



Prueba Técnica
Desarrollador

Tabla de contenidos

¿Como realizar la entrega?	3
1. Requerimiento.....	3
2. Base de datos	5
Sales Date Prediction	5
Get Client Orders	5
Get employees.....	6
Get Shippers.....	6
Get Products	6
Add New Order	6
3. Web API.....	8
4. Web App	9
Sales Date Prediction View.....	9
Orders View	9
New Order Form	11
5. Graficando con D3	12

¿Como realizar la entrega?

A vuelta de correo enviar el enlace a un repositorio en el servicio de preferencia (GitHub, Azure Devops, Bitbucket, etc.)

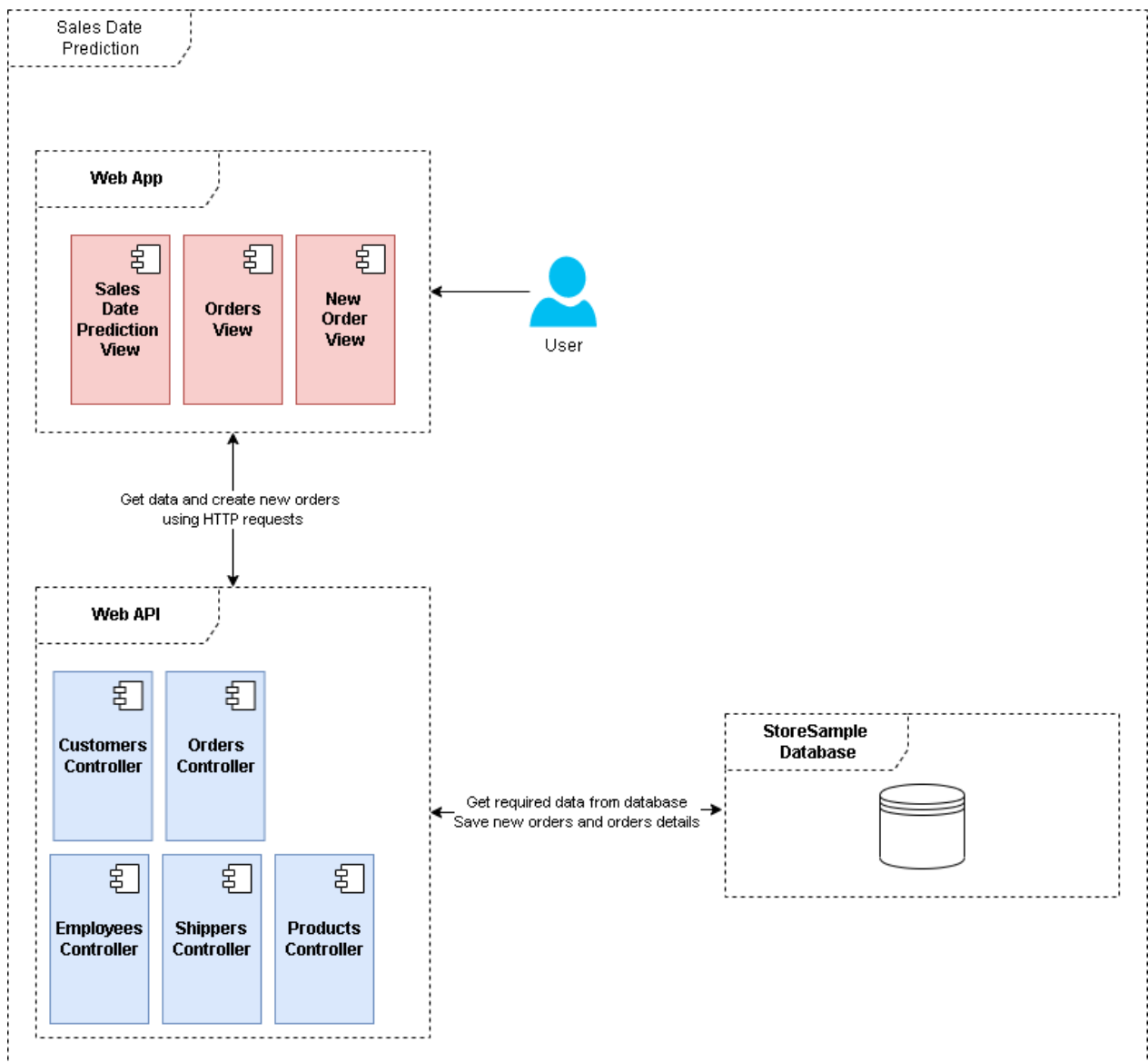
El repositorio debe incluir README que incluya la siguiente información:

- Información relevante para hacer build y ejecutar el proyecto.
- Cualquier información relevante sobre la prueba que se quiera mencionar para ser tomada en cuenta durante la revisión.
- Breve explicación sobre como se ejecutó la prueba.

1. Requerimiento

Desarrollar una aplicación llamada “**Sales Date Prediction**” que tiene como objetivo crear órdenes y predecir cuándo ocurrirá la próxima orden por cliente de acuerdo con los registros almacenados en la base de datos. Los componentes por desarrollar los podemos visualizar en el siguiente diagrama de arquitectura:

Sales Date Prediction Architecture Diagram



2. Base de datos

Prerrequisitos:

- Tener una instancia de SQL Server
- Ejecutar el script llamado "DBSetup.sql", el script crea la base de datos StoreSample, crea esquemas, crea tablas, crea índices, inserta registros en las tablas y crea vistas.

La aplicación requiere construir sentencias DML que serán usadas por el componente de Web API (descrito en la siguiente sección)

Sales Date Prediction

- Crear una consulta que retorne por cliente la fecha de cuando ocurrirá la próxima orden, esta fecha se calcula sumando el promedio de días entre las órdenes existentes del cliente a la fecha de la última orden.
 - La base de datos contiene el histórico de registros necesarios para el cálculo de la predicción.
 - Columnas:
 - Customer Name: Nombre del cliente
 - Last Order Date: La fecha de la última orden realizada por el cliente
 - Next Predicted Order: Fecha posible de la siguiente orden (fecha ultima orden + promedio de días entre ordenes anteriores)
- Tablas a consultar: Orders, Customers

Resultado esperado:

CustomerName	LastOrderDate	NextPredictedOrder
Customer AHPOP	2008-02-04 00:00:00.000	2008-03-23 00:00:00.000
Customer AHXHT	2008-05-05 00:00:00.000	2008-08-09 00:00:00.000
Customer AZJED	2008-04-09 00:00:00.000	2008-05-20 00:00:00.000

Get Client Orders

- Crear una consulta que retorne las ordenes de un cliente
- Tabla: Orders
- Columnas
 - Orderid
 - Requireddate
 - Shippeddate
 - Shipname

- Shipaddress
- Shipcity

Get employees

- Crear una consulta que retorne todos los empleados
- Tabla: Employees
- Columnas
 - Empid
 - FullName: concatenar firstname y lastname

Get Shippers

- Crear una consulta que retorne todos los transportistas
- Tabla: Shippers
- Columnas
 - Shipperid
 - Companyname

Get Products

- Crear una consulta que retorne todos los productos
- Tabla: Products
- Columnas
 - Productid
 - Productname

Add New Order

- Crear un script usando Transact SQL que permita ingresar una nueva orden y agregar un producto a dicha orden
- Columnas de la Orden
 - Empid: Employee Id
 - Shipperid
 - Shipname
 - Shipaddress
 - Shipcity
 - Orderdate
 - Requireddate
 - Shippeddate
 - Freight
 - Shipcountry
- Columnas de OrderDetails
 - Orderid: Id de la orden creada anteriormente

- Productid
- Unitprice
- Qty
- Discount

3. Web API

Crear un proyecto Web API (.Net Core o .Net Framework) que exponga los siguientes servicios REST a través de controladores:

- Listar clientes con fecha de ultima orden y fecha de posible orden
- Listar ordenes por cliente
- Listar todos los empleados
- Listar todos los transportistas (Shippers)
- Listar todos los productos
- Crear una orden nueva con un producto

Recomendaciones:

- Aplicar principios SOLID
- Agregar pruebas unitarias usando el framework de su preferencia (nUnit, xUnit Microsoft unit testing framework)

4. Web App

Construir un SPA (Single Page Application) Angular 17+ que permita visualizar la siguiente información:

Sales Date Prediction View

Crear una vista donde se muestre la lista de clientes con su respectivo nombre, fecha de ultima orden y fecha de posible siguiente orden.

Sales Date Prediction App

Customer Name

Customer Name ↑

Last Order Date

Next Predicted Order

Customer AHPOP	2/4/2008	3/23/2008	VIEW ORDERS	NEW ORDER
Customer AHXHT	5/5/2008	8/9/2008	VIEW ORDERS	NEW ORDER
Customer AZJED	4/9/2008	5/20/2008	VIEW ORDERS	NEW ORDER
Customer BSVAR	2/16/2008	6/23/2008	VIEW ORDERS	NEW ORDER
Customer CCFIZ	4/23/2008	7/4/2008	VIEW ORDERS	NEW ORDER
Customer CCKOT	5/6/2008	7/11/2008	VIEW ORDERS	NEW ORDER
Customer CQRAA	3/16/2008	5/13/2008	VIEW ORDERS	NEW ORDER
Customer CYZTN	4/27/2008	5/30/2008	VIEW ORDERS	NEW ORDER
Customer DVFMB	4/13/2008	6/23/2008	VIEW ORDERS	NEW ORDER
Customer EEALV	4/24/2008	5/29/2008	VIEW ORDERS	NEW ORDER

Rows per page:

10

1-10 of 89

Requerimientos Funcionales:

- Configurar esta vista como el home de la aplicación.
- Cargar la lista de clientes en una tabla cuando la vista sea cargada
- La tabla debe tener paginación y permitir el ordenamiento desde cualquier columna.
- Cada registro de la tabla debe tener los siguientes botones:
 - **VIEW ORDERS:** Al dar click en este botón se debe abrir un modal con la vista “Orders View”, mostrando las órdenes del cliente seleccionado.
 - **NEW ORDER:** Al dar click en este botón se debe abrir un modal con el formulario “New Order”, desde donde se puede crear una nueva orden relacionada al cliente seleccionado.
- EL control de búsqueda debe filtrar la información de la tabla por el campo Customer Name desde el lado del SERVIDOR.

Requerimientos No Funcionales:

- La vista debe lucir como se muestra en el mockup

Orders View

Crear una vista donde se muestre una lista de órdenes del cliente seleccionado

Customer AHPOP - Orders					
Order ID	Required Date	Shipped Date	Ship Name	Ship Address	Ship City
10359	12/19/2006, 12:00:00 AM	11/26/2006, 12:00:00 AM	Ship to 72-C	1234 Wadhurst Rd.	London
10377	1/6/2007, 12:00:00 AM	12/13/2006, 12:00:00 AM	Ship to 72-C	1234 Wadhurst Rd.	London
10388	1/16/2007, 12:00:00 AM	12/20/2006, 12:00:00 AM	Ship to 72-C	1234 Wadhurst Rd.	London
10472	4/9/2007, 12:00:00 AM	3/19/2007, 12:00:00 AM	Ship to 72-A	0123 Wadhurst Rd.	London
10523	5/29/2007, 12:00:00 AM	5/30/2007, 12:00:00 AM	Ship to 72-C	1234 Wadhurst Rd.	London
10547	6/20/2007, 12:00:00 AM	6/2/2007, 12:00:00 AM	Ship to 72-C	1234 Wadhurst Rd.	London
10800	1/23/2008, 12:00:00 AM	1/5/2008, 12:00:00 AM	Ship to 72-C	1234 Wadhurst Rd.	London
10804	1/27/2008, 12:00:00 AM	1/7/2008, 12:00:00 AM	Ship to 72-C	1234 Wadhurst Rd.	London
10869	3/4/2008, 12:00:00 AM	2/9/2008, 12:00:00 AM	Ship to 72-A	0123 Wadhurst Rd.	London
			Rows per page:	10	1-9 of 9
CLOSE					

Requerimientos Funcionales:

- La tabla debe tener paginación y permitir el ordenamiento desde cualquier columna.

Requerimientos No Funcionales:

- La vista debe lucir como se muestra en el mockup

New Order Form

Crear un formulario donde el usuario pueda crear nuevas Órdenes con su respectivo detalle (un producto).

Customer AHPOP - New Order

Order

Employee*

Shipper*

Ship Name*

Ship Address*

Ship City*

Ship Country*

Order Date*

Required Date*

Shipped Date*

\$ Freight*

Order Details

Product*

\$ Unit Price*

Quantity*

Discount*

CLOSE

SAVE

Requerimientos Funcionales:

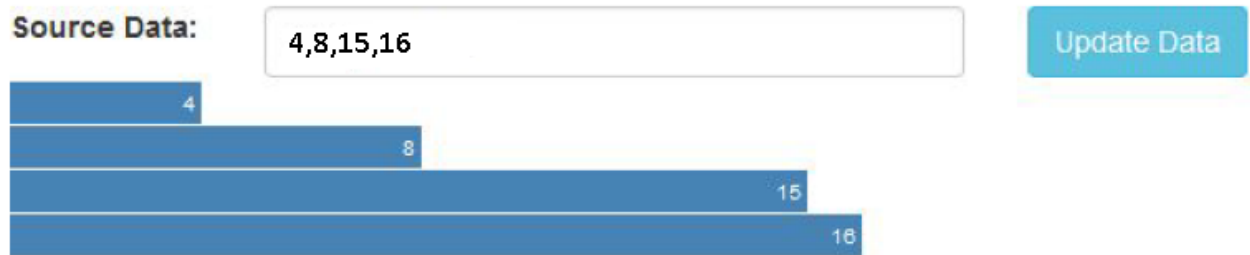
- El formulario debe validar los tipos de datos y los campos requeridos
- Solo se debe llenar un detalle para la orden.

Requerimientos No Funcionales:

- El formulario debe lucir como se muestra en el mockup

5. Graficando con D3

Construya una aplicación Frontend usando vanilla javascript que permita crear un gráfico de barras con [D3.js](#) a partir de unos datos ingresados a una caja de texto. La aplicación tiene la siguiente estructura:



- Source Data: es un input de texto donde el usuario puede ingresar números enteros separados por coma.
- Botón Update Data: ejecutara un evento que separa los datos ingresados en la caja de texto y los utiliza como datos de entrada del grafico de barras en la parte inferior.
- Las barras deben mostrarse usando un color diferente para cada uno. Se puede limitar la gama de colores a 5, pero garantizando que las barras contiguas no tengan el mismo color.
- Chart: Lugar donde se muestra el grafico de barras luego de dar click en el botón “Update Data”

Recomendaciones:

- Se evaluará validación de datos de entrada y seguimiento de buenas prácticas de codificación con javascript