

Documentación Técnica del Proyecto - Sales Date Prediction

1. Introducción

Este documento describe la solución desarrollada para la prueba técnica del sistema 'Sales Date Prediction'. El objetivo principal es permitir la creación de órdenes y predecir la próxima orden por cliente, utilizando una arquitectura moderna basada en Angular para el frontend, ASP.NET Core para el backend, SQL Server como base de datos y D3.js para visualización de datos.

2. Arquitectura del Proyecto

El sistema está compuesto por los siguientes módulos:

- Frontend SPA en Angular 17+.
- Backend RESTful API en ASP.NET Core.
- Base de datos SQL Server.
- Visualización con D3.js en Vanilla JavaScript.

3. Base de Datos

Se utilizó un script llamado 'DBSetup.sql' para crear la base de datos 'StoreSample'.

Las tablas principales incluyen: Orders, Customers, Employees, Shippers, Products.

Consultas implementadas:

- Predicción de próxima orden por cliente.
- Órdenes por cliente.
- Listado de empleados, transportistas y productos.
- Inserción de nueva orden con detalle de producto.

4. Backend (.NET Core)

El backend sigue una arquitectura en capas:

- Controllers: exponen los endpoints REST.
- Application: lógica de negocio y DTOs.
- Domain: entidades y contratos.
- Infrastructure: acceso a datos y servicios externos.

Validaciones:

- Uso de anotaciones de data y ModelState.
- Principios SOLID aplicados.

Pruebas unitarias:

- Se implementaron pruebas unitarias utilizando Microsoft Unit Testing Framework.
- Se cubren los servicios principales y lógica de negocio.

5. Frontend (Angular)

La aplicación SPA fue desarrollada en Angular 17+ con componentes standalone.

Estructura del proyecto:

- app/core: servicios, modelos y lógica compartida.
- app/features: componentes funcionales.
- app/shared: componentes reutilizables.
- assets: recursos estáticos.
- environments: configuración por entorno.

Vistas implementadas:

- Sales Date Prediction View: tabla con clientes, última orden y predicción.
- Orders View: listado de órdenes por cliente.
- New Order Form: formulario para crear nueva orden con validaciones.

6. Visualización con D3.js

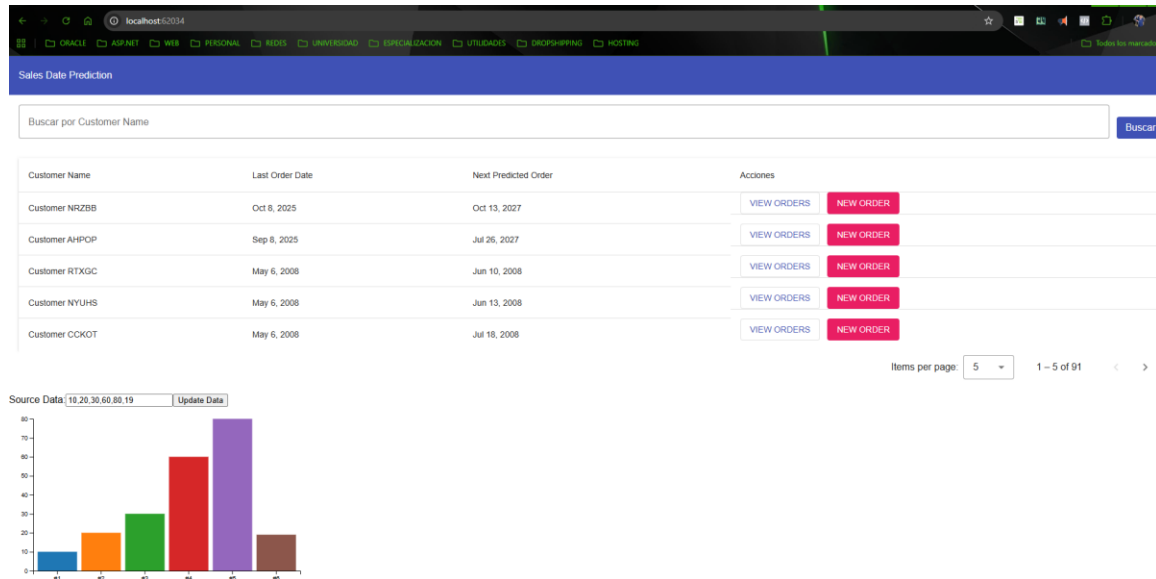
Esta sección presenta la funcionalidad de visualización de datos implementada en el proyecto utilizando la biblioteca D3.js en una aplicación Angular. El objetivo es permitir al usuario ingresar una serie de números y generar un gráfico de barras interactivo que represente visualmente esos datos.

Estructura de la funcionalidad

- **Source Data:** Campo de texto donde el usuario ingresa números enteros separados por comas (por ejemplo: 4,8,15,16).
- **Botón "Update Data":** Al hacer clic, se procesa la entrada y se actualiza el gráfico.
- **Colores Alternos:** Las barras se muestran con colores distintos, limitados a una paleta de 5 colores, evitando que barras contiguas compartan el mismo color.

Consideraciones Técnicas

- La funcionalidad fue desarrollada como un componente standalone en Angular.
- Se utilizó D3.js para manipular el DOM SVG y renderizar el gráfico.
- Se aplicaron buenas prácticas de validación de entrada y separación de responsabilidades.
- El componente permite reutilización y fácil integración en otras vistas del sistema.



8. Capturas de Funcionamiento del Aplicativo

A continuación, se presentan imágenes que ilustran el funcionamiento del sistema *Sales Date Predicción*. Estas capturas muestran las vistas principales de la aplicación, incluyendo la predicción de órdenes, la gestión de órdenes por cliente y el formulario para crear nuevas órdenes.

