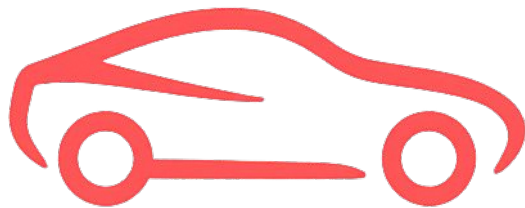




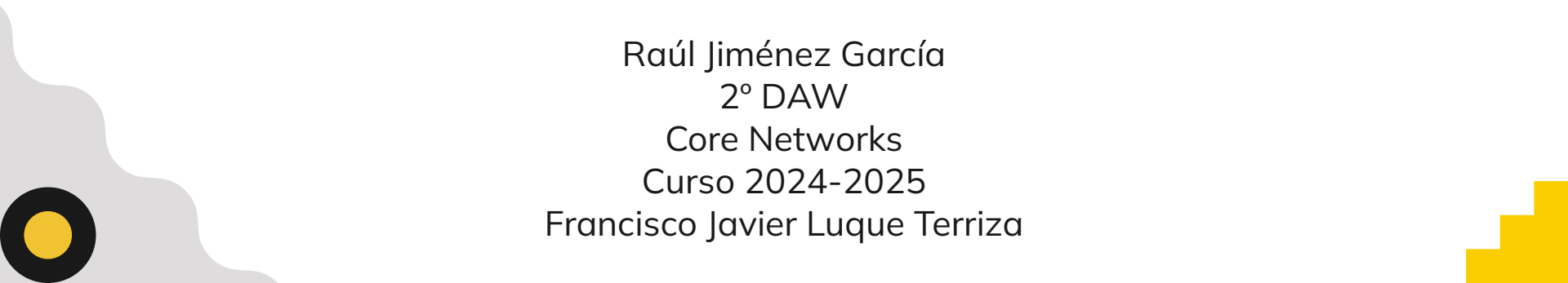
Cochemani



a



Raúl Jiménez García
2º DAW
Core Networks
Curso 2024-2025
Francisco Javier Luque Terriza



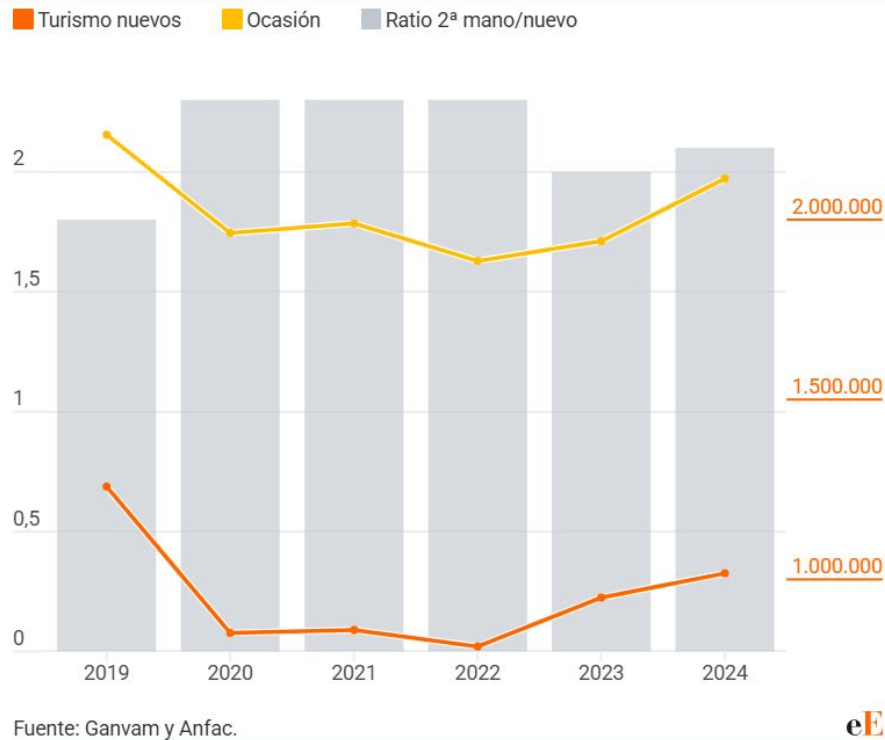
Índice

1. Resumen.....	3
2. Contextualización de problemas.....	4
3. Objetivos.....	5
4. Análisis de requisitos.....	6
5. Arquitectura del sistema.....	7
6. Base de datos y almacenamiento.....	8
7. Diseño de seguridad.....	9
8. Descripción funcional del sistema.....	10
9. Pruebas y validación.....	14
10. Líneas de mejora y evolución futura.....	15
11. Conclusiones.....	16

Resumen

Mi proyecto es una aplicación de compraventa y alquiler de vehículos. Esta está diseñada para su funcionamiento tanto en móvil como en web.

La razón por la que he decidido hacer una app sobre alquiler y venta de vehículos es porque la venta de vehículos de segunda mano y kilómetro 0 está en auge desde hace varios años.



Contextualización de problemas


Ya hay gran cantidad de aplicaciones para la venta y el alquiler de vehículos de segunda mano eso dificultará bastante el progreso de la aplicación en el mercado. Por eso he decidido crear una app que implemente ambas.

En muchas de las aplicaciones en las que los vendedores no son parte de la app sino que son vendedores particulares, existe el riesgo de la estafa ya que en muchas aplicaciones publicas el vehículo pero no requieren comprobación de titularidad del vehículo ni nada así. Por lo que pueden haber publicados vehículos robados con documentación falsa.

También está el riesgo de que la descripción, el estado y otras cosas no sean reales. Como por ejemplo alterar el cuentakilómetros para reducir los kilómetros indicados. Para ver si esto pasa es tan fácil como solicitar un informe a la DGT con el historial completo del vehículo.



Objetivos

- Mi objetivo principal es el de crear un sistema de gestión por el cual los propietarios puedan gestionar todo lo relacionado con sus vehículos. Que ellos tomen las decisiones por ejemplo con los alquileres, ellos son los que deben decidir si aprueban o rechazan.
 - Hacer que la conexión entre usuarios y propietarios sea rápida y segura de cara a las estafas en las ventas de vehículos.
 - Que los usuarios tengan una buena experiencia en la app, teniendo un diseño intuitivo y que permita en todo momento a los usuarios ser conscientes de lo que están haciendo, y con todas las facilidades para que puedan encontrar de manera fácil y rápida el vehículo que más se adapte a sus expectativas y necesidades.
- 

Análisis de requisitos

Los usuarios que entren a la aplicación podrán realizar búsquedas mediante filtros, acceder a la vista detallada con la información de los vehículos y registrarse.

Una vez registrados también podrán realizar reservas, escribir reseñas, crear alertas y publicar vehículos.

Si tienen vehículos publicados también podrán gestionar las reservas que hagan otros usuarios en su vehículo y gestionar las reseñas.

Los administradores gestionan la aprobación de vehículos por parte de usuarios, gestionan los usuarios y también las reservas y reseñas de todos los vehículos.

Arquitectura del sistema

El backend lo he desarrollado en Spring Boot usando Java como lenguaje, Spring Security para la autenticación y autorización, Spring Data JPA para la persistencia y el acceso a datos y Lombok para la reducción y limpieza de código.

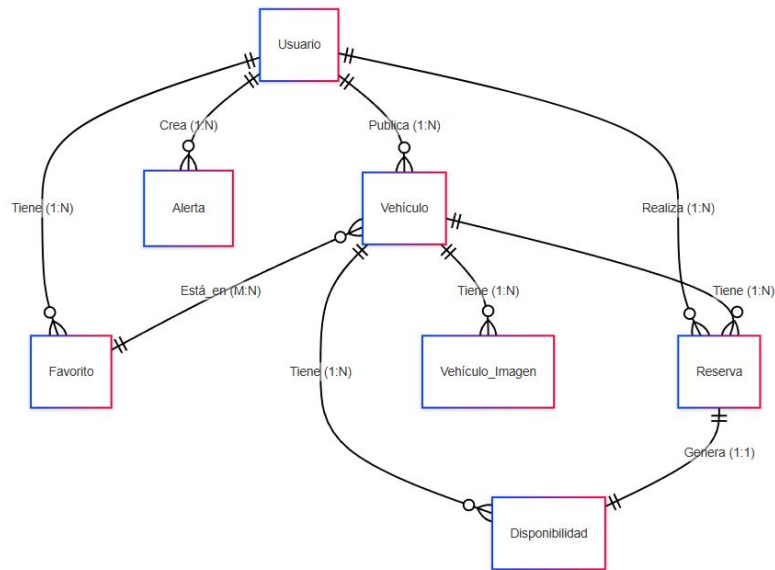
El frontend lo he hecho con React Native en Visual Studio usando JavaScript y CSS, implementando librerías como la de lucide para el uso de iconos dentro de la app e instalando dependencias de react como es "react-native-calendar" para el diseño del calendario para las reservas. Está configurado mediante el uso de '`Platform.OS === 'web'`' para configurar los estilos de web sin cambiar los usados para móvil.

Base de datos y almacenamiento

La base de datos está hecha en MySQL porque es fácil de conectar con el back.


Hay implementada eliminación en cascada para la eliminación de vehículos y de usuarios.

Las imágenes que se publican con la creación del vehículo se guardan de manera local en la misma carpeta donde se encuentran el back y el front. Las imágenes se guardan con un formato para distinguir el vehículo al que pertenece (veh_idvehiculo_idunico.jpg).





Diseño de seguridad

- Sistema de autenticación JWT.
 - Filtro para autenticar los tokens.
 - Uso de distintos roles para controlar el nivel de acceso.
 - Encriptación con BCrypt para las contraseñas.
 - Protección de endpoints.
 - Expiración de sesiones.
 - Documentación en swagger.
- 

Descripción funcional del sistema

Mi aplicación tiene un CRUD completo para usuarios y casi completo para vehículos, ya que no tengo implementada edición de vehículos.

Los usuarios registrados pueden:

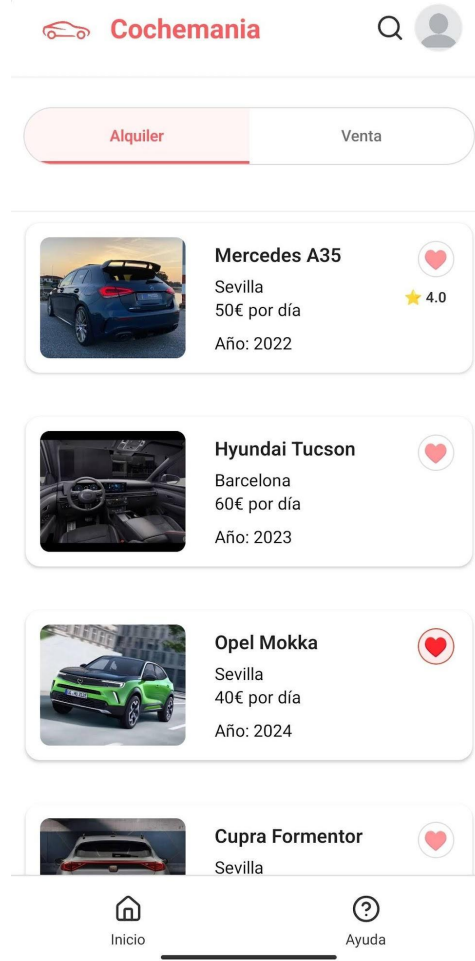
- Publicar vehículos.
- Buscar vehículos.
- Realizar reservas.
- Escribir reseñas.
- Añadir favoritos.
- Crear alertas.

Los propietarios pueden:

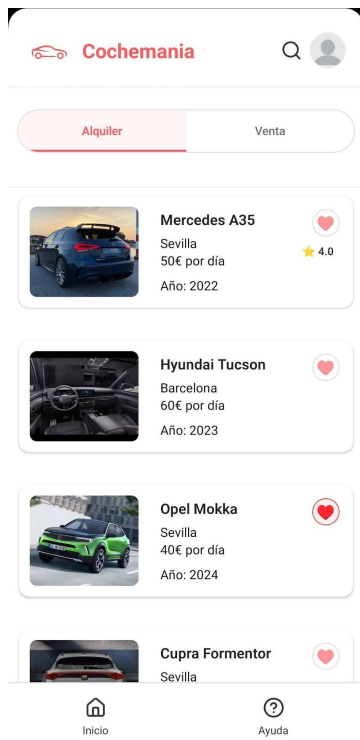
- Gestionar las reseñas de sus vehículos.
- Gestionar las reservas en sus vehículos.

Los administradores pueden:

- Gestionar todos los vehículos y lo que conlleva.
- Gestionar los usuarios.



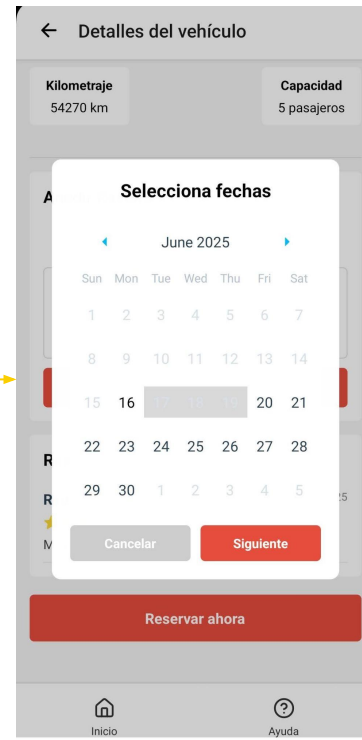
Flujo reserva



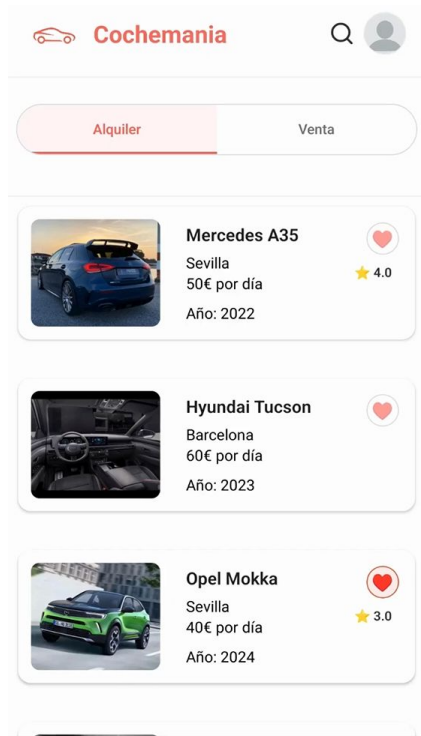
Accedemos a la
vista detallada
del vehículo



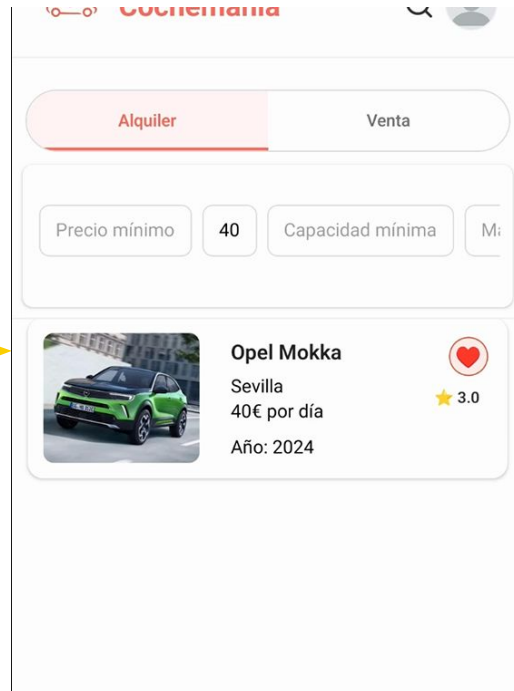
Clicamos en reservar
y seleccionamos las
fechas



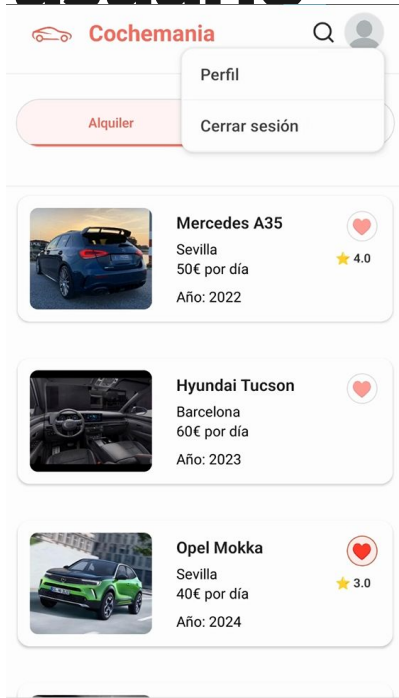
Búsqueda mediante filtros



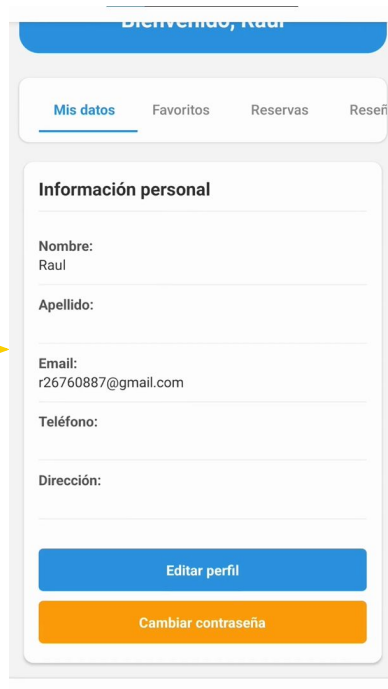
Rellenamos
los filtros



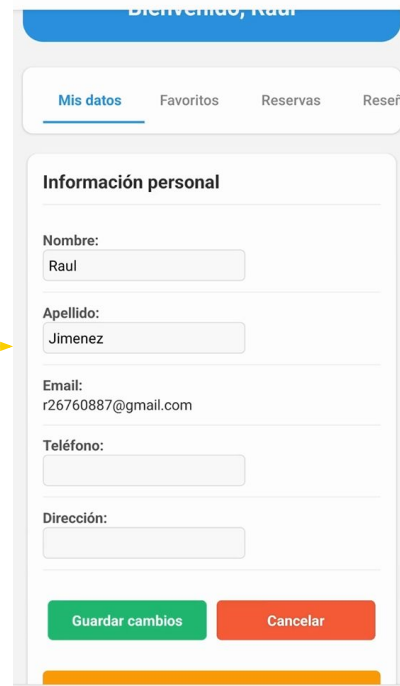
Edición de datos del usuario



Le damos a
editar perfil



Cambiamos
los datos



Pruebas y validación

- Pruebas unitarias: mientras desarrollaba el back, he ido haciendo pruebas para comprobar el funcionamiento desde Postman.
- Pruebas de integración: una vez ya sabía que funcionaban los endpoints del back, los iba implementando al front y hacía los respectivos cambios o creación de componentes y vistas necesarios.
- Pruebas funcionales: con todo ya implementado en el front, iniciaba la aplicación con Expo Go y comprobaba que funcionará sin errores.



Líneas de mejora y evolución futura

- Algo que sería muy importante implementar sería un sistema de chat en vivo entre usuarios y propietarios. Esto ayudaría muchísimo para el proceso de venta de vehículos.
- Como he dicho, me gustaría que fuese completamente segura la venta de vehículos, por eso la implementación de un sistema para verificar que el usuario que publica el vehículo sea el dueño del mismo. Para que el propietario tenga que subir un documento con su DNI o algún documento identificador y los papeles del coche.
- Un sistema de tasación de vehículos. Implementar un sistema que mediante scrapping sea capaz de darle un valor a tu vehículos siguiendo unas características.
- Implementación de un servidor de correo SMTP para el funcionamiento del sistema de alertas y notificaciones, y el restablecimiento de contraseñas.
- Un filtro de búsqueda para filtrar los vehículos de alquiler por los días disponibles.
- Modificar el alquiler de vehículos para que las reservas en los vehículos que estén publicados por administradores se aprueben automáticamente.

Conclusiones

Mi aplicación de compraventa y alquiler de vehículos funciona para web y móvil y creo que puede ser útil debido a la falta de aplicaciones que incluyen compraventa y alquiler de vehículos en una misma. Aunque mi aplicación tiene limitaciones y cosas que mejorar, creo que con el auge del mercado de vehículos de segunda mano puede ser útil.



Muchas gracias por su tiempo

Instalando Expo Go, podéis probar la app escaneando este QR
o poniendo la siguiente url `exp://192.168.157.164:8081`

