

# **Ingeniería Web – Proyecto Web Colaborativo**

**Título:** - Diseño e implementación de una web de comida a domicilio

**Curso:** 4º Doble Grado en Administración y Dirección de Empresas e Ingeniería Informática (Semestre 1º)

**Asignatura:** Ingeniería Web

**Estudiantes:** Manuel Garcés Laviñeta

Carlos González Díaz

Iker González Requejo

Jorge González Recondo

**Grupo:** 5

**Profesor:** Unai Arambarri Yeregui

**Facultad de Ingeniería**

**UNIVERSIDAD DE DEUSTO**

**Donostia-San Sebastián, diciembre de 2023**



## **Resumen**

Nuestro proyecto consiste en la elaboración de una aplicación web dedicada al consumo a domicilio de hamburguesas. En esta aplicación, se ofrecen los servicios de 10 hamburgueserías, cada una de las cuales, dispone de cuatro categorías. Una categoría de un establecimiento determinado es exclusiva de ese establecimiento, por ejemplo, la categoría vegana de Goiko no ofrece el mismo surtido de hamburguesas que la categoría vegana de Mc Donalds.

Además, se realiza una categorización de los establecimientos según la calidad del producto ofrecido. De tal manera, que, al acceder a la aplicación, el usuario puede elegir entre tres distintas categorías: La categoría prime (la de mayor calidad), categoría sub-prime (de calidad media) y la categoría estándar (de calidad estándar).

Dentro de cada restaurante, la categorización es la misma. Todos contarán con las siguientes categorías: hamburguesa vegana, hamburguesa kilómetro cero, Smashburger y hamburguesa de pollo.

No se ofrecen menús, se debe comprar por separado. La hamburguesa no varía en tamaño ni elaboración independientemente del establecimiento. La hamburguesa es un producto cerrado.

Es necesario un inicio de sesión, o un registro si no se ha registrado anteriormente, cada vez que se quiera realizar un pedido. De esta manera, nuestra aplicación ofrece funcionalidades adicionales.

El servicio de compra funciona de la siguiente manera: Un pedido sólo puede ser realizado en un establecimiento determinado, dentro de este, pueden ser seleccionados todos los productos que desee el consumidor entre las diferentes categorías disponibles.

Se muestra en todo momento los productos añadidos a la cesta, con la opción de eliminar productos añadidos, para filtrar y seleccionar aquellos que se quieran incluir en la compra final.

En el momento de la compra, se piden los datos correspondientes. No hay opción de cancelación una vez se realice el pedido y se envíen los datos correspondientes al correo. En el correo se remiten datos relevantes de la compra como: precio de la compra, número de pedido, hora estimada de llegada y listado de pedidos.

## **Descriptores**

Las 5 palabras clave que ayudan a describir y clasificar nuestro proyecto son: Hamburguesas, personalización, categorización, exclusividad y en línea.

## Índice

<b>1. EXPLICACIÓN GENERAL DEL PROYECTO.....</b>	<b>1</b>
1.1 descripción de los escenarios y funcionalidades básicas.....	1
1.2 descripción de modelo de datos utilizado.....	2
1.3 descripción de las plantillas utilizadas para las vistas.....	3
1.4 Descripción de la estructura del proyecto web.....	4
1.5 Descripción de las funcionalidades añadidas implementadas.....	4
<b>2. EXPLICACIÓN GENERAL DEL PROYECTO VUE.....</b>	<b>5</b>
2.1 Descripción de la estructura general del proyecto.....	5
2.2 Descripción de las funcionalidades extras añadidas.....	5
<b>3. CONCLUSIONES SOBRE EL PROYECTO.....</b>	<b>5</b>
<b>4. BIBLIOGRAFÍA.....</b>	<b>6</b>

# 1. EXPLICACIÓN GENERAL DEL PROYECTO

---

En este primer punto se realiza una explicación general del proyecto, entrando en detalle sobre la descripción de los escenarios y las funcionalidades básicas presentadas, el modelo de datos diseñado para el proyecto, plantillas utilizadas para las vistas, la estructuración del proyecto web y descripción de las funcionalidades extras añadidas.

## 1.1 DESCRIPCIÓN DE LOS ESCENARIOS Y FUNCIONALIDADES BÁSICAS.

Dentro de nuestro proyecto, encontramos dos escenarios principales que recorren la totalidad del proyecto:

En el primer escenario, consideramos que el usuario ya está registrado. En este caso, cuando el cliente entra en la página web, inicia sesión y accede a las diferentes categorías de establecimientos. Una vez seleccionada la categoría, se muestran diferentes restaurantes acordes a estas categorías. Dentro de cada restaurante se muestran diferentes categorías de hamburguesas, y dentro de estas categorías de productos, se ofertan los diferentes productos.

El usuario añade los productos que quiera (del mismo establecimiento) a la cesta. Una vez finalizada su compra, selecciona la cesta, comprueba que los productos que se muestran son los correctos, y procede con el pago de la compra. En esta parte deberá introducir sus datos personales para completar la compra. Se confirma la compra mediante correo electrónico.

En el segundo escenario, el proceso seguido por el usuario es el mismo que en el usuario registrado, pero en este caso, deberá registrarse e iniciar sesión para poder acceder a las categorías de establecimientos.

Por otro lado, dentro de nuestro proyecto podemos encontrar las siguientes funcionalidades básicas:

- Un login que recoge los datos principales del usuario.
- Un proceso de registro para todos aquellos usuarios no registrados.
- Una ventana "home" inicial que categoriza los establecimientos según la calidad de los productos que ofrecen (prime, sub-prime y estándar).
- Una ventana que muestra los diferentes establecimientos dentro de cada categoría.
- Una ventana que muestra los diferentes tipos de hamburguesas que ofrece cada uno de los establecimientos.

- Una ventana que muestra las hamburguesas que encontramos dentro de cada categoría de hamburguesa.
- Una cesta de producto que recopila todos los productos añadidos durante el proceso de compra. En la cesta se puede eliminar un producto añadido, así como aumentar o disminuir la cantidad de productos añadidos.
- Un proceso de pago, para poder realizar la transferencia bancaria correspondiente al pedido realizado.
- Servicio de correo automatizado, que envía al usuario información sobre su pedido realizado.

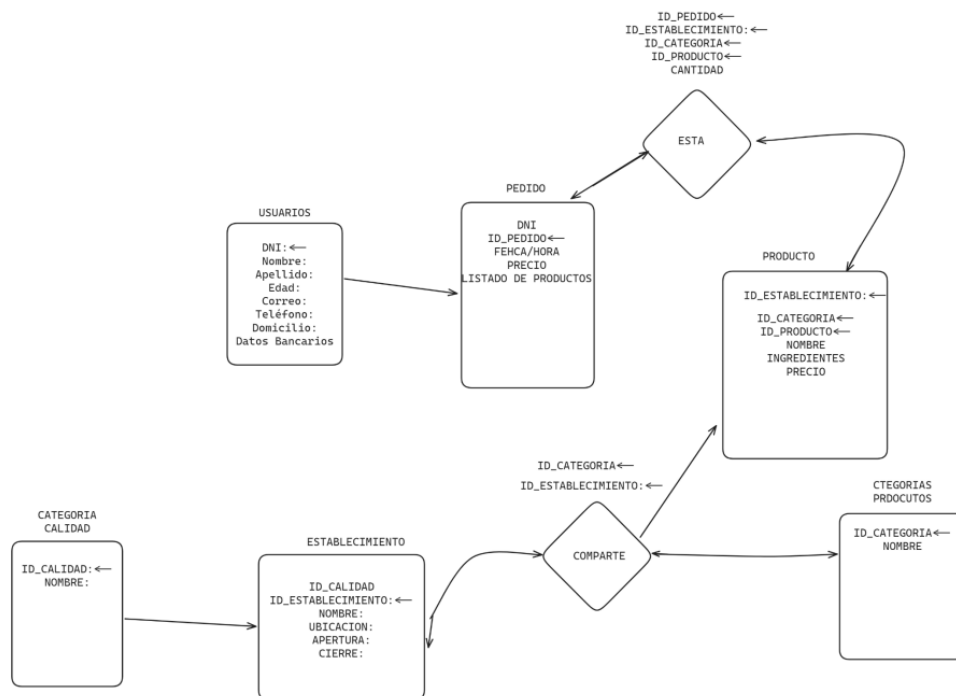
### 1.2 DESCRIPCIÓN DE MODELO DE DATOS UTILIZADO.

En nuestro proyecto los principales modelos utilizados han sido los siguientes:

- **Usuario:** Es el usuario de la aplicación. Los datos recogidos sobre el usuario son: DNI, nombre, apellido, edad, correo, teléfono, domicilio y sus datos bancarios.
- **Pedido:** Pedido realizado desde la aplicación. Los datos que forman el pedido son: Usuario que realiza el pedido, hora del pedido, fecha del pedido, precio total del pedido y listado de productos pedidos.
- **Producto:** Productos que podemos encontrar dentro de nuestra aplicación. Un producto está formado por los siguientes atributos: Establecimiento del que proviene, nombre, ingredientes, categoría, precio y una imagen del producto en cuestión.
- **Establecimiento:** Establecimientos en los que se ofertan los diferentes productos. Un establecimiento está compuesto por: Calidad del establecimiento, nombre, ubicación, hora de apertura y hora de cierre.
- **Categoría del establecimiento:** Como hemos mencionado anteriormente, los establecimientos están divididos en tres categorías principales. La categoría de un establecimiento está compuesta únicamente por su nombre.

- **Categoría de productos:** Dentro de cada establecimiento encontramos cuatro categorías diferentes (Pollo, Vegana, kilómetro – cero y Smash Burger). Cada una de estas categorías está compuesta únicamente por su nombre.

De manera visual los modelos y su relación pueden ser representada visualmente de la siguiente manera:



### 1.3 DESCRIPCIÓN DE LAS PLANTILLAS UTILIZADAS PARA LAS VISTAS

Para este proyecto se han utilizado un total de doce documentos HTML diferentes, cada uno con su correspondiente CSS.

En relación con los HTML destacamos el uso de contenedores como head, body, main o footer en todos los documentos. Por otro lado, los estilos utilizados en los diferentes CSS son similares, de tal manera que todas las diferentes ventanas son coherentes y consistentes en cuanto a estilo se refiere.

En cuanto al uso de javascript, encontramos el código al final de la estructura de los HTML, de tal manera que se mantiene una estructura coherente en todos ellos.

Por último, en cuanto al código realizado con DJANGO, encontramos la carpeta utils en la que están definidas todas las clases y todos los métodos utilizados.

## 1.4 DESCRIPCIÓN DE LA ESTRUCTURA DEL PROYECTO WEB

La estructura que hemos seguido teniendo en cuenta el patrón MVC (model, view, controller es la siguiente):

En la parte de modelos se han definido todas las clases que han sido mencionadas previamente (usuario, pedido, producto, establecimiento, categoría del establecimiento y categoría de productos). Cada una de estas clases representa una tabla de bases de datos.

Estos modelos nos permiten gestionar los datos, definir la estructura de los datos, realizar consultas y manejar relaciones entre diferentes estructuras.

Por otro lado, en el patrón MVC, el controlador actúa como intermediario entre la vista y los modelos. En nuestro caso, los controladores están definidos en la clase de vistas. Esta clase de vistas nos permite recoger solicitudes webs, interactuar con los modelos que previamente hemos definido para obtener datos y devolver una respuesta.

En nuestro proyecto, las funciones básicas que realiza la vista son las siguientes:

- Autenticación.
- Registro.
- Funcionalidades específicas como mostrar las categorías de bares, productos...
- Manejo del carrito de compra.
- Manejo del pago final del pedido.

Por último, la vista del patrón MVC se utiliza para presentar toda la información y datos que se manejan y procesan entre los modelos y los controladores. En nuestro caso, esta información se presenta mediante las plantillas HTML y CSS que hemos mencionado anteriormente.

## 1.5 DESCRIPCIÓN DE LAS FUNCIONALIDADES AÑADIDAS IMPLEMENTADAS

Durante la realización del proyecto se han añadido cuatro principales funcionalidades extra: interacción enriquecida con javascript, implementación de un modelo IA de google MediaPipe, carga de datos iniciales para la aplicación y despliegue de la aplicación a cloud mediante Heroku.

En cuanto a la utilización de javascript, aunque se ha utilizado numerosas veces en el proyecto para pequeñas funciones, mejoras de estilos... Destacamos su uso principalmente en el desarrollo del carrito de compra y el proceso de pago.

En segundo lugar, hemos utilizado el modelo IA de google MediaPie para crear un pequeño chat que nos permite enviar feedback a la empresa, aunque no está optimizado al 100% permite mantener pequeñas conversaciones con respuestas coherentes.

En relación a la carga de datos iniciales, se han aplicado una serie de comandos para extraer toda la información necesaria de la base de datos y cargar los nuevos datos a cada modelo.



Por último, se ha realizado el despliegue de la aplicación a cloud mediante Heroku. Heroku es una plataforma que permite subir y construir proyectos completamente en la nube. Para realizar esto hemos creado tres archivos, requirements.txt para las librerías, Procfile que indica a Heroku como ejecutar el código y el runtime para indicarle a heroku la versión de Django.

Mencionar también que se han realizado ajustes en setting.py para importar heroku.

## **2. EXPLICACIÓN GENERAL DEL PROYECTO VUE**

---

En este primer punto se realiza una explicación general del proyecto VUE, entrando en detalle sobre la estructura general del proyecto y las funcionalidades extras que han sido implementadas.

En este proyecto hemos integrado código Vue en las plantillas html del proyecto. Concretamente, en las plantillas 'user\_register.html' y 'user\_profile.html'.

### **2.1 DESCRIPCIÓN DE LA ESTRUCTURA GENERAL DEL PROYECTO**

En la plantilla 'user\_register.html', si el usuario inserta un nombre de usuario que ya existe, saltará un mensaje de error.

En la plantilla 'user\_profile.html', hemos creado la misma instancia. Si el usuario quiere cambiar el nombre de usuario, pero inserta un nombre existente, se le mostrará un mensaje de error.

Por otro lado, iniciamos el código de desarrollo del carrito en Vue, pero generaba muchos problemas así que optamos por desarrollar el carrito con JavaScript.

### 3. CONCLUSIONES SOBRE EL PROYECTO

---

En resumen, el proyecto engloba el desarrollo de una aplicación web centrada en la gestión de pedidos y productos de establecimientos, centrándonos en la categorización de productos, el proceso de compra, y la implementación de funcionalidades adicionales utilizando tecnologías como Django.

Realizar este proyecto nos ha permitido aprender a desarrollar por completo una aplicación web. Hemos aprendido todos los pasos del desarrollo, desde el diseño de modelo de datos hasta el uso de las vistas y las plantillas HTML y CSS.

Creemos que el aprendizaje obtenido en este proyecto será muy útil en nuestro futuro profesional tanto si nos dedicamos específicamente a la ingeniería web, como si no.

### 4. BIBLIOGRAFÍA

---

Durante la realización de este proyecto se han utilizado las siguientes fuentes de información:

- **CODEPEN:** <https://codepen.io>
- **W3SCHOOLS:** <https://www.w3schools.com/>
- **STACKOVERFLOW:** <https://stackoverflow.com/>
- **CHATGPT:** <https://chat.openai.com/>