

Special

Alumno: Cristóbal Alcalá Cazorla

Tutor: Eladio Blanco López

2º DAW

06/24

Especificaciones del proyecto	3
Descripción del proyecto	3
Funcionalidades del Sistema	3
1. Administrador:	3
2. Cliente:	4
3. Recepcionista:	4
4. Chef:	4
5. Delivery (Repartidor):	5
Flujo de Trabajo	5
1. Registro y Pedido del Cliente:	5
2. Pedido por el Recepcionista:	5
3. Preparación del pedido por el Chef:	5
4. Gestión del Recepcionista de pedidos completados:	6
5. Entrega del pedido mediante Delivery:	6
Beneficios del Sistema	6
Elementos innovadores	7
1. Librerías de Componentes Externas:	7
2. Gestión de Fechas y Horarios:	7
3. Transiciones:	7
4. Iconos:	7
5. Formularios:	8
6. Comunicación en Tiempo Real:	8
7. Notificaciones:	8
8. Framework de Estilos:	8
9. Validación de datos:	8
10. Integración con Servicios en la Nube:	8
11. Seguridad y Encriptación de Contraseñas:	9
Estudio Previo y Prototipo	10
Prototipo y visualización	10
Documentación del Backend	13
1. Lenguajes y Entorno:	13
2. Framework y Middleware:	13
3. Seguridad:	13
4. Utilidades:	14
5. Diseño de la base de datos:	14
6. Documentación de todos los servicios CRUD:	15
Documentación del Frontend	22
1. Lenguajes y Tecnologías:	22
2. Bibliotecas y Herramientas Adicionales:	22
Propuestas de mejora	23
Bibliografía	24
Manuales	25
Manual de Administrador	26
Panel principal:	26
Panel de Empleados y Usuarios:	26
Formularios de Empleados y Usuarios:	27
Panel de Productos:	28
Formulario de Productos:	28
Panel de Secciones:	29
Panel de Pedidos:	29
Panel de Cupones:	30
Manual de Usuario	31
Página principal:	31
Login / Register:	31
Perfil:	32
Pedidos:	32
Menú:	33

Especificaciones del proyecto

Descripción del proyecto

Zpecial es una Web App (Aplicación Web) orientada a negocios de comida rápida donde el negocio tiene la facilidad de poder gestionar su carta, trabajadores, los pedidos, seguimiento en tiempo real de estos y sus usuarios.

Los roles involucrados dentro del sistema son los siguientes:

- Administrador
- Recepcionista
- Chef
- Delivery
- Usuario

Cada uno de los roles tiene una finalidad y funcionalidad diferente la cual asegura un desarrollo de trabajo eficiente y organizado las cuales se especificarán más adelante.

Funcionalidades del Sistema

1. Administrador:

- Gestión de horarios de apertura del comercio.
- Gestión y administración de empleados.
- Gestión y administración de usuarios.
- Gestión y administración de productos.
- Gestión y administración de secciones.
- Visión global de los pedidos del negocio.
- Gestión y administración de cupones de descuento.

2. Cliente:

- Registro y autenticación de su cuenta.
- Visualización de la carta.
- Realización de pedidos y aplicación de cupones de descuento.
- Seguimiento en tiempo real del estado de los pedidos.

3. Recepcionista:

- Gestión de pedidos para entregar, repartir o posibilidad de cancelarlos.
- Recepción y creación de pedidos.
- Capacidad de notificar a repartidores sobre nuevos pedidos.
- Área personal de información donde poder consultar y modificar sus datos.

4. Chef:

- Recepción y visualización de pedidos desde recepción.
- Capacidad de marcado como completado o cancelar pedidos.
- Área personal de información donde poder consultar y modificar sus datos.

5. Delivery (Repartidor):

- Recepción de notificaciones para recogida de nuevos pedidos en tienda.
- Capacidad de marcado como entregado o cancelar pedidos.
- Área personal de información donde poder consultar y modificar sus datos.

Flujo de Trabajo

1. Registro y Pedido del Cliente:

- El cliente se registra en la aplicación, completa sus datos personales y navega por la carta.
- Selecciona los productos que desea pedir, accede al carrito, completa las opciones a sus necesidades y realiza el pedido.
- El pedido es enviado automáticamente al chef para su posterior preparación.

2. Pedido por el Recepcionista:

- El recepcionista puede realizar pedidos desde su terminal y marcarlos como “Delivery” o “Para recoger”.
- El pedido se envía directamente al chef para su posterior preparación.

3. Preparación del pedido por el Chef:

- Recibe el pedido en cocina.
- Si no hay ningún inconveniente realiza el pedido y lo marca a su posterior estado que será dependiendo del tipo de pedido “Por recoger” o “Por Enviar”.

4. Gestión del Recepcionista de pedidos completados:

- El recepcionista recibe de vuelta los pedidos completados por el Chef.
- Puede marcar el pedido como cancelado, recogido (en caso de que sea para recoger) o en reparto (el pedido pasa a manos del delivery).

5. Entrega del pedido mediante Delivery:

- Puede recibir notificaciones para recoger un nuevo pedido en el caso de que no esté presente en tienda.
- Puede cancelar el pedido o marcarlo como entregado una vez finaliza el reparto.

Beneficios del Sistema

- Eficiencia: Gestión del pedido desde la recepción hasta la entrega de este, solo se tiene que preocupar los trabajadores de procesar o cancelar el pedido.
- Transparencia: El cliente puede ver en tiempo real el estado de su pedido y estar informado en todo momento.
- Gestión integral: La gestión de todos los parámetros del sistema como clientes, empleados, productos... se realizan desde un mismo lugar.

Elementos innovadores

Durante el desarrollo de este proyecto, se han incorporado diversos elementos y tecnologías que no fueron enseñados a lo largo del ciclo. Estos componentes y librerías externas han sido el resultado de investigación propia cuya finalidad ha sido mejorar la funcionalidad y la experiencia del usuario.

A continuación, detallaré por encima algunas de estas librerías y componentes que he decidido implementar en mi proyecto:

1. Librerías de Componentes Externas:

- Shadcn y Aceternity: Estas librerías proporcionan componentes y estilos predefinidos para mejorar el uso de datos, diseño y la apariencia visual de la aplicación.

2. Gestión de Fechas y Horarios:

- Librería Date-fns: Utilizada para el manejo simplificado y más eficiente entre fechas y horas facilitando por ejemplo la visualización de estas.

3. Transiciones:

- Framer Motion: Se ha introducido para generar transiciones más suaves y fluidas en el cambio de rutas de la aplicación. En general da un toque más atractivo e el momento de navegar entre los diferentes apartados de la aplicación.

4. Iconos:

- Lucide React: Para el uso de iconos he utilizado esta librería ya que es muy completa, a parte de ser simples y minimalistas permite modificar tanto el tamaño como el grosor de los iconos adaptándolo a mis necesidades.

5. Formularios:

- React Hook Form: Empleado para la validación de formularios en la aplicación, proporcionando una mejor forma de gestionar la entrada y garantizar la integridad de los datos.

6. Comunicación en Tiempo Real:

- Socket.io: Se ha utilizado para la implementación de la comunicación en tiempo real entre componentes de la aplicación como el estado del pedido en todo momento en la parte de los trabajadores como la visualización de este por parte del usuario final.

7. Notificaciones:

- Sonner: Se ha incorporado esta librería que permite mostrar pop-ups cuando se realiza una acción e informar a la persona que está usando la aplicación, muy personalizables y agradables a la vista.

8. Framework de Estilos:

- TailwindCSS: Este framework tan famoso se ha utilizado para agilizar el proceso de diseño y desarrollo, permitiéndome crear la interfaz y el UI de una manera más efectiva y liviana.

9. Validación de datos:

- Zod: Implementado para la validación de datos en los formularios y así garantizar la integridad y seguridad de los datos enviados y que recibirá el servidor.

10. Integración con Servicios en la Nube:

- AWS SDK para s3 y Cloudflare R2: Utilizando AWS SDK se integró con Cloudflare R2 para optimizar el rendimiento y la entrega de contenido estático, lo que incluye la obtención del contenido multimedia del bucket de R2.

11. Seguridad y Encriptación de Contraseñas:

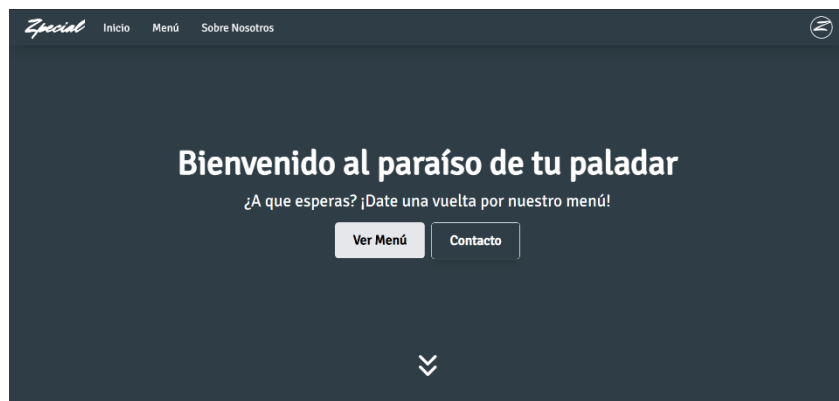
- Bcrypt: Empleado para la encriptación segura de contraseñas de los usuarios, garantizando la protección de datos y evitando vulnerabilidades de seguridad como la exposición de contraseñas en texto plano.

Estudio Previo y Prototipo

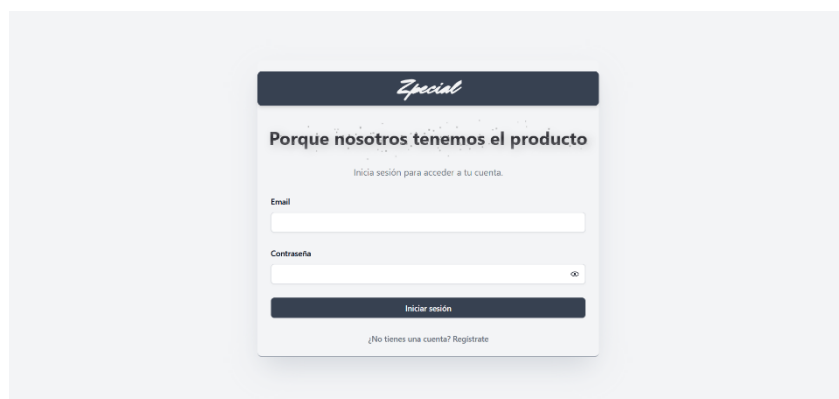
La idea de realizar el proyecto me vino tras ver la página de pedidos que tiene un negocio local de donde resido, este negocio utiliza un servicio SaaS que proporciona la empresa [Hiopos](#), me propuse a realizar una opción que pudiese ser más funcional e intuitiva en su uso.

La idea es que cada trabajador tenga su espacio de trabajo pero que esté comunicado con los otros puestos fácilmente.

Prototipo y visualización



Landing Page



Login

Special

¡Hola Cristóbal!

Panel principal

Empleados

Usuarios

Productos

Secciones

Pedidos

Cupones

Horarios de apertura

(Dejar vacío para mantener cerrado)

Lunes

Ingreso al horario

Cerrado

Miércoles

12:00 - 16:00 / 20:00 - 00:00

Cerrado

Viernes

12:00 - 16:00 / 20:00 - Cierre

Cerrado

Domingo

12:00 - 16:00

Cerrado

Martes

12:00 - 16:00 / 20:00 - 00:00

Cerrado

Jueves

12:00 - 16:00 / 20:00 - 00:00

Cerrado

Sábado

12:00 - 16:00 / 20:00 - Cierre

Cerrado

Administrador

¡Hola, Cristóbal!

Hamburguesas

Pizzas Platos Combinados Borrachillos Patatas Asadas Entrantes Postres Bebidas

<p>Classic chicken 5.90 €</p> <p>Inгредиенты: ○</p> <p>Cantidad: - 1 +</p> <p>Agregar al carrito</p>	<p>Zpecial classic 6.90 €</p> <p>Inгредиенты: ○</p> <p>Cantidad: - 1 +</p> <p>Agregar al carrito</p>	<p>Cabrita 6.90 €</p> <p>Inгредиенты: ○</p> <p>Cantidad: - 1 +</p> <p>Agregar al carrito</p>	<p>The ring 7.00 €</p> <p>Inгредиенты: ○</p> <p>Cantidad: - 1 +</p> <p>Agregar al carrito</p>
--	--	--	---

Recepcionista

Chef

Special

¡Hola, Cristóbal!



Siguiente >

#ccc6c042-882e-4309-b926-906b8d066582

06/01/2024, 07:07 PM

🕒 En reparto

7.90 €
Total

Producto	Cantidad	Precio
Cebolla	1	6.9 €

📍 Dirección Falsa

No entregado

Marcar como entregado

Delivery

Documentación del Backend

1. Lenguajes y Entorno:

- Node.js: Entorno de ejecución de JS en el servidor.
- JavaScript: Lenguaje de programación utilizado.

2. Framework y Middleware:

- Express.js: Framework para la creación de la API RESTful.
- Body-parser: Middleware para parsear los cuerpos de las solicitudes HTTP.
- CORS: Middleware para habilitar el Cross-Origin y que no de conflicto al intentar acceder desde otro dominio.
- Morgan: Middleware para mostrar/registrar solicitudes que se hagan a servidor.
- Apicache: Middleware para cachear respuestas.
- Socket.io: Biblioteca para poder hacer uso de Websockets.

3. Seguridad:

- Bcrypt: Biblioteca para la encriptación de contraseñas.

4. Utilidades:

- Dotenv: Necesario para poder gestionar variables de entorno.
- UUID: Herramienta para generar identificadores únicos.
- Nodemon: Herramienta para reiniciar el servidor automáticamente al realizar cambios en este.
- AWS-SDK: SDK para interactuar con los servicios de Amazon Web Services.

5. Diseño de la base de datos:

Para el diseño de la base de datos he optado por un modelo no relacional guardando todos los datos de mi aplicación en ficheros JSON.

6. Documentación de todos los servicios CRUD:

- Empleados:

The first screenshot shows a GET request to `http://localhost:3001/api/v1/employees`. The response is a JSON array with two employee objects. The second screenshot shows a GET request to `http://localhost:3001/api/v1/employees?name=Cristóbal`. The response is a JSON array with one employee object. The third screenshot shows a GET request to `http://localhost:3001/api/v1/employees?email=chef@gmail.com`. The response is a JSON array with one employee object.

GET `http://localhost:3001/api/v1/employees` Send

Query Headers Auth Body Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content

1

Status: 200 OK Size: 155 KB Time: 3 ms

Response Headers Cookies Results Docs

```
[
  {
    "name": "Cristóbal",
    "surname": "Alcalá",
    "dni": "22222222Y",
    "address": "Pedro Herrera Zamora 3",
    "position": "",
    "phone": "666666666",
    "userName": "Alcazor",
    "email": "cristobal.alcala@fernando3martos.com",
    "password": "$2b$10$AS8bwG65q70rzqHfMD9.LI2GHFLLRR.WbRTTgxb4.QhRTPIAT.",
    "admin": true,
    "id": "71f3548f-5d69-4f59-888b-4be8d1878291",
    "created_at": "04/03/2024, 05:35:12 PM",
    "updated_at": "05/18/2024, 02:49:57 PM"
  },
  {
    "name": "Cristóbal",
    "surname": "Alcalá",
    "dni": "22222222Y",
    "address": "Pedro Herrera Zamora 3",
    "position": "Receptionist",
    "phone": "666666666",
    "userName": "Alcazor",
    "email": "receptionist@gmail.com",
    "password": "$2b$10$AS8bwG65q70rzqHfMD9.LI2GHFLLRR.WbRTTgxb4.QhRTPIAT.",
    "admin": false,
    "id": "71f3548f-5d69-4f59-888b-4be8d1878291",
    "created_at": "04/03/2024, 05:35:12 PM",
    "updated_at": "05/18/2024, 02:49:57 PM"
  }
]
```

GET `http://localhost:3001/api/v1/employees?name=Cristóbal` Send

Query Headers Auth Body Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content

1

Status: 200 OK Size: 155 KB Time: 3 ms

Response Headers Cookies Results Docs

```
[
  {
    "name": "Cristóbal",
    "surname": "Alcalá",
    "dni": "22222222Y",
    "address": "Pedro Herrera Zamora 3",
    "position": "",
    "phone": "666666666",
    "userName": "Alcazor",
    "email": "cristobal.alcala@fernando3martos.com",
    "password": "$2b$10$AS8bwG65q70rzqHfMD9.LI2GHFLLRR.WbRTTgxb4.QhRTPIAT.",
    "admin": true,
    "id": "71f3548f-5d69-4f59-888b-4be8d1878291",
    "created_at": "04/03/2024, 05:35:12 PM",
    "updated_at": "05/18/2024, 02:49:57 PM"
  }
]
```

GET `http://localhost:3001/api/v1/employees?email=chef@gmail.com` Send

Query Headers Auth Body Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content

1

Status: 200 OK Size: 387 Bytes Time: 3 ms

Response Headers Cookies Results Docs

```
[
  {
    "name": "Cristóbal",
    "surname": "Alcalá",
    "dni": "22222222Y",
    "address": "Pedro Herrera Zamora 3",
    "position": "Chef",
    "phone": "666666666",
    "userName": "Alcazor",
    "email": "chef@gmail.com",
    "password": "$2b$10$AS8bwG65q70rzqHfMD9.LI2GHFLLRR.WbRTTgxb4.QhRTPIAT.",
    "admin": false,
    "id": "71f3548f-5d69-4f59-888b-4be8d1878291",
    "created_at": "04/03/2024, 05:35:12 PM",
    "updated_at": "05/18/2024, 02:49:57 PM"
  }
]
```

GET

POST ▼ http://localhost:3001/api/v1/employees Send

Query Headers ² Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1  {
2    "name": "Cristóbal",
3    "surname": "Alcalá",
4    "dni": "22222227Y",
5    "address": "Pedro Herrera Zamora 3",
6    "position": "Chef",
7    "phone": "666666666",
8    "userName": "Alcazor",
9    "email": "prueba@gmail.com",
10   "password": "prueba",
11   "admin": false
12 }
```

POST

PATCH ▼ http://localhost:3001/api/v1/employees/71f3548f-5dd9-4f59-888b-4be0d1878 Send

Query Headers ² Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1  {
2    "admin": true
3 }
```

PATCH

DELETE ▼ http://localhost:3001/api/v1/employees/71f3548f-5dd9-4f59-888b-4be0d1878 Send

DELETE

• Usuarios:

GET ▼ http://localhost:3001/api/v1/users?email=cristobal.alcala.smr@gmail.com Send

Query Headers ² Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1  [
2    {
3      "name": "Cris",
4      "surname": "Alcala",
5      "address": "Mi casa",
6      "phone": "666666666",
7      "userName": "Alcazor",
8      "email": "cristobal.alcala.smr@gmail.com",
9      "password":
10       "$2b$10$MqYQyFvQI5o9wOvnx6oeEt8VeZqY35YzrB
11       /f4Bq39zpkJ3e.le",
12      "id": "fd836eca-2f63-4a13-bbfa-c3bcf0db0557",
13      "created_at": "06/01/2024, 11:17:20 PM",
14      "updated_at": "06/01/2024, 11:17:20 PM"
15    }
16  ]
```

Status: 200 OK Size: 334 Bytes Time: 0 ms

GET ▼ http://localhost:3001/api/v1/users Send

Query Headers ² Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1  [
2    {
3      "name": "Cris",
4      "surname": "Alcala",
5      "address": "Mi casa",
6      "phone": "666666666",
7      "userName": "Alcazor",
8      "email": "cristobal.alcala.smr@gmail.com",
9      "password":
10       "$2b$10$MqYQyFvQI5o9wOvnx6oeEt8VeZqY35YzrB
11       /f4Bq39zpkJ3e.le",
12      "id": "fd836eca-2f63-4a13-bbfa-c3bcf0db0557",
13      "created_at": "06/01/2024, 11:17:20 PM",
14      "updated_at": "06/01/2024, 11:17:20 PM"
15    }
16  ]
```

Status: 200 OK Size: 334 Bytes Time: 0 ms

GET


```
POST http://localhost:3001/api/v1/users Send
Query Headers 2 Auth Body Tests Pre Run
JSON XML Text Form Form-encode GraphQL Binary
JSON Content Format
1 {
2   "name": "Cris",
3   "surname": "Alcala",
4   "address": "Mi casa",
5   "phone": "666666666",
6   "userName": "Alcazor",
7   "email": "crisobal.alcala@gmail.com",
8   "password": "prueba"
9 }
```

POST

```
PATCH http://localhost:3001/api/v1/users/fd836ece-2f63-4413-bbfa-c3bcf0db0557 Send
Query Headers 2 Auth Body Tests Pre Run
JSON XML Text Form Form-encode GraphQL Binary
JSON Content Format
1 {
2   "email": "crisobal.alcala@gmail.com"
3 }
```

PATCH

```
DELETE http://localhost:3001/api/v1/users/fd836ece-2f63-4413-bbfa-c3bcf0db0557 Send
```

DELETE

- Categorías:

```
GET http://localhost:3001/api/v1/categories Send
Query Headers 1 Auth Body 1 Tests Pre Run
JSON XML Text Form Form-encode GraphQL Binary
JSON Content Format
1 [
2   {
3     "name": "Hamburguesas",
4     "id": "1",
5     "updated_at": "05/05/2024, 18:05:58 PM",
6     "available": true
7   },
8   {
9     "name": "Pizzas",
10    "id": "2",
11    "updated_at": "05/05/2024, 09:40:50 PM",
12    "available": true
13  },
14  {
15    "name": "Platos Combinados",
16    "id": "3",
17    "updated_at": "2016-06-01T12:00:00Z",
18    "available": true
19  },
20  {
21    "name": "Bocadillos",
22    "id": "4",
23    "updated_at": "2016-06-01T12:00:00Z",
24    "available": true
25  },
26  {
27    "name": "Patatas Asadas",
28    "id": "5",
29    "updated_at": "05/05/2024, 09:40:52 PM",
30    "available": true
31  },
32 ]
```

```
GET http://localhost:3001/api/v1/categories?name=Bocadillos Send
Query Headers 1 Auth Body 1 Tests Pre Run
JSON XML Text Form Form-encode GraphQL Binary
JSON Content Format
1 [
2   {
3     "name": "Bocadillos",
4     "id": "4",
5     "updated_at": "2016-06-01T12:00:00Z",
6     "available": true
7   }
8 ]
```

GET

POST

http://localhost:3001/api/v1/categories

Send

QueryHeaders²AuthBodyTestsPre Run

JSONXMLTextFormForm-encodeGraphQLBinary

JSON ContentFormat

```
1 {
2   "name": "Kebabs",
3   "available": true
4 }
```

POST

PATCH

http://localhost:3001/api/v1/categories/4

Send

QueryHeaders²AuthBodyTestsPre Run

JSONXMLTextFormForm-encodeGraphQLBinary

JSON ContentFormat

```
1 {
2   "available": false
3 }
```

PATCH

DELETE

http://localhost:3001/api/v1/categories/4

Send

DELETE

• Cupones:

GET

http://localhost:3001/api/v1/coupons

Send

Status: 200 OK Size: 265 Bytes Time: 3 ms

QueryHeaders²AuthBodyTestsPre Run

JSONXMLTextFormForm-encodeGraphQLBinary

JSON ContentFormat

```
1 [
2   {
3     "name": "JUNE2024",
4     "id": 1,
5     "discount": 40,
6     "description": "40% off your order",
7     "available": true,
8     "updated_at": "05/05/2024, 05:35:25 PM"
9   },
10  {
11    "name": "JULY2024",
12    "id": 2,
13    "discount": 10,
14    "description": "10% off your order",
15    "available": true,
16    "updated_at": "05/19/2024, 08:36:58 PM"
17  }
18 ]
```

GET

GET

http://localhost:3001/api/v1/coupons?name=june2024

Send

Status: 200 OK Size: 133 Bytes Time: 3 ms

QueryHeaders²AuthBodyTestsPre Run

JSONXMLTextFormForm-encodeGraphQLBinary

JSON ContentFormat

```
1 [
2   {
3     "name": "JUNE2024",
4     "id": 1,
5     "discount": 40,
6     "description": "40% off your order",
7     "available": true,
8     "updated_at": "05/05/2024, 05:35:25 PM"
9   }
10 ]
```

POST ▼ http://localhost:3001/api/v1/coupons Send

Query Headers ² Auth Body Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1  {
2    "name": "JUNE2025",
3    "discount": 40,
4    "description": "40% off your order",
5    "available": true
6  }

```

POST

PATCH ▼ http://localhost:3001/api/v1/coupons/1 Send

Query Headers ² Auth Body Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1  {
2    "available": false
3  }

```

PATCH

DELETE ▼ http://localhost:3001/api/v1/coupons/1 Send

DELETE

• Comidas:

GET ▼ http://localhost:3001/api/v1/foodServices Send

Query Headers ¹ Auth Body Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1  {
2    "id": "2",
3    "name": "Classic chicken",
4    "ingredients": [
5      "pan de hamburguesa",
6      "pechuga de pollo",
7      "queso cheddar",
8      "lechuga",
9      "tomate",
10     "cebolla",
11     "mayonesa"
12   ],
13   "price": 8.9,
14   "image": "chickenburger.jpg",
15   "description": "Hamburguesa de pollo con queso cheddar, lechuga, tomate, cebolla y mayonesa.",
16   "section": "1",
17   "available": true,
18   "updated_at": "05/25/2024, 06:40:43 PM"
19 },
20 {
21   "id": "3",
22   "name": "Special classic",
23   "ingredients": [
24     "pan de hamburguesa",
25     "carne de ternera",
26     "queso cheddar",
27     "lechuga",
28     "tomate",

```

GET ▼ http://localhost:3001/api/v1/foodServices?category=3 Send

Query Headers ¹ Auth Body Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1  {
2    "id": "6",
3    "name": "Combinado Lomo",
4    "ingredients": [
5      "huevo",
6      "pimiento",
7      "patatas",
8      "lomo de cerdo",
9      "mayonesa"
10   ],
11   "price": 9.9,
12   "image": "combinadolomo.webp",
13   "description": "Combinado de lomo de cerdo con huevo, pimiento, patatas y mayonesa.",
14   "section": "3",
15   "available": true,
16   "updated_at": "05/05/2024, 09:40:35 PM"
17 },
18 {
19   "id": "7",
20   "name": "Combinado Pollo",
21   "ingredients": [
22     "huevo",
23     "pimiento",
24     "patatas",
25     "pechuga de pollo",
26     "mayonesa"
27   ],
28   "price": 10.9,
29   "image": "combinadopollo.jpg",

```

GET

```

POST http://localhost:3001/api/v1/foodServices
{
  "name": "Combinado Vegetariano",
  "ingredients": [
    "huevo",
    "pimiento",
    "patatas",
    "hamburguesa de garbanzos",
    "mayonesa"
  ],
  "price": 10,
  "image": "combinadoveg.webp",
  "description": "Combinado vegetariano",
  "section": "3",
  "available": true
}

```

POST

```

PATCH http://localhost:3001/api/v1/foodServices/6
{
  "available": false
}

```

PATCH

```

DELETE http://localhost:3001/api/v1/foodServices/6

```

DELETE

• Pedidos:

```

GET http://localhost:3001/api/v1/orders
{
  "products": [
    {
      "special classic",
      1,
      6.9
    }
  ],
  "comments": "Sin cebolla",
  "address": "MI casa",
  "subtotal": 6.9,
  "total": 5.14,
  "type": "Para llevar",
  "completed": false,
  "status": "En proceso",
  "receipt_date": "2024-06-01T22:23:22.232Z",
  "user": "4a6b1a9-865d-43d9-bf51-2ff1b81cf6b",
  "discount": 0,
  "ip": "4a623c09-6160-495c-a3ee-dc8ff683ec1"
}

```

```

GET http://localhost:3001/api/v1/orders/user=4a6b1a9-865d-43d9-bf51-2ff1b81cf6b
{
  "products": [
    {
      "special classic",
      1,
      6.9
    }
  ],
  "comments": "Sin cebolla",
  "address": "MI casa",
  "subtotal": 6.9,
  "total": 5.14,
  "type": "Para llevar",
  "completed": false,
  "status": "En proceso",
  "receipt_date": "2024-06-01T22:23:22.232Z",
  "user": "4a6b1a9-865d-43d9-bf51-2ff1b81cf6b",
  "discount": 0,
  "ip": "4a623c09-6160-495c-a3ee-dc8ff683ec1"
}

```

GET

POST Send

Query Headers 2 Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1 {
2   "products": [
3     [
4       "Zpecial classic",
5       1,
6       6.9
7     ]
8   ],
9   "comments": "Sin cebolla",
10  "address": "Mi casa",
11  "subtotal": 6.9,
12  "total": 5.14,
13  "type": "Para llevar",
14  "completed": false,
15  "state": "En proceso",
16  "user": "4a6bf1a9-865d-43d9-bf51-2ff1b861cf6b",
17  "discount": 40
18 }

```

DELETE Send

POST

DELETE

• Horarios:

GET Send

Query Headers 2 Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

Status: 200 OK Size: 253 Bytes Time: 13 ms

Response Headers 2 Cookies Results Docs {}

```

1 {
2   "monday": "",
3   "tuesday": "12:00 - 16:00 / 20:00 - 00:00",
4   "wednesday": "12:00 - 16:00 / 20:00 - 00:00",
5   "thursday": "12:00 - 16:00 / 20:00 - 00:00",
6   "friday": "12:00 - 16:00 / 20:00 - Cierre",
7   "saturday": "12:00 - 16:00 / 20:00 - Cierre",
8   "sunday": "12:00 - 16:00"
9 }

```

GET

PATCH Send

Query Headers 2 Auth **Body** Tests Pre Run

JSON XML Text Form Form-encode GraphQL Binary

JSON Content Format

```

1 {
2   "monday": "12:00 - 16:00 / 20:00 - 00:00",
3   "tuesday": "12:00 - 16:00 / 20:00 - 00:00",
4   "wednesday": "12:00 - 16:00 / 20:00 - 00:00",
5   "thursday": "12:00 - 16:00 / 20:00 - 00:00",
6   "friday": "12:00 - 16:00 / 20:00 - Cierre",
7   "saturday": "12:00 - 16:00 / 20:00 - Cierre",
8   "sunday": "12:00 - 16:00"
9 }

```

PATCH

Documentación del Frontend

1. Lenguajes y Tecnologías:

- JavaScript y TypeScript: Lenguajes de programación utilizados para desarrollar la lógica.
- React: Biblioteca principal utilizada.
- TailwindCSS: Framework de CSS utilizado para diseños.

2. Bibliotecas y Herramientas Adicionales:

- Shadcn y Aceternity: Componentes reutilizables que he utilizado para una mejor estética y muestreo de datos.
- React Router Dom: Para manejar el routing de mi aplicación.
- React Hook Form: Biblioteca utilizada para el manejo de formularios.
- Framer Motion: Biblioteca que he utilizado para el fin de brindar unas transiciones más agradables al usuario entre elementos del routing.
- Socket.io-client: Cliente de socket io que he utilizado para emitir eventos mediante websocket y recibir los emitidos por el servidor.

Propuestas de mejora

- Uploader y galería de imágenes a cloudflare.
- Pasarela de pagos.
- Sistemas de puntos para clientes y poder redimir promociones.
- Gestión de stock en tienda.
- Información de alérgenos.
- Conversión en un SaaS y poder establecer entornos por cliente.
- Base de datos escalable.

Bibliografía

- [Shadcn](#)
- [Aceternity](#)
- [v0](#)
- [React Hook Form](#)
- [Framer Motion](#)
- [Socket.io](#)
- [TailwindCSS](#)

Manuales

Manual de Administrador

Panel principal:

¡Hola Cristóbal!

Panel principal

Empleados

Usuarios

Productos

Secciones

Pedidos

Cupones

Horarios de apertura

(Dejar vacío para mantener cerrado)

Lunes

Ingresar el horario

Cerrado ☒

Martes

12:00 - 16:00 / 20:00 - 00:00

Cerrado ☐

Miércoles

12:00 - 16:00 / 20:00 - 00:00

Cerrado ☒

Jueves

12:00 - 16:00 / 20:00 - 00:00

Cerrado ☐

Viernes

12:00 - 16:00 / 20:00 - Cierre

Cerrado ☒

Sábado

12:00 - 16:00 / 20:00 - Cierre

Cerrado ☐

Domingo

12:00 - 16:00

Cerrado ☒

En este apartado podremos personalizar los horarios de apertura que se reflejarán en la Landing page. El funcionamiento es el siguiente:

- Para abrir un día desbloquea la casilla con el switch y escribe en el campo.
- Para cerrar un día tienes que dejar el campo vacío y automáticamente al guardar los cambios se marcará como cerrado.

Panel de Empleados y Usuarios:

Nombre	Apellidos	DNI	Dirección	Teléfono	Alias	Email	Puesto	Activo	Última actualización	Acciones
Cristóbal	Alcalá	22222222	Pedro Herrera Zamora 3	666666666	Alcaxor	cristobal.alcala@fernandofernando.com	Recepcionista	X	18/05/2024 14:49	✎ ✖
Cristóbal	Alcalá	22222222	Pedro Herrera Zamora 3	666666666	Alcaxor	recepcionista@gmail.com		X	02/06/2024 05:59	✎ ✖
Cristóbal	Alcalá	22222227	Pedro Herrera Zamora 3	666666666	Alcaxor	chef@gmail.com	Chef	X	18/05/2024 14:49	✎ ✖
Cristóbal	Alcalá	22222228	Pedro Herrera Zamora 3	666666666	Alcaxor	delivery@gmail.com	Delivery	X	18/05/2024 14:49	✎ ✖

Nombre	Apellidos	Dirección	Teléfono	Alias	Email	Última actualización	Acciones
--------	-----------	-----------	----------	-------	-------	----------------------	----------

No hay usuarios para mostrar

Aquí se encuentran las listas de usuarios y empleados, las listas son similares pero los formularios tienen sus diferencias, no pueden haber empleados que accedan como clientes y viceversa, tendrán que ser migradas las cuentas o usar cuentas diferentes.

Formularios de Empleados y Usuarios:

The 'Nuevo empleado' form is divided into three main sections: 'Datos personales', 'Datos de acceso', and 'Permisos'. The 'Datos personales' section includes fields for 'Nombre', 'Apellidos', 'DNI', 'Teléfono', and 'Dirección Postal'. The 'Datos de acceso' section includes fields for 'Usuario', 'Correo electrónico', and 'Contraseña'. The 'Permisos' section includes a 'Cargo' dropdown and an 'Admin' toggle switch.

The 'Nuevo usuario' form is divided into three main sections: 'Datos personales', 'Datos de acceso', and 'Permisos'. The 'Datos personales' section includes fields for 'Nombre', 'Apellidos', 'Teléfono', and 'Dirección Postal'. The 'Datos de acceso' section includes fields for 'Usuario', 'Correo electrónico', and 'Contraseña'. The 'Permisos' section includes a 'Cargo' dropdown and an 'Admin' toggle switch.

Como podrás observar los formularios son diferentes ya que para cada tipo de usuario se toman datos diferentes, hay tanto campos obligatorios (email, contraseña...) como opcionales (nombre, dirección, usuario...).

The 'Editar empleado' form is divided into three main sections: 'Datos personales', 'Datos de acceso', and 'Permisos'. The 'Datos personales' section includes fields for 'Nombre', 'Apellidos', 'DNI', 'Teléfono', and 'Dirección Postal'. The 'Datos de acceso' section includes fields for 'Usuario', 'Correo electrónico', and 'Contraseña'. The 'Permisos' section includes a 'Cargo' dropdown and an 'Admin' toggle switch. A red box highlights the 'Transferir a usuarios >>' button at the bottom left of the form.

Aquí podremos destacar el botón de transferir que estará disponible tanto cuando modifiques un usuario o un empleado, este botón sirve para transferir entre las diferentes categorías de usuarios que puede recoger la aplicación sin necesidad de eliminar y crear nuevos.

Panel de Productos:

Special

¡Hola Cristóbal!

Panel principal

Empleados

Usuarios

Productos

Secciones

Pedidos

Cupones

Productos (56)

Q Buscar...

Imagen	Nombre	Ingredientes	Descripción	Sección	Tamaño	Precio	Disponible	Última actualización	Acciones
	9 fromage	Masa fina, Salsa de tomate, Mozzarella, Gouda, Cheddar, Emmental, Queso de cabra, Gorgonzola, Taleggio, Brie, Queso curado Italiano, Queso azul	Para los cheese lovers aquí tenemos una opción para vosotros, os aseguramos que será una victoria asegurada.	Pizzas		8.5	<input checked="" type="checkbox"/>	26/05/2024 18:17	
	Africanito	Pan de barra, Pollo o Lomo, Mayonesa, Especies, Cebolla	Proveniente del sur de Andalucía os traemos esta delicia de bocadillo que hará que repitas siempre que nos visites.	Bocadillos		4	<input checked="" type="checkbox"/>	02/06/2024 01:01	
	Agua			Bebidas	1.5L	1.2	<input checked="" type="checkbox"/>	26/05/2024 17:38	
	Balsámica	Masa fina, Cebolla morada, Pimiento rojo, Alcachofas, Piñones, Salsa balsámica, tomate, Queso	Pizza con una deliciosa salsa balsámica y acompañada de jugosa salsa de tomate y queso fundido, combinación de sabores mediterráneos que deleitará tu paladar.	Pizzas		9	<input checked="" type="checkbox"/>	25/05/2024 19:20	
	Bandido	Pollo, Salsa barbacoa, Corn flakes	Para los más bandidos tenemos esta opción picante y dulce que no hará arrepentirte de haberlo pedido.	Entrantes		5	<input checked="" type="checkbox"/>	26/05/2024 17:08	
	Bocadillo de Lomo	baguette, lomo de cerdo, pimiento, mayonesa	Bocadillo de lomo de cerdo con pimiento y mayonesa.	Bocadillos		5	<input checked="" type="checkbox"/>	19/05/2024 21:31	
	Cabrita	Pan bríoche, Carne de ternera, Cebolla caramelizada. Queso de cabra. Salsa de	Deliciosa hamburguesa que volverá loco a tu paladar con ese sabor tan intenso como dulce de	Hamburguesas		6.9	<input checked="" type="checkbox"/>	26/05/2024 15:27	

Aquí podremos ver la información de nuestros productos de la tienda, tendremos acciones rápidas como mostrarlo o no mostrarlo en el menú, eliminarlo o editarlo, también se pueden filtrar y ordenar por diferentes campos como el nombre o l última actualización.

Formulario de Productos:

Special

¡Hola Cristóbal!

Panel principal

Empleados

Usuarios

Productos

Secciones

Pedidos

Cupones

Nuevo producto

ComidaBebida

Especificaciones principales

Nombre

Precio

Sección

Ingredientes (separar por coma)

Descripción (Max 200)

Disponible

Special

¡Hola Cristóbal!

Panel principal

Empleados

Usuarios

Productos

Secciones

Pedidos

Cupones

Nuevo producto

ComidaBebida

Especificaciones principales

Nombre

Precio

Tamaño

Disponible

Aquí podemos destacar dos formularios separados por tabs, uno diferente para las comidas o productos que no sean simples bebidas y otra exclusivamente para bebidas.

Panel de Secciones:

Nombre	Disponible	Última actualización	Acciones
Bebidas	<input checked="" type="checkbox"/>	17/05/2024 15:46	<input type="checkbox"/> <input type="checkbox"/>
Bocadillos	<input checked="" type="checkbox"/>	01/06/2024 14:00	<input type="checkbox"/> <input type="checkbox"/>
Entrantes	<input checked="" type="checkbox"/>	01/05/2024 22:05	<input type="checkbox"/> <input type="checkbox"/>
Hamburguesas	<input checked="" type="checkbox"/>	01/05/2024 22:05	<input type="checkbox"/> <input type="checkbox"/>
Patatas Asadas	<input checked="" type="checkbox"/>	01/05/2024 21:40	<input type="checkbox"/> <input type="checkbox"/>
Pizzas	<input checked="" type="checkbox"/>	01/05/2024 21:40	<input type="checkbox"/> <input type="checkbox"/>
Platos Combinados	<input checked="" type="checkbox"/>	01/06/2024 14:00	<input type="checkbox"/> <input type="checkbox"/>
Postres	<input checked="" type="checkbox"/>	01/05/2024 21:17	<input type="checkbox"/> <input type="checkbox"/>

Nueva sección

Nombre

Disponible ☐

Guardar Cancelar

Aquí podemos observar tanto la tabla como el formulario de las secciones que aguardará nuestro negocio de comidas, estas secciones son las que podrán adoptar posteriormente cada uno de los productos.

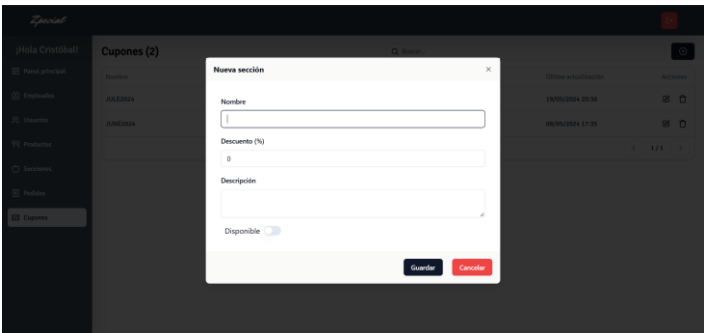
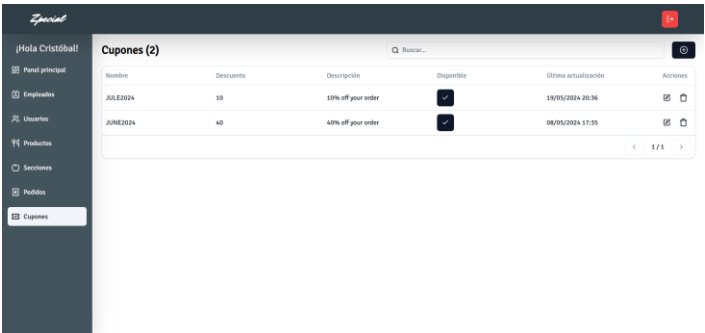
Panel de Pedidos:

Fecha de recibo	Completado	Estado	Usuario	Subtotal	Total	Descuento	Productos	Tipo	Comentarios	Dirección	Acciones
-----------------	------------	--------	---------	----------	-------	-----------	-----------	------	-------------	-----------	----------

No hay pedidos para mostrar

Este panel es meramente informativo, mostrará los pedidos que se han ido realizando en tu negocio, los pedidos se podrán eliminar pero no modificar.

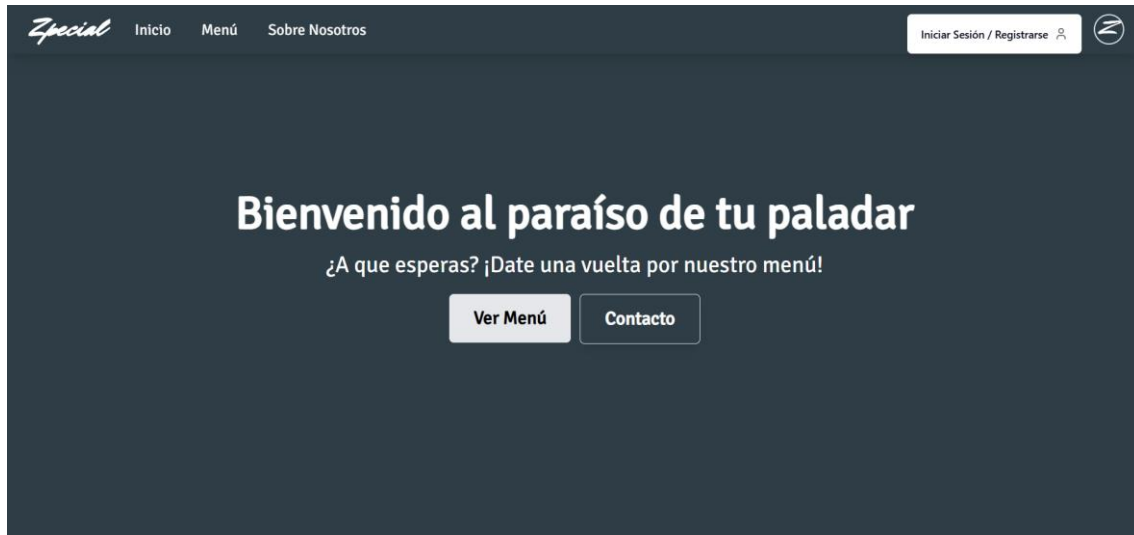
Panel de Cupones:



En este panel tendremos acceso a una lista y formulario para crear cupones de descuento que pueda tener nuestro negocio.

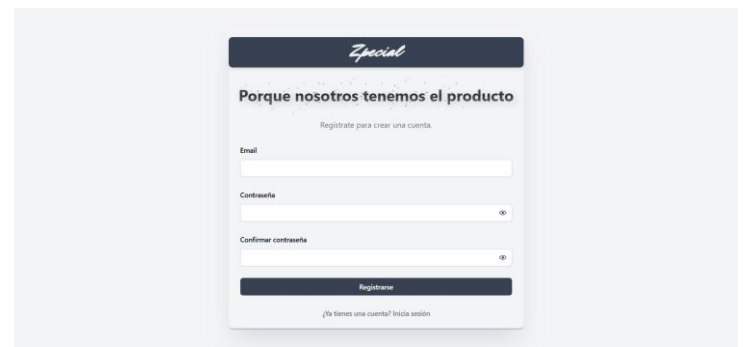
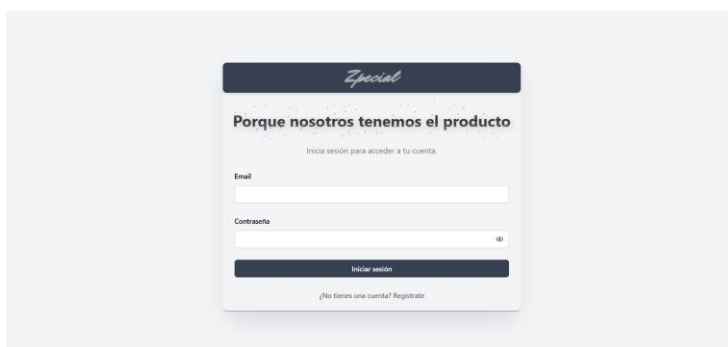
Manual de Usuario

Página principal:



Aquí podrás encontrar la Landing Page del comercio con información, horarios de apertura, contacto, historia y sobre qué podrás encontrar en él. También verás un dropdown en la parte superior derecha que te permitirá acceder o registrarte como nuevo usuario.

Login / Register:



Aquí podrás observar los formularios de registrarse e iniciar sesión, además en ellos se esconde un secreto, ¿Por qué no lo pruebas?

Perfil:

Special Inicio Menú

¡Hola!

Perfil

Mis pedidos

Cerrar Sesión

< Ajustes

Datos personales

Nombre

Apellidos

Teléfono

Dirección Postal

Datos de acceso

Usuario

Correo electrónico

Contraseña (Dejar en blanco para mantenerla)

cristobal.alcala.smr@gmail.com

En este apartado encontrarás tu espacio personal donde podrás tratar tus datos y modificarlos siempre que quieras y necesites.

Pedidos:

Special Inicio Menú

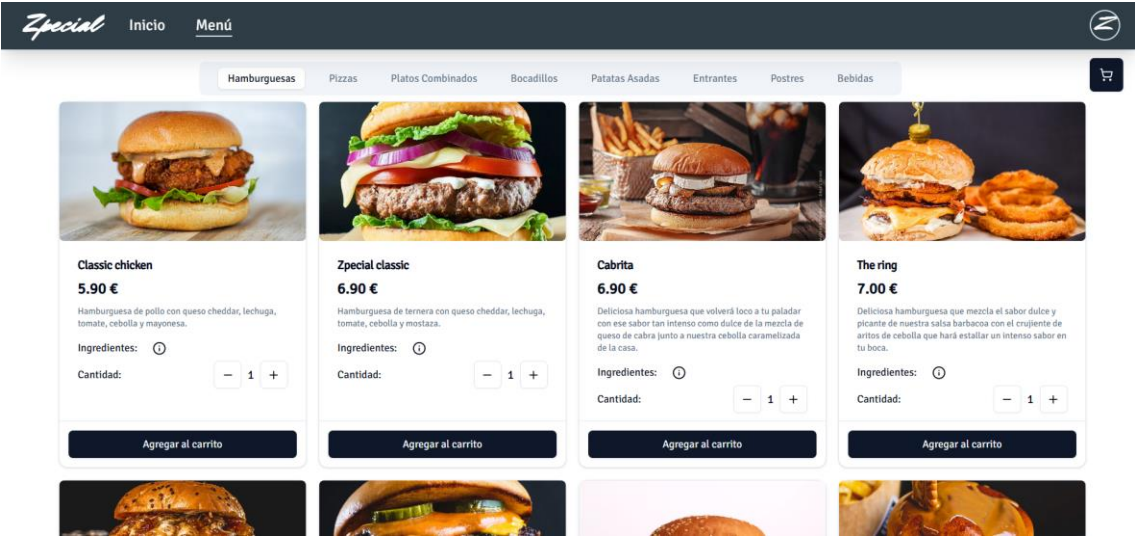
Pedidos (0)

Buscar...

Fecha de recibo	Estado	Subtotal	Total	Descuento	Productos	Tipo	Comentarios	Dirección	Acciones
No hay pedidos para mostrar									

En este apartado encontrarás tanto el historial de pedidos como el estado de tus pedidos activos en tiempo real.

Menú:



Este es el apartado de Menú, donde podrás hacer tus pedidos y disfrutar de la deliciosa comida.