




TRABAJO FIN DE CICLO FP IES JUAN DE HERRERA

Antonio Monzó Ansuátegui





Para mis padres que nunca me entienden cuando hablo de código y mis cachivaches.

Para Javi que cualquier parecido es cuestión de genética.

Para Adriana por ser el mejor patito que me podía presentar la vida.

Para César por darme una oportunidad en un mundo cada día mas complicado.

ÍNDICE

RESUMEN.....	3
English Translation.....	4
JUSTIFICACIÓN DEL PROYECTO	5
PLANIFICACIÓN DE LA EJECUCIÓN DEL PROYECTO	6
3.1. Fases y planificación temporal	6
3.2. Recursos materiales y humanos	9
3.3. Valoración económica del proyecto	11
3.4. Identificación de riesgos y plan de prevención	12
OBJETIVOS DEL PROYECTO	13
DESARROLLO.....	15
5.a. Revisión bibliográfica y fundamentación teórica	15
5.b. Materiales y métodos.....	16
5.c. Resultados y análisis	17
CONCLUSIONES.....	18
LÍNEAS DE INVESTIGACIÓN FUTURAS.....	19
BIBLIOGRAFÍA	21
8.a. Referencias	21
8.b. Bibliografía complementaria	21

1. RESUMEN

Este proyecto consiste en el desarrollo de una plataforma web de comercio electrónico, también conocida coloquialmente como *e-commerce*, centrada en la venta de libros y otros productos relacionados con el ámbito cultural y literario, denominada “**Los Círculos de Atenea**”. El objetivo principal es ofrecer una experiencia de compra intuitiva, segura y eficiente, tanto para los usuarios finales como para los administradores del sistema.

El proyecto ha sido desarrollado utilizando las tecnologías web aprendidas durante el ciclo formativo de grado superior. Para el apartado *frontend* se han empleado **HTML, CSS y JavaScript**, permitiendo la creación de una interfaz visual atractiva y dinámica. En el *backend* se ha trabajado con **PHP y MySQL**, encargándose de la lógica del servidor, la gestión de usuarios y el almacenamiento de datos. Asimismo, se han utilizado herramientas como **Git y GitHub** para el control de versiones, **Visual Studio Code** como entorno de desarrollo y **Hostinger** para el despliegue y alojamiento del proyecto en internet.

La aplicación sigue el modelo de arquitectura **MVC (Modelo-Vista-Controlador)**, con un sistema de rutas propio que mejora la organización del código y facilita su mantenimiento y escalabilidad. Entre sus funcionalidades principales se incluyen el registro e inicio de sesión de usuarios, la navegación por el catálogo de productos, la gestión de un carrito de compra persistente mediante **LocalStorage**, la realización de pedidos y la consulta del historial de compras.

Además, el sistema incorpora un panel de administración desde el cual se pueden gestionar usuarios, productos, stock y pedidos. Se ha prestado especial atención a la seguridad, implementando cifrado de contraseñas, validaciones en el servidor y control de accesos por roles. El resultado es una aplicación web funcional, robusta y preparada para futuras ampliaciones.

English Translation

This project consists of the development of an e-commerce web platform, focused on the sale of books and other products related to the cultural and literary field, called “**Los Círculos de Atenea.**” The main objective is to offer an intuitive, secure, and efficient shopping experience for both end users and system administrators.

The project has been developed using the web technologies learned during the higher vocational training program. For the frontend, **HTML, CSS, and JavaScript** have been used, enabling the creation of an attractive and dynamic visual interface. On the backend, **PHP and MySQL** have been employed, handling server-side logic, user management, and data storage. In addition, tools such as **Git and GitHub** have been used for version control, **Visual Studio Code** as the development environment, and **Hostinger** for the deployment and hosting of the project on the internet.

The application follows the **MVC (Model-View-Controller)** architectural pattern, with its own routing system that improves code organization and facilitates maintenance and scalability. Its main features include user registration and login, browsing the product catalog, managing a persistent shopping cart through **LocalStorage**, placing orders, and viewing purchase history.

Furthermore, the system includes an administration panel from which users, products, stock, and orders can be managed. Special attention has been paid to security by implementing **password encryption**, server-side validations, and role-based access control. The result is a functional, robust web application prepared for future expansions.

2. JUSTIFICACIÓN DEL PROYECTO

La elección de este proyecto nace de la necesidad de aplicar de forma práctica y global los conocimientos adquiridos a lo largo del ciclo formativo de **Desarrollo de Aplicaciones Web**, integrando en un único trabajo los contenidos de programación, bases de datos, diseño de interfaces, arquitectura de software y despliegue de aplicaciones.

El desarrollo de una plataforma de **comercio electrónico** se presenta como una opción especialmente adecuada, ya que este tipo de aplicaciones reproduce de forma fiel los retos técnicos y organizativos que se encuentran en el entorno profesional del desarrollo web. Una tienda en línea requiere la correcta gestión de usuarios, productos, pedidos y stock, así como la implementación de mecanismos de seguridad, control de accesos y persistencia de la información, aspectos todos ellos trabajados durante el ciclo formativo.

La temática elegida, centrada en la venta de libros y productos culturales, ofrece una estructura clara y bien definida que facilita la modelización de datos y la implementación de funcionalidades reales. El uso de catálogos, categorías y procesos de compra permite desarrollar un sistema coherente, comprensible y alineado con plataformas profesionales existentes, aportando realismo al proyecto.

Asimismo, este proyecto ha permitido simular un entorno de desarrollo similar al profesional, en el que ha sido necesario tomar decisiones técnicas fundamentadas, planificar el trabajo por fases y resolver problemas reales surgidos durante la implementación. La utilización de herramientas como Git y GitHub ha reforzado la importancia del control de versiones y del trabajo ordenado en proyectos de software.

Desde un punto de vista formativo y personal, este trabajo supone una **consolidación de competencias técnicas y organizativas**, así como una mejora de la autonomía y la capacidad de resolución de problemas. Además, el proyecto puede considerarse una **carta de presentación profesional**, ya que demuestra la capacidad para diseñar, desarrollar, documentar y desplegar una aplicación web completa, funcional y escalable, acorde a las exigencias del mercado laboral actual.

3. PLANIFICACIÓN DE LA EJECUCIÓN DEL PROYECTO

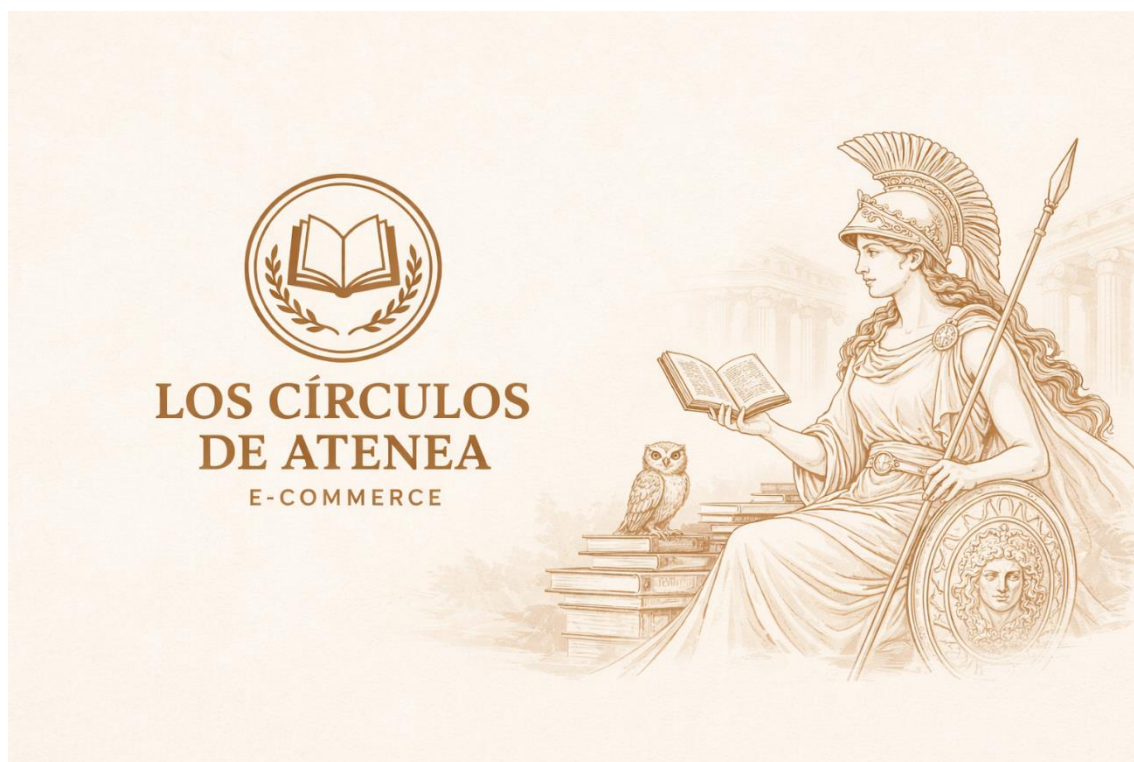


Ilustración 1 Imagen principal de Los Círculos de Atenea

La ejecución del proyecto se ha desarrollado a lo largo de un período aproximado de **tres meses**, comprendido entre los meses de **octubre y finales de diciembre**, siguiendo una planificación estructurada por fases. Esta organización ha permitido distribuir el trabajo de forma progresiva y ordenada, garantizando el cumplimiento de los objetivos establecidos dentro del calendario académico.

3.1. Fases y planificación temporal

- **Fase 1. Elección de tecnologías**

Periodo: Primera semana de octubre

En esta fase inicial se realizó un análisis de las distintas tecnologías disponibles y más adecuadas para el tipo de proyecto a desarrollar. Se decidió utilizar PHP y MySQL para el backend, y HTML, CSS y JavaScript para el frontend, por ser tecnologías ampliamente utilizadas, bien documentadas y adaptadas al alcance del proyecto. Asimismo, se seleccionaron las herramientas de desarrollo y despliegue, como Visual Studio Code, Git y GitHub para el control de versiones, y un servicio de hosting con

soporte para PHP y MySQL. Esta fase fue clave para sentar las bases técnicas del proyecto y garantizar su viabilidad.

- **Fase 2. Análisis y diseño**

Periodo: Segunda y tercera semana de octubre

Durante esta fase se definieron los requisitos funcionales y no funcionales del sistema, determinando qué funcionalidades debía ofrecer la aplicación y bajo qué condiciones debía operar. Se diseñó la arquitectura general del sistema siguiendo el modelo MVC, se planificó el sistema de rutas y se elaboró un diseño inicial de la base de datos. También se realizaron esquemas preliminares de las principales pantallas de la aplicación, con el objetivo de visualizar la estructura general del sistema antes de comenzar la implementación.

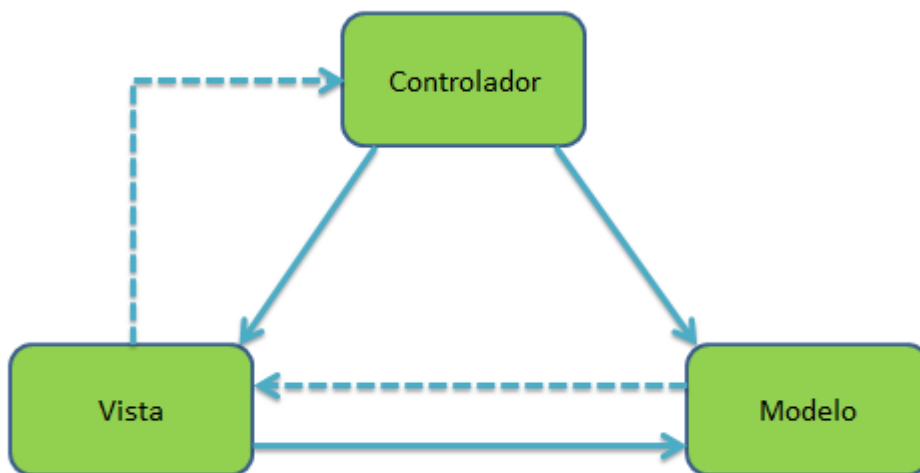


Ilustración 2 Esquema de la Arquitectura MVC (Modelo-Vista-Controlador)

- **Fase 3. Implementación de la base de datos**

Periodo: Cuarta semana de octubre

En esta fase se procedió a la creación de la base de datos en MySQL, definiendo las tablas necesarias, sus relaciones y las restricciones de integridad. Se realizaron pruebas de inserción y consulta de datos para verificar el correcto funcionamiento del modelo de datos y asegurar que la estructura era adecuada para soportar las operaciones principales de la aplicación.

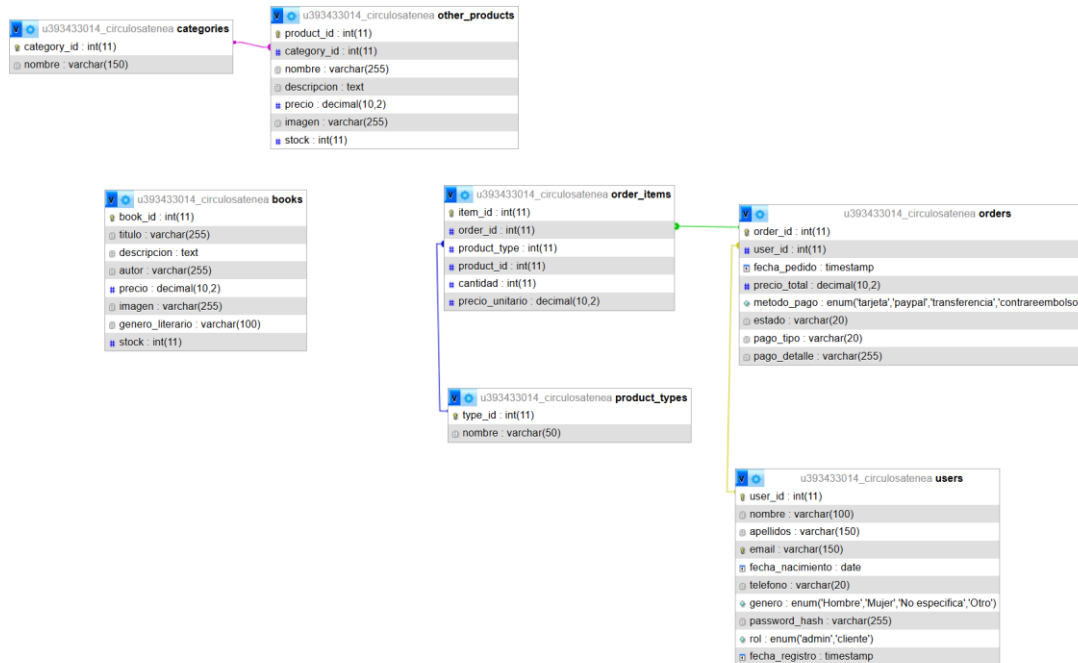


Ilustración 3 Base de datos

- **Fase 4. Desarrollo del frontend**

Periodo: Noviembre (semanas 1 a 3)

Esta fase estuvo centrada en la construcción de la interfaz de usuario. Se desarrollaron las distintas vistas de la aplicación utilizando HTML, CSS y JavaScript, prestando especial atención a la usabilidad, al diseño responsive y a la coherencia visual del conjunto. Se implementaron elementos como el catálogo de productos, las páginas de detalle, el carrito de compra y los formularios de registro e inicio de sesión, dejando preparada la base visual sobre la que posteriormente se integraría la lógica del servidor.

- **Fase 5. Desarrollo del backend**

Periodo: Finales de noviembre y primera quincena de diciembre

Durante esta fase se implementó toda la lógica de negocio del sistema en PHP. Se desarrollaron los controladores y modelos correspondientes, se gestionaron las sesiones de usuario, se implementaron los sistemas de autenticación y control de accesos por roles y se creó el panel de administración para la gestión de usuarios, productos, stock y pedidos. Esta fase constituyó el núcleo funcional del proyecto.

- **Fase 6. Pruebas y corrección de errores**

Periodo: Segunda quincena de diciembre

Una vez implementadas todas las funcionalidades principales, se realizó una fase intensiva de pruebas funcionales. Se verificó el correcto funcionamiento de cada módulo del sistema y de la interacción entre ellos, detectando y corrigiendo errores tanto a nivel de lógica como de interfaz. Esta fase fue fundamental para mejorar la estabilidad y fiabilidad del sistema.

- **Fase 7. Documentación y preparación para la entrega**

Periodo: Finales de diciembre

En la fase final se elaboró la documentación completa del proyecto, se revisó el código y se preparó el material necesario para la entrega y la presentación. También se realizaron los últimos ajustes estéticos y funcionales antes de dar el proyecto por finalizado.

3.2. Recursos materiales y humanos

Para la correcta ejecución del proyecto fue necesaria la utilización de una serie de recursos humanos, materiales y software, que permitieron cubrir todas las fases del desarrollo, desde el análisis inicial hasta el despliegue final de la aplicación.

Recursos humanos:

El proyecto ha sido desarrollado íntegramente por un único alumno, que asumió todas las responsabilidades asociadas al ciclo de vida del software. Entre estas tareas se incluyen el análisis de requisitos, el diseño de la arquitectura, la implementación del frontend y backend, la gestión de la base de datos, la realización de pruebas, la documentación del proyecto y el despliegue en el entorno de hosting.

Este enfoque ha permitido adquirir una visión global del proceso completo de desarrollo de una aplicación web, fomentando la autonomía, la capacidad de organización y la toma de decisiones técnicas de forma independiente.

Recursos materiales y software:

Para el desarrollo del proyecto se utilizaron los siguientes recursos:

- Ordenador personal con sistema operativo compatible con herramientas de desarrollo web.
- Conexión a internet estable para el acceso a documentación técnica, repositorios y servicios de hosting.
- Visual Studio Code como entorno de desarrollo integrado.
- Git y GitHub para el control de versiones y la gestión del código fuente.
- Servidor web con soporte para PHP y MySQL.
- Servicio de hosting Hostinger, utilizado para el despliegue y la publicación del proyecto en internet.
- Herramientas FTP (FileZilla) para la transferencia de archivos entre el entorno local y el servidor.
- Herramientas de IA para la generación de imágenes así como practicas de Pair-programming en momentos de duda.



Ilustración 4 imagen de ejemplo creada por IA

La elección de estas herramientas se basó en su fiabilidad, amplia documentación y uso extendido en entornos profesionales.

3.3. Valoración económica del proyecto

Aunque el proyecto se ha desarrollado con fines académicos, resulta relevante realizar una **estimación económica aproximada** que permita contextualizar su valor en un entorno profesional real.

En primer lugar, se consideran los **costes directos de infraestructura** necesarios para el despliegue de la aplicación:

- Hosting y dominio: **120 € anuales**
- Herramientas software: **0 €**, ya que se han utilizado herramientas de código abierto o versiones gratuitas para uso educativo.

Coste directo estimado: 120 €

Por otro lado, si se realiza una valoración del proyecto en función del tiempo invertido, se estima que el desarrollo completo ha requerido aproximadamente **300 horas de trabajo**, incluyendo análisis, diseño, programación, pruebas y documentación. Asignando un coste medio conservador de **9 € por hora**, el coste de desarrollo sería:

$$300 \text{ horas} \times 9 \text{ €/hora} = \mathbf{2700 \text{ €}}$$

Sumando los costes de infraestructura, el **coste total aproximado del proyecto** en un entorno profesional ascendería a:

$$\mathbf{2700 \text{ €} + 120 \text{ €} = 2820 \text{ €}}$$

Esta estimación pone de manifiesto el valor real del proyecto y permite entender el esfuerzo técnico y temporal necesario para desarrollar una aplicación web de estas características.

3.4. Identificación de riesgos y plan de prevención

Durante la planificación y ejecución del proyecto se identificaron una serie de riesgos potenciales que podrían afectar tanto al desarrollo como al resultado final del sistema. Para cada uno de ellos se definieron medidas preventivas con el objetivo de minimizar su impacto y garantizar la correcta finalización del proyecto.

Riesgo 1: Pérdida de datos o errores graves en el código

Este riesgo incluye la posibilidad de corrupción de archivos, borrado accidental de información o introducción de errores críticos durante el desarrollo.

Medidas de prevención: Se utilizó un sistema de control de versiones mediante Git y GitHub, permitiendo mantener un historial completo de cambios y recuperar versiones anteriores del proyecto en caso de fallo. Además, se realizaron copias de seguridad periódicas tanto del código como de la base de datos.

Riesgo 2: Retrasos en el desarrollo por dificultades técnicas o falta de tiempo

Dado que el proyecto se desarrolló en paralelo a otras actividades académicas, existía el riesgo de no poder cumplir los plazos previstos.

Medidas de prevención: Se estableció una planificación por fases con objetivos intermedios, lo que permitió llevar un seguimiento continuo del progreso y redistribuir el trabajo cuando fue necesario.

Riesgo 3: Fallos de seguridad en la aplicación

Las aplicaciones web son especialmente sensibles a vulnerabilidades como inyecciones SQL, accesos no autorizados o exposición de datos sensibles.

Medidas de prevención: Se implementaron medidas como el cifrado de contraseñas, validaciones de datos en el servidor, control de accesos por roles y revisión del código para evitar vulnerabilidades comunes.

Riesgo 4: Problemas de rendimiento o estabilidad del sistema

El crecimiento del número de productos, usuarios o pedidos podría afectar al rendimiento de la aplicación.

Medidas de prevención: Se diseñó una estructura de base de datos optimizada, se aplicaron buenas prácticas en el acceso a datos y se realizaron pruebas de carga básica para comprobar el comportamiento general del sistema.

Riesgo 5: Problemas en el despliegue o configuración del entorno de hosting

Podrían surgir incompatibilidades entre el entorno de desarrollo y el entorno de producción.

Medidas de prevención: Se realizaron pruebas previas en el servidor de hosting y se verificó la compatibilidad de versiones de PHP y MySQL antes del despliegue final.

Gracias a la identificación temprana de estos riesgos y a la aplicación de las medidas preventivas descritas, fue posible reducir significativamente la probabilidad de incidencias graves y asegurar el correcto desarrollo y funcionamiento del proyecto.

4. OBJETIVOS DEL PROYECTO

El objetivo principal de este proyecto es **diseñar y desarrollar una aplicación web de comercio electrónico plenamente funcional**, aplicando de forma integrada los conocimientos adquiridos durante el ciclo formativo de **Desarrollo de Aplicaciones Web**.

De manera específica, se establecen los siguientes objetivos:

- Diseñar y desarrollar una plataforma web siguiendo el modelo de arquitectura **MVC (Modelo-Vista-Controlador)**, que facilite la organización, mantenimiento y escalabilidad del código.

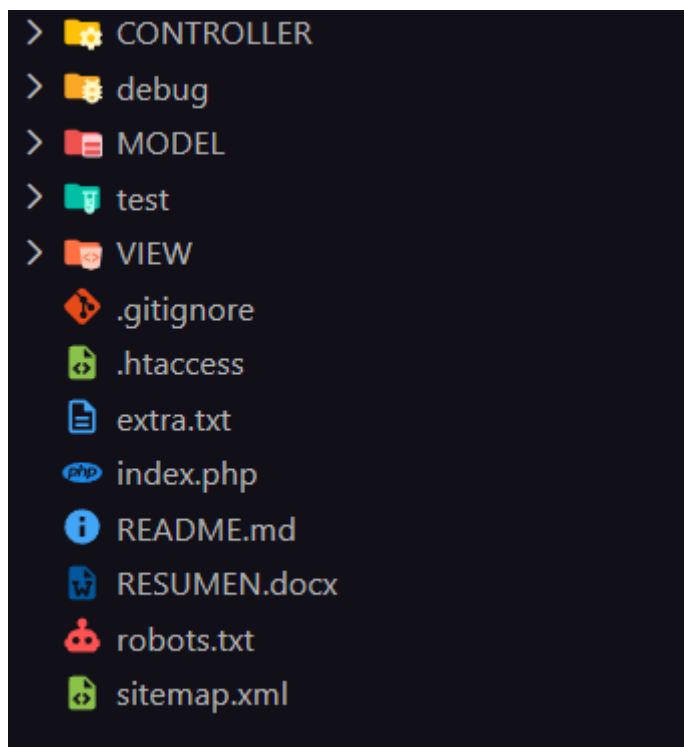


Ilustración 5 Organización del proyecto mediante MVC

- Implementar un sistema de **registro e inicio de sesión de usuarios**, garantizando la seguridad mediante cifrado de contraseñas y validaciones en el servidor.
- Desarrollar un **catálogo de productos** estructurado por categorías, permitiendo la consulta detallada de los mismos.



Ilustración 6, 7, y 8 Visualización de a tienda y catálogos



- Implementar un **carrito de compra persistente** utilizando **LocalStorage**, que mantenga la información del usuario entre sesiones.
- Gestionar el **proceso completo de realización de pedidos**, desde la selección de productos hasta su confirmación y registro en el sistema.
- Desarrollar un **panel de administración** que permita la gestión de usuarios, productos, stock y pedidos.

- Implementar un sistema de **control de accesos por roles**, diferenciando entre usuarios y administradores.
- Garantizar el **correcto funcionamiento de la aplicación** mediante la realización de pruebas funcionales y verificaciones del sistema.

5. DESARROLLO

5.a. Revisión bibliográfica y fundamentación teórica

Antes de iniciar el desarrollo de la aplicación, se realizó una revisión de conceptos y tecnologías relacionados con el **desarrollo de aplicaciones web** y el **comercio electrónico**. Se analizaron distintas plataformas de venta en línea ampliamente utilizadas con el objetivo de comprender su estructura general, su organización funcional y la experiencia de usuario que ofrecen.

En particular, se tomaron como referencia tiendas en línea consolidadas como **Casa del Libro** y **FNAC**, por tratarse de plataformas muy próximas a la temática del proyecto. El análisis de estas permitió identificar patrones comunes como la organización por categorías, los sistemas de búsqueda, el proceso de compra y la gestión del carrito.

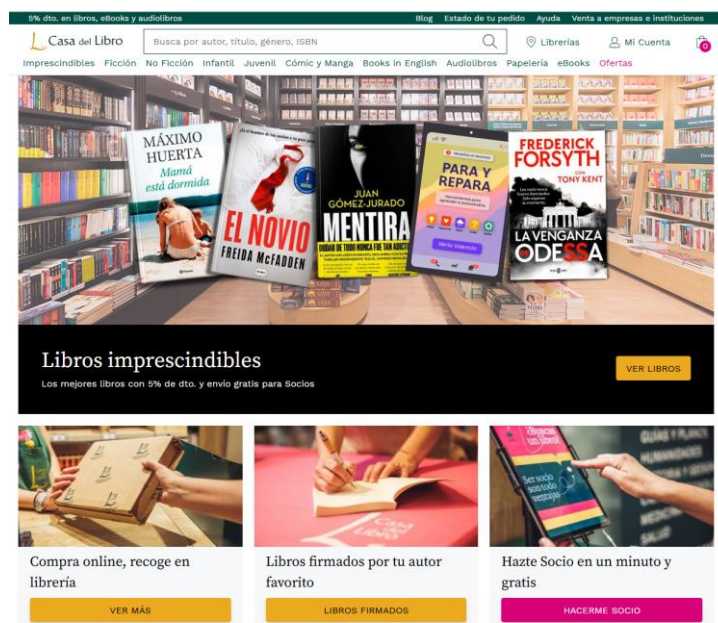


Ilustración 7 Imagen de referencia Casa del Libro

Asimismo, se revisaron los principios del modelo de arquitectura **MVC (Modelo-Vista-Controlador)**, así como diversas buenas prácticas de desarrollo web, entre ellas la separación de responsabilidades, la modularidad del código, la reutilización de componentes y la importancia de la seguridad en aplicaciones web.

También se tuvieron en cuenta conceptos fundamentales relacionados con la **gestión de bases de datos relacionales**, la **autenticación de usuarios**, el **control de accesos por roles** y la protección de la información sensible, todos ellos elementos esenciales en cualquier plataforma de comercio electrónico moderna.

5.b. Materiales y métodos

Para la realización del proyecto se utilizaron los siguientes recursos y tecnologías:

- **Lenguajes de marcado y programación:** HTML, CSS, JavaScript y PHP.
- **Base de datos:** MySQL.
- **Herramientas de desarrollo:** Visual Studio Code, Git y GitHub.
- **Servidor y hosting:** Hostinger con soporte para PHP y MySQL.
- **Herramientas de despliegue:** FileZilla para la transferencia de archivos.

La metodología empleada fue un **desarrollo iterativo e incremental**, avanzando por módulos y funcionalidades. Cada parte de la aplicación fue diseñada, implementada y probada de forma independiente antes de continuar con la siguiente, lo que permitió detectar errores de manera temprana y mejorar progresivamente la estabilidad del sistema.

Aunque se trabajó de forma iterativa durante todo el proceso, se reservó una fase específica para la **realización de pruebas globales y corrección de errores**, con el fin de garantizar el correcto funcionamiento conjunto de todos los componentes del sistema.



Ilustración 8 Stack tecnológico utilizado en el desarrollo del proyecto

5.c. Resultados y análisis

Como resultado del desarrollo, se ha obtenido una **aplicación web de comercio electrónico plenamente funcional**, que cumple con los objetivos planteados inicialmente. La plataforma permite la gestión completa de usuarios, productos, stock y pedidos, así como la administración del sistema mediante un panel exclusivo para administradores.

El uso del modelo de arquitectura MVC ha permitido mantener una **estructura de código clara, modular y organizada**, facilitando tanto el mantenimiento como futuras ampliaciones del sistema. El sistema de rutas propio y la separación de responsabilidades entre modelos, vistas y controladores han contribuido de forma notable a la escalabilidad del proyecto.

Las pruebas realizadas confirman el **correcto funcionamiento de las principales funcionalidades**: registro e inicio de sesión, gestión del carrito persistente, realización de pedidos y administración del sistema. Asimismo, se ha verificado la correcta implementación de las medidas de seguridad, especialmente en los procesos de autenticación, control de accesos y protección de datos sensibles.

6. CONCLUSIONES

El desarrollo del proyecto “**Los Círculos de Atenea**” ha supuesto la aplicación práctica e integrada de los conocimientos adquiridos durante el ciclo formativo de **Desarrollo de Aplicaciones Web**, permitiendo abordar un proyecto completo desde su fase inicial de análisis y diseño hasta su implementación, pruebas y despliegue final.

A lo largo del trabajo se ha conseguido diseñar e implementar una **plataforma de comercio electrónico funcional, estable y segura**, capaz de gestionar de forma eficiente usuarios, productos, stock y pedidos, y de ofrecer una experiencia de uso clara e intuitiva tanto para clientes como para administradores del sistema.

Uno de los aspectos más relevantes del proyecto ha sido la adopción del **modelo de arquitectura MVC**, que ha permitido mantener una estructura de código organizada, modular y fácilmente mantenible. Este enfoque ha facilitado la separación de responsabilidades, mejorando tanto la legibilidad del código como la posibilidad de ampliación futura del sistema.

Desde el punto de vista técnico, el proyecto ha servido para consolidar conocimientos en **PHP, JavaScript, bases de datos MySQL y tecnologías frontend**, así como en el uso de herramientas profesionales como Git y GitHub para el control de versiones y la gestión del código fuente. Además, se ha reforzado la importancia de aplicar **buenas prácticas de desarrollo y seguridad**, como el cifrado de contraseñas, la validación de datos en el servidor y el control de accesos por roles.

En cuanto a la metodología de trabajo, el desarrollo iterativo ha permitido detectar y corregir errores de forma progresiva, mejorando la calidad final del producto. Asimismo, la fase de pruebas ha sido fundamental para verificar la estabilidad del sistema y asegurar el correcto funcionamiento conjunto de todos sus módulos.

Desde un punto de vista formativo y personal, este proyecto ha supuesto una experiencia muy valiosa, ya que ha permitido adquirir una **visión global del proceso completo de desarrollo de una aplicación web real**, similar a la que se encuentra en un entorno profesional.

En conclusión, los objetivos planteados al inicio del proyecto se han cumplido satisfactoriamente, obteniendo como resultado una aplicación sólida, bien estructurada, ampliable y preparada para evolucionar hacia un entorno de producción real.

7. LÍNEAS DE INVESTIGACIÓN FUTURAS

Aunque la aplicación desarrollada cumple con todos los objetivos planteados y es plenamente funcional, el proyecto ofrece un amplio margen de evolución tanto a nivel funcional como técnico. Las siguientes líneas de desarrollo permitirían transformar la plataforma en un sistema de comercio electrónico mucho más completo, robusto y cercano a un entorno de producción real.

Una de las principales mejoras sería la **integración de pasarelas de pago reales**, como PayPal, Stripe o Redsys, permitiendo gestionar transacciones económicas completas con todas las garantías de seguridad y trazabilidad. Esto implicaría también la implementación de sistemas de validación de pagos, gestión de devoluciones y control de fraudes.

Otra línea importante de evolución sería la **mejora del sistema de gestión de pedidos y envíos**, incorporando múltiples estados (pendiente, en preparación, enviado, entregado, cancelado, etc.), así como un sistema de notificaciones automáticas por correo electrónico tanto para el cliente como para el administrador del sistema.

Desde el punto de vista de la experiencia de usuario, podría incorporarse un **sistema de valoraciones, reseñas y comentarios**, así como un **motor de recomendaciones personalizadas** basado en el historial de compras, productos visitados o preferencias del usuario, lo que aumentaría notablemente la usabilidad y el potencial comercial de la plataforma.

A nivel técnico, una línea de mejora relevante sería la **optimización del rendimiento y la escalabilidad**, mediante la implementación de sistemas de caché, optimización de consultas a la base de datos, paginación avanzada de resultados y posible migración a una arquitectura más distribuida.

También podría reforzarse el apartado de **seguridad**, incorporando sistemas avanzados de protección contra ataques habituales en aplicaciones web, como inyecciones SQL, XSS, CSRF

o fuerza bruta, así como la implementación de autenticación en dos factores y registros de auditoría de acciones sensibles.

Otra posible línea de evolución sería el desarrollo de una **API REST** que permita el acceso a los datos del sistema desde aplicaciones externas, como aplicaciones móviles o sistemas de terceros, ampliando el ecosistema del proyecto.

Finalmente, podría estudiarse la **internacionalización de la plataforma**, permitiendo el uso en varios idiomas y monedas, así como la adaptación a diferentes normativas legales en función del país de uso.

Todas estas líneas de investigación y desarrollo permitirían convertir el proyecto en una plataforma de comercio electrónico profesional, escalable y preparada para su explotación en un entorno real.

8. BIBLIOGRAFÍA

8.a. Referencias

- Martin, R. C. (2009). *Código limpio (Clean Code): Manual de estilo para el desarrollo ágil de software*. Prentice Hall.
- Documentación oficial de PHP. <https://www.php.net>
- Documentación oficial de MySQL. <https://dev.mysql.com/doc/>
- MDN Web Docs (HTML, CSS, JavaScript). <https://developer.mozilla.org>

8.b. Bibliografía complementaria

Plataformas y documentación técnica:

- Sitio oficial de Git. <https://git-scm.com>
- Sitio oficial de GitHub. <https://github.com>
- Documentación de Hostinger. <https://www.hostinger.es/tutoriales>
- Documentación de Visual Studio Code. <https://code.visualstudio.com/docs>

Canales de YouTube en español sobre programación:

- Píldoras Informáticas – <https://www.youtube.com/@pildorasinformaticas>
- HolaMundo – <https://www.youtube.com/@HolaMundoDev>
- Programación ATS – <https://www.youtube.com/@ProgramacionATS>
- Fazt Code – <https://www.youtube.com/@FaztCode>
- Sergie Code – <https://www.youtube.com/@SergieCode>