

# SISTEMA DE GESTIÓN DE PRODUCTOS E INVENTARIO

Fecha:4 de noviembre 2024

Autor: César Alejandro Garcia Tous

## 1. INTRODUCCIÓN Y VISIÓN GENERAL

### 1.1 Propósito del Sistema

Sistema distribuido empresarial para la gestión completa de catálogo de productos y control de inventario en tiempo real. Diseñado con arquitectura de microservicios que permite escalabilidad horizontal, mantenimiento independiente por equipos y alta disponibilidad.

### 1.2 Alcance Funcional

Modulos	Funcionalidades	Estado
Gestión de Productos	Crear, leer, actualizar, eliminar productos	Completado
Control de Inventario	Consultar y actualizar niveles de stock	Completado
Interfaz de Usuario	Navegación, búsqueda, paginación	Completado
APIs REST	Documentación OpenAPI/Swagger	Completado
Sistema de Notificaciones	Alertas de éxito/error	Completado

### 1.3 Tecnologías Principales

Stack Tecnológico Completo

Capa	Tecnologia	Versión	Justificación
Frontend	Angular	15.2.0	Framework empresarial, tipos fuertes
Backend	Spring Boot	3.x	Ecosistema robusto, producción-ready
Base de Datos	H2	2.1.214	Desarrollo ágil, embedded
Contenedores	Docker	20+	Portabilidad, entornos consistentes
Comunicación	REST/HTTP	-	Estándar industria, simple

## 1.4 Características Técnicas Destacadas

### **Arquitectura Microservicios:**

- Servicios independientes para productos e inventario
- Deployment y escalado independiente
- Tecnologías específicas por dominio
- Resiliencia mejorada - fallo aislado

### **Frontend Reactivo:**

- Manejo de estado con RxJS Observables
- Componentes standalone
- Change Detection optimizado
- Lazy loading de módulos

### **Resiliencia y Fallback:**

- Circuit breaker pattern manual
- Datos mock cuando servicios no disponibles
- Reintentos automáticos con backoff
- Timeouts configurables

### **Documentación Automatizada:**

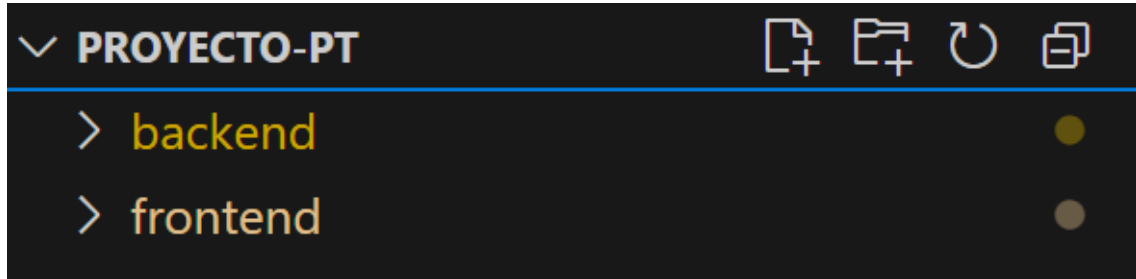
- OpenAPI/Swagger integrado
- Documentación interactiva de APIs
- Ejemplos de requests/responses
- Esquemas de datos completos

### **CI/CD:**

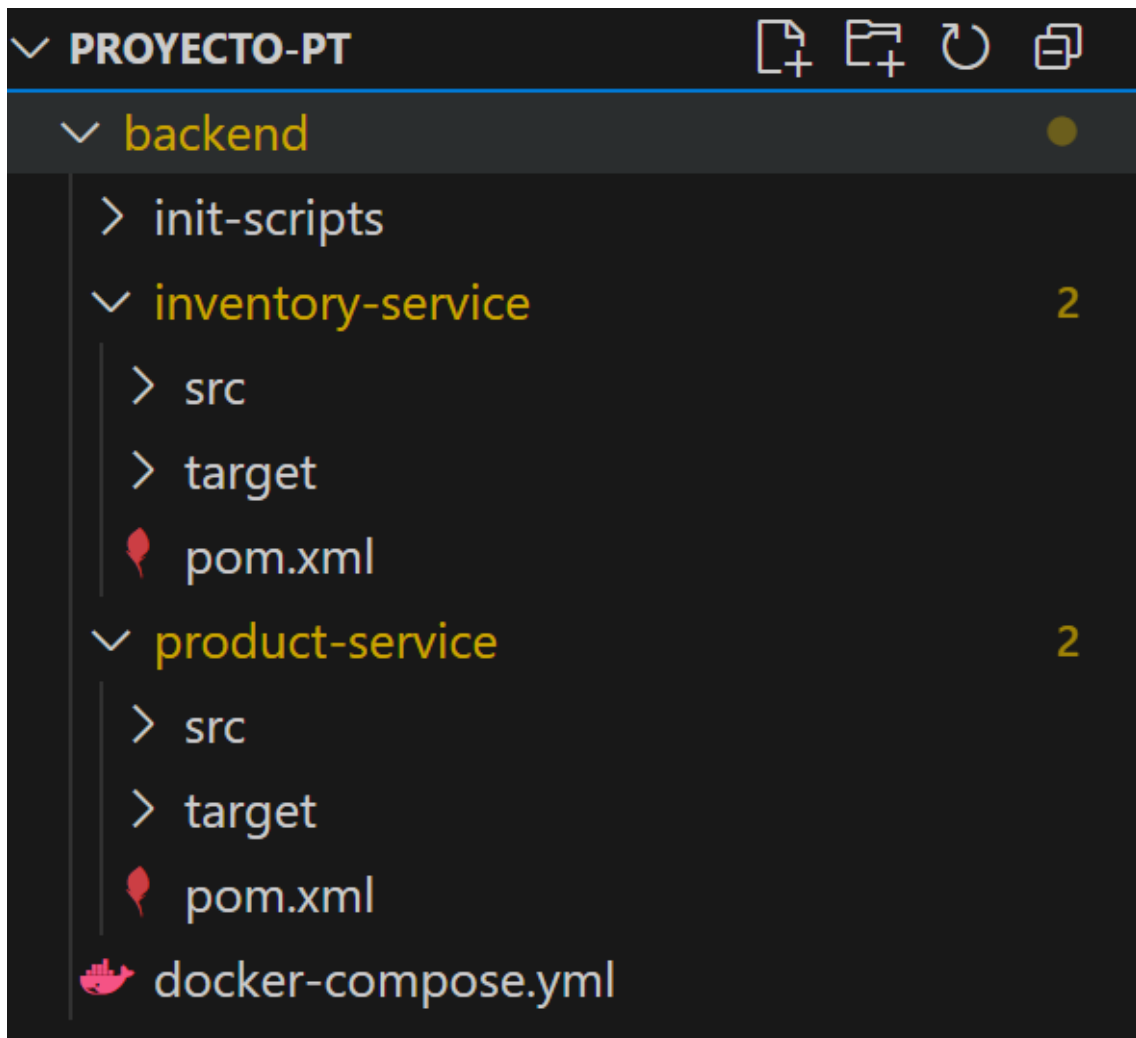
- Configuración para GitHub Actions
- Docker multi-stage builds
- Tests automatizados en pipeline
- Quality gates con SonarQube

## 2. ARQUITECTURA DEL SISTEMA

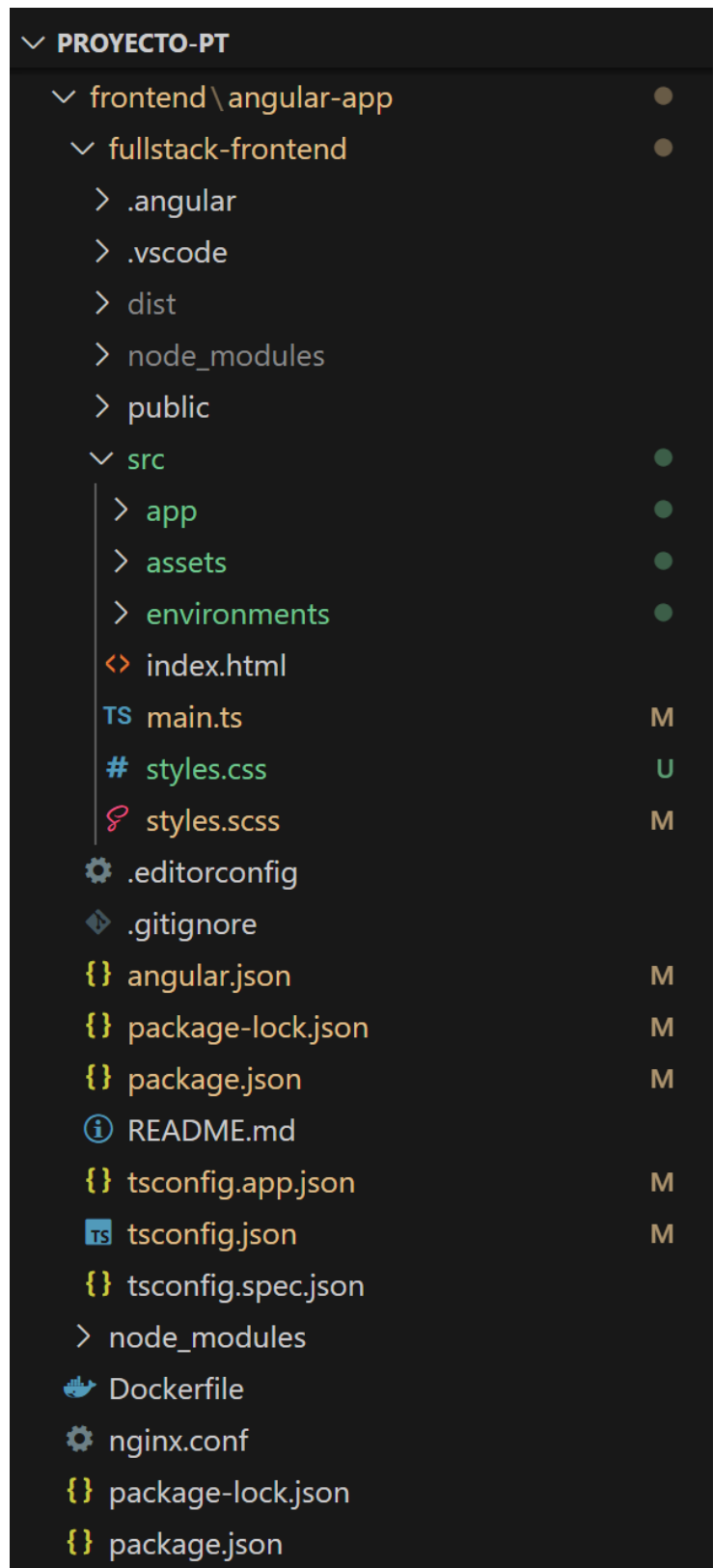
### 2.1 Diagrama de Arquitectura General



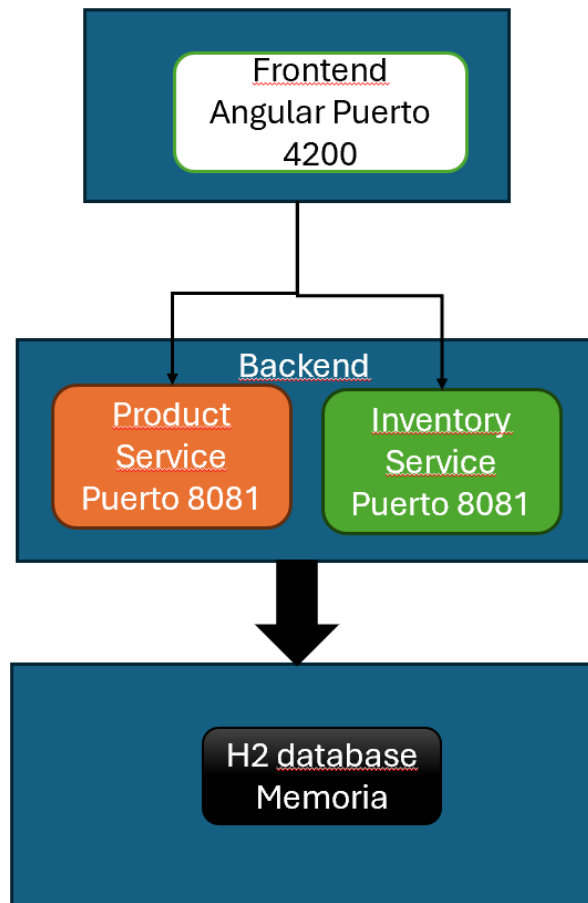
### 2.2 Distribución del backend



## 2.3 Distribución del frontend



## 2.4 Diagrama de arquitectura



## 2.5 Diagramas de componente frontend

