

Requerimientos Funcionales:

1. Autenticación de usuarios con roles de cliente y administrador.
2. Acceso al catálogo de productos desde una interfaz interactiva.
3. Visualización detallada de productos incluyendo imagen, descripción y precio.
4. Sistema de calificación de productos basado en estrellas.
5. Opción para agregar productos a la lista de pedidos.
6. Realización de pedidos con opción de asignar número de mesa.
7. Página de gestión de productos para el administrador (CRUD).
8. Sistema de notificaciones en tiempo real para nuevos pedidos.
9. Registro de interacciones de usuarios con productos.
- 10.Recomendaciones personalizadas basadas en el historial de interacciones.
- 11.Algoritmo de filtrado colaborativo para sugerencias de productos.
- 12.Funcionalidad de búsqueda avanzada utilizando procesamiento de lenguaje natural.
- 13.Integración de servicios independientes para recomendaciones, popularidad y búsqueda.
- 14.Gestión y almacenamiento de pedidos realizados por los clientes.
- 15.Pantalla de inicio personalizada para clientes y administradores.
- 16.Registro y análisis de las interacciones en una base de datos orientada a grafos (Neo4j).
- 17.Presentación de productos populares basados en estadísticas de uso.
- 18.Guardado de historial de pedidos y visualización en el panel de administración.
- 19.Posibilidad de agregar y editar categorías de productos en el catálogo.
- 20.Recepción de notificaciones sobre pedidos completados para administradores.

Requerimientos No Funcionales:

1. Escalabilidad del sistema para manejar un número creciente de usuarios y productos.
2. Modularidad para facilitar actualizaciones y mantenimientos sin afectar otros servicios.
3. Compatibilidad con todos los navegadores web principales.
4. Seguridad en la autenticación y gestión de pedidos.
5. Respuesta en tiempo real para recomendaciones basadas en interacciones recientes.
6. Despliegue utilizando Docker para asegurar consistencia en diferentes entornos.
7. Rendimiento óptimo en la entrega de recomendaciones, incluso bajo alta carga.
8. Uso de Nginx como API Gateway para gestionar y balancear el tráfico.
9. Alta disponibilidad del sistema para minimizar tiempos de inactividad.
- 10.Documentación técnica detallada para desarrolladores y manual de usuario.
- 11.Facilidad de uso y navegación intuitiva para los usuarios finales.
- 12.Integración continua para facilitar la implementación de nuevas funciones.
- 13.Cumplimiento con estándares de accesibilidad web.
- 14.Tiempo de respuesta rápido en la búsqueda y visualización de productos.
- 15.Protección de datos sensibles con encriptación adecuada.
- 16.Capacidad para manejar conexiones simultáneas de múltiples usuarios.