

**Ingeniería Web**

**Documentación proyecto web colaborativo**

**Grupo 6:**

**Jon Ander, Gaizka y Andrea**

**Vitoria-Gasteiz - 28/05/2024**

Dibujo en blanco y negro

Descripción generada automáticamente con confianza baja

Índice

[INTRODUCCIÓN 1](#_Toc167780330)

[OBJETIVOS DEL PROYECTO 1](#_Toc167780331)

[Resumen para la dirección 1](#_Toc167780332)

[Tareas principales 1](#_Toc167780333)

[Planificación temporal 1](#_Toc167780334)

[ESPECIFICACIONES DE REQUISITOS DEL SISTEMA 2](#_Toc167780335)

[Descripción general 2](#_Toc167780336)

[Catálogo de requisitos 2](#_Toc167780337)

[Modelo lógico de datos 3](#_Toc167780338)

[Representación entidad relación 3](#_Toc167780339)

[Transformación a relacional 3](#_Toc167780340)

[Normalizar Campos 4](#_Toc167780341)

[MANUAL DE USUARIO 5](#_Toc167780342)

[INCIDENCIAS DEL PROYECTO Y CONCLUSIONES 9](#_Toc167780343)

[Bibliografía 10](#_Toc167780344)

Dibujo en blanco y negro

Descripción generada automáticamente con confianza baja

Dibujo en blanco y negro

Descripción generada automáticamente con confianza baja

ÍNDICE DE ILUSTRACIONES

[Ilustración 1 - Diagrama E/R 3](#_Toc167756398)

[Ilustración 2 – Pantalla de registro 5](#_Toc167756399)

[Ilustración 3 – Pantalla de inicio de sesión 5](#_Toc167756400)

[Ilustración 4 - Página principal 6](#_Toc167756401)

[Ilustración 5 - Creación de producto 6](#_Toc167756402)

[Ilustración 6 - Pantalla de soporte 7](#_Toc167756403)

[Ilustración 7 - Validación de campos 7](#_Toc167756404)

ÍNDICE de tablas

[Tabla 1 - Producto 3](#_Toc167756392)

[Tabla 2 – Componente 3](#_Toc167756393)

[Tabla 3 – ProductoComponente 3](#_Toc167756394)

[Tabla 4 - Cliente 4](#_Toc167756395)

[Tabla 5 – Pedido 4](#_Toc167756396)

[Tabla 6 – ProductoPedido 4](#_Toc167756397)

Dibujo en blanco y negro

Descripción generada automáticamente con confianza baja

# 

# INTRODUCCIÓN

El proyecto desarrollado es una aplicación web diseñada para la gestión eficiente de productos, clientes, pedidos y componentes. Ha sido desarrollado utilizando Django, un framework de desarrollo web de alto nivel basado en Python.

# OBJETIVOS DEL PROYECTO

## Resumen para la dirección

La aplicación desarrollada para Deustronic S. L. es una plataforma web diseñada para optimizar la gestión de inventarios, clientes, pedidos y componentes dentro de una empresa. Utilizando el framework Django, se busca crear una herramienta intuitiva y eficiente que permita a los usuarios administrar estos elementos de manera centralizada y efectiva. La aplicación proporcionará funcionalidades completas para la creación, modificación, visualización y eliminación de productos, clientes y pedidos, además de facilitar la asignación de componentes a productos y productos a pedidos. Con esta solución, se espera mejorar significativamente la eficiencia operativa, reducir errores y aumentar la satisfacción del cliente.

## Tareas principales

A la hora de realizar el proyecto, las tareas principales han sido las siguientes:

1. Programación en Python
2. Renderización en HTML
3. Diseño y aplicación de estilos en css
4. Diseño de la BBDD

## Planificación temporal

(qué tareas, cómo nos hemos organizado, reparto)

* Andrea
* Gaizka
* Jon Ander

# ESPECIFICACIONES DE REQUISITOS DEL SISTEMA

## Descripción general

### Catálogo de requisitos

El sistema debe cumplir con las siguientes funciones:

1. Gestión de productos:
   1. Crear nuevos productos, ingresando detalles como nombre, descripción, precio y stock.
   2. Modificar los detalles de productos existentes.
   3. Eliminar productos del inventario.
   4. Visualizar una lista de todos los productos, con opciones de búsqueda y filtrado.
   5. Detalle de producto con información completa.
2. Gestión de clientes:
   1. Registrar nuevos clientes con información como nombre, dirección, correo electrónico y teléfono.
   2. Modificar la información de clientes existentes.
   3. Eliminar clientes del sistema.
   4. Visualizar una lista de todos los clientes, con opciones de búsqueda y filtrado.
   5. Detalle de cliente con información completa.
3. Gestión de pedidos:
   1. Modificar la información de clientes existentes.
   2. Eliminar clientes del sistema.
   3. Visualizar una lista de todos los clientes, con opciones de búsqueda y filtrado.
   4. Detalle de cliente con información completa.
4. Gestión de componentes:
   1. Crear nuevos componentes para productos.
   2. Asignar componentes a productos específicos.
   3. Modificar y eliminar componentes.
   4. Visualizar una lista de todos los componentes, con opciones de búsqueda y filtrado.
5. Interfaz de usuario:
   1. Proveer una interfaz intuitiva y fácil de usar para todas las funcionalidades mencionadas.
   2. Utilizar un diseño responsivo que funcione bien en dispositivos móviles y de escritorio.
   3. Incluir formularios claros y validaciones para la entrada de datos.
6. Seguridad y Soporte:
   1. Autenticación de usuarios para proteger el acceso a la aplicación.
   2. Gestión de mensajes de soporte recibidos a través de e-mail.

¿¿¿Inconvenientes actuales del cliente???

¿¿¿Necesidades de la nueva aplicación???

## Modelo lógico de datos

### Diagrama Descripción generada automáticamente con confianza mediaRepresentación entidad relación

Ilustración 1 - Diagrama E/R

**Precio total es un atributo derivado ya que se obtiene de la multiplicación de cantidad de producto por el precio.**

### Transformación a relacional

|  |  |  |
| --- | --- | --- |
| Campo | Tipo | Descripción |
| id (PK) | Integer | Identificador único del producto |
| nombre | String | Nombre del producto |
| categoria | String | Categoría del producto |
| precio | Decimal | Precio del producto |
| cantidad | Integer | Cantidad disponible en inventario |
| descripcion | Text | Descripción del producto |

Tabla 1 - Producto

|  |  |  |
| --- | --- | --- |
| Campo | Tipo | Descripción |
| id (PK) | Integer | Identificador único del componente |
| nombre | String | Nombre del componente |
| marca | String | Marca del componente |

Tabla 2 – Componente

|  |  |  |
| --- | --- | --- |
| Campo | Tipo | Descripción |
| id (PK) | Integer | Identificador único de la relación |
| producto\_id (FK) | Integer | Identificador del producto |
| componente\_id (FK) | Integer | Identificador del componente |
| cantidad | Integer | Cantidad del componente en el producto |

Tabla 3 – ProductoComponente

### Normalizar Campos

|  |  |  |
| --- | --- | --- |
| Campo | Tipo | Descripción |
| id (PK) | Integer | Identificador único del cliente |
| cif | String | Identificador fiscal del cliente |
| nombre\_empresa | String | Nombre de la empresa del cliente |
| direccion | String | Dirección física del cliente |
| telefono | String | Número de teléfono del cliente |
| email | String | Correo electrónico del cliente |

Tabla 4 - Cliente

|  |  |  |
| --- | --- | --- |
| Campo | Tipo | Descripción |
| id (PK) | Integer | Identificador único del pedido |
| cliente\_id (FK) | Integer | Identificador del cliente |
| fecha | Date | Fecha del pedido |
| precio\_total | Decimal | Precio total del pedido |

Tabla 5 – Pedido

|  |  |  |
| --- | --- | --- |
| Campo | Tipo | Descripción |
| id (PK) | Integer | Identificador único de la relación |
| pedido\_id (FK) | Integer | Identificador del pedido |
| producto\_id (FK) | Integer | Identificador del producto |
| cantidad | Integer | Cantidad del producto en el pedido |

Tabla 6 – ProductoPedido

**Siguiendo el esquema entidad relación y tratando de evitar el mayor número de redundancias, se han obtenido las anteriores tablas, especificadas y generadas a través del models.py de la aplicación.**

# MANUAL DE USUARIO

Para poder acceder como usuario con rol de administrador en nuestra aplicación debemos seguir estos pasos:

* (Usuario: deusto, pssw: deusto)

Si queremos registrar un usuario debemos seguir los pasos del formulario de la *Ilustración 3 y cumplir con los requisitos.*

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamenteCaptura de pantalla de un celular

Descripción generada automáticamenteInterfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

Ilustración 3 – Pantalla de registro

Ilustración 2 – Pantalla de inicio de sesión

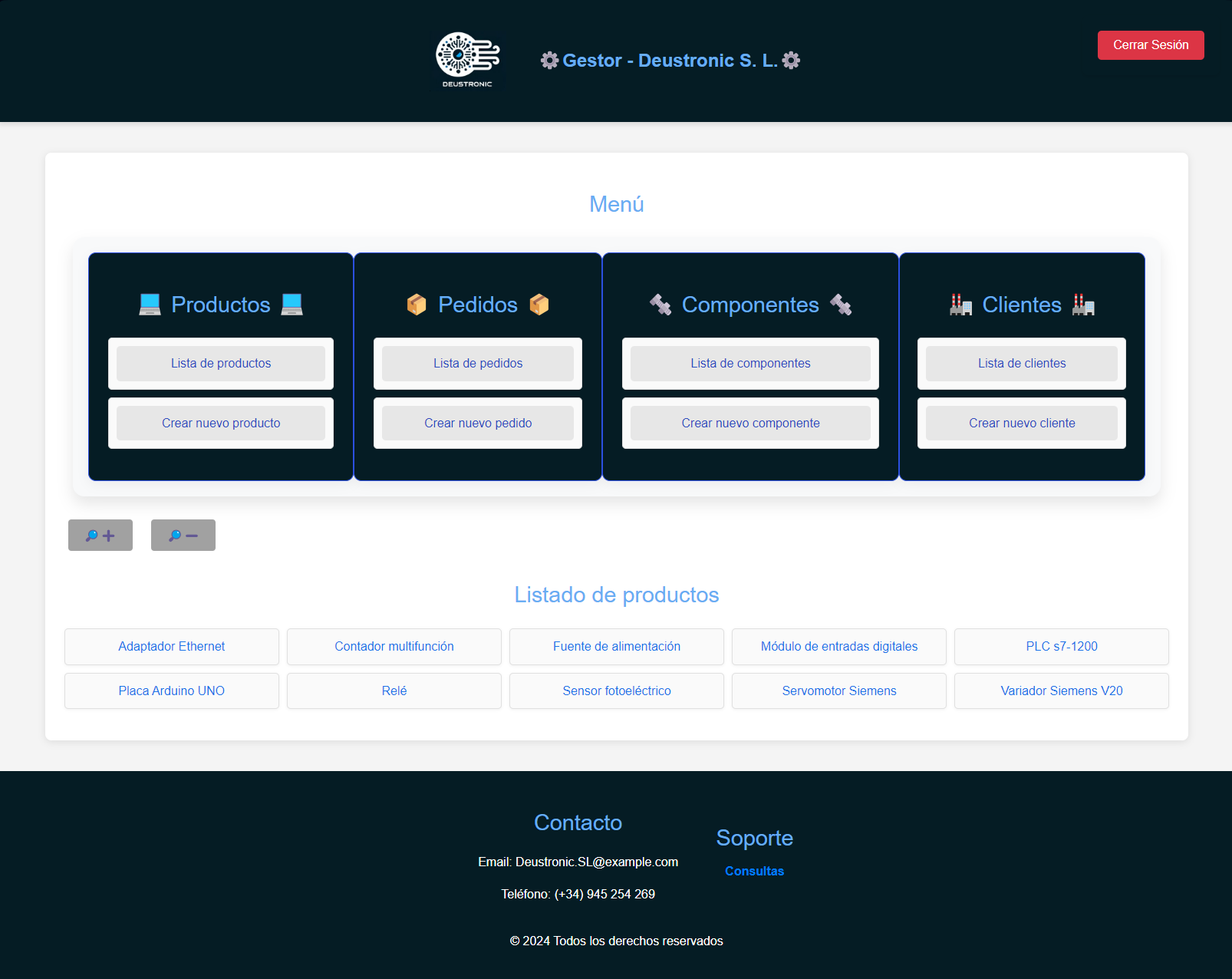


Ilustración 4 - Página principal

En nuestra aplicación web hemos implementado varios listados que siguen un formato y diseño prácticamente similar. A continuación, se describe cómo están estructurados estos listados, utilizando el menú principal como referencia visual.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Ilustración 5 - Creación de producto

Hemos implementado varios formularios de creación que siguen un diseño y formato prácticamente similar.

1. **Encabezado:** Cada formulario de creación tiene un encabezado que indica claramente el propósito del formulario (por ejemplo, "Nuevo producto", "Nuevo pedido").
2. **Campos de Entrada:** Los formularios incluyen campos de entrada claramente etiquetados para capturar la información necesaria. Por ejemplo, en el formulario de creación de producto, los campos pueden incluir nombre del producto, precio, descripción y categoría.
3. **Botón de Envío:** Al final del formulario, se incluye un botón de envío que permite al usuario guardar el nuevo registro. El botón está claramente etiquetado (por ejemplo, "Nuevo producto").

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Ilustración 6 - Pantalla de soporte

En el apartado de soporte, hemos implementado una validación de campos para asegurar que los mensajes enviados no contengan palabras prohibidas. Esto ayuda a mantener un lenguaje

adecuado y profesional en las comunicaciones. El formulario consta de un campo de texto y un botón de envío.

A continuación, se detalla cómo funciona esta validación utilizando el formulario de mensaje de soporte como ejemplo. La pantalla de soporte como se puede ver en la ilustración 6, la pantalla incluye un formulario simple donde el usuario puede escribir su mensaje.

.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ilustración 7 - Validación de campos

En la ilustración 7, se muestra un ejemplo donde el usuario intenta enviar un mensaje que contiene palabras prohibidas ("palabra1" y "palabra2"). Cuando el usuario escribe estas palabras inapropiadas y trata de enviar el mensaje, el sistema no permite el envío y muestra un mensaje de advertencia.

Esta funcionalidad se muestra visualmente al usuario en tiempo real, y además, se realiza una verificación adicional en el servidor para mantener la integridad del sistema. Este enfoque dual de validación ayuda a mantener una comunicación clara y profesional con el equipo de soporte.

# INCIDENCIAS DEL PROYECTO Y CONCLUSIONES

* Problemas con el repositorio virtual.¿?

Conclusiones posibles:¿?

- Logros del Proyecto: Resalta los objetivos cumplidos, el impacto en la operación diaria de la empresa y cualquier mejora notable en eficiencia o usabilidad.

- Desafíos y Aprendizajes: Comenta sobre los retos enfrentados durante el desarrollo y las soluciones aplicadas.

- Futuras Mejoras: Propone posibles expansiones o mejoras basadas en la retroalimentación recibida y la evolución de las necesidades empresariales

# Bibliografía

Login: <https://youtu.be/6rMSouSx3II?si=XbTXD1qkjXikD4Eh>

Vista de detalle de pedido: <https://stackoverflow.com/questions/53023775/simple-math-on-django-views-with-decimals>

<https://docs.djangoproject.com/en/5.0/ref/models/expressions/>

Validación de texto en formulario:

<https://www.freecodecamp.org/news/form-validation-in-javascript/>

<https://stackoverflow.com/questions/72188432/form-validation-must-contain-a-specific-word-from-a-list>

Fetch API:

<https://developer.mozilla.org/en-US/docs/Learn/Forms/Sending_forms_through_JavaScript>

<https://www.freecodecamp.org/news/javascript-fetch-api-for-beginners/>