

# TRABAJO DE FIN DE GRADO

## Tienda de Señales de Tráfico



Alonso Hernández Robles

Desarrollo de Aplicaciones Web

IES Francisco Ayala

2024/2025

Este documento describe el desarrollo de una aplicación web en forma de tienda online especializada en la venta de señales de tráfico. La plataforma ha sido construida utilizando tecnologías básicas de desarrollo web, como *HTML*, *CSS* y *JavaScript* para el *front-end*, y *PHP* junto con *MySQL* para el *back-end*. El sistema incluye funcionalidades como la compra y valoración de productos, gestión de usuarios, categorías, pedidos y productos. También se ha implementado una interfaz intuitiva para los usuarios y las vistas correspondientes para administradores, permitiendo una gestión eficiente de la tienda. El trabajo abarca desde el análisis de requisitos hasta la implementación y pruebas finales.

This document describes the development of a web application designed as an online store specialized in the sale of traffic signs. The platform was built using basic web development technologies, such as *HTML*, *CSS*, and *JavaScript* for the front-end, and *PHP* together with *MySQL* for the back-end. The system includes features such as product purchasing and rating, as well as user, category, order, and product management. An intuitive user interface has been implemented, along with corresponding views for administrators, allowing for efficient store management. The project covers all stages from requirements analysis to implementation and final testing.

### **GitHub:**

[https://github.com/AloncraftMC/TFG\\_DAW\\_AlonsoHernandezRobles](https://github.com/AloncraftMC/TFG_DAW_AlonsoHernandezRobles)

1. Justificación .....	5
1.1. Características generales .....	5
1.2. Restricciones generales .....	5
1.3. Aspectos a cubrir .....	6
1.4. Estudio de las prestaciones de la herramienta .....	6
2. Tecnología empleada .....	6
3. Requerimientos de hardware y software .....	7
4. Análisis y diseño .....	8
4.1. Diagrama de casos de uso .....	8
4.2. Diagrama de clases .....	9
4.3. Descripción de la base de datos .....	10
4.4. Modelo Entidad-Relación .....	10
4.5. Diseño Lógico .....	11
5. Implementación .....	12
5.1. Archivos Raíz .....	12
5.2. assets/ .....	12
5.3. controllers/ .....	13
5.4. database/tienda.sql .....	13
5.5. docs/ .....	13
5.6. helpers/Utils.php .....	14
5.7. js/ .....	14
5.8. lib/BaseDatos.php .....	15
5.9. logs/error.log .....	15
5.10. mails/ .....	15
5.11. models/ .....	15
5.12. vendor/ .....	16
5.13. views/ .....	16
6. Evaluación y prueba .....	19
7. Manual de estilos .....	20
7.1. Sketches .....	20
7.2. Criterios de accesibilidad .....	27
7.3. Criterios de usabilidad .....	27

7.4. Tipografía.....	28
7.5. Mapa de colores.....	28
7.6. Dispositivos/Vistas.....	29
8. Software utilizado.....	29
9. Mejoras posibles y aportaciones.....	30
9.1. Funcionalidad .....	30
9.2. Código .....	30
9.3. Información .....	30
10. Recuento de trabajo.....	30
10.1. Reporte general .....	30
10.2. Información y archivos .....	30

# 1. Justificación

## 1.1. Características generales

La aplicación consta de una tienda online dedicada a la venta de señales de tráfico, tanto como para señalización vial como para accesorios necesarios en vehículos o situaciones de emergencia, entre otros ejemplos.

Sin iniciar sesión, se pueden visualizar una serie de productos recomendados al cliente y ver por separado cada producto en su página correspondiente.

Un usuario de la aplicación, habiendo iniciado sesión, puede introducir productos al carrito de compra y realizar un pedido a una dirección concreta, mediante una pasarela de pago implementada con *PayPal*. No obstante, el usuario puede pagar con tarjeta de crédito o débito si así lo prefiere.

El usuario registrado que ha hecho algún pedido puede ver en una página una lista de sus pedidos realizados anteriormente, con diversos detalles informativos como fecha, hora, o ubicación aproximada en **Google Maps** del destino del pedido.

Un producto que ha sido comprado por el usuario actual en un pedido que ya esté enviado puede ser valorado, con una calificación de 1 a 5 estrellas más un comentario adicional que este usuario puede proporcionar.

Los datos del carrito de un usuario se guardan en una *cookie* (local) multiusuario que se elimina una vez han pasado 3 días sin actividad por la aplicación web.

Al iniciar sesión, el usuario puede elegir ser recordado con una *cookie* que dura 7 días.

Todos los usuarios pueden modificar sus datos personales en una página, donde figura información como nombre, apellidos, color personalizable de la interfaz, imagen de perfil, correo electrónico o contraseña.

Un usuario puede adoptar el rol administrador, el cual obtiene un poder superior sobre el usuario con rol convencional. Estos administradores pueden crear, editar y eliminar categorías, productos y otros usuarios a su antojo. También pueden confirmar, enviar o eliminar pedidos de usuarios. Por último, pueden eliminar cualquier valoración de cualquier usuario en un producto determinado.

Al realizarse acciones de interés al usuario como creación, modificación o eliminación de cuenta, o cambio en el estado de un pedido, la aplicación envía un correo electrónico al usuario notificando dichas acciones con detalle.

Se pueden buscar productos por coincidencias parciales en el nombre del producto o de su categoría, mediante un buscador que se encuentra en el *header* de la página.

Se pueden filtrar productos por categorías en la página principal.

## 1.2. Restricciones generales

Los administradores no pueden buscar categorías, productos, pedidos o usuarios en los paneles de administración.

### 1.3. Aspectos a cubrir

Se incluyen funciones como autenticación, carrito de compra, gestión de productos, pedidos y pagos. Todas estas funciones son propias de lo que se esperaría de una tienda a nivel fundamental. No se ha abordado la implementación de una *app* móvil ni el uso de tecnologías avanzadas como inteligencia artificial o *machine learning*.

### 1.4. Estudio de las prestaciones de la herramienta

Frente a otras soluciones de comercio electrónico como *Shopify* o *Prestashop*, esta herramienta está totalmente personalizada y permite controlar todo el código fuente, lo que la hace más flexible y didáctica para un entorno formativo.

## 2. Tecnología empleada

**HTML5:** Lenguaje de marcado de hipertexto. Básico para construir el armazón de las páginas web.

**CSS3:** Lenguaje de estilado. Sirve para realizar las hojas de estilo que definen cómo se comportan estilísticamente los elementos *HTML*.

**JavaScript:** Lenguaje de programación para otorgar funcionalidades necesarias en el cliente.

**PHP:** Lenguaje de programación para manejar toda la lógica del servidor, incluyendo manejo de sesiones, conexión con la base de datos y gestión de la lógica de negocio.

**MySQL:** Sistema Gestor de Base de Datos relacional ligado con **phpMyAdmin** al lenguaje *PHP* del servidor.

**Composer:** Sistema de gestión de paquetes de código *PHP* para manejar dependencias y librerías en el proyecto.

**PHPMailer:** Biblioteca de códigos para enviar correos electrónicos de forma segura y sencilla mediante código *PHP* desde el servidor web.

**GeoAPI España:** *API* para normalizar datos postales en *JSON* para todo el territorio español.

**Guzzlehttp:** Cliente *PHP* que facilita el envío de peticiones *HTTP* y hace trivial la integración con servicios web, en este caso se ha usado para integrar **PayPal**.

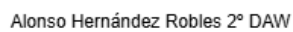
### 3. Requerimientos de hardware y software

Dependiendo de si la aplicación se ejecuta localmente o en remoto, se necesitará cumplir con unas especificaciones u otras.

Si la aplicación se ejecuta en remoto, el único hardware necesario será un dispositivo con acceso a internet y, como software, un navegador actualizado como *Chrome*, *Firefox* o *Edge*.

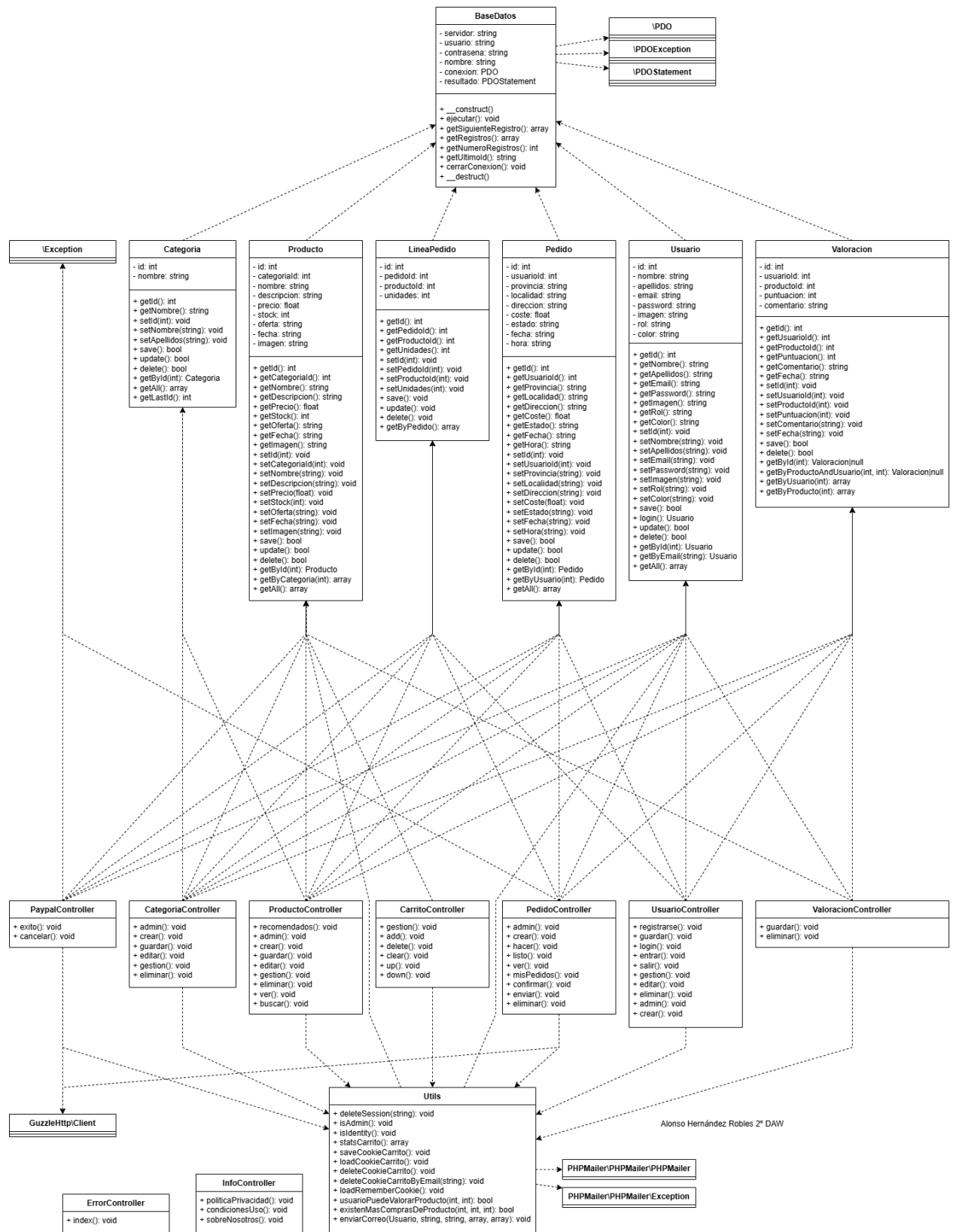
En cambio, si se opta por ejecutar localmente, también será necesario configurar un servidor local. Para ello, será necesario instalar **XAMPP**, un paquete público de servicios para iniciar dicho servidor de forma local en el mismo dispositivo que se ha mencionado anteriormente, y **Composer**, el gestor de dependencias y paquetes usado en este proyecto.

#### 4.1. Diagrama de casos de uso





## 4.2. Diagrama de clases

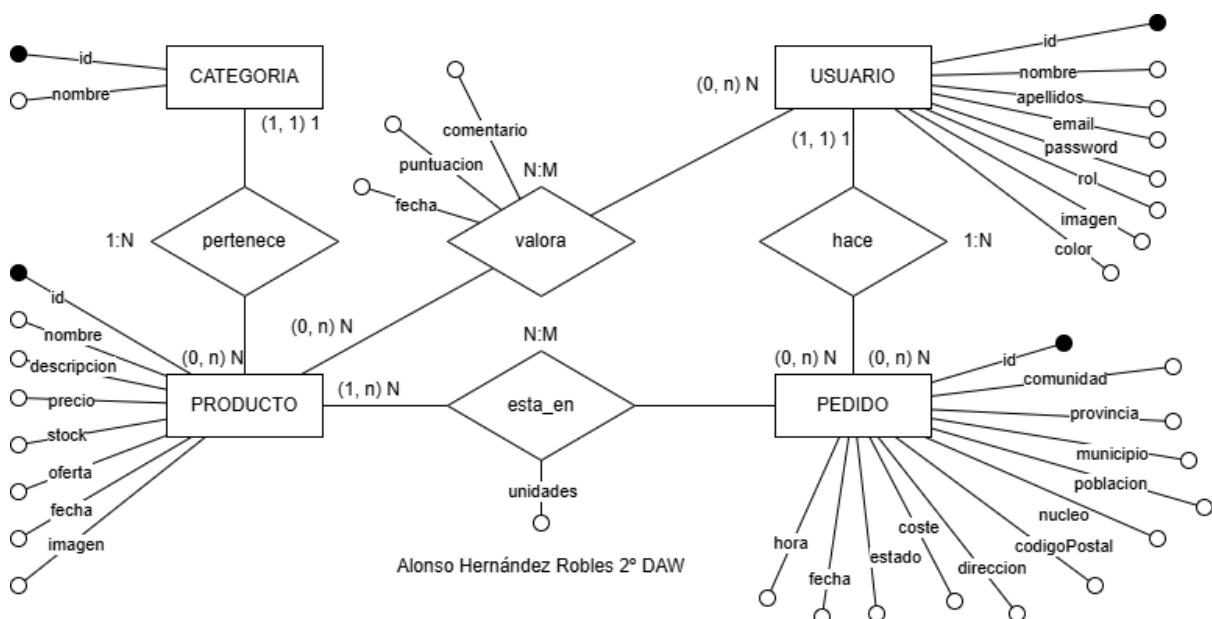


### 4.3. Descripción de la base de datos

Se precisa la necesidad de usar una base de datos con las siguientes características:

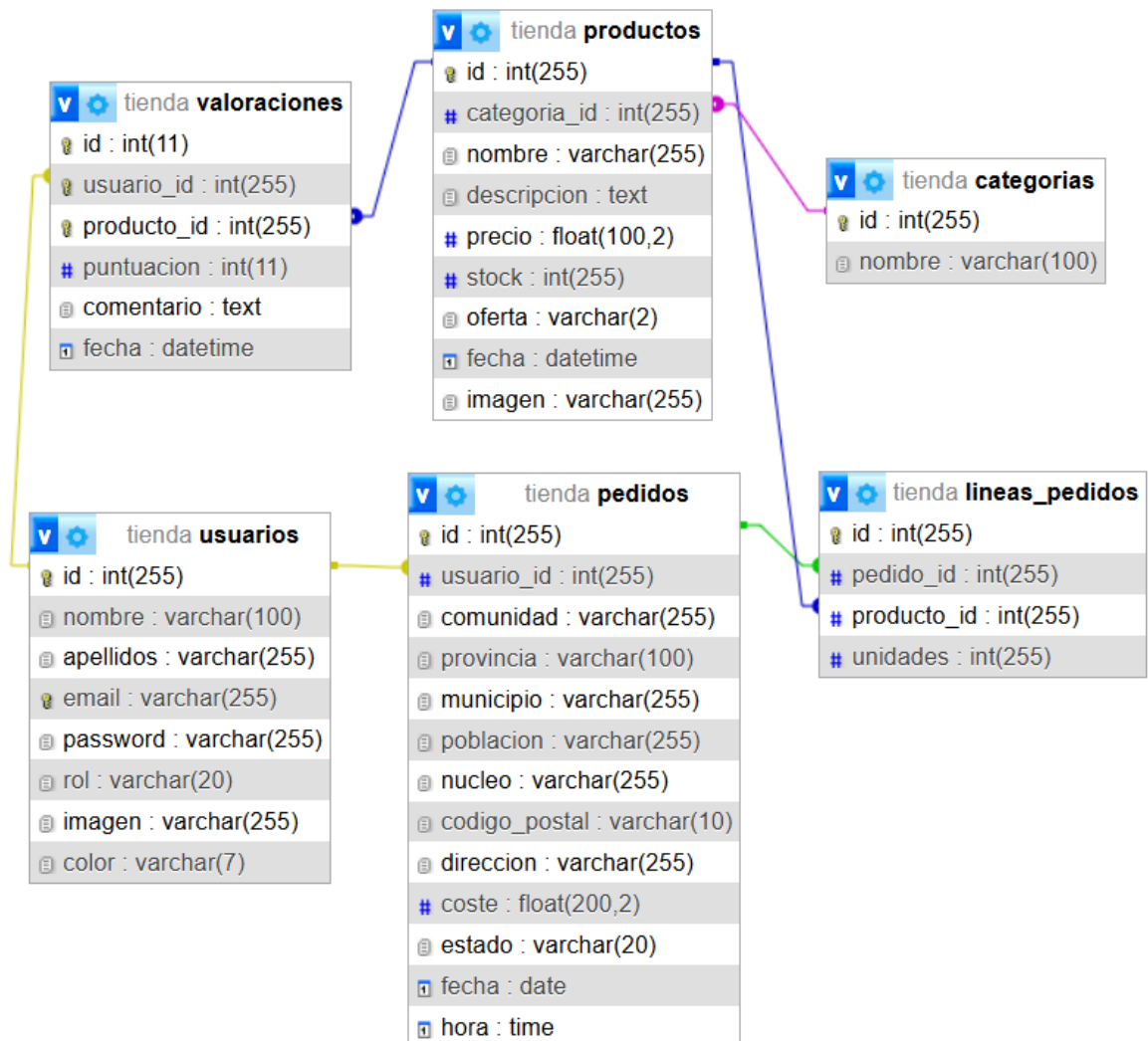
- **Categorías:** Serán los conjuntos de productos que contengan en común una tipología. De estas categorías, solamente será necesario establecer un nombre para cada una de ellas.
- **Productos:** Estos serán los artículos que se vendan en la tienda online, y cada uno tendrá una imagen, un nombre, una descripción, un precio y una cantidad en *stock*. También se desea conocer si el producto tiene algún porcentaje de descuento y la fecha en la que fue añadido/modificado por última vez.
- **Pedidos:** Serán cada una de las listas de productos que pidan los usuarios por la web y sean finalmente enviadas a los domicilios de estos. Por cada producto en el pedido, se desea saber las unidades que se han ordenado. En el pedido, se detallará la siguiente información: la comunidad autónoma de envío, la provincia, el municipio, la población, el núcleo urbano, el código postal, la dirección, el coste total del pedido, el estado del pedido en un momento determinado (será pendiente, confirmado o enviado) y la fecha y la hora a la que fue creado.
- **Usuarios:** Estos son las personas que usarán la aplicación web, y tendrán un rol de tanto convencionales como administradores. De ellos se quiere saber su nombre, apellidos y email. También se quiere almacenar una contraseña para garantizar un acceso correcto y seguro, además de una imagen de perfil y un color que el usuario podrá elegir para la personalización de la interfaz gráfica.
- Los usuarios que han comprado al menos una vez un producto determinado pueden valorarlo con una puntuación de 1 a 5 junto a un comentario, y dicha valoración es publicada en una fecha determinada.

### 4.4. Modelo Entidad-Relación



## 4.5. Diseño Lógico

- CATEGORIA(**id**, nombre)
- PRODUCTO(**id**, nombre, descripcion, precio, stock, oferta, fecha, imagen, categoria\_id)
  - categoria\_id: Clave foránea de CATEGORIA(id)
- PEDIDO(**id**, comunidad, provincia, municipio, poblacion, nucleo, codigo\_postal, direccion, coste, estado, fecha, hora, usuario\_id)
  - usuario\_id: Clave foránea de USUARIO(id)
- USUARIO(**id**, nombre, apellidos, email, password, rol, imagen, color)
- LINEA\_PEDIDO(**id**, unidades, producto\_id, pedido\_id)
  - producto\_id: Clave foránea de PRODUCTO(id)
  - pedido\_id: Clave foránea de PEDIDO(id)
- VALORACION(**id**, puntuacion, comentario, fecha, usuario\_id, producto\_id)
  - usuario\_id: Clave foránea de USUARIO(id)
  - producto\_id: Clave foránea de PRODUCTO(id)



## 5. Implementación

A continuación se detallará y explicará la estructura de archivos y directorios del proyecto.

### 5.1. Archivos Raíz

- **index.php**: Archivo principal de la aplicación. Se encarga de iniciar la sesión, cargar todas las clases de los controladores, el archivo de configuración, cargar la identidad del usuario, establecer el título de la página, renderizar el *header* y el *footer* y cargar la acción necesaria del controlador necesario.
- **config.php**: Archivo de configuración de variables globales de la aplicación. En él se requiere el *autoloader* de **Composer** (en **vendor/**) y se cargan las variables de entorno, y se definen constantes como la ruta base de la aplicación, la conexión a la base de datos, el controlador y acción por defecto, los elementos de la paginación y las credenciales tanto del servicio de *SMTP* de **Gmail** como de **PayPal Developer**.
- **autoloader.php**: Archivo que se encarga de cargar automáticamente las clases que se vayan a utilizar en la aplicación.
- **.htaccess**: Archivo de configuración de **Apache**. En él, se activa el módulo de *PHP mod\_rewrite*, se define la página de error 404 y cómo se redirigen las peticiones a la acción del controlador y se bloquea el acceso a carpetas y archivos sensibles o privados, como por ejemplo **.env** o **vendor/**.
- **.env**: Archivo donde se almacenan las variables de entorno privadas. En él, se encuentran las credenciales de la base de datos, del servicio *SMTP* de **Gmail** y de **PayPal Developer**. Funciona gracias a la biblioteca de **Composer** *vlucas/phpdotenv*. No disponible en **GitHub**.
- **.gitignore**: Archivo que contiene los archivos y carpetas que se deben ignorar al subir el proyecto a un repositorio de **Git**. En este caso, se ignora **.env** y la carpeta **vendor/**.
- **README.md**: Archivo con información de introducción al proyecto en formato *Markdown*.
- **composer.json**: Archivo con una lista de las dependencias necesarias de **Composer** para que funcione el proyecto. Contiene *PHPMailer* (necesario para enviar correos), *PHP Dotenv* (necesario para ocultar y cargar las variables de entorno y *GuzzleHttp* (necesario para hacer peticiones a la *API* de **PayPal**).
- **composer.lock**: Archivo con información acerca de las dependencias y subdependencias *PHP* relacionadas con **Composer**.

### 5.2. assets/

- **css/style.css**: Archivo *CSS* que contiene todos los estilos generales de la aplicación.
- **fonts/Rubik-Regular.ttf**: Fuente Rubik utilizada estilísticamente en la aplicación.
- **images/**: Esta carpeta contiene las imágenes estáticas inmutables que se utilizan en la aplicación, en su mayoría archivos *.svg*.
  - **uploads/**: Carpeta con imágenes subidas al servidor. Pueden ser:
    - **productos/**: Esta carpeta contiene las imágenes de los productos subidas por los usuarios administradores en la creación y edición de productos.

- **usuarios/**: Esta carpeta contiene las imágenes de perfil de los usuarios subidas por los usuarios tanto convencionales como administradores en la creación y edición de usuarios, además de en la gestión de usuarios convencionales.

### 5.3. controllers/

- **CarritoController.php**: Archivo con la lógica de negocio del carrito de compras. Tiene acciones para ver el carrito, vaciarlo, añadir o eliminar un producto e incrementar o decrementar en uno las unidades de un producto.
- **CategoriaController.php**: Archivo con la lógica de negocio de las categorías. Tiene acciones para crear, editar y eliminar categorías y mostrar la lista de categorías y el formulario de creación y edición de categorías.
- **ErrorController.php**: Archivo con la lógica de negocio del error 404. Tiene una sola acción, la llamada por el archivo **.htaccess** cuando no se encuentra una página, y muestra una vista con un mensaje de error.
- **InfoController.php**: Archivo con la lógica de negocio de las páginas informativas. Tiene acciones para mostrar las vistas de estas páginas, que son tres: Política de Privacidad, Condiciones de Uso y Sobre Nosotros.
- **PaypalController.php**: Archivo con la lógica de negocio de **PayPal**. Tiene acciones para crear el pedido en caso de éxito en el pago y para mostrar la vista que se requiere al cancelar el pago.
- **PedidoController.php**: Archivo con la lógica de negocio de los pedidos. Tiene acciones para crear, confirmar, enviar y eliminar pedidos y mostrar la lista de pedidos tanto global como para el usuario actual, el formulario de creación del pedido y la página inmediata después de pagar en la pasarela de pago de **PayPal**.
- **ProductoController.php**: Archivo con la lógica de negocio de los productos. Tiene acciones para ver, buscar, crear, editar y eliminar productos, mostrar la lista de productos y el formulario de creación y edición de productos y mostrar la página de inicio, es decir, los productos recomendados.
- **UsuarioController.php**: Archivo con la lógica de negocio de los usuarios. Tiene acciones para crear, editar y eliminar usuarios, mostrar la lista de usuarios y el formulario de registro, creación, inicio de sesión y edición de usuarios e iniciar y cerrar la sesión.
- **ValoracionController.php**: Archivo con la lógica de negocio de las valoraciones. Tiene acciones para crear y eliminar valoraciones.

### 5.4. database/tienda.sql

Archivo generado por **phpMyAdmin** listo para importar en un servidor en producción, con una base de datos con todas las categorías y todos los productos insertados, sin pedidos, sin valoraciones y con 5 usuarios de prueba; un administrador y 4 convencionales.

### 5.5. docs/

Carpeta con la documentación y los diagramas de las distintas fases de este proyecto, incluyendo esta misma documentación.

## 5.6. helpers/Utils.php

Clase de funciones útiles estáticas llamadas frecuentemente a lo largo de casi todas las clases del proyecto. Contiene métodos para eliminar slots en la variable superglobal `$_SESSION`, comprobar si el usuario es administrador o está *logueado*, devolver datos del carrito, guardar, cargar o eliminar la *cookie* del carrito, eliminar la *cookie* del carrito de un usuario en específico, cargar la *cookie* de “recuérdame”, comprobar si un usuario puede valorar un producto o si existen más un pedido en el que figura un mismo producto y para enviar correos.

## 5.7. js/

Carpeta con *scripts* de *JavaScript*, destinados a ser ejecutados en el cliente al descargar la página web una vez haya sido traída del servidor. Contiene los archivos:

- **actualizarPaginacion.js**: Sirve para reemplazar “?” por “&” en la *URL* que contenga el parámetro *pag* por *GET*, ya que al tener el archivo **.htaccess** configurado de tal manera, no es capaz de leer los parámetros si no es con “&”.
- **actualizarToken.js**: Sirve para reemplazar “?” por “&” en la *URL* que contenga el parámetro “*token*” y “*PayerID*” por *GET*, ya que al tener el archivo **.htaccess** configurado de tal manera, no es capaz de leer los parámetros si no es con “&”.
- **adminPopup.js**: Crea un *pop-up* que se muestra al cargar la página para advertir al usuario administrador sobre las consecuencias de eliminar categorías, productos, pedidos o usuarios.
- **ajusteImagenesProductos.js**: Ajusta las imágenes de productos en páginas de recomendados y derivadas.
- **ajusteImagenesTabla.js**: Ajusta las imágenes de productos en tablas. Aparece sobre todo en pedidos y administración de productos.
- **atencion.js**: Hace aparecer el aviso de atención en las páginas de información cuando se hace clic en el botón de "Atención".
- **carrusel.js**: Maneja la lógica del carrusel de categorías en la página principal.
- **extraerDatosGeoAPI.js**: *Script* para extraer datos de la **API GeoAPI España** y llenar los selectores de dirección.
- **generarPaletaColores.js**: Genera una paleta de colores a partir de un color base personalizado en formato *HEX*.
- **headerCombinado.js**: Contiene la lógica para manejar el menú de administración y el buscador en la barra de navegación.
- **previsualizarColores.js**: Permite previsualizar los colores seleccionados en un *input* de tipo color.
- **subirImagenCrear.js**: Contiene la carga de imágenes en un formulario de creación.
- **subirImagenEditar.js**: Contiene la carga de imágenes en un formulario de edición.
- **validarFormularioCrear.js**: Contiene la lógica para validar un formulario de creación.
- **validarFormularioEditar.js**: Contiene la lógica para validar un formulario de edición.



### 5.8. lib/BaseDatos.php

Clase que contiene la conexión a la base de datos usando el objeto *PDO*. Tiene un método universal para ejecutar consultas *SQL* y métodos para obtener registros, entre otros.

### 5.9. logs/error.log

Archivo vacío a priori en que quedarán registrados todos los errores relacionados con el envío de correos de la aplicación en el proceso de trabajo del *SMTP* de **Gmail** con *PHPMailer*.

### 5.10. mails/

Carpeta que contiene todas las vistas *HTML* que se enviarán a la bandeja de entrada de los usuarios cada que vez que se precise el envío de un correo electrónico.

- **pedido/**
  - **confirmar.html**: Correo enviado cuando el pedido ha pasado al estado de “Confirmado”.
  - **eliminar.html**: Correo enviado cuando el pedido ha sido eliminado.
  - **enviar.html**: Correo enviado cuando el pedido ha pasado al estado de “Enviado”.
  - **hacer.html**: Correo enviado cuando el pedido ha sido realizado.
- **usuario/**
  - **crear.html**: Correo enviado cuando un administrador le ha creado una cuenta al usuario.
  - **destruir.html**: Correo enviado cuando el usuario ha sido eliminado por un administrador.
  - **editar.html**: Correo enviado cuando el usuario ha sido editado por un administrador.
  - **eliminar.html**: Correo enviado cuando el usuario ha eliminado su cuenta.
  - **gestion.html**: Correo enviado cuando el usuario ha editado su cuenta.
  - **registro.html**: Correo enviado cuando el usuario ha creado su cuenta.
- **valoracion/**
  - **eliminar.html**: Correo enviado cuando la valoración ha sido eliminada.

### 5.11. models/

- **Categoria.php**: Clase que representa una categoría. Sus interacciones con la base de datos permiten guardar, editar y eliminar categorías, además de encontrar una categoría por su *id* o extraer todas las categorías.
- **LineaPedido.php**: Clase que representa una línea en un pedido. Sus interacciones con la base de datos permiten guardar y eliminar líneas de pedido, además de extraer todas las líneas de un pedido o relacionadas con un producto.
- **Pedido.php**: Clase que representa un pedido. Sus interacciones con la base de datos permiten guardar, editar y eliminar pedidos, además de encontrar un pedido por su *id*, encontrar todos los pedidos de un usuario o extraer todos los pedidos.
- **Producto.php**: Clase que representa una categoría. Sus interacciones con la base de datos permiten guardar, editar y eliminar productos, además de encontrar un producto

por su *id*, encontrar productos por categoría o coincidencias por nombre y/o categoría o extraer todos los productos.

- **Usuario.php**: Clase que representa un usuario. Sus interacciones con la base de datos permiten guardar, editar y eliminar usuarios, además de comprobar si el inicio de sesión es correcto, encontrar un usuario por su *id* o email, extraer la posición general de un usuario en la base de datos o extraer todos los usuarios.
- **Valoracion.php**: Clase que representa una valoración. Sus interacciones con la base de datos permiten guardar y eliminar valoraciones, además de encontrar una valoración por su producto y usuario o extraer todas las valoraciones de un usuario o de un producto.

### 5.12. vendor/

Carpeta generada por **Composer** que contiene todos los módulos necesarios para el correcto funcionamiento de las bibliotecas, librerías y dependencias instaladas en el proyecto. No disponible en **GitHub**. En este caso, contiene las librerías:

- **guzzlehttp/guzzle**: Cliente que permite la fácil interacción con **PayPal Developer** mediante solicitudes *HTTP*.
- **phpmailer/phpmailer**: Librería que permite enviar correos electrónicos mediante un *SMTP*, en este caso ha sido configurado mediante la creación de códigos de aplicación de **Gmail** y unido al correo [señalesdetrafico.store@gmail.com](mailto:señalesdetrafico.store@gmail.com).
- **vlucas/phpdotenv**: Librería que permite la carga de variables de entorno en el proyecto extrayéndolas del archivo **.env** y colocándolas en la variable superglobal nativa de *PHP* `$_ENV`. Gracias a este complemento, la información sensible como credenciales de la base de datos y las contraseñas de la aplicación de **Gmail** y **PayPal Developer** queda encapsulada y, en consecuencia, oculta.

### 5.13. views/

- **carrito/**
  - **gestion.php**: Vista de gestión del carrito de la compra. Muestra los productos en el carrito, permite modificar cantidades y eliminar productos. También muestra el total del carrito y permite vaciarlo o proceder a la compra.
- **categoria/**
  - **admin.php**: Vista de administración de categorías. Usa paginación para mostrar las categorías en la base de datos. Permite crear, editar y eliminar categorías.
  - **crear.php**: Vista para crear una nueva categoría. Muestra un formulario para ingresar el nombre de la categoría. Valida el nombre y muestra mensajes de error si es necesario.
  - **editar.php**: Vista para editar una categoría. Muestra un formulario con el nombre de la categoría y un botón un guardar los cambios. Si se produce un error, se muestra un mensaje de error.
- **info/**
  - **condiciones-uso.php**: Vista de las condiciones de uso. Hace uso del botón de aviso para advertir al usuario que el sitio web no es real.



- **politica-privacidad.php**: Vista de la política de privacidad. Hace uso del botón de aviso para advertir al usuario que el sitio web no es real.
- **sobre-nosotros.php**: Vista de sobre nosotros. Hace uso del botón de aviso para advertir al usuario que el sitio web no es real.
- **layout/**
  - **footer.php**: Pie de página. Contiene los enlaces a las páginas de información.
  - **header.php**: Encabezado de la aplicación. Contiene el logo, el título de la tienda, los botones de navegación y el carrito. También incluye el *script* para el menú de administración si el usuario es administrado, y el *script* para el *pop-up* de administración si es la primera vez que accede. También tiene el *script* para el buscador de productos.
- **paypal/**
  - **cargando.php**: Vista de carga para redirigir a la página de “Pedido Listo” tras realizar con éxito el pago en **PayPal**.
- **pedido/**
  - **admin.php**: Vista de administración de pedidos. Si no hay pedidos, se muestra un mensaje informativo, si hay pedidos, se muestran en una tabla con paginación. Cada fila de la tabla tiene un botón para ver el pedido, otro para eliminarlo y otro para confirmar o enviar el pedido. Tiene un botón para ir a la ubicación aproximada en **Google Maps** por cada pedido.
  - **crear.php**: Vista para crear un pedido. Hace uso de una *API GeoAPI España*, que es una *API* de geolocalización, para obtener los datos de la dirección.
  - **listo.php**: Vista que se muestra cuando el pedido ha sido realizado correctamente.
  - **mis-pedidos.php**: Vista de los pedidos del usuario. Tiene un botón para ir a la ubicación aproximada en **Google Maps** por cada pedido.
  - **ver.php**: Vista para ver un pedido específico. Muestra los detalles del pedido, incluyendo los productos y su coste total. También incluye un botón para ir a la ubicación aproximada en **Google Maps**. Si el usuario es administrador, muestra el nombre del usuario que realizó el pedido.
- **producto/**
  - **admin.php**: Vista de administración de productos. Si no hay productos, se muestra un mensaje informativo. Si no hay categorías, no se pueden crear productos y se muestra también un mensaje informativo. Si hay productos, se muestran en una tabla con paginación. Cada producto tiene un botón para ver, editar y eliminar.
  - **buscar.php**: Vista para la búsqueda de productos. Muestra los resultados de la búsqueda y permite la paginación. Si no se encuentran resultados, muestra un mensaje indicando que no se encontraron productos.
  - **crear.php**: Vista para crear un producto. Tiene los campos *categoría*, *nombre*, *descripción*, *precio*, *stock*, *oferta* y una imagen.
  - **editar.php**: Vista para editar un producto. Tiene los campos *categoría*, *nombre*, *descripción*, *precio*, *stock*, *oferta* y una imagen.

- **recomendados.php**: Vista de los productos recomendados. Es la página principal de toda la tienda. Primero, recuperamos todos los productos de la base de datos. Luego, si se pasa una categoría por *GET*, se filtran los productos por esa categoría. Si no se pasa ninguna categoría, se muestran tantos productos como esté configurado en **config.php**. Si no hay productos en la base de datos, se muestra un mensaje informativo, y también si no hay productos en la categoría actual o si todos los productos están agotados. En la situación normal de que haya productos y estén catalogados, se muestran en una tabla de 3 columnas en escritorio y 1 columna en móvil.
- **ver.php**: Vista del producto. Muestra la información del producto, la imagen y un botón para añadirlo al carrito con un número determinado de unidades. Si el número de unidades es mayor que el *stock*, se muestra un mensaje de error. Debajo, se muestran las valoraciones, con puntuaciones y comentarios de los usuarios en ese producto mediante paginación. Si el usuario ha comprado el producto y no lo ha valorado ya, se le permite valorar el producto. Si el usuario es propietario de la valoración, la puede eliminar. Si el usuario es administrador, puede eliminar cualquier valoración. Si el usuario no ha comprado el producto, no puede valorarlo.
- **usuario/**
  - **admin.php**: Vista de administración de usuarios. Muestra una tabla con los usuarios existentes y permite crear, editar y eliminar usuarios. También incluye paginación para navegar entre las páginas de los usuarios.
  - **crear.php**: Vista para crear un nuevo usuario desde el panel de administración. Contiene un formulario para ingresar los datos del nuevo usuario. Se valida la información ingresada y se muestra un mensaje de error si es necesario. Tiene más campos que el formulario de registrarse, como rol y color. Se previsualiza la imagen seleccionada antes de enviarla y el color de la interfaz.
  - **editar.php**: Vista para la edición de un usuario por parte de un administrador. Se muestra un formulario con los datos del usuario a editar. El administrador tiene la posibilidad de otorgar o quitar el rol administrador al usuario. También puede cambiar la imagen de perfil del usuario, entre otros.
  - **gestion.php**: Vista de gestión del usuario. Esta vista es mostrada cuando el usuario quiere modificar su perfil. Puede personalizar su nombre, apellidos, email, contraseña, color y foto de perfil. También puede eliminar su cuenta.
  - **login.php**: Vista del inicio de sesión. Contiene el formulario con nombre, contraseña y un *checkbox* para recordar al usuario.
  - **registrarse.php**: Vista para registrarse. Contiene un formulario para ingresar los datos. Se valida la información ingresada y se muestra un mensaje de error si es necesario. Se previsualiza la imagen seleccionada antes de enviarla.

## 6. Evaluación y prueba

A lo largo de todo el desarrollo del proyecto, se han ido probando las distintas funcionalidades y comprobando que todas estén cubiertas y completas. Absolutamente todos los campos poseen validación de datos tanto en el cliente como en el servidor, así como en la base de datos, todo con el objetivo de mantener la seguridad e integridad referencial de los datos, así como evitar ataques de inyección *XSS*, *PHP* o *SQL*. Cada vez que el usuario introduce un dato no válido, es informado de dónde se encuentra el fallo en la validación.

A continuación, se presenta una lista completa de todas las validaciones de campos en la aplicación:

- *Nombre* en **Crear Categoría**.
- *Nombre* en **Editar Categoría**.
- *Categoría, Nombre, Descripción, Precio, Stock, Oferta e Imagen* en **Crear Producto**.
- *Categoría, Nombre, Descripción, Precio, Stock, Oferta e Imagen* en **Editar Producto**.
- *Nombre, Apellidos, Email, Contraseña, Rol, Color e Imagen* en **Crear Usuario**.
- *Nombre, Apellidos, Email, Contraseña, Rol, Color e Imagen* en **Editar Usuario**.
- *Nombre, Apellidos, Email, Contraseña e Imagen* en **Registrarse**.
- *Nombre, Apellidos, Email, Contraseña, Color e Imagen* en **Gestión de Usuario**.
- *Comunidad, Provincia, Municipio, Población, Núcleo, Código Postal y Dirección* en **Crear Pedido**.
- *Texto* en **Buscar Producto**.

## 7. Manual de estilos

### 7.1. Sketches

- Productos Recomendados:



- Ver Producto:



- Buscar Productos:



- Administración de Productos:

**Administración de Productos**

[Crear Producto](#)

Pág. 1

ID	Categoría	Nombre	Descripción	Precio	Stock	Oferta	Fecha	Imagen	Acciones
1	Peligro	P-1 Intersección con prioridad	Peligro por la proximidad de una intersección con una vía, cuyos usuarios deben ceder el paso.	30 €	80	No	01/04/2025 13:25:28		
2	Peligro	P-1a Intersección con prioridad sobre vía a la derecha	Peligro por la proximidad de una intersección con una vía a la derecha, cuyos usuarios deben ceder el paso.	30 €	35	No	27/02/2025 19:57:19		
3	Peligro	P-1b Intersección con prioridad sobre vía a la izquierda	Peligro por la proximidad de una intersección con una vía a la izquierda, cuyos usuarios deben ceder el paso.	30 €	25	74%	27/02/2025 19:57:51		
4	Peligro	P-1c Intersección con prioridad sobre vía a la izquierda y a la derecha	Peligro por la proximidad de una intersección con una vía a la izquierda y a la derecha, cuyos usuarios deben ceder el paso.	30 €	25	74%	27/02/2025 19:57:51		

- Crear y Editar Producto:

**Crear Producto**

Categoría: Peligro

Nombre: P-1 Intersección con prioridad

Descripción: Peligro por la proximidad de una intersección con una vía, cuyos usuarios deben ceder el paso.

Precio: 30 €

Stock: 80

Oferta: No

Fecha: 01/04/2025 13:25:28

Imagen:

[Crear Producto](#)

**Editar Producto**

Categoría: Peligro

Nombre: P-1 Intersección con prioridad

Descripción: Peligro por la proximidad de una intersección con una vía, cuyos usuarios deben ceder el paso.

Precio: 30 €

Stock: 80

Oferta: No

Fecha: 01/04/2025 13:25:28

Imagen:

[Guardar Cambios](#)

- Administración de Categorías:

**Administración de Categorías**

[Crear Categoría](#)

Pág. 1

ID	Nombre	Acciones
1	Peligro	
2	Prioridad	
3	Prohibición de entrada	
4	Restricción de paso	
5	Otras prohibiciones o restricciones	

- Crear y Editar Categoría:

**Crear Categoría**

Nombre: Peligro

[Crear Categoría](#)

**Editar Categoría**

Peligro

Nombre: Peligro

[Guardar Cambios](#)

- Registrarse:

The screenshot shows the 'Registrarse' page. At the top, there is a blue header with the site name 'Tienda de Señales de Tráfico', a home icon, a search icon, and two buttons: 'Registrarse' and 'Iniciar Sesión'. The main content area has a light blue background with the title 'Registrarse' in bold. Below the title is a white form box containing several input fields: 'Nombre', 'Apellidos', 'Email', 'Contraseña', and 'Imagen'. The 'Imagen' field has a file selection button and shows 'Ningún archivo seleccionado'. At the bottom of the form is a blue 'Registrarse' button.

- Iniciar Sesión:

The screenshot shows the 'Iniciar Sesión' page. It has the same blue header as the registration page. The main content area has a light blue background with the title 'Iniciar Sesión' in bold. Below the title is a white form box with input fields for 'Email' and 'Contraseña'. Below these fields is a checkbox labeled 'Recordar usuario durante 7 días'. At the bottom of the form is a blue 'Iniciar Sesión' button. At the very bottom of the page, there is a dark blue footer with links for 'Política de Privacidad', 'Condiciones de Uso', and 'Sobre Nosotros', and a copyright notice 'Alonso Hernández Robles © 2025'.

- Gestión de Usuario:
















The screenshot shows the 'Gestión de Usuario' page. The header is blue and includes the site name, home and search icons, and a navigation bar with links for 'Categorías', 'Productos', 'Pedidos', 'Usuarios', 'A', 'Carrito (2)', and 'Cerrar Sesión'. The main content area has a light blue background with the title 'Gestión de Usuario' and a subtitle 'Admin Admin'. Below this is a white form box for editing a user. The form contains input fields for 'Nombre', 'Apellidos', 'Email', 'Contraseña', 'Color', and 'Imagen'. The 'Imagen' field shows a circular profile picture with the letter 'A'. At the bottom of the form are two buttons: a blue 'Guardar Cambios' button and a red 'Eliminar Usuario' button.

- Administración de Usuarios:

Tienda de Señales de Tráfico

Administración de Usuarios

Crear Usuario

ID	Nombre	Apellidos	Email	Rol	Imagen	Color	Acciones
1	Admin	Admin	admin@admin.com	Administrador		No	 
2	Alonso	Hernández Robles	alonso.ensibemot@gmail.com	Usuario		#F599F5	 
3	Pablo	Sánchez Martín	pablosanchez808@gmail.com	Usuario		No	 
4	Zapi	Pipo	samueluca2004@gmail.com	Usuario		#CC00FF	 
5	David	Vilchez Martínez	davidvzm@gmail.com	Usuario		#FF0000	 

Política de Privacidad Condiciones de Uso Sobre Nosotros

- Crear y Editar Usuario:

Tienda de Señales de Tráfico

Crear Usuario

Nombre

Apellidos

Email

Contraseña

Rol

Color

Imagen

Crear Usuario

Editar Usuario

Nombre

Apellidos

Email

Contraseña

Rol

Color

Imagen

Editar Usuario

Política de Privacidad Condiciones de Uso Sobre Nosotros

- Crear Pedido:

Tienda de Señales de Tráfico

Pedido (2 productos)

Comunidad Autónoma

Selecciona una comunidad

Provincia

Selecciona una provincia

Municipio

Selecciona un municipio

Población

Selecciona una población

Núcleo

Selecciona un núcleo

Código Postal

Selecciona un código postal

Dirección 1

Selecciona una calle

Dirección 2

Número, Piso, Puerta...

Ir a la pasarela de pago

Política de Privacidad Condiciones de Uso Sobre Nosotros

- Pedido Listo:



- Ver Pedido:



- Cargando (PayPal):





- Mis Pedidos:

Tienda de Señales de Tráfico

Mis Pedidos

Carrito (0)

Cerrar Sesión

Mis Pedidos

Fecha	Dirección	Ubicación	Nº Productos	Coste	Estado	Ver
17/04/2025 13:17:22	Andalucía, Cádiz, Conil de la Frontera, Conil de la Frontera, Conil de la Frontera, 11140, Alheli 1		1	0.1 €	Pendiente	

Política de Privacidad

Condiciones de Uso

Sobre Nosotros

Alonso Hernández Robles © 2025

- Administración de Pedidos:

Tienda de Señales de Tráfico

Mis Pedidos

Carrito (0)

Cerrar Sesión

Administración de Pedidos

ID	Nº Productos	Usuario	Dirección	Ubicación	Coste	Fecha	Estado	Acciones
1	1 producto	Admin Admin	Andalucía, Cádiz, Conil de la Frontera, Conil de la Frontera, Conil de la Frontera, 11140, Alheli 1		0.1 €	17/04/2025 13:17:22	Pendiente	<div></div> <div></div> <div></div>

Política de Privacidad

Condiciones de Uso

Sobre Nosotros

Alonso Hernández Robles © 2025

- Carrito:

Tienda de Señales de Tráfico

Cat.

Prod.

Ped.

Usu.

Carrito (2)

Cerrar Sesión

Carrito (2 productos)

Veclar carrito

Imagen	Producto	Cantidad	Precio	Total	Eliminar
	S-235 Preseñalización con señales sobre la calzada en autopista o autovía hacia cualquier carretera	<div>- 1 +</div>	105 €	105 €	<div></div>
	P-12 Aeropuerto	<div>- 1 +</div>	30 €	30 €	<div></div>

Total: 135 €

Hacer pedido

- Condiciones de Uso:



- Política de Privacidad:



- Sobre Nosotros:



- Pop-up de Administradores:

## ATENCIÓN

**Hay algunas consideraciones a tener en cuenta como usuario con rol administrador:**

- **Si eliminas una categoría** se eliminarán todos los productos que pertenezcan a dicha categoría y:
  - Se eliminarán todas las valoraciones que pertenezcan a dicho producto.
  - Se eliminarán todos los pedidos que se compongan únicamente de dicho producto.
- **Si eliminas un producto**, se eliminarán todas las valoraciones que pertenezcan a dicho producto y todos los pedidos que se compongan únicamente de dicho producto.
- **Si eliminas un pedido**, se eliminarán todas las valoraciones del usuario que tengan los productos de dicho pedido y, si para cada producto, éste no figura en otro pedido del mismo usuario a partir del cual el mismo haya hecho una valoración.
- **Si eliminas un usuario**, se eliminarán todas las valoraciones y los pedidos que haya realizado dicho usuario.

Entendido

## 7.2. Criterios de accesibilidad

Se han implementado distintas características para asegurar que la experiencia de usuarios con menos accesibilidad de la normativa tenga una experiencia tan cercana a la prevista como sea posible.

- Uso de etiquetas semánticas de *HTML* (*<header>*, *<main>*, *<footer>*, etc.).
- Inclusión de atributos "alt" en las imágenes para que los lectores de pantalla puedan describirlas.
- Contraste adecuado entre texto y fondo, para facilitar la lectura.
- Navegación mediante teclado: el sitio puede ser recorrido sin necesidad de ratón.
- Formularios accesibles con etiquetas *<label>* correctamente asociadas a sus campos *<input>*.

## 7.3. Criterios de usabilidad

A su vez, se ha implementado la aplicación web siempre desde un punto de vista usable y fácil e intuitivo, para garantizar una correcta experiencia de usuario. Esto se refleja en textos, botones, tamaños y ventanas, entre otros muchos factores. Algunos aspectos para tener en cuenta son:

- Estructura clara y lógica del contenido y menús.
- Diseño adaptable a diferentes tamaños de pantalla (móviles, tabletas, etc.).
- Botones de acción visibles y comprensibles (como “Añadir al carrito” o “Registrarse”).

- Mensajes de error claros y útiles en formularios, indicando qué campo necesita corrección.
- Carga rápida de páginas y una experiencia de navegación fluida, gracias a consultas eficientes y un buen servicio de hosting (**Hostinger**).

#### 7.4. Tipografía

La aplicación web hace uso únicamente de la fuente Rubik, en distintos tamaños y pesos.

**Aa Bb Cc Dd Ee Ff Gg Hh Ii**  
**Jj Kk Ll Mm Nn Oo Pp Qq Rr**  
**Ss Tt Uu Vv Ww Xx Yy Zz**  
**1 2 3 4 5 6 7 8 9 0**

#### 7.5. Mapa de colores

1864AF	CAE6FF	12508C	CFE6FF	2C3E50	34495E	3498DB	2980B9	99C1E4	004FAD
24, 100, 175	202, 230, 255	18, 80, 140	207, 236, 255	44, 62, 80	52, 73, 94	52, 152, 219	41, 128, 185	153, 193, 228	0, 79, 173
Green Blue	Beau Blue	Yale Blue	Columbia Blue	Charcoal	Charcoal	Carolina Blue	Star Command Blue	Pale Cerulean	Sapphire

## 7.6. Dispositivos/Vistas

La aplicación web es adaptable a dispositivos de todas resoluciones excepto a pantallas con menos de 400 píxeles de ancho como límite estético, y menos de 360 píxeles de ancho como límite de usabilidad.

Para implementar esta característica, se ha hecho uso de *media queries* en el archivo **assets/css/style.css**, reduciendo principalmente el tamaño de muchos elementos a lo largo de todo el documento *HTML* y ocultando muchos otros cuando se ha considerado necesario.

También se ha usado *JavaScript* para ajustar ciertos elementos en el *header* cuando la resolución del ancho de la pantalla ha comprometido los elementos y el diseño original de la página.

## 8. Software utilizado

**Visual Studio Code:** Editor de código fuente desarrollado por **Microsoft** para **Windows**, **Linux**, **macOS** y **Web**. Incluye soporte para la depuración, control integrado de **Git**, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.

**XAMPP:** Paquete de software libre que incluye las herramientas necesarias para configurar un servidor local y desarrollar aplicaciones web en el *back-end* mediante *PHP*, y el sistema gestor de base de datos *MySQL*.

**phpMyAdmin:** Gestor visual del sistema gestor de base de datos *MySQL*, donde el usuario puede ver, editar, crear, eliminar... tablas y realizar consultas *SQL*.

**Google Chrome:** Navegador de pruebas.

**Git:** Sistema de control de versiones para manejar el flujo de trabajo del proyecto.

**GitHub:** Plataforma ligada a **Git** para subir repositorios de forma remota y almacenar código y ficheros.

**PayPal Developer:** Entorno de pruebas para el pago de los pedidos. Primero se ha usado el modo *Sandbox* para desarrollo, y luego se ha usado el modo *Live* para producción.

**Hostinger:** Sitio web dedicado al hosting de páginas web de manera efectiva y eficiente. Permite administrar los sitios web con un gestor de archivos y tiene integrado **phpMyAdmin**.

## 9. Mejoras posibles y aportaciones

### 9.1. Funcionalidad

- Implementar un buscador de categorías, productos, pedidos o usuarios para cada una de las páginas de administración.
- Filtrar los productos por distintos campos, como ofertas, *stock* o más comprados.
- Edición de valoraciones en los productos por parte de los usuarios.
- Búsqueda de producto por palabras clave, esto se traduce en un nuevo campo oculto en la base de datos, preparado para proporcionar mejores resultados de búsqueda.
- Ordenar resultados por distintos criterios, como *stock*, precio u oferta.

### 9.2. Código

- Se podría haber confeccionado de forma más eficiente la lógica de negocio, es decir, cada una de las acciones en los controladores. Un ejemplo es que, en ciertas acciones, dependiendo de si un valor de la variable superglobal `$_GET`, se hace una cosa u otra, pero ambas opciones se construyen de forma prácticamente igual a excepción de unas pocas líneas de código distintas.
- También se podrían haber separado algunos fragmentos de *HTML* que son frecuentemente repetidos a lo largo de la aplicación, como los botones de paginación.
- Encriptar la contraseña con mayor número de iteraciones *BCRYPT* o usar un algoritmo más avanzado de encriptación de información.

### 9.3. Información

La información mostrada actualmente está mostrada de la mejor manera posible a nivel de planteamiento, pero si se implementasen nuevas funcionalidades, sería preciso asegurar la correcta adaptabilidad a la versión actual de la aplicación por parte de los nuevos elementos de la interfaz gráfica.

## 10. Recuento de trabajo

### 10.1. Reporte general

[https://github.com/AloncraftMC/TFG\\_DAW\\_AlonsoHernandezRobles/tree/main/docs/Fase%20Final/TFG\\_Report\\_WakaTime.pdf](https://github.com/AloncraftMC/TFG_DAW_AlonsoHernandezRobles/tree/main/docs/Fase%20Final/TFG_Report_WakaTime.pdf)

### 10.2. Información y archivos

[https://github.com/AloncraftMC/TFG\\_DAW\\_AlonsoHernandezRobles/tree/main/docs/Fase%20Final/TFG\\_WakaTime.pdf](https://github.com/AloncraftMC/TFG_DAW_AlonsoHernandezRobles/tree/main/docs/Fase%20Final/TFG_WakaTime.pdf)