

# Especificaciones de Requisitos

Autor: Alejandro Santos Mejías  
Fecha última modificación: 2025/03/29  
Versión: 0.2.0  
Nombre del proyecto: “Rent a Car”

## Requisitos Funcionales

- **RF-1 Gestión de vehículos:** El sistema permitirá gestionar vehículos.
  - **RF-1.1 Recoger información de los vehículos:** La aplicación debe ser capaz de recoger información de los vehículos como categoría (turismo, motocicleta, furgoneta), tipo (pequeña, mediano, lujo), fecha de mantenimiento según vehículo (Tabla 1) y precio por día (cambiará según el resto de parámetros). Las furgonetas podrán ser de los subtipos estándar o gran carga.
  - **RF-1.2 Dar de alta:** Los administradores deben poder añadir vehículos a la base de datos.
  - **RF-1.3 Eliminar vehículo:** Los administradores deben poder eliminar vehículos a la base de datos.
  - **RF-1.4 Modificar información de los vehículos:** Los administradores deben poder modificar la información de los vehículos a la base de datos.
  - **RF-1.5 Insertar una nueva fecha de mantenimiento:** Los administradores deben poder modificar la fecha de mantenimiento de los vehículos en la base de datos.
  - **RF-1.6 Listar vehículos:** La aplicación debe ser capaz de listar todos los vehículos en la base de datos junto con su información disponible.
  - **RF-1.7 Consultar vehículo:** Los usuarios deben ser capaz de consultar la información de un vehículo en particular.
  - **RF-1.8 Inmutabilidad de los vehículos alquilados:** El sistema no permitirá modificar los vehículos alquilados.

Tipo Vehículo	Antigüedad	Mantenimiento
Coche y Motocicletas	Hasta 4 años	24 meses
	De 4 años en adelante	12 meses
Furgonetas	Hasta 2 años	12 meses
	De 2 años en adelante	6 meses

Tabla 1. Tabla resumen de los estandares de fechas de mantenimiento por vehículo.

- **RF-2 Gestión de usuarios:** La aplicación deberá poder gestionar distintos tipos de usuarios.
  - **RF-2.1 Roles de usuarios:** Los usuarios podrán ser de empleados, administradores y/o clientes.
  - **RF-2.2 Privilegios de los administradores:** Los administradores podrán dar de alta, eliminar y añadir vehículos, modificar fecha de mantenimiento y datos del vehículo.
  - **RF-2.3 Privilegios de los empleados:** Los empleados podrán realizar todas las funciones relacionadas con el alquiler de vehículos a clientes registrados.

- **RF-2.4 Actualizar datos personales:** Los usuarios podrán actualizar su información personal.
- **RF-3 Reglas de mantenimiento:** La disponibilidad para alquilar los vehículos se verá afectada por reglas de mantenimiento según tipo y antigüedad (Tabla 1).
  - **RF-3.1 Alerta de estado de fecha de mantenimiento:** Se alertará a los usuarios mediante colores aquellos vehículos con la fecha de mantenimiento caducada (rojo) o con caducidad próxima de 1 mes (naranja). Seguirán siendo visualizables en el listado de vehículos, pero no se podrán alquilar.
  - **RF-3.2 Fecha de mantenimiento caducada:** No se podrán alquilar vehículos con la fecha de mantenimiento vencidas.
- **RF-4 Gestión del alquiler de vehículos:** La aplicación debe poder gestionar y efectuar alquileres de coches.
  - **RF-4.1 Registro de clientes:** Los clientes deben estar registrados para poder alquilar vehículos.
  - **RF-4.2 Filtro por tipo:** Se podrá filtrar por tipos de vehículo disponibles a la hora de realizar el alquiler.
  - **RF-4.3 Vehículos alquilados:** No se podrán alquilar vehículos ya alquilados para una fecha concreta.
- **RF-5 Vistas:** La aplicación tendrá una vista de escritorio y una vista web.
  - **RF-5.1 Vista web:** se mostrará información general y se mostrará información sobre los vehículos.
    - **RF-5.1.1 Estructura página web:** El sitio web incluirá las siguientes secciones:
      - Inicio.
        - Información general de la empresa.
        - RSS con las últimas novedades de interés de la empresa.
      - Alquiler
        - Vehículos disponibles
        - Vehículos alquilados
      - Contacto.
        - Situación geográfica de la empresa.
        - Datos de contacto.
        - Formulario (que deberá comprobar su correcta cumplimentación)
    - **RF-5.1.2 Filtro por tipos:** El sitio web permitirá filtrar por tipo de vehículo disponible.
    - **RF-5.1.3 Información del cliente:** El sitio web mostrará los vehículos alquilados con la información de los clientes que lo tienen en uso.
  - **RF-5.2 Vista escritorio:** Proporcionará una interfaz para los administradores y empleados.

## Requisitos No Funcionales

- **RNF-1 Arquitectura:** Se dividirán las clases en paquetes con bajo acoplamiento y alta cohesión.
- **RNF-2 Metodología de desarrollo:** Se utilizará la metodología scrum con sprints de 1 semana de duración para el desarrollo del proyecto.

- **RNF-3 Control de versiones:** Se realizará el control de versiones del proyecto mediante Git, el proyecto se alojará en un repositorio de Github, controlando las tareas pendientes mediante un Kanba.
- **RNF-4 Calidad de código:** Se utilizará SonarQube para evaluar la calidad del código, la aplicación final debe tener evaluación “passed”.
- **RNF-5 Pruebas de software:** Se realizará un plan pruebas que cubra todo el código.
  - **RNF-5.1 Salidas y entradas esperadas:** Se indicará en el plan de pruebas tanto la salida esperada como la real y si la prueba fue correcta o incorrecta.
  - **RNF-5.2 Tecnología para las pruebas:** Se utilizará JUnit para realizar las pruebas.
  - **RNF-5.3 Estructura:** Las clases de pruebas se guardarán en un paquete diferente al código fuente.
- **RNF-6 Documentación:** Se utiliza JavaDoc para generar comentarios sobre las clases implementadas en el proyecto. Los comentarios generados por JavaDoc se guardará en un paquete diferente al código fuente. Se comenta correctamente el código JavaScript.