

**Alumno:** Cristóbal Alcalá Cazorla

**Tutor:** Eladio Blanco López

2º DAW

*06/24*

[Especificaciones del proyecto 3](#_Toc168584718)

[Descripción del proyecto 3](#_Toc168584719)

[Funcionalidades del Sistema 3](#_Toc168584720)

[1. Administrador: 3](#_Toc168584721)

[2. Cliente: 4](#_Toc168584722)

[3. Recepcionista: 4](#_Toc168584723)

[4. Chef: 4](#_Toc168584724)

[5. Delivery (Repartidor): 5](#_Toc168584725)

[Flujo de Trabajo 5](#_Toc168584726)

[1. Registro y Pedido del Cliente: 5](#_Toc168584727)

[2. Pedido por el Recepcionista: 5](#_Toc168584728)

[3. Preparación del pedido por el Chef: 5](#_Toc168584729)

[4. Gestión del Recepcionista de pedidos completados: 6](#_Toc168584730)

[5. Entrega del pedido mediante Delivery: 6](#_Toc168584731)

[Beneficios del Sistema 6](#_Toc168584732)

[Elementos innovadores 7](#_Toc168584733)

[1. Librerías de Componentes Externas: 7](#_Toc168584734)

[2. Gestión de Fechas y Horarios: 7](#_Toc168584735)

[3. Transiciones: 7](#_Toc168584736)

[4. Iconos: 7](#_Toc168584737)

[5. Formularios: 8](#_Toc168584738)

[6. Comunicación en Tiempo Real: 8](#_Toc168584739)

[7. Notificaciones: 8](#_Toc168584740)

[8. Framework de Estilos: 8](#_Toc168584741)

[9. Validación de datos: 8](#_Toc168584742)

[10. Integración con Servicios en la Nube: 8](#_Toc168584743)

[11. Seguridad y Encriptación de Contraseñas: 9](#_Toc168584744)

[Estudio Previo y Prototipo 10](#_Toc168584745)

[Prototipo y visualización 10](#_Toc168584746)

[Documentación del Backend 13](#_Toc168584747)

[1. Lenguajes y Entorno: 13](#_Toc168584748)

[2. Framework y Middleware: 13](#_Toc168584749)

[3. Seguridad: 13](#_Toc168584750)

[4. Utilidades: 14](#_Toc168584751)

[5. Diseño de la base de datos: 14](#_Toc168584752)

[6. Documentación de todos los servicios CRUD: 15](#_Toc168584753)

[Documentación del Frontend 22](#_Toc168584754)

[1. Lenguajes y Tecnologías: 22](#_Toc168584755)

[2. Bibliotecas y Herramientas Adicionales: 22](#_Toc168584756)

[Propuestas de mejora 23](#_Toc168584757)

[Bibliografía 24](#_Toc168584758)

[Manuales 25](#_Toc168584759)

[Manual de Administrador 26](#_Toc168584760)

[Panel principal: 26](#_Toc168584761)

[Panel de Empleados y Usuarios: 26](#_Toc168584762)

[Formularios de Empleados y Usuarios: 27](#_Toc168584763)

[Panel de Productos: 28](#_Toc168584764)

[Formulario de Productos: 28](#_Toc168584765)

[Panel de Secciones: 29](#_Toc168584766)

[Panel de Pedidos: 29](#_Toc168584767)

[Panel de Cupones: 30](#_Toc168584768)

[Manual de Usuario 31](#_Toc168584769)

[Página principal: 31](#_Toc168584770)

[Login / Register: 31](#_Toc168584771)

[Perfil: 32](#_Toc168584772)

[Pedidos: 32](#_Toc168584773)

[Menú: 33](#_Toc168584774)

# 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

### 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.

### 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.

### 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.

### 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.

### 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

### 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.

### 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.

### 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”.

### 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).

### 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 librería 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:

### 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.

### 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.

### 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.

### 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.

### 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.

### 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.

### 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.

### 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.

### 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.

### 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.

### 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](https://www.hiopos.com/es/), 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.

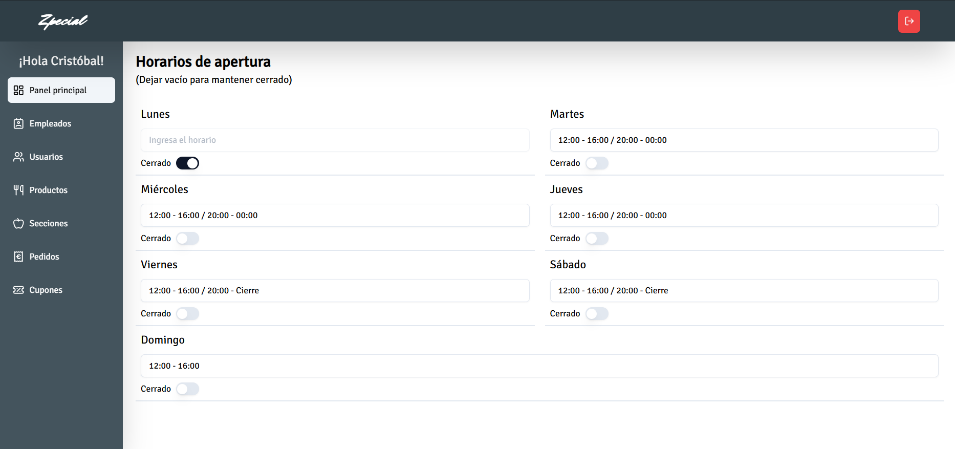
## Landing PagePrototipo y visualización

Landing Page

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Login



Administrador

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

Recepcionista

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Chef

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Delivery

# Documentación del Backend

### Lenguajes y Entorno:

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

### 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.

### Seguridad:

* **Bcrypt:** Biblioteca para la encriptación de contraseñas.

### 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.

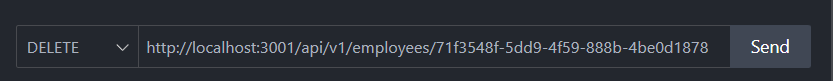
### Diseño de la base de datos:

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.

### Documentación de todos los servicios CRUD:

#### Texto Descripción generada automáticamenteTexto Descripción generada automáticamenteTexto Descripción generada automáticamenteEmpleados:

GET

**Captura de pantalla de un celular

Descripción generada automáticamenteCaptura de pantalla de computadora

Descripción generada automáticamente**

DELETE

PATCH

POST

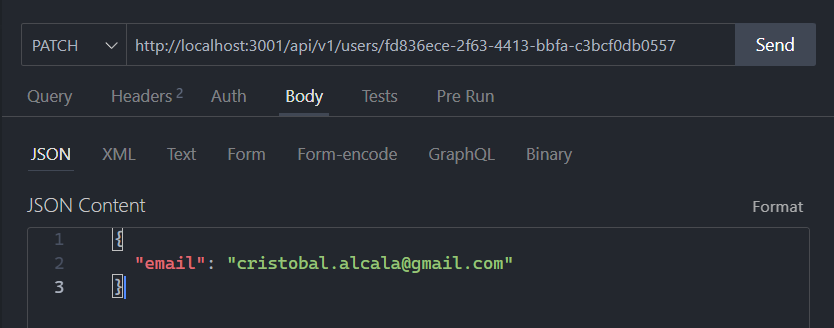
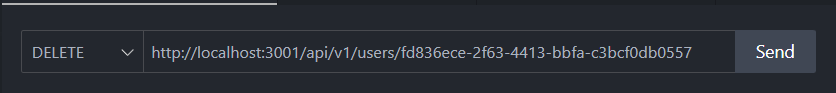
#### Usuarios:

**Texto

Descripción generada automáticamenteCaptura de pantalla de un celular

Descripción generada automáticamente**

GET

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

DELETE

PATCH

POST

#### Categorías:

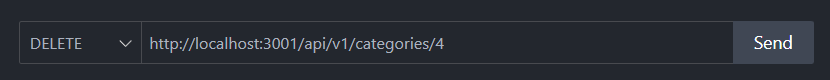
**Texto

Descripción generada automáticamente**Captura de pantalla de un celular

Descripción generada automáticamente

GET

GET

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteCaptura de pantalla de un celular

Descripción generada automáticamente

DELETE

PATCH

POST

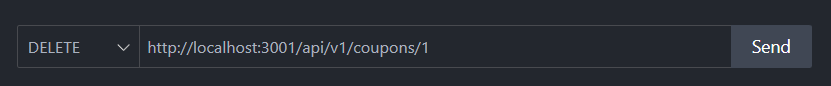
#### Cupones:

**Texto

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto

Descripción generada automáticamente**

GET

Captura de pantalla de un celular

Descripción generada automáticamenteCaptura de pantalla de un celular

Descripción generada automáticamente

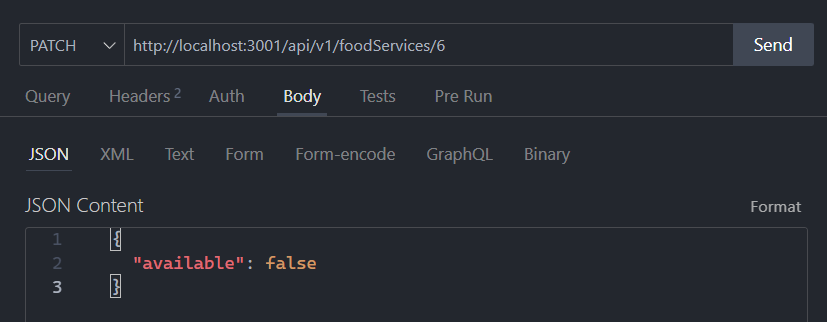
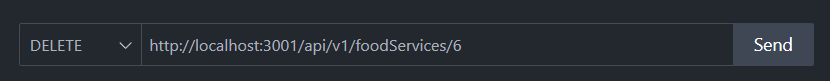
DELETE

PATCH

POST

#### Texto Descripción generada automáticamenteComidas:

GET

Texto

Descripción generada automáticamente

DELETE

PATCH

POST

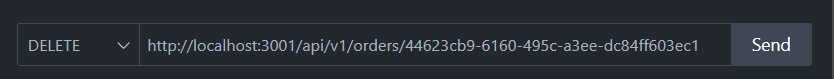
#### Pedidos:

Captura de pantalla de un celular

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto

Descripción generada automáticamente

GET

Texto

Descripción generada automáticamente

DELETE

POST

#### Captura de pantalla con letras Descripción generada automáticamenteInterfaz de usuario gráfica, Texto Descripción generada automáticamenteHorarios:

PATCH

GET

# Documentación del Frontend

### 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.

### 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**](https://ui.shadcn.com/)
* [**Aceternity**](https://ui.aceternity.com/)
* [**v0**](https://v0.dev/)
* [**React Hook Form**](https://react-hook-form.com/)
* [**Framer Motion**](https://www.framer.com/motion/)
* [**Socket.io**](https://socket.io/)
* [**TailwindCSS**](https://tailwindcss.com/)

# Manuales

## Manual de Administrador

### Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico Descripción generada automáticamentePanel principal:

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:

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:

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…).

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

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:

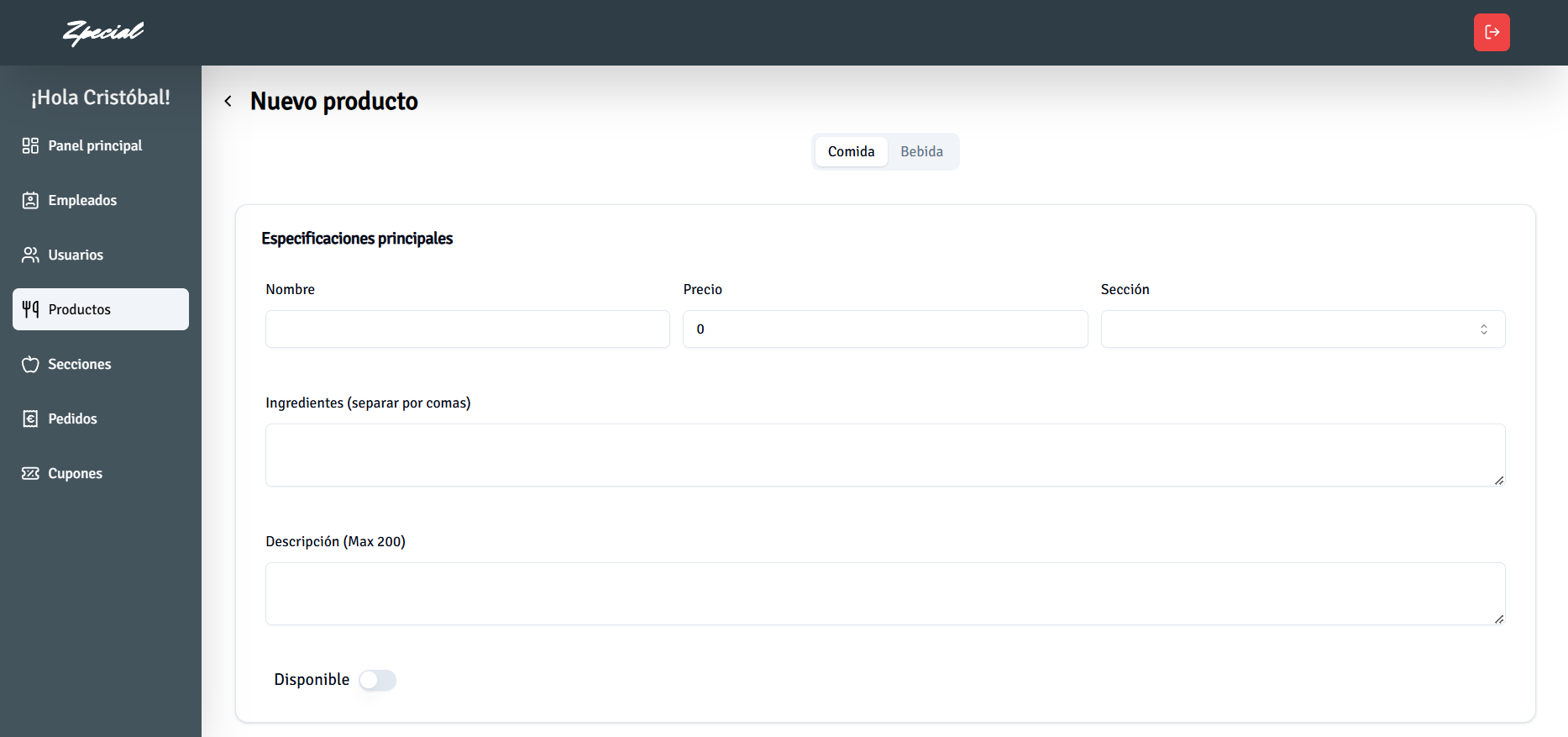
Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

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:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

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:

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.

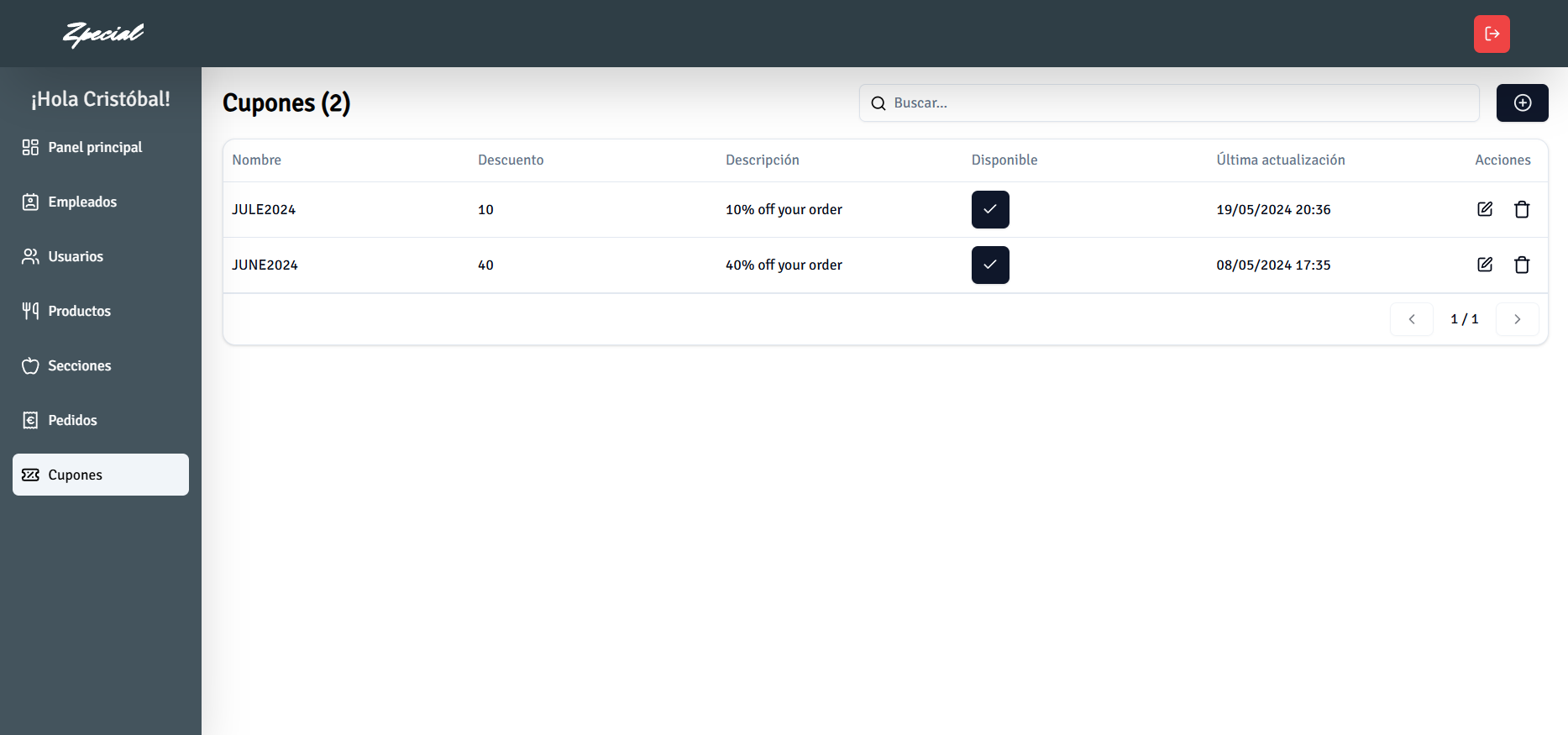
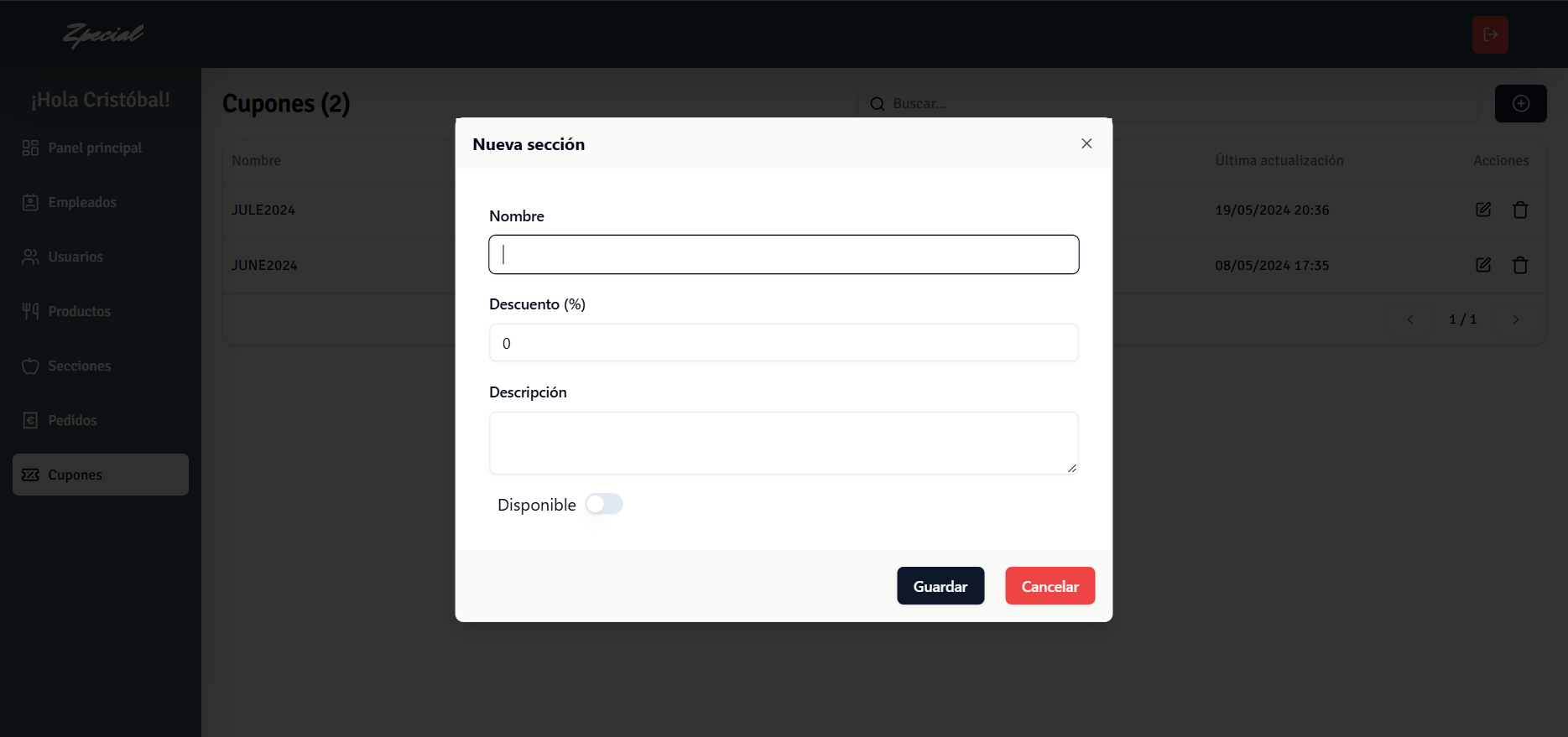
### Panel de Pedidos:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

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:

Interfaz de usuario gráfica

Descripción generada automáticamente

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:

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

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

### Captura de pantalla de un celular Descripción generada automáticamentePerfil:

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

### Pedidos:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

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

### Menú:

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

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