

---

# CHARGING POINTS

---



20 DE MAYO DE 2022

JOSE MARIA TENREIRO EIRANOVA 2º DAW

### **AGRADECIMIENTOS**

Para mí este proyecto significa la continuación de mi gran proyecto de vida y de las innumerables experiencias, que me han hecho crecer de forma exponencial y, por otro lado, supone para mí el comienzo de una nueva etapa en otro ámbito profesional.

Durante la realización de este ciclo superior he aprendido a valorar mucho más a mis seres queridos, amigos y todas aquellas personas que han sido un constante apoyo para llegar al final de esta etapa y a quienes indiscutiblemente dedico este símbolo de esfuerzo tan importante para mí.

Primero antes de nada se lo dedico a mis padres, por ser las personas con las que he contado, cuento y contaré incondicionalmente durante toda mi vida.

A mi madre, soy consciente de aquella espina clavada por no terminar mis estudios superiores en su momento, quiero que sepas que durante toda mi vida siempre te escuché y aprovecho la oportunidad para agradecerte toda mi formación, mil gracias por existir.

A mi padre, por su educación diaria, de esas que no se aprende en ninguna escuela, la persona en la que me he convertido es gracias a ti, esfuerzo y dedicación son materias aplicables a cualquier profesión, quiero que sepas que para mí siempre serás inalcanzable.

A mis hermanas, por sus consejos en todos los momentos estos dos años, escuchar su voz muchas veces me ha levantado el ánimo, aunque ellas no lo sepan, han sido un apoyo importante.

A todo el profesorado de Edix, han puesto la primera piedra en el camino para el comienzo de este proyecto de vida. Sin duda, mucho que aprender aun, pero los cimientos de la estructura sin duda son gracias a ellos.

Por último y no menos importante, sino el agradecimiento más fundamental es para ti, Ariadna, mi pareja:

Solamente tú y yo sabemos la montaña rusa de emociones que hemos vivido en estos dos años, en todos los ámbitos, tanto profesionalmente como personalmente. Tu gran apoyo ha sido fundamental para poder alcanzar este reto. Tú has sido mi confidente en todos los momentos, cuando me ilusionaba con un triunfo y cuando me frustraba con una derrota. Soy consciente de todo el tiempo que nos he robado, espero poder devolvértelo

próximamente. Sin duda alguna, sin ti, no estaría escribiendo la memoria de este proyecto. Gracias infinitas por tu amor incondicional.

**Gracias a todos**, os agradezco vuestro apoyo y participación en el proyecto final de ciclo, de todo corazón,

Jose Maria Tenreiro Eiranova

**ÍNDICE****CAPÍTULO 1 – DESARROLLO DE LA IDEA****1.1-INTRODUCCIÓN**

<i>1.1.1 – INTERACCIÓN</i> .....	Pág 6
<i>1.1.2 – SOSTENIBILIDAD</i> .....	Pág 6
<i>1.1.3 – BARRERAS</i> .....	Pág 7
<i>1.1.4 – PROPOSITOS</i> .....	Pág 7
<i>1.1.1 – LEGACY</i> .....	Pág 8
<i>1.1.1 – EJEMPLOS</i> .....	Pág 8
<i>1.1.1 – RAPIDEZ EN LA CARGA</i> .....	Pág 9

**1.2-OBJETIVO DEL PROYECTO**

<i>1.2.1 – OBJETIVO DEL PROYECTO</i> .....	Pág 10
--	--------

**1.3-ESTUDIO DE MERCADO**

<i>1.3.1 – ESTUDIO DE MERCADO</i> .....	Pág 11
<i>1.3.2 – SISTEMAS EXISTENTES EN EL MERCADO</i> .....	Pág 12

**1.4-PALABRAS CLAVE**

<i>1.4.1 – CONECTOR SHUCKO</i> .....	Pág 13
<i>1.4.2 – CARGA LENTA</i> .....	Pág 13
<i>1.4.3 – CARGA SEMI-RÁPIDA</i> .....	Pág 14
<i>1.4.4 – CARGA RAPIDA O CHADEMO</i> .....	Pág 15

**CAPÍTULO 2 –FORMACIÓN APLICADA AL PROYECTO****2.1-MÓDULOS APLICADOS**

<i>2.1.1 - BASES DE DATOS</i> .....	Pág 17
<i>2.1.2 – PROGRAMACIÓN</i> .....	Pág 17
<i>2.1.3 – ENTORNOS DE DESARROLLO</i> .....	Pág 17
<i>2.1.4 – LENGUAJE DE MARCAS</i> .....	Pág 17
<i>2.1.5 – DESARROLLO WEB EN ENTORNO CLIENTE</i> .....	Pág 17
<i>2.1.6 – DESARROLLO WEB EN ENTORNO SERVIDOR</i> .....	Pág 17
<i>2.1.7 – DISEÑO DE INTERFACES</i> .....	Pág 17

2.1.8 – DESPLIEGUE DE APLICACIONES WEB.....	Pág 17
---	--------

## 2.2-MODELO DE DATOS UTILIZADO

2.1.1 – MODELO VISTA CONTROLADOR(MVC) .....	Pág 19
---	--------

## 2.3-TECNOLOGÍAS

2.3.1 – JAVA EE.....	Pág 21
2.3.2 – JSP.....	Pág 22
2.3.3 – HTML5.....	Pág 22
2.3.4 – CSS.....	Pág 23
2.3.5 – JAVASCRIPT.....	Pág 22
2.3.6 – MYSQL.....	Pág 24
2.3.7 – GIT.....	Pág 24
2.3.8 – JAVADOC.....	Pág 25

## 2.4 – FRAMEWORKS

2.4.1 – SPRINGBOOT.....	Pág 27
2.4.2 – BOOTSTRAP.....	Pág 27
2.4.3 – SWEET ALERT.....	Pág 28

## 2.5 – HERRAMIENTAS UTILIZADAS

2.5.1 – ECLIPSE.....	Pág 29
2.5.2 – MYSQL WORKBENCH.....	Pág 30
2.5.3 – GIT BASH.....	Pág 31
2.5.4 – GIT HUB.....	Pág 32

## CAPÍTULO 3 – ESTRUCTURA DEL PROYECTO

### 3.1 – CASOS DE USO

3.1.1 – DEFINICIÓN.....	Pág 33
3.1.2 – CASOS DE USO USUARIO ADMINISTRADOR.....	Pág 34
3.1.3 – CASOS DE USO USUARIO CLIENTE.....	Pág 34
3.1.4– CASOS DE USO USUARIO EMPRESA.....	Pág 35

### 3.2 – WIREFRAME

3.2.1 – DEFINICIÓN.....	Pág 36
3.2.2 – WIREFRAME PRINCIPAL.....	Pág 37
3.2.3 – WIREFRAME LOGIN.....	Pág 37
3.2.4 – WIREFRAME PANTALLAS USUARIO.....	Pág 37

### 3.3 – BASE DE DATOS

3.3.1 – DEFINICIÓN.....	Pág 38
3.3.2 – SCHEMA BASE DE DATOS PROYECTO.....	Pág 40

## **CAPÍTULO 4 – FUNCIONAMIENTO DE LA APLICACIÓN**

### 4.1 – PANTALLA PRINCIPAL

4.1.1 - LOGIN Y REGISTRO.....	Pág 41
-------------------------------	--------

### 4.2 – USUARIO ADMINISTRADOR

4.2.1 – FUNCIONES USUARIO ADMINISTRADOR.....	Pág 43
--	--------

### 4.3 – USUARIO CLIENTE

4.3.1 – FUNCIONES USUARIO CLIENTE.....	Pág 44
--	--------

### 4.4 – USUARIO EMPRESA

4.4.1 – FUNCIONES USUARIO EMPRESA.....	Pág 48
--	--------

## **CAPÍTULO 5 – CONCLUSIONES Y FUTURAS MEJORAS**

5.1 – CONCLUSIONES .....	Pág 50
--------------------------	--------

## **CAPÍTULO 6 - BIBLIOGRAFIA Y ANEXOS**

6.1 – ANEXOS Y GITHUB.....	Pág 52
----------------------------	--------

### **CAPITULO 1 – DESARROLLO DE LA IDEA**

#### **INTRODUCCIÓN**

##### **INTERACCIÓN**

Este proyecto consiste en el desarrollo de una aplicación web destinada para la creación de una plataforma en donde habrá una interacción entre proveedores de puntos de recarga para vehículos eléctricos y usuarios que necesiten esas recargas.



##### **SOSTENIBILIDAD**

En el marco de un sector transporte caracterizado por un elevado consumo de combustibles fósiles y un fuerte impacto ambiental, la promoción del coche eléctrico se presenta como una de las vías esenciales para conseguir una eficiente transición energética y, por el lado de la oferta, para avanzar hacia la sostenibilidad del sector.



# BARRERAS

En la actualidad el coche eléctrico se encuentra con diferentes barreras que hacen que sus compradores sean escépticos a la hora de decidirse a adquirir uno.

Una de sus mayores barreras es la autonomía y los consiguientes viajes largos que puede hacer el usuario, debido a que existen pocos puntos de carga en la actualidad en la red de carreteras estatales.



## PROPOSITOS

Charging Points nace con el propósito de facilitar este proceso. Una plataforma en donde usuarios buscan un punto de carga próximo a su trayectoria en un viaje y usuarios que facilitan estos puntos de carga, de ahora en adelante a estos últimos los denominaremos Empresa.





### LEGACY



Charging Points **no vende sus servicios directamente a los usuarios** de vehículos eléctricos ni opera ningún punto de carga. Ofrece una plataforma de interoperabilidad de empresa a usuario final. Trabaja desde dos perspectivas principales; los poseedores de puntos de carga (Estaciones) y los usuarios que necesitan un punto de carga en su trayectoria (Clientes).

### EJEMPLOS



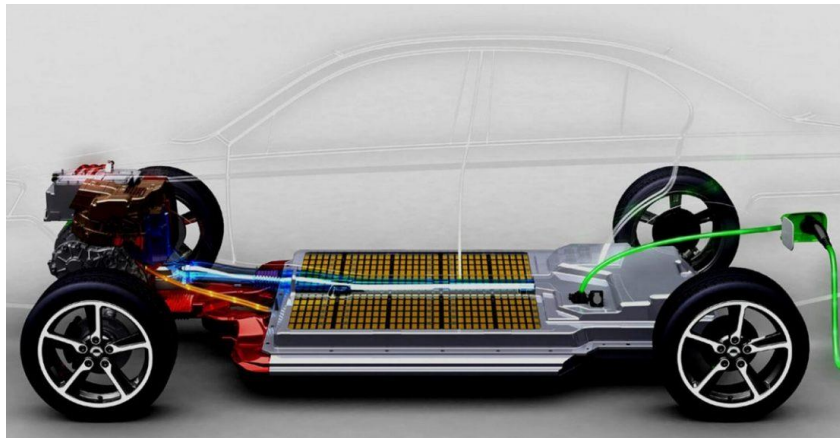
Un ejemplo de Empresa suministradora con un punto de carga podría ser tan simple como una casa de campo que decide instalar un punto de carga. De esta forma, se convertiría en suministrador. Si este punto se encuentra en un punto estratégico como puede ser en la salida de una autopista, un usuario que hiciera el trayecto por ejemplo Barcelona-Cádiz de más de 1000 kilómetros podría contar con multitud de puntos de carga para su vehículo sin preocuparse de la autonomía.

## CHARGING POINTS



Otro ejemplo podría ser un restaurante en las cercanías de una autopista o carretera convencional. Con la instalación de un punto de carga eléctrico y su posterior registro en Charging Points podría dar un impulso a su negocio atrayendo a múltiples clientes que aprovecharían su punto de carga para recargar su vehículo mientras consumen el resto de sus servicios de restauración.

### **RAPIDEZ EN LA CARGA**



Además, hoy en día existen muchas opciones de carga rápida que consiguen cargar el 80% de la batería de los vehículos en tiempo record. Con la plataforma Charging Points, se crearían cientos de nodos de recarga para vehículos eléctricos que terminarían con la barrera del miedo a la autonomía de un coche eléctrico en un trayecto largo.

### **OBJETIVO DEL PROYECTO**



Con la realización del proyecto Charging Points se pretende alcanzar los siguientes objetivos:

- Que un usuario cliente, sea capaz de registrarse en la plataforma, añadir su vehículo con su conector correspondiente, conseguir hacer una reserva de un punto de carga para una franja horaria (Mañana o Tarde) en un día que el mismo selecciona. Cancelar reservas si lo cree conveniente. Eliminar su vehículo y añadir otro. Consultar sus datos y modificarlos. Ver todas sus reservas pendientes y terminadas.
- Que un usuario empresa, sea capaz de registrarse en la plataforma, añadir los conectores con los que contará su estación de carga. Aumentar conectores o eliminarlos. Cancelar reservas de clientes. Consultar lista de clientes.
- Que un usuario administrador, sea capaz de realizar el mantenimiento de la plataforma. Eliminar usuarios, añadir nuevos administradores, consultar reservas y consultar usuarios registrados.

### ESTUDIO DE MERCADO

En este estudio de mercado desarrollamos una investigación para analizar la viabilidad comercial y económica del proyecto empresarial. Se trata fundamentalmente de anticipar la respuesta de los clientes potenciales y la competencia ante el producto o servicio que ofrecemos. En la sociedad actual, más competitiva, digital y cambiante que nunca, el estudio de mercado tiene mucha importancia ya que nos ayuda a saber en todo momento como están evolucionando nuestros clientes y competidores.



### SISTEMAS EXISTENTES EN EL MERCADO

Actualmente existen multitud de puntos de carga en la red de carreteras españolas, pero se encuentran muy dispersos. Como sabemos todo apunta a un futuro sin coches de combustión. Por lo que los puntos de carga existentes en el mercado serán insuficientes ante la avalancha de vehículos que necesiten cargar su batería.

Según un estudio de Seguros.es Los españoles conducen una media de 12.000 kilómetros y 300 horas anuales. Lo que significa que conducen una media de cerca de 300 horas al año, lo que representaría una conducción sin descanso día y noche durante doce días y medio.

## CHARGING POINTS

Cuando todos los vehículos existentes sean eléctricos, habrá una gran demanda de puntos de carga. Por lo que una plataforma organizada y funcional de usuarios cliente-empresa conseguirá una gran cuota de mercado en este sistema.

Hoy en día, existen infinidad de puntos de carga incluso gratuitos para los vehículos, pero la tendencia del mercado indica que esto va a ir cambiando con el paso de los años, debido a que será imposible atender toda la demanda y menos aún gratuita del parque de automóviles eléctricos.

### PALABRAS CLAVES

En la plataforma Chargin Points, es necesario conocer los tipos de puntos de carga que pueden existir en el mercado actual para elegir adecuadamente la estación según el conector que utilice el vehículo:

#### **MODO DE RECARGA 1: SCHUKO**

El modo de recarga 1 es aquel modo de carga de un vehículo eléctrico en una toma no destinada exclusivamente a la carga del mismo. Dicho de otra forma, es aquel que se realiza en un enchufe clásico doméstico (una toma SCHUKO) como el que usamos para electrodomésticos como la lavadora.

Este sistema es el más sencillo de los modos de recarga, pues consiste únicamente en llevar un cable de nuestra clavija tipo “Schuko” 230 V a nuestro vehículo. Normalmente se cargan así pequeñas motos o bicis eléctricas u otro tipo de vehículo pequeño (patinetes, hoverboards, etc.). La toma no incluye seguridad y no se aconseja para vehículos o motos de mayor potencia.



#### **MODO DE RECARGA 2: CARGA LENTA**

La carga lenta está pensada para la recarga doméstica, en nuestro garaje. Este tipo de recarga es monofásica se realiza con un voltaje de 230V a un máximo de 3,7kW. El vehículo eléctrico o híbrido se conecta a la red eléctrica mediante su conector / adaptador correspondiente para dotar de seguridad a la recarga. El modo de carga 2 consiste en la instalación por parte del usuario de una *caja* con un enchufe tipo Schuko normalmente de

## CHARGING POINTS

forma preferente para la carga del vehículo eléctrico en cuestión (su uso no es exclusivo, pero suele serlo). Esta caja deberá estar provista de los sistemas de protección adecuados. Este modo de recarga nos interesa si tenemos algunos tipos de coche híbrido enchufable o, sobre todo, un vehículo eléctrico *pequeño* como son los cuadriciclos (por ejemplo, EV Tazzari Zero EM2 Space). Además del sistema de recarga, existen otras muchas diferencias como el motor de propulsión o autonomía que debemos tener en cuenta al elegir un coche híbrido o eléctrico. Un ejemplo de un punto de recarga apto para el modo 2 es LugKey.



### MODO 3: CARGA SEMI-RÁPIDA

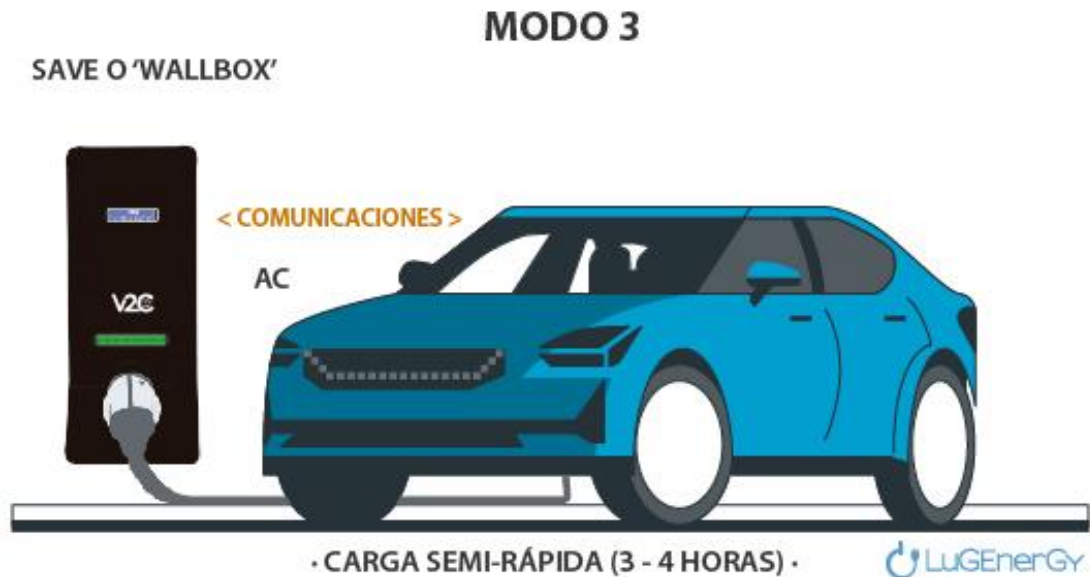
Este modo de carga requiere de un dispositivo llamado Wallbox: un punto de recarga destinado exclusivamente a recargar vehículos eléctricos. Este punto de recarga o wallbox incorpora varios sistemas de protección necesarios para la seguridad de la instalación eléctrica y del vehículo. Según la legislación española (ITC-BT 52), este Modo de recarga es obligatorio para los puntos de recarga de uso público (usualmente es complemento de Modo 4).

La conexión del vehículo eléctrico a la red de corriente alterna se realiza mediante equipamiento dedicado en exclusiva a la recarga del vehículo eléctrico. Este modo 3 nos interesa especialmente para uso doméstico si tenemos un híbrido enchufable con una autonomía considerablemente alta o un coche 100% eléctrico. El modo 3 de recarga



permite una recarga monofásica o trifásica. Es decir, si nuestro vehículo lo permite, podemos recargarlo hasta 32A (a más de 7.2kWh y 400V).

Para este modo de carga necesitaremos un conector específico, que será Tipo 1 (SAE J1772) o Tipo 2 (IEC 62196-2) dependiendo de las características de nuestro vehículo. Progresivamente, este modo 3 se está generalizando dejando atrás el modo 2. Esto es debido al avance tecnológico en cuanto a potencia y capacidad de batería de los EV coches eléctricos: el modo 3 es más seguro, fiable y mucho más rápido.



### MODO 4: LA MÁS RÁPIDA

La recarga en Modo 4 aquella que se realiza en una estación fuera de nuestra vivienda y nos permite recargar como mínimo un 70% de la batería en menos de 30 minutos. El conector estándar más utilizado para este tipo de recarga es el japonés CHAdeMO. Es decir que por el lado del vehículo tendremos un conector de Tipo 1 (SAE J1772 o Yazaki) o de Tipo 2 (IEC, Mennekes) y, a su vez, un conector CHAdeMo o CCS según nuestro EV. Por el lado del punto de recarga la extensión del cable deberá ser CHAdeMO.

El modo 4 es considerado como tal a partir de 50 kWh. Debemos distinguir entre el modo 4 con recarga 'super rápida' y el modo 4 con recarga 'ultra rápida'. Esta última no es aconsejable para carga diaria puesto que puede dañar la batería si hacemos un uso habitual. Está especialmente pensado para las estaciones de uso público exteriores como las electrolineras, donde podemos recargar el vehículo durante trayectos largos o situaciones concretas en las que dispongamos de poco tiempo.



## CHARGING POINTS

La recarga en modo 4 se realiza en corriente continua, a diferencia de los modos anteriores, que se realizan en corriente alterna. Los puntos de recarga que usan una carga de tipo 4, en cualquier caso, no están diseñados ni son recomendables para garajes particulares. Esto es debido principalmente a su elevado coste tanto de producto como de instalación. Además, realmente son el modo 2 o el modo 3 los más aconsejables para particulares por uso y condiciones.



## **CAPITULO 2 –FORMACIÓN APLICADA AL PROYECTO**

### **MÓDULOS FORMATIVOS APLICADOS EN EL TFG**

#### **MP-0484 BASES DE DATOS**

Se crean los casos de uso de la aplicación, el diagrama de clases y la base de datos basándose en los conocimientos adquiridos en esta asignatura.

#### **MP-0485 PROGRAMACIÓN**

La parte del BackEnd de la aplicación va a estar programada en Java en su totalidad, poniendo de manifiesto los conocimientos adquiridos en esta asignatura.

#### **MP-0487 ENTORNOS DE DESARROLLO**

Se ampliarán los casos de uso y diagrama de clases con los fundamentos adquiridos en esta asignatura. Además, se utilizará Git como control de versiones.

#### **MP-0373 LENGUAJES DE MARCAS Y SISTEMAS DE GESTIÓN DE INFORMACIÓN**

La parte de las vistas se realiza en HTML5, estos conocimientos han sido adquiridos en esta asignatura.

#### **MP-0612 DESARROLLO WEB EN ENTORNO CLIENTE**

Cuando se dota de las vistas de dinamismo, se hace con el lenguaje JavaScript, el cual fue el lenguaje desarrollado en esta asignatura.

#### **MP-0613 DESARROLLO WEB EN ENTORNO SERVIDOR**

La mayor parte de la aplicación se encuentra desarrollada con los fundamentos adquiridos en esta asignatura. El modelo vista-controlador es clave en el desarrollo del proyecto.

#### **MP-0615 DISEÑO DE INTERFACES WEB**

Ampliando los conocimientos de Lenguaje de Marcas, con esta asignatura se consigue una interfaz de la aplicación más atractiva en la parte del cliente.

#### **MP-0614 DESPLIEGUE DE APLICACIONES WEB**

Sin duda, tiene una importancia significativa para el proyecto. Tanto para el despliegue en local de nuestra aplicación y probar su funcionalidad como para el posible despliegue

de la aplicación en un servidor externo, son necesarios los conocimientos de esta asignatura.

## MODELO DE DATOS UTILIZADO

### MODELO VISTA CONTROLADOR (MVC)

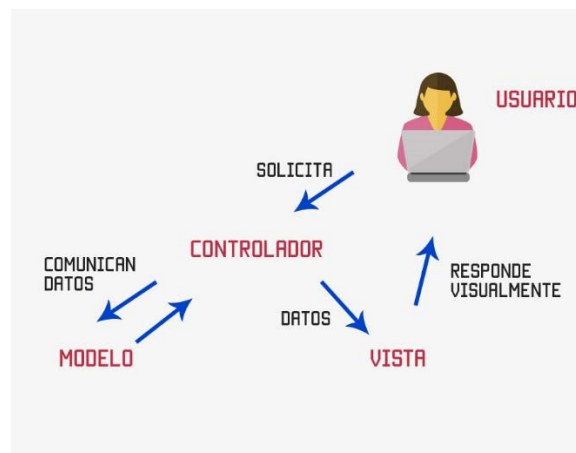
Modelo Vista Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Se trata de un modelo muy maduro y que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo.

El Modelo que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.

La Vista, o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste.

El Controlador, que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.



#### **El modelo es el responsable de:**

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- Define las reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor".
- Lleva un registro de las vistas y controladores del sistema.

- Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo (por ejemplo, un fichero por lotes que actualiza los datos, un temporizador que desencadena una inserción, etc.).

### **El controlador es responsable de:**

- Recibe los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- Contiene reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada al método "Actualizar()". Una petición al modelo puede ser "Obtener\_tiempo\_de\_entrega ( nueva\_orden\_de\_venta )".

### **Las vistas son responsables de:**

- Recibir datos del modelo y los muestra al usuario.
- Tienen un registro de su controlador asociado (normalmente porque además lo instancia).
- Pueden dar el servicio de "Actualización()", para que sea invocado por el controlador o por el modelo (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).

### TECNOLOGIAS QUE SE USARAN EN LA APLICACIÓN DEL PROYECTO

#### JAVA EE



Java Platform, Enterprise Edition (Java EE) se basa en la especificación Java SE. Representa una colaboración entre varios proveedores y líderes del sector y proporciona el soporte de la infraestructura para las aplicaciones.

En la infraestructura de Java EE, añade las reglas en dos niveles:

- En la capa de la aplicación, para gestionar la lógica empresarial dinámica y el flujo de tareas.

- En la capa de presentación, para personalizar el flujo de páginas y el flujo de trabajo y para construir páginas personalizadas basándose en el estado de la sesión.

Java EE es portable y escalable, y soporta la integración existente y los componentes basados en la arquitectura EJB. Java EE simplifica las aplicaciones empresariales definiendo y especificando un complejo conjunto de servicios estándar comunes, como denominación, gestión de transacciones, simultaneidad, seguridad y acceso a base de datos.

Java EE también define un modelo de contenedor, que aloja y gestiona instancias de componentes de aplicaciones Java EE. Los contenedores están a su vez alojados en servidores Java EE.

### JSP



JSP es un acrónimo de Java Server Pages, que en castellano vendría a decir algo como Páginas de Servidor Java. Es, pues, una tecnología orientada a crear páginas web con programación en Java.

Con JSP podemos crear aplicaciones web que se ejecuten en variados servidores web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java. Por tanto, las JSP podremos escribirlas con nuestro editor HTML/XML habitual.

### HTML5



HTML5 (HyperText Markup Language) es la quinta revisión del lenguaje de marcado estándar que se emplea para la web. Es uno de los lenguajes de marcado más usados en todo el mundo y la razón es bastante obvia: gracias a HTML5 podemos crear la estructura de una página web. Texto, imágenes y material multimedia pueden mostrarse correctamente gracias a HTML5.

Aparte de HTML5, existen otros lenguajes que son necesarios para dar formato e interactividad a un sitio, pero la estructura básica de toda página se define primero en lenguaje HTML5.

### CSS



Abreviado en sus siglas en inglés, Cascading Style Sheets, que significa hojas de estilo en cascada, el CSS es una función que se agrega a HTML que proporciona tanto a los desarrolladores de sitios Web, así como a los usuarios, más control sobre cómo se muestran las páginas. Con CSS, los diseñadores y los usuarios pueden crear hojas de estilo que definen cómo aparecen los diferentes elementos, como los encabezados y los enlaces. Estas hojas de estilo se pueden aplicar a cualquier página Web, y nos permite optimizarla para mejorar su posicionamiento.

### JAVASCRIPT



JavaScript es un lenguaje de programación diseñado en un principio para añadir interactividad a las páginas webs y crear aplicaciones web. A pesar de la similitud en el nombre, no está relacionado con Java. Se emplea en el desarrollo de páginas web para tareas como cambiar automáticamente la fecha de una página, hacer que una página aparezca en una ventana emergente al hacer clic en un enlace o que un texto o imagen cambien al pasar el ratón por encima. También suele emplearse para hacer encuestas y formularios. Se ejecuta en el ordenador del visitante a la web, por lo que no requiere descargas constantes desde el sitio web.



### MYSQL



MySQL es un sistema de gestión de bases de datos relacionales (RDBMS) de código abierto respaldado por Oracle y basado en el lenguaje de consulta estructurado (SQL). MySQL funciona prácticamente en todas las plataformas, incluyendo Linux, UNIX y Windows. Aunque puede utilizarse en una amplia gama de aplicaciones, MySQL se asocia más a menudo con las aplicaciones web y la publicación en línea.

### GIT



Git es un sistema de control de versiones distribuido. Esto significa que un clon local del proyecto es un repositorio de control de versiones completo. Estos repositorios locales plenamente funcionales permiten trabajar sin conexión o de forma remota fácilmente. Los desarrolladores confirman su trabajo localmente y, a continuación, sincronizan su copia del repositorio con la copia en el servidor. Este paradigma es distinto del control de versiones centralizado, donde los clientes deben sincronizar el código con un servidor antes de crear nuevas versiones.

## JAVADOC



Javadoc es una utilidad de Oracle para la generación de documentación de APIs en formato HTML a partir de código fuente Java. Javadoc es el estándar de la industria para documentar clases de Java. La mayoría de los IDEs los generan automáticamente.

Para generar API con Javadoc han de usarse etiquetas (tags) de HTML o ciertas palabras reservadas precedidas por el carácter "@". Estas etiquetas se escriben al principio de cada clase, miembro o método, dependiendo de qué objeto se desee describir, mediante un comentario iniciado con "/\*" y acabado con "\*/".

Javadoc es muy útil para que el usuario pueda comprender la utilidad de cada clase y cada método que se trabaja en el proyecto.

Ejemplo código Javadoc en el proyecto:

```

70@ @Autowired
71 private PasswordEncoder pwenco;
72
73@ /**
74  * Este metodo con su anotacion GetMapping actúa como un acceso directo para @RequestMapping (method
75  * = RequestMethod.GET). Nos permite acceder a la pagina de cliente
76  *
77  * @param model Es usado para definir un modelo en una aplicacion Spring MVC
78  * @return jsp de la pantalla index con los elementos para cliente
79  */
80@ @GetMapping("/")
81
82 public String index(Model model) {
83     // Recuperamos al usuario de la sesion
84     Usuario usuario = (Usuario) sesion.getAttribute("usuario");
85     // Recuperamos la lista que nos va a dar el numero que tiene el Ver Carrito a su lado
86     List<String> numeroCarrito = (List<String>) sesion.getAttribute("numeroCarrito");
87     model.addAttribute("numeroCarrito", numeroCarrito.size());
88     model.addAttribute("listaReservasPendientes", resDao.findReservasPorUsuarioAndEstadoPendiente(usuario.getUsername()));
89     model.addAttribute("listaReservasPorCliente", resDao.findReservaPorUsuario(usuario.getUsername()));
90     return "principal/index";
91 }
92

```

Este sería el resultado de nuestra documentación en Javadoc al consultarla:

CHARGING POINTS

OVERVIEW

PACKAGE

CLASS

USE

TREE

DEPRECATED

INDEX

HELP

ALL CLASSES

Packages

Package

fp.charging.points

fp.charging.points.controller

fp.charging.points.modelo.beans

fp.charging.points.modelo.dao

fp.charging.points.repository

fp.charging.points.security

OVERVIEW

PACKAGE

CLASS

USE

TREE

DEPRECATED

INDEX

HELP

OVERVIEW

PACKAGE

CLASS

USE

TREE

DEPRECATED

INDEX

HELP

ALL CLASSES

SEARCH

Package fp.charging.points.controller

Class Summary

Class	Description
AdministradorController	Esta clase define el Controlador del Administrador y sus funcionalidades
ClienteController	Esta clase define el Controlador del Cliente y sus funcionalidades
EmpresaController	Esta clase define el Controlador del usuario Empresa y sus funcionalidades
HomeController	Esta clase define el HomeController, por el pasa siempre el usuario al entrar en la pantalla principal, ser�� la encargada de redirigir a la pantalla correcta de usuario dependiendo de la autentificaci��n que proviene de WebSecurityData.

OVERVIEW

PACKAGE

CLASS

USE

TREE

DEPRECATED

INDEX

HELP

findUsuariosByNombre

```
public java.util.List<Usuario> findUsuariosByNombre(java.lang.String nombre)
```

Metodo para buscar usuarios por nombre

Specified by:  
findUsuariosByNombre in interface IntUsuarioDao

Parameters:  
nombre - Nombre del usuario a buscar

Returns:  
Lista de Usuarios

## FRAMEWORKS

### SPRING Y SPRINGBOOT



Spring Boot es una tecnología que nos permite crear aplicaciones auto contenidas, con esto nos podemos olvidar de la arquitectura y enfocarnos únicamente en desarrollo, delegando a Spring Boot labores como configuración de dependencias, desplegar nuestro servicio o aplicación a un servidor de aplicaciones y enfocarnos únicamente en crear nuestro código.

Para esto Spring Boot utiliza internamente un servidor de aplicaciones embebido, por defecto utiliza Tomcat, pero también lo podemos hacer con jetty o undertow.

Y no solo esto, Spring Boot también nos provee un completo gestor de dependencias como maven o gradle, configuraciones automáticas y mucho más para que nuestra aplicación quede a la medida.

Spring Boot facilita la creación de aplicaciones independientes basadas en Spring de grado de producción que puede "simplemente ejecutar".

### BOOTSTRAP



Bootstrap es un framework CSS de código abierto que favorece el desarrollo web de un modo más sencillo y rápido. Incluye plantillas de diseño basadas en HTML y CSS con la que es posible modificar tipografías, formularios, botones, tablas, navegaciones, menús

desplegables, etc. También existe la posibilidad de utilizar extensiones de Javascript adicionales.

### SWEET ALERT



SweetAlert es un plugin de jQuery y con el cual podremos dar un aspecto profesional a los mensajes que lancemos a los usuarios acorde a las tendencias actuales. Además, tenemos la posibilidad de configurar el plugin de muchas formas diferentes.

### HERRAMIENTAS UTILIZADAS

#### ECLIPSE



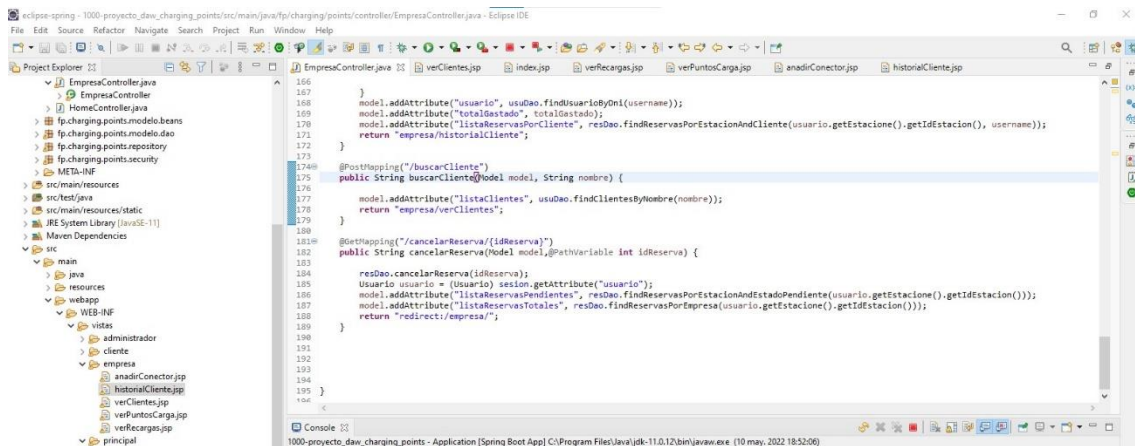
Sin duda, el IDE básico en el proyecto.

Eclipse es un entorno de desarrollo software multi-lenguaje construido alrededor de un workspace al que pueden incluirse un gran número de plug-ins que proporcionan funcionalidades concretas relacionadas con lenguajes específicos o con la interacción con otras herramientas implicadas en el desarrollo de una aplicación. Pese a ser un entorno multi-lenguaje, está desarrollado en Java, siendo el desarrollo en este lenguaje su aplicación principal.

Entre las principales características de Eclipse se encuentran:

- Herramienta de código abierto.
- Salvo el núcleo de la aplicación, todas las funcionalidades de Eclipse están desarrolladas como plug-ins.
- La funcionalidad del paquete básico puede ampliarse mediante la descarga de otros plug-ins, y además es posible desarrollar plug-ins personalizados.
- Proporciona una plataforma RCP (Rich Client Platform) para el desarrollo de aplicaciones de propósito general.
- Permite el desarrollo de aplicaciones para el servidor y servicios web.
- Entre sus plug-ins se encuentran funcionalidades para la interacción con repositorios de código compartido.

## CHARGING POINTS



## MYSQL WORKBENCH



Se trata de una herramienta que te permite diseñar de forma visual las bases de datos, facilitándote la tarea de trabajar con tablas y vistas.

Algunas de las características más interesantes de MySQL Workbench son:

- Edición de diagramas basada en Cairo, con posibilidad de realizar una salida en los formatos como OpenGL, Win32, X11, Quartz, PostScript, PDF...
- Proporciona una representación visual de las tablas, vistas, procedimientos y funciones almacenadas y claves foráneas.
- Permite acceso a bases de datos e ingeniería inversa de las mismas para crear los SQL de creación
- Ofrece sincronización con la base de datos y el modelo.
- Permite generar los scripts SQL a partir del modelo creado.
- Ofrece una arquitectura extensible.
- Tiene soporte para exportar los datos como script SQL CREATE.
- Permite importar modelos de DBDesigner4.
- Ofrece soporte completo a las características de MySQL 5.

## CHARGING POINTS

The screenshot shows the MySQL Workbench interface. On the left, the 'Schemas' pane shows the 'charging\_points' database with various tables. The 'Table: usuarios' structure is listed with columns: username, password, enabled, nombre, apellidos, direccion, email, matricula, fecha\_registro, fecha\_ultima\_carga, numero\_cargas, id\_estacion, provincia, and cpostal. The main pane shows a SQL query: `SELECT * FROM charging_points.usuarios;` and the resulting data grid with 5 rows of user information.

username	password	enabled	nombre	apellidos	direccion	email	matricula	fecha_registro	fecha_ult
12345678a	\$2a\$10\$eyew60Pyhzc...	1	Jose Maria	Tenreiro Eiranova	C/Corro	jose@gmail.com	234490D	2019-06-08	2021-02-1
34567891B	\$2a\$10\$G0Mx12LbG9/KS...	1	Antonio	Martinez	C/Ordenador	antonio@gmail.com	234490D	2020-12-30	2021-02-1
53162796A	\$2a\$10\$Q6KDYfIB0x...	1	Manuel	Hernandez Ginez	C/Mercadona	cliente@cliente.com	635150F	2022-05-08	2022-05-08
53162796B	\$2a\$10\$Q6KDYfIB0x...	1	Restaurante La Merced	Alberto	C/Arco del Mar	empresa2@empresa.com	2022-05-08	2022-05-08	2022-05-08
53162796C	\$2a\$10\$Q6KDYfIB0x...	1	Zapateria Maruja	Maria	Vazquez Alvarez	empresa@empresa.com	2022-05-08	2022-05-08	2022-05-08

## GIT BASH



Git Bash es una aplicación para entornos de Microsoft Windows que ofrece una capa de emulación para una experiencia de líneas de comandos de Git. Bash es el acrónimo en inglés de Bourne Again Shell. Una shell es una aplicación de terminal que se utiliza como interfaz con un sistema operativo mediante comandos escritos. Bash es una shell predeterminada popular en Linux y macOS. Git Bash es un paquete que instala Bash, algunas utilidades comunes de bash y Git en un sistema operativo Windows.

```
MINGW64/c:/Users/JMTE/eclipse-spring/1000-proyecto_daw_charging_points
src/main/webapp/WEB-INF/vistas/empresa/verPuntosCarga.jsp
src/main/webapp/WEB-INF/vistas/empresa/verRecargas.jsp
g
JMTE@DESKTOP-693IBKV MINGW64 ~/eclipse-spring/1000-proyecto_daw_charging_points
(master)
$ git add .

JMTE@DESKTOP-693IBKV MINGW64 ~/eclipse-spring/1000-proyecto_daw_charging_points
(master)
$ git commit -m "Se aplican cambios en el back y en el diseño de las vistas"
[master 9b7916c] Se aplican cambios en el back y en el diseño de las vistas
14 files changed, 30 insertions(+), 17 deletions(-)

JMTE@DESKTOP-693IBKV MINGW64 ~/eclipse-spring/1000-proyecto_daw_charging_points
(master)
$ git push origin master
Enumerating objects: 56, done.
Counting objects: 100% (56/56), done.
Delta compression using up to 4 threads
Compressing objects: 100% (25/25), done.
Writing objects: 100% (29/29), 3.33 KiB | 81.00 KiB/s, done.
Total 29 (delta 17), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (17/17), completed with 15 local objects.
```



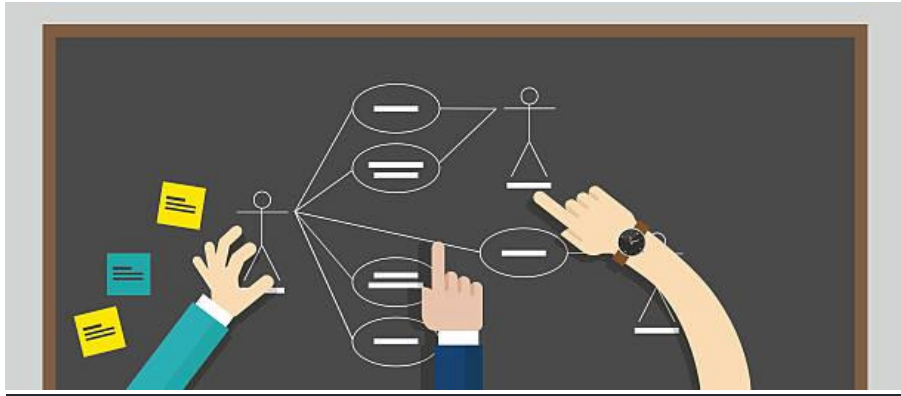
### GITHUB



Github es un portal creado para alojar el código de las aplicaciones de cualquier desarrollador, y que fue comprada por Microsoft en junio del 2018. La plataforma está creada para que los desarrolladores suban el código de sus aplicaciones y herramientas, y que como usuario no sólo puedas descargar la aplicación, sino también entrar a su perfil para leer sobre ella o colaborar con su desarrollo.

## **CAPITULO 3 – ESTRUCTURA DEL PROYECTO**

### **CASOS DE USO**



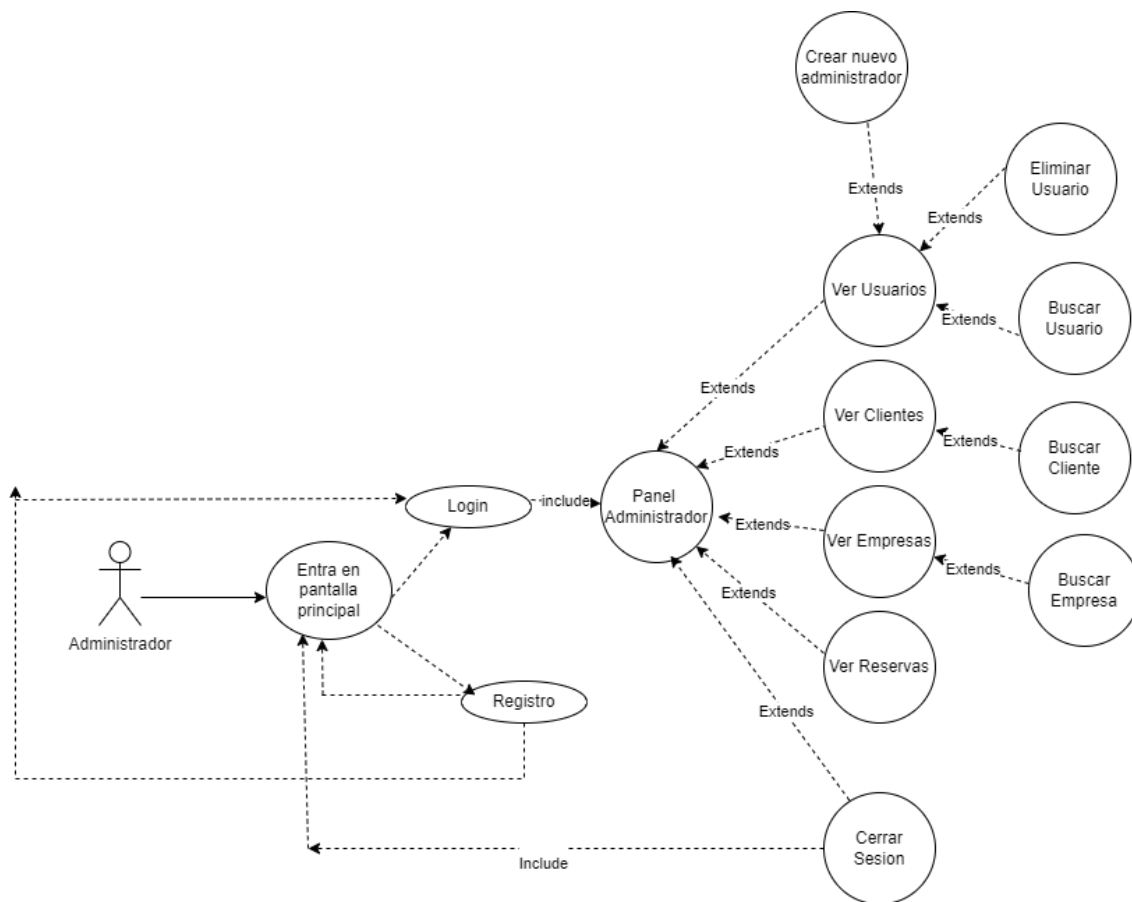
En el diagrama de casos de uso, las funciones del sistema en cuestión se representan desde el punto de vista del usuario (llamado “actor” en UML). Este actor no tiene que ser necesariamente un usuario humano, sino que el rol también puede atribuirse a un sistema externo que accede a otro sistema. De este modo, el diagrama de casos de uso muestra la relación entre un actor y sus requisitos o expectativas del sistema, sin representar las acciones que tienen lugar o ponerlas en un orden lógico.

En la práctica, esta estructura es adecuada para representar claramente las funciones y/o objetivos más importantes de un sistema. Por esta razón, a la hora de desarrollar un software o planificar nuevos procesos empresariales, crear un diagrama de casos de uso suele ser uno de los primeros pasos, ya que permite visualizar clara y fácilmente qué casos de uso deben tenerse en cuenta durante el desarrollo para que los actores (y, en un sentido más amplio, también los operadores o clientes) logren su objetivo, en principio independientemente de la viabilidad técnica.

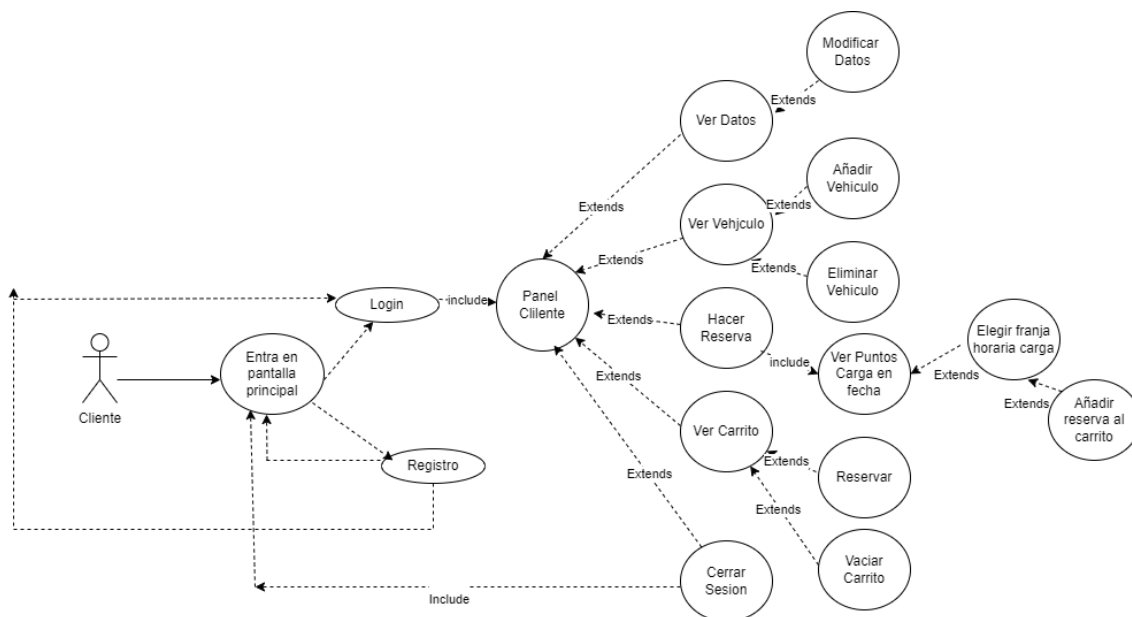
Como nuestros usuarios son de 3 tipos, representamos 3 diagramas de caso de uso.

NOTA: Debido la complejidad de la aplicación, algunos casos de uso se han ignorado, centrando la atención en los más importantes.

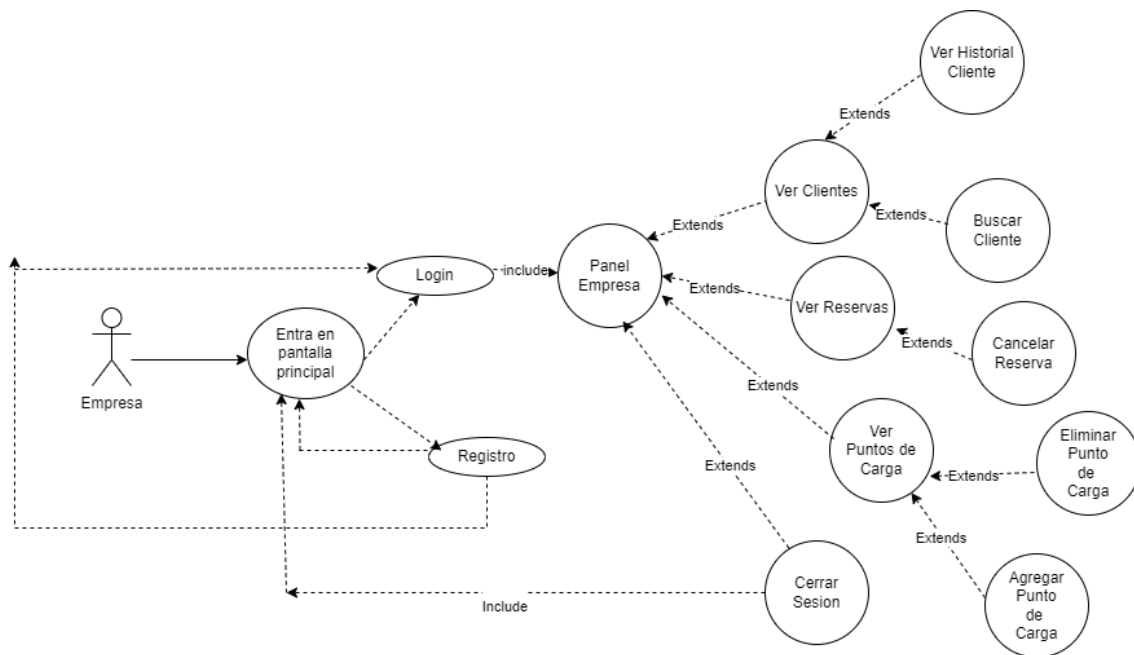
## CASOS DE USO PARA USUARIO ADMINISTRADOR



## CASOS DE USO PARA USUARIO CLIENTE



## CASOS DE USO PARA USUARIO EMPRESA



**WIREFRAME****MI PROPÓSITO CON EL WIREFRAME**

El wireframe reemplaza la naturaleza abstracta del mapa de sitio, que por lo general es el primer paso en el desarrollo de sitios web, con algo más tangible y fácil de entender. El wireframe tienen muchos propósitos interrelacionados:

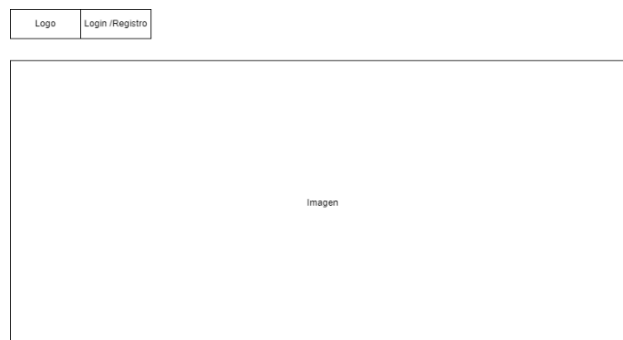
- Garantizar que el sitio o la aplicación se desarrolle de conformidad con los fines acordados. Ver las funciones claramente con mínima influencia creativa permite a los interesados centrarse en otros aspectos del proyecto. La creación de wireframes establece expectativas sobre cómo se implementarán las funciones, mostrando cómo funcionarán, dónde estarán ubicadas y cuántos beneficios ofrecerán. Se puede eliminar una función porque no se adecúa a los objetivos de tu página web.

- Centrarse en la facilidad de uso. La creación de diagramas ofrece una mirada objetiva de los nombres de enlaces, rutas de conversión, facilidad de uso, navegación, y disposición de las funciones. Los wireframes te ayudan a identificar fallas en la arquitectura del sitio o las funciones y te indican qué tan bien fluye desde la perspectiva del usuario.

- Capacidad de crecimiento del contenido. Si sabes que tu página web crecerá en un futuro próximo, tu sitio debe estar preparado para que ese crecimiento tenga un impacto mínimo en el diseño, la facilidad de uso y la arquitectura del sitio. La creación de wireframes puede revelar estas importantes oportunidades de crecimiento del contenido y cómo adaptarse a ellas.

-Comentarios e iteración sin esfuerzo. En lugar de unir la funcionalidad completa, la disposición y los elementos creativos en un solo paso, los wireframes garantizan que estos factores se aborden de forma separada. Esto permite a los interesados brindar comentarios en etapas más tempranas del proceso. La creación de wireframes mediante software permite que el proceso iterativo inherente al diseño web sea una tarea mucho menos tediosa.

### WIREFRAME PANTALLA INICIAL



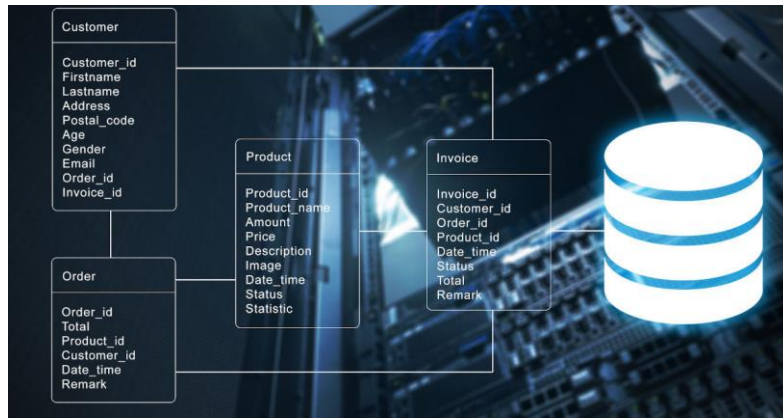
### WIREFRAME LOGIN



### WIREFRAME PANTALLAS USUARIOS



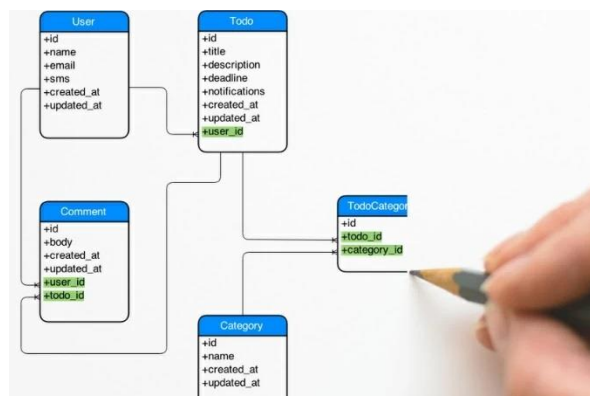
## BASE DE DATOS



Una base de datos se define como un conjunto de datos organizados, los cuales guardan una cierta relación entre sí. Así, los sistemas de información se encargan de recolectarlos y, posteriormente, gestionarlos y administrarlos para unos objetivos concretos. Las bases de datos tienen una serie de características que resulta interesante conocer.

Desde el lado de la informática, la base de datos se conoce como un sistema conformado por un abanico de datos almacenados en distintos discos, físicos o virtuales, que permiten el acceso directo a los mismos y un software capaz de manipular dichos datos.

Cada base de datos se compone de una o varias tablas, las cuales guardan un grupo de datos. Cada tabla se conforma de una o más columnas y filas. De este modo, cada columna almacena una parte de la información acerca de cada elemento que se desee guardar en la tabla, mientras que cada fila de la tabla compone un registro.



Por un lado, todos los datos almacenados guardan una independencia tanto lógica como mínima. Además, todos ellos tienen una redundancia mínima.

Por otro lado, uno o varios usuarios pueden tener acceso a los datos almacenados en la base mediante un acceso seguro.

MySQL es un sistema de gestión de base de datos que actualmente cuenta con más de seis millones de clientes en todo el mundo. Un software libre que se engloba en el grupo de licencias GNU GPL. MySQL es a día de hoy ampliamente utilizado en una gran selección de aplicaciones web tales como WordPress o Drupal, entre otras. Una de sus principales ventajas es que su lectura es muy rápida, por lo que es una opción estupenda para este tipo de apps.

A continuación señalamos las principales características que definen las bases de datos MySQL.

Arquitectura cliente y servidor: MySQL, al igual que cualquier otro sistema de registros de datos, es un programa de registro basado en un sistema entre cliente y servidor. Se trata por tanto de un software en el que se engloban un amplio abanico de clientes y servidores que establecen comunicación entre ellos.

Compatibilidad con SQL: SQL es un lenguaje de programación que permite tanto la consulta como la renovación de datos para la gestión de una base en la que se almacena un conjunto de datos.

Búsqueda de texto completo: la búsqueda de texto completo acelera y facilita en gran medida el sistema de búsqueda de las palabras de los datos contenidos en la base.

Lenguaje de programación: la base de datos MySQL está escrita en C y C++, dos de los lenguajes de programación más demandados y populares de todo el mundo.

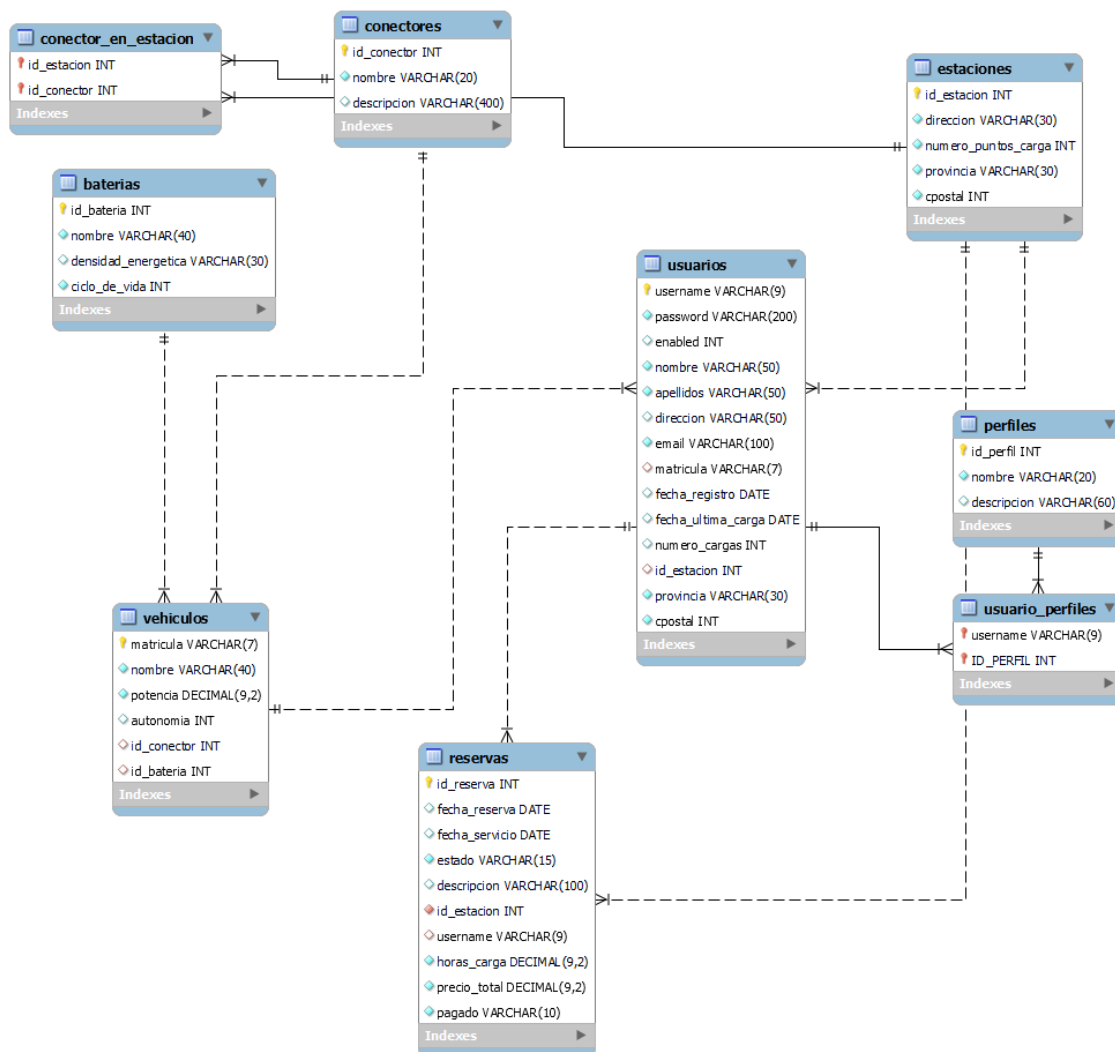
Sistemas de almacenamiento: este tipo de bases proporciona sistemas de almacenamiento tanto transaccionales como no transaccionales.



## CHARGING POINTS

La base de datos de nuestro proyecto está construida en MySQL y la hemos creado con el MySQL WorkBench.

A continuación, se presenta el Schema de nuestra Base de Datos del proyecto:



## CAPITULO 4 – FUNCIONAMIENTO DE LA APLICACIÓN

### PANTALLA PRINCIPAL

El funcionamiento de la plataforma se estructura de la siguiente manera:

El usuario accede al portal y se encuentra con la pantalla de acceso:



En este momento el usuario tiene dos opciones, o loguearse si ya está registrado o acceder al formulario de registro.

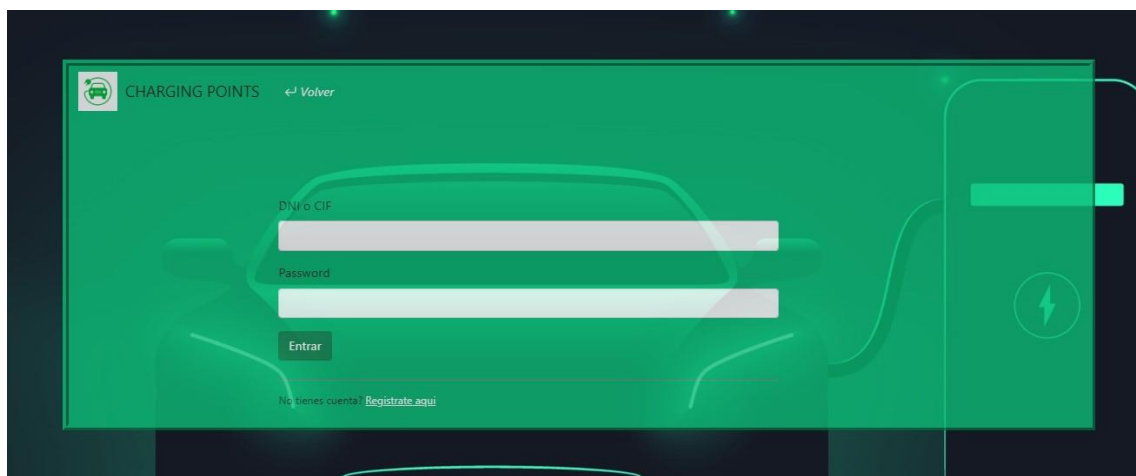
Cuando accede al formulario de registro, debe de cumplimentar todos los datos que se le requieren:

## CHARGING POINTS

En este momento el usuario debe decidir si va a ser un usuario Cliente o un usuario Empresa, los datos a introducir serán los mismos, teniendo en cuenta que la dirección del usuario empresa será la dirección de su estación de carga.

En el caso de que el DNI del usuario ya se encontrara registrado, nos advierte con un mensaje y no nos dejaría registrar. Cuando introduce todos los datos y pulsa en alta usuario le redirige a la pantalla principal.

Cuando el usuario accede a login se encuentra la siguiente pantalla:



Si el usuario introducido o la contraseña no es válida, no podremos acceder a la plataforma.

Dependiendo del tipo de perfil del usuario (Administrador, Cliente o Empresa) la aplicación deriva al usuario a su correspondiente panel de control personalizado y con sus funciones específicas.

## PANEL ADMINISTRADOR

Este perfil será el encargado de llevar el mantenimiento de la plataforma y solucionar posibles incidencias ya que cuenta con unos permisos que los otros usuarios no poseen.

Su panel de usuario cuenta con las siguientes funciones:

DNI/CIF	Nombre	Apellidos	Perfil	Historial
53162796C	Zapateria Maruja	Maria	Empresa	Historial ^v
53162796B	Restaurante La Merced	Alberto	Empresa	Historial ^v
53162796A	Manuel	Hernandez Gimenez	Cliente	Historial ^v

En esta pantalla el administrador puede ver los últimos usuarios registrados y cuenta con una barra-menú en donde puede consultar el total de usuarios, los usuarios que son clientes, los usuarios que son empresa y las reservas totales realizadas.

Si accede a usuarios totales:

Nombre	Apellido	DNI/CIF	Fecha Alta	Email	Perfil	Acciones	Eliminar
→ Jose Maria	Tenreiro Eiranova	12345678a	2019-06-08	jose@gmail.com	Administrador	Historial ^v	Eliminar
→ Antonio	Martinez	34567891B	2020-12-30	antonio@gmail.com	Cliente	Historial ^v	Eliminar
→ Manuel	Hernandez Gimenez	53162796A	2022-05-08	cliente@cliente.com	Cliente	Historial ^v	Eliminar
→ Restaurante La Merced	Alberto	53162796B	2022-05-08	empresa2@empresa.com	Empresa	Historial ^v	Eliminar
→ Zapateria Maruja	Maria	53162796C	2022-05-08	empresa@empresa.com	Empresa	Historial ^v	Eliminar

Añadir Administrador +

El administrador puede ver el historial de clientes y empresas, eliminar un usuario determinado y al mismo tiempo es el único que puede dar de alta un nuevo usuario.

Hay que tener en cuenta que la aplicación no permite borrar un usuario administrador si es el único que existe.

## CHARGING POINTS

### PANEL CLIENTE

El usuario cliente cuenta en su panel con las siguientes funciones:

The screenshot shows the 'CHARGING POINTS' client panel. At the top, there's a navigation bar with a car icon, the text 'CHARGING POINTS', and links for 'Mis datos', 'Vehículo', 'Fecha para Reserva', and 'Buscar'. Below this, a welcome message says 'Bienvenido Manuel' with links for 'Cerrar Sesión' and 'Ver Carrito(0)'. The main content is divided into two sections: 'PROXIMAS RECARGAS' and 'TODAS LAS RECARGAS'. Each section contains a table with columns for 'Direccion', 'Fecha Servicio', 'Conector', 'Precio Total', 'Franja Horaria', and 'Estado'. In the 'PROXIMAS RECARGAS' section, two sessions are listed, both with 'Pendiente' status and 'Cancelar'/'Finalizar' buttons. The 'TODAS LAS RECARGAS' section lists four sessions with various statuses including 'Terminada' and 'Pendiente'.

Direccion	Fecha Servicio	Conector	Precio Total	Franja Horaria	Estado
C/Areosa del Mar	2022-06-02	combinado CSS	4.60	Tarde	Pendiente
Vazquez Alvarez	2022-05-15	CHAdMo	4.60	Tarde	Pendiente

Direccion	Fecha Servicio	Conector	Precio Total	Franja Horaria	Estado
Vazquez Alvarez	2022-05-28	combinado CSS	2.40	Mañana	Terminada
C/Areosa del Mar	2022-06-02	combinado CSS	4.60	Tarde	Pendiente
Vazquez Alvarez	2022-05-15	CHAdMo	4.60	Tarde	Pendiente
C/Areosa del Mar	2022-05-20	combinado CSS	46.80	Mañana	Terminada

En la pantalla principal puede consultar las próximas recargas y las que tiene finalizadas. Al mismo tiempo puede cancelar y finalizar las recargas que tiene pendientes.

En la barra menú si selecciona la opción Mis Datos:

The screenshot shows the 'Datos Usuario' form. It has a title 'Datos Usuario' and a list of input fields: 'Usuario' (53162796A), 'Password' (masked), 'Nombre' (Manuel), 'Apellidos' (Hernandez Gimenez), 'Direccion' (C/Mercadona), 'Email' (cliente@cliente.com), 'Provincia' (Burgos), and 'Código Postal' (12334). At the bottom, there are two buttons: 'Modificar' and 'Aplicar Cambios'.

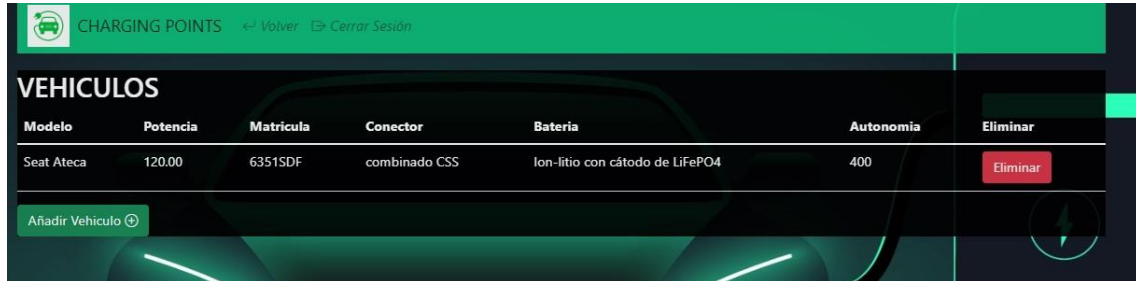
Datos Usuario
Usuario
53162796A
Password
.....
Nombre
Manuel
Apellidos
Hernandez Gimenez
Direccion
C/Mercadona
Email
cliente@cliente.com
Provincia
Burgos
Código Postal
12334

La aplicación cuenta con una protección mediante JavaScript por la cual los datos en un principio solo son visibles, pero no se pueden modificar. Si el usuario quiere modificar

## CHARGING POINTS

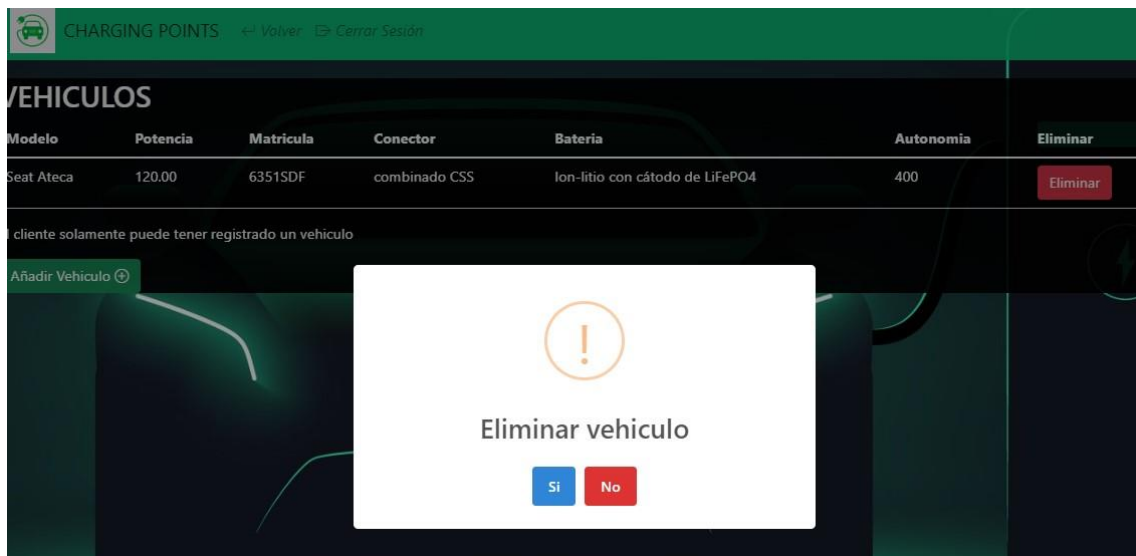
los datos, primero tiene que pulsar en Modificar y posteriormente cambiar los datos que considere (Menos el DNI que no se puede) y aplicar cambios.

Si el usuario cliente en su panel principal pulsa en Vehículo:



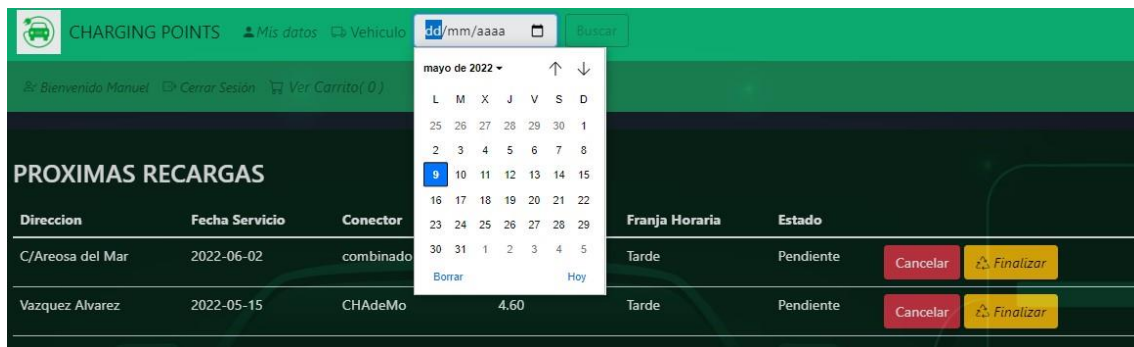
Puede consultar el vehículo que tiene dado de alta. De la misma forma, puede eliminarlo y añadir uno nuevo con un formulario de alta de vehículo. El usuario cliente solamente puede tener un vehículo dado de alta.

Cuando el usuario intenta eliminar el vehículo, un mensaje creado con sweet alert y javascript nos alerta de la acción que se va a cometer:



## CHARGING POINTS

Si el usuario Cliente quiere realizar una reserva. En el panel principal introduce una fecha en el apartado fecha para reserva:



Si introduce una fecha anterior a la actual, no se le permitirá acceder al panel de reserva de punto de carga. Si introduce una fecha valida (posterior a la actual) le mostrará el panel de puntos de carga disponibles para esa fecha:



El usuario debe de seleccionar una estación acorde a su necesidad de carga, en la dirección que considere oportuna y en la franja horaria que necesite. Si la franja horaria en ese punto de carga estuviera ocupada, le avisaría de que no se encuentra disponible.

Cuando selecciona una franja horaria disponible, tiene que rellenar el formulario de aceptación de la reserva para enviarlo al carrito de compra:



## CHARGING POINTS

CHARGING POINTS [Volver](#) [Cerrar Sesión](#)

### INFORMACIÓN NUEVA CARGA

Estacion: 17

Direccion: Vazquez Alvarez

Dia Carga: 20-05-2022

Franja Horaria: Tarde

Conector: combinado CSS

Minutos Carga: 233

Precio: 46.6

[Añadir al carrito](#)

El usuario debe de introducir los minutos que necesita de carga para su vehículo y el precio se calcula automáticamente según el precio asignado por la estación (En estos momentos este precio no está implementado y calcula uno establecido por defecto).

Cuando pulsa añadir al carrito, la reserva se pasa al carrito de la compra. El usuario puede añadir al carrito más reservas y después realizar la reserva de todas ellas juntas desde este.

En este ejemplo, el usuario ha añadido 2 reservas al carrito:

CHARGING POINTS [Mis datos](#) [Vehículo](#) Fecha para Reserva [Buscar](#)

Bienvenido Manuel [Cerrar Sesión](#) [Ver Carrito \(2\)](#)

Si accede a este apartado:

DNI	Fecha Servicio	Conector	Precio Total	Franja Horaria	Estado	
53162796A	Fri May 20 00:00:00 CEST 2022	combinado CSS	46.6	Tarde	Pendiente	<a href="#">Eliminar</a>
53162796A	Fri May 27 00:00:00 CEST 2022	combinado CSS	4.4	Mañana	Pendiente	<a href="#">Eliminar</a>

[Reservar](#)

El usuario puede eliminar del carrito la reserva que crea conveniente o hacer la reserva de las reservas que tiene. Cuando pulsa reservar, esas nuevas reservas se formalizarán.

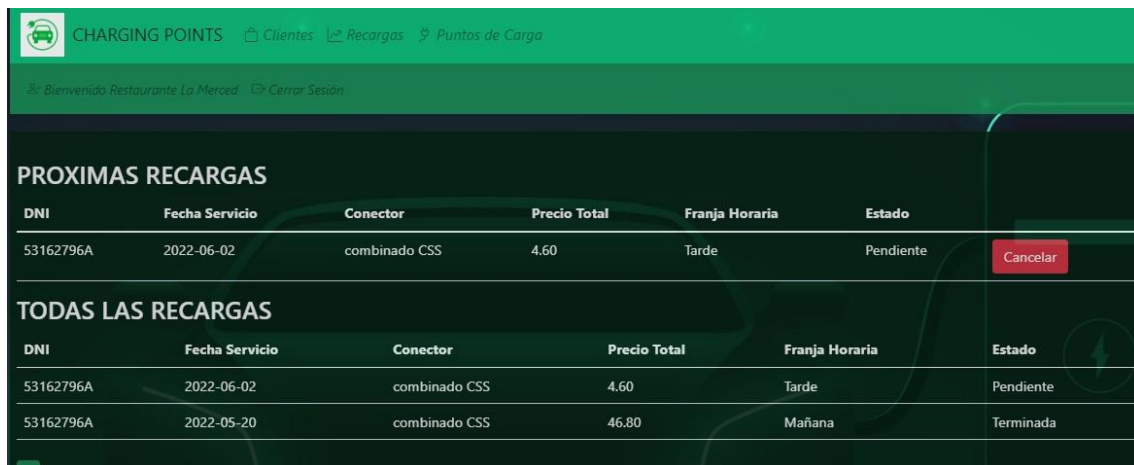


## CHARGING POINTS

### PANEL EMPRESA

Los usuarios Empresa, se registran como empresa y van a facilitar un punto de carga. También facilitan sus datos personales, en los que la dirección de su estación de carga es muy importante porque será la forma que tendrán los clientes de localizarlos para saber si están en su trayectoria de viaje o no.

Cuando el usuario empresa se loguea con sus credenciales, accede a su panel personalizado:



DNI	Fecha Servicio	Conector	Precio Total	Franja Horaria	Estado
53162796A	2022-06-02	combinado CSS	4.60	Tarde	Pendiente

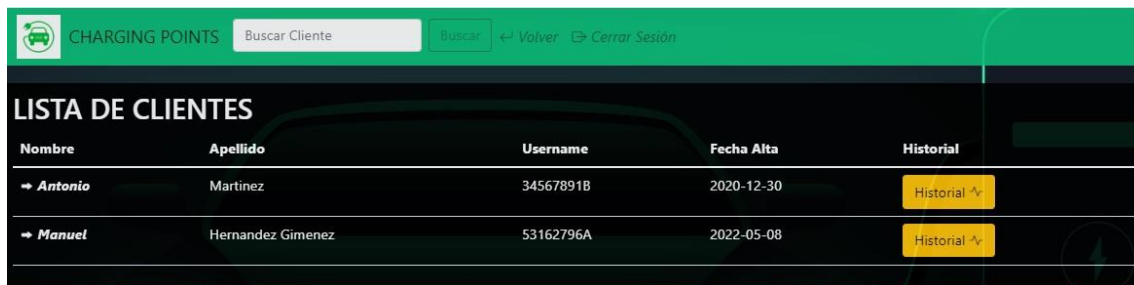
DNI	Fecha Servicio	Conector	Precio Total	Franja Horaria	Estado
53162796A	2022-06-02	combinado CSS	4.60	Tarde	Pendiente
53162796A	2022-05-20	combinado CSS	46.80	Mañana	Terminada

En este panel puede ver todas las recargas que tiene en su estación, tanto las que se encuentran pendientes como las que están finalizadas.

También tiene la opción de cancelar las reservas que crea oportuno desde esta pantalla.

Después en su menú tiene la opción de:

-Ver los clientes:



Nombre	Apellido	Username	Fecha Alta	Historial
→ Antonio	Martinez	34567891B	2020-12-30	Historial ▾
→ Manuel	Hernandez Gimenez	53162796A	2022-05-08	Historial ▾

-Consultar todas las recargas

## CHARGING POINTS

CHARGING POINTS <a href="#">Volver</a> <a href="#">Cerrar Sesión</a>						
LISTA DE RECARGAS						
DNI	Fecha Servicio	Conector	Precio Total	Franja Horaria	Estado	
53162796A	2022-06-02	combinado CSS	4.60	Tarde	Pendiente	<button>Cancelar</button>
53162796A	2022-05-20	combinado CSS	46.80	Mañana	Terminada	

-Añadir puntos de carga a su estación

-Si el usuario selecciona esta última opción:

CHARGING POINTS <a href="#">Volver</a> <a href="#">Cerrar Sesión</a>		
LISTA DE CONECTORES DISPONIBLES EN LA ESTACIÓN		
Nombre	Descripción	Eliminar
→ CHAdeMo	En la mayoría de modelos donde va montado también se acepta el Tipo 1 y son algunos como los Nissan Leaf y e-NV200, los Mitsubishi Outlander e iMiev o el Kia Soul eléctrico, entre otros.	<button>Eliminar</button>
→ combinado CSS	La clave del CSS es su potencial. En corriente alterna ya podría llegar hasta los 63 amperios y 43 kW como el Mennekes, mientras que en corriente continua podría alcanzar una recarga rápida de 100 kW	<button>Eliminar</button>
<button>Añadir Conector</button>		

Observa todos los conectores que tiene implementados en su estación. Puede eliminar el que considere oportuno al mismo tiempo, puede escoger la opción de añadir uno nuevo:

CHARGING POINTS <a href="#">Volver</a>	
AÑADIR CONECTOR A LA ESTACIÓN	
Novedad	<div>Mennekes (Tipo 2)</div>
<button>Alta Conector</button>	

De esta forma, su estación contaría con un nuevo conector:

CHARGING POINTS <a href="#">Volver</a> <a href="#">Cerrar Sesión</a>		
LISTA DE CONECTORES DISPONIBLES EN LA ESTACIÓN		
Nombre	Descripción	Eliminar
→ Mennekes (Tipo 2)	Se trata de uno de los enchufes más comunes y está presente en un buen número de eléctricos hechos por fabricantes europeos (principalmente los grandes alemanes). Marcas como BMW, Mercedes o Volkswagen (incluyendo marcas como Audi o Porsche) lo usan, aunque también está presente en otras como Renault, Volvo o Tesla.	<button>Eliminar</button>
→ CHAdeMo	En la mayoría de modelos donde va montado también se acepta el Tipo 1 y son algunos como los Nissan Leaf y e-NV200, los Mitsubishi Outlander e iMiev o el Kia Soul eléctrico, entre otros.	<button>Eliminar</button>
→ combinado CSS	La clave del CSS es su potencial. En corriente alterna ya podría llegar hasta los 63 amperios y 43 kW como el Mennekes, mientras que en corriente continua podría alcanzar una recarga rápida de 100 kW	<button>Eliminar</button>
<button>Añadir Conector</button>		

### **CAPITULO 5 – CONCLUSIONES Y FUTURAS MEJORAS**

**¿Qué he aprendido una vez realizado el proyecto? ¿En qué ha cambiado mi comprensión del tema del que se trata en el proyecto?**

He aprendido a realizar una aplicación desde sus cimientos, empezando de cero, creando los casos de uso, implementando la base de datos, siendo consciente de los errores que se producen y ser capaz de solucionarlos.

Me he dado cuenta de que el tema de carga de vehículos eléctricos es muy extenso. Se necesita un gran conocimiento de la materia para poder dar servicio a todos los usuarios. He conseguido formarme en algunos aspectos, pero el recorrido que falta es muy largo.

**¿Se confirmaron nuestras sospechas respecto al resultado? ¿Son valiosos, reveladores o sugerentes los resultados obtenidos?**

Cuando empiezas a desarrollar una idea, por el camino te encuentras con problemas que piensas que pueden interferir en el resultado final, no hay duda de que hubo que adaptarse durante la realización del proyecto para llevarlo a cabo de forma satisfactoria. Los resultados obtenidos nos reflejan que el estudio del tema a tratar tiene que ser muy profundo y sus conocimientos también, si se trata de un caso tan complicado como este.

**¿Qué cosas haríamos diferente si tuviéramos que repetir el proyecto y por qué?  
¿Qué aspectos del mismo tuvieron debilidades y cómo podrían subsanarse a futuro?**

Si tuviera que repetir el proyecto, intentaría tener muchos más claros los casos de uso al principio antes de realizar la base de datos. Aunque bien es cierto que hasta en las propias empresas estos aspectos se corrigen con el paso del tiempo y las incidencias.

Sin duda alguna, me quedan muchos puntos pendientes para el futuro:

- Implementar el front en angular y mejorar su diseño.
- Implementar más casos de uso
- Implementar que la búsqueda pueda ser por provincia y código postal (La base de datos ya está preparada para ello)
- Implementar geolocalización de las estaciones con un mapa y sus ubicaciones

**Una vez observado el resultado del proyecto, ¿Hacia dónde tendría que marchar una continuación y por qué?**

La continuación pasa por mejorar la interacción usuario aplicación, dotar de más funcionalidades con el tiempo a Charging Points para que el usuario final se decida a utilizarla y de esta manera poder crecer su uso.

### **CAPITULO 6 - BIBLIOGRAFIA Y ANEXOS**

**<https://es.stackoverflow.com>**

Si eres un programador aficionado o profesional, es probable que hayas oído hablar de Stack Overflow. La plataforma goza de gran popularidad para haber sido lanzada en 2008, como una alternativa de mayor calidad a sitios de preguntas y respuestas tradicionales. Por supuesto, fomentando una dinámica que se adapta a las necesidades y que resuelve dudas de las personas interesadas y versadas en programación.

**<https://app.diagrams.net/>**

Espacio gratuito y sin registro para poder realizar nuestros casos de uso UML.

### **GITHUB**

**El repositorio con todo el código de la aplicación se encuentra en la siguiente dirección:**

**<https://github.com/JMTE/ChargingPoints.git>**