

# AutoMobile



Artashes Asatryan 2ºDAW 2023-2024



# Índice

<b>Introducción</b>	<b>3</b>
<b>Necesidades del proyecto</b>	<b>3</b>
<b>Análisis de mercado</b>	<b>4</b>
<b>Justificación</b>	<b>4</b>
<b>Stack tecnológico</b>	<b>5</b>
Back-End	5
Front-End	5
<b>Base de datos</b>	<b>6</b>
<b>Prototipo</b>	<b>7</b>
<b>API Rest</b>	<b>19</b>
Forma de acceso	19
Documentación completa	19
Acceso	19
Inicio de sesión	19
Usuarios	20
Coches	20
Imágenes	21
Noticias	22
Favoritos	22
<b>Despliegue</b>	<b>23</b>
Explicación de Docker-compose.yml:	<b>25</b>
Explicación de Dockerfile de API Spring:	<b>26</b>
Explicación de Dockerfile de Angular:	<b>26</b>
<b>Conclusión</b>	<b>28</b>



## Introducción

La página web de compra-venta de coches ofrece a los usuarios una plataforma completa y confiable para explorar una amplia variedad de vehículos disponibles para la venta, así como acceder a información valiosa sobre el mundo automotriz.

Los usuarios pueden navegar por los listados de coches publicados por otros usuarios sin necesidad de registrarse, pero se necesita un registro para disfrutar de funciones adicionales como ver anuncios, publicar coches, añadir otros a favoritos.

Se podrá hacer un filtro de los coches para encontrar el coche adecuado para cada persona, y ordenarlos según el precio o kilometraje.

Permitimos contactar con los vendedores para adquirir un coche, y la modificación del perfil para personalizar el usuario a sus gustos.

## Necesidades del proyecto

Funciones implementados:

- **RF01 - El usuario podrá registrarse.**
- **RF02 - El usuario podrá iniciar sesión.**
- **RF03 - El usuario podrá hacer cambios en su perfil.**
  
- **RF04 - Habrá un pequeño formulario en la pantalla de inicio para hacer la búsqueda del coche deseado.**
- **RF05 - Se podrá ver los coches publicados.**
- **RF06 - Se podrá filtrar la búsqueda de los coches.**
- **RF07 - Se podrá ordenar la búsqueda de los coches.**
- **RF08 - Se podrá contactar con el vendedor (número de tlf o email).**
- **RF09 - Se podrá guardar coches en favoritos.**



- **RF10 - Se podrá publicar coches.**
- **RF11 - Se podrá editar coches publicados.**
- **RF12 - Habrá una sección de noticias e información.**
  
- **RF13 - Panel de Administración: Los administradores tendrán acceso a un panel de control desde el cual pueden gestionar todos los aspectos de la plataforma.**
- **RF14 - Gestión de Usuarios: Los administradores pueden ver y administrar la lista de usuarios.**
- **RF15 - Gestión de Contenido: Los administradores pueden supervisar y gestionar los anuncios publicados por los usuarios para garantizar que cumplan con las directrices de la plataforma y no contengan contenido inapropiado o fraudulento.**

## **Análisis de mercado**

De competencia encontramos diferentes páginas web, cómo AutoScout24, Coches.net, Milanuncios, Wallapop. De ellos, se asemejara más a AutoScout24 y Coches.net, ya que poseen secciones de ventas de coches, y además, otras de información. Milanuncios y Wallapop son páginas de venta de productos en general, sin un enfoque en los coches solamente y sin secciones de información general.

## **Justificación**

La falta de páginas serias y confiables en el mercado de compra-venta de coches me ha llevado a desarrollar esta idea. Como entusiasta del mundo automotriz, deseo una plataforma que no solo ofrezca una amplia variedad de vehículos, sino que también garantice seguridad. También, se pone el foco en el mantenimiento de la web y la supervisión de los anuncios que se publican en esta web. Se busca que la falta de confianza de los usuarios en la competencia se traslade a esta web.



## Stack tecnológico

### Back-End

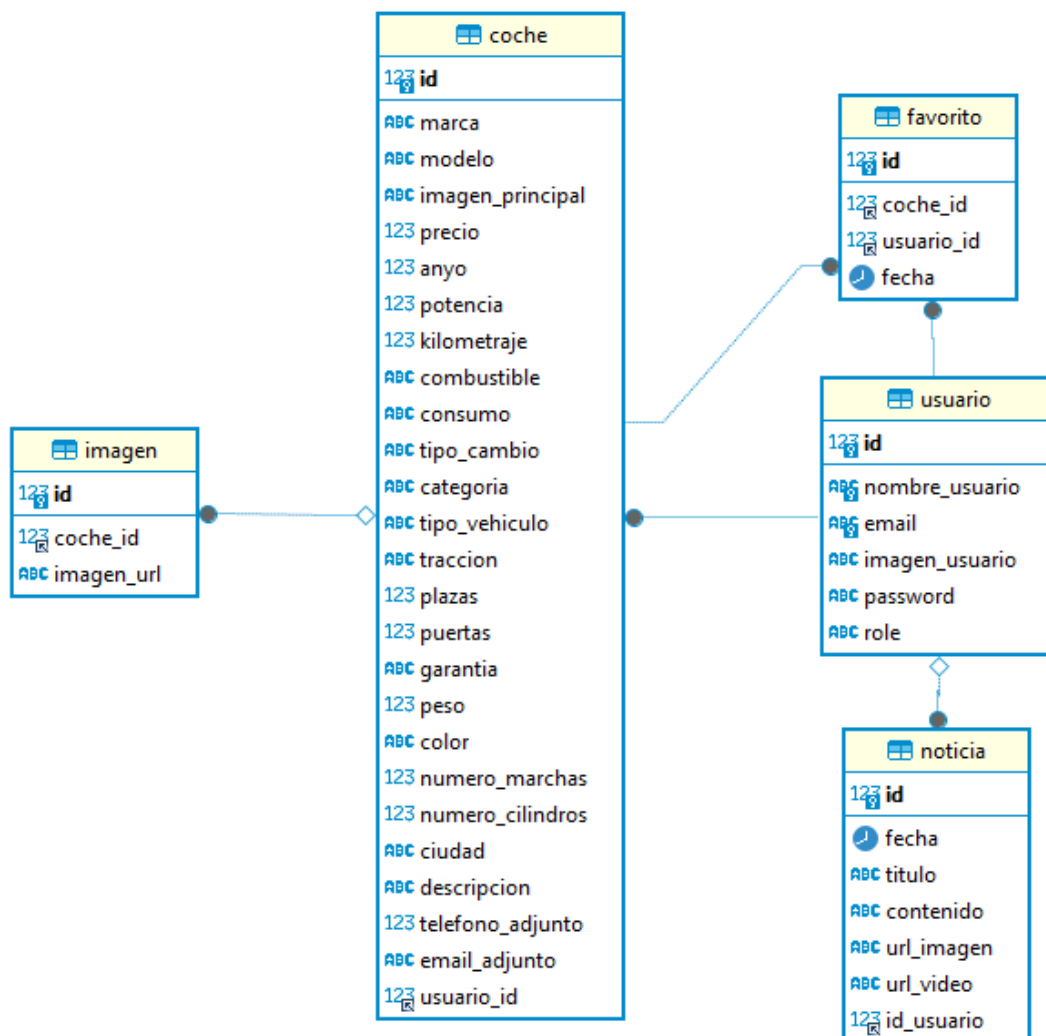
Para el back-end se ha utilizado Spring Framework combinado con la base de datos MySQL. Debido a que son una buena combinación para usarse como API, además de que el mantenimiento no requiere mucho esfuerzo.

### Front-End

Para el front-end se ha utilizado el framework Angular (HTML + Bootstrap, CSS + SCSS y TypeScript). Ofrece una buena organización del código en componentes y una buena navegación.



## Base de datos



La base de datos consta de:

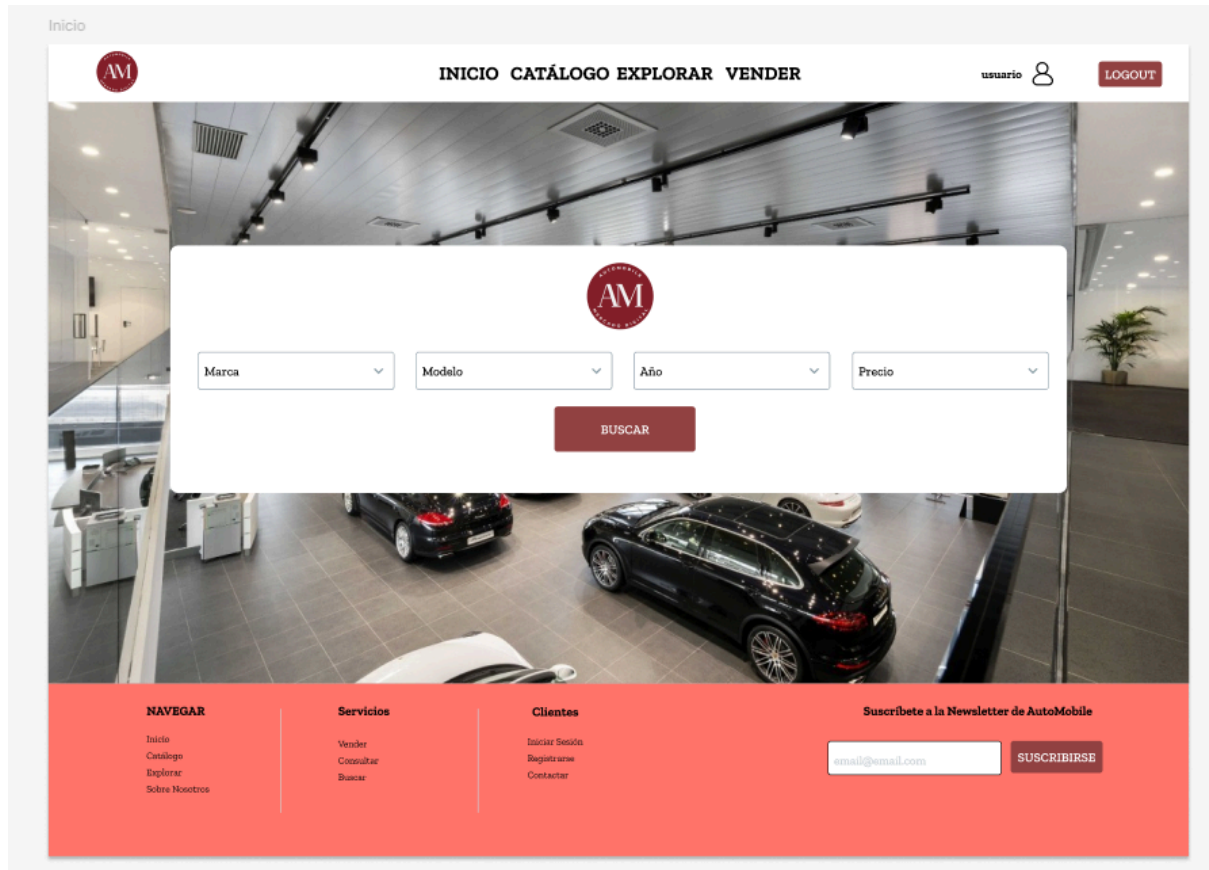
- Tabla Coche: Es la tabla principal que va a contener la información del coche que se va a publicar y el usuario al que pertenece.
- Tabla Imagen: Almacena las imágenes adicionales del coche.
- Tabla Usuario: Se utiliza para el inicio de sesión, registro, explorar y publicar coches. En algunos casos se necesita un inicio de sesión para acceder a funcionalidades.
- Tabla Favorito: Contiene los coches que ha añadido el usuario a favoritos. Se actualiza cuando el usuario añade un coche a favoritos o lo borra.
- Tabla Noticia: Es una tabla que almacena las noticias que se van a ver en la sección de Explorar.



# Prototipo

El prototipo se ha realizado en Figma, Url de Figma completo: [url](#)

Inicio:






## Catálogo:

Catálogo

AM

INICIO CATÁLOGO EXPLORAR VENDER

usuario  LOGOUT

Inicio > Catálogo

**Datos básicos**

**Marca**  
Todos ▾

**Modelo**  
Todos ▾

**Carrocería**  
Todos ▾


**Combustible**  
Todos ▾

**Año**  
Desde ▾ Hasta ▾


**Precio**  
Desde ▾ Hasta ▾

**Kilometraje**  
Desde ▾ Hasta ▾


**Potencia**  
Desde ▾ Hasta ▾




**BMW Serie 3**  
2000 KM, 2020, Manual, etc  
**35000,00 €**



**Opel Corsa**  
70000 KM, 2018, Manual, etc  
**23000,00 €**



**Volkswagen Golf**  
35000 KM, 2019, Manual, etc  
**22000,00 €**



**Mercedes Benz E-Class**  
15000 KM, 2021, Automático, etc  
**40000,00 €**

**NAVEGAR**

[Inicio](#)  
[Catálogo](#)  
[Explorar](#)  
[Sobre Nosotros](#)

**Servicios**

[Vender](#)  
[Consultar](#)  
[Buscar](#)

**Cientes**

[Iniciar Sesión](#)  
[Registrarse](#)  
[Contactar](#)

Suscríbete a la Newsletter de AutoMobile


**SUSCRIBIRSE**






## Coche:


[Ver coche](#)




INICIO CATÁLOGO EXPLORAR VENDER

usuario  [LOGOUT](#)

[Inicio](#) > [Catálogo](#)



 [Añadir a favoritos](#)

[Compartir](#)





BMW 320d

Madrid

€ 35000,-

[CONTACTAR](#)

[EMAIL / TLF](#)



<b>Año</b>	<b>Potencia</b>	<b>Kilometraje</b>
2018	190	2000
<b>Combustible</b>	<b>Color</b>	
Diesel	Azul	

**Datos básicos**

Categoría **Sedán**

Tipo de vehículo **Ocasión**

Tracción **Trasera**

Plazas **5**

Puertas **5**

Color **Azul**

Peso **1700**

**Datos básicos**

Kilometraje **2000**

Año **2018**

Garantía **1 año**

**Descripción**

Sedán muy popular con bajo consumo de combustible y alto rendimiento.

**NAVEGAR**

[Inicio](#)  
[Catálogo](#)  
[Explorar](#)  
[Sobre Nosotros](#)

**Servicios**

[Vender](#)  
[Consultar](#)  
[Buscar](#)

**Cientes**

[Iniciar Sesión](#)  
[Registrarse](#)  
[Contactar](#)


Suscríbete a la Newsletter de AutoMobile

[SUSCRIBIRSE](#)




Explorar:


Explorar




INICIO CATÁLOGO EXPLORAR VENDER

usuario  [LOGOUT](#)


[Inicio](#) > [Explorar](#)




Presentación de BMW




Nuevos motores




Mejores eléctricos



Nuevo Range Rover Sport



Nuevos proyectos de coches



Probamos los nuevos Toyota

**NAVEGAR**

- [Inicio](#)
- [Catálogo](#)
- [Explorar](#)
- [Sobre Nosotros](#)

**Servicios**

- [Vender](#)
- [Consultar](#)
- [Buscar](#)

**Cientes**

- [Iniciar Sesión](#)
- [Registrarse](#)
- [Contactar](#)

Suscríbete a la Newsletter de AutoMobile

[SUSCRIBIRSE](#)



Ver Noticia:


Explorar

INICIO CATÁLOGO EXPLORAR VENDER

usuario LOGOUT

Inicio > Explorar

### El nuevo BMW Serie 3 2020



Presentamos el último sedán de BMW muy popular por su bajo consumo de combustible y alto rendimiento. Respecto a la oferta mecánica debería ser idéntica. En gasolina, las opciones comenzarían con el 320i (184 CV), seguido del 330i (258 CV), rematadas por el M340i xDrive con 374 CV. Las de ciclo diésel estarían representadas por el 318d (150 CV), el 320d (190 CV) y el 330d (265 CV). Sobre una hipotética versión M3 existen muchos más rumores, pero de momento no hay nada confirmado.

**NAVEGAR**

- Inicio
- Catálogo
- Explorar
- Sobre Nosotros

**Servicios**

- Vender
- Consultar
- Buscar

**Cientes**

- Iniciar Sesión
- Regístrate
- Contactar


Suscríbete a la Newsletter de AutoMobile

SUSCRIBIRSE




Vender:

Vender



INICIO CATÁLOGO EXPLORAR VENDER

usuario  [LOGOUT](#)

[Inicio](#) > [Vender](#)

**Datos básicos**

Marca

Modelo

Año

**Motor**

Potencia

Cilindros

Combustible

**NAVEGAR**

- [Inicio](#)
- [Catálogo](#)
- [Explorar](#)
- [Sobre Nosotros](#)

**Servicios**

- [Vender](#)
- [Consultar](#)
- [Buscar](#)

**Clientes**

- [Iniciar Sesión](#)
- [Registrarse](#)
- [Contactar](#)

Suscríbete a la Newsletter de AutoMobile

[SUSCRIBIRSE](#)



Login:

Login

INICIO CATÁLOGO EXPLORAR VENDER

usuario [LOGOUT](#)

☐ Recordarme [Olvidaste la contraseña?](#)

[INICIAR SESIÓN](#)

No tienes cuenta ? [Regístrate](#)

NAVEGAR

[Inicio](#)  
[Catálogo](#)  
[Explorar](#)  
[Sobre Nosotros](#)

Servicios

[Vender](#)  
[Consultar](#)  
[Buscar](#)

Cientes

[Iniciar Sesión](#)  
[Registrarse](#)  
[Contactar](#)


Suscribete a la Newsletter de AutoMobile

[SUSCRIBIRSE](#)



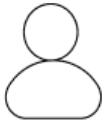
## Registro:

Registro



INICIO CATÁLOGO EXPLORAR VENDER

Iniciar SesiónRegistrarse



Nombre de usuario

Email

Contraseña

Registrarte

Ya tienes cuenta ? Iniciar Sesión

NAVEGAR

- Inicio
- Catálogo
- Explorar
- Sobre Nosotros

Servicios

- Vender
- Consultar
- Buscar

Clientes

- Iniciar Sesión
- Registrarse
- CONTACTAR


Suscribete a la Newsletter de AutoMobile

SUSCRIBIRSE




## Perfil:

Perfil



INICIO CATÁLOGO EXPLORAR VENDER

usuario  LOGOUT

[Inicio](#) > [Perfil](#)

## PERFIL

Perfil

Gestión de usuarios

Coches Publicados


Favoritos

Nombre de usuario: admin

Email: admin@gmail.com

Contraseña: \*\*\*\*\*

Tipo de usuario: Administrador



Modificar

**NAVEGAR**  
[Inicio](#)  
[Catálogo](#)  
[Explorar](#)  
[Sobre Nosotros](#)

**Servicios**  
[Vender](#)  
[Consultar](#)  
[Buscar](#)

**Cientes**  
[Inicio Sesión](#)  
[Registrarse](#)  
[Contactar](#)

Suscribete a la Newsletter de AutoMobile

SUSCRIBIRSE



## Gestión de usuarios (Admin):

Gestor

INICIO CATÁLOGO EXPLORAR VENDER

usuario LOGOUT

Inicio > Gestión de usuarios

### GESTIÓN DE USUARIOS

Imagen	Usuario	Email	Rol	Gestión
	admin	admin@gmail.com	Admin	<div>Editar</div> <div>Eliminar</div>
	usuario2	usuario2@gmail.com	Usuario	<div>Editar</div> <div>Eliminar</div>
	usuario3	usuario3@gmail.com	Usuario	<div>Editar</div> <div>Eliminar</div>
	usuario4	usuario4@gmail.com	Usuario	<div>Editar</div> <div>Eliminar</div>

Perfil

**Gestión de usuarios**

Coches Publicados

Favoritos

**NAVEGAR**

[Inicio](#)

[Catálogo](#)

[Explorar](#)

[Sobre Nosotros](#)

**Servicios**

[Vender](#)

[Consultar](#)

[Buscar](#)

**Cientes**

[Iniciar Sesión](#)

[Registrarse](#)

[Contacto](#)

Suscríbete a la Newsletter de AutoMobile


SUSCRIBIRSE






## Coches Publicados:

Publicados



INICIO CATÁLOGO EXPLORAR VENDER

usuario  LOGOUT

Inicio > Publicados


### Publicados

Perfil


Gestión de usuarios

**Coches Publicados**


Favoritos




**BMW Serie 3**  
2000 KM, 2020, Manual, etc  
**35000,00 €**



**Opel Corsa**  
70000 KM, 2018, Manual, etc  
**23000,00 €**



**Volkswagen Golf**  
35000 KM, 2019, Manual, etc  
**22000,00 €**



**Mercedes Benz E-Class**  
15000 KM, 2021, Automático, etc  
**40000,00 €**

**NAVEGAR**

Inicio  
Catálogo  
Explorar  
Sobre Nosotros

**Servicios**

Vender  
Consultar  
Buscar

**Cientes**

Iniciar Sesión  
Registrarse  
Contactar

Suscríbete a la Newsletter de AutoMobile

**SUSCRIBIRSE**




## Coches Favoritos:

Favoritos

AM

INICIO CATÁLOGO EXPLORAR VENDER

usuario  LOGOUT

Inicio > Favoritos


Favoritos

Perfil


Gestión de usuarios

Coches Publicados

Favoritos



BMW Serie 3  
2000 KM, 2020, Manual, etc  
**35000,00 €**



Mercedes Benz E-Class  
15000 KM, 2021, Automático, etc  
**40000,00 €**

NAVEGAR

Inicio  
Catálogo  
Explorar  
Sobre Nosotros

Servicios

Vender  
Consultar  
Buscar

Clientes

Iniciar Sesión  
Registrarse  
Contactar

Suscríbete a la Newsletter de AutoMobile

SUSCRIBIRSE



## API Rest

### Forma de acceso

Todos los métodos indicados en el documento se forman desde la URL base "<http://localhost:8080/api/v1>"

### Documentación completa

<https://documenter.getpostman.com/view/31938682/2sA3JNaKt7>

### Acceso

Recurso	Descripción
/public	Es de acceso <b>público</b>
/adminuser	Pueden acceder <b>Administradores</b> y <b>Usuarios</b> registrados
/admin	Pueden acceder sólo los <b>Administradores</b> registrados
/user	Pueden acceder sólo los <b>Usuarios</b> registrados

### Inicio de sesión

Recurso	Descripción
<b>POST</b> auth/signup	Crea una cuenta con el email y contraseña que se pase en el body.
<b>POST</b> auth/signin	Se inicia sesión con el email y contraseña que se pase en el body.



## Usuarios

Recurso	Descripción
GET admin/usuario	Devuelve todos los usuarios que hay en la base de datos.
GET admin/usuario/{id}	Devuelve el usuario que tenga el id pasado.
GET adminuser/usuario/actual	Devuelve el usuario que tiene la sesión iniciada
POST public/usuario	Crea el usuario pasado por body.
PUT admin/usuario/{id}	Actualiza el usuario con el id buscado con los datos pasados por el body.
PUT adminuser/usuario	Actualiza el perfil del usuario que está iniciado sesión.
DELETE admin/usuario/{id}	Borra el usuario con el id buscado.

## Coches

Recurso	Descripción
GET public/coche	Devuelve todos los coches que hay en la base de datos.
GET public/coche/{id}	Devuelve el coche que tenga el id pasado.
GET adminuser/coche/usuario/{id}	Devuelve los coches que tenga el usuario con id pasado.
GET adminuser/coche/favorito/usu	Devuelve los coches favoritos del usuario con id pasado.



ario/{id}	
<b>GET</b> https://www.autoscout24.es/listing-form/models/C/{id}	Devuelve los modelos de coche por la marca (se pasa un id por cada marca)
<b>POST</b> adminuser/coche	Crea el coche pasado por body.
<b>PUT</b> admin/coche/{id}	Actualiza el coche con el id buscado con los datos pasados por el body.
<b>PUT</b> adminuser/coche/{id}	Actualiza el coche con el id buscado con los datos pasados por el body. Solo si el coche le pertenece al usuario que lo actualiza.
<b>DELETE</b> admin/coche/{id}	Borra el coche con el id buscado.
<b>DELETE</b> adminuser/coche/{id}	Borra el coche con el id buscado. Solo si el coche le pertenece al usuario que lo borra.

## Imágenes

Recurso	Descripción
<b>GET</b> public/imagen	Devuelve todas las imágenes que hay en la base de datos.
<b>GET</b> public/imagen/{id}	Devuelve la imagen que tenga el id pasado.
<b>GET</b> public/imagen/coche/{id}	Devuelve las imágenes del coche con el id pasado.
<b>POST</b> adminuser/imagen	Crea la imagen pasada por body.
<b>PUT</b> admin/imagen/{id}	Actualiza la imagen con el id buscado con los datos pasados por el body.



<b>PUT</b> adminuser/imagen/{id}	Actualiza la imagen con el id buscado con los datos pasados por el body. Solo si la imagen le pertenece al usuario que lo actualiza.
<b>DELETE</b> adminuser/imagen/{id}	Borra la imagen con el id buscado.
<b>DELETE</b> admin/imagen/{id}	Borra la imagen con el id buscado. Solo si la imagen le pertenece al usuario que lo borra.

## Noticias

Recurso	Descripción
<b>GET</b> adminuser/noticia	Devuelve todas las noticias que hay en la base de datos.
<b>GET</b> adminuser/noticia/{id}	Devuelve la noticia que tenga el id pasado.
<b>POST</b> admin/noticia	Crea la noticia pasada por body.
<b>PUT</b> admin/noticia/{id}	Actualiza la noticia con el id buscado con los datos pasados por el body.
<b>DELETE</b> admin/noticia/{id}	Borra la noticia con el id buscado.

## Favoritos

Recurso	Descripción
<b>GET</b> adminuser/favorito	Devuelve todos los favoritos que hay en la base de datos.
<b>GET</b> adminuser/favorito/{id}	Devuelve el favorito que tenga el id

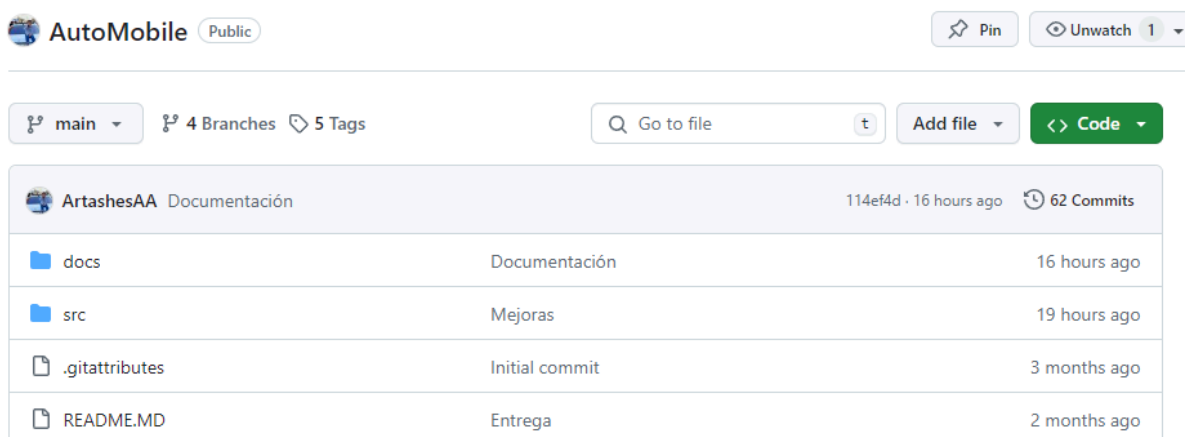


	pasado.
<b>GET</b> adminuser/favorito/usuario/{id}	Devuelve los favoritos que tiene el usuario con el id pasado.
<b>POST</b> adminuser/favorito	Crea el favorito pasado por body.
<b>PUT</b> adminuser/favorito/{id}	Actualiza el favorito con el id buscado con los datos pasados por el body.
<b>DELETE</b> admin/favorito/{id}	Borra el favorito con el id buscado.
<b>DELETE</b> adminuser/favorito/{id}	Borra el favorito con el id buscado. Solo si el favorito le pertenece al usuario que lo borra.

## Despliegue

Para ejecutar el proyecto, se debe seguir los siguientes pasos:

1. Hacer pull del repositorio / Descargar el repositorio.



2. Iniciar Docker.
3. Abrir un Terminal/Símbolo del sistema.



- Dirigirse al directorio src del repositorio en la terminal (donde se encuentra el archivo docker-compose.yml).

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.3593]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\artas>cd C:\Users\artas\OneDrive\Documentos\GitHub\AutoMobile\src
C:\Users\artas\OneDrive\Documentos\GitHub\AutoMobile\src>
```

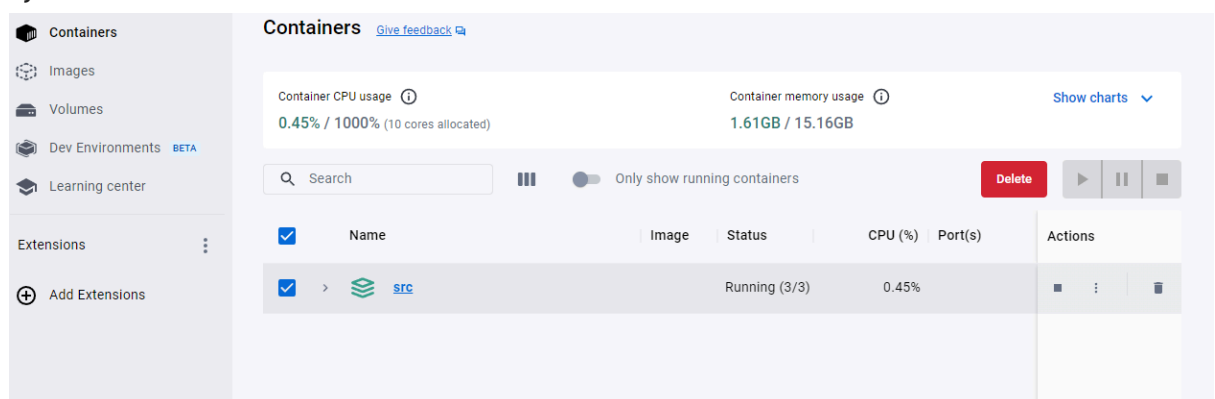
- Ejecutar el comando:.

```
docker-compose build
```

- Ejecutar el comando: docker-compose up.

```
docker-compose up
```

- Comprobamos que en la sección Containers, el contenedor se está ejecutando.







## Explicación de Docker-compose.yml:

```
1  version: "3"
2  services:
3    angular-app:
4      build:
5        context: ./src-frontend
6        dockerfile: Dockerfile
7      ports:
8        - "4200:4200"
9      depends_on:
10       - spring-boot-app
11
12   spring-boot-app:
13     build:
14       context: ./src-api
15       dockerfile: Dockerfile
16     ports:
17       - "8080:8080"
18     environment:
19       - SPRING_DATASOURCE_URL=jdbc:mysql://mysql:3306/automobile
20       - SPRING_DATASOURCE_USERNAME=root
21       - SPRING_DATASOURCE_PASSWORD=root
22     depends_on:
23       - mysql
24
25   mysql:
26     image: mysql:8.0
27     ports:
28       - "3306:3306"
29     environment:
30       MYSQL_ROOT_PASSWORD: root
31       MYSQL_DATABASE: automobile
32     volumes:
33       - mysql-data:/var/lib/mysql
34
35 volumes:
36   mysql-data:
```

### Servicios:

1. **angular-app:** Es la parte front de la aplicación.
  - **context:** Indica la ruta del directorio donde se encuentra el código fuente del frontend (./src-frontend).
  - **dockerfile:** Especifica que se debe utilizar el archivo Dockerfile ubicado en el directorio context para construir la imagen de Docker.
  - **ports:** Mapea el puerto 4200 del contenedor al puerto 4200 del host.
  - **depends\_on:** spring-boot-app: Indica que este servicio debe iniciarse después de la API



2. **spring-boot-app**: Es la API de la aplicación.
  - **context**: Indica la ruta del directorio donde se encuentra el código fuente del backend (./src-api).
  - **dockerfile**: Especifica que se debe utilizar el archivo Dockerfile ubicado en el directorio context para construir la imagen de Docker.
  - **ports**: Mapea el puerto 8080 del contenedor al puerto 8080 del host.
  - **environment**: Define variables de entorno necesarias para la configuración del datasource de Spring Boot.
  - **depends\_on**: spring-boot-app: Indica que este servicio debe iniciarse después de la base de datos MySQL.
3. **mysql**: Es la base de datos donde se almacenan los datos de la API.
  - **image**: Especifica la imagen de Docker a utilizar (mysql:8.0).
  - **ports**: Mapea el puerto 3306 del contenedor al puerto 3306 del host.
  - **environment**: Define variables de entorno necesarias para la configuración inicial de la base de datos MySQL.
  - **volumes**: Crea un volumen llamado mysql-data en el directorio /var/lib/mysql dentro del contenedor para persistir los datos de la base de datos.

#### Volúmenes:

**mysql-data**: Define un volumen Docker persistente para almacenar los datos de MySQL fuera del contenedor, asegurando que los datos no se pierdan cuando el contenedor se detenga o elimine.

#### Explicación de Dockerfile de API Spring:

```
1 FROM openjdk:17-jdk-alpine
2 VOLUME /tmp
3 COPY target/*.jar app.jar
4 ENTRYPOINT ["java", "-jar", "/app.jar"]
5
```

1. **FROM openjdk:17-jdk-alpine**: Especifica la imagen base para el contenedor (openjdk:17-jdk-alpine).
2. **VOLUME /tmp**: Crea un volumen en el contenedor.
3. **COPY target/\*.jar app.jar**: Copia el archivo JAR de la aplicación al contenedor.
4. **ENTRYPOINT ["java", "-jar", "/app.jar"]**: Inicia la aplicación Java cuando el contenedor se inicie.



## Explicación de Dockerfile de Angular:

```
1 FROM node:18 as build
2 WORKDIR /app
3 COPY package*.json ./
4 RUN npm ci
5 RUN npm install -g @angular/cli
6 COPY . .
7 RUN npm run build --configuration=production
8
9 FROM node:18
10 WORKDIR /app
11 COPY --from=build /app /app
12 RUN npm install -g @angular/cli
13
14 EXPOSE 4200
15 CMD ["ng", "serve", "--host", "0.0.0.0"]
```

1. **FROM node:18 as build**: Usa la imagen node:18 como base
2. **WORKDIR /app**: Usa el directorio de /app
3. **COPY package\*.json ./**: Copia los archivos package.json y package-lock.json al directorio de trabajo del contenedor.
4. **RUN npm ci**: Instala las dependencias de la aplicación.
5. **RUN npm install -g @angular/cli**: Instala el Angular CLI.
6. **COPY . .**: Copia todos los archivos del directorio actual (en la máquina host) al directorio de trabajo del contenedor.
7. **RUN npm run build --configuration=production**: Construye la aplicación Angular en modo producción.
8. **FROM node:18**: Usa la imagen node:18 como base
9. **WORKDIR /app**: Establece el directorio de trabajo dentro del contenedor en /app.
10. **COPY --from=build /app /app**: Copia los archivos del directorio /app de la etapa build al directorio /app de esta etapa.
11. **RUN npm install -g @angular/cli**: Instala el Angular CLI.
12. **EXPOSE 4200**: Expone el puerto 4200 del contenedor.
13. **CMD ["ng", "serve", "--host", "0.0.0.0"]**: Ejecuta angular una vez iniciado el contenedor.



## Conclusión

He logrado realizar un proyecto que no pensaba que sería capaz, sobre todo debido a la dificultad que tenía para entender el TypeScript de Angular. Para mejorar mis habilidades con este Framework, decidí utilizarlo, y a mi parecer logré unos resultados mejores de los que me imaginaba. Este proyecto me abrió los ojos para ver la infinidad de cosas en las que se puede mejorar una aplicación web. Por mi parte, estoy contento con los resultados, y seguiré actualizando este proyecto