

# Manual de Proyecto

## Requerimientos previos a la instalación

- Conexión a internet
- Un Servidor, en nuestro caso usaremos el facilitado por el centro (PROXMOX)
- Una máquina virtual desplegada en el servidor

# 1. Definición del proyecto

## 1.1 Obtención de la información necesaria para la definición del proyecto.

La información del proyecto ha sido facilitada por varios concesionarios de automoción. La idea surge de mi experiencia personal en el sector y las escasas herramienta con la que disponíamos, la idea es facilitar la publicación y gestión de anuncios en diversas páginas web, facilitando así una galería y un sistema de seguimiento de cliente para potenciar la visibilidad y ventas de una empresa dedicada a este tipo de actividad.

## 1.2 Descripción detallada del proyecto.

Se trata de una aplicación destinada a la gestión de un concesionario donde el empresario o sus operarios podrán gestionar los vehículos y los datos de los clientes además de una rápida visualización y publicación en la web, también permitirá generar informes de clientes o vehículos dependiendo de determinados filtros que elijamos facilitando así el seguimiento de vehículos y clientes para una mayor eficiencia en el departamento de marketing.

### 1.3 Ámbito del proyecto.

El proyecto busca la creación de una aplicación que sirva para centralizar y facilitar la gestión de concesionario, permitiendo la asignación de clientes a vehículos y generando informes para su posterior uso, como la gestión y publicación en la web de dicho concesionario facilitando dicho proceso.

### 1.4 ¿Adaptación o Creación?

El proyecto se trata de una adaptación con algunos matices de herramientas ya obsoletas de gestión en el mundo de la automoción aunando la publicación en la web con la gestión en cualquier parte de nuestro concesionario.

### 1.5 Conceptos básicos del proyecto software

- Framework: Es una estructura conceptual y tecnológica de soporte definido, normalmente con módulos de software concretos, que pueden servir de base para la organización y desarrollo de software.
  - TypeScript: Es un lenguaje de programación libre y de código abierto desarrollado y mantenido por Microsoft, se trata de un súper conjunto de JavaScript.
  - Angular: Es un framework para aplicaciones webs desarrollado en TypeScript, de código abierto que se utiliza para crear y mantener aplicaciones web de una sola página.
  - Spring Framework: se trata de un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java.
- Popup: Se trata de ventanas emergentes que genera la aplicación, con el fin de dar información, realizar confirmaciones o avisar al usuario.
- Base de datos Relacional: es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

## 1. Planificación del proyecto

### 1.6 Definición de tareas

<i>Definición de la idea de proyecto.</i>	<i>Definición de la idea de proyecto, que es lo que se pretende lograr que la aplicación haga la aplicación.</i>
<i>Alcance del proyecto.</i>	<i>Se fija el ámbito sobre el cual se va a trabajar.</i>
<i>Stack tecnológico.</i>	<i>Consiste en un listado de todas las tecnologías que se van a emplear en planificación,</i>
<i>Listado de entidades y atributos.</i>	<i>Listado de todas las entidades que formarán parte del modelo de datos y cuáles serán sus atributos.</i>
<i>Especificación de requisitos.</i>	<i>Se crean los requisitos que va a tener que cumplir el proyecto, tanto los funcionales como los no funcionales.</i>
<i>Mapa del sitio.</i>	<i>Se realiza una descripción de las pantallas necesarias.</i>

<i>Realización de maqueta.</i>	<i>Empleando el software Axure, se realiza un MockUp con el aspecto que se desea que tenga la aplicación.</i>
<i>Modelado de datos en Backend</i>	<i>Trabajaremos con spring boot, modelaremos las clases con sus respectivas anotaciones para el mapeado a BBDD</i>
<i>Creación de repositorios y servicios</i>	<i>repositorio y servicios que incluirá todos los métodos que se van a emplear en la aplicación.</i>
<i>Creación de los controladores</i>	<i>Seran los encargados de gestionar las rutas de entrada a nuestra app</i>
<i>Implementación de seguridad</i>	<i>Implementaremos la seguridad con JWT (JSON Web Token)</i>
<i>Creacion de la base de datos.</i>	<i>Creación de datos en la base de datos relacional empleada.</i>
<i>Desplegar Angular</i>	<i>Despliegue e instalacion de los necesario para usar angular</i>
<i>Implementacion dependencias PrimeNg</i>	<i>PrimeNG es una colección de componentes de interfaz de usuario para Angular.</i>

<i>Implementacion dependencias Bootstrap</i>	<i>Para la maquetacion y gestion de componentes.</i>
<i>Maquetado Responsive</i>	<i>Correccion del maquetado responsive</i>
<i>Despliegue de la aplicación.</i>	<i>Despliegue de la aplicaicon en proxmox</i>
<i>Correccion de errores despues de la puesta en despliegue</i>	<i>Correccion de errores de ruta, maquetacion, etc...</i>
<i>Creación de la documentación</i>	<i>Creacion de la documentacion necesaria para la correcta interpretacion de la aplicación.</i>

### 1.7 Estimación de tiempos

La estimación de los tiempos aproximada ha sido:

Planificación inicial: 50H

Codificación BackEnd & FronEnd: 160H

Despliegue: 15h

Documentación: 10h

# 1. Análisis del Proyecto

## 1.8 Requisitos técnicos de la web

- Desarrollo de BackEnd

Para esta parte del desarrollo se ha empleado el lenguaje de programación Java y hemos trabajado en el entorno de desarrollo Spring Boot Tool Suit que es un framework de eclipse, llevando acabo todo el desarrollo del código.

- Desarrollo de FrontEnd

La parte de FrontEnd se ha desarrollado en el entorno Visual Studio Code, empleando lenguaje de TypeScript con el framework Angular 10, para el diseño nos hemos apoyado en HTML y CSS, mientras que para la maquetación hemos utilizado Boostratp

Hemos agregados módulos de la interfaz PrimeNg para facilitar el trabajo de determinados puntos ya que nos ofrece una gran potencia de trabajo.

- Base de Datos

Para la persistencia de datos hemos utilizado MySQL, y para su más sencillo uso y visualización lo hemos complementado con la herramienta MySQL Workbench .

- Gestor de versiones



Durante el desarrollo del proyecto se ha realizado un control de versiones, empleando GitHub.

- Gestor de versiones

Se ha llevado a cabo el despliegue de la aplicación en una máquina virtual Proxmox.

## 1.9 Elementos del Contenido

- Persistencia de datos

La persistencia la logramos mediante MySQL , con el uso de hibernate logramos mediante anotaciones crear las correspondientes tablas en la base de datos, la inserción de los datos se pudo realizar bien desde la misma lógica de negocio o mediante script en el entorno de Workbench.

- Lógica de negocio

La lógica de negocio se ha realizado con Spring boot, el lenguaje de programación elegido es Java y para la gestión de dependencias el proyecto se desarrolla empleando Maven.

- Vista de negocio

Se ha empleado mayormente el framework de TypeScript Angular 10, apoyándonos en CSS, Bootstrap 4 y PrimeNg

### 1.10 Diseño grafico

Para el diseño gráfico nos hemos basado en los colores de una determinada marca de coches, ya que cuadraban con nuestros objetivos más próximos a la hora de ofertar nuestro producto.

Los colores elegidos son el azul oscuro que otorga profesionalidad y seriedad, mientras que hemos combinado con algunos tonos claro para dar un toque de innovación a la web intentando simular led de vehículos.

El tipo de letra es también de la determinada marca de vehículos pues pensamos que es un tipo de letra elegante y profesional.

En futuras actualizaciones se plantea desde un panel de control que el usuario pueda elegir entre dos tipos de temas “oscuro” y “claro” para sí hacer la experiencia del usuario más personalizable.

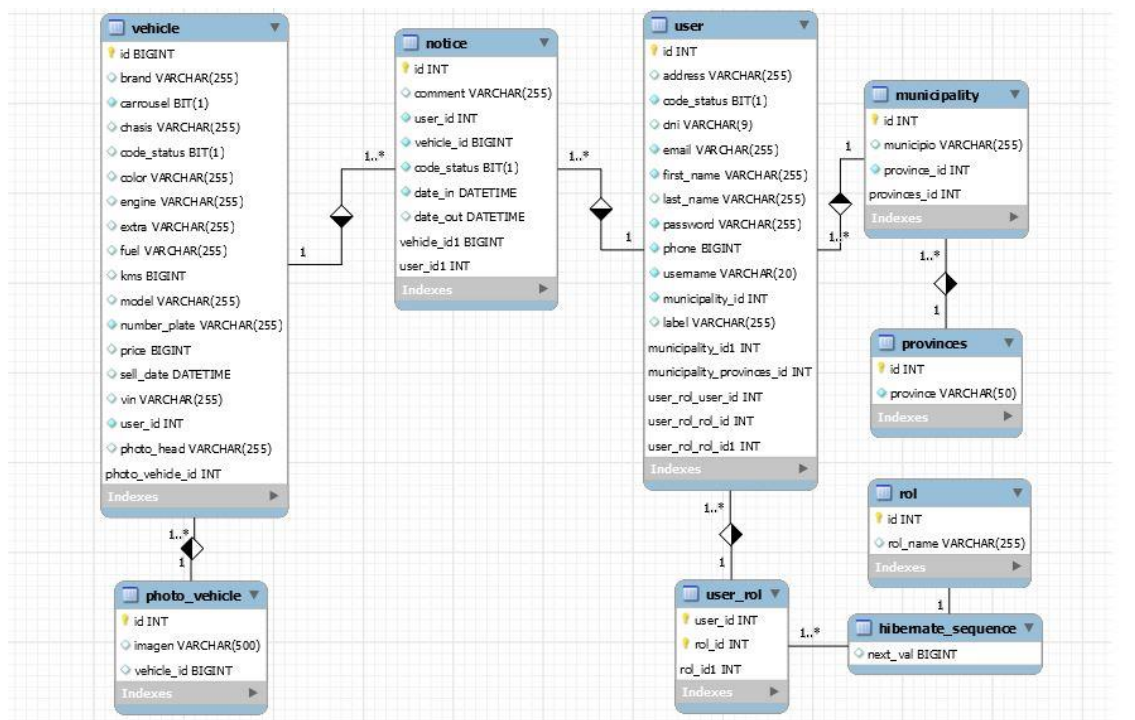
### 1.11 Diseño del software

Se ha decido utilizar Angular para modelización mas el código en la parte del FrontEnd con la intención de hacerlo reutilizable para otros proyectos, mientras que el parte del BackEnd se ha modelización tanto los repositorios, servicios y controller con la misma intención, teniendo incluso separada la seguridad para poder cambiarla por otra en un

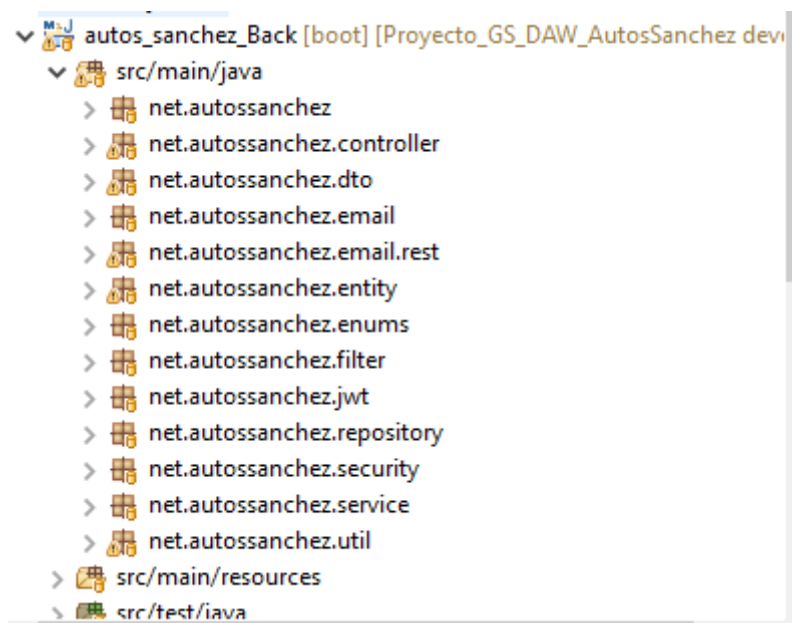
futuro, como también la implementación de adapter para el mapeo de DTO para una mayor seguridad.

Por otra parte, el utilizar PrimeNg nos permite cambiar módulos por otro y conservar las mismas listas para mostrar en pantalla, haciendo la aplicación muy configurable a futuros cambios.

### 1.13 Diseño de datos.



### 1.14 Diseño del sistema



Net.autossanchez: La main de nuestro proyecto, desde donde iniciamos la aplicación.

Net.autossanchez.controller: Gestiona los accesos y los servicios que llegan desde el FrontEnd.

Net.autossanchez.dto: Sirven para crear objetos planos que contienen información de múltiples fuentes.

Net.autossanchez.email: el main de un micro aplicación para el envío de email.

Net.autossanchez.email.ret: gestiona la micro aplicación para el envío de email.

Net.autossanchez.entity: los modelos de las distintas clases.

Net.autossanchez.filter: modelización de los filtros de las búsquedas siendo estos en si modelos para los dto.

Net.autossanchez.jwt: clases para la gestión de la seguridad de JSON Web Token.

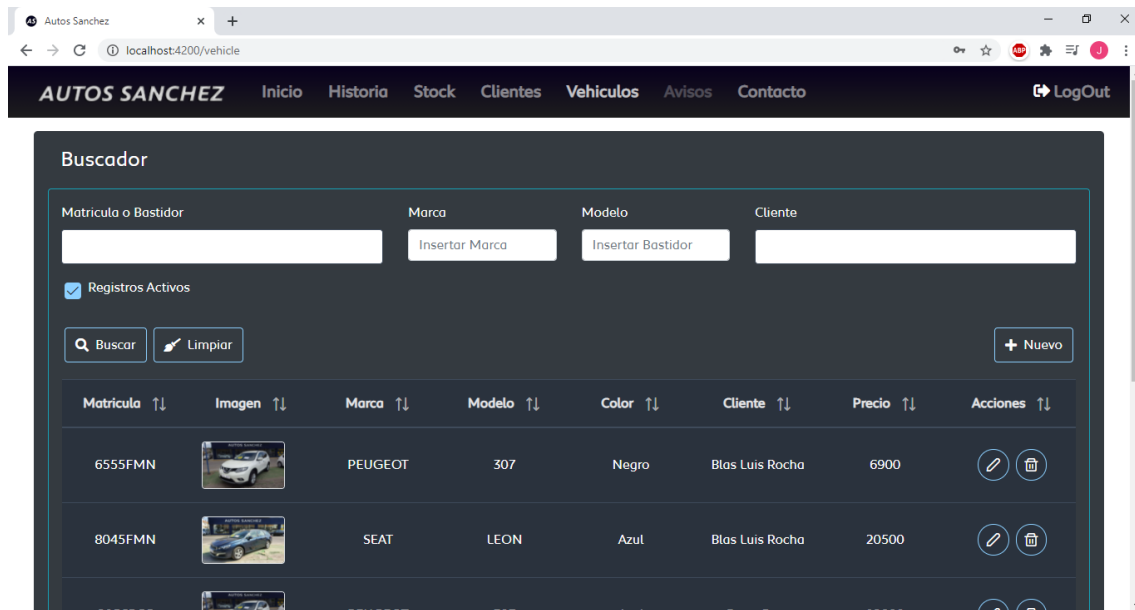
Net.autossanchez.repository: los distintos repositorios para su implementación en los servicios implementando así JpaRepository para mayor comodidad.

Net.autossanchez.security: Recoge todo lo relativo a la seguridad implementada en el aplicativo mediante el empleo de JSON Web Token.

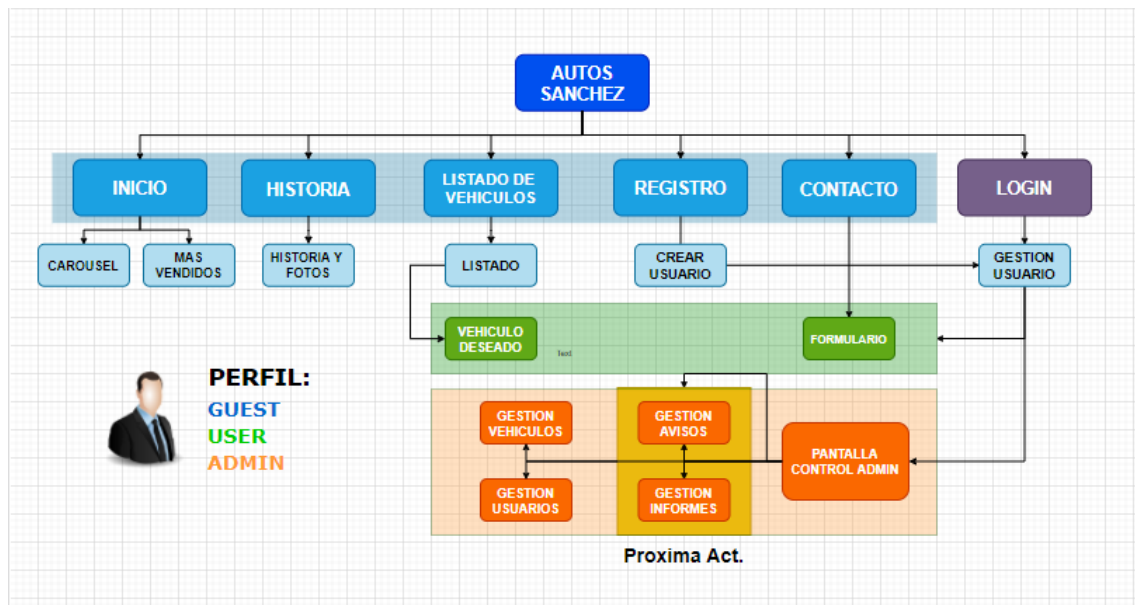
Net.autossanchez.service: Sirve como intermediario entre los repositorios y los controladores.

Net.autossanchez.util: Sirve como almacenamiento de clases útiles

### 1.15 Diseño de la interfaz de usuario



## Mapa del sitio



### 1.16 Diseño del contenido

El contenido corre a cargo del usuarios, tanto los usuarios normales como los usuarios empresa, los primeros para la búsqueda de vehículos e inserccion de sus datos de contacto, mientras que los de empresa introducirán los distintos vehículos que se expondrán en la web y distintas casas de anuncios, en futuras actualizaciones se implementaran también avisos y generación de informes que también serán manejada por estos últimos.

### 1.17 Diseño del contenido

Para el diseño de contenido multimedia se ha hecho uso de adobe Photoshop y imágenes cedidas por una empresa local.

### 1.18 Identificacion y cuantificación de contingencias.

Para el diseño de contenido multimedia se ha hecho uso de adobe Photoshop y imágenes

- Implementar la seguridad del JWT de JSON Web Token.
- Para el almacenamiento de imágenes subidas por el usuario, para resolverlo se optó por usar almacenamiento en Firebase.

### 1.19 Aseguramiento de calidad

Con el firme propósito de mejorar la calidad del dispositivo, durante toda la etapa de desarrollo se han llevado a cabo numerosas pruebas para la detección de errores, así como se han empleado pruebas para comprobar la seguridad del sistema.