



universidad  
de león



# Escuela de Ingenierías Industrial, Informática y Aeroespacial

## GRADO EN INGENIERÍA ELÉCTRICA

Trabajo de Fin de Grado

Plataforma online para la creación sencilla de páginas  
web o blogs

Online platform for the simple creation of websites or  
blogs

Autor: Jose María Arias Rodríguez  
Tutor: Jose Alberto Benítez Andrades  
Cotutor: Martín Bayón Gutiérrez



(Julio, 2023)

**UNIVERSIDAD DE LEÓN**  
**Escuela de Ingenierías Industrial, Informática y**  
**Aeroespacial**

**GRADO EN INGENIERÍA ELÉCTRICA**  
**Trabajo de Fin de Grado**

**ALUMNO:** Jose María Arias Rodríguez

**TUTOR:** Jose Alberto Benítez Andrades

**COTUTOR:** Martín Bayón Gutiérrez

**TÍTULO:** Plataforma online para la creación de páginas web o blogs.

**TITLE:** Online platform for the creation of websites or blogs.

**CONVOCATORIA:** Julio, 2023

**RESUMEN:**

El proyecto surge con una visión clara: proporcionar a personas mayores o poco familiarizadas con el mundo de la tecnología, que tienen un comercio físico y desean aumentar el rango de clientes realizando ventas de forma online, una solución accesible y sencilla. Para ello la plataforma online les ofrecerá la opción de crear blogs o páginas webs de una forma totalmente guiada y rápida, y en la que si lo requieren podría contactar con un servicio de ayuda que les dará asistencia técnica. En el proceso de creación, los usuarios encontrarán una interfaz intuitiva que les permitirá seleccionar los datos principales, como el nombre del blog o el dominio donde se alojará la página web si estos así lo requieren. Además, tendrán la opción de elegir entre diversos diseños y plantillas prediseñadas que se adaptarán a sus necesidades. En las tiendas online terminaría con la inserción de los artículos en venta y en el blog con la escritura del primer relato. Durante la creación de estos, tendrán pequeños cuadros de información con ayudas y todo tipo de explicaciones para que tengan conocimientos de sus actos en todo, ya que, en muchos casos, los tipos de personas a los que se orienta la página web desconocen la terminología o conocimientos técnicos relacionados con el mundo de Internet. Una vez finalicen el proceso de creación, dispondrán de un área de usuario propia, donde podrán realizar cambios en todas las tiendas online de las que dispongan o blogs. Además, será aquí donde podrán descargarse los ficheros necesarios para poder ejecutar sus blogs y tiendas online creada, y así elegir qué hacer con estos, donde instalarlos, instalarlos en un dominio propio, en una propia página web donde desean añadir secciones, etc. Para la creación de este proyecto se usa una base de datos propia, donde se guardan y almacenan todos los usuarios de nuestros clientes y todas sus creaciones a las que tendrán acceso en cualquier instante para trabajar con ellas.

**ABSTRACT:**

The project has a clear vision: to provide an affordable and simple solution for seniors or people new in the world of technology who have a physical store and want to expand their reach of customers selling online. Our online platform offers the option to develop blogs or websites in a guided and fast way. In the case they need help with some process, they can contact with help service to obtain technical assistance. On the



creation process, users will find an intuitive interface which allow them to select main data, such as the name of the blog or website; or, if these require it, the domain on which they will be hosted. Furthermore, users have the option to choose between several designs and predetermined templates that suit their needs. Web pages end with the insertion of the articles for sale, and blogs with the writing of the first story. While creating their website, users will be able to interact with information tables and explanations to help them to understand what they need to do. People targeted by the website don't know many terminologies or technical knowledge of the internet world. Once they finish the creation process, they will have their own user area, where they can make changes in all websites or blogs. In addition, users will find here the necessary files to run their blogs and web pages created, and so choose what to do with them, where to put them, how to put them in a domain, on their own website where they want to add sections, etc. For the creation of this project, we use our own database, where all our customers' users and all their creations are stored and accessed at any time to work with them.

**Palabras clave:** HTML, JavaScript, MySQL, React, CSS

**Firma del alumno:**

**VºBº Tutor/es:**



# Índice de contenidos

Índice de figuras .....	6
Índice de tablas.....	8
Glosario.....	9
Resumen .....	10
Abstract .....	11
1 Introducción .....	12
1.1 ORIGEN.....	12
1.2 OBJETIVOS.....	12
1.3 TECNOLOGÍAS .....	13
1.4 PLANIFICACIÓN .....	14
2 Estudio de mercado .....	17
2.1 PÁGINAS WEB .....	17
2.2 BLOGS.....	18
2.3 CONCLUSIONES.....	19
2.4 CARACTERÍSTICAS IMPRESCINDIBLES .....	20
3 Tecnologías.....	21
4 Presupuesto.....	30
5 Cronología .....	35
6 Desarrollo del código .....	37
6.1 ESTRUCTURA Y EJEMPLOS DE IMPLEMENTACIÓN .....	37
6.2 METODOLOGÍA DE DESARROLLO .....	47
6.3 PATRONES DE DISEÑO .....	47
7 Base de datos .....	49
8 La plataforma .....	52
8.1 PÁGINA PRINCIPAL.....	52
8.2 CREACIÓN DE BLOGS Y PÁGINAS WEB .....	54
8.3 INTERFAZ DE USUARIO .....	58
9 Privacidad.....	60
10 Planes a futuro.....	62



11	Conclusiones.....	65
12	Bibliografía.....	66



# Índice de figuras

FIGURA 3.1. LOGO DE GOLAND [10] .....	21
FIGURA 3.2. LOGO DE HTML5 [11] .....	22
FIGURA 3.3. LOGO DE CSS [12] .....	23
FIGURA 3.4. LOGO DE JAVASCRIPT [13] .....	24
FIGURA 3.5. FICHERO XML [14] .....	24
FIGURA 3.6. LOGO DE REACT [15] .....	25
FIGURA 3.7. LOGO DE TAILWIND CSS [16] .....	25
FIGURA 3.8. LOGO DE NODE.JS [17] .....	26
FIGURA 3.9. LOGO DE EXPRESSJS [18] .....	27
FIGURA 3.10. LOGO DE MYSQL [19] .....	28
FIGURA 3.11. LOGO DE GITHUB [20] .....	28
FIGURA 5.2. ÚLTIMOS AVANCES EN EL PROYECTO .....	35
FIGURA 5.1. PRIMEROS AVANCES EN EL PROYECTO .....	36
FIGURA 6.1. ORGANIZACIÓN DEL PROYECTO .....	37
FIGURA 6.2. CÓDIGO DEL COMPONENTE VINETA.JS .....	38
FIGURA 6.3. VISUALIZACIÓN DEL COMPONENTE EN LA PLATAFORMA .....	38
FIGURA 6.4. LÍNEAS DE CÓDIGO NECESARIAS PARA RENDERIZAR EL CÓDIGO PASADO POR LA APP .....	39
FIGURA 6.5. COMPONENTES ROUTE QUE CREAN LA COMUNICACIÓN EN LA PLATAFORMA ONLINE .....	39
FIGURA 6.6. DECLARACIÓN Y CREACIÓN DE USUARIOS EN EL CONTEXTO GENERAL .....	40
FIGURA 6.7. FUNCIONES Y OBJETOS A LOS QUE DA ACCESO EL CONTEXTO GENERAL .....	41
FIGURA 6.8. EJEMPLO DE DECLARACIÓN DE USESTATE .....	41
FIGURA 6.9. EJEMPLO DE USO DE USEEFFECT .....	41
FIGURA 6.10. EJEMPLO DE CREACIÓN DE CONTEXTO .....	42
FIGURA 6.11. EJEMPLO DE USO DEL COMPONENTE 'PROVIDER' .....	42
FIGURA 6.12. EJEMPLO DE DECLARACIÓN DE UNA CONSTANTE QUE ACCEDE AL CONTEXTO .....	42
FIGURA 6.13. EJEMPLO DE USO DE TAILWIND .....	43
FIGURA 6.14. EJEMPLO DE IMPORTACIÓN DE ICONOS .....	44
FIGURA 6.15. DECLARACIÓN DE DEPENDENCIAS Y CONSTANTES .....	44
FIGURA 6.16. MÉTODO GET PARA RECUPERAR USUARIOS DE LA BASE DE DATOS. ....	45



FIGURA 6.17. MÉTODO POST PARA INSERTAR UN USUARIO EN LA BASE DE DATOS.....	45
FIGURA 6.18. FUNCIÓN USADA PARA RECUPERAR LOS USUARIOS DE LA BASE DE DATOS. ....	46
FIGURA 6.19. FUNCIÓN ENCARGA DE INSERTAR UN USUARIO EN LA BASE DE DATOS. ....	46
FIGURA 7.1. DIAGRAMA ENTIDAD-RELACIÓN DE LA BASE DE DATOS DEL PROYECTO .....	49
FIGURA 9.1.2. SEGUNDA SECCIÓN DE LA PÁGINA PRINCIPAL .....	53
FIGURA 9.1.3. TERCERA SECCIÓN DE LA PÁGINA PRINCIPAL .....	53
FIGURA 9.2.1. PRIMER PASO DE CREACIÓN DE BLOGS .....	54
FIGURA 9.2.3. SEGUNDO PASO DE LA CREACIÓN DE BLOGS .....	55
FIGURA 9.2.4. SEGUNDO PASO (1) EN LA CREACIÓN DE PÁGINAS WEB .....	55
FIGURA 9.2.5. SEGUNDO PASO (2) EN LA CREACIÓN DE PÁGINAS WEB .....	55
FIGURA 9.2.6. TERCER PASO EN LA CREACIÓN DE PÁGINAS WEB .....	56
FIGURA 9.2.7. TERCER Y ÚLTIMO PASO EN LA CREACIÓN DE BLOGS.....	57
FIGURA 9.2.8. CUARTO Y ÚLTIMO PASO EN LA CREACIÓN DE PÁGINAS WEB. ....	57
FIGURA 9.3.1. PRIMERA SECCIÓN: DATOS PERSONALES DEL USUARIO.....	58
FIGURA 9.3.2. SEGUNDA SECCIÓN: BLOGS.....	58
FIGURA 9.3.3. TERCERA SECCIÓN: PÁGINAS WEB .....	59



# Índice de tablas

TABLA 1.4.1 DIAGRAMA DE GANTT.....	15
TABLA 1.4.2. LISTA DE TAREAS.....	16
TABLA 3.1. DEPENDENCIAS USADAS CON SUS CORRESPONDIENTES VERSIONES .....	29
TABLA 4.1. CÁLCULO DE COSTES.....	31
TABLA 4.1. TARIFAS DE PRECIOS DE PLATAFORMAS ONLINE DE HOSTING .....	32
TABLA 4.2. ESTIMACIÓN DE USUARIOS EN PLATAFORMAS ONLINE.....	33
TABLA 4.2. TABLA CON EL PRESUPUESTO FINAL.....	34
TABLA 5.1. DIAGRAMA GANTT.....	35



# Glosario

- SEO (Search Engine Optimization): optimización para motores de búsqueda.
- API (Application Programming Interface): conjunto de protocolos y reglas que permiten la comunicación entre distintos sistemas software.
- Front-end: engloba todo el apartado visual e interactivo que se muestra a todo usuario en una página web o aplicación.
- Back-end: conjunto de elementos encargados de tratar con la información de la aplicación. Es la parte del sistema o aplicación web encargada de procesar y gestionar los datos y la lógica.
- Controlador de versiones: herramientas que permiten gestionar y controlar los cambios realizados en un proyecto o un conjunto de archivos.
- Interfaz de usuario: medio visual de una aplicación o página web, a través del cual un usuario interactúa con ellos.
- Framework: conjunto de herramientas, bibliotecas o pautas de desarrollo que proporciona facilidades para la creación de proyectos informáticos.
- IDE (Integrated Development Environment): *software* o aplicación que combina varias herramientas en una interfaz para facilitar el desarrollo *software*.
- Add-ons: componentes *software* adicionales que se agregan a páginas web para ampliar sus funciones.



# Resumen

El proyecto surge con una visión clara: proporcionar a personas mayores o poco familiarizadas con el mundo de la tecnología, que tienen un comercio físico y desean aumentar el rango de clientes realizando ventas de forma online, una solución accesible y sencilla. Para ello la plataforma online les ofrecerá la opción de crear blogs o páginas webs de una forma totalmente guiada y rápida, y en la que si lo requieren podría contactar con un servicio de ayuda que les dará asistencia técnica. En el proceso de creación, los usuarios encontrarán una interfaz intuitiva que les permitirá seleccionar los datos principales, como el nombre del blog o el dominio donde se alojará la página web si estos así lo requieren. Además, tendrán la opción de elegir entre diversos diseños y plantillas prediseñadas que se adaptarán a sus necesidades. En las páginas web terminaría con la inserción de los artículos en venta y en el blog con la escritura del primer relato. Durante la creación de estos, tendrán pequeños cuadros de información con ayudas y todo tipo de explicaciones para que tengan conocimientos de sus actos en todo, ya que, en muchos casos, los tipos de personas a los que se orienta la página web desconocen la terminología o conocimientos técnicos relacionados con el mundo de Internet. Una vez finalicen el proceso de creación, dispondrán de un área de usuario propia, donde podrán realizar cambios en todas las páginas web de las que dispongan o blogs. Además, será aquí donde podrán descargarse los ficheros necesarios para poder ejecutar sus blogs y páginas web creada, y así elegir qué hacer con estos, donde instalarlos, instalarlos en un dominio propio, en una propia página web donde desean añadir secciones, etc. Para la creación de este proyecto se usa una base de datos propia, donde se guardan y almacenan todos los usuarios de nuestros clientes y todas sus creaciones a las que tendrán acceso en cualquier instante para trabajar con ellas.



# Abstract

The project has a clear vision: to provide an affordable and simple solution for seniors or people new in the world of technology who have a physical store and want to expand their reach of customers selling online. Our online platform offers the option to develop blogs or websites in a guided and fast way. In the case they need help with some process, they can contact with help service to obtain technical assistance. On the creation process, users will find an intuitive interface which allow them to select main data, such as the name of the blog or website, the domain who they will be hosted, if these require it. Furthermore, they have the option to choose between several designs and predetermined templates that suit their needs. Web pages end with the insertion of the articles for sale, and blogs with the writing of the first story. While they are making these, they will have tables with information and explanations to help them to understand what they need to do. People targeted by the website don't know many terminologies or technical knowledge of the internet world. Once they finish the creation process, they will have their own user area, where they can make changes in all websites or blogs. In addition, you will find here the necessary files to run your blogs and web pages created, and so choose what to do with them, where to put them, how to put them in a domain, on your own website where you want to add sections, etc. For the creation of this project, we use our own database, where all our customers' users and all their creations are stored and accessed at any time to work with them.



# 1 Introducción

## 1.1 ORIGEN

El tema del Trabajo de Fin de Grado (TFG) seleccionado consiste en una creación innovadora, una plataforma online de creación de páginas webs y blogs. Tras realizar numerosas búsquedas y consultando variados blogs [1][2] y a personas de mi entorno, se llegó a la conclusión de que muchas de las plataformas existentes para la creación de páginas web, y blogs, resultaban tediosas y presentaban un nivel de dificultad, relacionado con conceptos de informática, superior al que la mayoría de los usuarios tenían. Basado en estos resultados, la plataforma se enfoca en la simplicidad y la accesibilidad, logrando ser una posible solución para todos aquellos que busquen sencillez, para que en lugar de abrumar a los usuarios con una gran cantidad de información en pantalla y múltiples opciones que puedan distraer su atención, se centra en ofrecer una experiencia intuitiva y libre de obstáculos para cualquier tipo de usuario de Internet, sin importar su nivel de conocimientos técnicos respecto a este. De esta forma, se busca captar personas de mayor edad o aquellas que sienten cierto rechazo hacia Internet, debido a que lo perciben como algo complejo o desconocido. Para ello se sacrifican muchas características, se predeterminan muchos procedimientos y se ofrecen diseños predefinidos.

## 1.2 OBJETIVOS

El objetivo principal del proyecto es claro: ofrecer una solución aún más sencilla e intuitiva, de las que podemos encontrar en internet, para aquellos usuarios que buscan sencillez por encima de todo. Para esas personas en concreto se desarrolla la plataforma online, con el fin de que logren ampliar su negocio introduciéndoles en internet, en forma de un comercio online, donde puedan vender todos los productos que ellos deseen; o en forma de blog, para que puedan compartir sus ideas o promocionarse. Se busca que puedan realizar este proceso de una forma sencilla y accesible, sin que sus conocimientos informáticos les pueda resultar un obstáculo. Para que este proyecto funcione, también es necesario que no solo la creación de dichos sitios web sea sencilla, sino que también sea sencillo para los clientes realizar compras o leer blogs. Por eso, es necesario enfatizar



en el estilo simple durante la creación y en las creaciones, para que la plataforma llegue a mayor número de usuarios atraídos por una idea, el poder tener su propio espacio en la web de una forma rápida y sencilla.

Para lograr estos objetivos y respaldar el proyecto, la plataforma cuenta con un amplio número de funcionalidades:

- Durante la creación de blogs como páginas web, los usuarios dispondrán de numerosas ayudas, en forma de cuadros de información o indicaciones para que puedan crear a su gusto ambos sitios web.
- La plataforma ofrece un servicio de asistencia técnica para ayudarles con los obstáculos que pueda surgir.
- La plataforma dispone de una serie de rasgos personalizables para ambos sitios web (blogs o tienda online), desde plantillas predefinidas hasta opciones de diseños innovadores para que puedan hacer suyas sus creaciones, a su gusto.
- Los usuarios pueden crear y guardar varios blogs y páginas web, los cuales pueden ser modificados desde una interfaz destinada a ello.
- Los usuarios pueden descargar sus creaciones para alojarlas y ejecutarlas donde ellos deseen, brindando así flexibilidad a la plataforma online.
- El proyecto cuenta con una base de datos propia donde se almacenan los perfiles de los usuarios junto con sus creaciones.

### 1.3 TECNOLOGÍAS

La plataforma online está basada en su mayor parte en código en *HyperText Markup Language* (HTML), ya que es un lenguaje ampliamente reconocido y utilizado en la estructuración y despliegue de páginas web. Además, para facilitar la creación de diversas interfaces, se usa la librería de React, basada en el concepto de componentes, piezas individuales que se pueden combinar y reutilizar para construir interfaces más complejas. Como complemento se ha usado el *framework* de CSS Tailwind, el cual ha sido fundamental para agilizar el proceso de diseño y estilización de los componentes que forman la plataforma online.



En lo referido al *back-end*, la plataforma online se apoya en una base de datos MySQL, que permite almacenar y administrar de manera eficiente los datos necesarios para el funcionamiento correcto de esta. Para realizar la conexión entre nuestra aplicación y la base de datos, disponemos de la infraestructura para aplicaciones web que ofrece Express, cuya *API* contiene miles de métodos basados en conexiones HTTP útiles para realizar las conexiones con bases de datos. Express se apoya en *Node.js*, entorno que permite crear todo tipo de clases de herramientas, tanto del lado del servidor como en JavaScript, el lenguaje usado para añadir funcionalidad a toda la plataforma. La utilización de *Node.js* ha proporcionado una mayor cohesión entre el *back-end* y el *front-end*, debido al uso de JavaScript como lenguaje para los dos ámbitos. Para realizar las consultas a la base de datos se han usado consultas nativas de SQL, un lenguaje de programación específico para la gestión de bases de datos relacionales. Durante todo el desarrollo del proyecto se ha usado GitHub como controlador de versiones.

La plataforma online ha sido desarrollada utilizando una combinación de tecnologías modernas y punteras en el sector informático, que permiten crear una experiencia accesible e intuitiva para el usuario. Para lograrlo, se buscó aprovechar las mejores ventajas ofrecidas por cada una de las tecnologías mencionadas anteriormente.

## 1.4 PLANIFICACIÓN

El desarrollo de la aplicación se ha basado sobre todo en dos etapas claramente diferenciadas: el desarrollo del *front-end*, con la posterior implementación del *back-end*, con pequeños momentos en los que por razones obvias y necesarias para el correcto desarrollo del apartado visual se ha requerido de algunas implementaciones del apartado de *back-end*. La plataforma se divide en cuatro partes principales: la primera, la página de inicio, donde se puede elegir que acción realizar, crear un blog o una tienda online; continuación, se programaron las ventanas pertinentes para la creación completa de un blog, donde se solicitan datos y se ofrece al usuario las variadas opciones para diseñar el blog a su gusto; seguidamente, las ventanas de creación de la tienda online, donde como para los blogs se solicitarán datos y se escogerá el diseño que más guste al usuario, además de introducir los primeros artículos en su comercio online; y por último, se



desarrolló una interfaz de usuario, donde estos podrán modificar sus datos personales, además de editar sus blogs y páginas web. Una vez desarrolladas las vistas de la plataforma, se implementaron los métodos usados por la plataforma para mantener en todo momento una conexión segura y estable con la base de datos. Primero se realizó la comunicación con esta para que posteriormente se creasen los métodos para añadir, modificar y borrar datos de este desde la plataforma online.

Por último, destinado a ilustrar de forma gráfica la planificación inicial y mejorar su comprensión, se han creado un diagrama de Gantt y una lista de tareas:

<b>Tareas</b>	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8	Semana 9
Página principal – <i>Front-end</i>									
Creación de blogs - <i>Front-end</i>									
Creación de blogs - <i>Back-end</i>									
Creación de webs – <i>Front-end</i>									
Creación de webs – <i>Back-end</i>									
Implementación de la base de datos									
Inicio de Sesión y Registro – <i>Front-end</i>									
Inicio de Sesión y Registro – <i>Back-end</i>									
Interfaz de Usuario – <i>Front-end</i>									
Interfaz de Usuario – <i>Back-end</i>									
Descarga de creaciones									
Creación de la memoria *									

TABLA 1.4.1 DIAGRAMA DE GANTT

\*La memoria se realiza progresivamente con la realización del proyecto.



La lista de tareas se realiza ordenada de forma cronológica, indicando la duración de estas:

Tarea	Fecha de inicio	Duración
Creación del proyecto	4/4/2023	1 día
Creación de la vista de la página principal	5/4/2023	1 semana
Creación de la vista del apartado creación de blogs	12/4/2023	3 días
Creación de la vista del apartado creación de webs	15/4/2023	3 días
Implementación de la base de datos	19/4/2023	1 día
Creación del <i>back-end</i> del apartado de creación de blogs	20/4/2023	3 días
Creación del <i>back-end</i> del apartado de creación de webs	22/4/2023	4 días
Creación del <i>front-end</i> del apartado de Inicio de Sesión y Registro	26/4/2023	3 días
Creación del <i>back-end</i> del apartado de Inicio de Sesión y Registro	28/4/2023	3 días
Creación del <i>front-end</i> del apartado de Interfaz de Usuario	31/4/2023	10 días
Creación del <i>back-end</i> del apartado de Interfaz de Usuario	9/5/2023	8 días
Creación de la funcionalidad de descarga de creaciones	17/5/2023	17 días
Creación de la memoria	4/4/2023	9 semanas

TABLA 1.4.2. LISTA DE TAREAS



## 2 Estudio de mercado

Para la realización de este proyecto se ha desarrollado un pequeño estudio de mercado en el que se valoran todas las posibilidades actuales que se pueden encontrar en Internet, con plataformas digitales como WordPress, Ionos, Shopify, etc, para la creación de páginas web; y otras como HubSpot, Blogger, SquareSpace, etc, para la creación de blogs [3][4]. A continuación, se mostrarán una serie de ventajas e inconvenientes de cada una de las 3 plataformas punteras en cada sector.

### 2.1 PÁGINAS WEB

#### Ionos [5]

- Ventajas:
  - Plantillas predeterminadas que facilitan las creaciones.
  - Ofrecen servicio de hosting de la propia de empresa.
- Inconvenientes:
  - Soporta poco tráfico de forma simultánea.
  - Desventaja de SEO con otras empresas punteras.

#### WordPress [1]

- Ventajas:
  - Cuentan con un plan gratuito
  - Soporte incluido
  - Dispone de miles de *Add-ons* de terceros que aportan funcionalidad.
- Desventajas:
  - Desventaja de SEO con otras empresas punteras, salvo contratación de plan de pago.
  - Editor poco intuitivo.
  - Múltiples funcionalidades de pago.

#### Shopify [6]

- Ventajas:
  - Interfaz intuitiva y accesible
  - Gran variedad de diseños predefinidos con posibilidad de edición.



- Gran soporte
- Desventajas:
  - No existe plan gratuito, además de tener comisiones por transacciones.
  - Precio elevado
  - Problemas de SEO

## 2.2 BLOGS

Blogger: [7]

- Ventajas:
  - Plataforma totalmente gratuita
  - Integrado con GoogleAdSense y GoogleAds para generar ingresos del blog.
- Desventajas:
  - Herramienta de generación de comentarios limitada.
  - Terminología específica o avanzada para determinada audiencia.

SquareSpace: [8]

- Ventajas:
  - Cuenta con una gran variedad de plantillas predefinidas.
  - Sin límite de páginas.
- Desventajas:
  - Baja usabilidad del editor, además de poco intuitivo.
  - Mala experiencia de usuario a la hora de visualizar blogs.

Medium: [9]

- Ventajas:
  - Ofrece gran capacidad de segmentación.
  - Experimentación con un modelo de negocio de pago por contenidos.
- Desventajas:
  - No ofrece dominios personalizados.
  - Baja audiencia y malas críticas.



## 2.3 CONCLUSIONES

Tras la realización de este estudio, se han podido sacar las siguientes conclusiones:

- Grandes similitudes: la mayoría de las páginas web cuentan con un gran número de críticas y opiniones muy similares, tanto en las páginas web alabando sus plantillas predefinidas y las que ofrecen un gran soporte; como las críticas recibidas relacionadas con el precio de los planes que estas ofrecen. En cuanto a los blogs, la gran mayoría coinciden en un mismo asunto, y es el de la visualización de este blog una vez escrito, y la forma de acceder a él.
- Temas económicos: en todas las páginas consultadas se ha mencionado el factor económico, ya bien sea en forma de pago para recibir servicios exclusivos con ellos, para salir en mejores posiciones en las búsquedas de Google, en forma de anuncios que podemos integrar en nuestros blogs o páginas web con los que podremos obtener beneficio. Todo esto son factores a tener muy en cuenta en cuanto a la forma de rentabilizar la plataforma online.
- Gran importancia de las interfaces: en todas las plataformas se podemos encontrar comentarios relacionados con las interfaces, ya bien sea de la propia plataforma, como con la que realizamos las ediciones de nuestras creaciones. Por ello, un punto clave en cualquier sitio web de creación de contenido es contar y ofrecer una experiencia única e intuitiva a los usuarios que vayan a tratar con ellas.
- Soporte: es una de las características a las que menos importancia se ha dado en todas las referencias consultadas durante este estudio, sin embargo, siempre un buen soporte era algo alabado y mencionado como un punto fuerte. Esto se debe a que en muchas ocasiones el público que accede a estas páginas web desconoce un gran número de aspectos y términos que son necesarios para desenvolverse adecuadamente en Internet.
- Plantillas predefinidas: al igual que el soporte, no es uno de los aspectos más importantes en muchas páginas web de creación de contenido, pero es una ayuda a la hora de reducir tiempos, sobre todo para personas con poca creatividad que prefieren diseños predefinidos a los que poder realizar unas pequeñas modificaciones ellos mismo.



Estos resultados arrojan una gran cantidad de información muy importante a la hora de enfocar correctamente el proyecto para que destaque entre todas estas plataformas punteras en la creación de contenido, y de esta forma crear un producto innovador que recoge las mejores virtudes de cada uno y trata de evitar y alejarse de características y comentarios que han generado un gran número de críticas por parte de los usuarios que han interactuado con estos sitios web.

## 2.4 CARACTERÍSTICAS IMPRESCINDIBLES

Por ello, tras realizar el estudio de mercado explorando las diferentes páginas web punteras en cada sector, y analizando los aspectos más valorados; podemos extraer las características más importantes para los usuarios. Tras haber identificado estas características, se desarrolla una lista con las más importantes que deberá cumplir el proyecto, cuyo objetivo es destacar y asentarse entre los mejores sitios web de creación de contenido:

- Interfaz intuitiva: es uno de los factores más importantes para los usuarios, ya que es su forma de relacionarse con cada página web.
- Diseños predefinidos: muy importante para todas las personas que deseen un proceso de creación breve, pero obteniendo un producto final atractivo.
- Planes gratuitos: es una de las principales ventajas mencionadas en muchas páginas web, el disponer de un producto sencillo pero gratuito.
- Pagos opcionales: disponer de diferentes alternativas y opciones que no requieran de pagos por parte del cliente, pero que pueda realizarlos para obtener algún beneficio con ellos.
- Soporte: muy bien valorado al aportar una forma de resolver dudas o cuestiones que surgen a los usuarios.
- Servicios de hosting: con costes bajos o elevados dependiendo de las necesidades de cada usuario.
- Buen posicionamiento SEO: clave para aumentar el alcance de usuarios que acceden a nuestra página web.



## 3 Tecnologías

Para la realización del proyecto se ha hecho uso de una serie de *frameworks* y herramientas punteras en el sector tecnológico para poder realizar el proyecto de programación. Al tratarse de una plataforma online, la mayoría de estas tecnologías se basan en desarrollo de aplicaciones web. Lo primero de todo, es mencionar el IDE donde se ha alojado el proyecto y se ha ejecutado para realizar todas las pruebas durante el desarrollo del código:

- GoLand: es un poderoso entorno de desarrollado integrado (IDE) creado por la empresa JetBrains, y ofrecido de forma gratuita a estudiantes de grado. Está diseñado específicamente para el lenguaje de aplicaciones web. Proporciona a los programadores un gran número de características y herramientas muy útiles durante el proceso de desarrollo de *software*. Este IDE ofrece una interfaz de usuario personalizable que se puede adaptar a las necesidades y preferencias de cada programador, mediante una serie de extensiones; un claro ejemplo de modificaciones que podemos realizar en la interfaz con estas extensiones es: edición automática de código, resaltado de sintaxis, refactorización inteligente, etc. Además, GoLand permite a los desarrolladores cargar *scripts* que permiten ejecutar, compilar, probar y depurar las aplicaciones en un solo clic desde su propia interfaz. Todas estas características explican la elección de GoLand como IDE para desarrollo del proyecto.



FIGURA 3.1. LOGO DE GOLAND [10]

Una vez explicado el IDE de desarrollo usado durante este proyecto, pasamos a exponer las tecnologías y lenguajes usados para la composición total del código:



- HyperText MarkUp Language (HTML): es uno de los lenguajes de nivel mundial más usados para estructurar y crear aplicaciones web. El estándar usado durante el proyecto es el HTML 5, el más actual. Se basa en una estructura de etiquetas o elementos que definen la semántica y estructura del contenido de la página web. Estas etiquetas están delimitadas por símbolos de menor (<) y mayor (>), donde entre estos símbolos escribiremos el tipo de elemento semántico deseado. Según el elemento usado, se tienen una serie de características y acciones, llamadas atributos, que se definen para dar forma a la interfaz y hacerla en parte funcional. El conjunto de todos los elementos y atributos se combinan para que la propia aplicación pueda ser interpretada por el navegador web, encargado de renderizar y presentar el contenido al usuario final. Asimismo, es importante destacar que HTML como tal no permite la programación e interacción dinámica de la página web, para ello existen las tecnologías que veremos seguidamente, como pueden ser CSS (Cascading Style Sheet), para el diseño y el uso de animaciones; y JavaScript, encargado de agregar interactividad y una completa funcionalidad a la página web.



FIGURA 3.2. LOGO DE HTML5 [11]

- Cascade Style Sheet (CSS): lenguaje de programación usado para estilizar los documentos de HTML o XML, en él se define como se deben mostrar los elementos y el diseño de la página web. El estándar usado durante el proyecto es el CSS3, el más actual. Este lenguaje se compone por un selector, del elemento que deseamos editar y añadir estilo; y una declaración, donde añadiremos el estilo y las características a aplicar a dicho elemento. Algunas de las propiedades de edición incluyen el color, tamaño y tipo de fuente, el espaciado de los elementos, los márgenes, los bordes, como el color o el ancho, el color del fondo de un elemento, entre otras. Además de estas propiedades básicas, CSS permite realizar

una selección avanzada de los elementos de la aplicación web, apuntando a un único elemento en concreto, como a un tipo de elemento o a una estructura entera, lo que proporciona un alto nivel de flexibilidad y precisión a aplicar estilos a elementos específicos.



FIGURA 3.3. LOGO DE CSS [12]

- JavaScript (JS): es un lenguaje de programación interpretado y definido como orientado a objetos, multiplataforma, de los más usados en el mundo en el ámbito de las aplicaciones web. Su principal funcionalidad reside en crear interactividad y dinamismo en aplicaciones y aplicaciones web. Es usado tanto en el lado del cliente (navegador web), como en el del servidor (Node.js), dando lugar a aplicaciones completas. Algunas de las características más destacables de este lenguaje de programación pueden ser:
  - Tipado dinámico: no requiere de declaraciones de variables explícitamente. Además, las variables son mutables durante la ejecución del programa.
  - Orientado a objetos: permite la creación de objetos, clases y propiedades, así como la definición de constructores para inicializar objetos.
  - Manipulación del DOM (*Document Object Model*): JS permite acceder y modificar contenido del HTML de la página web, agregando o eliminando eventos, permitiendo la interacción dinámica al usuario con la página web.
  - Bibliotecas y *frameworks*: Como pueden ser el ejemplo de React, Angular o jQuery, las cuales facilitan el desarrollo de aplicaciones web proporcionando componentes y funcionalidades predefinidas.



## JavaScript



FIGURA 3.4. LOGO DE JAVASCRIPT [13]

- Extensible Markup Language: es un lenguaje diseñado con el fin de transmitir y almacenar datos de forma estructurada, el cual no cuenta con una sintaxis preestablecida, si no que permite al programador definir la estructura y el contenido de los datos. Sin embargo, la estructura debe seguir unas pautas, ya que los elementos se deberán organizar en una jerarquía de etiquetas de apertura y de cierre. XML también permite la definición de entidades y referencias a entidades predefinidas o definidas por el usuario, utilizadas para representar caracteres especiales o reservados en XML. Este lenguaje es ampliamente utilizado en los ficheros de configuración de cualquier tipo aplicación, debido a su sencillez y lectura de los datos guardados.



FIGURA 3.5. FICHERO XML [14]

Estos han sido los cuatro lenguajes utilizados para el desarrollo del proyecto, a continuación, se muestran las tecnologías y *frameworks* usados para facilitar y agilizar el trabajo de desarrollo de código:

- React: es una librería complementaria a JavaScript, destinada a la creación de interfaces de usuario con código abierto, es decir, a disposición de cualquier programador para que use sus recursos. La versión implementada en el proyecto es la 18.2.0, la más actual en el momento de creación del proyecto. El principal enfoque es la construcción de componentes reutilizables. React define sus componentes como piezas aisladas de la interfaz de usuario que pueden contener lógica y estado propio. Estos componentes son complementarios entre ellos,

pudiendo dar lugar a interfaces más complejas. React utiliza una sintaxis llamada JSX (JavaScript XML), la cual combina JavaScript con HTML, y es renderizada como JavaScript puro antes de ser interpretado por el navegador. Una de las grandes ventajas que ofrece React es su amplia comunidad y herramientas complementarias, facilitando tareas como enrutamiento, gestión de estado, pruebas, entre muchas otras.

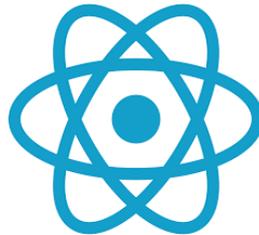


FIGURA 3.6. LOGO DE REACT [15]

- Tailwind CSS: como su nombre indica es un *framework* basado en CCS para acelerar el proceso de diseño de páginas web sin tener que escribir el código fuera del propia HTML. La versión implementada en el proyecto es la 3.3.1, la más actual en el momento de creación del proyecto. A diferencia de otros *frameworks*, como Bootstrap o Bulma, Tailwind suministra una amplia gama de utilidades CSS combinables para construir estilos de una manera mucho más eficiente. Una de las principales características de este *framework* se basa en nomenclaturas descriptivas e intuitivas, para facilitar su uso y comprensión y aumentar la productividad del programador. Además, tailwind ofrece una serie de atributos ya predefinidos como colores, anchuras, bordes, y muchos más que se pueden editar y crear unos propios desde el archivo de configuración del *framework*.



FIGURA 3.7. LOGO DE TAILWIND CSS [16]

- Node.js: es un entorno de ejecución de JavaScript de código abierto y multiplataforma que se utiliza para desarrollar aplicaciones tanto del lado del



cliente como del servidor. Está basado en el motor de ejecución V8 de Google Chrome, y una principal característica que le diferencia de otros entornos basados en hilos, es su enfoque en el modelo E/S (Entidad Relación) basado en eventos y sin bloqueos. Este solo trabaja con un solo hilo para manejar las múltiples solicitudes mediante la implementación de un bucle de Eventos. Además, Node cuenta con un gran conjunto de bibliotecas conocidas como repositorio npm (*Node Package Manager*), lo cual facilita a los desarrolladores instalar, actualizar y compartir fácilmente módulos y dependencias para su propio uso. Node es ampliamente utilizado en el desarrollo de aplicaciones web debido a su popularidad obtenida gracias a su rendimiento, escalabilidad y capacidad para manejar grandes volúmenes de información de forma eficaz. Por último, este entorno permite utilizar JavaScript tanto del lado del cliente como del servidor, simplificando el desarrollo y la transferencia de conocimientos entre estos dos entornos.



FIGURA 3.8. LOGO DE NODE.JS [17]

- **ExpressJS:** es un *framework* rápido y minimalista para aplicaciones web basadas en Node.js. Proporciona una capa de abstracción sobre el servidor HTTP ofrecido por Node.js, facilitando la creación de aplicaciones web y APIs de manera sencilla y eficiente. La versión implementada en el proyecto es la 4.18.2, la más actual en el momento de creación del proyecto. Express destaca por su simplicidad y flexibilidad, permitiendo manejar de forma sencilla rutas, solicitudes y respuestas HTTP, además del procesamiento de parámetros y el envío de datos desde el lado del cliente. Algunas de las principales características de este *framework* son:
  - o Enrutamiento: proporciona un enrutador con el que definir las rutas de una aplicación web de una manera sencilla e intuitiva.



- Gestión de solicitudes y respuestas: simplifica todos los procesos de manipulación de solicitudes y respuestas HTTP, proporcionando métodos y utilidades para acceder a los datos de estas.
- *Middleware*: *express* hace un amplio uso de *middleware*, funciones ejecutadas antes de que una solicitud llegue a su destino final, permitiendo realizar tareas de registro, autenticación o compresión de respuestas entre muchas otras; durante el proceso de estas.



FIGURA 3.9. LOGO DE EXPRESSJS [18]

- MySQL: es un sistema de gestión de bases de datos relacionales más utilizados a nivel mundial para el desarrollo de aplicaciones web y empresariales. Es propiedad de Oracle Corporation. La versión implementada en el proyecto es la 2.18.1, la más actual en el momento de creación del proyecto. Las bases de datos que ofrece son altamente escalables, confiables y compatibles con diferentes lenguajes de programación y plataformas de desarrollo. Algunos de los aspectos básicos de las bases de datos de MySQL son los siguientes:
  - Organización de datos en tablas: estas consisten en filas y columnas. Las columnas tienen un nombre y almacenan tipos de datos específicos en ellas, en las filas se representan los registros individuales de cada tabla de la base de datos.
  - Lenguaje de consulta: utiliza SQL (*Structured Query Language*) para realizar todas las operaciones con la base de datos, ya que este lenguaje proporciona una forma estándar e intuitiva de interactuar con las bases de datos.
  - Integridad de datos: soporta integridad de datos mediante la definición de restricciones, como claves primarias, foráneas y restricciones de integridad referencial. Estas garantizan que todos los datos almacenados en las tablas cumplan con una serie de reglas y restricciones predefinidas por el programador.



Además, también presenta otras características esenciales como son su compatibilidad con transacciones, su seguridad y permitir replicación.



FIGURA 3.10. LOGO DE MYSQL [19]

- GitHub: es una plataforma en línea enfocada en desarrolladores *software* para que puedan alojar, administrar y colaborar en proyectos utilizando un sistema de control de versiones Git. GitHub es uno de los servicios más populares para el desarrollo colaborativo y el control de versiones de código fuente. En GitHub, los desarrolladores pueden crear y manejar distintos repositorios que contienen el código fuente de sus proyectos, estos pueden ser manipulados por varios desarrolladores o simplemente usados por uno solo para almacenar sus proyectos. Existen herramientas con las que se puede visualizar una serie de estadísticas sobre el trabajo realizado por cada integrante del repositorio, así como ver cronológicamente la serie de modificaciones que se han realizado en el repositorio (esta herramienta es usada en la sección de Cronología). Otra de las funcionalidades de GitHub, quizás de las menos conocidas, es la de seguir a otros desarrolladores, explorar otros proyectos y tener la posibilidad de contribuir en aquellos que son de código abierto.



FIGURA 3.11. LOGO DE GITHUB [20]

Estas son las principales tecnologías, lenguajes de programación y librerías que dan una fuerte base a todo el proyecto y que permiten una cómoda y productiva programación. A continuación, se indican las tecnologías implementadas con sus respectivas versiones:



Tecnología	Versión
Axios	1.4.0
Bootstrap-icon	1.10.4
Cors	2.8.5
Express	4.18.2
File saver	2.0.5
Mysql	2.18.1
React	18.2.0
React router dom	6.10.0
React <i>scripts</i>	5.0.1
Validator	13.9.0

*TABLA 3.1. DEPENDENCIAS USADAS CON SUS CORRESPONDIENTES VERSIONES*



## 4 Presupuesto

El éxito de cualquier proyecto depende en gran medida de adoptar una gran planificación financiera que se base en todos los factores, internos y externos, que influyen en nuestro trabajo. En esta sección se expone el presupuesto detallado y la puesta en marcha de la plataforma online. El objetivo buscado con esta sección es mostrar de forma clara los gastos de recursos humanos necesarios, los costes totales del desarrollo del proyecto, además de otros factores. A continuación, se muestra el desglose del presupuesto empleado para la creación y correcto funcionamiento de la plataforma online:

### 1. Recursos humanos:

En este caso, el proyecto ha sido realizado por un único desarrollador *software*. Esto implica que los costes de recursos humanos se calculen en base al número de horas dedicadas y el salario medio de un programador en España.

- Horas dedicadas: Como se ha indicado en la planificación, el proyecto total se ha realizado en 9 semanas, con tiempo de trabajo de 32 horas semanales, darían un resultado de 288 horas.
- Salario medio: En este caso se toma de referencia el salario de un desarrollador Full-stack en España [21], que es de 34.000€ brutos anuales trabajando a jornada completa (40 horas/semana, 160 horas/mes). El coste para una empresa de mantener a un trabajador con ese sueldo es de 44.931€ [22].

El resultado final de costes humanos se calcula con ambos factores. Teniendo que contratar al trabajador 2 meses, supone un coste de 7488,5€.

### 2. Costos de infraestructura:

Durante la etapa de producción, la plataforma ha sido alojada en IONOS, utilizando el pack estándar, lo que supone un gasto de 5€ mensuales. Al realizarse el proyecto en dos meses, la cantidad asciende a 10€. A mayores debemos añadir el gasto del ordenador de desarrollo utilizado, alrededor de 900€.

### 3. Costos de asesoría fiscal:

Con el objetivo de cumplir con la privacidad del usuario, ha sido necesaria asesoría jurídica. El coste asciende a 200€ [23].



En resumen, haciendo cálculos con todos los datos de los que se disponen para hallar el coste total, obtenemos:

Sección	Coste
Personal	7.488,50€
Infraestructura	910,00€
Fiscal	200,00€
<b>COSTE TOTAL:</b>	<b>8.598,50€</b>

*TABLA 4.1. CÁLCULO DE COSTES*

Para obtener beneficio económico, la empresa crea su estrategia basándose en opciones como la publicidad, venta de datos o estrategia de marketing de afiliados; con las que al no depender directamente de pagos de clientes o usuarios logra aumentar su estabilidad financiera, además de crecer directamente con el número de usuarios que simplemente visiten la web. Todos estos planteamientos se verán de forma extendida a continuación, con datos económicos que sostienen el desarrollo del proyecto:

- Programas de afiliados: es una estrategia que está ganando cada vez un mayor número de usuarios en la red que la realizan debido a su gran éxito. Esta estrategia implica asociarse con otras empresas y recibir una comisión por cada cliente referido que realice una compra o realice una acción en concreto en nuestra página web que indirectamente también repercuta en la empresa afiliada. Esta sería una forma de definir en que se basan los programas de afiliados, la forma en la que se aplicaría al proyecto sería en la sección de hosting, donde los clientes podrían elegir entre una serie de dominios disponibles ofrecidos por empresas de hosting, como pueden ser el caso de Ionos, 1&1, Hostinger, y muchas otras. De esta forma, podrían alojar sus páginas web creadas o sus blogs en dominios de una forma sencilla y accesible para todos. En lo referido al tema económico, se ha realizado un análisis de todas las páginas web punteras que ofrecen servicios de hosting compatibles con marketing de afiliados. Se ha agrupado toda la información en una tabla de forma que resulte más visual la comprensión de los datos adquiridos:

Empresa	Precios
IONOS [24]	Packs: <ul style="list-style-type: none"><li>- Standard: 5€/mes</li><li>- Plus: 9€/mes</li><li>- Premium: 14€/mes</li><li>- Ultimate: 25€/mes</li></ul>
Hostinger [25]	Packs: <ul style="list-style-type: none"><li>- Hosting premium: 10,50€ + 6€/mes</li><li>- Hosting empresarial: 16,00€ + 8,00€/mes</li><li>- Cloud startup: 20,00€ + 17,00€/mes</li></ul>
GoDaddy [26]	Packs: <ul style="list-style-type: none"><li>- Hosting web: 2,99€/mes</li><li>- Alojamiento web plus: 16,99€/mes</li></ul>
HostGator [27]	Packs: <ul style="list-style-type: none"><li>- Hatchling plan: 2,75€/mes</li><li>- Baby plan: 3,50€</li><li>- Business plan: 5,25€</li></ul>

TABLA 4.1. TARIFAS DE PRECIOS DE PLATAFORMAS ONLINE DE HOSTING

Después de realizar el análisis de datos, y constatando que de media en España un afiliado recibe de media el 30% del total del ingreso de la página web con la acción realizada por el usuario, y haciendo una media con todos los datos obtenidos (9,62€), obtenemos el número de acciones mínimas que se necesita que realice un cliente para amortiguar el gasto inicial y obtener ingresos en los futuros meses. El número de acciones es de 893 acciones, lo que puede variar dependiendo de la comisión negociada y de los planes o acciones realizadas por los clientes. A continuación, se indica de forma aproximada el número de clientes que tienen



algunas de las plataformas online más famosas del mundo, como apoyo para realizar una estimación del número de usuarios que podrían visitar la plataforma de forma anual:

Plataforma online	Estimación
WordPress [28]	60,8 millones de usuarios/año
Shopify [29]	140 millones de usuarios/año
Blogger [30]	7 millones de usuarios/año
Medium [31]	1 millones de usuarios/año

TABLA 4.2. ESTIMACIÓN DE USUARIOS EN PLATAFORMAS ONLINE

Analizando estos datos, y valorando que el proyecto sería nuevo en el mercado, se puede estimar que el número de usuarios sea un 0,0005% en comparación con estas plataformas, que son las más punteras de Internet. Ese 0,0005% supondría un total de 500 de usuarios que interactúan en la plataforma, lo que supondría unos ingresos de aproximadamente 4810€.

- Google AdSense: esta es una plataforma de publicidad en línea desarrollada y mantenida por Google. Permite a los propietarios de sitios web generar ingresos económicos del tráfico que recibe su página web, a raíz de anuncios dentro de esta. Estos anuncios que se muestran son generados por los anunciantes que utilizan Google Ads, otra plataforma publicitaria de Google. AdSense ofrece la posibilidad a los propietarios de las páginas web de realizar un seguimiento del impacto de los anuncios para optimizar sus ingresos. En lo referido al proyecto, se puede incluir este tipo de anuncios, pero en baja cantidad para que no resulten molestos a los usuarios mientras navegan por la plataforma, pero que ayuda a sumar otra fuente de ingresos al presupuesto. Utilizando [32] e ingresando la región de Europa, Oriente Medio y África junto a la categoría de Informática y electrónica, se obtienen ingresos de 9.063\$ por cada 98.850 visitas en la página al mes, suponiendo un total de 500 visitas, la beneficio final son 5453€. Esto supone un ingreso importante, para amortizar el gasto inicial y para además sumar ingresos los siguientes años, ya que esta cifra subirá progresivamente con el crecimiento de la plataforma online y el aumento de visitas a esta.



Estas dos fuentes de ingresos son suficientes para sacar rentabilidad del proyecto sin tener que añadir más fuentes de ingresos, ya que como podemos ver, en ambas fuentes de ingresos el dinero se genera gracias a los usuarios, pero sin tener que realizar ellos ningún pago. Estas fuentes son escogidas ya que cumplen con la filosofía del proyecto, facilitar la navegación y disminuir el número total de obstáculos a los usuarios, logrando de esta forma una buena impresión y tranquilidad sobre todo tipo de usuarios que accedan a nuestra plataforma. En total, se obtienen ingresos de 10.263€.

<b>Tipo</b>	<b>Costes</b>	<b>Tipo</b>	<b>Ingresos</b>
Personal	7.488,50€	Afiliados	4.810,00€
Infraestructura	910,00€	Google AdSense	5.433,00€
Fiscal	200,00€		
<b>Costes totales:</b>	<b>8.598,50€</b>	<b>Ingresos totales:</b>	<b>10.263,00€</b>
<b>Beneficio total:</b>		<b>1.664,50€</b>	

TABLA 4.2. TABLA CON EL PRESUPUESTO FINAL.



## 5 Cronología

Al usar como controlador de versiones GitHub, se tiene acceso de forma visual a todos los cambios e inserciones que se han realizado a lo largo del proyecto mediante el uso de GitKraken, una herramienta de Git que permite analizar y ver datos actualizados de los repositorios. De esta forma se pueden mostrar los avances según el paso del tiempo y como se ha ido modificando progresivamente y de forma estructurada el proyecto. A continuación, se adjunta un diagrama Gantt con los pasos seguidos para la realización del proyecto, los cuales han sufrido cambios comparándolos con los de la planificación:

Tareas	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8	Semana 9
Página principal	█								
Creación de blogs		█							
Creación de webs			█	█					
Implementación base de datos					█				
Inicio de Sesión y Registro					█				
Interfaz de Usuario						█	█	█	█
Descarga de creaciones							█	█	█
Creación de la memoria *	█	█	█	█	█	█	█	█	█

TABLA 5.1. DIAGRAMA GANTT

\* La memoria se ha realizado progresivamente con la realización del proyecto.

Todos estos datos se pueden ver de forma ampliada en las siguientes imágenes obtenidas de la herramienta Git Kraken:

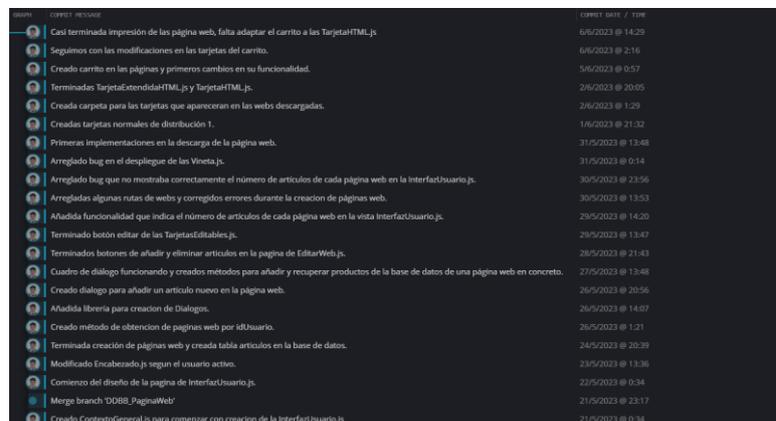


FIGURA 5.2. ÚLTIMOS AVANCES EN EL PROYECTO

commit	commit message	commit hash / time
	Comienzo del diseño de la página de insertarUsuario.js.	22/5/2023 @ 12:34
	Merge branch 'DDBB_PáginaWeb'	21/5/2023 @ 23:17
	Creado ContextoGeneral.js para comenzar con creación de la InterfazUsuario.js	21/5/2023 @ 0:34
	Completado Registro.js totalmente funcional, y comienzo de inserción de datos en la base de datos (tabla de páginas web creada).	19/5/2023 @ 14:04
	Completado InicioSesion.js ya funcional y comenzado con Registro.js, donde ya se guardan los datos en el estado.	19/5/2023 @ 1:31
	Implementado back-end en la aplicación. Primera conexión con la base de datos realizada	17/5/2023 @ 14:16
	Modificado artículos del blog (finalizado) y creación de paginas de InicioSesion.js y Registro.js.	9/5/2023 @ 17:01
	Modificado aspecto del blog, introduciendo 'artículos' para cada blog, Y corregido aspecto de las Tarjetas.js	9/5/2023 @ 0:46
	Creada Web_Paso_4.js ya con Tarjetas dinamicas y guardado de datos en una lista.	8/5/2023 @ 14:22
	Adaptando las paginas de creación de la página web a las distintas distribuciones.	4/5/2023 @ 13:18
	Creada página para seleccionar la distribución a usar durante la página web.	4/5/2023 @ 0:46
	Creada página de inserción de artículos para la página web.	3/5/2023 @ 13:14
	Agregadas viñetas explicativas en todas las páginas creadas hasta el momento.	3/5/2023 @ 1:57
	Creando viñetas explicativas para cada elemento.	29/4/2023 @ 14:05
	Terminado diseño de creación de plantillas de Blogs y Página web, siguiente paso mantenimiento y cambios.	26/4/2023 @ 2:19
	Comienzo de la parte de creación de páginas web y añadido modo oscuro a algunos elementos.	24/4/2023 @ 0:34
	Añadidos los datos de la primera página de la creación del blog al contexto y en desarrollo última página de creación del blog.	15/4/2023 @ 1:35
	Creado ContextoBlog.js con los datos necesarios en las distintas ventanas de la creación del blog.	14/4/2023 @ 2:51
	Terminado Blog_Paso_2.js y modificado componente TextoForm.js por TextoLargo.js de cara a usarlo en cualquier texto a mostrar en la a...	12/4/2023 @ 2:44
	Terminada vista principal y primer paso de creación de blog.	10/4/2023 @ 3:39
	Primeros pasos en la creación de la vista de la página principal.	4/4/2023 @ 17:04

FIGURA 5.1. PRIMEROS AVANCES EN EL PROYECTO

Esta herramienta es muy útil e intuitiva ya que ofrece una configuración, donde seleccionar solo los datos que deseamos visualizar. En este caso, se visualizan los datos necesarios para interpretar la planificación de trabajo cronológicamente. En la primera columna se ve la foto del autor de los cambios en el proyecto; a continuación, el *commit* realizado por la persona en cuestión informando de las actualizaciones realizadas; y en la última columna se muestra el momento exacto del *commit*. Todas estas columnas están ordenadas de forma cronológica, desde los cambios más actuales a los más antiguos.



# 6 Desarrollo del código

En esta sección se presentarán y explicarán diversos aspectos relacionados con todo el ámbito de la programación y el desarrollo de código. Uno de los aspectos a tratar es la estructuración de los ficheros que componen el proyecto completo, otro aspecto serán determinados fragmentos de código extraídos de diferentes partes del proyecto, que muestran una funcionalidad o características clave del proyecto. El objetivo principal de esta sección es la de ofrecer una visión más detallada del desarrollo del proyecto, destacando decisiones de diseño y las distintas formas de uso de las tecnologías implementadas.

## 6.1 ESTRUCTURA Y EJEMPLOS DE IMPLEMENTACIÓN

En primer lugar explicar la estructura usada para organizar los ficheros del proyecto, esta se divide en 4 carpetas principales: una donde almacenar los componentes que usaremos en la aplicación, ya que cabe recordar que el proyecto usa React como librería de JavaScript para ayudar en el desarrollo de interfaces de usuario; otra que contendrá los contextos usados por la plataforma para su correcto funcionamiento; todos los ficheros de contenido multimedia se agrupan en una carpeta llamada imágenes; y por último todos los ficheros que corresponden a las páginas como tal se agrupan en otra carpeta llamada vista. Estas carpetas junto a demás archivos de configuración o acceso de bases de datos son almacenadas dentro de la carpeta src:

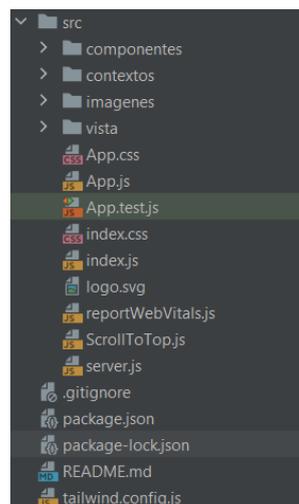


FIGURA 6.1. ORGANIZACIÓN DEL PROYECTO



A continuación, se muestran algunos de los aspectos más importantes y a tener en cuenta del proyecto:

- Uno de los principales componentes y que más peso tiene en el proyecto, es un fichero llamado 'Vineta.js', el cual representa la ideología de un proyecto sencillo e intuitivo. Este componente representa un círculo con una 'i' en su interior y que despliega un cuadro con texto dentro de este. Es usado principalmente como cuadros de ayuda, que surten información y tratan de facilitar todas las tareas que podemos encontrar en la interfaz. Está orientado a las personas que tienen una base de conocimientos técnicos baja sobre el sector informático. El código usado es el siguiente:

```
function Vineta({texto}) { 15+ usages 1 janar01
  const [showPopup, setShowPopup] = useState()

  return (
    <div onMouseOver={() => setShowPopup(true)}
      onMouseOut={() => setShowPopup(false)}
      className={"w-5 h-5 rounded-full bg-amber-200 border-2 border-amber-800 relative"}>
      <p className={"flex justify-center items-start text-xs font-semibold"}>i</p>
      {
        showPopup && (
          <main
            className={"absolute top-0 left-6 w-max h-min text-lg bg-amber-200 border-2 border-amber-800 p-1 rounded-lg"}
            style={{zIndex: 1}}
          >
            <p className={""}>{texto}</p>
          </main>
        )
      }
    </div>
  )
}
```

FIGURA 6.2. CÓDIGO DEL COMPONENTE VINETA.JS

Como podemos apreciar, el cuadro no se despliega hasta que el usuario no pase con el cursor sobre el círculo de información. Luego, en este cuadro se indexarán los cuadros de texto correspondientes a cada situación que sirvan de ayuda al usuario. A continuación, podemos ver cómo sería dicho elemento dentro de la plataforma online:

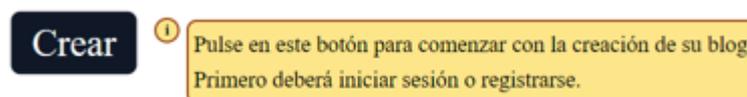


FIGURA 6.3. VISUALIZACIÓN DEL COMPONENTE EN LA PLATAFORMA

- Al trabajar con React, es necesario renderizar todo el código empleado, ya que este no trabaja con HTML, si no con JavaScript, por lo que existe un archivo

principal que ejecuta la aplicación renderizando todo el código con el que trabajamos para hacer funcional la plataforma online. En este caso ese fichero se llama index.js, y el fragmento de código usado para renderizar el demás código es este:

```
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <BrowserRouter>
    <ScrollToTop/>
    <App/>
  </BrowserRouter>
);
```

FIGURA 6.4. LÍNEAS DE CÓDIGO NECESARIAS PARA RENDERIZAR EL CÓDIGO PASADO POR LA APP

- Como la plataforma online cuenta con varias ventanas diferentes, es necesario añadir navegación entre ficheros al proyecto, para ello se utiliza las librerías Route y Routes de React-router-dom. Estas librerías aportan una serie de componentes a la aplicación para añadir las rutas necesarias para acceder a cada fichero. Su implementación se verá de la siguiente forma:

```
<GeneralProvider>
  <BlogProvider>
    <WebProvider>
      <Routes>
        <Route exact path="/" element={<PaginaPrincipal/>}/>
        <Route exact path="vista/general/InicioSesion" element={<InicioSesion/>}/>
        <Route exact path="vista/general/Registro" element={<Registro/>}/>
        <Route exact path="vista/general/InterfazUsuario" element={<InterfazUsuario/>}/>
      </Routes>
      <Routes>
        <Route exact path="vista/blogs/Blog_Paso_1" element={<Blog_Paso_1/>}/>
        <Route exact path="vista/blogs/Blog_Paso_2" element={<Blog_Paso_2/>}/>
        <Route exact path="vista/blogs/Blog_Paso_3" element={<Blog_Paso_3/>}/>
      </Routes>
      <Routes>
        <Route exact path="vista/webs/Web_Paso_1" element={<Web_Paso_1/>}/>
        <Route exact path="vista/webs/Web_Paso_2" element={<Web_Paso_2/>}/>
        <Route exact path="vista/webs/Web_Paso_3" element={<Web_Paso_3/>}/>
        <Route exact path="vista/webs/Web_Paso_4" element={<Web_Paso_4/>}/>
        <Route exact path="vista/webs/EditarWeb" element={<EditarWeb/>}/>
      </Routes>
    </WebProvider>
  </BlogProvider>
</GeneralProvider>
```

FIGURA 6.5. COMPONENTES ROUTE QUE CREAN LA COMUNICACIÓN EN LA PLATAFORMA ONLINE

Como podemos apreciar en la imagen, las rutas van contenidas dentro de los contextos a los que tendrán acceso, los cuales ofrecen una serie de métodos y

variables usadas por demás componentes de la aplicación, y que dan persistencia a la aplicación entre sus distintas ventanas.

- Uno de los principales conceptos que se deben aprender como base para usar React son los contextos, necesarios para añadir consistencia y persistencia a cualquier aplicación web. El proyecto cuenta con 3 tipos de contextos: el contexto general, al cual se tiene acceso desde cualquier fichero de la aplicación que contiene los datos del usuario con el que operamos; un contexto web, que contiene los datos relacionados con la creación de páginas web; y, por último, un contexto blog, que al igual que el de la web, contiene todos los datos relacionados con la creación de blogs. En estos hay variables u objetos como datos para la creación de blogs, el diseño implementado en cada uno, los artículos de cada tienda online, además de una gran cantidad de métodos que dan acceso a estos datos para poder modificarlos. Como ejemplo veremos los métodos y objetos del contexto general:

```
const [usuario, setUsuario] = useState(JSON.parse(localStorage.getItem("usuario")) || {});

function iniciarSesionUsuario(datos) { 1 usage  Jariar01
  setUsuario({
    idUsuario: datos.idUsuario,
    NombreUsuario: datos.NombreUsuario,
    Nombre: datos.Nombre,
    Apellidos: datos.Apellidos,
    Correo: datos.Correo,
  })
  localStorage.setItem("usuario", JSON.stringify(datos));
}
```

FIGURA 6.6. DECLARACIÓN Y CREACIÓN DE USUARIOS EN EL CONTEXTO GENERAL

En esta imagen vemos como se declara el objeto 'usuario', para posteriormente crear un método que cree un usuario. Estos son pasados a través del componente Provider, que da acceso a todo lo que este contenga como atributos:



```
return (  
  <ContextoGeneral.Provider  
    value={{  
      usuario,  
      iniciarSesionUsuario,  
      cerrarSesionUsuario  
    }}>  
    {children}  
  </ContextoGeneral.Provider>  
);
```

FIGURA 6.7. FUNCIONES Y OBJETOS A LOS QUE DA ACCESO EL CONTEXTO GENERAL

- Los contextos es uno de los tipos de *hooks* más utilizados. Estos son funciones especiales que permiten a los componentes de React tener estados y otras características sin tener que escribir clases. Algunos de los hooks más importantes y usados en el proyecto son *useState*, *useEffect* o *useContext*, como hemos podido comprobar.
  - o UseState: permite a las constantes tener un estado interno y gestionar sus actualizaciones de forma dinámica. Se importa de la librería React, y su sintaxis para su declaración es:

```
const [usuarios, setUsuarios] = useState([])
```

FIGURA 6.8. EJEMPLO DE DECLARACIÓN DE USESTATE

En este caso se declara un objeto como valor inicial al escribir los corchetes en el paréntesis. Los valores usados dentro del corchete de la constante definen el valor actual del estado (usuarios), y el método con el que alterar ese estado (setUsuarios).

- o UseEffect: permite ejecutar efectos secundarios en los componentes funcionales. Estos ocurren en respuesta a ciertos eventos y pueden usarse para diversos propósitos como pueden ser realizar solicitudes HTTP, actualizar títulos de páginas o gestionar animaciones entre otros. Un ejemplo de su sintaxis básica es:

```
useEffect(loadData, [])
```

FIGURA 6.9. EJEMPLO DE USO DE USEEFFECT



En este ejemplo se ejecutará la función `loadData` del componente según se cargue dicho componente.

- UseContext: permite acceder al contexto concreto de una aplicación desde un componente funcional. Este proporciona una forma de compartir datos entre componentes sin tener que pasar estos mediante argumentos entre los distintos niveles de jerarquías. El uso de contextos implica un mayor número de conocimientos para su uso. En primer lugar, debemos crear el contexto:

```
export const ContextoGeneral = createContext();
```

FIGURA 6.10. EJEMPLO DE CREACIÓN DE CONTEXTO

Seguidamente debemos utilizar el componente 'Provider' para envolver los componentes a los que se otorgará acceso al contexto.

```
<ContextoGeneral.Provider  
  value={{  
    usuario,  
    iniciarSesionUsuario,  
    cerrarSesionUsuario  
  }}>  
  {children}  
</ContextoGeneral.Provider>
```

FIGURA 6.11. EJEMPLO DE USO DEL COMPONENTE 'PROVIDER'

Dentro del *provider* se escribirán los componentes a los que se otorgará acceso, en este caso escribimos *children* para otorgar acceso a todos los componentes que se escriban dentro de este. El atributo *value* indica los datos compartidos por el contexto. Para acceder a dicho contexto deberemos declararlo dentro del componente envuelto donde deseemos acceder a él de la forma:

```
const {usuario} = useContext(ContextoGeneral)
```

FIGURA 6.12. EJEMPLO DE DECLARACIÓN DE UNA CONSTANTE QUE ACCEDE AL CONTEXTO

Como se puede apreciar, en este caso accedemos al estado usuario del Contexto General.



- Otro aspecto muy importante es el uso del *framework* de CSS Tailwind, con el que se ha dado aspecto al proyecto. Este se basa en CSS, sin embargo, una de las principales diferencias es el poder escribir el estilo del propio componente dentro del fichero HTML, pero de una forma sencilla, intuitiva y ágil, debido a que aporta muchos valores por defecto y además debe su fama a las abreviaciones de los diferentes atributos de CSS, lo que agiliza su escritura. Para escribir el estilo de un componente debemos declararlo dentro de este de la forma :

```
<section className={"text-black h-screen rounded-lg grid content-center align-middle grid-cols-2 gap-0"}>
```

FIGURA 6.13. EJEMPLO DE USO DE TAILWIND

A continuación, se explican los atributos:

- *Text-black*: es el reemplazo de la propiedad color del CSS, en este caso con el negro como color predeterminado
- *H-screen*: es el reemplazo de la propiedad *height* de CSS, en este caso indica que el largo del componente será el largo de la pantalla.
- *Rounded-lg*: es el reemplazo de la propiedad *border-radius* de CSS, en este caso indica que el borde será de 0.5 rem, ya que es el valor predeterminado.
- *Grid*: es el reemplazo de la propiedad *display* con el valor *grid* de CSS, en este caso con solo usar esta palabra ya se especifica el tipo de *display*.
- *Content-center*: es el reemplazo de *align-content* con el valor center de CSS, en este caso se especifica de esta forma.
- *Align-middle*: es el reemplazo de *vertical-align* con el valor *middle* de CSS.
- *Grid-cols-2*: es el reemplazo de *grid-template-columns* de CSS, en este caso especificando que el número de columnas es 2.
- *Gap-0*: es el reemplazo de gap en CSS, donde aquí se escribe con el valor directamente.

Estos son unos pocos ejemplos de cómo el *framework* Tailwind facilita y agiliza el trabajo a la hora de estilizar nuestros componentes. Este fue elegido por delante de otros como Bootstrap o Foundation debido a la amplia cantidad de utilidades que se pueden combinar para construir interfaces personalizadas, ya que los otros



dos *frameworks* ofrecen menos opciones de personalización y más diseños predefinidos por ellos.

- Sin embargo, de Bootstrap provienen todos los iconos que se han utilizado en los diferentes componentes de la aplicación, ya que debido a su fácil implementación ofrecen una oportunidad de estilizar los componentes muy atractiva, con lo que se puede facilitar la comprensión de determinados componentes a los usuarios haciéndolo de una forma visual. Un ejemplo de uso de estos iconos es la siguiente:

```
import {CodeSquare, Person} from "react-bootstrap-icons";
```

FIGURA 6.14. EJEMPLO DE IMPORTACIÓN DE ICONOS

De esta forma se importan los iconos de 'CodeSquare' y 'Person' de la página [33], donde podemos encontrar muchos más iconos para uso de forma gratuita.

- En lo relacionado con el acceso de bases de datos, se han realizado todos los métodos encargados de esto en un solo fichero denominado 'server.js'. En este se utiliza express junto a mysql, dependencias necesarias para realizar las consultas HTTP contra la base de datos. Express usará la función 'cors' (*Cross-Origin Resource Sharing*), *middleware* propio de express que habilita que el servidor acepte solicitudes HTTP desde un origen diferente al suyo. Todas estas declaraciones e importaciones de dependencias son de la siguiente forma:

```
const express = require('express')
const cors = require('cors')
const mysql = require('mysql')
const app = express()

app.use(cors())
app.use(express.json())
```

FIGURA 6.15. DECLARACIÓN DE DEPENDENCIAS Y CONSTANTES.

Como se puede apreciar en la imagen, instamos a express a tratar con datos en forma *json*. Para acceder a la base datos es necesario indexar las credenciales de acceso a esta, lo que se realizará a través de una constante, cuyos valores son *host*, *password*, *user* y *database*. Los métodos usados para realizar el intercambio

de datos contra la base de datos usan los métodos de *get* o *post*. Se puede ver un ejemplo de uso de cada uno a continuación:

```
app.get("/api/listadoUsuarios", (req, res) => {
  const connection = mysql.createConnection(credenciales);
  connection.connect((error) => {
    if (error) {
      res.status(500).send(error);
    } else {
      connection.query("SELECT * FROM usuarios", (error, result) => {
        if (error) {
          res.status(500).send(error);
        } else {
          res.status(200).send(result);
        }
      });
      connection.end();
    }
  });
});
```

FIGURA 6.16. MÉTODO GET PARA RECUPERAR USUARIOS DE LA BASE DE DATOS.

En esta imagen se ve un método *get* para recuperar un listado con todos los usuarios que tenemos almacenados en la base de datos. Para realizar la consulta se utiliza SQL nativo.

```
app.post("/api/crearUsuario", (req, res) => {
  const { nombre, apellidos, nombreUsuario, contrasena, correo } = req.body;
  const parametros = [[nombre, apellidos, nombreUsuario, contrasena, correo]];
  const connection = mysql.createConnection(credenciales);

  connection.query("INSERT INTO usuarios (Nombre, Apellidos, NombreUsuario, Contraseña, Correo) VALUES ?", [parametros], (error, result) => {
    if (error) {
      res.status(500).send(error);
    } else {
      res.status(200).send(result);
    }
  });
  connection.end();
});
```

FIGURA 6.17. MÉTODO POST PARA INSERTAR UN USUARIO EN LA BASE DE DATOS

En esta imagen se ve un método *post* usado para insertar un usuario en la base de datos, el cual deberá tener todos los campos con valores válidos.

Para llamar a estos métodos desde los componentes se utiliza la biblioteca de JavaScript 'axios', la cual es basada en promesas. Por lo que para llamar al método *get* de recuperación de usuarios sería de la forma:

```
const loadData = () => { 1 usage  Jariar01
  axios.get('http://localhost:4000/api/listadoUsuarios')
    .then(result => {
      setUsuarios(result.data)
    })
}
```

FIGURA 6.18. FUNCIÓN USADA PARA RECUPERAR LOS USUARIOS DE LA BASE DE DATOS.

De esta forma se almacenará en el estado usuario todos los usuarios recuperados de la base de datos por el método *get* mostrado anteriormente. Para llamar al método *post*, aumenta el nivel de complejidad al tener que pasar parámetros en la llamada al método. La inserción de usuarios en la base de datos se realiza de la siguiente forma:

```
const registro = (e) => { 1 usage  Jariar01
  e.preventDefault()
  try {
    axios.post('http://localhost:4000/api/crearUsuario', usuario)
      .then(() => {
        manejadorNavegacion();
      })
  } catch (error) {
    console.error(error);
  }
};
```

FIGURA 6.19. FUNCIÓN ENCARGA DE INSERTAR UN USUARIO EN LA BASE DE DATOS.

En esta función es necesario incluir los bloques *try* y *catch* para manejar posibles excepciones tiradas por el método encargado de insertar usuarios en el fichero *server.js*. En este caso, se pasa como argumento el usuario a añadir, que dentro del método correspondiente creará una constante con esos datos para añadirlo de forma correcta a la base de datos.

Estos son unos de los métodos y funciones principales, entre muchos otros; con los que se realiza todo tipo de manipulación y transferencia de datos con la base de datos. Todos estos están controlados por bloques *try catch* y sentencias condicionales para que al usuario no le llegue ninguna excepción, simplemente un mensaje de error; y además no surja ningún problema con los datos una vez



introducidos que pueda tener consecuencias fatales en el uso de la plataforma online.

## 6.2 METODOLOGÍA DE DESARROLLO

En la actualidad existen diversas metodologías de desarrollo debido a la creciente demanda de productos y servicios digitales. Esto ha producido que se busque la forma más óptima de entregar *software* de manera rápida y eficaz, sin comprometer la calidad. Para lograr estos objetivos se hace uso de una metodología en cascada.

Una de las principales características de dicha metodología es el enfoque secuencial y su énfasis en la planificación y documentación detallada de cada etapa del desarrollo. Con el enfoque secuencial y estricto, se busca seguir un planteamiento estricto en la etapa de desarrollo de *software*, en la que no se esperan grandes cambios. Además, se prohíbe avanzar de fase hasta no haber completado la correspondiente en el planteamiento.

Por todo esto ha sido la metodología de desarrollo utilizada, al ser el proyecto de un tamaño considerable en el cual los requisitos se establecen en un inicio y no existe la posibilidad de que sean modificados en ningún momento [34].

## 6.3 PATRONES DE DISEÑO

En el ámbito de desarrollo *software*, los patrones de diseño son una herramienta valiosa para mejorar la calidad del producto, la eficacia del desarrollador y la mantenibilidad de los sistemas. Estos brindan soluciones típicas a problemas comúnmente recurrentes de diseño *software*. En el caso del proyecto, al trabajar con la librería React, de gran popularidad, disponemos de una amplia variedad de patrones de diseño. En esta sección se abarcan el más importante usado en el proyecto:

- Composición: como se indica en el primer apartado de este capítulo, una de las principales características de React es la capacidad de combinar elementos para formar uno en concreto. Con esta técnica se evita construir excesivos componentes, de gran similitud, y de esta forma optimizar el código.

A partir de la versión 16.8.0, React introdujo una nueva función denominada Hooks. Estas funciones hacen posible el uso de métodos de estado y ciclo de vida



de React. Estos los podemos ver desarrollados y explicados en la primera sección de este capítulo [35].



## 7 Base de datos

Como se menciona en el apartado de Tecnologías, la base de datos usada para el desarrollo y uso del proyecto está alojada en un host local en MySQL. Uno de los motivos principales para alojar la base de datos en MySQL es por sencillez de uso y previa experiencia adquirida en trabajos universitarios, lo que permite trabajar y manejar los servicios que esta ofrece con gran soltura. Además, es una aplicación gratuita que permite gestionar un gran número de bases de datos sin coste alguno al ser alojadas en un host local, perfecto para el desarrollo y prueba de proyectos. En concreto, la aplicación usada es MySQL Workbench, interfaz de usuarios oficial de MySQL, que permite diseñar, crear y buscar todos nuestros esquemas de bases de datos. Un factor a tener en cuenta de cara a futuras vías de desarrollo del proyecto, es el hecho de que esta interfaz también permite migrar bases de datos desde y para otros proveedores de gestión de bases de datos. A continuación, se muestra un diagrama de la base de datos que contiene todas las tablas con los datos:

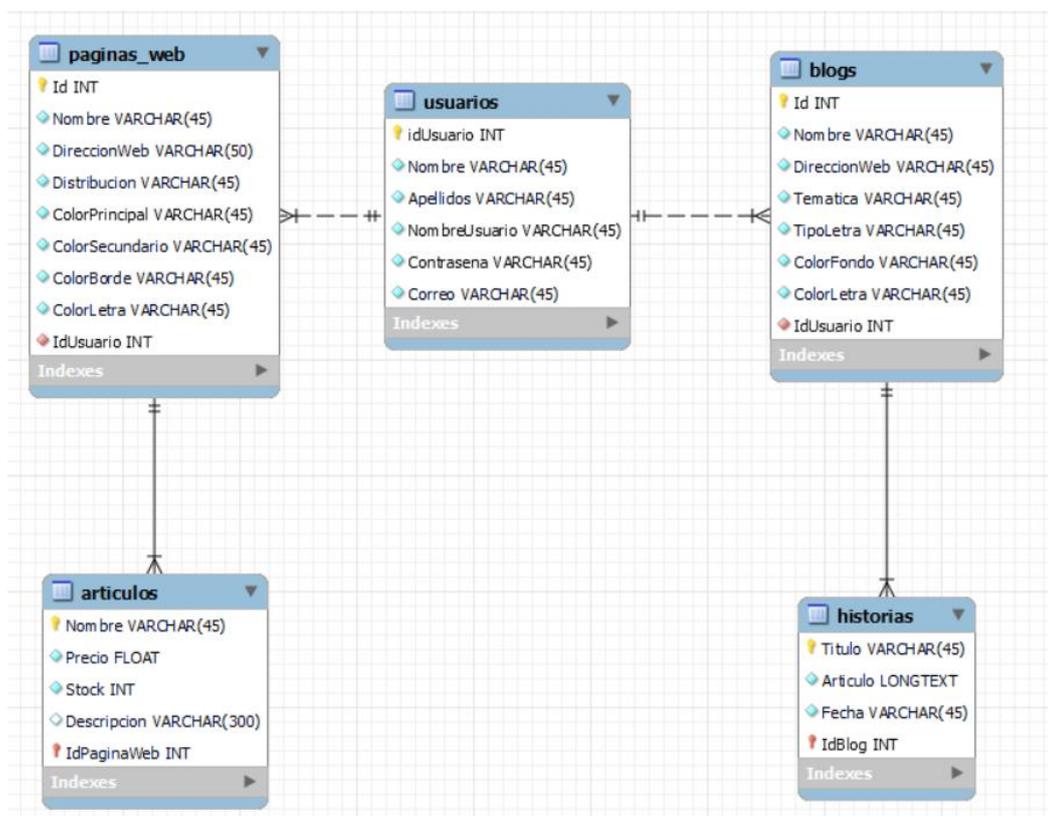


FIGURA 7.1. DIAGRAMA ENTIDAD-RELACIÓN DE LA BASE DE DATOS DEL PROYECTO



Como se puede apreciar, la base de datos cuenta con solamente 5 tablas: usuarios, donde se almacenan todos los usuarios que se han registrado; páginas web, donde se almacenan todas las páginas web con sus respectivos aspectos escogidos por los usuarios durante la creación de cada una de ellas; blogs, donde se almacenan todos los blogs creados por los usuarios con sus respectivos aspectos seleccionados durante la creación de este; artículos, tabla donde se almacenan los artículos que se mostrarán en cada tienda online a la que corresponden; y por último, historias, tabla donde se almacenan todos artículos escritos por los usuarios en sus blogs.

En MySQL es importante conocer los tipos de datos que podemos dar a las variables para optimizar al máximo la base de datos y acelerar el manejo de datos entre el proyecto y la propia base de datos. Por ello, se muestran los principales tipos de datos usados en la base de datos:

- *Int*: es usado mayoritariamente para las *primary key*, identificadores individuales en cada tabla para poder distinguir todos los datos almacenados en esta. Este tipo de datos guarda valores numéricos no muy elevados.
- *Varchar*: es de los tipos de datos más usados al ser el principal para almacenar cadenas de texto de todo tipo. En la declaración de este tipo de datos es necesario indicar el valor máximo de caracteres a almacenar. Además, es recomendable modificar el conjunto de caracteres admitidos por este tipo de datos a 'utf8', para evitar posibles fallos al introducir caracteres especiales o propios del español, como puede ser por ejemplo la 'ñ'.
- *Float*: es un tipo de datos que como *int*, almacena valores numéricos, sin embargo, *float* también admite números con decimales, lo que resulta la elección correcta para almacenar datos relacionados con el dinero, por ejemplo.
- *Longtext*: similar a *varchar*, *longtext* es un tipo de datos que almacena cadenas de caracteres, pero en este caso soportando un número amplísimo de estos. Este es usado para que nuestros usuarios con blogs, puedan escribir historias sin tener que preocuparse por la extensión de estas.



Existen otro gran número de tipos de datos que no usamos en la base de datos, como pueden ser bit, decimal, date, entre muchos otros que son específicos para guardar un tipo de datos en concreto, optimizando así la base de datos.



## 8 La plataforma

En esta sección se muestra como es la plataforma en concreto desarrollada, mediante el uso de imágenes con explicaciones. Para comenzar mostrando la interfaz con la que se encuentran los usuarios, mostraremos la página inicial, con estilo monocroma que se mantendrá durante las demás ventanas del proyecto. Los botones amarillos con una 'i' en su interior son explicativos, para facilitar la comprensión de todos los elementos y pasos a realizar en la plataforma, como se explica en el capítulo de Tecnologías.

### 8.1 PÁGINA PRINCIPAL



FIGURA 8.1.1. PRIMERA SECCIÓN DE LA PÁGINA PRINCIPAL

Como podemos ver en esta primera sección, se da la bienvenida al usuario con dos eslóganes. Como se puede ver en la parte superior, existe un encabezado que será común a toda la plataforma, donde se inicia sesión, se registra o cierra sesión. Además, el logo de la parte superior izquierda devolverá al usuario siempre a la página principal sin importar donde estemos. Seguidamente, bajando en la página, veremos las secciones dedicadas a la creación de blogs y páginas web respectivamente.

# Blog

Cree su blog donde podrá conectar con millones de usuarios para compartir sus ideas. Nosotros le ayudamos en todo.

## Representa todas tus ideas

Con WebDesign podrás elegir entre numerosas combinaciones para lograr la imagen mas fiel al contenido que usted va a representar.

## Creación guiada

Tendrá en cada elemento un texto de ayuda para indicarle y disolverle todas las dudas que puedan surgirle en el proceso de creación del blog.



Crear <sup>1</sup>

FIGURA 8.1.2. SEGUNDA SECCIÓN DE LA PÁGINA PRINCIPAL

Esta segunda sección de la página principal es la dedicada a la creación de blogs. Donde pulsando en el botón 'Crear', accederemos al proceso de creación.

# Página web

Cree su comercia online con una página web accesible para todos sus clientes y con la que logrará expandir sus ventas. Nosotros le ayudamos en todo.

## Cree un catálogo con sus productos

Con WebDesign podrás introducir todo su catálogo de productos de una forma sencilla para que todos sus clientes puedan verlos y comprarlos.

## Creación guiada

Tendrá en cada elemento un texto de ayuda para indicarle y disolverle todas las dudas que puedan surgirle en el proceso de creación del blog.

Crear <sup>1</sup>

FIGURA 8.1.3. TERCERA SECCIÓN DE LA PÁGINA PRINCIPAL

Por último, tenemos la sección dedicada a la creación de páginas web, donde al igual que en la sección de blogs, con únicamente pulsar en el botón 'Crear', accederemos al proceso de creación.



## 8.2 CREACIÓN DE BLOGS Y PÁGINAS WEB

Web Design

Jose [Cerrar sesión](#)

# Blog - Datos

Escriba el nombre de su blog: ⓘ

Ejemplo123...

Escriba la dirección web del blog: ⓘ

www.miBlog.es...

Escriba la temática de su blog: ⓘ

Cocina, naturaleza, sociedad...

Continuar ⓘ

FIGURA 8.2.1. PRIMER PASO DE CREACIÓN DE BLOGS



FIGURA 8.2.2. PRIMER PASO DE CREACIÓN DE PÁGINAS WEBS

En estos primeros pasos, lo solicitado al usuario es la inserción de los datos principales de ambas creaciones, desde su nombre hasta la dirección web donde se realizaría el hosting de la página web. Como se puede observar, para realizar ambos procesos es necesario estar registrado y haber iniciado sesión en la plataforma, de otra forma no se podrán comenzar las creaciones. Esto se debe a que todos los blogs y páginas web creadas se asignan a un usuario en concreto en la base de datos, para que luego solo este pueda acceder, modificar o eliminar dichas creaciones.



## Blog - Estilo <sup>1</sup>

Seleccione el tipo de letra que desea para su blog:

Seleccione el color de fondo de su blog:

Seleccione el color de letra de su blog:

Previsualizado (puede escribir aquí)

1

FIGURA 8.2.3. SEGUNDO PASO DE LA CREACIÓN DE BLOGS

Web Design Jose Cerrar sesión

## Web - Distribución <sup>1</sup>

Seleccione la distribución que desea usar en su página web: <sup>1</sup>

- Primera opción:



NOMBRE:  
PRECIO: €  
STOCK: ●



NOMBRE:  
PRECIO: €  
STOCK: ●



NOMBRE:  
PRECIO: €  
STOCK: ●



NOMBRE:  
PRECIO: €  
STOCK: ●

FIGURA 8.2.4. SEGUNDO PASO (1) EN LA CREACIÓN DE PÁGINAS WEB

- Segunda opción:



NOMBRE:  
DESCRIPCION:

PRECIO:  
STOCK: ●



NOMBRE:  
DESCRIPCION:

PRECIO:  
STOCK: ●

1

FIGURA 8.2.5. SEGUNDO PASO (2) EN LA CREACIÓN DE PÁGINAS WEB



En estos segundos pasos se realiza el diseño. En el caso de los blogs se seleccionarán datos como colores y tipos de letra, y en el de las páginas web, se seleccionará la distribución con la que se mostrarán los artículos en la tienda online creada. En el siguiente paso de creación de páginas web es donde se seleccionando los datos de diseño similares a los seleccionados en este paso de la creación de blogs. Cabe destacar, que en todos los pasos de diseño se muestra una pantalla donde se previsualizan los datos del estilo escogidos, actualizándose al momento.

## Web - Diseño •

Seleccione el color principal de su página web:

Seleccione el color secundario de su página web:

Seleccione el color del borde de la tarjeta:

Seleccione el color de la letra de los productos:

 NOMBRE: PRECIO: € STOCK: ● AÑADIR AL CARRITO <input type="button" value="✚"/>	 NOMBRE: PRECIO: € STOCK: ● AÑADIR AL CARRITO <input type="button" value="✚"/>	 NOMBRE: PRECIO: € STOCK: ● AÑADIR AL CARRITO <input type="button" value="✚"/>	 NOMBRE: PRECIO: € STOCK: ● AÑADIR AL CARRITO <input type="button" value="✚"/>
---	---	---	---

•

*FIGURA 8.2.6. TERCER PASO EN LA CREACIÓN DE PÁGINAS WEB*

Como podemos ver, en este paso, ya se seleccionan los demás factores de diseño.



# Blog - Artículo <sup>1</sup>

A continuación escribirá su primer artículo (abajo del todo verá previsualizado como queda)

Introduzca el título que se verá al principio del Artículo:

Comience a escribir su artículo:

Deje volar sus ideas y de rienda suelta a su creatividad

FIGURA 8.2.7. TERCER Y ÚLTIMO PASO EN LA CREACIÓN DE BLOGS

Como se puede apreciar en la figura previa, en el último paso de la creación de blogs se escribe el primer artículo que se publicará en el blog reciente creado. Para ello existen dos cuadros, donde se escribirá el título del artículo, y en el posterior todo el contenido de este. El resultado final se mostrará según se escriban ambos en el cuadro del final de la página.



FIGURA 8.2.8. CUARTO Y ÚLTIMO PASO EN LA CREACIÓN DE PÁGINAS WEB.



Como se puede apreciar en la figura previa, en el último paso de la creación de páginas web, se introducirán los artículos que habrá a la venta en esta. En el cuadro del final de la ventana se mostrarán los artículos, según se vayan añadiendo, con la distribución establecida y el diseño introducido en los pasos previos. Estos podrán ser luego modificados en la interfaz propia de cara usuario.

### 8.3 INTERFAZ DE USUARIO

Se ha implementado una interfaz donde los usuarios puedan consultar sus datos personales y sus creaciones. Esta está formada por tres secciones\*:

Información personal >	<p><b>Datos personales:</b></p> <p>Nombre: Jose</p> <p>Apellidos: Arias</p> <p>Nombre de usuario: Jose</p> <p>Correo: correo</p>
Mis blogs >	
Mis páginas web >	

FIGURA 8.3.1. PRIMERA SECCIÓN: DATOS PERSONALES DEL USUARIO.

Información personal >	<p><b>Mis blogs:</b></p> <table border="1"> <tr> <td>Nombre del blog: Aventurero Dirección web: www.aventurero.es Temática: Ciencia Ficción</td> <td></td> <td></td> </tr> <tr> <td>Nombre del blog: Las aventuras de Gullivert Dirección web: www.gulli.es Temática: Ciencia Ficción</td> <td></td> <td></td> </tr> <tr> <td>Nombre del blog: Subaru Dirección web: www.subaru.es Temática: Motor</td> <td></td> <td></td> </tr> </table>	Nombre del blog: Aventurero Dirección web: www.aventurero.es Temática: Ciencia Ficción			Nombre del blog: Las aventuras de Gullivert Dirección web: www.gulli.es Temática: Ciencia Ficción			Nombre del blog: Subaru Dirección web: www.subaru.es Temática: Motor		
Nombre del blog: Aventurero Dirección web: www.aventurero.es Temática: Ciencia Ficción										
Nombre del blog: Las aventuras de Gullivert Dirección web: www.gulli.es Temática: Ciencia Ficción										
Nombre del blog: Subaru Dirección web: www.subaru.es Temática: Motor										
Mis blogs >										
Mis páginas web >										

FIGURA 8.3.2. SEGUNDA SECCIÓN: BLOGS



Información personal >	<h3>Mis páginas web:</h3> <table border="1"><tr><td>Nombre: Comercial Dirección web: comercialArias Número de artículos: 3</td><td> </td></tr><tr><td>Nombre: ComercialArias Dirección web: www.comercial.es Número de artículos: 0</td><td> </td></tr><tr><td>Nombre: Madridistas Dirección web: www.HalaMadrid.es Número de artículos: 2</td><td> </td></tr></table>	Nombre: Comercial Dirección web: comercialArias Número de artículos: 3	 	Nombre: ComercialArias Dirección web: www.comercial.es Número de artículos: 0	 	Nombre: Madridistas Dirección web: www.HalaMadrid.es Número de artículos: 2	 
Nombre: Comercial Dirección web: comercialArias Número de artículos: 3	 						
Nombre: ComercialArias Dirección web: www.comercial.es Número de artículos: 0	 						
Nombre: Madridistas Dirección web: www.HalaMadrid.es Número de artículos: 2	 						
Mis blogs >							
Mis páginas web >							

FIGURA 8.3.3. TERCERA SECCIÓN: PÁGINAS WEB

\* Las secciones contienen creaciones de ejemplo.

Las secciones de blogs y páginas web muestran cada creación mediante listas de tarjetas. Estas contienen los datos introducidos en el primer paso de creación de ambos y dos botones, uno para editar esa creación en concreto, y otro para descargar los ficheros de la creación correspondiente.



## 9 Privacidad

Debido a que el proyecto cuenta con una base de datos que manejará datos personales de los usuarios, es fundamental asegurarse de cumplir con el Reglamento (UE) 2016/679 del parlamento europeo y del consejo, de 27 de abril de 2016 [36]. El RGPD es una regulación impuesta por la Unión Europea, que establece las normas y principios fundamentales a cumplir con el tratamiento de datos personales con el objetivo de garantizar la privacidad y los derechos de los individuos. Los aspectos clave a cumplir mientras trabajamos con la plataforma son:

- Consentimiento informado: la plataforma debe solicitar el consentimiento explícito de los usuarios de manera clara y completa, especificando como se tratarán sus datos personales, además del acceso que tendrán a ellos.
- Minimización y finalidad de datos: la plataforma debe recopilar y utilizar el menor número de datos necesarios para cumplir con los objetivos establecidos.
- Seguridad de los datos: la plataforma debe contar con medidas de seguridad adecuadas para proteger los datos personales contra accesos no autorizados o pérdidas de estos.
- Derechos de los usuarios: la plataforma debe permitir a los usuarios ejercer sus derechos de acceder, rectificar, suprimir y oponerse al tratamiento de sus datos personales.
- Transferencias internacionales de datos: si la plataforma realiza transferencias de datos personales fuera de la Unión Europea, se deben cumplir los requisitos adecuados para garantizar la protección de datos equivalente al país de destino.
- Registro de actividades de tratamiento: el proyecto debe contar con un registro de todas las actividades realizadas relacionadas con el tratamiento de datos, incluyen todo tipo de detalles.
- Evaluaciones de impacto en la protección de datos: deben realizarse evaluaciones periódicas de impacto en la protección de datos para abordar posibles riesgos para la privacidad de los usuarios y asociados.

Estos son unos de los aspectos más importantes con los que deberá cumplir nuestro proyecto.



El cumplimiento del RGPD no es un evento único, es un enfoque continuo que deberá garantizar la protección de los datos a los usuarios en todo momento. Por ello es importante contar con asesoramiento legal especializado y mantenerse actualizado sobre los cambios en la legislación que puedan realizarse con el paso del tiempo. Es importante tener en cuenta que cumplir el RGPD además de ser una obligación legal, también puede suponer una oportunidad para generar confianza en los usuarios que interactúan con nuestra plataforma online.



# 10 Planes a futuro

El proyecto actual ya tiene una base muy fuerte en lo referido a sus funcionalidades ya implementadas, pero como es obvio aún está en vías de desarrollo y con muchos cambios e implementaciones posibles que mejoren el producto. A continuación, se indican una serie de posibles implementaciones que mejorarían la apariencia y el funcionamiento del proyecto:

- Diseño: la plataforma online tiene como principio el ofrecer una interfaz intuitiva y simple al usuario, con la que este a su vez pueda diseñar libremente sus blogs o páginas web. Sin embargo, los diseños ofrecidos pueden llegar a resultar tan simple como nuestra interfaz, por lo tanto, en vías de futuro una gran mejora para aumentar el número de usuarios que interaccionan con nuestra plataforma online sería introducir nuevos aspectos y más elementos personalizables dentro de la creación de ambos sitios web. Claros ejemplos serían añadir más distribuciones en las páginas web, poder añadir opciones de búsqueda dentro de cada una, el poder introducir imágenes o elementos audiovisuales dentro de los blogs, y muchos otros más que ofrecerían más personalidad a cada creación de nuestros usuarios. Estas implementaciones no requerirían costes adicionales. Además, otra alternativa a tener en cuenta es la de ofrecer plantillas y diseños predeterminados, para los usuarios menos exigentes y que quieren el producto final en el momento, sin tener que perder su tiempo diseñando y personalizando su página web. De esta forma lograríamos llegar a otro sector de público distinto al que llegaríamos actualmente.
- Funcionalidades: actualmente las funcionalidades que se ofrecen en la creación de blogs y páginas webs son bajas, ya que también hay que tener en cuenta que la plataforma busca a llegar a aquellas personas que requieren de un producto simple, que ellos mismos logren comprender a la perfección y que sus clientes también puedan hacerlo al igual que ellos. Añadiendo funcionalidades nuevas en las creaciones, podría desvirtuarse el enfoque simple que ofrecemos, por ello es importante siempre pensar en si realmente la funcionalidad a añadir es realmente necesaria, y si se debe implementar buscar el enfoque perfecto para no añadir



- dificultad de comprensión o un extra muy abultado de conocimientos que puedan suponer un obstáculo para el público menos habilidoso en el mundo informático.
- Hosting y dominios: actualmente debido a la inexistencia de presupuesto y que el proyecto no se ha lanzado, no ofrece servicio de hosting ni la posibilidad de adquirir un subdominio propio. Este debería ser el primer cambio a implementar en la aplicación en un futuro ya que es parte de la obtención de ingresos económicos como ya mencionamos anteriormente. Para ello es necesario contactar con páginas que ofrecen hosting y crear un programa de afiliados con ellos que genere ingresos a ambos, todo sin tener que suponer un coste adicional al usuario más que el propio de la página de hosting. Además, para ofrecer dominios propios, la empresa deberá realizar un gasto significativo al adquirir un dominio propio con las características adecuadas como para poder ofrecer subdominios de calidad a los clientes.
  - Idiomas: la plataforma actualmente solo se encuentra disponible en español, más concretamente en castellano. Un cambio significativo que ampliaría enormemente nuestro público sería tener implementado otro tipo de lenguajes con los que llegar a un mayor número de personas. Entre los idiomas a implementar que más personas comprenden, son el inglés (segundo idioma más hablado del mundo), y el portugués y francés. Esta implementación de diversos idiomas podría darse como podemos ver en muchas páginas web actuales, mediante un único botón con la bandera del idioma correspondiente para que todo tipo de persona, sin conocimientos informáticos, logre modificar el idioma de toda la plataforma, haciendo de esta más accesible para todo tipo de usuarios.
  - Sección de contacto: actualmente la sección de contacto para que el cliente pueda contactar con un asistente técnico no está implementada. Una vez se lance el proyecto de una forma definitiva, es un factor muy a tener en cuenta que los usuarios que interactúan con nuestra plataforma puedan contactar con asistentes para poder comentar cualquier tipo de error u obstáculo que se les ha aparecido en cualquier momento, ya bien sea creando como editando sus creaciones. Para ello, se deberá contratar personal encargado de responder a todo este tipo de problemas de una forma ágil y eficaz.



- Ayuda mediante asistente: una implementación que supondría un gran avance en el proyecto sería la implementación de un cuadro donde los usuarios puedan dialogar con un asistente virtual. Este asistente resolverá todo tipo de dudas y suministrará ayuda a los usuarios en cualquier ventana en la que se hallen. Para ello, se deberá desarrollar una IA que sea la encargada de interactuar con nuestros usuarios, ya que es más veloz que contactar continuamente con nuestro centro de contacto, tanto para el usuario a la hora de generar la pregunta, como tanto para nuestros trabajadores suministrarle a este la mejor alternativa a su consulta.
- Aplicación móvil: este supondría mucho esfuerzo ya que es un avance significativo, pero el hecho de poder acceder a todas las creaciones desde una aplicación móvil supondría un aumento de la accesibilidad de cara al usuario, que podría editar y modificar sus sitios web desde cualquier lugar simplemente con su teléfono móvil. Además, también se podrían modificar otro tipo de aspectos desde la aplicación móvil, como migrar el dominio para otro o gestionar el hosting.
- Integración con redes sociales: en el mundo actual es muy importante la publicidad que hacemos de nuestros productos, por ello un cambio importante es contar con integración fluida con redes sociales populares, facilitando de esta forma el poder compartir todo nuestro contenido y productos a través de ellas. Esto ayuda también a aquellas personas que apenas tienen conocimientos sobre redes sociales, pero que quieren expandir su público a través de ellas y aumentar su alcance.

Estos son unos de los cambios más destacables a realizar en el proyecto en un futuro, aunque también existen otros relacionados con la seguridad o colaboraciones con otras herramientas que ayuden a mejorar, además de tener en cuenta todos los avances tecnológicos que se producen en la actualidad y pueden resultar útiles en nuestro proyecto.



# 11 Conclusiones

Las plataformas online de creación de tiendas online o blogs se han vuelto cada vez más populares, esto se debe principalmente al impacto que ha tenido en los últimos años la venta online. Por ello, el proyecto ha tenido como objetivo aprovechar ese aumento, pero centrándose principalmente en las personas con un menor conocimiento informático o sobre Internet.

Unas de las características más destacadas son la sencillez y las ayudas, primordiales para la consecución de los objetivos propuestos. La plataforma ofrece diversas funciones como creaciones de blogs y tiendas online con un amplio abanico de personalización, ya que es un aspecto muy importante en el éxito y la popularidad que puede obtener la plataforma. Estos factores también son de gran importancia en el cumplimiento del presupuesto, al estar estrechamente relacionado con el número de usuarios que interactúan con el proyecto.

Para lograr todas las características expuestas en este documento, se han utilizado las mejores tecnologías que se adaptaban a los objetivos propuestos, facilitando el trabajo al trabajador, y logrando mejores resultados finales. Además, se han estudiado distintas metodologías y patrones consiguiendo reducir tiempos y una mejor planificación del desarrollo total del proyecto.

En última instancia, el proyecto puede ser descrito como una plataforma en línea diseñada para facilitar a aquellos usuarios con un bajo nivel de conocimientos informáticos, el obtener una tienda online o un blog propios.



## 12 Bibliografía

- [1] J. García, “WordPress Opiniones (2023): ¿Vale la pena la versión con hosting?,” 2023. <https://www.tooltester.com/es/opiniones/wordpress-com/>.
- [2] “Web.com - Opiniones, precios y características - Capterra España 2023,” 2023. <https://www.capterra.es/software/151967/web-com#reviews>.
- [3] P. Londoño, “19 plataformas para crear un blog en 2023 (y cómo elegir),” 2023. <https://blog.hubspot.es/website/paginas-para-crear-un-blog>.
- [4] “Mejores10-creadores de paginas web,” 2022. [https://www.mejores10-creadoresdepaginasweb.es/es/web/?matchtype=b&utm\\_source=GoogleAds&utm\\_campaign=15360199578&gclid=CjwKCAjwvpCkBhB4EiwAujULMrnvu03rjZ9LDMlzwjsex6WC2UiHqo8zpLfYcnYdyDw031C\\_tNfG7xoCIN0QAvD\\_BwE](https://www.mejores10-creadoresdepaginasweb.es/es/web/?matchtype=b&utm_source=GoogleAds&utm_campaign=15360199578&gclid=CjwKCAjwvpCkBhB4EiwAujULMrnvu03rjZ9LDMlzwjsex6WC2UiHqo8zpLfYcnYdyDw031C_tNfG7xoCIN0QAvD_BwE).
- [5] O. de la Fuente, “1and1 IONOS - ¿Por qué tiene tan mala fama? Cuidado si vas a contratarlo... - Haciaelautoempleo,” 2023. <https://www.haciaelautoempleo.com/1and1-ionos-hosting/#Pros-de-contratar-1and1>.
- [6] J. García, “Shopify Opiniones (2023): ¿Realmente el Rey del ecommerce?,” 2023. <https://www.tooltester.com/es/opiniones/shopify/#conclusion>.
- [7] Miri Colman, “Blogger opiniones 2023: ¿Sigue siendo una buena opción para tu blog?,” 2023. <https://www.websiteplanet.com/es/website-builders/blogger/>.
- [8] Josep García, “Squarespace Opiniones (2023): ¿Tan bueno es?,” 2023. <https://www.tooltester.com/es/opiniones/squarespace/#pros-cons>.
- [9] Jose María Valero, “Medium iba a comerse el mundo editorial online, pero por ahora es poco más que un Blogger bonito,” 2023. <https://www.xataka.com/empresas-y-economia/medium-iba-a-comerse-mundo-editorial-online-ahora-no-que-blogger-bonito>.
- [10] “GoLand de JetBrains: más que solo un IDE de Go,” 2023. <https://www.jetbrains.com/es-es/go/>.
- [11] “HTML5 - Wikipedia, la enciclopedia libre,” 2023. <https://es.wikipedia.org/wiki/HTML5>.
- [12] “CSS - Wikipedia, la enciclopedia libre,” 2023. <https://es.wikipedia.org/wiki/CSS>.
- [13] “JavaScript Logo, symbol, meaning, history, PNG, brand,” 2023. <https://logos-world.net/javascript-logo/>.
- [14] “Diseño plano moderno de icono de archivo xml para web estilo simple | Vector Premium,” 2023. [https://www.freepik.es/vector-premium/disenio-plano-moderno-icono-archivo-xml-web-estilo-simple\\_23279801.htm](https://www.freepik.es/vector-premium/disenio-plano-moderno-icono-archivo-xml-web-estilo-simple_23279801.htm).
- [15] “React,” 2023. <https://es.react.dev/>.



- [16] "Tailwind CSS - Rapidly build modern websites without ever leaving your HTML.," 2023.  
<https://tailwindcss.com/>.
- [17] "Node.js," 2023. <https://nodejs.org/es>.
- [18] "Express - Node.js web application framework," 2023. <https://expressjs.com/>.
- [19] "MySQL," 2023. <https://www.mysql.com/>.
- [20] "GitHub: Let's build from here · GitHub," 2023. <https://github.com/>.
- [21] "Salario para Desarrollador Full Stack en España - Salario Medio," 2023.  
<https://es.talent.com/salary?job=desarrollador+full+stack>.
- [22] "Calculadora coste empresa contratar un trabajador | Factorial," 2023.  
<https://factorialhr.es/calculadora-coste-trabajador>.
- [23] "¿Cuánto cuesta un asesor legal? Precios para 2023," 2023.  
<https://www.cronoshare.com/cuanto-cuesta/asesor-legal>.
- [24] "Hosting desde 1 €/mes » Alojamiento web profesional," 2023.  
<https://www.ionos.es/alojamiento/alojamiento-web>.
- [25] "Hostinger: Mucho Más Que Alojamiento Web. Tres Dos Online," 2023.  
[https://www.hostinger.es/?ppc\\_campaign=google\\_search\\_brand&bidkw=hostinger&gclid=EAlaIQobChMI4fXy-eDK\\_wIV0u7mCh3vpw2OEAAAYASAAEgK\\_yPD\\_BwE](https://www.hostinger.es/?ppc_campaign=google_search_brand&bidkw=hostinger&gclid=EAlaIQobChMI4fXy-eDK_wIV0u7mCh3vpw2OEAAAYASAAEgK_yPD_BwE).
- [26] "Hosting | Elige el alojamiento web adecuado para cualquier página - GoDaddy ES," 2023.  
<https://www.godaddy.com/es-es/hosting>.
- [27] "Shared Website Hosting - Easy and Affordable | HostGator.com," 2023.  
<https://www.hostgator.com/web-hosting>.
- [28] "Estadísticas de Cuota de Mercado y Uso de Shopify en 2023," 2023.  
<https://kinsta.com/es/cuota-de-mercado-de-shopify/>.
- [29] "¿Cuántas personas usan Wordpress en 2020? - Blog de Vleeko," 2020.  
<https://vleeko.net/cuantas-personas-usan-wordpress-en-2020/>.
- [30] "Estas 32 estadísticas demuestran que el blogging es un pilar del marketing digital," 2023.  
<https://blog.cliento.mx/estadisticas-blogging-pilar-marketing-digital>.
- [31] "(24) ¿Por qué deberías empezar a escribir tu blog en Medium? | LinkedIn," 2023.  
<https://www.linkedin.com/pulse/por-qué-deberías-empezar-escribir-tu-blog-en-medium-besné-eseverri/?originalSubdomain=es>.
- [32] "Gana dinero con la monetización de tu sitio web - Google AdSense," 2023.  
<https://adsense.google.com/start/#calculator>.
- [33] "Bootstrap Icons · Official open source SVG icon library for Bootstrap," 2023.  
<https://icons.getbootstrap.com/>.



- [34] “Modelo cascada, qué es y cuándo conviene usarlo,” 2023.  
<https://blog.ganttpro.com/es/metodologia-de-cascada/>.
- [35] “Patrones de diseño: Composición en React » CloudAPPi,” 2023.  
<https://cloudappi.net/patrones-de-diseno-composicion-en-react/>.
- [36] “REGLAMENTO (UE) 2016/ 679 DEL PARLAMENTO EUROPEO Y DEL CONSEJO - de 27 de abril de 2016 - relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga.”