

Trabajo Final- Food Store

Información del Proyecto

Nombre: Food Store - Sistema de Gestión de Pedidos de Comida

Tecnologías: TypeScript, Vite, HTML5, CSS3

Backend: Spring Boot REST API

Autenticación: Sin módulo de seguridad (gestión básica con localStorage)

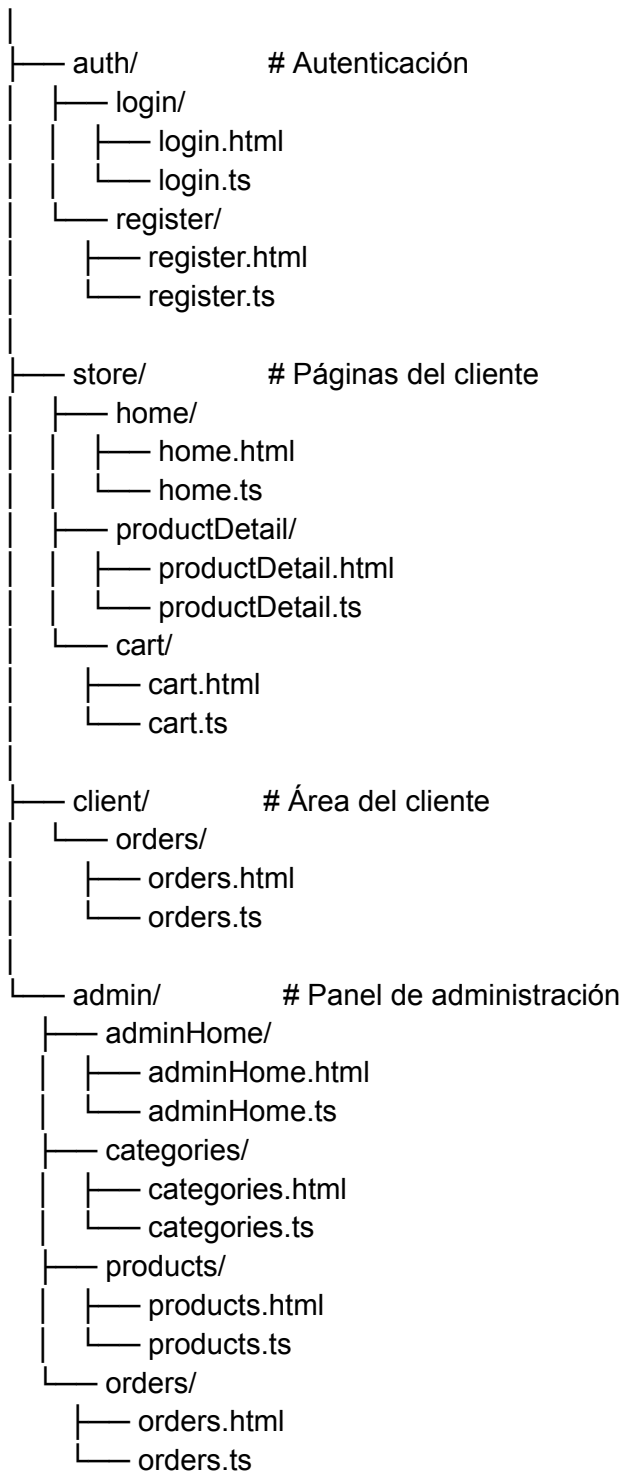
Objetivos del Proyecto

Desarrollar una aplicación web completa para la gestión de un negocio de comidas, que permita:

1. **A los administradores:** Gestionar categorías, productos y pedidos
2. **A los clientes:** Navegar productos, realizar compras y seguir sus pedidos
3. **Sistema de carrito:** Funcional con persistencia en localStorage
4. **Integración con API:** Conexión completa con backend Spring Boot





Estructura del Proyecto

```
final-prog3/
├── index.html           # Redirección a login
├── package.json         # Dependencias y scripts
├── tsconfig.json       # Configuración TypeScript
├── vite.config.ts      # Configuración Vite
├── src/
│   ├── main.ts         # Punto de entrada (vacío)
│   ├── style.css       # Estilos globales
│   ├── types/          # Definiciones de tipos TypeScript
│   │   ├── IUser.ts   # Interface de Usuario
│   │   ├── ICategoria.ts # Interface de Categoría
│   │   ├── IProduct.ts # Interface de Producto
│   │   ├── ICart.ts    # Interface de Carrito
│   │   └── IOrders.ts  # Interface de Pedidos
│   ├── utils/          # Utilidades y helpers
│   │   ├── api.ts      # Funciones de conexión con API
│   │   ├── auth.ts     # Gestión de autenticación
│   │   ├── cart.ts     # Lógica del carrito
│   │   └── navigate.ts  # Navegación entre páginas
│   └── pages/          # Páginas de la aplicación
```













Roles y Permisos

Administrador (Admin)

-  Acceso completo al panel de administración
-  Gestión de categorías (CRUD)
-  Gestión de productos (CRUD)
-  Gestión de pedidos (ver todos, actualizar estado)

Cliente

-  Ver catálogo de productos
-  Filtrar por categorías
-  Buscar productos
-  Ver detalle de productos
-  Agregar productos al carrito
-  Gestionar carrito (agregar, quitar, modificar cantidades)
-  Realizar pedidos
-  Ver historial de pedidos propios
-  Ver detalle y estado de sus pedidos
-  NO tiene acceso al panel de administración

Sistema de Autenticación

Flujo de Autenticación

1. **Login/Registro:**
 - Usuario ingresa credenciales
 - Sistema valida con API Spring Boot
 - Si es exitoso, guarda datos en localStorage
 - Redirecciona según el rol
2. **Validación de Sesión:**
 - Cada página protegida verifica localStorage
 - Si no hay sesión, redirecciona a login
 - Valida permisos según el rol
3. **Datos Almacenados en localStorage:**

```
{  
  "name": "Juan Pérez",  
  "email": "juan@example.com",  
  "id": 1,  
  "role": "admin" | "cliente"  
}
```

4. Cierre de Sesión:

- Limpia localStorage
- Redirecciona al login

Funcionalidades por Módulo

1. Módulo de Autenticación

Login (/src/pages/auth/login/)

- **Funcionalidad:**
 - Formulario con email y contraseña
 - Validación de campos requeridos
 - Conexión con API
 - Manejo de errores de autenticación
 - Redirección según rol del usuario

Registro (/src/pages/auth/register/)

- **Funcionalidad:**
 - Formulario con nombre, email y contraseña
 - Validación de campos (email válido, contraseña mínimo 6 caracteres)
 - Conexión con API
 - Solo se pueden registrar clientes
 - Auto-login después del registro

2. Módulo de Cliente - Store

Home / Catálogo (/src/pages/store/home/)

Características:

- Sidebar con categorías
- Búsqueda en tiempo real
- Filtros:
 - Por categoría
 - Ordenamiento (nombre A-Z, Z-A, precio ascendente/descendente)
- Grid de productos con:
 - Imagen
 - Nombre
 - Descripción
 - Precio
 - Badge de disponibilidad
 - Click para ir al detalle
- Badge del carrito con cantidad de ítems
- Contador de productos encontrados
- Toggle del sidebar para mobile

Detalle de Producto (/src/pages/store/productDetail/)

Características:

- Imagen grande del producto
- Información completa:
 - Nombre
 - Descripción
 - Precio
 - Stock disponible
 - Estado (disponible/no disponible)
- Selector de cantidad con validación de stock
- Botón "Agregar al Carrito"
- Mensaje de confirmación
- Botón "Volver"
- Validaciones:
 - No permite agregar si no hay stock
 - No permite cantidad mayor al stock disponible
 - No permite agregar productos inactivos

Carrito de Compras (/src/pages/store/cart/)

Características:

- Lista de productos en el carrito con:
 - Imagen
 - Nombre
 - Precio unitario
 - Controles de cantidad (+/-)
 - Precio total por producto
 - Botón eliminar
- Resumen del pedido:
 - Subtotal
 - Costo de envío (fijo: \$500)
 - Total
- Botones:
 - "Proceder al Pago"
 - "Vaciar Carrito"
- Modal de checkout con formulario:
 - Teléfono (requerido)
 - Dirección (requerida)
 - Método de pago (efectivo/tarjeta/transferencia)
 - Notas adicionales (opcional)

- Validaciones:
 - Stock disponible al modificar cantidad
 - Campos requeridos en checkout
- Estado vacío con mensaje y botón a la tienda

Persistencia:

- Carrito guardado en localStorage
- Persiste entre sesiones
- Se limpia al confirmar pedido

3. Módulo de Cliente - Mis Pedidos

Historial de Pedidos (/src/pages/client/orders/)

Características:

- Lista de pedidos del usuario autenticado
- Tarjetas de pedido mostrando:
 - Número de pedido
 - Fecha y hora
 - Estado con badge de color
 - Resumen de productos (primeros 3 + contador)
 - Total del pedido
- Click en pedido abre modal con detalle completo
- Modal de detalle muestra:
 - Estado con icono visual
 - Información de entrega
 - Lista completa de productos
 - Desglose de costos (subtotal, envío, total)
 - Mensaje según estado del pedido
- Estado vacío si no hay pedidos

Estados de Pedido:

- pending: 🕒 Pendiente
- processing: 👨‍🍳 En Preparación
- completed: ✅ Entregado
- cancelled: ❌ Cancelado

4. Módulo de Administración

Dashboard (/src/pages/admin/adminHome/)

Características:

- Sidebar de navegación administrativa
- 4 tarjetas con estadísticas:
 - Total de categorías
 - Total de productos
 - Total de pedidos
 - Productos disponibles
- Panel de resumen detallado:
 - Categorías activas
 - Productos activos/inactivos
 - Pedidos por estado
- Enlaces directos a cada módulo

Gestión de Categorías (/src/pages/admin/categories/)

Características:

- Tabla con todas las categorías:
 - ID
 - Imagen (thumbnail)
 - Nombre
 - Descripción
 - Acciones (Editar/Eliminar)
- Botón "Nueva Categoría"
- Modal para crear/editar:
 - Nombre (requerido)
 - Descripción (requerida)
 - URL de imagen (requerida)
- Validaciones:
 - Campos requeridos
 - URL válida para imagen
- Confirmación al eliminar

Gestión de Productos (/src/pages/admin/products/)

Características:

- Tabla con todos los productos:
 - ID
 - Imagen (thumbnail)
 - Nombre
 - Descripción
 - Precio
 - Categoría (nombre)
 - Stock
 - Estado (disponible/no disponible)
 - Acciones (Editar/Eliminar)
- Botón "Nuevo Producto"
- Modal para crear/editar:
 - Nombre (requerido)
 - Descripción (requerida)
 - Precio (requerido, > 0)
 - Stock (requerido, >= 0)
 - Categoría (select con categorías disponibles)
 - URL de imagen (requerida)
 - Checkbox "Producto disponible"
- Validaciones:
 - Todos los campos requeridos
 - Precio y stock numéricos válidos
 - Categoría debe existir
- Confirmación al eliminar

Gestión de Pedidos Admin (/src/pages/admin/orders/)

Características:

- Filtro por estado de pedido
- Lista de TODOS los pedidos (no solo del usuario)
- Tarjetas de pedido mostrando:
 - Número de pedido
 - Nombre del cliente
 - Fecha y hora
 - Estado con badge
 - Cantidad de productos
 - Total
- Click en pedido abre modal con detalle
- Modal muestra:
 - Información del cliente
 - Dirección y teléfono
 - Método de pago
 - Notas del cliente

- Lista completa de productos
- Desglose de costos
- **Select para cambiar estado del pedido**
- Botón "Actualizar Estado"
- Ordenados por fecha (más recientes primero)

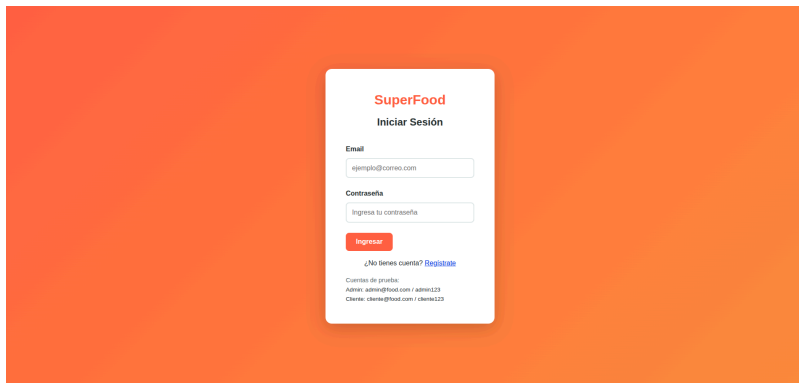
Diseño y UX

Paleta de Colores (brindados por el profesor, ustedes pueden elegir otros)

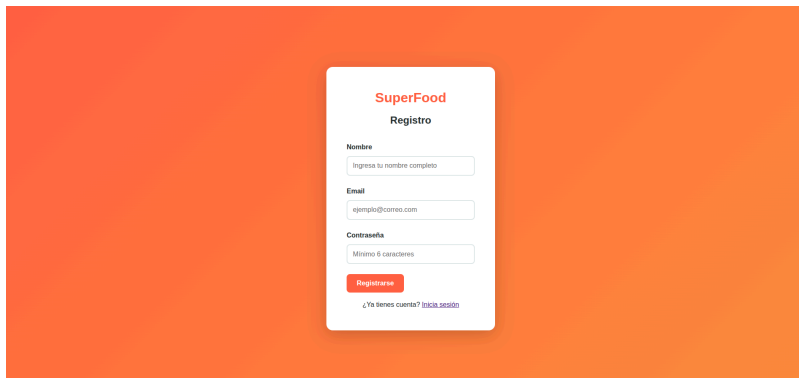
```
--primary: #ff6b35    /* Naranja principal */
--secondary: #f7931e  /* Naranja secundario */
--dark: #2d3436       /* Texto oscuro */
--light: #f5f6fa      /* Fondo claro */
--success: #00b894    /* Verde éxito */
--danger: #d63031     /* Rojo peligro */
--gray: #dfe6e9       /* Gris borders */
```

Pantallas de referencia

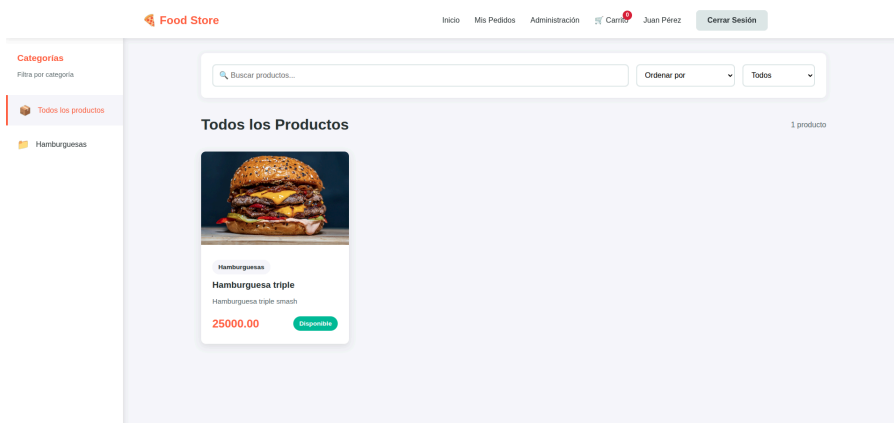
Login: las credenciales no hace falta que se vean
Inicio de sesión: No es necesario que las credenciales sean visibles.



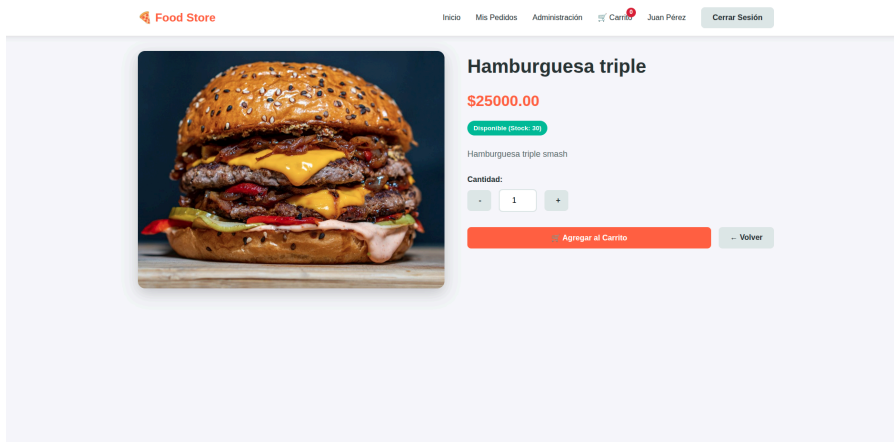
Registro:



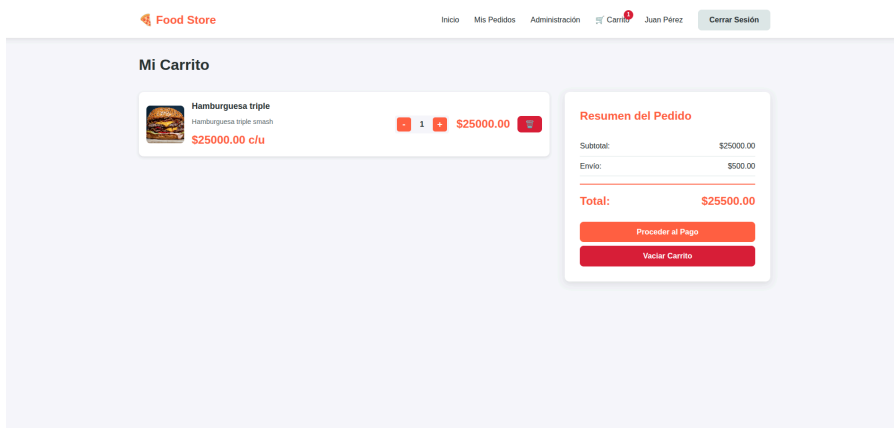
Vista home store



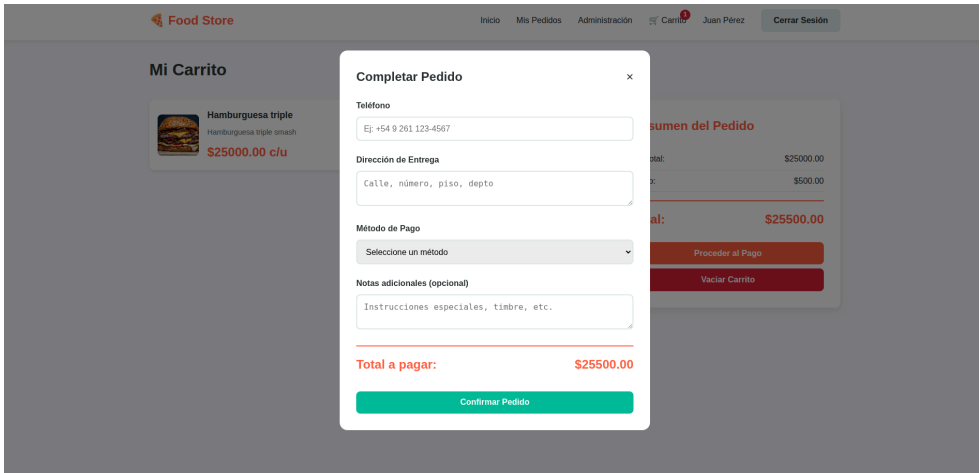
Vista detalle de producto



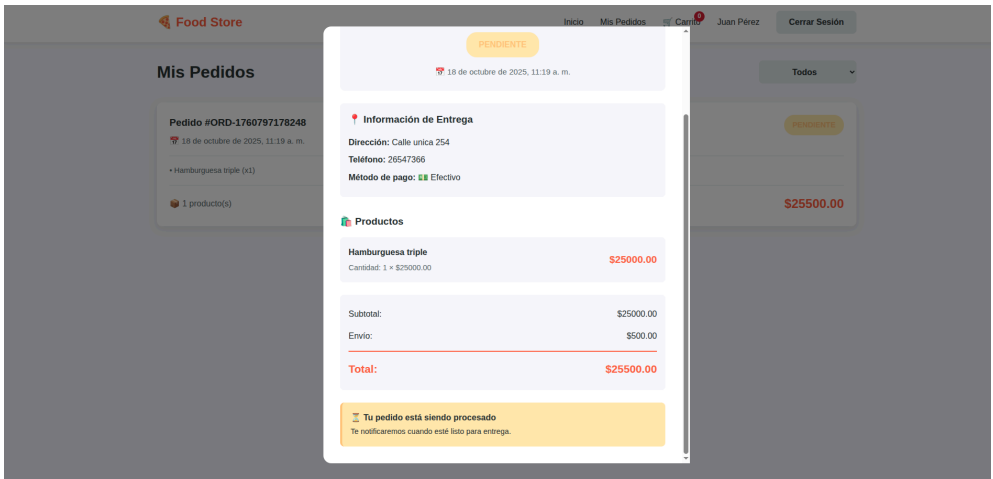
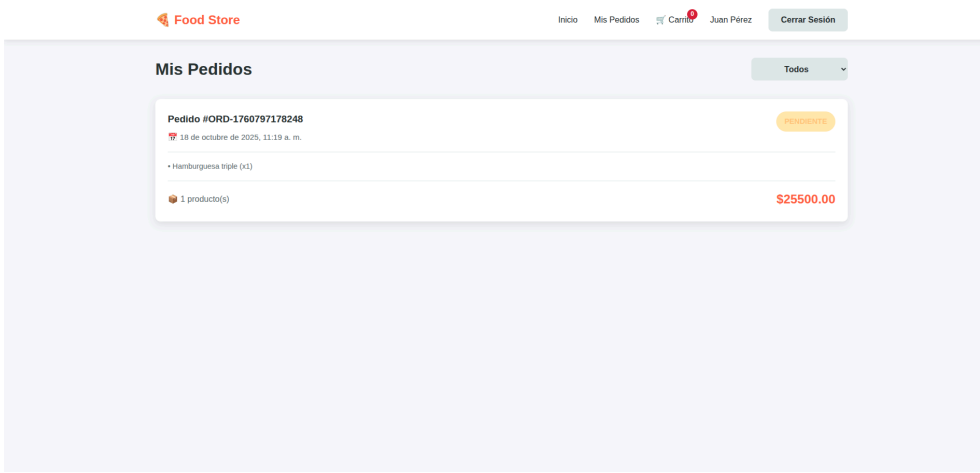
Vista de carrito



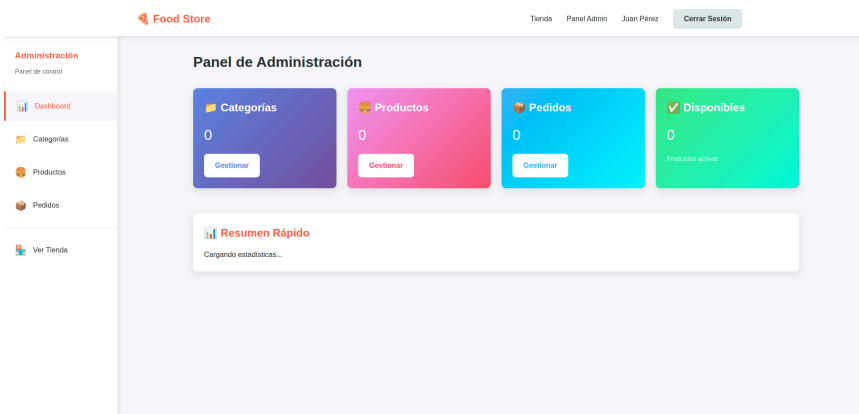
confirmación de pedido:



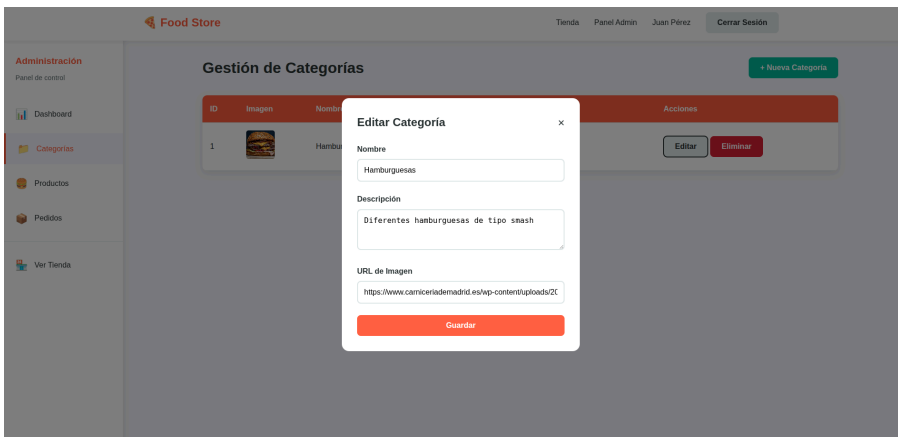
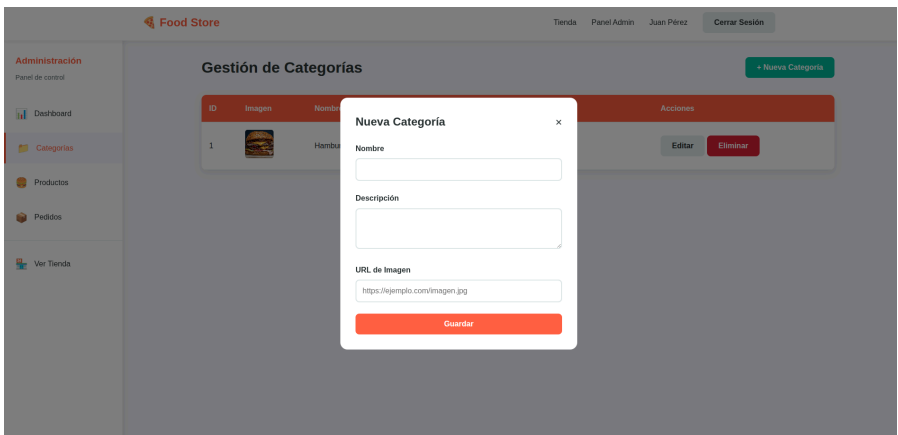
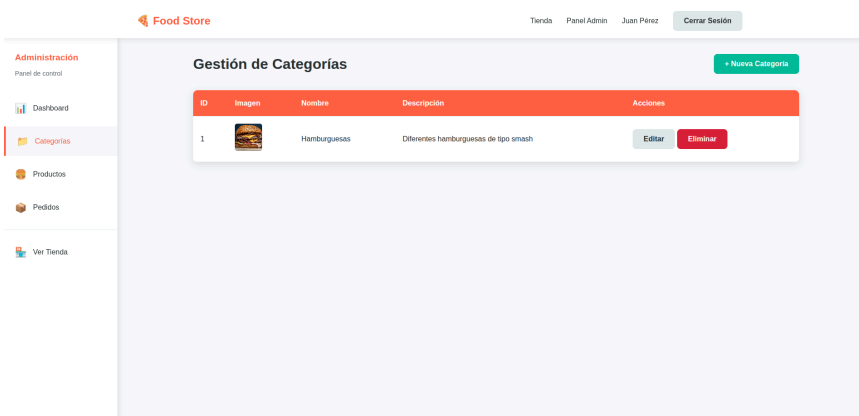
pedidos:



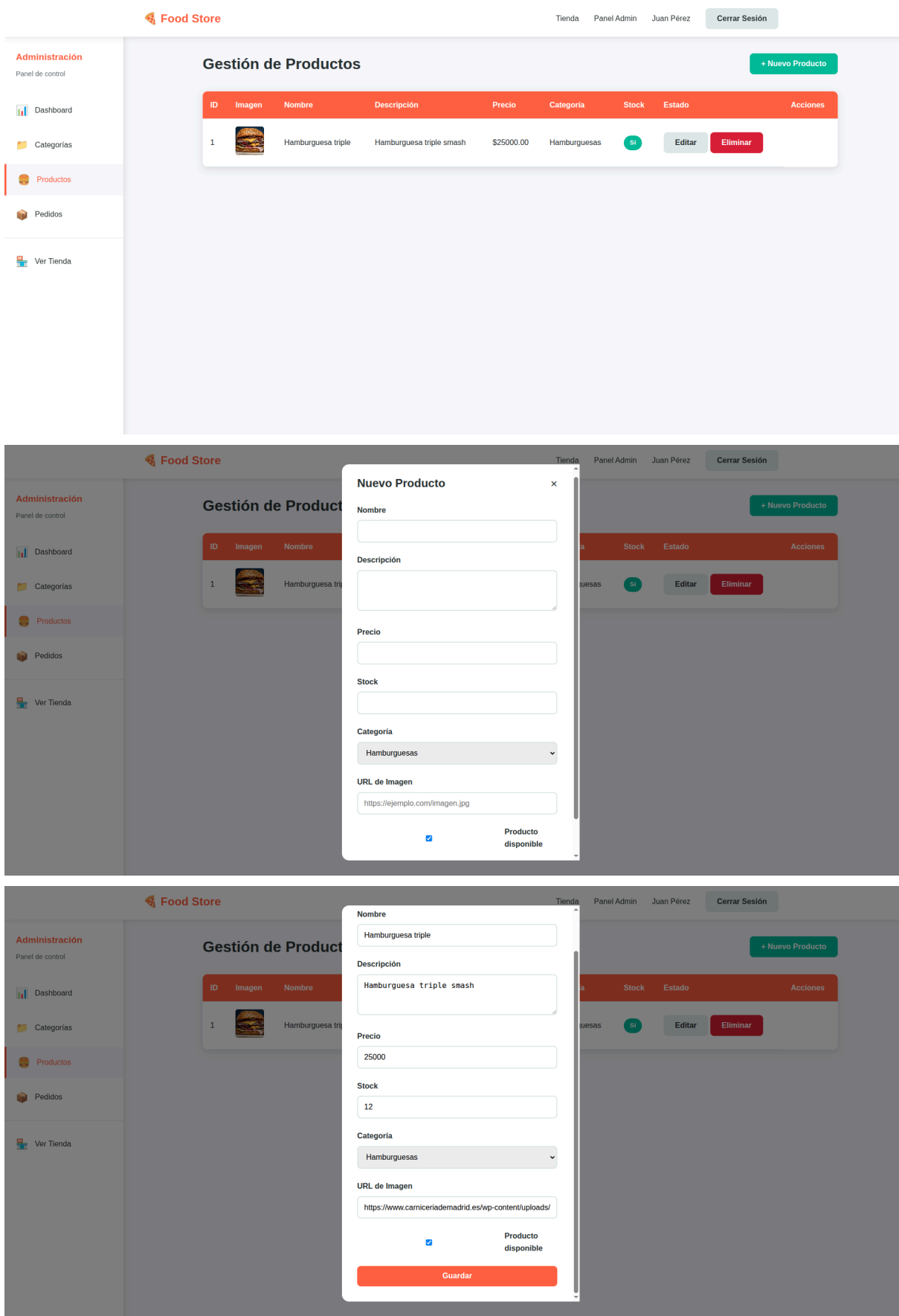
Home admin



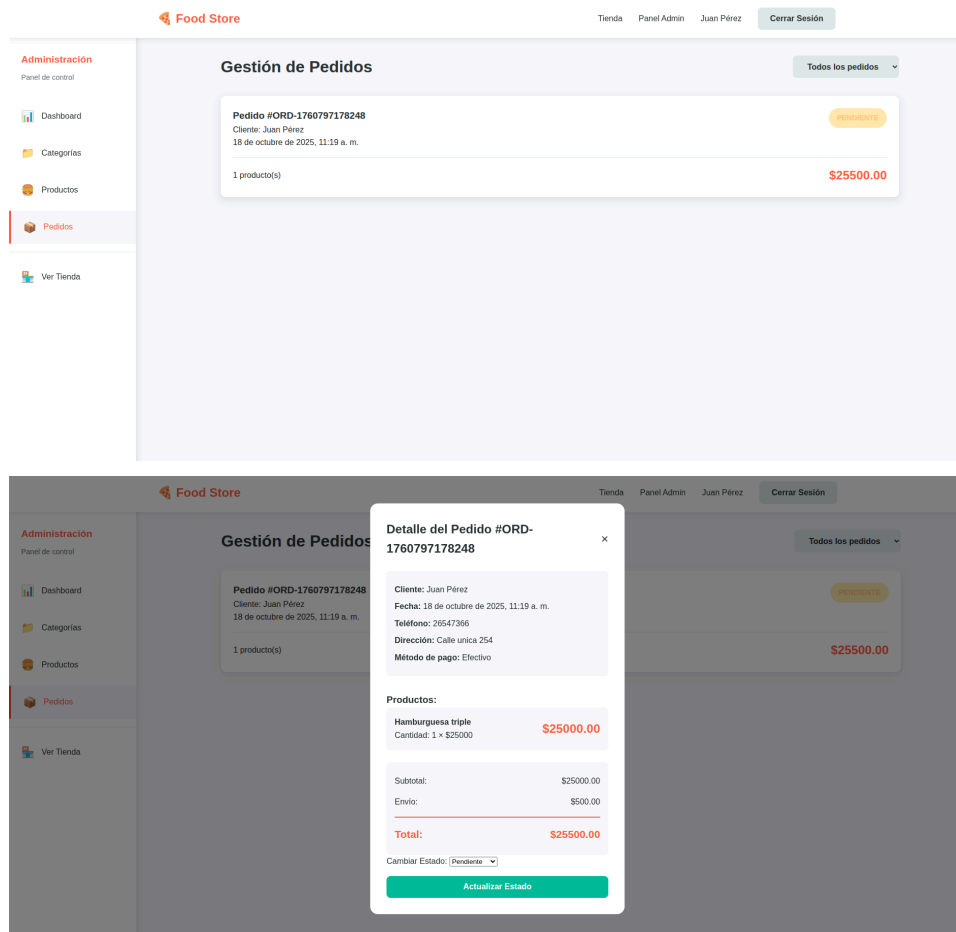
Crud Categorías



Crud Productos



Pedidos



Componentes Reutilizables (se repiten secciones en varias páginas)

- Navbar:**
 - Logo con link al home
 - Links de navegación
 - Badge del carrito
 - Nombre del usuario
 - Botón logout
- Sidebar:**
 - Versión cliente: Categorías
 - Versión admin: Módulos administrativos
 - Responsive (oculto en mobile)
 - Toggle button
- Cards:**
 - Productos
 - Pedidos
 - Estadísticas
- Modales:**
 - Formularios de creación/edición
 - Detalle de pedidos
 - Checkout

5. Badges:

- Estados de pedido
- Disponibilidad de productos
- Contador del carrito
-

Flujos de Usuario

Flujo de Compra (Cliente)

1. Usuario se autentica
2. Navega por el catálogo (puede filtrar/buscar)
3. Click en producto → Ver detalle
4. Selecciona cantidad → Agregar al carrito
5. Continúa comprando o va al carrito
6. En el carrito: revisa productos, modifica cantidades
7. Click "Proceder al Pago"
8. Completa formulario de checkout
9. Confirma pedido
10. Ver mensaje de éxito
11. Es redirigido a "Mis Pedidos"
12. Puede ver estado y detalle del pedido

Flujo de Gestión de Producto (Admin)

1. Admin se autentica
2. Va a "Panel Admin" → "Productos"
3. Click en "Nuevo Producto"
4. Completa formulario
5. Selecciona categoría del dropdown
6. Marca checkbox si está disponible
7. Guarda producto
8. Ve el producto en la tabla
9. Puede editar/eliminar cuando necesite

Flujo de Gestión de Pedido (Admin)

1. Admin va a "Pedidos"
2. Ver lista de todos los pedidos
3. Puede filtrar por estado
4. Click en pedido → Ver detalle completo
5. Cambia estado en el select
6. Click "Actualizar Estado"
7. Cliente ve el cambio en "Mis Pedidos"

Tecnologías y Herramientas

Frontend

- **TypeScript:** Tipado estático
- **Vite:** Build tool y dev server
- **HTML5:** Estructura
- **CSS3:** Estilos (sin frameworks)
- **localStorage:** Persistencia de carrito y sesión

Utilidades





- **Fetch API:** Peticiones HTTP
- **ES Modules:** Sistema de módulos
- **Async/Await:** Manejo de promesas

Desarrollo






- **pnpm | npm | yarn:** Gestor de paquetes
- **Vite + typescript :** Desarrollo

Puntos de Evaluación

1. Estructura y Organización (15%)

-  Estructura de carpetas clara y lógica
-  Separación de concerns (tipos, utils, pages)
-  Nomenclatura consistente
-  Código modular y reutilizable

2. TypeScript (20%)

-  Uso correcto de interfaces
-  Tipado fuerte en todas las funciones
-  Evitar any
-  Uso de tipos genéricos donde corresponda
-  Manejo de opcionales y null checks

3. Funcionalidad (30%)

- ☒ Todas las funcionalidades CRUD funcionan
- ☒ Autenticación y autorización correcta
- ☒ Navegación entre páginas funcional
- ☒ Carrito persiste correctamente
- ☒ Validaciones en formularios
- ☒ Manejo de errores apropiado

4. Integración con API (20%)

- ☒ Todas las peticiones HTTP funcionan
- ☒ Manejo correcto de respuestas y errores
- ☒ Formato JSON correcto en requests
- ☒ Parsing correcto de responses

5. UI/UX y Diseño (15%)

- ☒ Interfaz intuitiva
- ☒ Feedback visual al usuario
- ☒ Estados de carga y error
- ☒ Consistencia visual



Consideraciones Importantes

Seguridad

⚠ IMPORTANTE: Este proyecto NO implementa seguridad real:

- Las contraseñas se envían en texto plano
- No hay tokens JWT
- La validación de rol es solo frontend
- localStorage es fácilmente modificable
- **SOLO para fines educativos**

Datos de Prueba

// Usuarios de prueba (configurar en backend)

Admin:

email: admin@food.com

password: admin123

Cliente:

email: cliente@food.com

password: cliente123

Mejoras Futuras

- Implementar JWT para autenticación real
- Agregar validación de email
- Sistema de notificaciones en tiempo real
- Paginación en listados
- Búsqueda avanzada con filtros combinados
- Sistema de favoritos
- Valoraciones y comentarios de productos
- Historial de compras con recompra rápida
- Panel de estadísticas con gráficos