



# Deusto

Universidad de Deusto  
Deustuko Unibertsitatea

## **Ingeniería Web – Proyecto Web Colaborativo**

**Título:** Proyecto web colaborativo

**Curso:** 2º Grado en Industria Digital (Semestre 2º)

**Materia:** Ingeniería Web

**Estudiantes:** Guillermo Domaica Neila

Josu Elizondo Sáenz de Ugarte

Mikel Cascón Molino

**Grupo:** IW-01

**Profesor:** Jon Vadillo Romero

**Facultad de Ingeniería**

**Universidad de Deusto**

**Vitoria - Gasteiz, mayo de 2023**



## Resumen

Este proyecto tiene el objetivo de digitalizar la gestión de pedidos de la empresa Deustronic Componentes S.L. La razón de querer digitalizar este proceso de la empresa es que quieren evitar los problemas que tienen actualmente debido a la gestión de pedidos a través de Excel. Para ello, han analizado diferentes alternativas y nos han pedido que basándonos en su modelo de excel les hagamos una aplicación web a medida.

La aplicación diseñada cuenta con una página principal la cual nos muestra la gestión de clientes que tenemos registrados en el sistema, desde ahí se puede añadir nuevos clientes, borrarlos o modificarlos además de navegar entre las diferentes ventanas donde se gestionan pedidos, productos o componentes.

El proyecto se ha desarrollado a través de 2 lenguajes de programación, Python (framework de desarrollo web, Django) y JavaScript con el que damos funcionalidades extra que Python no nos permite como por ejemplo, realizar eventos al pulsar un botón.

## Descriptores

- Python(Django)
- Entorno web
- Gestión de empresa
- JavaScript

# Índice

<b>Resumen.....</b>	<b>iii</b>
<b>1. Objetivos del proyecto.....</b>	<b>1</b>
1.1. Tareas principales.....	1
1.1.1. HTML Y CSS.....	1
1.1.2. PYTHON.....	1
1.1.3. JAVASCRIPT.....	2
1.1.4. DOCUMENTACIÓN.....	2
1.2. Planificación temporal.....	2
1.2.1. Mikel Cascón.....	3
1.2.2. Guillermo Domaica.....	3
1.2.3. Josu Elizondo.....	3
<b>2. Requisitos iniciales.....</b>	<b>4</b>
<b>3. Funcionalidades añadidas a los requisitos iniciales.....</b>	<b>4</b>
<b>4. Diseño de la aplicación.....</b>	<b>5</b>
4.1. Modelo de datos.....	5
4.2. Diseño de las interfaces.....	6
4.3. Diagrama de secuencia.....	6
4.4. Diagrama de caso de uso.....	7
4.4.1. Caso de uso cliente.....	7
4.4.2. Caso de uso componente.....	7
4.4.3. Caso de uso producto.....	8
4.4.4. Caso de uso pedido.....	8
4.4.5. Caso de uso detalle pedido.....	9
4.5. Diagrama de navegación.....	9
<b>5. Manual de usuario.....</b>	<b>10</b>
5.1. Página principal (listado de clientes).....	10
5.2. Paginación en listas.....	10
5.3. Funcionalidad detalle.....	11
5.4. Funcionalidad meter nuevos datos (crear).....	11
5.5. Funcionalidad modificar datos.....	12
5.6. Funcionalidad eliminar datos.....	12
5.7. Modo noche/día.....	13
5.8. Validación de campos en formulario.....	13
<b>6. Incidencias.....</b>	<b>14</b>
<b>7. Conclusiones.....</b>	<b>14</b>
<b>8. Bibliografía.....</b>	<b>15</b>

# 1. Objetivos del proyecto

Deustronic Componentes S.L. es una empresa dedicada a fabricar productos electrónicos de los cuales tiene diferentes categorías y modelos que vende por todo el mundo, el objetivo de la aplicación va a ser la gestión total (crear, listar, modificar, borrar) los clientes, los pedidos realizados por cada cliente, los productos que van en cada pedido y los componentes por los que está fabricado cada producto.

## 1.1. Tareas principales

### 1.1.1. HTML Y CSS

- Crear templates HTML:
  - Base (plantilla de la que heredan todas las demás)
  - Templates de crear (cliente, pedido, detalle del pedido, componente, producto).
  - Templates de listar (cliente, pedido, componente, producto).
  - Templates de modificar (cliente, pedido, detalle del pedido, componente, producto).
- Estilos CSS:
  - DarkMode.
  - Estilos\_base.
  - Estilos\_footer.
  - Estilos\_header.
  - Estilos\_main.
  - Media\_queries.

### 1.1.2. PYTHON

- Modelo de base de datos.
- Detalles del modelo pedido.
- Funciones crear, modificar y borrar de cada modelo de la base de datos.
- Implementar paginación de listas.
- Envío email de confirmación al crear un nuevo cliente y registrar un pedido.

### **1.1.3. JAVASCRIPT**

- Pop-up al pulsar el botón de eliminar antes ejecutar la acción en BBDD.
- Alternativa para cambio de visualización (lightMode y darkMode).
- Validación de algunos de los campos de los formularios antes de ser enviados.

### **1.1.4. DOCUMENTACIÓN**

- Resumen.
- Objetivos del proyecto, tareas y planificación temporal.
- Requisitos iniciales.
- Funcionalidades añadidas a los requisitos iniciales.
- Diseño de aplicación, modelo de datos, descripción de interfaces...
- Manual de usuario.
- Incidencias.
- Conclusiones.
- Bibliografía.

## **1.2. Planificación temporal**

Lo primero que hicimos fue reunirnos en clase y ver que reto nos parecía el más adecuado, una vez lo teníamos, planteamos un modelo inicial de BBDD del cual partimos para poder crear los modelos, una vez teníamos los modelos bien definidos creamos las vistas correspondientes para posteriormente crear las templates de HTML donde empezamos a ver de manera más gráfica lo que estábamos creando, al mismo tiempo que íbamos desarrollando algunas funcionalidades básicas de las plantillas, comenzamos a dar estilo mediante CSS.

Cuando conseguimos tener una página funcional, nos centramos en añadir diferentes funcionalidades extras en PYTHON, como por ejemplo la paginación. A su vez otro compañero más enfocado en las funcionalidades de JavaScript desarrollaba la validación de algunos campos en los formularios e implementa el modo oscuro y por último creaba un mensaje de alerta cuando se intenta borrar un registro que hay en BBDD.

Finalmente dedicamos las últimas horas de trabajo en desarrollar el presente escrito en el que se recoge el trabajo realizado y la funcionalidad del proyecto.

### **1.2.1. Mikel Cascón**

- Creación de modelos de base de datos.
- Enlazar formularios con BBDD.
- Implementación función borrar.
- CSS de toda la aplicación.
- HTML (base, cliente, componente).
- Mejora de los formularios.
- Widgets en formularios.
- Envío de emails.
- DOC. Diseño de aplicación.
- DOC. Manual de usuario.

### **1.2.2. Guillermo Domaica**

- Creación de vistas.
- Modelo de BBDD.
- Detalles de la clase producto.
- HTML (pedido, producto).
- Validación de campos con JS.
- Implementación de darkMode con JS.
- Mensaje de aviso al borrar con JS.
- Solución de algunos problemas que surgían.
- DOC. Funcionalidades añadidas a los requisitos iniciales.
- DOC. incidencias.

### **1.2.3. Josu Elizondo**

- Creación de URLs.
- Creación de modelo inicial de formularios.
- Implementación modificación de clientes.
- Creación de algunos botones.
- Implementación de la paginación.
- DOC. Objetivos del proyecto
- DOC. Requisitos iniciales.
- DOC. Conclusiones.
- DOC. Bibliografía.

## 2. Requisitos iniciales

El cliente nos requirió unas funcionalidades básicas iniciales para simplificar su proceso de gestión de pedidos y poder empezar a trabajar evitando problemas:

- Apartado gestión de productos:

En este apartado el cliente nos pedía poder tener controlados los productos que tiene en fábrica, para poder controlarlos quiere dar de alta nuevos productos, visualizar en una tabla los productos que ya están creados y por último poder borrar algunos productos que ya no le interesara tener en el sistema.

- Apartado gestión de pedidos:

En este apartado el cliente nos pedía poder tener controlados los pedidos que entran en el sistema, para poder controlarlos quiere tener la posibilidad de dar de alta nuevos pedidos, visualizar en una tabla los pedidos que ya están creados además de una opción de detalle en donde se pueda ver qué productos componen los pedidos y por último poder modificar algún dato del pedido.

## 3. Funcionalidades añadidas a los requisitos iniciales

Además de las funcionalidades básicas que nos habían pedido añadimos algunas más para que la utilidad de la página sea mayor, las funcionalidades añadidas son:

- **Modificar un producto ya creado:** Esta opción te da la posibilidad de poder editar los datos iniciales de un producto si surge algún cambio.
- **Eliminar un pedido:** Esta función la hemos añadido al listado de pedidos, dado que es imprescindible poder borrar pedidos si los clientes cancelan sus pedidos.
- **Gestión completa de componentes y clientes:** Vimos necesario implementar estas funciones ya que resultaba un poco complicado manejar la aplicación sin disponer de ellas en la plataforma web.
- **Paginación de registros listados:** Cuando registrábamos varios datos nos quedaba una tabla demasiado grande y poco manejable, con lo que esta funcionalidad nos brinda la facilidad de organizar mejor los registros guardados.
- **Envío de emails:** A modo de complementar la experiencia utilizando el entorno web, el envío de email al registrar un cliente o emitir un pedido, garantizan al usuario o empresa que la acción ha sido completada satisfactoriamente.



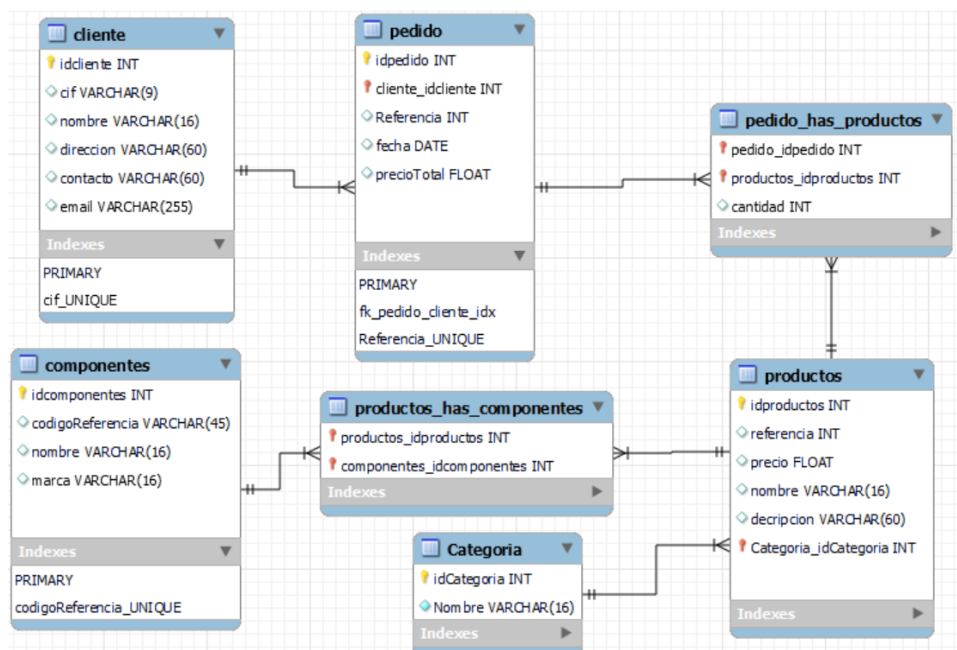
- **Hovers con CSS:** Hacen que la navegación por parte del usuario sea más intuitiva y amigable. “user-friendly”
- **Mensajes de alerta:** Como el borrado de datos es una parte crítica ya; debido a que no se pueden recuperar, vimos necesario la implementación de una confirmación antes de eliminar el registro de la base de datos.
- **Validación de campos:** Para evitar que el formulario añada datos erróneos a la base de datos, estos se comprueban antes avisando si ocurre algún error.
- **DarkMode:** Modo de visualización oscura del entorno web para evitar la fatiga ocular.
- **Placeholder:** Ayuda al usuario a rellenar el formulario con el formato correcto.

## 4. Diseño de la aplicación

Aquí vamos a poder entender de manera más gráfica cómo está diseñada la aplicación viendo de manera sencilla cómo está estructurado el diseño de la interfaz, diferentes diagramas para entender el flujo de información y también el esquema entidad-relación a partir del cual hemos creado la base de datos.

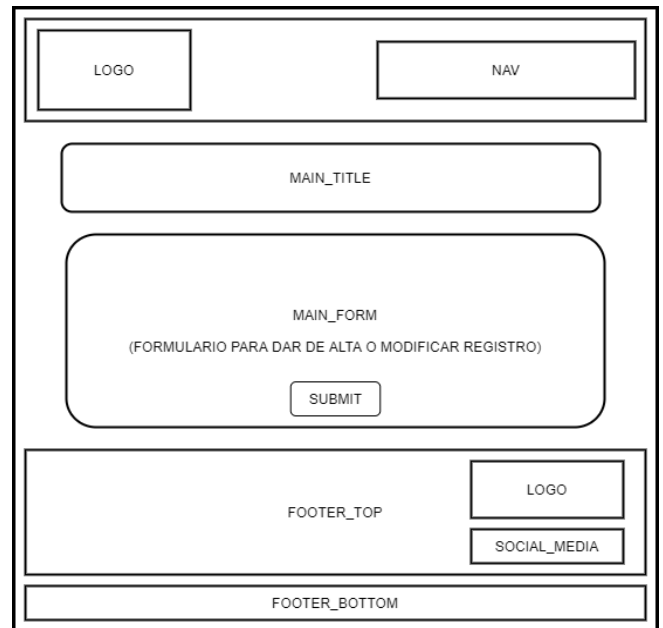
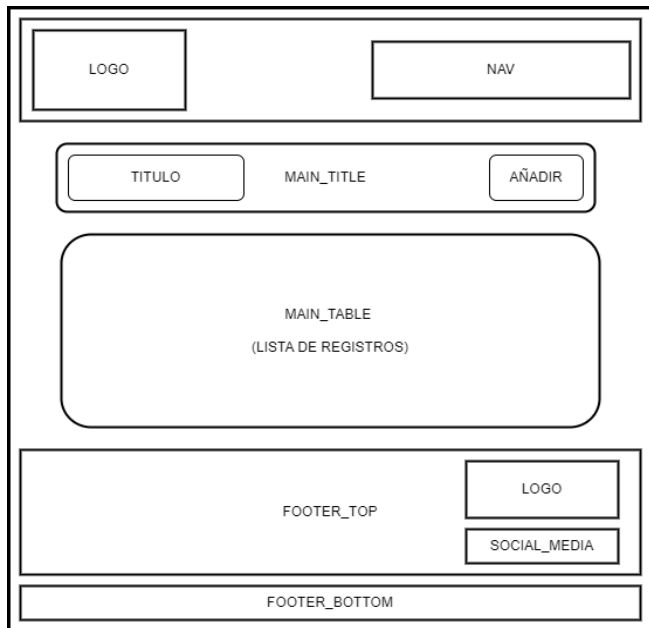
### 4.1. Modelo de datos

El diagrama de entidad-relación nos permite ver de manera rápida y sencilla cuáles son las clases que componen la base de datos del proyecto, estas son, cliente, pedido, componente, producto, categoría y productos que componen un pedido. Además, se puede observar fácilmente si la relación entre cada clase es de 1:n o de n:m con lo que nos queda claro de manera rápida cómo va a funcionar el sistema.



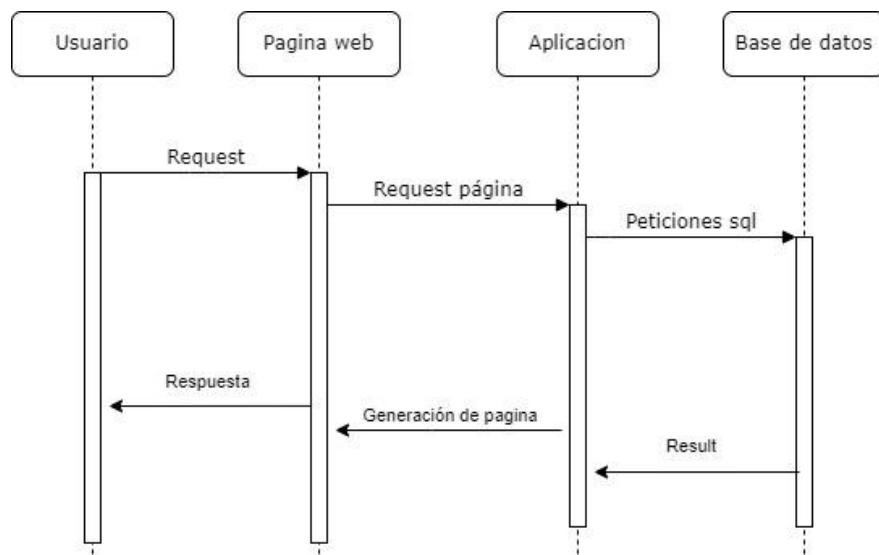
## 4.2. Diseño de las interfaces

A continuación, se muestra el diseño de las dos interfaces del entorno web, en la imagen de la izquierda se muestran los registros listados de los clientes, pedidos, productos o componentes y en el caso de la imagen de la derecha muestra el diseño de interfaz cuando se añade, modifica, crea o asocia un cliente, pedido, producto o componente mediante el uso de un formulario. Si bien es verdad que a la hora de crear un cliente no tenemos los mismos campos que a la hora de generar un pedido, los diseños mantienen el mismo formato y solo se diferencian estos dos.



## 4.3. Diagrama de secuencia

Aquí podemos observar cómo es el flujo de la información dentro de la aplicación. Cada vez que el usuario realice alguna acción estará haciendo una petición a la página web la cual mediante una URL se enlaza con las funciones del servidor. Este realizará sentencias SQL para obtener los datos necesarios de la BBDD en caso de que sea necesario, luego se genera un HTML que la página web mostrará al usuario.

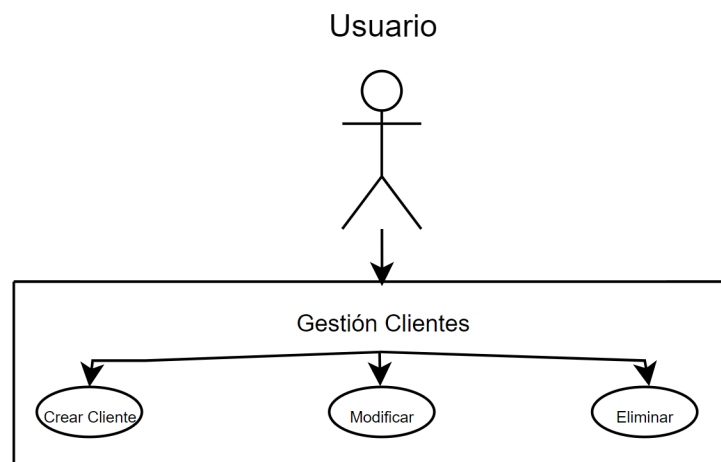


## 4.4. Diagrama de caso de uso

Estos diagramas nos permiten ver de forma gráfica las funcionalidades que hay en las diferentes ventanas del proyecto, para que la aplicación sea más intuitiva las funcionalidades son parecidas en todas las ventanas.

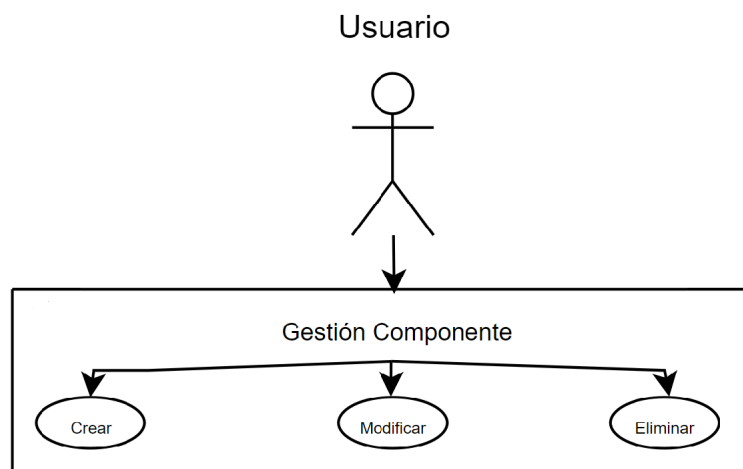
### 4.4.1. Caso de uso cliente

En el caso de uso de los clientes podremos ver un listado con los diferentes clientes que tenemos guardados, crear un nuevo cliente, modificar un cliente ya guardado o eliminar un cliente.



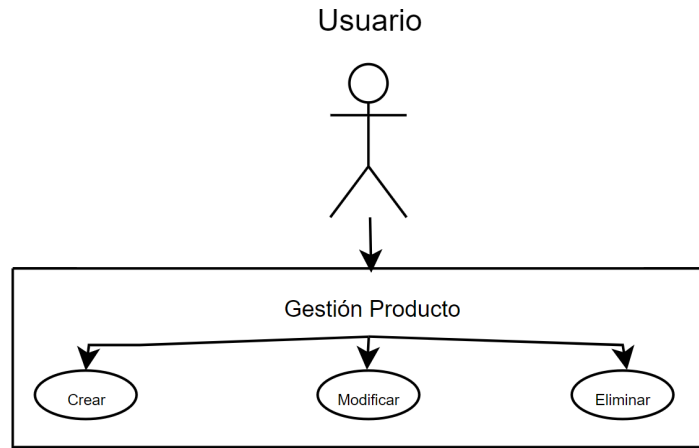
### 4.4.2. Caso de uso componente

En el caso de uso de los componentes podremos ver un listado con los diferentes componentes que tenemos guardados, crear un nuevo componente, modificar un componente ya guardado o eliminar un componente.



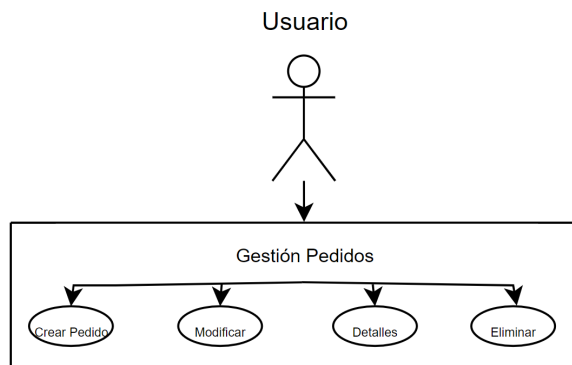
#### 4.4.3. Caso de uso producto

En el caso de uso de los productos podremos ver un listado con los diferentes productos que tenemos guardados, crear un nuevo producto, modificar un producto ya guardado o eliminar un producto.



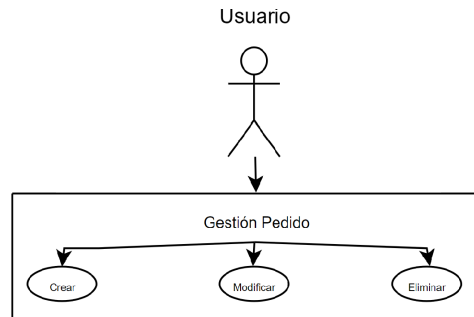
#### 4.4.4. Caso de uso pedido

En el caso de uso de los pedidos podremos ver un listado con los diferentes pedidos que tenemos guardados, ir a su detalle donde veremos de qué artículos se compone cada pedido, crear un nuevo pedido, modificar un pedido ya guardado o eliminar un pedido.



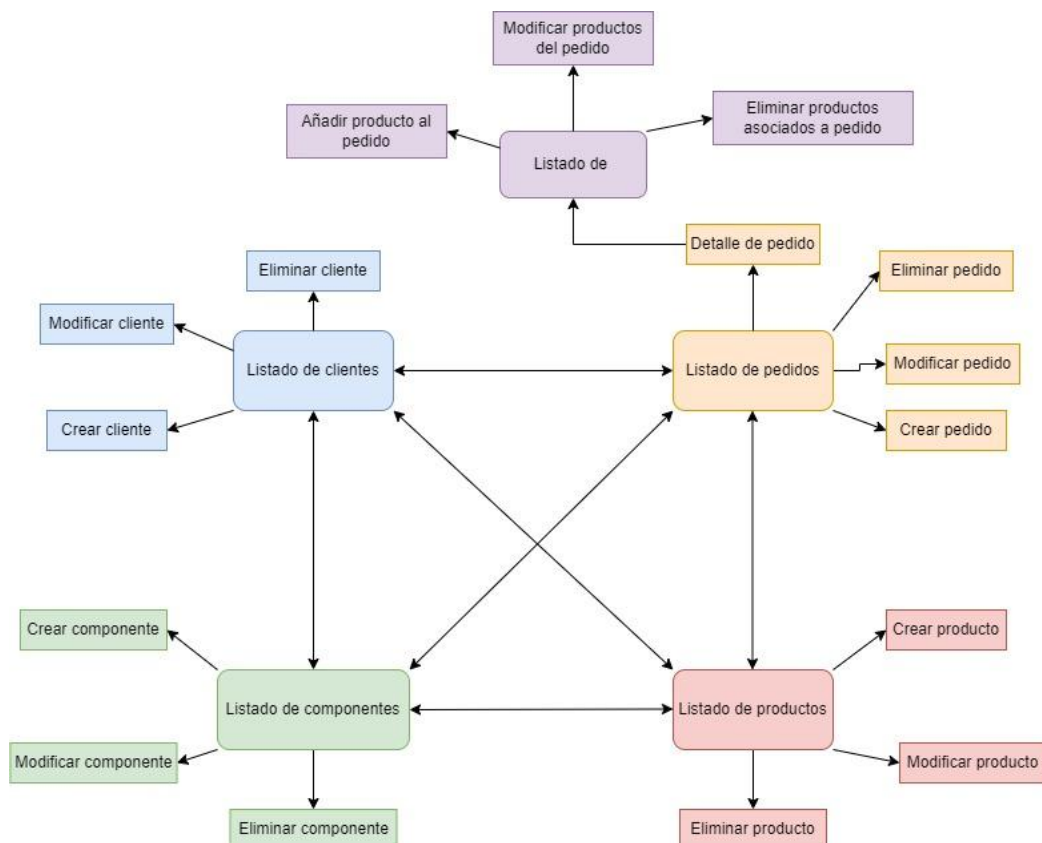
#### 4.4.5. Caso de uso detalle pedido

En el caso de uso del detalle de pedido podremos ver un listado con los diferentes productos que tiene asociados un pedido, añadir nuevos productos al pedido, modificar un producto perteneciente al pedido ya guardado o eliminar un producto del pedido.



### 4.5. Diagrama de navegación

Este diagrama muestra cómo se navega dentro de las diferentes ventanas del proyecto. Una vez estamos en el listado de clientes (página principal) podemos navegar entre todos los listados pedidos, componentes, productos y dentro de cada uno de ellos pinchando en los botones podemos crear, modificar o eliminar registros en BBDD. Si estamos listando pedidos también tenemos la posibilidad de ver el detalle de estos donde se mostrará que productos tiene asociado cada pedido.



## 5. Manual de usuario

Los ejemplos del manual van a ser sacados de la página principal de nuestro proyecto ya que la navegación por las demás páginas es exactamente igual.

### 5.1. Página principal (listado de clientes)

Cuando entramos en la página web se nos abre la ventana de listado de clientes que tiene guardados la empresa. En la cabecera podemos ver el menú de navegación hacia las demás ventanas: pedidos, productos, componentes. En el centro está el contenido principal con la tabla de clientes y las diferentes opciones que se pueden ejecutar y en el pie de página tenemos algunos enlaces de interés.

**DEUSTRONIC COMPONENTES**

Cientes Pedidos Productos Componentes

**LISTADO DE CLIENTES** **ANADIR CLIENTE**

CIF	NOMBRE	DIRECCION	TELEFONO	EMAIL	OPCIONES
A12345678	Mikel	Calle Postas 4	568542565	mikel@deusto.es	<a href="#">MODIFICAR</a> <a href="#">BORRAR</a>
E52465845	Michelin	Calle portal de foronda	685412658	michelin@gmail.com	<a href="#">MODIFICAR</a> <a href="#">BORRAR</a>
D54253658	Mercedes	Poligono Jundiz	658956254	mercedes@gmail.com	<a href="#">MODIFICAR</a> <a href="#">BORRAR</a>
D42563528	Joni	Calle San martin	654852569	jondizcendeo.je@gmail.com	<a href="#">MODIFICAR</a> <a href="#">BORRAR</a>

1 2 3

**Menu**  
Cientes  
Pedidos  
Productos  
Componentes

**I+D+i**  
Proyectos I+D+i  
Programa de Innovación  
Innovación Tecnológica

**Sostenibilidad**  
Compromiso  
Responsabilidad Social  
Sistema de Gestión

**Contacto**  
Contacto  
Trabaja con nosotros

**DEUSTRONIC COMPONENTES**

[Instagram](#) [Twitter](#) [Facebook](#) [YouTube](#) [LinkedIn](#)

[Aviso legal](#) [Política de privacidad](#) [Política de cookies](#)

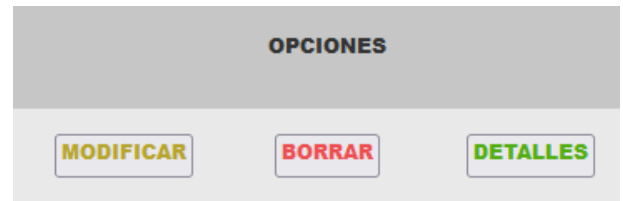
### 5.2. Paginación en listas

Si tenemos más de 4 clientes creados los siguientes que creemos se irán acumulando en diferentes páginas las cuales podemos ver debajo de la tabla y podemos acceder pinchando sobre el número de página deseado, esto es para que no quede una tabla excesivamente grande en cada página. En el siguiente ejemplo tendríamos creados por lo menos 17 clientes repartidos de 4 en 4 en las 4 páginas que vemos que hay.

1 2 3 4

### 5.3. Funcionalidad detalle

La opción de detalle solo la tenemos disponible cuando estamos listando todos los pedidos, si pulsamos sobre el botón verde “detalles” este nos llevará a ver la información que tenemos guardada del pedido donde podremos ver que productos tiene asociados y la cantidad de cada producto.



<b>DETALLES DEL PEDIDO: 3</b> REFERENCIA: 1 FECHA: Feb. 25, 2023 CLIENTE: A12345678 - Mikel PRECIO TOTAL: 50.0					ASOCIAR PRODUCTO
REF. PRODUCTO	NOMBRE PRODUCTO	CANTIDAD	PRECIO UNITARIO	OPCIONES	
4824	Lavadora	50	50.0	MODIFICAR	BORRAR
94535	Cable luz	7	20.0	MODIFICAR	BORRAR

### 5.4. Funcionalidad meter nuevos datos (crear)

Pinchando sobre el botón de añadir cliente nos dirigirá al formulario de creación de un cliente, en cada campo del formulario veremos escrito por defecto lo que hay que poner para rellenar el formulario de manera correcta, una vez relleno podemos darle a enviar formulario para almacenar los datos en BBDD, si por cualquier razón queremos volver a la lista sin enviar datos debemos seleccionar en el menú de navegación la página a la que queramos ir.



#### Nuevo cliente

<b>CIF:</b> <input type="text" value="Ej.: A12147193"/>	<b>Nombre de la empresa:</b> <input type="text" value="Ej.: Forjas Alavesas"/>	<b>Dirección:</b> <input type="text" value="Ej.: Calle Postas 4"/>
<b>Telefono:</b> <input type="text" value="Ej.: 660862421"/>	<b>Email:</b> <input type="text" value="Ej.: elnano33@gmail.com"/>	
<input type="button" value="Nuevo cliente"/>		

## 5.5. Funcionalidad modificar datos

Si ya tenemos algún registro almacenado en la BBDD los campos de este pueden ser modificados pinchando sobre el botón modificar, cuando pulsemos, este nos llevará al formulario para modificar campos el cual por defecto saldrá relleno con los datos que se tengan almacenados, en esa ventana se pueden cambiar los datos deseados y debemos pulsar el botón de confirmación para que se realicen las modificaciones correspondientes en la BBDD.

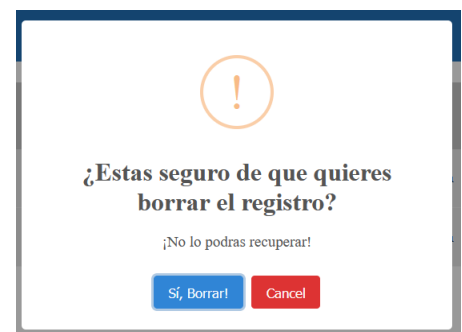
CIF	NOMBRE	DIRECCION	TELEFONO	EMAIL	OPCIONES
72857357G	CASCON SA	Calle Jaen	660862423	mikelcasmon@gmail.com	<b>MODIFICAR</b> <b>BORRAR</b>

### Modificar cliente

CIF:	Nombre de la empresa:	Dirección:
<input type="text" value="72857357G"/>	<input type="text" value="CASCON SA"/>	<input type="text" value="Calle Jaen"/>
Telefono:		Email:
<input type="text" value="660862423"/>		<input type="text" value="mikelcasmon@gmail.com"/>
<input type="button" value="Confirmar cambios de cliente"/>		

## 5.6. Funcionalidad eliminar datos

Cuando le demos al botón de borrar nos va a aparecer un aviso de que estamos intentando borrar un dato de la base de datos y como este no lo podremos recuperar nos pide una confirmación, esto se hace para asegurarse de que no hemos pulsado el botón sin querer, por lo tanto, debemos darle a “sí, borrar” si realmente deseamos eliminarlo o a “cancelar” para paralizar la acción de borrar.

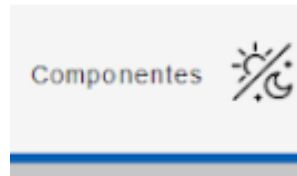


CIF	NOMBRE	DIRECCION	TELEFONO	EMAIL	OPCIONES
72857357G	CASCON SA	Calle Jaen	660862423	mikelcasmon@gmail.com	<b>MODIFICAR</b> <b>BORRAR</b>



## 5.7. Modo noche/día

La página web tiene la opción de poner el fondo en un tono más oscuro, lo llamado modo noche, esto se activa y se desactiva pulsando en la imagen de un sol y una luna localizada arriba a la derecha al lado del menú de navegación. Una vez esté activado el modo no se desactiva hasta que no volvamos a pulsar sobre la imagen.



## 5.8. Validación de campos en formulario

Con el fin de evitar que los datos sean introducidos de manera errónea en la base de datos se hace una comprobación de algunos de los campos antes de almacenar el registro paralizando el guardado de datos si hubiera algún error.

A screenshot of a web form titled 'Nuevo cliente'. The form contains several input fields: 'CIF:' with the value 'd456732', 'Telefono:' with the value '945326789', 'Dirección:', 'Correo electrónico:', and 'Email:'. A modal dialog box is displayed in the center, featuring a red 'X' icon, the text 'Oops...', and the message 'El primer digito del CIF debe de ser una letra mayuscula.' (The first digit of the CIF must be an uppercase letter). An 'OK' button is at the bottom of the dialog.

## 6. Incidencias

En la primera entrega del proyecto no conseguimos hacer que se modificaran los registros de la base de datos, sino que cuando queríamos cambiar algún campo directamente se nos creaba un nuevo registro. Tras inspeccionar el código vimos que al darle a confirmar cambios se hacía una llamada a una URL que no era necesaria y tras quitar esa URL la función modificar ya funcionaba adecuadamente.

Error en el diseño inicial de la base de datos de tal manera que cuando empezamos a desarrollar la aplicación nos encontramos con problemas derivados de una mala relación entre las tablas de la base de datos. Tras una tutoría redefinimos los modelos de la base de datos con lo que solucionamos este primer problema que nos surgió.

Mikel tuvo un problema a la hora de mandar los correos tras registrar un usuario o emitir un pedido, debido a un error de autenticación SMTP, por parte del acceso al servidor de Google el cual no le permitía acceder y enviar los correos. Finalmente, y tras buscar información al respecto, tuvo que hacer la confirmación en dos pasos de Google, solicitar una contraseña de aplicación (la que está definida en settings) e identificar el puerto correspondiente.

Guillermo se encontró dificultades a la hora de añadir "id" a los campos del formulario debido a la manera en la que estaban creados los formularios, tras buscar en varias páginas descubrió que mediante el uso de widgets se le pueden dar atributos a los campos de los formularios.

## 7. Conclusiones

Tras unos meses de trabajo por fin vemos el resultado de todo el esfuerzo dedicado al desarrollo de este proyecto cumpliendo con todos los objetivos que se nos habían planteado inicialmente. Estamos muy contentos con el resultado obtenido y con la gran cantidad de conocimientos que hemos adquirido gracias a las intensas búsquedas de información para conseguir que el código funcione de manera adecuada.

Desde el inicio vimos que hacer la aplicación solo con los requisitos básicos que nos habían pedido iba a ser algo poco útil, por ello decidimos ampliar la funcionalidad metiendo más requisitos que nos parecían imprescindibles, por ello hemos podido conseguir una aplicación bastante intuitiva y funcional que ayudar a resolver los problemas que tenía la empresa con sus datos guardados en hojas de Excel.

La implementación de JavaScript en el código nos ha resultado muy útil ya que te permite realizar acciones que, si no es muy difícil de hacer, como por ejemplo, estar a la espera de que se pulse algún botón para que se realice la función programada. Nos hubiera gustado meter muchas más acciones con JavaScript que harían que la aplicación sea más funcional aún pero se nos ha echado el tiempo encima.

## 8. Bibliografía

- <https://stackoverflow.com/>
- <https://github.com/jvadillo/curso-django-paso-a-paso>
- <https://www.youtube.com/>
- <https://programadorwebvalencia.com/>
- <https://programacion.net/>
- <https://carontestudio.com/blog/anadir-tema-oscuro-css-javascript/>
- <https://www.techiedelight.com/>
- <https://sweetalert2.github.io/>
- <https://carontestudio.com/blog/anadir-tema-oscuro-css-javascript/>
- <https://www.scrapaperapi.com/>
- <https://developer.mozilla.org/>