

INTRODUCCIÓN

El documento forma parte del desarrollo del proyecto “**Rent a Car**”, el objetivo es crear una aplicación informática para gestionar el alquiler de vehículos en una empresa dedicada a este sector.

La solución permitirá a los usuarios, consultar la disponibilidad de vehículos, realizar alquileres y acceder a información general desde una plataforma web. Además, los empleados de la empresa, tanto administradores como usuarios, podrán gestionar el inventario de vehículos, controlar los mantenimientos, llevar un seguimiento de los alquileres y administrar los datos de los clientes.

Este documento tiene como finalidad definir de forma clara los requisitos funcionales y no funcionales del sistema.

REQUISITOS FUNCIONALES (RF)

1. Consulta y búsqueda de vehículos

- RF01 - Consulta de vehículos disponibles:
 - Los clientes registrados pueden ver qué vehículos están disponibles en la web.
 - Sólo se mostrarán los que no estén alquilados ni en mantenimiento.
- RF02 - Filtros de búsqueda por tipo de vehículo:
 - Se podrá filtrar por tipo de vehículo:

Turismo

Motocicleta

Furgoneta

- También habrá subcategorías:

Turismo de categoría alta

De 3 puertas

De 5 puertas

Turismos de categoría media

De 3 puertas

De 5 puertas

Turismos de categoría baja

De 3 puertas

De 5 puertas

Furgoneta de gran carga

Furgonetas

Motocicletas de 49 cc

Motocicletas de 125cc

Motocicletas de mas de 125 cc

2. Gestión de alquileres

- RF03 - Alquiler de vehículo:
 - Un cliente registrado podrá alquilar un vehículo si está disponible.
 - No se puede alquilar si está en mantenimiento.
 - Podrá dar de baja un vehículo ya alquilado

3. Administración de vehículos

- RF04 - Alta de nuevos vehículos:
 - Los administradores podrán registrar vehículos con datos como:

Categoría y tipo

Precio por día

Placa, modelo, año

Estado y disponibilidad

- RF05 - Modificación de vehículos:
 - Un administrador podrá modificar la información de un vehículo.
 - No podrá hacerlo si el vehículo está alquilado.

4. Mantenimiento de vehículos

- RF06 - Registro de mantenimientos:
 - Se apuntarán las fechas de mantenimiento.

- Se verán los tipos de mantenimiento realizados.
- RF07 - Advertencias visuales de mantenimiento:
 - Los vehículos con mantenimiento cercano tendrán advertencias:

Naranja: Próximo mantenimiento.

Rojo: No se puede alquilar hasta mantenimiento.

5. Gestión de usuarios y empleados

- RF08 - Registro de clientes:
 - Todo usuario debe registrarse antes de alquilar un vehículo.
 - Registrar nuevos vehículos
 - Alquilar , comprar vehículos
 - Vender o alquilar vehículos

RF10 - Gestión de roles de empleados:

Existen dos tipos de empleados:

- Básicos: Pueden gestionar clientes.
- Administradores: Pueden gestionar vehículos, precios y mantenimiento

REQUISITOS NO FUNCIONALES (RNF)

1. Rendimiento y disponibilidad

RNF01 - Tiempo de respuesta:

La plataforma debe responder a cualquier solicitud del usuario en un tiempo máximo de 2 segundos bajo condiciones normales de uso.

RNF02 - Disponibilidad del sistema:

El sistema estará disponible un 99.5% del tiempo durante el año, con posibles mantenimientos programados fuera del horario laboral.

2. Usabilidad

RNF03 - Interfaz amigable:

La aplicación web deberá contar con una interfaz intuitiva y fácil de usar, tanto para clientes como para empleados.

RNF04 - Accesibilidad:

El sistema debe ser accesible desde cualquier navegador moderno y adaptarse a distintos dispositivos (ordenador, tablet, móvil).

3. Seguridad

RNF05 - Autenticación de usuarios:

Todos los usuarios deben autenticarse con credenciales válidas para acceder a funcionalidades personalizadas.

RNF06 - Protección de datos:

Los datos personales de los clientes estarán protegidos y se manejarán conforme a la normativa de protección de datos vigente (por ejemplo, GDPR).

RNF07 - Control de acceso:

Cada tipo de usuario (cliente, empleado básico, administrador) tendrá acceso únicamente a las funciones que le correspondan según su rol.

4. Mantenimiento y escalabilidad

RNF08 - Escalabilidad del sistema:

La arquitectura debe permitir la incorporación de nuevas funcionalidades sin afectar el rendimiento del sistema.

RNF09 - Facilidad de mantenimiento:

El código fuente debe seguir buenas prácticas de programación, estar documentado y ser modular para facilitar futuras modificaciones.