicaciones web que utilizan tecnologías modernas para brindar una experiencia similar a la de una aplicación nativa. Pueden instalarse en el dispositivo del usuario, funcionar offline y enviar notificaciones push.

**API RESTful:** Interfaz de programación de aplicaciones (API) que sigue los principios del estilo arquitectónico REST. Permite la comunicación entre diferentes sistemas y aplicaciones a través de solicitudes HTTP, utilizando métodos estándar como GET, POST, PUT y DELETE.

**Microservicios:** Un enfoque arquitectónico para el desarrollo de aplicaciones en el que la funcionalidad se divide en pequeñas unidades de servicio independientes y escalables. Los microservicios se comunican entre sí a través de API y pueden implementarse, actualizarse y escalar de forma individual.

**Inteligencia Artificial (IA) y Machine Learning (ML):** Tecnologías utilizadas para mejorar la personalización, la recomendación de contenido y la automatización de tareas en el desarrollo web. Los chatbots y los asistentes virtuales basados en IA también se utilizan para interactuar con los usuarios en los sitios web.

**Realidad Virtual (VR) y Realidad Aumentada (AR):** Tecnologías que se han incorporado al desarrollo web para crear experiencias inmersivas. La realidad virtual permite a los usuarios sumergirse en entornos virtuales, mientras que la realidad aumentada integra elementos virtuales en el entorno real, enriqueciendo la experiencia del usuario.

**Interfaz de usuario (UI):** La forma en que los usuarios interactúan con un sitio web o una aplicación. Incluye elementos visuales, diseño, navegación y la experiencia general de uso.

**Geolocalización:** Utilización de la información de la ubicación del dispositivo del usuario para brindar servicios personalizados o relevantes basados en la localización geográfica.

**Aplicación nativa:** Aplicación desarrollada específicamente para una plataforma o sistema operativo particular, como iOS o Android. Se instala directamente en el dispositivo y puede acceder a todas las funcionalidades y recursos del dispositivo.

**API (Interfaz de programación de aplicaciones):** Conjunto de reglas y protocolos que permiten a diferentes aplicaciones comunicarse entre sí. Las API permiten la integración de diferentes sistemas y la transferencia de datos y funcionalidades entre ellos.

**WebSocket:** Protocolo de comunicación bidireccional que permite una comunicación en tiempo real entre el navegador y el servidor. WebSocket es utilizado en aplicaciones web que requieren actualizaciones instantáneas y dinámicas de datos.

**Pruebas de funcionalidad:** Evaluaciones realizadas para verificar que un sistema o aplicación cumpla con los requisitos funcionales establecidos.

**Pruebas de rendimiento y escalabilidad:** Evaluaciones que se llevan a cabo para medir y asegurar el rendimiento y la capacidad de escalabilidad de un sistema, especialmente bajo condiciones de carga o demanda intensa.

**Pruebas de seguridad:** Evaluaciones realizadas para identificar y mitigar posibles vulnerabilidades en un sistema o aplicación, garantizando la protección de datos y la integridad del sistema.

**Pruebas de usabilidad:** Evaluaciones destinadas a medir la facilidad de uso y la experiencia del usuario al interactuar con un sistema o aplicación.

**Depuración:** Proceso de identificación y corrección de errores o fallos en un sistema o aplicación.

**Seguimiento de errores:** Registro y seguimiento sistemático de los errores encontrados durante el desarrollo y las pruebas de un sistema o aplicación.

**Autenticación:** Proceso de verificación de la identidad de un usuario para garantizar que tenga acceso autorizado a un sistema o aplicación.

**Autorización:** Proceso de otorgar permisos y derechos específicos a los usuarios autenticados, según su rol o nivel de acceso.

**Cifrado:** Técnica que se utiliza para convertir información en un formato ilegible y protegerla de accesos no autorizados, utilizando algoritmos y claves criptográficas.

**Inyección SQL:** Técnica de ataque en la que se insertan comandos SQL maliciosos en las consultas a una base de datos, con el fin de obtener información no autorizada o manipular los datos.

**Cross-site scripting (XSS):** Vulnerabilidad que permite a los atacantes insertar y ejecutar código malicioso en un sitio web legítimo, comprometiendo la seguridad y la privacidad de los usuarios.

**Responsive design:** Enfoque de diseño web que busca que las páginas y aplicaciones se adapten y funcionen correctamente en diferentes dispositivos y tamaños de pantalla.

**Accesibilidad web:** Práctica de diseñar y desarrollar sitios web y aplicaciones de manera que sean accesibles y utilizables por personas con discapacidades o limitaciones.

**Texto alternativo:** Descripción textual que se proporciona como alternativa a una imagen, para que los usuarios con discapacidad visual puedan comprender su contenido.

**Lectores de pantalla:** Software o dispositivos que leen en voz alta el contenido de una pantalla o aplicación, permitiendo a las personas con discapacidad visual acceder a la información.

**Navegación por teclado:** Capacidad de navegar y utilizar una aplicación o sitio web utilizando únicamente el teclado, sin necesidad de un ratón u otro dispositivo de puntero.

**Registro de errores:** Sistema o mecanismo utilizado para capturar y registrar información detallada sobre los errores encontrados en un sistema o aplicación.

**Sistema de seguimiento de errores:** Herramienta o software utilizado para organizar, gestionar y dar seguimiento a los errores reportados, facilitando su resolución y comunicación entre los equipos de desarrollo. Ejemplos populares incluyen Jira y Bugzilla.

## 9. Bibliografía

### 9.1 Libros

#### **Marty, M. (2015). Killer UX Design. SitePoint**

- Este libro se centra en la experiencia del usuario (UX) y puede ayudarnos a comprender cómo diseñar una plataforma web intuitiva y atractiva, brindando a los usuarios una experiencia satisfactoria y fácil de usar.

#### **Freeman, E., Robson, E., & Bates, B. (2004). Head First Design Patterns. O'Reilly Media**

- Este libro introduce una variedad de patrones de diseño comunes, incluido el patrón *Singleton*, y explora cómo aplicarlos en el desarrollo de software.

#### **Fowler, M. (2002). Patterns of Enterprise Application Architecture. Addison-Wesley Professional**

- Este libro presenta una serie de patrones arquitectónicos utilizados en aplicaciones empresariales, incluido el patrón Modelo-Vista-Controlador (MVC). Proporciona una visión general de estos patrones y cómo se pueden aplicar en el desarrollo de software.

### 9.2 Sitos web

#### **SitePoint (<https://www.sitepoint.com>)**

- SitePoint es un recurso en línea que cubre una amplia gama de temas relacionados con el desarrollo web y la programación. Ofrece una gran cantidad de artículos, tutoriales y guías sobre PHP, arquitectura hexagonal, patrón MVC y muchos otros temas relevantes.

#### **DZone (<https://dzone.com>)**

- DZone es una comunidad en línea para desarrolladores que ofrece una amplia variedad de artículos y tutoriales técnicos. Podemos encontrar contenido sobre PHP, arquitectura de software, diseño de aplicaciones y otros temas relacionados.

#### **Laravel News (<https://laravel-news.com>)**

- Laravel News es un sitio web que se centra en el popular framework de PHP llamado Laravel. Publica artículos, tutoriales y noticias sobre Laravel, así como sobre otros temas relacionados con PHP y el desarrollo web en general.

#### **CodeWall (<https://www.codewall.co.uk>)**

- CodeWall es un sitio web que proporciona una variedad de tutoriales y artículos sobre programación y desarrollo web. Podemos encontrar contenido relacionado con PHP, arquitectura hexagonal y otros conceptos de programación.



#### Scotch.io (<https://scotch.io>)

- Scotch.io es una plataforma que ofrece tutoriales, cursos y artículos sobre desarrollo web y programación. Podemos encontrar contenido práctico sobre PHP, patrón MVC, arquitectura de aplicaciones y muchos otros temas relacionados.

#### Documentación de Bootstrap (<https://getbootstrap.com/docs>)

- Bootstrap es un framework de desarrollo front-end ampliamente utilizado. Su documentación oficial proporciona guías detalladas, ejemplos de código y componentes para ayudarnos a construir interfaces web responsivas y modernas.

#### Documentación de jQuery (<https://api.jquery.com>)

- jQuery es una biblioteca de JavaScript que simplifica la manipulación del DOM y el manejo de eventos en páginas web. Su documentación oficial ofrece una referencia completa de los métodos y características de jQuery, así como ejemplos de uso.

#### Documentación de PHP (<https://www.php.net/manual/en/index.php>)

- PHP es un lenguaje de programación ampliamente utilizado para el desarrollo web del lado del servidor. El sitio web oficial de PHP proporciona una documentación exhaustiva que abarca desde la sintaxis básica hasta las funciones avanzadas y la programación orientada a objetos.

#### Documentación de MySQL (<https://dev.mysql.com/doc>)

- MySQL es un sistema de gestión de bases de datos relacional ampliamente utilizado. Su documentación oficial incluye guías, referencias y ejemplos para ayudarnos a trabajar con MySQL, desde la instalación hasta la administración y consultas de bases de datos.

### 9.3 Artículos consultados

#### "A Complete Guide to Flexbox" - CSS-Tricks"

URL: <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

#### "JavaScript Promises: An Introduction"

URL: [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Using\\_promises](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Using_promises)

#### "A Complete Guide to CSS Grid"

URL: <https://css-tricks.com/snippets/css/complete-guide-grid/>

#### "Introduction to Responsive Web Design"

URL: [https://www.w3schools.com/css/css\\_rwd\\_intro.asp](https://www.w3schools.com/css/css_rwd_intro.asp)

#### "Cross-Site Scripting (XSS) Explained"

URL: <https://owasp.org/www-community/attacks/xss/>

**"Introduction to Cross-Site Scripting (XSS)"**

URL: <https://portswigger.net/web-security/cross-site-scripting>

**"UML Use Case Diagrams: Guidelines"**

URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-use-case-diagram/>

**"UML Sequence Diagram Tutorial"**

URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-sequence-diagram-tutorial/>

**"WebSockets Introduction"**

URL: [https://developer.mozilla.org/en-US/docs/Web/API/WebSockets\\_API](https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API)

**"Introduction to WebSockets in PHP"**

URL: <https://medium.com/@cn007b/super-simple-php-websocket-example-ea2cd5893575>

**"Three.js - JavaScript 3D library"**

URL: <https://threejs.org/>