



PROYECTO FINAL DAW TESSERACT SOLUTIONS

Iván García Carrasco



IES ARCIPRESTE DE HITA
13 de mayo de 2024

Índice

1. Introducción.....	3
2. Definición del proyecto	3
2.1. Descripción muy general del proyecto.....	3
2.2. Tipos de Empresas. Sectores Productivos	3
3. Diseño y fase del proyecto	4
3.1. Objetivos. Especificación de requisitos	4
3.2. Identificación de las fases del proceso	4
3.2.1. Análisis.....	4
3.2.2. Diseño	5
3.2.3. Codificación	21
3.2.4. Pruebas	22
3.2.5. Despliegue	22
3.2.6. Mantenimiento.....	22
3.3. Herramientas de diseño	22
3.3.1. Herramientas en el lado cliente	22
3.3.2. Herramientas en el lado servidor	22
3.3.3. Herramientas de desarrollo.....	22
3.4. Evaluación económica y financiación del proyecto.....	23
3.4.1. Estudio de viabilidad.....	23
3.4.2. Ayudas económicas	23
4. Planificación del proyecto	24
4.1 Diagrama de secuenciación del proyecto.....	24
4.2. Secuenciación de actividades o tareas	24
4.3 Asignación de recursos y tiempos en las actividades del proyecto	25
4.4. Procedimientos en las actividades.	25
4.5. Prevención de riesgos en el desarrollo del proyecto (Retrasos, bloqueos..etc.)	27
5. Pruebas y validación del Proyecto.....	27
5.1. Evaluación y seguimiento de las actividades.....	27
5.2. Indicadores de calidad. Elaborar una batería de pruebas para detectar errores.	28
5.3. Informe de evaluación de incidencias. Solución de las mismas.....	33
6. Implantación del proyecto.....	34
6.1. Plan de implantación.	34

6.1.1. Implantación en máquina virtual	34
6.1.2. Implantación en servidor propio	35
6.2. Manual de instalación.	36
6.2.1 Cambio de constantes del servidor	36
6.2.2 Cambio de constantes de la aplicación	36
6.2.3 Contraseñas	37
6.3. Manual de usuario.....	37
6.3.1. Introducción	37
6.3.2. Sesión.....	37
6.3.3 Jornada laboral	38
6.3.4. Planificación.....	39
6.3.5. Vehículos	40
6.3.6. Mercancía	42
6.3.7. Trabajadores.....	44
6.3.8. Clientes	46
6.3.9. Configuración.....	47
6.3.10. Registro de inicios de sesión	47
7. Conclusión	48
8. Bibliografía/Webgrafía	49

1. Introducción

En este documento se detallará toda la documentación técnica acerca de la aplicación web realizada como proyecto final del curso Desarrollo de Aplicaciones Web.

Este proyecto consiste en una web para la gestión de empresas logísticas.

2. Definición del proyecto

2.1. Descripción muy general del proyecto.

Este proyecto trata de una aplicación web para la gestión de todo tipo de empresas logísticas.

La logística es un sector muy amplio y es necesario llevar un registro muy riguroso de todo lo que pasa dentro de la empresa, es por ello que este proyecto abarcara todo lo que puede necesitar una empresa de este tipo para funcionar más rápido y sin errores.

Para los clientes es sumamente importante conocer que pasa en cada momento con la mercancía que han puesto en manos de una empresa externa, gracias a este proyecto se les dará a todos y cada uno la confianza necesaria de poder saber el estado de aquello que hayan enviado.

Respecto a la parte no visible para los clientes, este proyecto llevará toda la gestión interna de la empresa, será la parte más potente de la aplicación, se podrá hacer un registro de todas las posibles partes de la empresa.

2.2. Tipos de Empresas. Sectores Productivos

La aplicación va enfocada completamente al sector **logístico**, a cualquier tipo de empresa que se dedique a la **recepción / envíos de mercancía**.

Esta aplicación está desarrollada para agilizar el proceso logístico por el cual pasa una mercancía desde su recepción hasta su posterior entrega.

Muchas empresas aún utilizan sistemas comunes y antiguos además de inseguros para llevar un control de este proceso. Muchas de estas empresas quieren mejorar la calidad de su trabajo, por lo que es una opción ideal el desarrollo de esta aplicación para su posterior venta.

3. Diseño y fase del proyecto

3.1. Objetivos. Especificación de requisitos

Son dos los objetivos de esta aplicación, la gestión interna y la consulta de los clientes.

Gestión interna:

- Control de mercancía
 - Añadir mercancía
 - Eliminar mercancía
 - Editar mercancía
 - Ver mercancía entregada
 - Asignar mercancía a un vehículo
 - Desasignar mercancía a un vehículo
- Control de vehículos
 - Añadir vehículos
 - Eliminar vehículos
 - Editar vehículos
- Control de empleados
 - Añadir empleados
 - Eliminar empleados
 - Editar empleados
 - Crear cuentas web
 - Eliminar cuentas web
 - Registrar jornadas laborales
 - Gestionar planificación
- Control de clientes
 - Añadir clientes
 - Eliminar clientes
 - Editar clientes
 - Asignar cliente a una mercancía

Interfaz principal:

- Consulta de mercancías

3.2. Identificación de las fases del proceso

3.2.1. Análisis

Se evalúan las necesidades de las empresas y usuarios que hagan uso del software, con ello, se obtienen los requisitos de la aplicación, nombrados en el [punto 3.1](#) y se planean varias rutas a seguir para la realización del proyecto, finalmente se escoge una de ellas.

3.2.2. Diseño

- **Tecnologías**

Se escogen las tecnologías que mejor encajan según los requisitos obtenidos.

En el [punto 3.3](#) están definidas las tecnologías y herramientas utilizadas.

- **Prototipos de las interfaces**

Primero se realizaron todos sketch.

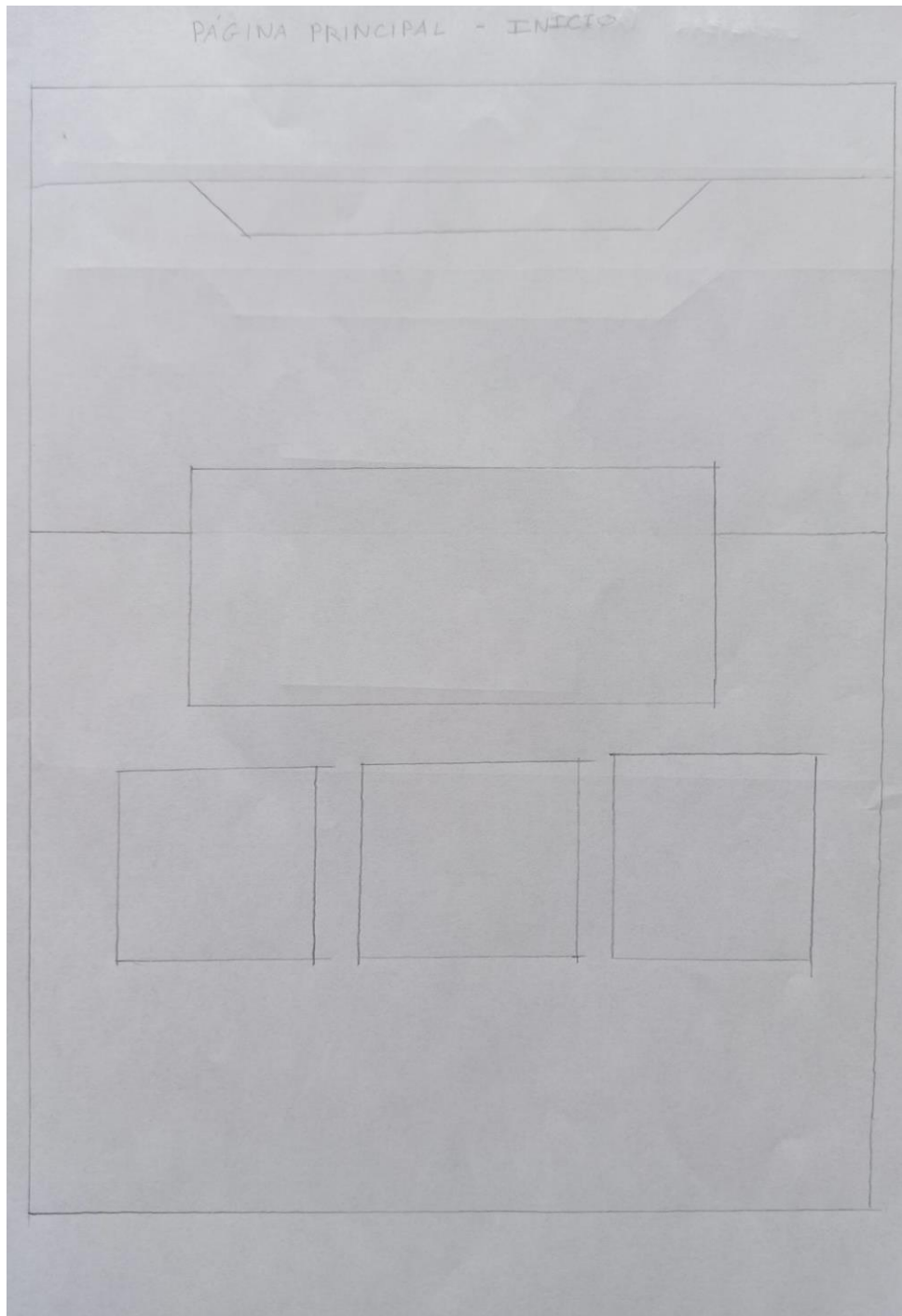


Imagen 1. Diseño principal

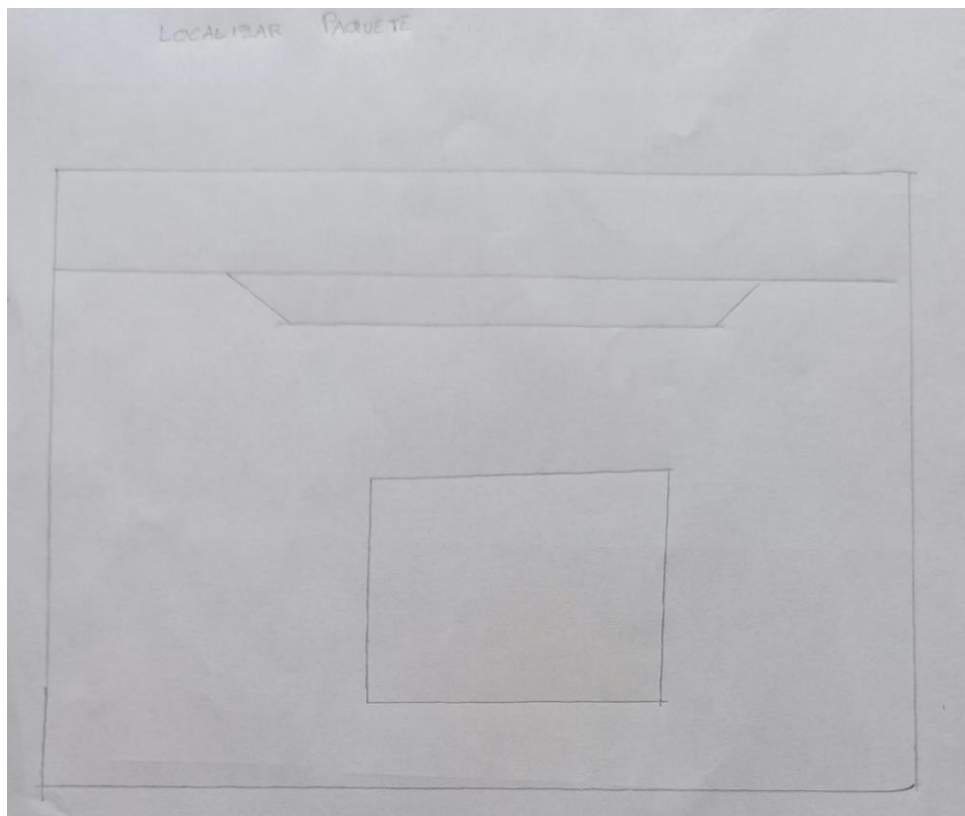


Imagen 2. Diseño localizar paquete

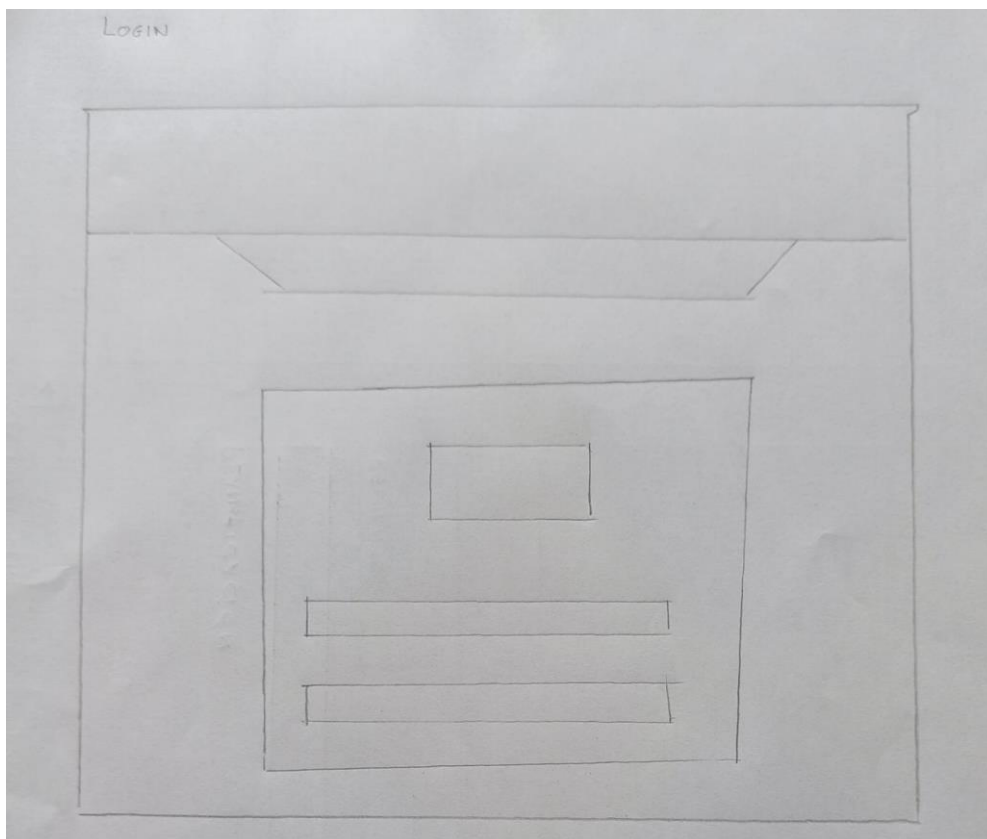


Imagen 3. Diseño iniciar sesión

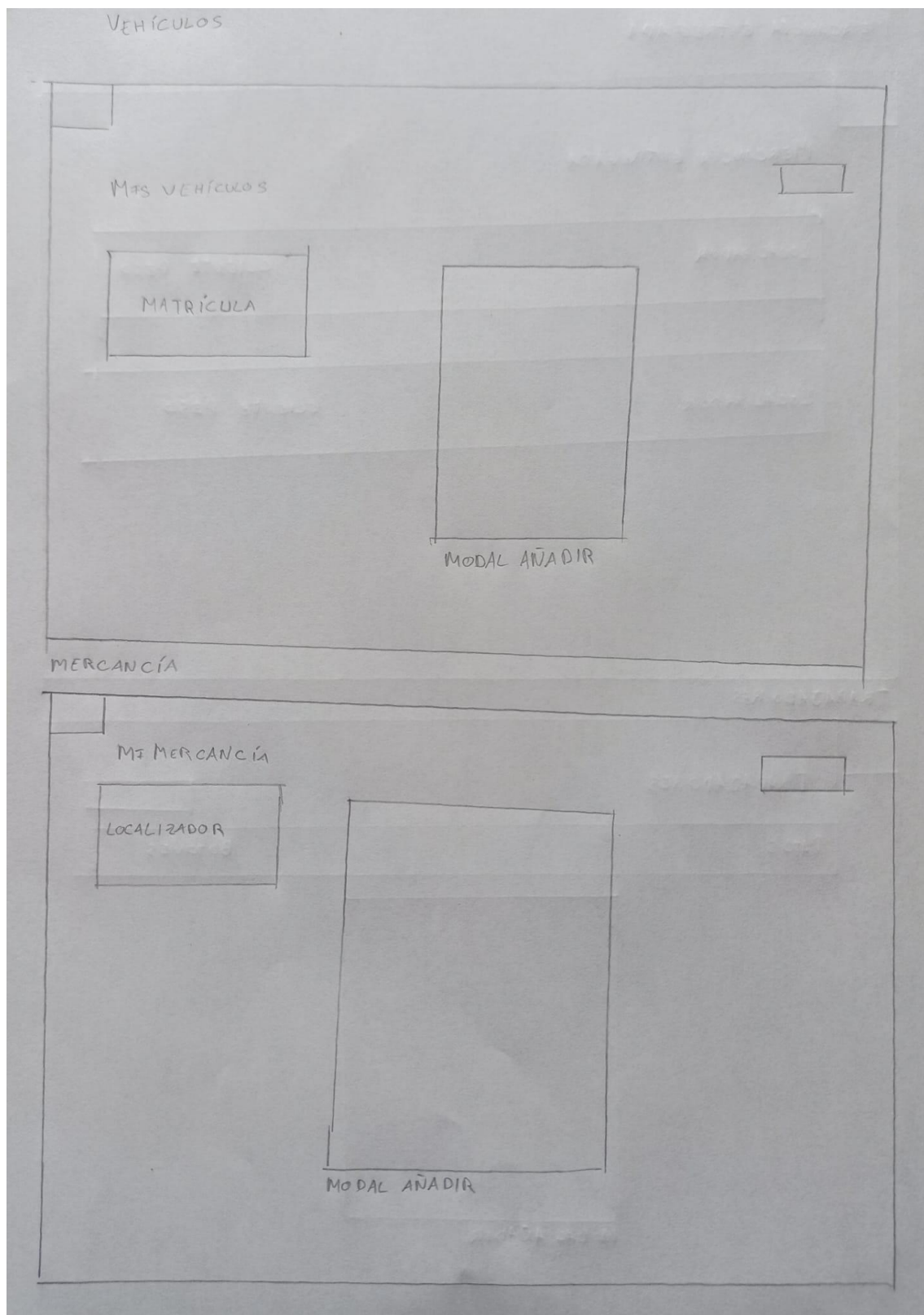


Imagen 5. Diseño vehículos y mercancía

MERCANCÍA ENTREGADA

MERCANCÍA ENTREGADA

LOCALIZADOR	CLIENTE PESO
-------------	--------------

LOCALIZADOR	CLIENTE PESO
-------------	--------------

TRABAJADORES

TRABAJADORES

DATOS	BOTONES
-------	---------

MODAL AÑADIR

Imagen 6. Diseño mercancía entregada y trabajadores

Cuentas Usuario

Cuentas de Usuario

Datos

ELIMINAR

Clientes

Clientes

Datos

ELIMINAR

Configuración

Configuración

Texto

Texto

Texto

Texto

Imagen 7. Diseño cuentas de usuario, clientes y configuración

Por último, se realizaron los prototipos finales.

Principal



Imagen 8. Diseño principal

Localizar paquete

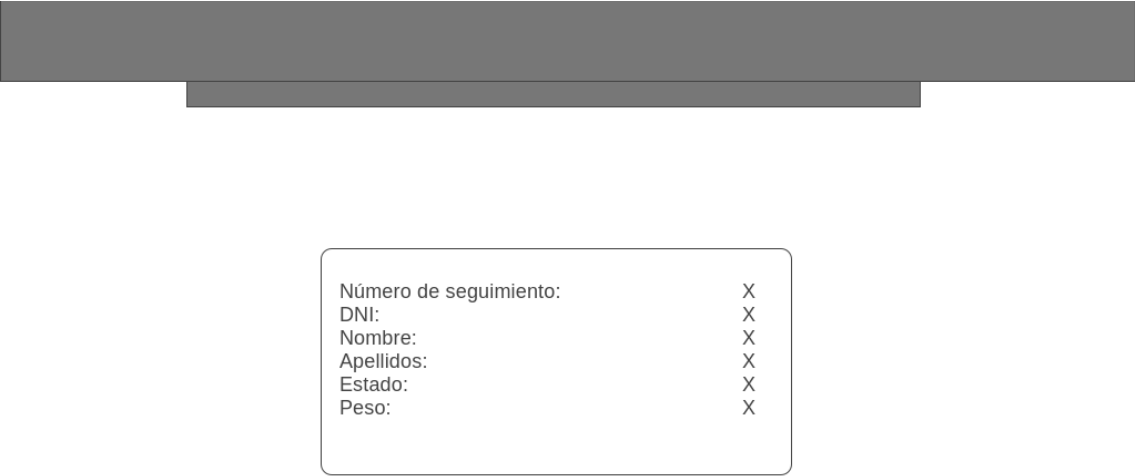
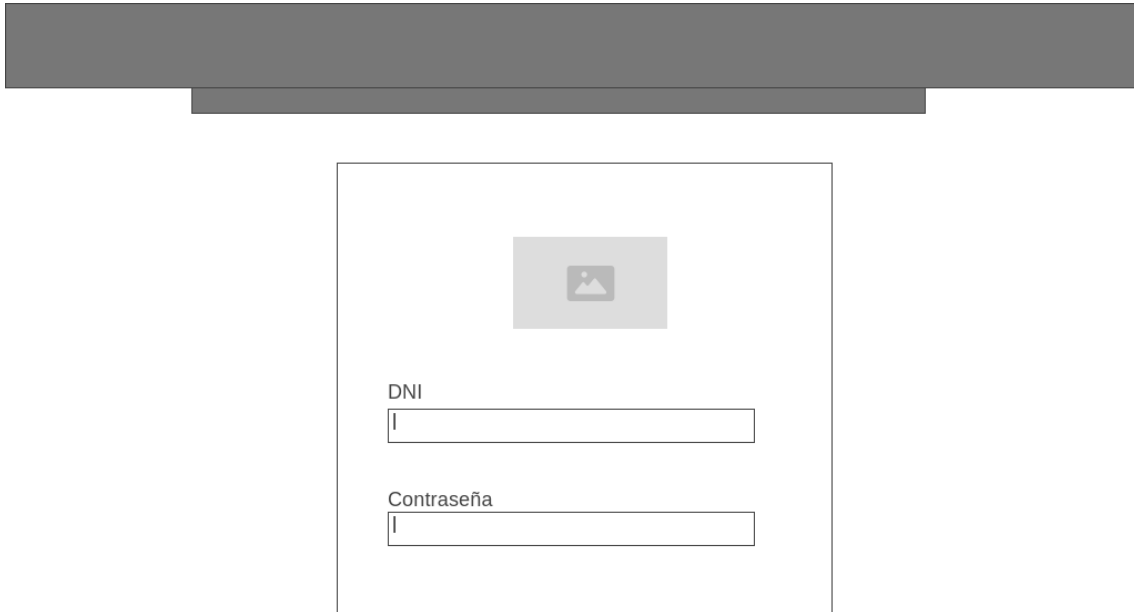


Imagen 9. Diseño localizar paquete

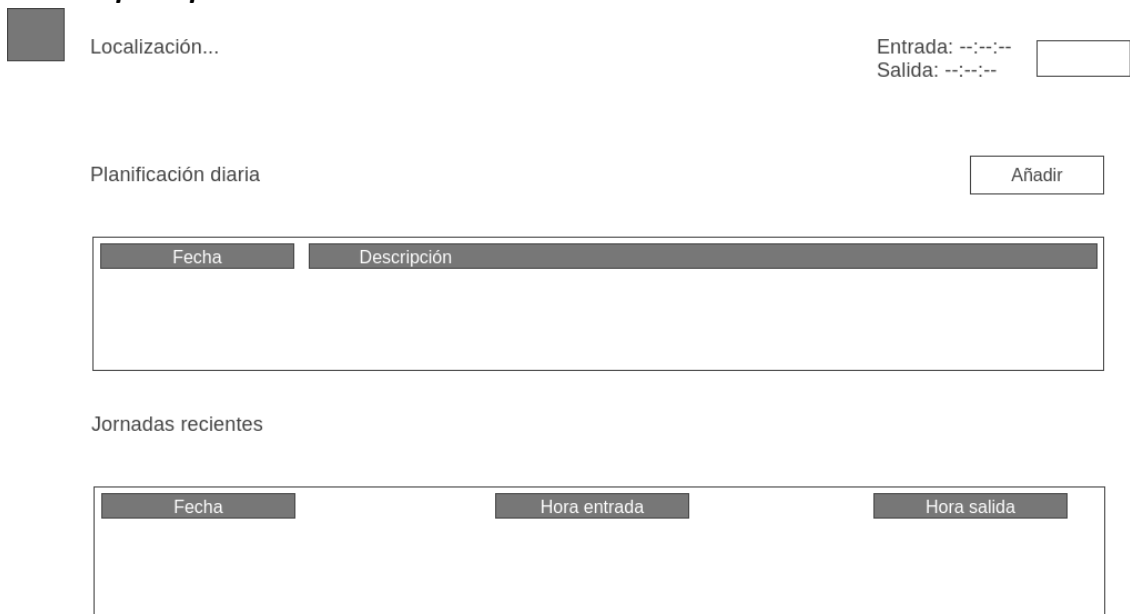
Login



The login form is centered on a light gray background. At the top, there is a dark gray header bar. Below the header, there is a light gray box containing a profile picture placeholder (a square with a camera icon). Below the profile picture, there are two input fields: one for 'DNI' and one for 'Contraseña' (Password). Both fields have a small vertical line on the left side, indicating a cursor.

Imagen 10. Diseño login

Gestión principal



The main management interface is divided into several sections. At the top, there is a dark gray header bar. Below the header, there is a section for 'Localización...' with a small square icon to the left. To the right of this, there are two input fields: 'Entrada: --:--:--' and 'Salida: --:--:--'. Below these, there is a section for 'Planificación diaria' with a button labeled 'Añadir'. Below this, there is a table with two columns: 'Fecha' and 'Descripción'. Below the table, there is a section for 'Jornadas recientes' with a table that has three columns: 'Fecha', 'Hora entrada', and 'Hora salida'.

Imagen 11. Diseño gestión principal

Vehículos

Localización...

Mis vehículos

Añadir

● Estado

Peso

Matricula

Nuevo vehículo

X

Matricula

Estado

Carga útil

Añadir

Modal

Imagen 12. Diseño vehículos

Mercancía

Localización...

Mi mercancía

Mercancía entregada

Añadir

● Estado

Peso

Localizador

Nueva mercancía

X

Peso

Estado

Descripción

Cliente existente

DNI

Añadir cliente

DNI

Nombre

Apellidos

Teléfono

Correo

Añadir

Modal

Imagen 13. Diseño mercancía

Mercancía entregada

Localización...

Mercancía entregada

Localizador

Vehículo: X Cliente: X Peso: X

Imagen 14. Diseño mercancía entregada

Trabajadores

Localización...

Trabajadores

Cuentas de usuario

Añadir

DNI --- Nombre --- Apellidos --- Teléfono --- Correo --- Fecha nacimiento --- Fecha inicio --- Cargo

Btn

Nuevo trabajador

X

DNI

Nombre

Apellidos

Teléfono

Correo

Cargo

Fecha nacimiento

Fecha inicio

Crear cuenta

Añadir

Modal

Imagen 15. Diseño trabajadores

Cuentas usuario

Localización...

Cuentas de usuario

DNI --- Nombre --- Apellidos --- Teléfono --- Correo --- Fecha nacimiento --- Fecha inicio --- Cargo

Eliminar

Imagen 9. Diseño cuentas usuario

Cientes



Localización...

Cientes

DNI: X Nombre: X Apellidos: X Teléfono: X Correo: X X

Imagen 16. Diseño clientes

Configuración



Localización...

Configuración

Guardar

DNI	<input type="text"/>
Nombre	<input type="text"/>
Apellidos	<input type="text"/>
Teléfono	<input type="text"/>
Correo	<input type="text"/>
Cargo	<input type="text"/>
Fecha nacimiento	<input type="text"/>
Fecha inicio empresa	<input type="text"/>
Contraseña	<input type="text"/>

Imagen 17. Diseño configuración

- Logos

Logos creados con el uso de distintas herramientas, la principal fue Photoshop.



Imagen 18. Logo página principal



Imagen 19. Logo página iniciar sesión

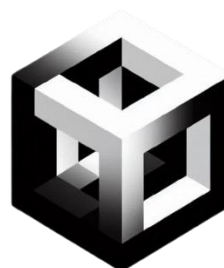


Imagen 20. Logo superior navegador

- Modelo entidad-relación

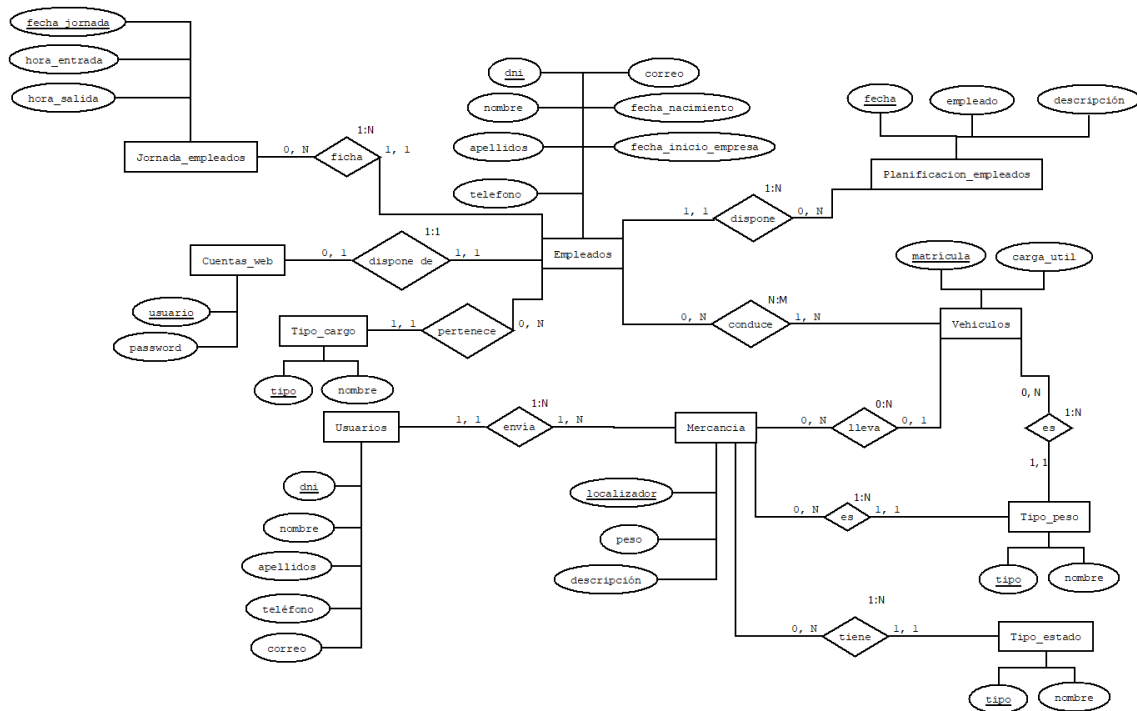


Imagen 21. Modelo E-R

- Casos de uso

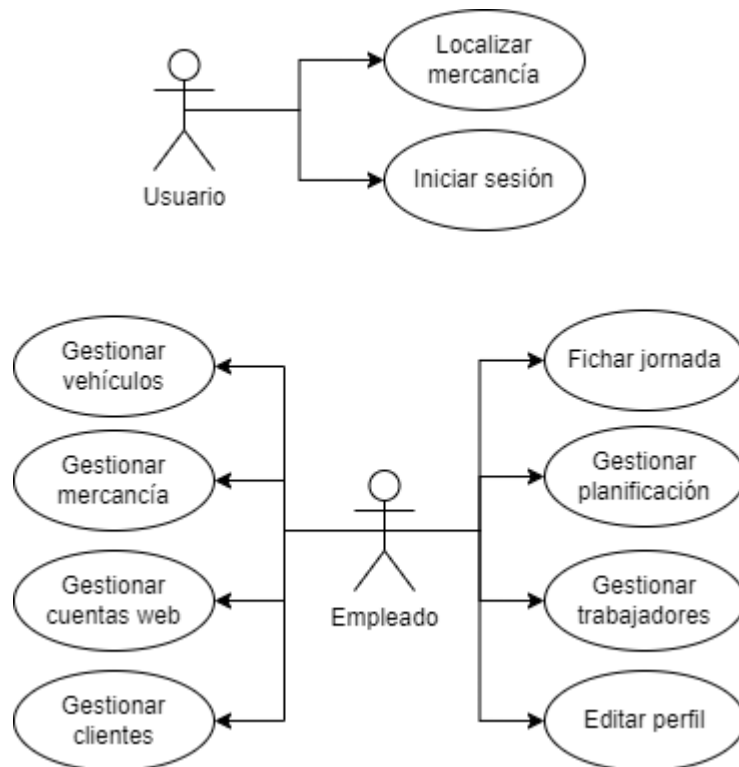


Imagen 22. Casos de uso

- Modelo relacional y normalización

Todas las tablas están normalizadas hasta al menos su **2FN**. En **negrita** las **PK** y subrayadas las **FK**.

Empleados (**dni**, nombre, apellidos, teléfono, correo, fecha_nacimiento, fecha_inicio_empresa, cargo)

Cuentas_web (**dni_ empleado**, password)

Jornada_empleados (**dni_ empleado**, **fecha_jornada**, **hora_entrada**, **hora_salida**)

Mercancía (**localizador**, cliente, peso, tipo_estado, tipo_peso, descripción)

Usuarios (**dni**, nombre, apellidos, teléfono, correo)

Planificación_empleados (**fecha**, **empleado**, **descripción**)

Vehículos (**matrícula**, carga_útil, tipo_peso, tipo_estado)

Transporte_mercancía (**localizador**, **matricula**)

Tipo_cargo (**tipo**, nombre)

Tipo_estado_mercancía (**tipo**, nombre)

Tipo_estado_vehículo (**tipo**, nombre)

Tipo_peso (**tipo**, nombre)

- Base de datos

Tabla: Empleados

CAMPO	TIPO	NULO	DESCRIPCIÓN
DNI	Cadena	no	PK
NOMBRE	Cadena	no	
APELLIDOS	Cadena	no	
TELÉFONO	Numérico	no	
CORREO	Cadena	no	
FECHA_NACIMIENTO	Numérico	no	
FECHA_INICIO_EMPRESA	Numérico	no	
CARGO	Numérico	no	FK (tipo_cargo)

Tabla 1. Empleados

Tabla: Cuentas_web

CAMPO	TIPO	NULO	DESCRIPCIÓN
DNI_EMPLEADO	Cadena	no	PK, FK (empleados)
PASSWORD	Cadena	no	Encriptado

Tabla 2. Cuentas_web

Tabla: Jornada_empleados

CAMPO	TIPO	NULO	DESCRIPCIÓN
DNI_EMPLEADO	Cadena	no	PK, FK (empleados)
FECHA_JORNADA	Cadena	no	PK
HORA_ENTRADA	Cadena	no	PK
HORA_SALIDA	Cadena	no	PK

Tabla 3. Jornada_empleados

Tabla: Mercancía

CAMPO	TIPO	NULO	DESCRIPCIÓN
LOCALIZADOR	Numérico	no	PK, AI
CLIENTE	Cadena	no	
PESO	Float	no	
TIPO_ESTADO	Numérico	no	FK (tipo_estado_mercancía)
TIPO_PESO	Numérico	no	FK (tipo_peso)
DESCRIPCIÓN	Cadena	no	

Tabla 4. Mercancía

Tabla: Usuarios

CAMPO	TIPO	NULO	DESCRIPCIÓN
DNI	Cadena	no	PK
NOMBRE	Cadena	no	
APELLIDOS	Cadena	no	
TELEFONO	Numérico	no	
CORREO	Cadena	no	

Tabla 5. Usuarios

Tabla: Planificación_empleados

CAMPO	TIPO	NULO	DESCRIPCIÓN
FECHA	Numérico	no	PK
EMPLEADO	Cadena	no	PK, FK (empleados)
DESCRIPCIÓN	Cadena	no	PK

Tabla 6. Planificación_empleados

Tabla: Vehículos

CAMPO	TIPO	NULO	DESCRIPCIÓN
MATRÍCULA	Cadena	no	PK
CARGA_UTIL	Float	no	
TIPO_PESO	Numérico	no	FK (tipo_peso)
TIPO_ESTADO	Numérico	no	FK (tipo_estado_vehículo)

Tabla 7. Vehículos

Tabla: Transporte_mercancía

CAMPO	TIPO	NULO	DESCRIPCIÓN
LOCALIZADOR	Numérico	no	PK, FK (mercancía)
MATRÍCULA	Cadena	no	PK, FK (vehículos)

Tabla 8. Transporte_mercancía

Tabla: Tipo_cargo

CAMPO	TIPO	NULO	DESCRIPCIÓN
TIPO	Numérico	no	PK
NOMBRE	Cadena	no	

Tabla 9. Tipo_cargo

Tabla: Tipo_estado_mercancía

CAMPO	TIPO	NULO	DESCRIPCIÓN
TIPO	Numérico	no	PK
NOMBRE	Cadena	no	

Tabla 10. Tipo_estado_mercancía

Tabla: Tipo_estado_vehículo

CAMPO	TIPO	NULO	DESCRIPCIÓN
TIPO	Numérico	no	PK
NOMBRE	Cadena	no	

Tabla 11. Tipo_estado_vehículo

Tabla: Tipo_peso

CAMPO	TIPO	NULO	DESCRIPCIÓN
TIPO	Numérico	no	PK
NOMBRE	Cadena	no	

Tabla 12. Tipo_peso

- Diseño físico de la BD

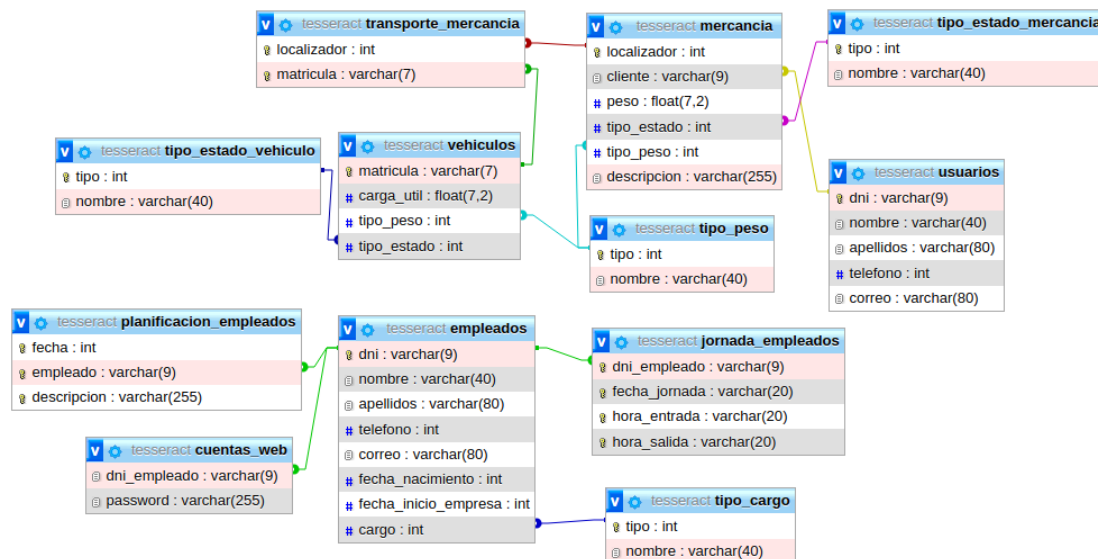


Imagen 23. Diseño físico de la BD

- Diagrama de clases UML

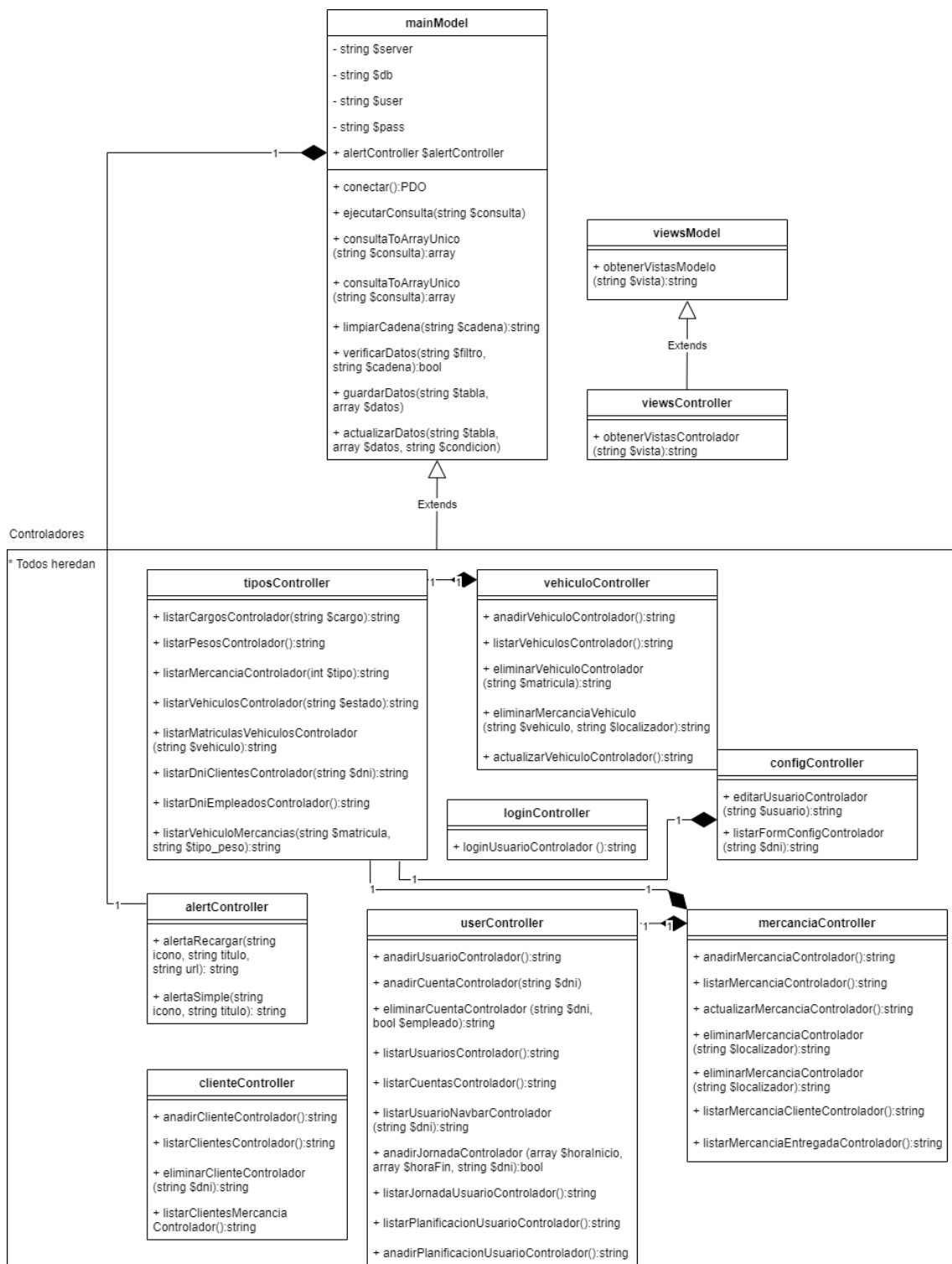


Imagen 24. Diagrama de clases UML

- Estructura de carpetas

La estructura será la siguiente:

- app
 - ajax
 - config
 - controllers
 - models
 - views
 - content
 - css
 - img
 - inc
 - js

La carpeta **app** contendrá todos los archivos de la aplicación web, excepto el index y el archivo htaccess.

En la carpeta **ajax** encontraremos todos los archivos php que sean llamados desde llamadas asíncronas.

En la carpeta **config** estarán todos los archivos de configuración.

En la carpeta **controllers** estarán contenidos los controladores, son la parte intermedia entre las vistas y los modelos.

En la carpeta **models** estarán contenidos los modelos, solo estos pueden interactuar directamente con la base de datos.

En la carpeta **views** encontraremos más carpetas relacionadas con las vistas, la parte visible que podrá ver el usuario.

En la carpeta **content** estarán las vistas como tal, solo estas serán las que vea el usuario.

En la carpeta **css** estarán contenidos todos los archivos css.

En la carpeta **img** estarán contenidas las imágenes de la web.

En la carpeta **inc** estarán contenidos los archivos que se incluyen en otros.

En la carpeta **js** estarán contenidos todos los archivos JavaScript.

3.2.3. Codificación

Esta es la fase más larga, se codifica todo lo necesario para que el software creado sea correcto y contenga el menor número de errores posible.

La codificación seguirá el modelo vista controlador (MVC), un modelo que relaciona la vista de los usuarios con la base de datos (modelos) a través de los controladores. Es una implementación sencilla de realizar y más cercana a los entornos de trabajo reales.

3.2.4. Pruebas

Durante la misma codificación se van realizando pruebas unitarias para encontrar errores ocultos. Al finalizar el desarrollo, se le da la aplicación a probar a un grupo de usuarios para que hagan uso del software durante varios días para detectar errores y poder corregirlos posteriormente.

3.2.5. Despliegue

Durante el despliegue, se pone en marcha la aplicación web para su posterior explotación, a lo largo de esta fase se pueden detectar errores y se deben corregir.

3.2.6. Mantenimiento

Se mantiene el software a lo largo de su vida útil, corrigiendo errores que hubiesen quedado ocultos y añadiendo nuevas funcionalidades que sean necesarias.

3.3. Herramientas de diseño

3.3.1. Herramientas en el lado cliente

- **HTML**: lenguaje de marcas utilizado para la estructura visual de la aplicación web.
- **CSS**: lenguaje de diseño utilizado para dar formato visual.
- **JavaScript**: utilizado para dotar de funcionalidad la aplicación.
- **Bootstrap**: framework utilizado como complemento al lenguaje css para dar diseño y formato.
- **SweetAlert2**: librería de JavaScript utilizada para dotar de funcionalidad en forma de alertas.
- **FontAwesome**: librería de iconos.

3.3.2. Herramientas en el lado servidor

- **PHP**: lenguaje utilizado para realizar toda la lógica interna invisible para el usuario.
- **phpMyAdmin**: herramienta para la gestión de la base de datos
- **MySQL**: base de datos utilizada para almacenar la información de la aplicación.
- **Apache2**: servidor web utilizado para alojar la aplicación.

3.3.3. Herramientas de desarrollo

- **Visual Studio Code** y **phpstorm**: entornos de trabajo para la codificación.
- **Oracle VM VirtualBox 7.0**: software de virtualización para crear un servidor aislado en el que realizar el proyecto.
- **Linux Mint 21.2**: sistema operativo utilizado para el servidor.

3.4. Evaluación económica y financiación del proyecto

3.4.1. Estudio de viabilidad

Este proyecto es viable económicamente ya que los únicos requisitos que necesita es un servidor donde almacenar la aplicación y la base de datos, lo cual cuesta unos 50€ al mes, dependiendo como de grande sea la empresa que hará uso del software y cuantos datos maneje habitualmente.

Respecto al tiempo también es viable, ya que se disponen de 3 meses para realizarlo y solo vamos a abarcar la gestión básica de la empresa, no es un software a medida para una empresa en concreto, sino que más adelante, en caso de que una empresa comprase el software, se añadiría la funcionalidad necesaria para la parte específica de cada una.

3.4.2. Ayudas económicas

No existen ayudas económicas para realizar este proyecto ya que es realizado solo por una persona gratuitamente en forma de proyecto final. Solo se necesitará una financiación para el mantenimiento, pero al ser una aplicación tan poco costosa solo será necesario pagar el alojamiento de la misma, valorado en unos 50€ al mes, es por ello que para un gasto tan insignificante no existen ayudas posibles.

4. Planificación del proyecto

4.1 Diagrama de secuenciación del proyecto

Para el desarrollo del proyecto he utilizado una metodología ágil en espiral, en las cuales se han identificado las siguientes fases:

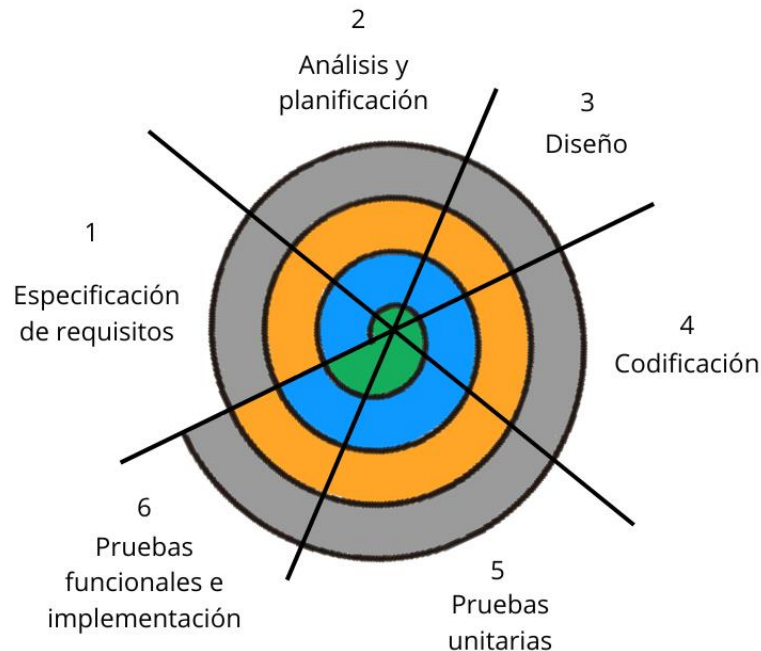


Imagen 25. Diagrama metodología ágil

4.2. Secuenciación de actividades o tareas

Por cada fase del [punto anterior](#) identificamos las siguientes tareas:

1. Especificación de requisitos
 - Requisitos funcionales (funcionalidad de la aplicación, que va a hacer...)
 - Requisitos no funcionales (velocidad, magnitud de los datos...)
2. Análisis y planificación
 - Análisis de los requisitos
 - Planificación de las tareas que se llevarán a cabo
3. Diseño
 - Especificación de las tecnologías
 - Diseños de las interfaces
 - Logos
 - Modelo E-R
 - Casos de uso
 - Modelo relacional y normalización
 - Base de datos y diseño físico
 - Diagrama de clases

4. Codificación

- Vistas de usuario
- Archivo htaccess
- Modelos
- Controladores

5. Pruebas unitarias

- Pruebas de la codificación realizada de forma unitaria

6. Pruebas funcionales e implementación

- Pruebas del conjunto de la aplicación
- Implementación en el entorno de producción

4.3 Asignación de recursos y tiempos en las actividades del proyecto

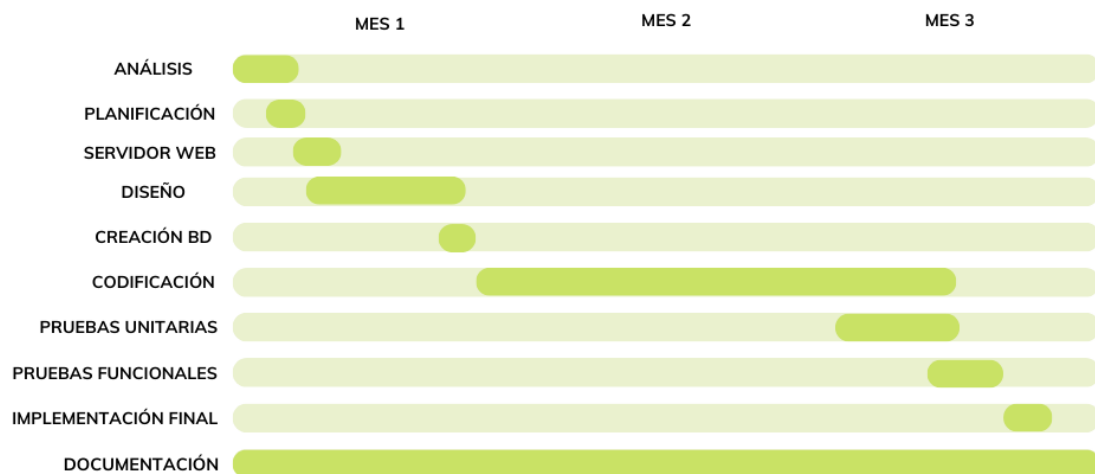


Imagen 26. Diagrama de Gantt, asignación de recursos y tiempos

4.4. Procedimientos en las actividades.

Los siguientes procedimientos se realizan en base a la metodología ágil elegida, en este caso, en espiral, es por ello que los procedimientos no tienen un principio y fin marcado, se desarrolla pasando varias veces por los mismos procedimientos, cada vuelta será más extensa que la anterior hasta el desarrollo final.

En cada tarea del [punto 4.2](#) podemos distinguir los siguientes procedimientos:

- Requisitos funcionales

Se detalla cuál va a ser la funcionalidad de la aplicación, que queremos abarcar y que no.

- Requisitos no funcionales

Se detalla cuales son los requisitos que no son parte de la funcionalidad, cuántos datos se van a manejar, velocidades...

- **Análisis de los requisitos**
Analizamos los requisitos anteriores para tener una idea clara de las tareas que se deben realizar en cada fase del desarrollo.
- **Planificación de las tareas que se llevarán a cabo**
Se hace una planificación de las tareas analizadas, se utiliza un software de gestión de proyectos, en este caso Trello, para especificar las tareas y ver el estado de ellas en cada momento. En este punto también especificamos que tareas tienen prioridad sobre otras, tiempos de desarrollo...
- **Especificación de las tecnologías**
Según el análisis anterior, especificamos que tecnología es la más adecuada en cada caso para los requisitos descritos.
- **Diseños de las interfaces**
Hacemos un primer acercamiento a lo que será la interfaz final a partir de unos prototipos o diseños.
- **Logos**
Generamos los logos de nuestra aplicación.
- **Modelo E-R**
Se crea el modelo entidad relación en base a nuestro análisis, abarcando los máximos requisitos posibles.
- **Casos de uso**
Creamos los casos de uso de la aplicación en base a la funcionalidad.
- **Modelo relacional y normalización**
A partir de nuestro modelo E-R hacemos un primer modelo relacional, si no está normalizado, debemos normalizarlo como mínimo hasta la 2FN
- **Base de datos y diseño físico**
Se diseña la base de datos a partir de nuestro modelo relacional.
- **Diagrama de clases**
Creamos el diagrama de clases, contendrá las propiedades y los métodos de todas las clases.
- **Vistas de usuario**
Codificamos las vistas que verán los usuarios, según avance la codificación de otras partes seguiremos con estas.

- Archivo htaccess
Se crea el archivo .htaccess, es fundamental para el funcionamiento de nuestra aplicación ya que realiza una conversión de la url.
- Modelos
Codificación de los modelos principales que interactuaran con la base de datos.
- Controladores
Con los modelos ya creados, se codifican los controladores y se implementan junto las vistas y los modelos.
- Pruebas de la codificación realizada de forma unitaria
Se realizan las pruebas de cada componente codificado de forma unitaria.
- Pruebas del conjunto de la aplicación
Si se pasan las pruebas unitarias, se realizan pruebas funcionales del conjunto.
- Implementación en el entorno de producción
Se implementa en el entorno de producción las nuevas funcionalidades.

4.5. Prevención de riesgos en el desarrollo del proyecto (Retrasos, bloqueos..etc.)

Para el desarrollo del proyecto se han usado distintas técnicas de prevención de riesgos, a continuación, se detallan algunas de ellas:

- Realizar una planificación detallada, creando un plan con tareas claras y concisas para no “irse por las ramas” durante el desarrollo.
- Uso de Trello (software de gestión de proyectos) para evaluar en todo momento las tareas restantes y de esta forma organizar los posibles retrasos.
- Control de versiones con Git para revertir cambios no deseados o posibles confusiones.
- Desarrollo haciendo uso de la técnica TDD para prevenir futuros fallos de la aplicación.

5. Pruebas y validación del Proyecto

5.1. Evaluación y seguimiento de las actividades.

Las tareas han ido según lo planificado gracias a la metodología utilizada, ya que en caso de usar otra, muy probablemente tendríamos que haber pasado de nuevo por alguna tarea para finalizarla, de esta forma se ha ido poco a poco según lo necesitado en cada ocasión.

El calendario también fue según lo planeado excepto un apartado, en el apartado en el que se diseñaban las interfaces a lo largo del mes 1, tuve que volver a realizar alguna más que me di cuenta que íbamos a necesitar, por lo que entre el mes 2 y 3 tuve que volver a hacer un par de diseños de interfaces más.

5.2. Indicadores de calidad. Elaborar una batería de pruebas para detectar errores.

Localizar mercancía					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Localizar mercancía existente	Localizador: Existente	Existe	Existe	Ok	
	Localizador: No existente	No existe	No existe	Ok	
No enviar localizador	Localizador: nulo	Redirección	Redirección	Ok	Se modifica el formulario para no enviarlo.

Tabla 13. Pruebas localizar mercancía

Iniciar sesión					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Inicio de sesión	DNI: Existente Contraseña: Correcta	Correcto	Correcto	Ok	
	DNI: No existente	No existe	No existe	Ok	
	DNI: Existente Contraseña: Incorrecta	Incorrecto	Incorrecto	Ok	
	DNI: Formato no válido	Incorrecto	Incorrecto	Ok	

Tabla 14. Pruebas iniciar sesión

Planificación					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Añadir planificación	Fecha inferior al día actual	No añadida	No añadida	Ok	
	Empleado no existente	No añadida	No añadida	Ok	
	DNI con formato inválido	No añadida	No añadida	Ok	
	Sin descripción	No añadida	No añadida	Ok	
	Todos los campos rellenos de forma válida	Añadida	Añadida	Ok	

Tabla 15. Pruebas planificación

Jornada					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Iniciar jornada		Iniciada	Iniciada	Ok	
		Iniciada	Iniciada	Ok	Salir de la aplicación y al volver se queda guardada.
Finalizar jornada		Añadida	Añadida	Ok	
Añadir jornadas		Añadidas	Añadidas	Ok	Sin recargar la página añadir y finalizar varias veces seguidas ya que hace uso de AJAX.

Tabla 16. Pruebas jornada

Vehículo					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Añadir vehículo	Matricula con formato no válido	No añadido	No añadido	Ok	
	Estado vacío	No añadido	No añadido	Ok	
	Estado incorrecto	No añadido	No añadido	Ok	
	Carga útil igual o menor a 0	No añadido	No añadido	Ok	
	Tipo de peso vacío	No añadido	No añadido	Ok	
	Tipo de peso no válido	No añadido	No añadido	Ok	
	Todos los campos válidos	Añadido	Añadido	Ok	
Eliminar vehículo	Existe	Eliminado	Eliminado	Ok	
	No existe	No eliminado	No eliminado	Ok	
Editar vehículo	Sin cambios	No editado	No editado	Ok	
	Estado no válido	No editado	No editado	Ok	
	Estado válido	Editado	Editado	Ok	
Asignar mercancía	No existe	No asignada	No asignada	Ok	
	Tipo de peso distinto	No asignada	No asignada	Ok	
	Supera la carga máxima	No asignada	No asignada	Ok	
	La mercancía ya fue entregada	No asignada	No asignada	Ok	
	Todo correcto	Asignada	Asignada	Ok	
Desasignar mercancía	No está en el vehículo	Incorrecto	Incorrecto	Ok	
	Está en el vehículo	Desasignada	Desasignada	Ok	

Tabla 17. Pruebas vehículo

Mercancía					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Añadir mercancía	Peso igual o menor a 0	No añadida	No añadida	Ok	
	Estado vacío	No añadida	No añadida	Ok	
	Estado incorrecto	No añadida	No añadida	Ok	
	Cliente no existente	No añadida	No añadida	Ok	Seleccionando opción cliente existente
	DNI sin rellenar	No añadida	No añadida	Ok	Seleccionando opción cliente nuevo
	DNI con formato incorrecto	No añadida	No añadida	Ok	
	Nombre sin rellenar	No añadida	No añadida	Ok	
	Apellidos sin rellenar	No añadida	No añadida	Ok	
	Teléfono sin rellenar	No añadida	No añadida	Ok	
	Teléfono con formato incorrecto	No añadida	No añadida	Ok	
	Correo sin rellenar	No añadida	No añadida	Ok	
	Todos los campos correctos	Añadida	Añadida	Ok	
Eliminar mercancía	No existe	No eliminada	No eliminada	Ok	
	Si existe	Eliminada	Eliminada	Ok	
	Existe y está asignada	Eliminada	Eliminada	Ok	Se desasigna
	Existe y está entregada	Eliminada	Eliminada		
Editar mercancía	Sin cambios	No editada	No editada	Ok	
	Cliente vacío	No editada	No editada	Ok	
	Cliente no válido	No editada	No editada	Ok	
	Estado vacío	No editada	No editada	Ok	
	Estado no válido	No editada	No editada	Ok	
	Cambios correctos	Editada	Editada	Ok	
	Cambios correctos	Editada	Editada	Ok	Si el estado es entregado se pasa a mercancía entregada y se desasigna.

Tabla 18. Pruebas mercancía

Trabajadores					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Añadir trabajador	DNI sin rellenar	No añadido	No añadido	Ok	
	DNI con formato no válido	No añadido	No añadido	Ok	
	Nombre sin rellenar	No añadido	No añadido	Ok	
	Apellidos sin rellenar	No añadido	No añadido	Ok	
	Teléfono sin rellenar	No añadido	No añadido	Ok	
	Teléfono con formato no válido	No añadido	No añadido	Ok	
	Correo sin rellenar	No añadido	No añadido	Ok	
	Cargo sin rellenar	No añadido	No añadido	Ok	
	Cargo incorrecto	No añadido	No añadido	Ok	
	Fecha de nacimiento sin rellenar	No añadido	No añadido	Ok	
	Fecha inicio empresa sin rellenar	No añadido	No añadido	Ok	
	Campos correctos	Añadido	Añadido	Ok	
	Campos correctos y crear cuenta web	Añadido	Añadido	Ok	Añade también una cuenta web
Eliminar trabajador	Existe	Eliminado	Eliminado	Ok	Si tiene cuenta se elimina también la cuenta
	No existe	No eliminado	No eliminado	Ok	
Crear cuenta web	Existe trabajador	Creada	Creada	Ok	Usuario: DNI Contraseña: DNI
	No existe trabajador	No creada	No creada	Ok	
Eliminar cuenta web	Existe cuenta	Eliminada	Eliminada	Ok	Si elimina la propia cuenta, se cierra la sesión
	No existe cuenta	No eliminada	No eliminada	Ok	

Tabla 19. Pruebas trabajadores

Clientes					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Añadir cliente	DNI sin rellenar	No añadido	No añadido	Ok	
	DNI con formato no válido	No añadido	No añadido	Ok	

	Nombre sin rellenar	No añadido	No añadido	Ok	
	Apellidos sin rellenar	No añadido	No añadido	Ok	
	Teléfono sin rellenar	No añadido	No añadido	Ok	
	Teléfono con formato no válido	No añadido	No añadido	Ok	
	Correo sin rellenar	No añadido	No añadido	Ok	
	Campos correctos	Añadido	Añadido	Ok	
Eliminar cliente	Existe	Eliminado	Eliminado	Ok	
	No existe	No eliminado	No eliminado	Ok	

Tabla 20. Pruebas clientes

Configuración					
Acción	Parámetros	Valor esperado	Valor obtenido	Resultado	Observaciones
Editar configuración propia	Sin cambios	No editada	No editada	Ok	En caso de modificar alguno de esos datos en concreto
	Nombre sin rellenar	No editada	No editada	Ok	
	Apellidos sin rellenar	No editada	No editada	Ok	
	Teléfono sin rellenar	No editada	No editada	Ok	
	Teléfono con formato no válido	No editada	No editada	Ok	
	Correo sin rellenar	No editada	No editada	Ok	
	Cargo sin rellenar	No editada	No editada	Ok	
	Cargo no válido	No editada	No editada	Ok	
	Fecha de nacimiento sin rellenar	No editada	No editada	Ok	
	Fecha de inicio sin rellenar	No editada	No editada	Ok	
	Contraseña sin rellenar	No editada	No editada	Ok	
	Cambios correctos	Editada	Editada	Ok	
De otro empleado	Empleado autorizado	Editada	Editada	Ok	Administradores y RR.HH
	Empleado no autorizado	No editada	No editada	Ok	Solo puede actualizar su propio usuario
	Editar contraseña	No editada	No editada	Ok	Empleado sin cuenta web
		Editada	Editada	Ok	Empleado con cuenta web

Tabla 21. Pruebas configuración

5.3. Informe de evaluación de incidencias. Solución de las mismas.

Tuve una incidencia al implementar la asincronía con AJAX al añadir jornadas laborales de los empleados, el formato en el que se insertaban las horas era incorrecto al que necesitaba, la solución fue darle formato directamente en el servidor en vez de en el cliente.

También tuve problemas al usar la librería de SweetAlert2, me hubiese gustado mostrar las alertas directamente desde JavaScript, pero para esto hubiese necesitado crear todo el programa de forma asíncrona, esto no era buena idea ya que había ocasiones que necesitaba recargar la página para recargar los buffer del navegador y del servidor para no perder rendimiento y velocidad, la solución fue crear un controlador desde el servidor que mostrase la alerta que necesitábamos y realizar una composición desde el modelo principal, del cual heredan el resto de controladores.

Por último, he tenido que llevar mucho cuidado con que podía hacer cada usuario, al tener bastantes opciones en cada CRUD, he tenido varias incidencias en las que permitía acciones a usuarios que no debían tener permiso, la solución a esto ha sido el realizar muchas pruebas y probar mucho la aplicación para ver que podía hacer cada rol en todo momento.

6. Implantación del proyecto

6.1. Plan de implantación.

6.1.1. Implantación en máquina virtual

Duración: 1 hora. Para esta opción debemos hacer uso de un software que permita la instalación de máquinas virtuales, la opción para la que se dará soporte de implementación será usando el software Oracle VM VirtualBox 7.0, el sistema operativo que utilizaremos será Linux Mint 21.2.

Descargar imagen VM

Debemos descargar la imagen de la máquina virtual en nuestro equipo, esta imagen se llama "Linux Mint 21.ova".

Importación en VirtualBox 7.0 o superior

En el menú superior de la aplicación debemos pulsar sobre Archivo > Importar servicio virtualizado

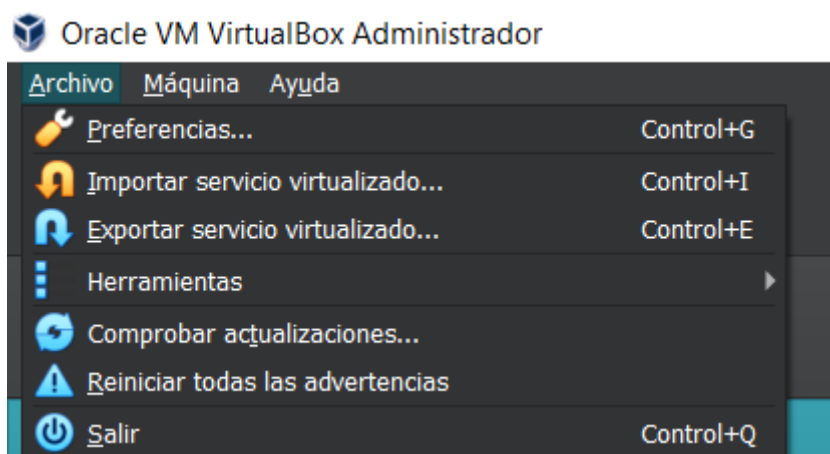


Imagen 27. Añadir máquina VirtualBox

Seleccionamos la imagen que nos hemos descargado en el [apartado anterior](#).



Imagen 28. Seleccionar imagen

Por último, podemos modificar su configuración y pulsamos en Terminar.



Imagen 29. Confirmar máquina virtual

Una vez termine la implantación ya nos aparecerá la máquina instalada, podremos cambiar su configuración o iniciarla.

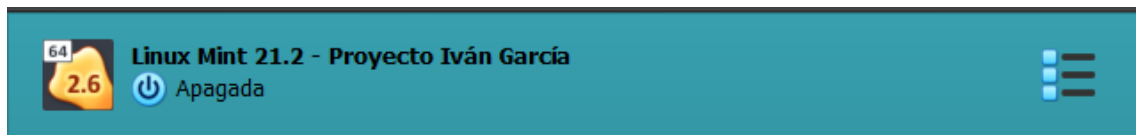


Imagen 30. Máquina virtual implantada

6.1.2. Implantación en servidor propio

Duración: 5-10 horas. Para esta opción debemos tener en cuenta que **no se dará soporte en caso de fallos no relacionados con la aplicación web**, también debemos tener en cuenta que al hacer uso de un servidor propio es más probable que tengamos distintos fallos a lo largo de la implementación o que la aplicación web haga uso de componentes no compatibles con el servidor. Aunque el sistema operativo sea igual al utilizado para el desarrollo puede tener problemas de incompatibilidades que habrá que solucionar manualmente.

Prerrequisitos

El servidor debe contar como mínimo con los siguientes componentes:

- Apache 2.4 (o servidores similares que permitan archivos .htaccess)
 - Módulos por defecto
 - Módulo rewrite
- PHP 8.1
- phpMyAdmin

Descarga de archivos

Debemos descargar los archivos contenidos en la carpeta “Archivos” o bien desde GitHub: <https://github.com/Ixvan22/ProyectoFinal>

Implementación en servidor web

En nuestro servidor web apache debemos indicar la ruta hacia el directorio que están contenidos nuestros archivos.

Habilitar archivos .htaccess

En nuestro servidor web debemos habilitar el uso de archivos .htaccess (Nota: si utilizamos un servidor Nginx no podremos hacer uso de este tipo de archivos, debemos buscar otra opción para implementar su contenido, es imprescindible para el funcionamiento de la aplicación)

Importar base de datos

Debemos hacer uso de un gestor de base de datos. Importaremos el archivo “db.sql” incluido en la carpeta “Archivos/db”.

Una vez importado para mejorar la seguridad de nuestra aplicación eliminaremos la carpeta.

6.2. Manual de instalación.

6.2.1 Cambio de constantes del servidor

Modificaremos el archivo “app/config/server.php”.

DB_SERVER: pondremos el host de nuestro servidor de base de datos

DB_NAME: pondremos el nombre de la base de datos

DB_USER: pondremos un usuario válido con acceso a la base de datos

DB_PASS: pondremos la contraseña del usuario indicado en DB_USER

Cambios a tener en cuenta en seguridad

Si queremos mejorar significativamente la seguridad de nuestra aplicación deberíamos obtener estos datos a partir de otro lugar como puede ser variables de entorno dentro de nuestro sistema. En el archivo “app/models/mainmodel.php”, al inicio deberíamos obtener estas variables para no alterar el funcionamiento de nuestra aplicación.

6.2.2 Cambio de constantes de la aplicación

Modificaremos el archivo “app/config/app.php”.

APP_URL: pondremos la url de nuestro servidor web.

APP_SESSION_NAME: pondremos un nombre referente a la empresa (Ej. Tesseract).

En **date_default_timezone_set(...)** podremos cambiar la zona horaria, en la siguiente web podemos ver los tipos de zonas horarias soportadas con el valor que hay que poner en cada caso: <http://www.php.net/manual/en/timezones.php>.

6.2.3 Contraseñas

Para poder empezar a hacer uso de la aplicación se ha dejado creado un usuario administrador, en el login pondremos los siguientes datos:

DNI: "00000000Z" - Contraseña: "00000000Z"

Todas las contraseñas de la máquina virtual serán **"alumno"**.

El usuario administrador de la base de datos por defecto en la máquina virtual es **"root"** y contraseña **"alumno"**.

6.3. Manual de usuario.

6.3.1. Introducción

En este apartado se va a detallar toda la funcionalidad de la aplicación.

6.3.2. Sesión

Iniciar sesión

En url/login, introducimos el dni y la contraseña y pulsamos en iniciar sesión.

El DNI debe llevar un formato válido y debe existir en la aplicación, si el trabajador está registrado pero sin una cuenta web no podrá acceder.

Si es la primera vez que iniciamos sesión en la aplicación, la contraseña será el DNI, a no ser que nos la hayan cambiado antes algún administrador.

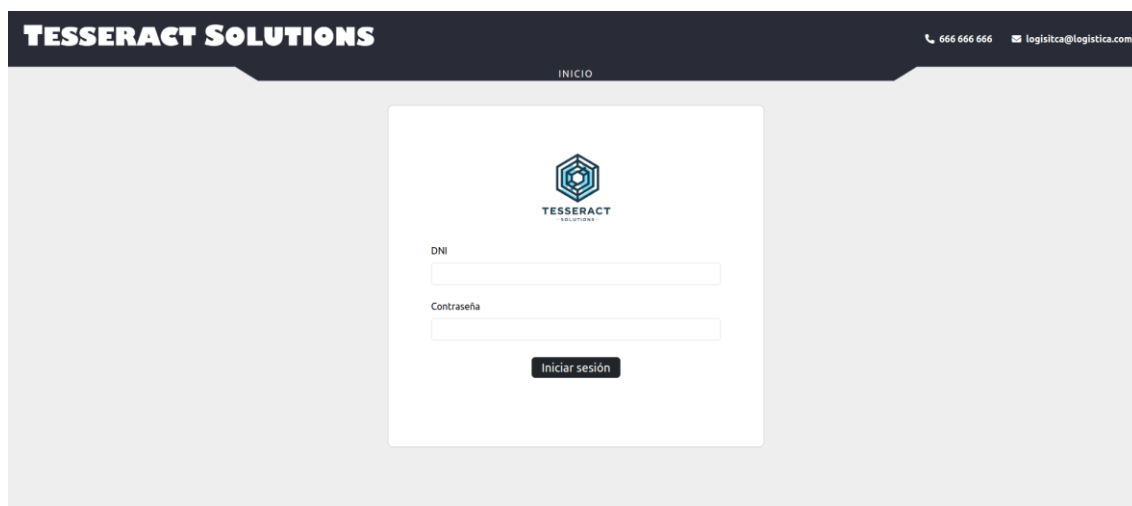


Imagen 31. Manual de usuario iniciar sesión

Cerrar sesión

Con la sesión iniciada, abrimos el navbar situado en el lateral izquierdo y pulsamos en “Salir”.

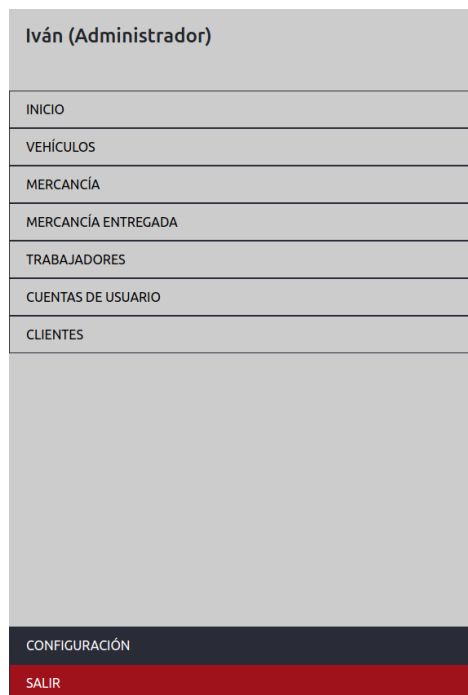


Imagen 32. Manual de usuario cerrar sesión

6.3.3 Jornada laboral

Fichar jornada

En url/gestionPrincipal pulsamos en “Iniciar”. Podemos salir de la aplicación una vez iniciada y se guardará al volver a entrar.



Imagen 33. Manual de usuario iniciar jornada

De la misma forma que se inicia la jornada, podemos finalizarla pulsando en “Finalizar”, al recargar la página podremos ver como consta en el primer registro de Jornadas recientes



Imagen 34. Manual de usuario finalizar jornada

Consultar jornadas

En url/gestionPrincipal, en el apartado inferior podremos consultar las jornadas más recientes. Tiene un límite, no se muestran todas las jornadas de la vida laboral del empleado, solo unas pocas.

Jornadas recientes

Fecha	Hora entrada	Hora salida
25/05/2024	14:07:30	14:07:42
25/05/2024	14:06:52	14:07:20

Imagen 35. Manual de usuario consultar jornadas

6.3.4. Planificación

Consultar planificación diaria

En url/gestionPrincipal, en el apartado superior podremos consultar nuestra planificación asociada en el día de hoy.

Planificación diaria

Añadir planificación

Fecha	Descripción
25/05/2024	Esta es una planificación de prueba.

Imagen 36. Manual de usuario consultar planificación

Añadir planificación a un empleado

En url/gestionPrincipal, en el apartado superior pulsamos en “Añadir planificación”. Se abre un menú, rellenamos los datos y pulsamos en “Añadir”.

Todos los datos son obligatorios, si no rellenamos alguno, no podremos añadir la planificación.

La fecha debe ser el mismo día del sistema o mayor, pero no podremos introducir una fecha anterior al sistema.

En la descripción se debe poner lo que queremos asignar a un usuario en el día que tengamos marcado.

Para añadir la planificación, debemos pulsar en el botón de “Añadir”.

Formulario de "Nueva planificación" con los siguientes campos:

- Fecha: dd/mm/aaaa (con icono de calendario)
- DNI: (con icono de flecha hacia abajo)
- Descripción: (campo de texto largo)
- Botón "Añadir" (verde)

Imagen 37. Manual de usuario añadir planificación

6.3.5. Vehículos

Añadir vehículo

En url/vehículos, en el apartado superior pulsamos en “Añadir vehículo”. Se abre un menú, rellenamos los datos y pulsamos en “Añadir”.

Todos los campos son obligatorios, si no introducimos algún campo no se insertará el vehículo.

La matrícula debe llevar un formato válido, los formatos son todas las matrículas españolas. Por ejemplo, 1234FFF y M1234FF serán matriculas válidas.

Se debe escoger el estado en el que queremos añadir el vehículo, debe ser uno válido del desplegable que se abre.

La carga útil debe ser mayor a 0, además debe ser de un tipo válido del desplegable que se abre.

Para añadir el vehículo, debemos pulsar en el botón de “Añadir”.



Formulario para añadir un nuevo vehículo. El formulario tiene un título "Nuevo vehículo" con un botón de cerrar (X) a la derecha. Hay tres campos de entrada: "Matrícula:" con un campo de texto, "Estado:" con un desplegable, y "Carga útil:" con un campo de texto y un desplegable para la unidad (KG). Hay un botón verde "Añadir" en la parte inferior derecha.

Imagen 38. Manual de usuario añadir vehículo

Eliminar vehículo

En url/vehículos, seleccionamos un vehículo y pulsamos en “Eliminar” en el apartado inferior.



Formulario para eliminar un vehículo. El formulario tiene un título "Vehículo: 1234AAA" con un botón de cerrar (X) a la derecha. Hay dos campos de entrada: "Añadir mercancía:" con un desplegable, y "Estado:" con un desplegable que muestra "Cargando mercancía". Hay tres botones en la parte inferior: "Eliminar" (rojo), "Mercancía asignada" (gris), y "Guardar cambios" (verde).

Imagen 39. Manual de usuario eliminar vehículo

Asignar mercancía

En url/vehículos, seleccionamos un vehículo rellenos el apartado “Añadir mercancía” y pulsamos en “Guardar cambios”.

La suma de las mercancías añadida no puede superar la carga útil del vehículo, tampoco podemos añadir una mercancía de un tipo que no sea el mismo que el del vehículo.

Para asignar la mercancía, debemos pulsar en el botón de “Guardar cambios”.

The screenshot shows a form titled 'Vehículo: 1234AAA' with a close button (X) in the top right corner. Below the title, there are two dropdown menus: 'Añadir mercancía:' and 'Estado:'. The 'Estado:' dropdown is currently set to 'Cargando mercancía'. At the bottom of the form, there are three buttons: 'Eliminar' (red), 'Mercancía asignada' (grey), and 'Guardar cambios' (green).

Imagen 40. Manual de usuario asignar mercancía

Desasignar mercancía

En url/vehículos, seleccionamos un vehículo, pulsamos en “Mercancía asignada”, se abre un menú y pulsamos en el icono de eliminar sobre la mercancía que queremos desasignar.

Cuando entregamos una mercancía, se desasigna del vehículo automáticamente.

The screenshot shows a table with one row of cargo data for vehicle 1234AAA. The row contains the number '1' in the first column, '10 KG' in the second column, and a red square button with a white 'X' icon in the third column, used for deleting the entry.

Imagen 41. Manual de usuario desasignar mercancía

Modificar estado

En url/vehículos, seleccionamos un vehículo modificamos el apartado “Estado” y pulsamos en “Guardar cambios”.

El estado debe ser uno válido de los del desplegable que se abre.

Para editar el vehículo, debemos pulsar en el botón de “Guardar cambios”.

The screenshot shows a form titled 'Vehículo: 1234AAA' with a close button (X) in the top right corner. Below the title, there are two dropdown menus: 'Añadir mercancía:' and 'Estado:'. The 'Estado:' dropdown is currently set to 'Cargando mercancía'. At the bottom of the form, there are three buttons: 'Eliminar' (red), 'Mercancía asignada' (grey), and 'Guardar cambios' (green).

Imagen 42. Manual de usuario modificar estado

6.3.6. Mercancía

Añadir mercancía

En url/mercancia, en el apartado superior pulsamos en “Añadir mercancía”, se abre un menú, rellenamos los datos y pulsamos en “Añadir”.

El peso y el estado son obligatorios, si no rellenamos alguno la mercancía no se añadirá.

El peso debe ser mayor de 0, de lo contrario no se añadirá, también debe ser de un tipo válido de los del desplegable que se abre.

El estado debe ser uno válido de los del desplegable que se abre.

Podemos añadir descripción o no, no es obligatoria.

Podemos asignar la mercancía a un cliente ya existente, en caso de hacerlo así, debe ser un cliente válido de los del desplegable que se abre.

Podemos añadir un nuevo cliente desde este mismo menú, todos los campos serán obligatorios, si no rellenamos alguno, no se insertará ni la mercancía ni el cliente.

El DNI debe llevar un formato correcto, un ejemplo válido es 37147813F.

El teléfono debe llevar un formato correcto, un ejemplo válido es 645543765.

Para añadir el trabajador, debemos pulsar en el botón de “Añadir”.

Nueva mercancía X

Mercancía

Peso: KG

Estado:

Descripción:

Cliente existente

DNI:

Añadir cliente

DNI:

Nombre:

Apellidos:

Teléfono:

Correo:

Añadir

Imagen 43. Manual de usuario añadir mercancía

Editar mercancía

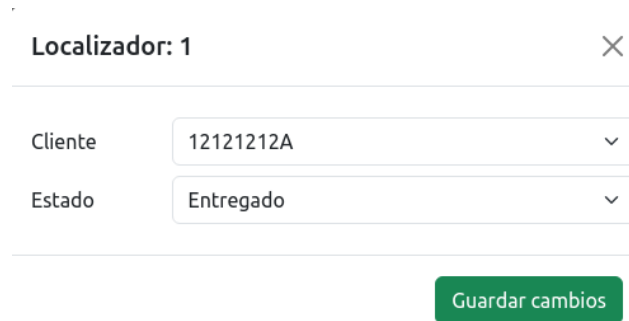
En url/mercancia, seleccionamos una mercancía, se abre un menú, modificamos los datos y pulsamos en “Guardar cambios”.

Si no modificamos nada nuevo saltará un aviso de que no hemos seleccionado nada nuevo para editar.

Si editamos el cliente, debe ser uno válido de los del desplegable que se abre.

Si editamos el estado, debe ser uno válido de los del desplegable que se abre.

Para editar la mercancía, debemos pulsar en el botón de “Guardar cambios”.



Localizador: 1

Cliente: 12121212A

Estado: Entregado

Guardar cambios

Imagen 44. Manual de usuario editar mercancía

Consultar mercancía entregada

En url/mercanciaEntregada podemos consultar la mercancía ya entregada



Gestión Interna > Mercancía > Mercancía entregada			
Mercancía entregada			
Localizador: 17	Descripción: das	Cliente: 11111111A	Peso: 25 KG
Localizador: 19	Descripción: asd	Cliente: 11111111A	Peso: 10 KG

Imagen 45. Manual de usuario consultar mercancía entregada

Eliminar mercancía entregada

En url/mercanciaEntregada podemos pulsar en el icono de eliminar para eliminar el registro de esa mercancía ya entregada.



Gestión Interna > Mercancía > Mercancía entregada			
Mercancía entregada			
Localizador: 17	Descripción: das	Cliente: 11111111A	Peso: 25 KG
Localizador: 19	Descripción: asd	Cliente: 11111111A	Peso: 10 KG

Imagen 46. Manual de usuario consultar mercancía entregada

Localizar mercancía

En url introducimos el localizador y nos redirige a una página con sus datos.

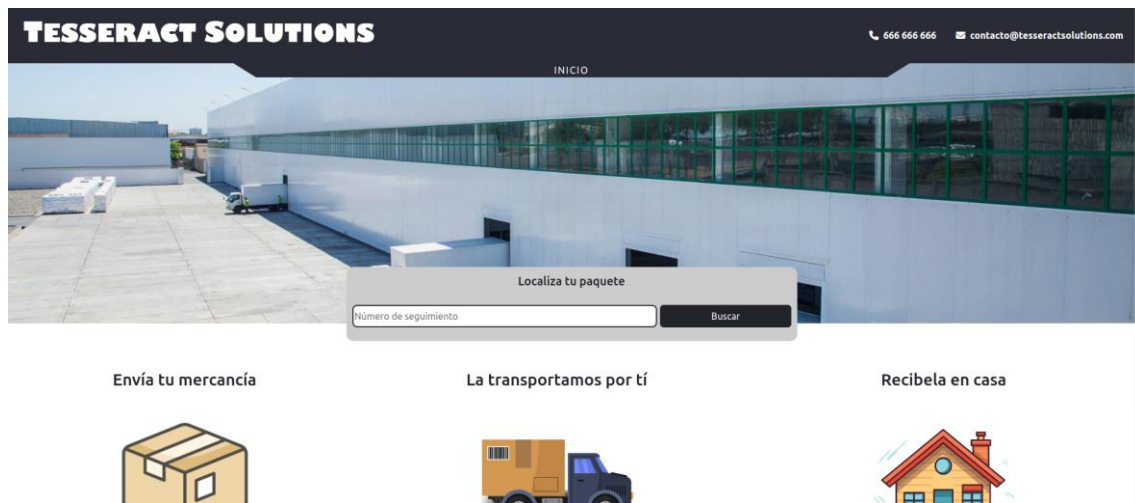


Imagen 47. Manual de usuario localizar mercancía

Nos redirigirá a una nueva página en la que nos muestra los datos de la mercancía consultada.

Si la mercancía consultada no existe, nos mostrará que no existe ninguna mercancía con ese localizador.



Imagen 48. Manual de usuario localizar mercancía

6.3.7. Trabajadores

Añadir trabajador

En url/trabajadores, en el apartado superior pulsamos en “Añadir trabajador”, se abre un menú, rellenamos los datos y pulsamos en “Añadir”.

Todos los campos son obligatorios, si no rellenamos alguno no se añadirá el trabajador.

El DNI debe llevar un formato correcto, un ejemplo válido es 37147813F.

El teléfono debe llevar un formato correcto, un ejemplo válido es 645543765.

El cargo debe ser uno válido de los del desplegable que se abre.

Si pulsamos en Crear cuenta, se creará junto al empleado una cuenta web para poder iniciar sesión.

Para añadir el trabajador, debemos pulsar en el botón de “Añadir”.

Nuevo trabajador

DNI:

Nombre:

Apellidos:

Teléfono:

Correo:

Cargo:

Fecha nacimiento:

Fecha inicio empresa:

Crear cuenta: ☐

Añadir

Imagen 49. Manual de usuario añadir trabajador

Eliminar trabajador

En url/trabajadores, sobre el trabajador deseado pulsamos en “Eliminar trabajador”.

Si el trabajador tenía una cuenta web creada, se eliminará también.

Gestión Interna > Trabajadores

Trabajadores

Cuentas de usuario Añadir trabajador

DNI	Nombre	Apellidos	Teléfono	Correo	Fecha nacimiento	Fecha inicio	Cargo	
12345678C	Prueba3	Prueba3	666666666	prueba@gmail.com	01/01/2020	01/01/2020	Rr.hh	Eliminar trabajador Editar

Imagen 50. Manual de usuario eliminar trabajador

Añadir cuenta usuario

En url/trabajadores, al añadir un trabajador podemos seleccionar el apartado “Crear cuenta” o en un trabajador ya añadido podemos pulsar en “Crear cuenta”. El usuario y contraseña será su DNI.

Gestión Interna > Trabajadores

Trabajadores

Cuentas de usuario Añadir trabajador

DNI	Nombre	Apellidos	Teléfono	Correo	Fecha nacimiento	Fecha inicio	Cargo	
12345678C	Prueba3	Prueba3	666666666	prueba@gmail.com	01/01/2020	01/01/2020	Rr.hh	Crear cuenta Eliminar trabajador Editar

Imagen 51. Manual de usuario añadir cuenta usuario

Eliminar cuenta usuario

En url/usuarios sobre la cuenta deseada pulsamos en “Eliminar”.

Gestión Interna > Trabajadores > Cuentas de usuarios

Cuentas de usuario

DNI	Nombre	Apellidos	Teléfono	Correo	Fecha nacimiento	Fecha inicio	Cargo	
12345678A	Pruebaaaaaa	pruebaaa	666666666	prueba@prueba.es	17/06/2004	05/01/2024	Transportista	Eliminar

Imagen 46. Manual de usuario eliminar cuenta usuario

Editar configuración trabajador

En url/trabajadores, sobre el trabajador deseado pulsamos en “Editar”, solo podrán hacer uso de esto los administradores y RR.HH., nos lleva al apartado de Configuración explicado en el [punto 6.3.9](#).

Si el usuario que pulsa en editar no tiene privilegios para editar la configuración de otros empleados, le redirigirá a editar su propia configuración.



Imagen 52. Manual de usuario editar configuración trabajador

6.3.8. Clientes

Añadir cliente

En url/clientes, en el apartado superior pulsamos en “Añadir cliente”, se abre un menú, rellenamos los datos y pulsamos en “Añadir”.

Todos los campos son obligatorios, si no rellenamos alguno no se añadirá el trabajador.

El DNI debe llevar un formato correcto, un ejemplo válido es 37147813F.

El teléfono debe llevar un formato correcto, un ejemplo válido es 645543765.

El cargo debe ser uno válido de los del desplegable que se abre.

Para añadir el cliente, debemos pulsar en el botón de “Añadir”.

Imagen 53. Manual de usuario añadir cliente

Eliminar cliente

En url/clientes, sobre el cliente deseado pulsamos en el icono de eliminar.

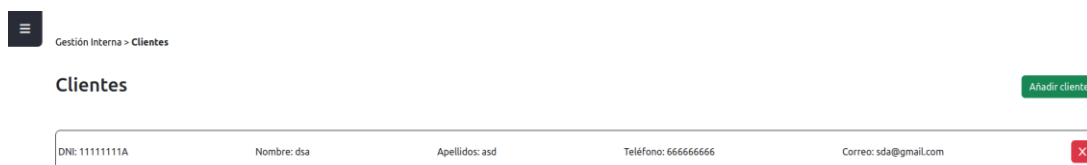


Imagen 54. Manual de usuario eliminar cliente

6.3.9. Configuración

Editar configuración

En url/configuración, modificamos los datos y pulsamos en “Guardar”. Solo los administradores y RR.HH. podrán ingresar a url/configuración/DNI, y modificar los datos de la persona asociada al DNI introducido.

Si no modificamos ningún cambio y pulsamos en guardar, saltará un aviso advirtiéndolo de que no hemos seleccionado nada nuevo.

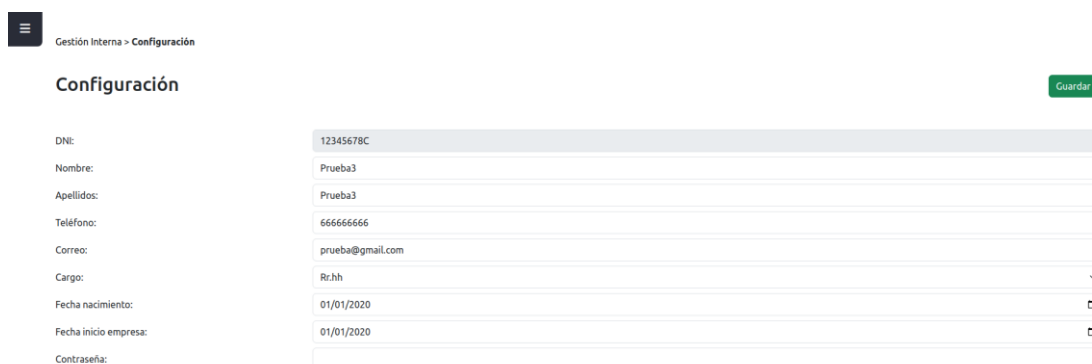
El DNI nunca se podrá modificar.

El teléfono debe llevar un formato correcto, un ejemplo válido es 645543765.

El cargo debe ser uno válido de los del desplegable que se abre, si el usuario no tiene privilegios para editar el cargo, no tendrá disponible esta opción.

El apartado de contraseña solo estará disponible para los empleados que dispongan de una cuenta web.

Para confirmar los cambios debemos pulsar en el botón de “Guardar”.



Gestión Interna > Configuración	
Configuración	
DNI:	12345678C
Nombre:	Prueba3
Apellidos:	Prueba3
Teléfono:	666666666
Correo:	prueba@gmail.com
Cargo:	Rrhh
Fecha nacimiento:	01/01/2020
Fecha inicio empresa:	01/01/2020
Contraseña:	

Imagen 55. Manual de usuario editar configuración

6.3.10. Registro de inicios de sesión

Cada vez que un usuario inicia sesión de forma válida en la aplicación web se añade un nuevo registro a un fichero de logs.

El fichero se llama login.log.

El formato que lleva es el siguiente: DNI;YYYYMMDD;HHMMSS

7. Conclusión

Echando la vista atrás, después de todos estos meses de trabajo, puedo decir con firmeza que se han cubierto todos los requisitos especificados al inicio de esta documentación.

A lo largo del desarrollo, he podido aprender a usar nuevas herramientas que antes no conocía, lo que ha convertido este proyecto en una aventura en vez de un trabajo más.

Durante este periodo, he podido ver con mis propios ojos la importancia de probar cosas nuevas y no tenerle miedo al cambio, no todo lo que ya conocemos es lo único que existe, cuanto más he aprendido, más me he dado cuenta de que me queda mucho por aprender.

Dejo pendiente para mejorar bastantes cosas de este proyecto, pero para una primera versión he quedado más que satisfecho. Me hubiese gustado implementar que en la base de datos no se eliminen los datos, sino que tuviesen una columna en la que nos indicara si ese registro está activo o no, sobre todo de cara a la legalidad, para no perder esos datos que debemos guardar. También se quedan pendientes muchas funcionalidades de todos los crud que hay a lo largo de la web, al ser una primera versión es difícil tener en cuenta todas las funcionalidades que puede tener una empresa en específico, para esta versión he preferido enfocarlo de forma más general, al igual que ocurre con todos los tipos de estado, de mercancía, los cargos...

Respecto al servidor propio que he creado para alojarlo, me gustaría en un futuro poder desplegarlo usando microservicios, comprar un dominio y un certificado para dotar de seguridad a la web.

Por último, quiero dar las gracias a mis profesores por todo el apoyo que me han dado y por haberme enseñado todo lo necesario para enfrentarme al entorno laboral, sin ellos hoy no estaría escribiendo esto.

8. Bibliografía/Webgrafía

Negocio

- Universidad Francisco de Vitoria. (s.f.). Empresa de logística. <https://www.ufv.es/cetys/blog/empresa-de-logistica/>
- Centro Europeo de Postgrado y Empresa. (s.f.). Empresa de logística: ¿Cómo funciona? <https://www.ceupe.com/blog/empresa-de-logistica-como-funciona.html>
- Euroinnova. (s.f.). ¿Cómo funciona una empresa de logística y transporte? <https://www.euroinnova.edu.es/como-funciona-una-empresa-de-logistica-y-transporte>
- Diario de transporte. (s.f.). ¿Qué es la gestión de logística de la empresa y cómo mejorarla?

PHP

- PHP Group. (s.f.). Manual de PHP. <https://www.php.net/manual/es/index.php>
- DevDocs. (s.f.). PHP: Apache. <https://devdocs.io/php-apache/>
- Stack Overflow. (s.f.). <https://stackoverflow.com/>
- McFarland, D. (2018). *Gran libro de PHP. Creación de páginas web dinámicas* (2ª ed.). Anaya Multimedia.
- Stokes, J. (2015). *Modern PHP: New features and good practices*. O'Reilly Media.

JavaScript

- JavaScript.info. (s.f.). Manual de JavaScript. <https://es.javascript.info/>
- W3Schools. (s.f.). JavaScript tutorial. <https://www.w3schools.com/>
- Haverbeke, M. (2018). *JavaScript elocuente. Una introducción moderna a la programación* (2ª ed.). Marcombo.

AJAX

- Mozilla Developer Network. (s.f.). Fetching data with JavaScript. https://developer.mozilla.org/es/docs/Learn/JavaScript/Client-side_web_APIs/Fetching_data
- LenguajeJS. (s.f.). AJAX. <https://lenguajejs.com/javascript/peticiones-http/ajax/>

MVC

- Dcreations. (s.f.). Curso de patrones de diseño en PHP: Patrón MVC en PHP. <https://dcreations.es/cursos/curso-patrones-de-diseno-en-php-gratis/patron-mvc-en-php>
- Fuentes, U. (2020). Desarrollar un proyecto MVC con PHP puro. <https://dev.to/ulisesafcdev/desarrollar-un-proyecto-mvc-con-php-puro-4akg>
- Udemy. (s.f.). Write PHP like a pro: Build a PHP MVC framework from scratch. <https://www.udemy.com/>

Bootstrap

- Bootstrap. (s.f.). <https://getbootstrap.com/>
- Udemey. (s.f.). Bootstrap 5: El curso completo, práctico y desde cero. <https://www.udemy.com/>

FontAwesome

- FontAwesome. (s.f.). <https://fontawesome.com/>

SweetAlert2

- SweetAlert2. (s.f.). <https://sweetalert2.github.io/>