



**"AÑO DE LA RECUPERACIÓN Y CONSOLIDACIÓN DE LA
ECONOMÍA PERUANA"**

PROYECTO:

"Sistema de Gestión de Pedidos para Hamburguesería Piccola"

INTEGRANTES:

Jean Pier David Vega Choque
Carlos David Laynez Peña

PROFESOR:

YNGVAR MERINO

CURSO: INTEGRADOR I

PERÚ-ICA

2025

1. Introducción	3
2. Planteamiento del Problema	3
3. Objetivos del Proyecto	3
3.1 Objetivo General	3
3.2 Objetivos Específicos	3
4. Justificación	4
5. Marco Teórico	4
6. Metodología del Proyecto	4
6.1 Lean Canvas	4
6.2 Planificación con PMBOK o	5
6.3 Requerimientos del Sistema	6
7. Diseño del Proyecto	7
7.1 Modelado de Procesos (BPM) o	
7.2 Diseño de Base de Datos o	
7.3 UX/UI, Reportes y Documentación	
8. Desarrollo del Proyecto o	
8.1 Arquitectura (MVC, DAO, SOLID, TDD) o	
8.2 Librerías y Recursos en Java o	
8.3 Control de Versiones (Git & GitHub)	
9. Pruebas y Calidad o	
9.1 Pruebas de Software o	
9.2 Pruebas de Seguridad	
10. Despliegue de la Solución	
11. Monitoreo y Mantenimiento	
12. Resultados y Discusión	
13. Conclusiones	
14. Recomendaciones	

15. Bibliografía

1. INTRODUCCIÓN

La **Hamburguesería Piccola** es un establecimiento gastronómico ubicado en Ica, especializado en la preparación y venta de hamburguesas gourmet y alimentos afines. A pesar de contar con una clientela fiel y un producto de calidad, el negocio enfrenta importantes desafíos debido a la gestión manual de sus operaciones diarias.

Actualmente, el registro de pedidos se realiza mediante libretas físicas, el control de inventario es inexistente y la generación de reportes depende completamente de cálculos manuales. Esta falta de digitalización genera errores frecuentes en los pedidos, retrasos en la atención al cliente y dificultades para analizar el desempeño del negocio.

Ante esta problemática, se propone el desarrollo de un **sistema de gestión integral** que digitalice y optimice todos los procesos operativos. El sistema estará enfocado no solo en resolver los problemas inmediatos de gestión, sino también en proporcionar una base tecnológica sólida que impulse el crecimiento y la profesionalización de la hamburguesería.

2. PLANTEAMIENTO DEL PROBLEMA

Los principales problemas identificados en la Hamburguesería Piccola son:

- **Gestión manual e ineficiente de pedidos** mediante libretas físicas
- **Falta de control de inventario** en tiempo real
- **Tiempos prolongados de atención** al cliente
- **Errores frecuentes** en la toma y transmisión de pedidos
- **Imposibilidad de generar reportes** automáticos de ventas
- **Dificultad para analizar** tendencias y comportamiento del cliente

3. OBJETIVOS DEL PROYECTO

3.1 OBJETIVO GENERAL

Desarrollar e implementar un sistema de gestión integral para la Hamburguesería Piccola que permita gestionar de manera eficiente el registro de pedidos, control de inventario y análisis de ventas, optimizando la toma de decisiones y fortaleciendo la administración del negocio.

3.2 OBJETIVOS ESPECÍFICOS

- Diseñar una base de datos que permita almacenar y organizar la información de productos, pedidos y clientes de manera estructurada y segura
- Implementar un módulo de control de inventario que registre las entradas y salidas de productos en tiempo real
- Automatizar el proceso de registro de pedidos, reduciendo errores humanos y mejorando la eficiencia operativa
- Aplicar buenas prácticas de programación y metodologías de desarrollo de software (MVC, SOLID, Git) para garantizar la calidad del sistema
- Facilitar la toma de decisiones mediante la recopilación y análisis de información útil para la planificación estratégica del negocio

4. JUSTIFICACIÓN

Este proyecto es necesario no solo porque representa una mejora significativa en la gestión actual de la Hamburguesería Piccola, sino también porque constituye una base sólida para el crecimiento y la sostenibilidad del negocio. Al implementar un sistema de gestión integral, la empresa podrá:

Optimizar procesos operativos reduciendo tiempos de atención en un 60% y eliminando el 95% de errores en pedidos. **Mejorar el control financiero** mediante reportes automáticos de ventas y análisis de rentabilidad por producto. **Fortalecer la experiencia del cliente** con servicio más ágil y personalizado. **Sentar las bases tecnológicas** para futuras expansiones, como servicio de delivery online o app móvil.

La inversión en este sistema se justifica por el retorno esperado en eficiencia, reducción de pérdidas y capacidad de crecimiento estratégico.

5. MARCO TEÓRICO

- **Sistemas de información para restaurantes** y establecimientos de comida rápida
- **Gestión de inventarios** en el sector gastronómico
- **Metodologías ágiles** de desarrollo de software (Scrum, Kanban)
- **Patrones de arquitectura** MVC y DAO para aplicaciones empresariales
- **Bases de datos relacionales** con PostgreSQL
- **Framework Spring Boot** para desarrollo web empresarial
- **Control de versiones** con Git y GitHub
- **Principios SOLID** y clean code para desarrollo mantenable

6. METODOLOGÍA DEL PROYECTO

6.1 LEAN CANVAS

PROBLEMAS

- Gestión manual de pedidos propensa a errores
- Falta de control de inventario en tiempo real
- Tiempos prolongados de atención al cliente
- Imposibilidad de análisis de datos de ventas
- Dificultad para tomar decisiones estratégicas

SOLUCIÓN

Sistema web especializado con:

- Registro digital de pedidos en tiempo real
- Control automático de inventario
- Generación de reportes automáticos
- Dashboard con métricas clave
- Interfaz intuitiva para personal no técnico

PROPUESTA DE VALOR ÚNICA

"Sistema especializado para hamburgueserías que combina simplicidad de uso con funcionalidades avanzadas de gestión"

SEGMENTOS DE CLIENTES

- Personal de atención (cajeros)
- Administradores del local
- Propietarios de la hamburguesería

CANALES

- Implementación presencial en el local
- Capacitación in-situ al personal
- Soporte técnico remoto vía WhatsApp/email

MÉTRICAS CLAVE

- Reducción del 60% en tiempos de atención
- Eliminación del 95% de errores en pedidos
- Reducción del 30% en mermas de inventario
- Incremento del 20% en ventas por eficiencia

ESTRUCTURA DE COSTOS

- Desarrollo: S/ 0 (interno)
- Hosting y dominio: S/ 80 mensual
- Mantenimiento: S/ 100 mensual

FLUJO DE INGRESOS

- Ahorro por reducción de errores: S/ 500/mes
- Ahorro por mejor gestión de inventario: S/ 300/mes
- Incremento de ventas: S/ 1,000/mes



6.2 PLANIFICACIÓN CON PMBOK

ACTA DE CONSTITUCIÓN DEL PROYECTO

COMPONENTE	DESCRIPCIÓN
Título del proyecto	Sistema de Gestión Integral para Hamburguesería Piccola
Gerente general del proyecto	Jean Pier David Vega Choque
Breve descripción del escenario	<ul style="list-style-type: none"> • La Hamburguesería Piccola gestiona sus pedidos de forma manual mediante libretas físicas, generando errores, retrasos en la atención y falta de control sobre el inventario. Se requiere un sistema digital que optimice los procesos operativos y proporcione herramientas de análisis.
Breve descripción del producto	<ul style="list-style-type: none"> • Sistema web desarrollado en Spring Boot con PostgreSQL que gestiona pedidos en tiempo real, controla automáticamente el inventario, registra clientes y genera reportes automáticos. Interfaz intuitiva adaptada al flujo de trabajo de una hamburguesería.
Justificación del proyecto	<ul style="list-style-type: none"> • Digitalizar procesos manuales para reducir errores en 95%, optimizar tiempos de atención en 60%, proporcionar control en tiempo real del inventario y facilitar la toma de decisiones basada en datos analíticos.
Patrocinador del proyecto	<ul style="list-style-type: none"> • Propietario de Hamburguesería Piccola
Autores involucrados	<ul style="list-style-type: none"> • Jean Pier David Vega Choque <ul style="list-style-type: none"> • Desarrollador Full-Stack • Diseñador UX/UI • Administrador de Base de Datos

- CRONOGRAMA DEL PROYECTO

• ID	• Fase	• Duración	• Inicio	• Final	• Responsable	• Descripción
• 1	• Proyecto Completo	• 15 semanas	• 1	• 15	• Jean Pier Vega	• Implementación total del sistema
• 2	• Inicio del Proyecto	• 1 semana	• 1	• 2	• Jean Pier Vega	• Identificación de stakeholders y planificación
• 3	• Análisis de Requerimientos	• 2 semanas	• 2	• 4	• Jean Pier Vega	• Investigación de procesos y definición de requerimientos
• 4	• Diseño del Sistema	• 2 semanas	• 4	• 6	• Jean Pier Vega	• Arquitectura, BD y prototipos de interfaz
• 5	• Desarrollo Backend	• 3 semanas	• 6	• 9	• Jean Pier Vega	• API REST, servicios y lógica de negocio
• 6	• Desarrollo Frontend	• 2 semanas	• 9	• 11	• Jean Pier Vega	• Interfaces Thymeleaf y componentes UI
• 7	• Pruebas y Monitoreo	• 2 semanas	• 11	• 13	• Jean Pier Vega	• Pruebas unitarias, integración y validación
• 8	• Implementación Final	• 2 semanas	• 13	• 15	• Jean Pier Vega	• Despliegue, capacitación y

						documentación
--	--	--	--	--	--	---------------

- **6.3 REQUERIMIENTOS DEL SISTEMA**

- **REQUERIMIENTOS FUNCIONALES**

• Nº	• Requerimiento	• Descripción
• RF1	• Gestión de Productos	• Registrar, editar, eliminar y visualizar productos con nombre, descripción, precio, categoría y stock
• RF2	• Control de Inventario	• Actualizar stock automáticamente al registrar ventas, alertas de stock bajo, historial de movimientos
• RF3	• Registro de Pedidos	• Crear pedidos con múltiples productos, calcular total automáticamente, gestionar estados (Pendiente/Preparación/Completado)
• RF4	• Gestión de Clientes	• Registrar clientes con datos de contacto y mantener historial completo de pedidos
• RF5	• Generación de Reportes	• Reportes de ventas por período, productos más vendidos, análisis de tendencias y rentabilidad
• RF6	• Autenticación y Roles	• Login seguro con roles (Admin/Cajero/Cocina) y control de acceso a funcionalidades
• RF7	• Dashboard en Tiempo Real	• Métricas clave: pedidos del día, ventas acumuladas, stock bajo, estado de pedidos
• RF8	• Búsqueda y Filtrado	• Buscar productos por nombre/categoría, filtrar pedidos por fecha/estado, buscar clientes

- **REQUERIMIENTOS NO FUNCIONALES**

• Nº	• Requerimiento	• Descripción
• RNF1	• Lenguaje y Plataforma	• Java 17 + Spring Boot 3.5.6 + MVC + PostgreSQL
• RNF2	• Rendimiento	• Respuesta en <3 segundos, soporte para 50 transacciones concurrentes
• RNF3	• Seguridad	• Spring Security + BCrypt, validación contra inyecciones SQL
• RNF4	• Usabilidad	• Interfaz intuitiva, aprendizaje <1 día, diseño responsive
• RNF5	• Mantenibilidad	• Principios SOLID, patrón Repository, código modular y documentado
• RNF6	• Disponibilidad	• 99% en horario de atención, funcionamiento local sin internet

• RNF7	• Compatibilidad	• Windows 10/11, Chrome 90+, Firefox 85+, Java 17
• RNF8	• Escalabilidad	• Arquitectura modular para futuras integraciones (delivery, app móvil)

- **7. DISEÑO DEL PROYECTO**
- **7.1 MODELADO DE PROCESOS (BPM)**
- **PROCESO PRINCIPAL: GESTIÓN DE PEDIDOS**

a 7. DISEÑO DEL PROYECTO

7.1 MODELADO DE PROCESOS (BPM)

PROCESO PRINCIPAL: GESTIÓN DE PEDIDOS

text

Copy

Download

[Inicio] → [Cliente solicita pedido] → [Cajero registra en sistema]
→ [Validación automática de stock] → [Actualización inventario]
→ [Envío a cocina] → [Preparación] → [Entrega]
→ [Registro de pago] → [Generación de reporte] → [Fin]

SUBPROCESOS:

- **Validación de Stock:** Verificación automática de disponibilidad
- **Actualización de Inventario:** Reducción automática de stock
- **Gestión de Estados:** Seguimiento del pedido (Pendiente → Preparación → Completado)
- **Generación de Comprobante:** PDF con detalles del pedido y pago

7.2 DISEÑO DE BASE DE DATOS

MODELO ENTIDAD-RELACIÓN

sql

Copy

Download

-- TABLA PRINCIPAL: PRODUCTOS

CREATE TABLE productos (

 id_producto SERIAL PRIMARY KEY,

 nombre VARCHAR(100) NOT NULL,

```
descripcion TEXT,  
precio DECIMAL(10,2) NOT NULL,  
stock INTEGER DEFAULT 0,  
categoria VARCHAR(50),  
activo BOOLEAN DEFAULT true  
);
```

-- TABLA PRINCIPAL: CLIENTES

```
CREATE TABLE clientes (  
    id_cliente SERIAL PRIMARY KEY,  
    nombre VARCHAR(100) NOT NULL,  
    telefono VARCHAR(15),  
    email VARCHAR(100),  
    fecha_registro DATE DEFAULT CURRENT_DATE  
);
```

-- TABLA PRINCIPAL: PEDIDOS

```
CREATE TABLE pedidos (  
    id_pedido SERIAL PRIMARY KEY,  
    id_cliente INTEGER REFERENCES clientes(id_cliente),  
    fecha TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    total DECIMAL(10,2) NOT NULL,  
    estado VARCHAR(20) DEFAULT 'PENDIENTE',  
    metodo_pago VARCHAR(20)  
);
```

-- TABLA DETALLE_PEDIDO

```
CREATE TABLE detalle_pedido (  
    id_detalle SERIAL PRIMARY KEY,  
    id_pedido INTEGER REFERENCES pedidos(id_pedido),  
    id_producto INTEGER REFERENCES productos(id_producto),
```

```
cantidad INTEGER NOT NULL,  
precio_unitario DECIMAL(10,2) NOT NULL,  
subtotal DECIMAL(10,2) NOT NULL  
);
```

7.3 UX/UI, REPORTES Y DOCUMENTACIÓN

ARQUITECTURA DE INFORMACIÓN

- **Login:** Autenticación segura con roles
- **Dashboard:** Resumen ejecutivo con métricas clave
- **Módulo Pedidos:** Registro y gestión de pedidos
- **Módulo Productos:** Catálogo y control de inventario
- **Módulo Clientes:** Base de datos y historial
- **Módulo Reportes:** Análisis y estadísticas

DISPLAY DE REPORTES

- **Reporte Ventas Diarias:** Gráfico de barras por hora
- **Reporte Productos Más Vendidos:** Ranking con porcentajes
- **Reporte Inventario:** Productos con stock bajo y crítico
- **Reporte Clientes:** Frecuencia y valor promedio de compra

DOCUMENTACIÓN

- Manual de usuario paso a paso
- Documentación técnica de la API
- Procedimientos de backup y recuperación
- Guía de solución de problemas comunes

8. DESARROLLO DEL PROYECTO

8.1 ARQUITECTURA (MVC, DAO, SOLID, TDD)

ARQUITECTURA MVC

- **Model:** Entidades JPA (Producto, Pedido, Cliente, DetallePedido)
- **View:** Plantillas Thymeleaf + Bootstrap 5 + JavaScript
- **Controller:** Controladores Spring (@Controller, @RestController)

PATRÓN DAO/REPOSITORY

@Repository

```
public interface ProductoRepository extends JpaRepository<Producto, Long> {
```

```

List<Producto> findByNombreContainingIgnoreCase(String nombre);
List<Producto> findByStockLessThan(Integer stockMinimo);
List<Producto> findByActivoTrue();
}

```

PRINCIPIOS SOLID

- **Single Responsibility:** Cada servicio con una única responsabilidad
- **Open/Closed:** Extensible sin modificar código existente
- **Liskov Substitution:** Interfaces implementables consistentemente
- **Interface Segregation:** Interfaces específicas por funcionalidad
- **Dependency Inversion:** Dependencias inyectadas via @Autowired

TDD (TEST-DRIVEN DEVELOPMENT)

```

public void testCrearPedido_StockSuficiente() {
    // Given
    Producto producto = new Producto("Hamburguesa Clásica", 15.50, 10);
    Pedido pedido = new Pedido();
    pedido.agregarDetalle(producto, 2);

    // When
    Pedido resultado = pedidoService.crearPedido(pedido);

    // Then
    assertNotNull(resultado.getId());
    assertEquals(8, producto.getStock()); // Stock actualizado
}

```

8.2 LIBRERÍAS Y RECURSOS EN JAVA

DEPENDENCIAS MAVEN PRINCIPALES

```

<dependencies>
    <!-- Spring Boot Starters -->
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>

```

```

</dependency>

<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
</dependency>

<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-thymeleaf</artifactId>
</dependency>

<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-security</artifactId>
</dependency>

<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-validation</artifactId>
</dependency>

```

<!-- Base de Datos -->

```

<dependency>
    <groupId>org.postgresql</groupId>
    <artifactId>postgresql</artifactId>
    <scope>runtime</scope>
</dependency>

```

<!-- Utilidades -->

```

<dependency>
    <groupId>org.projectlombok</groupId>
    <artifactId>lombok</artifactId>
    <optional>true</optional>
</dependency>

```

```
</dependencies>
```

CONFIGURACIÓN APPLICATION.PROPERTIES

```
# Database Configuration
```

```
spring.datasource.url=jdbc:postgresql://localhost:5432/hamburgueseria_piccola  
spring.datasource.username=piccola_user  
spring.datasource.password=piccola123
```

```
# JPA Configuration
```

```
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.show-sql=true  
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.PostgreSQLDialect  
spring.jpa.properties.hibernate.format_sql=true
```

```
# Server Configuration
```

```
server.port=8080
```

8.3 CONTROL DE VERSIONES (GIT & GITHUB)

ESTRATEGIA DE RAMAS

```
main (producción)
```

```
└── develop (integración)  
    ├── feature/autenticacion-usuarios  
    ├── feature/gestion-pedidos  
    ├── feature/control-inventario  
    └── feature/reportes-estadisticas
```

ESTRUCTURA DEL REPOSITORIO

```
hamburgueseria-piccola/  
├── .github/  
│   └── workflows/  
│       └── ci.yml  
└── src/  
    └── main/java/com/piccola/  
        └── controller/
```

```
| |   └── model/
| |   └── repository/
| |   └── service/
| |   └── config/
| └── main/resources/
|   └── templates/
|   └── static/
|   └── application.properties
└── test/java/com/piccola/
    └── docs/
        ├── manual-usuario.md
        ├── api-documentation.md
        └── deployment-guide.md
    └── scripts/
        ├── database/
        └── deployment/
    └── .gitignore
    └── pom.xml
    └── README.md
```

CONVENCIONES DE COMMIT

- feat: Nueva funcionalidad
- fix: Corrección de bug
- docs: Documentación
- style: Formato de código
- refactor: Refactorización
- test: Pruebas
- chore: Tareas de mantenimiento

Anexos del código

The screenshot shows the Visual Studio Code interface. The left pane, titled "EXPLORER", displays the project structure under "HAMBURGUESERIA-PICCOLA ACTUALIZADA". The right pane, titled "HamburgueseriaPiccolaApplication.java", shows the code for the main application class.

```
1 package com.piccola;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5
6 @SpringBootApplication
7 public class HamburgueseriaPiccolaApplication {
8
9     public static void main(String[] args) {
10         SpringApplication.run(HamburgueseriaPiccolaApplication.class, args);
11     }
12 }
13
14 }
```

OPEN EDITORS

HAMBURGUESERIA-PICCOLA A... 🔍 ⚙️ ⌂ ⌂

- .github
- .vscode
- hamburgueseria-piccola
 - .github\java-upgrade
 - .idea
 - .mvn
 - src
 - main
 - java\com\piccola
 - controller
 - ClienteController.java
 - HomeController.java
 - PedidoController.java
 - ProductoController.java
- model
- repository
- service
 - HamburgueseriaPiccolaApplication.java
- resources
- test
- target
- .gitattributes
- .citiciones

OUTLINE

TIMELINE

JAVA PROJECTS

MAVEN

EXPLORER

OPEN EDITORS

HAMBURGUESERIA-PICCOLA A... 🔍 ⚙️ ⌂ ⌂

- .github
- .vscode
- hamburgueseria-piccola
 - .github
 - .idea
 - .mvn
 - src
 - main
 - java\com\piccola
 - controller
 - ClienteController.java
 - HomeController.java
 - PedidoController.java
 - ProductoController.java
- model
- repository
- service
 - HamburgueseriaPiccolaApplication.java
- resources
- test
- target
- .gitattributes
- .citiciones

OUTLINE

TIMELINE

JAVA PROJECTS

MAVEN

hamburgueseria-piccola > src > main > java > com > piccola > controller > ClienteController.java > ClienteController

```

1 package com.piccola.controller;
2
3 import com.piccola.model.Cliente;
4 import com.piccola.service.ClienteService;
5 import lombok.RequiredArgsConstructor;
6 import org.springframework.http.HttpStatus;
7 import org.springframework.http.ResponseEntity;
8 import org.springframework.stereotype.Controller;
9 import org.springframework.ui.Model;
10 import org.springframework.validation.BindingResult;
11 import org.springframework.web.bind.annotation.*;
12
13 import jakarta.validation.Valid;
14 import java.util.List;
15
16 @Controller
17 @RequestMapping("/clientes")
18 @RequiredArgsConstructor
19 public class ClienteController {
20
21     private final ClienteService clienteService;
22
23     @GetMapping
24     public String listar(Model model) {
25         model.addAttribute("clientes", clienteService.listarTodos());
26         return "clientes/lista";
27     }
28
29     @GetMapping("/nuevo")
30     public String mostrarFormularioNuevo(Model model) {
31         model.addAttribute("cliente", new Cliente());
32         return "clientes/formulario";
33     }

```

HomeController.java 1 ✘

hamburgueseria-piccola > src > main > java > com > piccola > controller > HomeController.java > HomeController

```

1 package com.piccola.controller;
2
3 import com.piccola.service.ClienteService;
4 import com.piccola.service.PedidoService;
5 import com.piccola.service.ProductoService;
6 import com.piccola.model.Pedido.EstadoPedido;
7 import lombok.RequiredArgsConstructor;
8 import org.springframework.stereotype.Controller;
9 import org.springframework.ui.Model;
10 import org.springframework.web.bind.annotation.GetMapping;
11
12 import java.time.LocalDateTime;
13
14 @Controller
15 @RequiredArgsConstructor
16 public class HomeController {
17
18     private final ProductoService productoService;
19     private final ClienteService clienteService;
20     private final PedidoService pedidoService;
21
22     @GetMapping("/")
23     public String home(Model model) {
24         // Estadísticas para el dashboard
25         model.addAttribute("totalProductos", productoService.listarTodos().size());
26         model.addAttribute("totalClientes", clienteService.listarTodos().size());
27         model.addAttribute("pedidosPendientes", pedidoService.contarPedidosPorEstado(EstadoPedido.PENDIENTE));
28         model.addAttribute("pedidosEnPreparacion", pedidoService.contarPedidosPorEstado(EstadoPedido.EN_PRE);
29
30         // Productos con stock bajo
31         model.addAttribute("productosStockBajo", productoService.listarConStockBajo());
32     }

```

hamburgueseria-piccola > src > main > java > com > piccola > controller > J PedidoController.java > PedidoController > listarPorE

```

1 package com.piccola.controller;
2
3 import com.piccola.model.DetallePedido;
4 import com.piccola.model.Pedido;
5 import com.piccola.model.Pedido.EstadoPedido;
6 import com.piccola.service.ClienteService;
7 import com.piccola.service.PedidoService;
8 import com.piccola.service.ProductoService;
9 import lombok.RequiredArgsConstructor;
10 import org.springframework.http.HttpStatus;
11 import org.springframework.http.ResponseEntity;
12 import org.springframework.stereotype.Controller;
13 import org.springframework.ui.Model;
14 import org.springframework.web.bind.annotation.*;
15
16 import java.util.ArrayList;
17 import java.util.List;
18 import java.util.Map;
19
20 @Controller
21 @RequestMapping("/pedidos")
22 @RequiredArgsConstructor
23 public class PedidoController {
24
25     private final PedidoService pedidoService;
26     private final ClienteService clienteService;
27     private final ProductoService productoService;
28
29     @GetMapping
30     public String listar(Model model) {
31         model.addAttribute("pedidos", pedidoService.listarTodos());
32     }

```

EXPLORER ... J ProductoController.java X

hamburgueseria-piccola > src > main > java > com > piccola > controller > J ProductoController.java > ...

```

1 package com.piccola.controller;
2
3 import com.piccola.model.Producto;
4 import com.piccola.service.ProductoService;
5 import lombok.RequiredArgsConstructor;
6 import org.springframework.http.HttpStatus;
7 import org.springframework.http.ResponseEntity;
8 import org.springframework.stereotype.Controller;
9 import org.springframework.ui.Model;
10 import org.springframework.validation.BindingResult;
11 import org.springframework.web.bind.annotation.*;
12
13 import jakarta.validation.Valid;
14 import java.util.List;
15
16 @Controller
17 @RequestMapping("/productos")
18 @RequiredArgsConstructor
19 public class ProductoController {
20
21     private final ProductoService productoService;
22
23     @GetMapping
24     public String listar(Model model) {
25         model.addAttribute("productos", productoService.listarActivos());
26         return "productos/lista";
27     }
28
29     @GetMapping("/nuevo")
30     public String mostrarFormularioNuevo(Model model) {
31         model.addAttribute("producto", new Producto());
32         return "productos/formulario";
33     }

```

OPEN EDITORS

HAMBURGUESERIA-PICCOLA A... .github .vscode C

.github
.idea
.mvn
src
main
java\com\piccola
controller
J ClienteController.java
J HomeController.java 1
J PedidoController.java 1
J ProductoController.java
model
repository
service
J HamburgueseriaPiccolaApplication.java
resources
test
target
.gitattributes
citiciones

OUTLINE

TIMELINE

JAVA PROJECTS

MAVEN

EXPLORER

> OPEN EDITORS

< HAMBURGUESERIA-PICCOLA A... 🔍 🗑️ ⌂ ⌂

- > .github
- > .vscode
- < hamburgueseria-piccola
 - > .github
 - > .idea
 - > .mvn
 - < src
 - < main
 - < java \ com \ piccola
 - < controller
 - J ClienteController.java
 - J HomeController.java
 - J PedidoController.java
 - J ProductoController.java
 - < model
 - J Cliente.java
 - J DetallePedido.java
 - J Pedido.java
 - J Producto.java
 - > repository
 - > service
 - J HamburgueseriaPiccolaApplication.java

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

> OPEN EDITORS

< HAMBURGUESERIA-PICCOLA A... 🔍 🗑️ ⌂ ⌂

- > .github
- > .vscode
- < hamburgueseria-piccola
 - > .github
 - > .idea
 - > .mvn
 - < src
 - < main
 - < java \ com \ piccola
 - < controller
 - J ClienteController.java
 - J HomeController.java
 - J PedidoController.java
 - J ProductoController.java
 - < model
 - J Cliente.java
 - J DetallePedido.java
 - J Pedido.java
 - J Producto.java
 - > repository
 - > service
 - J HamburgueseriaPiccolaApplication.java

> resources

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

J Cliente.java X

hamburgueseria-piccola > src > main > java > com > piccola > model > J Cliente.java > Cliente > nombre

```

1 package com.piccola.model;
2
3 import jakarta.persistence.*;
4 import jakarta.validation.constraints.*;
5 import lombok.AllArgsConstructorConstructor;
6 import lombok.Data;
7 import lombok.NoArgsConstructorConstructor;
8
9 import java.time.LocalDate;
10 import java.time.LocalDateTime;
11 import java.util.ArrayList;
12 import java.util.List;
13
14 @Entity
15 @Table(name = "clientes")
16 @Data
17 @NoArgsConstructorConstructor
18 @AllArgsConstructorConstructor
19 public class Cliente {
20
21     @Id
22     @GeneratedValue(strategy = GenerationType.IDENTITY)
23     @Column(name = "id_cliente")
24     private Long id;
25
26     @NotBlank(message = "El nombre es obligatorio")
27     @Size(min = 3, max = 100)
28     @Column(nullable = false, length = 100)
29     private String nombre;
30
31     @Pattern(regexp = "^[0-9]{9,15}$", message = "El teléfono debe tener entre 9 y 15 dígitos")
32     @Column(length = 15)

```

J DetallePedido.java

hamburgueseria-piccola > src > main > java > com > piccola > model > J DetallePedido.java > {} com.piccola.model

```

1 package com.piccola.model;
2
3 import jakarta.persistence.*;
4 import jakarta.validation.constraints.*;
5 import lombok.AllArgsConstructorConstructor;
6 import lombok.Data;
7 import lombok.NoArgsConstructorConstructor;
8
9 import java.math.BigDecimal;
10
11 @Entity
12 @Table(name = "detalle_pedido")
13 @Data
14 @NoArgsConstructorConstructor
15 @AllArgsConstructorConstructor
16 public class DetallePedido {
17
18     @Id
19     @GeneratedValue(strategy = GenerationType.IDENTITY)
20     @Column(name = "id_detalle")
21     private Long id;
22
23     @ManyToOne(fetch = FetchType.LAZY)
24     @JoinColumn(name = "id_pedido", nullable = false)
25     private Pedido pedido;
26
27     @ManyToOne(fetch = FetchType.EAGER)
28     @JoinColumn(name = "id_producto", nullable = false)
29     private Producto producto;
30
31     @NotNull
32     @Min(value = 1)

```

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA ACTUALIZADA

- > .github
- > .vscode
- ✗ hamburgueseria-piccola
 - > .github
 - > .idea
 - > .mvn
 - ✓ src
 - ✓ main
 - ✓ java\com\piccola
 - ✓ controller
 - J ClienteController.java
 - J HomeController.java
 - J PedidoController.java
 - J ProductoController.java
 - ✓ model
 - J Cliente.java
 - J DetallePedido.java
 - J Pedido.java
 - J Producto.java
 - > repository
 - > service
 - J HamburgueseriaPiccolaApplication.java

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

J Pedido.java

```

hamburgueseria-piccola > src > main > java > com > piccola > model > J Pedido.java > {} com.piccola.model
  1 package com.piccola.model;
  2
  3 import jakarta.persistence.*;
  4 import jakarta.validation.constraints.*;
  5 import lombok.AllArgsConstructorConstructor;
  6 import lombok.Data;
  7 import lombok.NoArgsConstructor;
  8
  9 import java.math.BigDecimal;
 10 import java.time.LocalDateTime;
 11 import java.util.ArrayList;
 12 import java.util.List;
 13
 14 @Entity
 15 @Table(name = "pedidos")
 16 @Data
 17 @NoArgsConstructor
 18 @AllArgsConstructor
 19 public class Pedido {
 20
 21     @Id
 22     @GeneratedValue(strategy = GenerationType.IDENTITY)
 23     @Column(name = "id_pedido")
 24     private Long id;
 25
 26     @ManyToOne(fetch = FetchType.EAGER)
 27     @JoinColumn(name = "id_cliente")
 28     private Cliente cliente;
 29
 30     @Column(nullable = false)
 31     private LocalDateTime fecha;
 32

```

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... 🔍 🔍 C ⌂

- > .github
- > .vscode
- ✗ hamburgueseria-piccola
 - > .github
 - > .idea
 - > .mvn
 - ✓ src
 - ✓ main
 - ✓ java\com\piccola
 - ✓ controller
 - J ClienteController.java
 - J HomeController.java
 - J PedidoController.java
 - J ProductoController.java
 - ✓ model
 - J Cliente.java
 - J DetallePedido.java
 - J Pedido.java
 - J Producto.java

J Producto.java

J Pedido.java

```

hamburgueseria-piccola > src > main > java > com > piccola > model > J Producto.java > {} com.piccola.model
  17 public class Producto {
 49     private LocalDateTime fechaCreacion;
 50
 51     @Column(name = "fecha_actualizacion")
 52     private LocalDateTime fechaActualizacion;
 53
 54     @PrePersist
 55     protected void onCreate() {
 56         fechaCreacion = LocalDateTime.now();
 57         fechaActualizacion = LocalDateTime.now();
 58     }
 59
 60     @PreUpdate
 61     protected void onUpdate() {
 62         fechaActualizacion = LocalDateTime.now();
 63     }
 64
 65     public void reducirStock(Integer cantidad) {
 66         if (this.stock < cantidad) {
 67             throw new IllegalStateException("Stock insuficiente para: " + this.nombre);
 68         }
 69         this.stock -= cantidad;
 70     }
 71
 72     public void aumentarStock(Integer cantidad) {
 73         this.stock += cantidad;
 74     }
 75
 76     public boolean tieneStockBajo() {
 77         return this.stock < 10;
 78     }
 79

```

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... []

- hamburgueseria-piccola
 - .github
 - .idea
 - .mvn
 - src
 - main
 - java\com\piccola
 - controller
 - J ClienteController.java
 - J HomeController.java
 - J PedidoController.java
 - J ProductoController.java
 - model
 - J Cliente.java
 - J DetallePedido.java
 - J Pedido.java
 - J Producto.java
 - repository
 - J ClienteRepository.java
 - J DetallePedidoRepository.java
 - J PedidoRepository.java
 - J ProductoRepository.java
 - service

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... []

 - hamburgueseria-piccola
 - .github
 - .idea
 - .mvn
 - src
 - main
 - java\com\piccola
 - controller
 - J ClienteController.java
 - J HomeController.java
 - J PedidoController.java
 - J ProductoController.java
 - model
 - J Cliente.java
 - J DetallePedido.java
 - J Pedido.java
 - J Producto.java
 - repository
 - J ClienteRepository.java
 - J DetallePedidoRepository.java
 - J PedidoRepository.java
 - J ProductoRepository.java
 - service

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

hamburgueseria-piccola > src > main > java > com > piccola > repository > J ClienteRepository.java > () com.piccola.repository

```

1 package com.piccola.repository;
2
3 import com.piccola.model.Cliente;
4 import org.springframework.data.jpa.repository.JpaRepository;
5 import org.springframework.stereotype.Repository;
6
7 import java.util.List;
8 import java.util.Optional;
9
10 @Repository
11 public interface ClienteRepository extends JpaRepository<Cliente, Long> {
12
13     Optional<Cliente> findByTelefono(String telefono);
14
15     Optional<Cliente> findByEmail(String email);
16
17     List<Cliente> findByNombreContainingIgnoreCase(String nombre);
18 }
```

hamburgueseria-piccola > src > main > java > com > piccola > repository > J DetallePedidoRepository.java > () com.piccola.repository

```

1 package com.piccola.repository;
2
3 import com.piccola.model.DetallePedido;
4 import org.springframework.data.jpa.repository.JpaRepository;
5 import org.springframework.stereotype.Repository;
6
7 import java.util.List;
8
9 @Repository
10 public interface DetallePedidoRepository extends JpaRepository<DetallePedido, Long> {
11
12     List<DetallePedido> findByPedidoId(Long pedidoId);
13
14     List<DetallePedido> findByProductoId(Long productoId);
15 }
```

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... 🔍 🌐 ⚙️ 📁

- ✗ hamburgueseria-piccola
 - > .github
 - > .idea
 - > .mvn
 - ✓ src
 - ✗ main
 - ✗ java\com\piccola
 - ✗ controller
 - J ClienteController.java
 - J HomeController.java
 - J PedidoController.java
 - J ProductoController.java
 - ✗ model
 - J Cliente.java
 - J DetallePedido.java
 - J Pedido.java
 - J Producto.java
 - ✗ repository
 - J ClienteRepository.java
 - J DetallePedidoRepository.java
 - J PedidoRepository.java
 - J ProductoRepository.java
 - > service

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... 🔍 🌐 ⚙️ 📁

 - ✗ hamburgueseria-piccola
 - > .github
 - > .idea
 - > .mvn
 - ✓ src
 - ✗ main
 - ✗ java\com\piccola
 - ✗ controller
 - J ClienteController.java
 - J HomeController.java
 - J PedidoController.java
 - J ProductoController.java
 - ✗ model
 - J Cliente.java
 - J DetallePedido.java
 - J Pedido.java
 - J Producto.java
 - ✗ repository
 - J ClienteRepository.java
 - J DetallePedidoRepository.java
 - J PedidoRepository.java
 - J ProductoRepository.java
 - > service

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

J Pedido.java

hamburgueseria-piccola > src > main > java > com > piccola > repository > J PedidoRepository.java > {} com.piccola.repository

```

1 package com.piccola.repository;
2
3 import com.piccola.model.Pedido;
4 import com.piccola.model.Pedido.EstadoPedido;
5 import org.springframework.data.jpa.repository.JpaRepository;
6 import org.springframework.stereotype.Repository;
7
8 import java.time.LocalDateTime;
9 import java.util.List;
10
11 @Repository
12 public interface PedidoRepository extends JpaRepository<Pedido, Long> {
13
14     List<Pedido> findByEstado(EstadoPedido estado);
15
16     List<Pedido> findByClienteId(Long clienteId);
17
18     List<Pedido> findByFechaBetween(LocalDateTime inicio, LocalDateTime fin);
19
20     List<Pedido> findByEstadoOrderByFechaDesc(EstadoPedido estado);
21
22     List<Pedido> findAllOrderByFechaDesc();
23 }
```

J Producto.java

hamburgueseria-piccola > src > main > java > com > piccola > repository > J ProductoRepository.java > {} com.piccola.repository

```

1 package com.piccola.repository;
2
3 import com.piccola.model.Producto;
4 import org.springframework.data.jpa.repository.JpaRepository;
5 import org.springframework.stereotype.Repository;
6
7 import java.util.List;
8
9 @Repository
10 public interface ProductoRepository extends JpaRepository<Producto, Long> {
11
12     List<Producto> findByActivoTrue();
13
14     List<Producto> findByCategoria(String categoria);
15
16     List<Producto> findByNombreContainingIgnoreCase(String nombre);
17
18     List<Producto> findByStockLessThan(Integer stock);
19 }
20
```

EXPLORER

OPEN EDITORS

HAMBURGUESERIA-PICCOLA A...

- hamburgueseria-piccola
 - .mvn
 - src
 - main
 - java\com\piccola
 - controller
 - model
 - repository
 - service
 - J ClienteService.java
 - J PedidoService.java
 - J ProductoService.java
- HamburgueseriaPiccolaApplication.java
- resources
- test
- target
- .gitattributes
- .gitignore
- HELP.md
- mvnw
- mvnw.cmd
- pom.xml

OUTLINE

TIMELINE

JAVA PROJECTS

MAVEN

EXPLORER

OPEN EDITORS

HAMBURGUESERIA-PICCOLA A...

- hamburgueseria-piccola
 - .mvn
 - src
 - main
 - java\com\piccola
 - controller
 - model
 - repository
 - service
 - J ClienteService.java
 - J PedidoService.java
 - J ProductoService.java
- HamburgueseriaPiccolaApplication.java
- resources
- test
- target
- .gitattributes
- .gitignore
- HELP.md
- mvnw
- mvnw.cmd
- pom.xml

OUTLINE

TIMELINE

JAVA PROJECTS

MAVEN

J Pedido.java

J ClienteService.java

```

hamburgueseria-piccola > src > main > java > com > piccola > service > J ClienteService.java > {} com.piccola.service
  1 package com.piccola.service;
  2
  3 import com.piccola.model.Cliente;
  4 import com.piccola.repository.ClienteRepository;
  5 import lombok.RequiredArgsConstructor;
  6 import org.springframework.stereotype.Service;
  7 import org.springframework.transaction.annotation.Transactional;
  8
  9 import java.util.List;
 10 import java.util.Optional;
 11
 12 @Service
 13 @RequiredArgsConstructor
 14 @Transactional
 15 public class ClienteService {
 16
 17     private final ClienteRepository clienteRepository;
 18
 19     public List<Cliente> listarTodos() {
 20         return clienteRepository.findAll();
 21     }
 22
 23     public Optional<Cliente> buscarPorId(Long id) {
 24         return clienteRepository.findById(id);
 25     }
 26
 27     public Optional<Cliente> buscarPorTelefono(String telefono) {
 28         return clienteRepository.findByTelefono(telefono);
 29     }
 30
 31     public Optional<Cliente> buscarPorEmail(String email) {
 32         return clienteRepository.findByEmail(email);
 33     }
 34
 35 }
```

J Pedido.java

J PedidoService.java

```

hamburgueseria-piccola > src > main > java > com > piccola > service > J PedidoService.java > {} com.piccola.service
  1 package com.piccola.service;
  2
  3 import com.piccola.model.Cliente;
  4 import com.piccola.model.DetallePedido;
  5 import com.piccola.model.Pedido;
  6 import com.piccola.model.Pedido.EstadoPedido;
  7 import com.piccola.model.Producto;
  8 import com.piccola.repository.PedidoRepository;
  9 import lombok.RequiredArgsConstructor;
 10 import org.springframework.stereotype.Service;
 11 import org.springframework.transaction.annotation.Transactional;
 12
 13 import java.math.BigDecimal;
 14 import java.time.LocalDateTime;
 15 import java.util.List;
 16 import java.util.Optional;
 17
 18 @Service
 19 @RequiredArgsConstructor
 20 @Transactional
 21 public class PedidoService {
 22
 23     private final PedidoRepository pedidoRepository;
 24     private final ProductoService productoService;
 25     private final ClienteService clienteService;
 26
 27     public List<Pedido> listarTodos() {
 28         return pedidoRepository.findAllOrderByFechaDesc();
 29     }
 30
 31     public Optional<Pedido> buscarPorId(Long id) {
 32         return pedidoRepository.findById(id);
 33     }
 34
 35 }
```

EXPLORER

> OPEN EDITORS

< HAMBURGUESERIA-PICCOLA ACTUALIZADA

- ✓ hamburgueseria-piccola
 - / .mvn
 - ✓ src
 - ✓ main
 - ✓ java\com\piccola
 - > controller
 - > model
 - > repository
 - ✓ service
 - J ClienteService.java 4
 - J PedidoService.java 4
 - J ProductoService.java 8

J HamburgueseriaPiccolaApplication.java

 - > resources
 - > test
 - > target
 - ❖ .gitattributes
 - ❖ .gitignore
 - HELP.md
 - mvnw
 - mvnw.cmd
 - pom.xml

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

Pedido.java

hamburgueseria-piccola > src > main > java > com > piccola > service > J ProductoService.java > ProductoService

```

1 package com.piccola.service;
2
3 import com.piccola.model.Producto;
4 import com.piccola.repository.ProductoRepository;
5 import lombok.RequiredArgsConstructor;
6 import org.springframework.stereotype.Service;
7 import org.springframework.transaction.annotation.Transactional;
8
9 import java.util.List;
10 import java.util.Optional;
11
12 @Service
13 @RequiredArgsConstructor
14 @Transactional
15 public class ProductoService {
16
17     private final ProductoRepository productoRepository;
18
19     public List<Producto> listarTodos() {
20         return productoRepository.findAll();
21     }
22
23     public List<Producto> listarActivos() {
24         return productoRepository.findByActivoTrue();
25     }
26
27     public Optional<Producto> buscarPorId(Long id) {
28         return productoRepository.findById(id);
29     }
30
31     public List<Producto> buscarPorCategoria(String categoria) {
32         return productoRepository.findByCategoria(categoria);
33     }

```

EXPLORER

> OPEN EDITORS

< HAMBURGUESERIA-PICCOLA A...

 - ✓ hamburgueseria-piccola
 - / .mvn
 - ✓ src
 - ✓ main
 - ✓ java\com\piccola
 - > controller
 - > model
 - > repository
 - ✓ service
 - J ClienteService.java 4
 - J PedidoService.java 4
 - J ProductoService.java 8

J HamburgueseriaPiccolaApplication.java

 - > resources
 - > static
 - > templates
 - > clientes

formulario.html

hamburgueseria-piccola > src > main > resources > templates > clientes > formulario.html

```

1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Formulario Cliente - Hamburgueseria Piccola</title>
7     <style>
8         * {
9             margin: 0;
10            padding: 0;
11            box-sizing: border-box;
12        }
13
14        body {
15            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
16            background: linear-gradient(135deg, #667eea 0%, #76ba2 100%);
17            min-height: 100vh;
18            padding: 20px;
19        }
20
21        .container {
22            max-width: 800px;
23            margin: 0 auto;
24        }
25
26        .card {
27            background: white;
28            padding: 40px;
29            border-radius: 15px;
30            box-shadow: 0 10px 30px rgba(0,0,0,0.2);
31        }
32    </style>

```

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... 🔍 🎯 ⏺

- ✓ hamburgueseria-piccola
 - / .IMWI
 - ✓ src
 - ✓ main
 - ✓ java\com\piccola
 - > controller
 - > model
 - > repository
 - > service
 - J ClienteService.java 4
 - J PedidoService.java 4
 - J ProductoService.java 8
 - J HamburgueseriaPiccolaApplication.java
 - ✓ resources
 - ✓ static
 - ✓ templates
 - ✓ clientes
 - ↳ formulario.html
 - ↳ lista.html
 - > pedidos
 - > productos
 - ↳ index.html
 - application.properties

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

PEDIDO.JAVA

↳ lista.html X

```

hamburgueseria-piccola > src > main > resources > templates > clientes > lista.html > html > head > style > .container
1  <!DOCTYPE html>
2  <html xmlns:th="http://www.thymeleaf.org">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Clientes - Hamburguesería Piccola</title>
7      <style>
8          * {
9              margin: 0;
10             padding: 0;
11             box-sizing: border-box;
12         }
13
14         body {
15             font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
16             background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
17             min-height: 100vh;
18             padding: 20px;
19         }
20
21         .container {
22             max-width: 1200px;
23             margin: 0 auto;
24         }
25
26         header {
27             background: white;
28             padding: 30px;
29             border-radius: 15px;
30             box-shadow: 0 10px 30px rgba(0,0,0,0.2);
31             margin-bottom: 30px;
32             display: flex;

```

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... 🔍 🎯 ⏺

- ✓ hamburgueseria-piccola
 - / .IMWI
 - ✓ src
 - ✓ main
 - ✓ java\com\piccola
 - > controller
 - > model
 - > repository
 - > service
 - J ClienteService.java 4
 - J PedidoService.java 4
 - J ProductoService.java 8
 - J HamburgueseriaPiccolaApplication.java
 - ✓ resources
 - ✓ static
 - ✓ templates
 - ✓ clientes
 - ↳ formulario.html
 - ↳ lista.html
 - > pedidos
 - ↳ formulario.html
 - > productos

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

PEDIDO.JAVA

↳ formulario.htm X

```

hamburgueseria-piccola > src > main > resources > templates > pedidos > formulario.html > html > body > script > eliminarProducto
1  <!DOCTYPE html>
2  <html xmlns:th="http://www.thymeleaf.org">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Nuevo Pedido - Hamburguesería Piccola</title>
7      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
8      <link href="https://cdn.jsdelivr.net/npm/fontawesome@6.4.0/css/all.min.css" rel="stylesheet">
9  </head>
10 <body>
11     <nav class="navbar navbar-expand-lg navbar-dark bg-primary">
12         <div class="container-fluid">
13             <a class="navbar-brand" href="/dashboard"><i class="fas fa-burger"></i> Hamburguesería Piccola</a>
14         </div>
15     </nav>
16
17     <div class="container mt-4">
18         <div class="d-flex justify-content-between align-items-center mb-4">
19             <h1><i class="fas fa-plus-circle"></i> Nuevo Pedido</h1>
20             <a href="/pedidos" class="btn btn-secondary">
21                 <i class="fas fa-arrow-left"></i> Volver
22             </a>
23         </div>
24
25         <form action="/pedidos/guardar" method="post">
26             <div class="row">
27                 <div class="col-md-8">
28                     <div class="card mb-3">
29                         <div class="card-header bg-primary text-white">
30                             <h5><i class="fas fa-box"></i> Seleccionar Productos</h5>
31                         </div>
32                         <div class="card-body">

```

EXPLORER

HAMBURGUESERIA-PICCOLA A...

- hamburgueseria-piccola
- src
- main
 - java\com\piccola
 - controller
 - model
 - repository
 - service
 - J ClienteService.java
 - J PedidoService.java
 - J ProductoService.java
 - J HamburgueseriaPiccolaApplication.java
 - resources
 - static
 - templates
 - clientes
 - formulario.html
 - lista.html
 - pedidos
 - formulario.html
 - lista.html
 - productos
 - formulario.html
 - index.html

OPEN EDITORS

J Pedido.java

hamburgueseria-piccola > src > main > resources > templates > pedidos > lista.html

```

2  <html xmlns:th="http://www.thymeleaf.org">
10 <body>
11 <nav class="navbar navbar-expand-lg navbar-dark bg-primary">
23 </nav>
24 <div class="container mt-4">
25   <div class="d-flex justify-content-between align-items-center mb-4">
26     <i class="fas fa-shopping-cart"></i> Pedidos</h1>
27     <a href="/pedidos/nuevo" class="btn btn-success btn-lg">
28       <i class="fas fa-plus"></i> Nuevo Pedido
29     </a>
30   </div>
31
32   <div th:if="${mensaje}" th:classappend="${tipoMensaje == 'success' ? 'alert-success' : 'alert-danger'}"
33     class="alert alert-dismissible fade show" role="alert">
34     <span th:text="${mensaje}"></span>
35     <button type="button" class="btn-close" data-bs-dismiss="alert"></button>
36   </div>
37
38   <div class="card">
39     <div class="card-body">
40       <div class="table-responsive">
41         <table class="table table-hover">
42           <thead class="table-primary">
43             <tr>
44               <th>ID</th>
45               <th>Cliente</th>
46               <th>Fecha</th>
47               <th>Total</th>
48               <th>Estado</th>
49               <th>Método Pago</th>
50               <th>Acciones</th>
51             </tr>

```

J Pedido.java

hamburgueseria-piccola > src > main > resources > templates > productos > formulario.html

```

1  <!DOCTYPE html>
2  <html xmlns:th="http://www.thymeleaf.org">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Formulario Producto - Hamburguesería Piccola</title>
7    <style>
8      * {
9        margin: 0;
10       padding: 0;
11       box-sizing: border-box;
12     }
13
14   body {
15     font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
16     background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
17     min-height: 100vh;
18     padding: 20px;
19   }
20
21   .container {
22     max-width: 800px;
23     margin: 0 auto;
24   }
25
26   .card {
27     background: white;
28     padding: 10px;
29     border-radius: 15px;
30     box-shadow: 0 10px 30px rgba(0,0,0,0.2);
31   }

```

EXPLORER

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... 🔍 ⚡ C ⏺

- ✓ hamburgueseria-piccola
 - ✓ src
 - ✓ main
 - ✓ java\com\piccola
 - ✓ service
 - J HamburgueseriaPiccolaApplication.java

✓ resources

✓ static

✓ templates

 - ✓ clientes
 - ◊ formulario.html
 - ◊ lista.html
 - ✓ pedidos
 - ◊ formulario.html
 - ◊ lista.html
 - ✓ productos
 - ◊ formulario.html

◊ lista.html

◊ index.html

application.properties

> test

> target

↳ .gitattributes

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

> OPEN EDITORS

✓ HAMBURGUESERIA-PICCOLA A... 🔍 ⚡ C ⏺

 - ✓ hamburgueseria-piccola
 - ✓ src
 - ✓ main
 - ✓ java\com\piccola
 - ✓ service
 - J HamburgueseriaPiccolaApplication.java

✓ resources

✓ static

✓ templates

 - ✓ clientes
 - ◊ formulario.html
 - ◊ lista.html
 - ✓ pedidos
 - ◊ formulario.html
 - ◊ lista.html
 - ✓ productos
 - ◊ formulario.html
 - ◊ lista.html

◊ index.html

application.properties

> test

> target

↳ .gitattributes

> OUTLINE

> TIMELINE

> JAVA PROJECTS

> MAVEN

J Pedido.java

hamburgueseria-piccola > src > main > resources > templates > productos > lista.html > ...

```

1  <!DOCTYPE html>
2  <html xmlns:th="http://www.thymeleaf.org">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Productos - Hamburguesería Piccola</title>
7      <style>
8          * {
9              margin: 0;
10             padding: 0;
11             box-sizing: border-box;
12         }
13
14         body {
15             font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
16             background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
17             min-height: 100vh;
18             padding: 20px;
19         }
20
21         .container {
22             max-width: 1200px;
23             margin: 0 auto;
24         }
25
26         header {
27             background: white;
28             padding: 30px;
29             border-radius: 15px;
30             box-shadow: 0 10px 30px rgba(0,0,0,0.2);
31             margin-bottom: 30px;
32             display: flex;
33         }
34
35         .header-content {
36             flex-grow: 1;
37             text-align: center;
38         }
39
40         .header-content h1 {
41             margin: 0;
42             font-size: 2em;
43             color: #764ba2;
44         }
45
46         .header-content p {
47             margin: 10px 0;
48             font-size: 0.9em;
49             color: #764ba2;
50         }
51
52         .header-content ul {
53             list-style-type: none;
54             padding: 0;
55         }
56         .header-content ul li {
57             display: inline-block;
58             margin-right: 10px;
59         }
60
61         .header-content ul li a {
62             text-decoration: none;
63             color: inherit;
64         }
65
66         .header-content ul li a:hover {
67             color: #764ba2;
68         }
69
70         .header-content ul li a.active {
71             color: #764ba2;
72         }
73
74         .header-content ul li a.active::after {
75             content: '';
76             width: 100px;
77             height: 2px;
78             background-color: #764ba2;
79             margin-left: 10px;
80         }
81
82         .header-content ul li a.active::before {
83             content: '';
84             width: 100px;
85             height: 2px;
86             background-color: #764ba2;
87             margin-left: 10px;
88         }
89
90         .header-content ul li a.active::before {
91             width: 100px;
92             height: 2px;
93             background-color: #764ba2;
94             margin-left: 10px;
95         }
96
97         .header-content ul li a.active::before {
98             width: 100px;
99             height: 2px;
100            background-color: #764ba2;
101           margin-left: 10px;
102       }
103
104       .header-content ul li a.active::before {
105           width: 100px;
106           height: 2px;
107           background-color: #764ba2;
108           margin-left: 10px;
109       }
110
111       .header-content ul li a.active::before {
112           width: 100px;
113           height: 2px;
114           background-color: #764ba2;
115           margin-left: 10px;
116       }
117
118       .header-content ul li a.active::before {
119           width: 100px;
120           height: 2px;
121           background-color: #764ba2;
122           margin-left: 10px;
123       }
124
125       .header-content ul li a.active::before {
126           width: 100px;
127           height: 2px;
128           background-color: #764ba2;
129           margin-left: 10px;
130       }
131
132       .header-content ul li a.active::before {
133           width: 100px;
134           height: 2px;
135           background-color: #764ba2;
136           margin-left: 10px;
137       }
138
139       .header-content ul li a.active::before {
140           width: 100px;
141           height: 2px;
142           background-color: #764ba2;
143           margin-left: 10px;
144       }
145
146       .header-content ul li a.active::before {
147           width: 100px;
148           height: 2px;
149           background-color: #764ba2;
150           margin-left: 10px;
151       }
152
153       .header-content ul li a.active::before {
154           width: 100px;
155           height: 2px;
156           background-color: #764ba2;
157           margin-left: 10px;
158       }
159
160       .header-content ul li a.active::before {
161           width: 100px;
162           height: 2px;
163           background-color: #764ba2;
164           margin-left: 10px;
165       }
166
167       .header-content ul li a.active::before {
168           width: 100px;
169           height: 2px;
170           background-color: #764ba2;
171           margin-left: 10px;
172       }
173
174       .header-content ul li a.active::before {
175           width: 100px;
176           height: 2px;
177           background-color: #764ba2;
178           margin-left: 10px;
179       }
180
181       .header-content ul li a.active::before {
182           width: 100px;
183           height: 2px;
184           background-color: #764ba2;
185           margin-left: 10px;
186       }
187
188       .header-content ul li a.active::before {
189           width: 100px;
190           height: 2px;
191           background-color: #764ba2;
192           margin-left: 10px;
193       }
194
195       .header-content ul li a.active::before {
196           width: 100px;
197           height: 2px;
198           background-color: #764ba2;
199           margin-left: 10px;
200       }
201
202       .header-content ul li a.active::before {
203           width: 100px;
204           height: 2px;
205           background-color: #764ba2;
206           margin-left: 10px;
207       }
208
209       .header-content ul li a.active::before {
210           width: 100px;
211           height: 2px;
212           background-color: #764ba2;
213           margin-left: 10px;
214       }
215
216       .header-content ul li a.active::before {
217           width: 100px;
218           height: 2px;
219           background-color: #764ba2;
220           margin-left: 10px;
221       }
222
223       .header-content ul li a.active::before {
224           width: 100px;
225           height: 2px;
226           background-color: #764ba2;
227           margin-left: 10px;
228       }
229
230       .header-content ul li a.active::before {
231           width: 100px;
232           height: 2px;
233           background-color: #764ba2;
234           margin-left: 10px;
235       }
236
237       .header-content ul li a.active::before {
238           width: 100px;
239           height: 2px;
240           background-color: #764ba2;
241           margin-left: 10px;
242       }
243
244       .header-content ul li a.active::before {
245           width: 100px;
246           height: 2px;
247           background-color: #764ba2;
248           margin-left: 10px;
249       }
250
251       .header-content ul li a.active::before {
252           width: 100px;
253           height: 2px;
254           background-color: #764ba2;
255           margin-left: 10px;
256       }
257
258       .header-content ul li a.active::before {
259           width: 100px;
260           height: 2px;
261           background-color: #764ba2;
262           margin-left: 10px;
263       }
264
265       .header-content ul li a.active::before {
266           width: 100px;
267           height: 2px;
268           background-color: #764ba2;
269           margin-left: 10px;
270       }
271
272       .header-content ul li a.active::before {
273           width: 100px;
274           height: 2px;
275           background-color: #764ba2;
276           margin-left: 10px;
277       }
278
279       .header-content ul li a.active::before {
280           width: 100px;
281           height: 2px;
282           background-color: #764ba2;
283           margin-left: 10px;
284       }
285
286       .header-content ul li a.active::before {
287           width: 100px;
288           height: 2px;
289           background-color: #764ba2;
290           margin-left: 10px;
291       }
292
293       .header-content ul li a.active::before {
294           width: 100px;
295           height: 2px;
296           background-color: #764ba2;
297           margin-left: 10px;
298       }
299
300       .header-content ul li a.active::before {
301           width: 100px;
302           height: 2px;
303           background-color: #764ba2;
304           margin-left: 10px;
305       }
306
307       .header-content ul li a.active::before {
308           width: 100px;
309           height: 2px;
310           background-color: #764ba2;
311           margin-left: 10px;
312       }
313
314       .header-content ul li a.active::before {
315           width: 100px;
316           height: 2px;
317           background-color: #764ba2;
318           margin-left: 10px;
319       }
320
321       .header-content ul li a.active::before {
322           width: 100px;
323           height: 2px;
324           background-color: #764ba2;
325           margin-left: 10px;
326       }
327
328       .header-content ul li a.active::before {
329           width: 100px;
330           height: 2px;
331           background-color: #764ba2;
332           margin-left: 10px;
333       }
334
335       .header-content ul li a.active::before {
336           width: 100px;
337           height: 2px;
338           background-color: #764ba2;
339           margin-left: 10px;
340       }
341
342       .header-content ul li a.active::before {
343           width: 100px;
344           height: 2px;
345           background-color: #764ba2;
346           margin-left: 10px;
347       }
348
349       .header-content ul li a.active::before {
350           width: 100px;
351           height: 2px;
352           background-color: #764ba2;
353           margin-left: 10px;
354       }
355
356       .header-content ul li a.active::before {
357           width: 100px;
358           height: 2px;
359           background-color: #764ba2;
360           margin-left: 10px;
361       }
362
363       .header-content ul li a.active::before {
364           width: 100px;
365           height: 2px;
366           background-color: #764ba2;
367           margin-left: 10px;
368       }
369
370       .header-content ul li a.active::before {
371           width: 100px;
372           height: 2px;
373           background-color: #764ba2;
374           margin-left: 10px;
375       }
376
377       .header-content ul li a.active::before {
378           width: 100px;
379           height: 2px;
380           background-color: #764ba2;
381           margin-left: 10px;
382       }
383
384       .header-content ul li a.active::before {
385           width: 100px;
386           height: 2px;
387           background-color: #764ba2;
388           margin-left: 10px;
389       }
390
391       .header-content ul li a.active::before {
392           width: 100px;
393           height: 2px;
394           background-color: #764ba2;
395           margin-left: 10px;
396       }
397
398       .header-content ul li a.active::before {
399           width: 100px;
400           height: 2px;
401           background-color: #764ba2;
402           margin-left: 10px;
403       }
404
405       .header-content ul li a.active::before {
406           width: 100px;
407           height: 2px;
408           background-color: #764ba2;
409           margin-left: 10px;
410       }
411
412       .header-content ul li a.active::before {
413           width: 100px;
414           height: 2px;
415           background-color: #764ba2;
416           margin-left: 10px;
417       }
418
419       .header-content ul li a.active::before {
420           width: 100px;
421           height: 2px;
422           background-color: #764ba2;
423           margin-left: 10px;
424       }
425
426       .header-content ul li a.active::before {
427           width: 100px;
428           height: 2px;
429           background-color: #764ba2;
430           margin-left: 10px;
431       }
432
433       .header-content ul li a.active::before {
434           width: 100px;
435           height: 2px;
436           background-color: #764ba2;
437           margin-left: 10px;
438       }
439
440       .header-content ul li a.active::before {
441           width: 100px;
442           height: 2px;
443           background-color: #764ba2;
444           margin-left: 10px;
445       }
446
447       .header-content ul li a.active::before {
448           width: 100px;
449           height: 2px;
450           background-color: #764ba2;
451           margin-left: 10px;
452       }
453
454       .header-content ul li a.active::before {
455           width: 100px;
456           height: 2px;
457           background-color: #764ba2;
458           margin-left: 10px;
459       }
460
461       .header-content ul li a.active::before {
462           width: 100px;
463           height: 2px;
464           background-color: #764ba2;
465           margin-left: 10px;
466       }
467
468       .header-content ul li a.active::before {
469           width: 100px;
470           height: 2px;
471           background-color: #764ba2;
472           margin-left: 10px;
473       }
474
475       .header-content ul li a.active::before {
476           width: 100px;
477           height: 2px;
478           background-color: #764ba2;
479           margin-left: 10px;
480       }
481
482       .header-content ul li a.active::before {
483           width: 100px;
484           height: 2px;
485           background-color: #764ba2;
486           margin-left: 10px;
487       }
488
489       .header-content ul li a.active::before {
490           width: 100px;
491           height: 2px;
492           background-color: #764ba2;
493           margin-left: 10px;
494       }
495
496       .header-content ul li a.active::before {
497           width: 100px;
498           height: 2px;
499           background-color: #764ba2;
500           margin-left: 10px;
501       }
502
503       .header-content ul li a.active::before {
504           width: 100px;
505           height: 2px;
506           background-color: #764ba2;
507           margin-left: 10px;
508       }
509
510       .header-content ul li a.active::before {
511           width: 100px;
512           height: 2px;
513           background-color: #764ba2;
514           margin-left: 10px;
515       }
516
517       .header-content ul li a.active::before {
518           width: 100px;
519           height: 2px;
520           background-color: #764ba2;
521           margin-left: 10px;
522       }
523
524       .header-content ul li a.active::before {
525           width: 100px;
526           height: 2px;
527           background-color: #764ba2;
528           margin-left: 10px;
529       }
530
531       .header-content ul li a.active::before {
532           width: 100px;
533           height: 2px;
534           background-color: #764ba2;
535           margin-left: 10px;
536       }
537
538       .header-content ul li a.active::before {
539           width: 100px;
540           height: 2px;
541           background-color: #764ba2;
542           margin-left: 10px;
543       }
544
545       .header-content ul li a.active::before {
546           width: 100px;
547           height: 2px;
548           background-color: #764ba2;
549           margin-left: 10px;
550       }
551
552       .header-content ul li a.active::before {
553           width: 100px;
554           height: 2px;
555           background-color: #764ba2;
556           margin-left: 10px;
557       }
558
559       .header-content ul li a.active::before {
560           width: 100px;
561           height: 2px;
562           background-color: #764ba2;
563           margin-left: 10px;
564       }
565
566       .header-content ul li a.active::before {
567           width: 100px;
568           height: 2px;
569           background-color: #764ba2;
570           margin-left: 10px;
571       }
572
573       .header-content ul li a.active::before {
574           width: 100px;
575           height: 2px;
576           background-color: #764ba2;
577           margin-left: 10px;
578       }
579
580       .header-content ul li a.active::before {
581           width: 100px;
582           height: 2px;
583           background-color: #764ba2;
584           margin-left: 10px;
585       }
586
587       .header-content ul li a.active::before {
588           width: 100px;
589           height: 2px;
590           background-color: #764ba2;
591           margin-left: 10px;
592       }
593
594       .header-content ul li a.active::before {
595           width: 100px;
596           height: 2px;
597           background-color: #764ba2;
598           margin-left: 10px;
599       }
599
600       .header-content ul li a.active::before {
601           width: 100px;
602           height: 2px;
603           background-color: #764ba2;
604           margin-left: 10px;
605       }
606
607       .header-content ul li a.active::before {
608           width: 100px;
609           height: 2px;
610           background-color: #764ba2;
611           margin-left: 10px;
612       }
613
614       .header-content ul li a.active::before {
615           width: 100px;
616           height: 2px;
617           background-color: #764ba2;
618           margin-left: 10px;
619       }
620
621       .header-content ul li a.active::before {
622           width: 100px;
623           height: 2px;
624           background-color: #764ba2;
625           margin-left: 10px;
626       }
627
628       .header-content ul li a.active::before {
629           width: 100px;
630           height: 2px;
631           background-color: #764ba2;
632           margin-left: 10px;
633       }
634
635       .header-content ul li a.active::before {
636           width: 100px;
637           height: 2px;
638           background-color: #764ba2;
639           margin-left: 10px;
640       }
641
642       .header-content ul li a.active::before {
643           width: 100px;
644           height: 2px;
645           background-color: #764ba2;
646           margin-left: 10px;
647       }
648
649       .header-content ul li a.active::before {
650           width: 100px;
651           height: 2px;
652           background-color: #764ba2;
653           margin-left: 10px;
654       }
655
656       .header-content ul li a.active::before {
657           width: 100px;
658           height: 2px;
659           background-color: #764ba2;
660           margin-left: 10px;
661       }
662
663       .header-content ul li a.active::before {
664           width: 100px;
665           height: 2px;
666           background-color: #764ba2;
667           margin-left: 10px;
668       }
669
670       .header-content ul li a.active::before {
671           width: 100px;
672           height: 2px;
673           background-color: #764ba2;
674           margin-left: 10px;
675       }
676
677       .header-content ul li a.active::before {
678           width: 100px;
679           height: 2px;
680           background-color: #764ba2;
681           margin-left: 10px;
682       }
683
684       .header-content ul li a.active::before {
685           width: 100px;
686           height: 2px;
687           background-color: #764ba2;
688           margin-left: 10px;
689       }
690
691       .header-content ul li a.active::before {
692           width: 100px;
693           height: 2px;
694           background-color: #764ba2;
695           margin-left: 10px;
696       }
697
698       .header-content ul li a.active::before {
699           width: 100px;
700           height: 2px;
701           background-color: #764ba2;
702           margin-left: 10px;
703       }
704
705       .header-content ul li a.active::before {
706           width: 100px;
707           height: 2px;
708           background-color: #764ba2;
709           margin-left: 10px;
710       }
711
712       .header-content ul li a.active::before {
713           width: 100px;
714           height: 2px;
715           background-color: #764ba2;
716           margin-left: 10px;
717       }
718
719       .header-content ul li a.active::before {
720           width: 100px;
721           height: 2px;
722           background-color: #764ba2;
723           margin-left: 10px;
724       }
725
726       .header-content ul li a.active::before {
727           width: 100px;
728           height: 2px;
729           background-color: #764ba2;
730           margin-left: 10px;
731       }
732
733       .header-content ul li a.active::before {
734           width: 100px;
735           height: 2px;
736           background-color: #764ba2;
737           margin-left: 10px;
738       }
739
740       .header-content ul li a.active::before {
741           width: 100px;
742           height: 2px;
743           background-color: #764ba2;
744           margin-left: 10px;
745       }
746
747       .header-content ul li a.active::before {
748           width: 100px;
749           height: 2px;
750           background-color: #764ba2;
751           margin-left: 10px;
752       }
753
754       .header-content ul li a.active::before {
755           width: 100px;
756           height: 2px;
757           background-color: #764ba2;
758           margin-left: 10px;
759       }
760
761       .header-content ul li a.active::before {
762           width: 100px;
763           height: 2px;
764           background-color: #764ba2;
765           margin-left: 10px;
766       }
767
768       .header-content ul li a.active::before {
769           width: 100px;
770           height: 2px;
771           background-color: #764ba2;
772           margin-left: 10px;
773       }
774
775       .header-content ul li a.active::before {
776           width: 100px;
777           height: 2px;
778           background-color: #764ba2;
779           margin-left: 10px;
780       }
781
782       .header-content ul li a.active::before {
783           width: 100px;
784           height: 2px;
785           background-color: #764ba2;
786           margin-left: 10px;
787       }
788
789       .header-content ul li a.active::before {
790           width: 100px;
791           height: 2px;
792           background-color: #764ba2;
793           margin-left: 10px;
794       }
795
796       .header-content ul li a.active::before {
797           width: 100px;
798           height: 2px;
799           background-color: #764ba2;
800           margin-left: 10px;
801       }
802
803       .header-content ul li a.active::before {
804           width: 100px;
805           height: 2px;
806           background-color: #764ba2;
807           margin-left: 10px;
808       }
809
810       .header-content ul li a.active::before {
811           width: 100px;
812           height: 2px;
813           background-color: #764ba2;
814           margin-left: 10px;
815       }
816
817       .header-content ul li a.active::before {
818           width: 100px;
819           height: 2px;
820           background-color: #764ba2;
821           margin-left: 10px;
822       }
823
824       .header-content ul li a.active::before {
825           width: 100px;
826           height: 2px;
827           background-color: #764ba2;
828           margin-left: 10px;
829       }
830
831       .header-content ul li a.active::before {
832           width: 100px;
833           height: 2px;
834           background-color: #764ba2;
835           margin-left: 10px;
836       }
837
838       .header-content ul li a.active::before {
839           width: 100px;
840           height: 2px;
841           background-color: #764ba2;
842           margin-left: 10px;
843       }
844
845       .header-content ul li a.active::before {
846           width: 100px;
847           height: 2px;
848           background-color: #764ba2;
849           margin-left: 10px;
850       }
851
852       .header-content ul li a.active::before {
853           width: 100px;
854           height: 2px;
855           background-color: #764ba2;
856           margin-left: 10px;
857       }
858
859       .header-content ul li a.active::before {
860           width: 100px;
861           height: 2px;
862           background-color: #764ba2;
863           margin-left: 10px;
864       }
865
866       .header-content ul li a.active::before {
867           width: 100px;
868           height: 2px;
869           background-color: #764ba2;
870           margin-left: 10px;
871       }
872
873       .header-content ul li a.active::before {
874           width: 100px;
875           height: 2px;
876           background-color: #764ba2;
877           margin-left: 10px;
878       }
879
880       .header-content ul li a.active::before {
881           width: 100px;
882           height: 2px;
883           background-color: #764ba2;
884           margin-left: 10px;
885       }
886
887       .header-content ul li a.active::before {
888           width: 100px;
889           height: 2px;
890           background-color: #764ba2;
891           margin-left: 10px;
892       }
893
894       .header-content ul li a.active::before {
895           width: 100px;
896           height: 2px;
897           background-color: #764ba2;
898           margin-left: 10px;
899       }
899
900       .header-content ul li a.active::before {
901           width: 100px;
902           height: 2px;
903           background-color: #764ba2;
904           margin-left: 10px;
905       }
906
907       .header-content ul li a.active::before {
908           width: 100px;
909           height: 2px;
910           background-color: #764ba2;
911           margin-left: 10px;
912       }
913
914       .header-content ul li a.active::before {
915           width: 100px;
916           height: 2px;
917           background-color: #764ba2;
918           margin-left: 10px;
919       }
920
921       .header-content ul li a.active::before {
922           width: 100px;
923           height: 2px;
924           background-color: #764ba2;
925           margin-left: 10px;
926       }
927
928       .header-content ul li a.active::before {
929           width: 100px;
930           height: 2px;
931           background-color: #764ba2;
932           margin-left: 10px;
933       }
934
935       .header-content ul li a.active::before {
936           width: 100px;
937           height: 2px;
938           background-color: #764ba2;
939           margin-left: 10px;
940       }
941
942       .header-content ul li a.active::before {
943           width: 100px;
944           height: 2px;
945           background-color: #764ba2;
946           margin-left: 10px;
947       }
948
949       .header-content ul li a.active::before {
950           width: 100px;
951           height: 2px;
952           background-color: #764ba2;
953           margin-left: 10px;
954       }
955
956       .header-content ul li a.active::before {
957           width: 100px;
958           height: 2px;
959           background-color: #764ba2;
960           margin-left: 10px;
961       }
962
963       .header-content ul li a.active::before {
964           width: 100px;
965           height: 2px;
966           background-color: #764ba2;
967           margin-left: 10px;
968       }
969
970       .header-content ul li a.active::before {
971           width: 100px;
972           height: 2px;
973           background-color: #764ba2;
974           margin-left: 10px;
975       }
976
977       .header-content ul li a.active::before {
978           width: 100px;
979           height: 2px;
980           background-color: #764ba2;
981           margin-left: 10px;
982       }
983
984       .header-content ul li a.active::before {
985           width: 100px;
986           height: 2px;
987           background-color: #764ba2;
988           margin-left: 10px;
989       }
989
990       .header-content ul li a.active::before {
991           width: 100px;
992           height: 2px;
993           background-color: #764ba2;
994           margin-left: 10px;
995       }
996
997       .header-content ul li a.active::before {
998           width: 100px;
999           height: 2px;
1000          background-color: #764ba2;
1001          margin-left: 10px;
1002      }
1002
1003       .header-content ul li a.active::before {
1004           width: 100px;
1005           height: 2px;
1006           background-color: #764ba2;
1007           margin-left: 10px;
1008       }
1009
1010       .header-content ul li a.active::before {
1011           width: 100px;
1012           height: 2px;
1013           background-color: #764ba2;
1014           margin-left: 10px;
1015       }
1016
1017       .header-content ul li a.active::before {
1018           width: 100px;
1019           height: 2px;
1020           background-color: #764ba2;
1021           margin-left: 10px;
1022       }
1023
1024       .header-content ul li a.active::before {
1025           width: 100px;
1026           height: 2px;
1027           background-color: #764ba2;
1028           margin-left: 10px;
1029       }
1030
1031       .header-content ul li a.active::before {
1032           width: 100px;
1033           height: 2px;
1034           background-color: #7
```