**Web API**

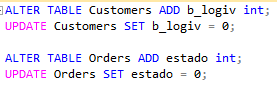
**Laboratorio de Angular**

**Objetivo**

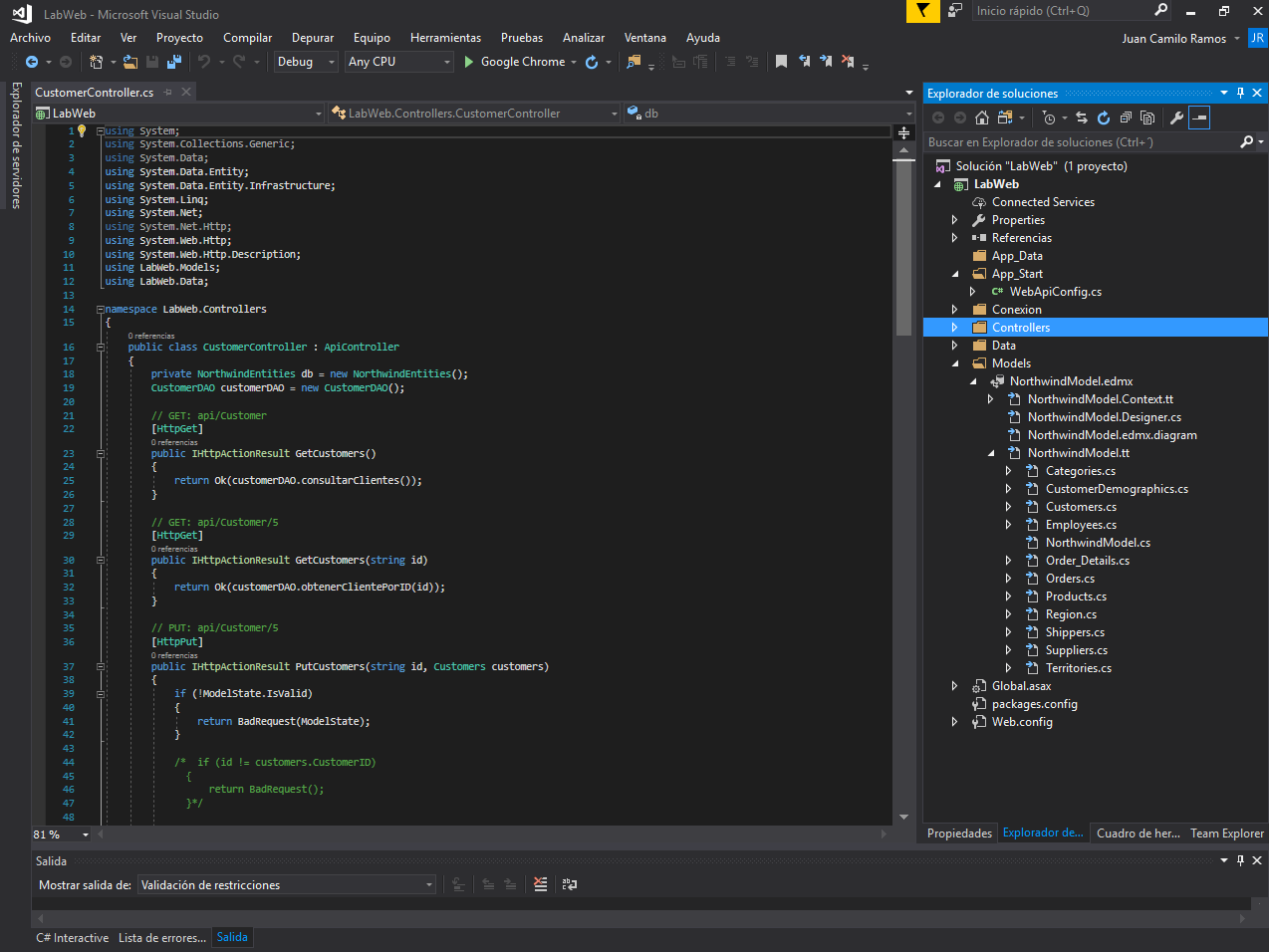
El siguiente laboratorio tiene como objetivo principal el desarrollo de un CRUD básico de pedidos de la base de datos de Northwind utilizando como FRONT – END, el framework de angular consumiendo una Web Api en .NET

**Pasos o recomendaciones**

1. Lo primero que vamos a realizar, es una modificación a la base de datos de Northwind, para agregar un campo más a la tabla “Customers” y a la tabla “Orders”, ya que para realizar la eliminación de un pedido o de un cliente, entra en conflicto con las llaves foráneas que se tienen, y para esto se agregan estos campos, para realizar un eliminado lógico. Las consultas que ejecutaremos serán las siguientes:

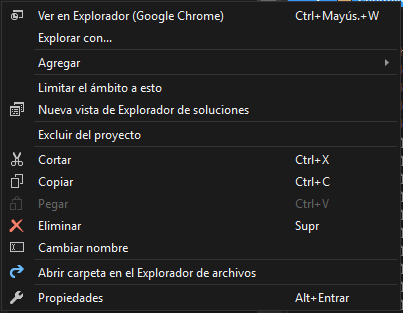


1. Inicialmente se tendrá que abrir el servicio de clientes que en el demo fue creado

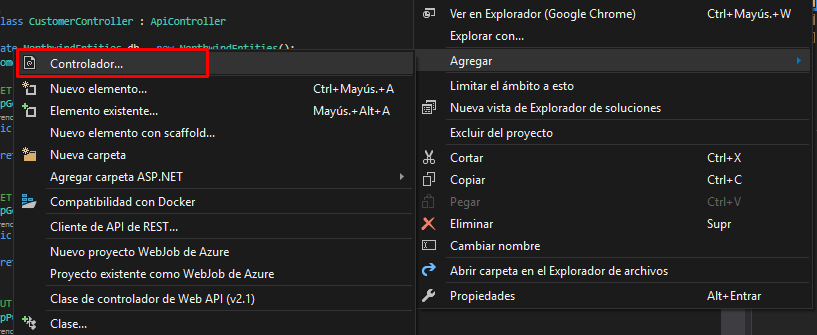


1. En la carpeta “Controllers”, agregaremos el controlador para pedidos, de esta manera:

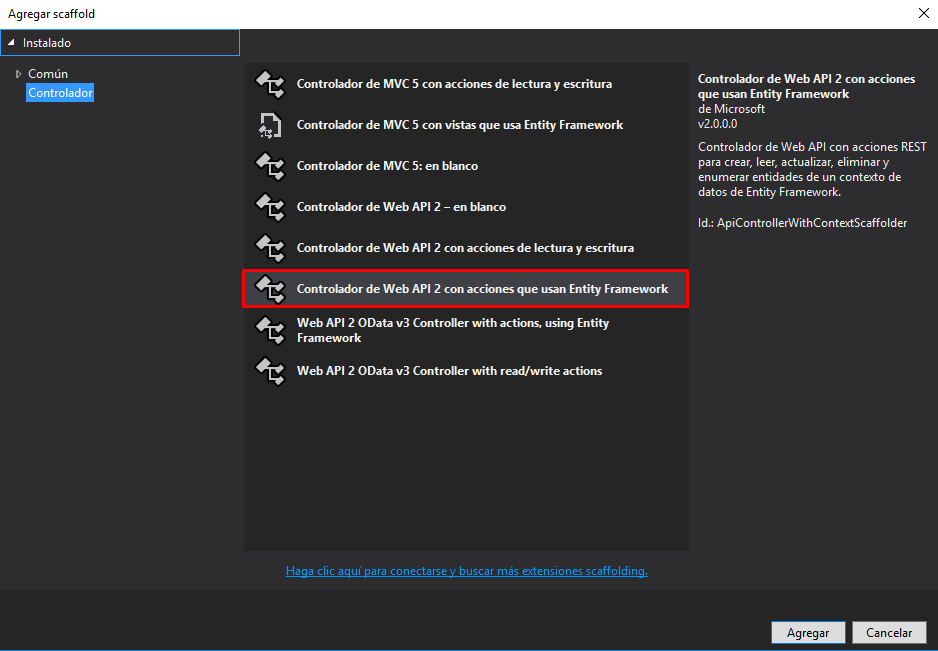
* Primero llevaremos el cursor hasta la carpeta, daremos click derecho y escogemos la opción agregar:



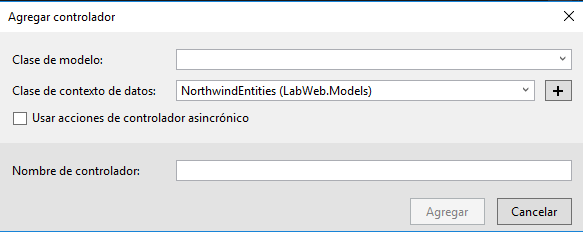
* Despues daremos click en “Controlador”



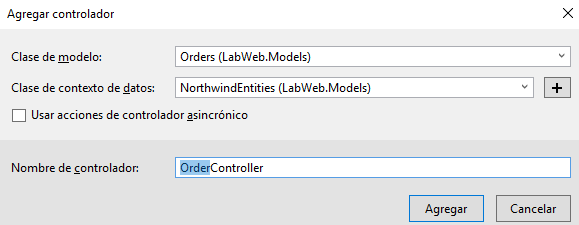
* Una vez hecho esto daremos click en la opción “Controlador con acciones en EntityFramework” y daremos click en “Agregar”



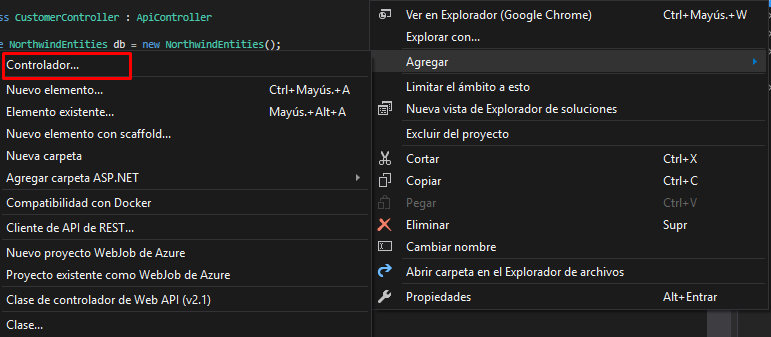
* Despues de dar click, el sistema abrirá la siguiente ventana:

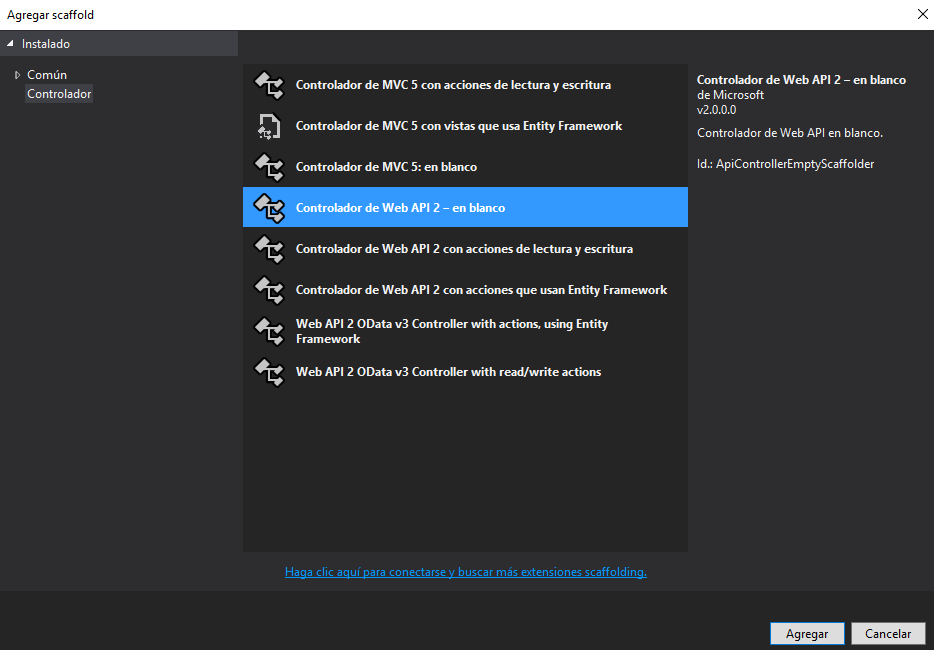


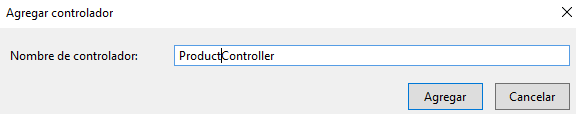
* En el comboBox de “Clase de modelo”, esocogeremos “Orders”, y en el comboBox de “Clase de contexto” escogeremos “NorthwindEntities”, y daremos click en “Agregar”:



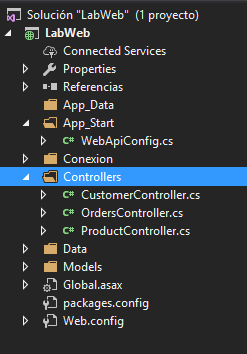
1. Una vez agregado este controlador, repetiremos los pasos anteriores, pero agregando un controlador para “Products”, pero este controlador será de tipo vacio, pues solo se necesitará el método que nos retorna la lista de productos, así:





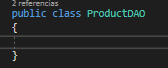


1. Una vez agregado los controladores, nuestro proyecto se vera de la siguiente manera:

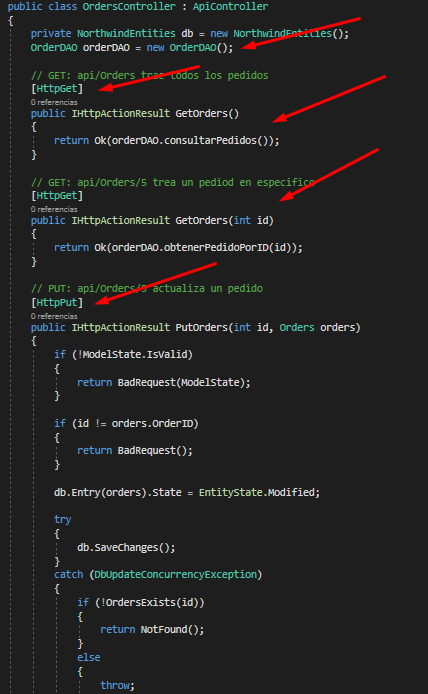


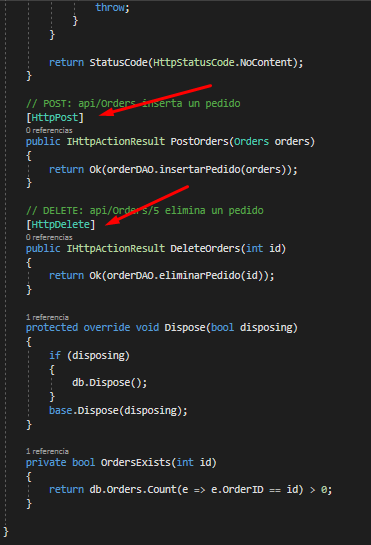
1. Despues de construir nuestros controladores, en la carpeta “Data”, agergaremos las clases “OrderDAO” y “ProductDAO”, y estas clases estarán vacias:



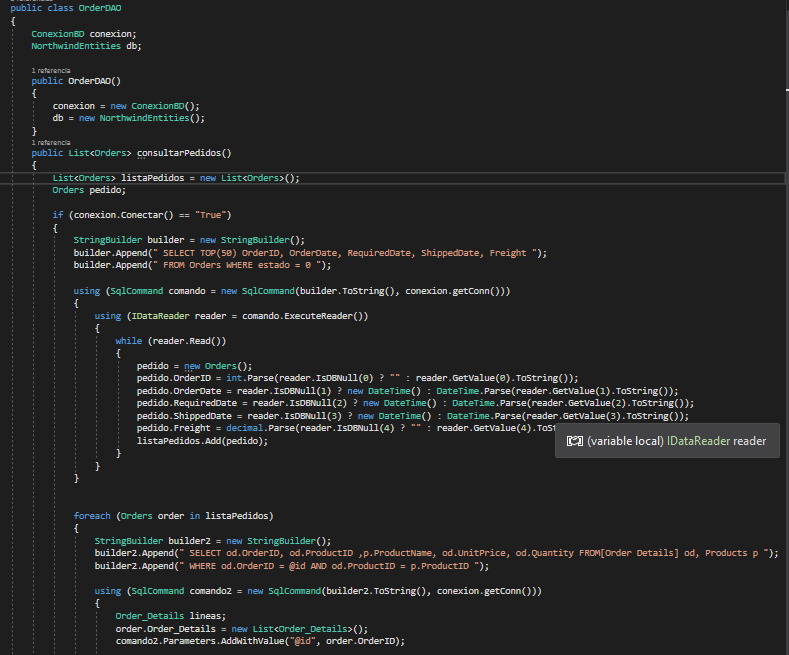


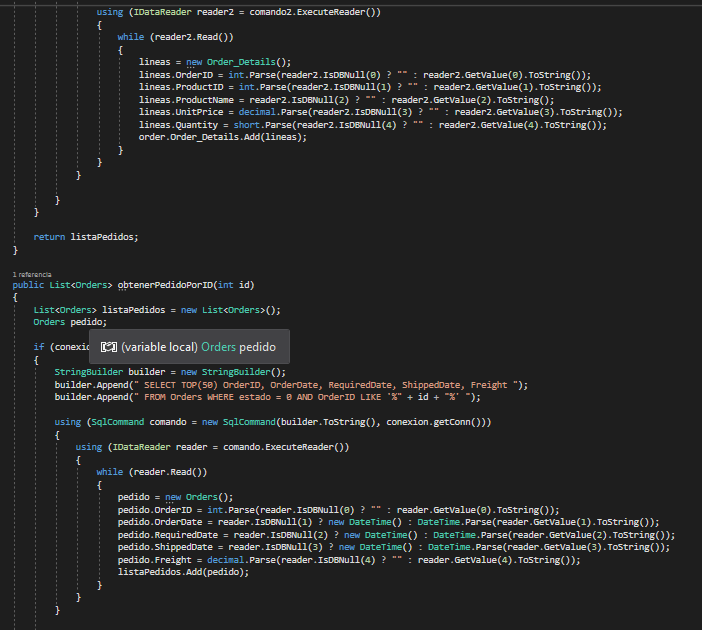
1. En el controlador “OrderController”, cambiaremos el código de los dos primeros métodos, y del método “DELETE”, para realizar un eliminado lógico en la BD, y estos métodos serán utilizados para personalizar la consulta que hacemos a la base de datos, también agregaremos una instancia de “OrderDAO”, para dividir esa consulta y pondremos esta línea de código arriba de cada método, dependiendo de su petición “[Http…]”, así:

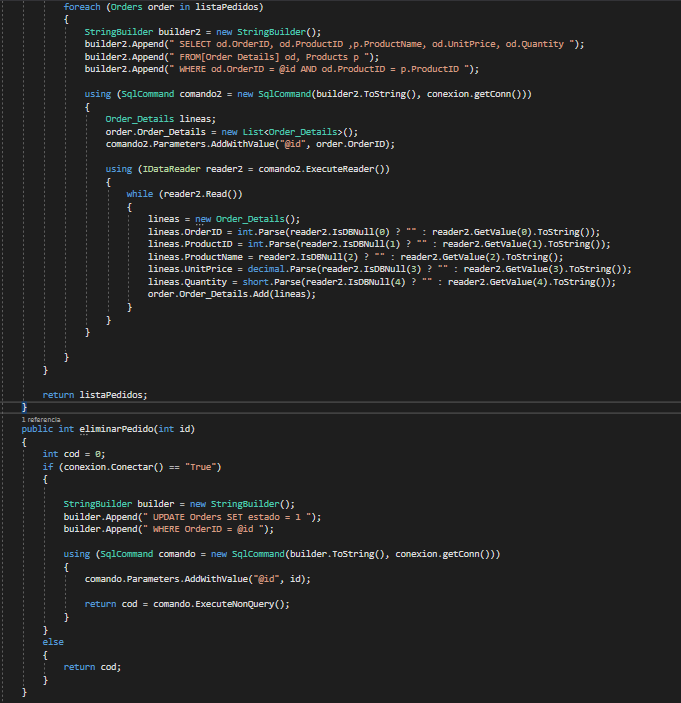


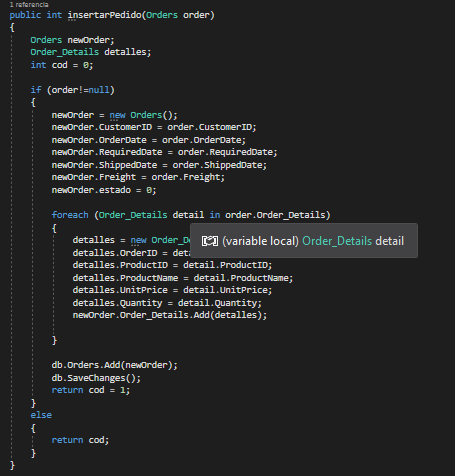


1. Despues de modificar el código del controlador, nos saldrán unos erroes, ya que en la clase “OrderDAO”, no se tienen creados estos métodos, entonces en esta clase agregaremos lo siguiente:

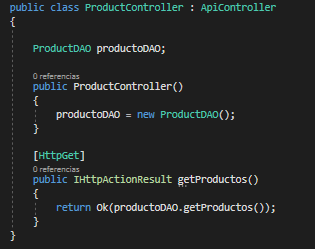




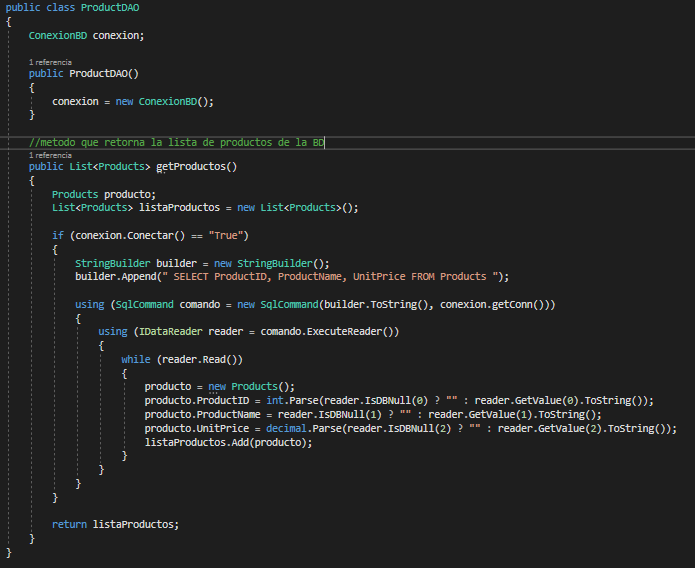




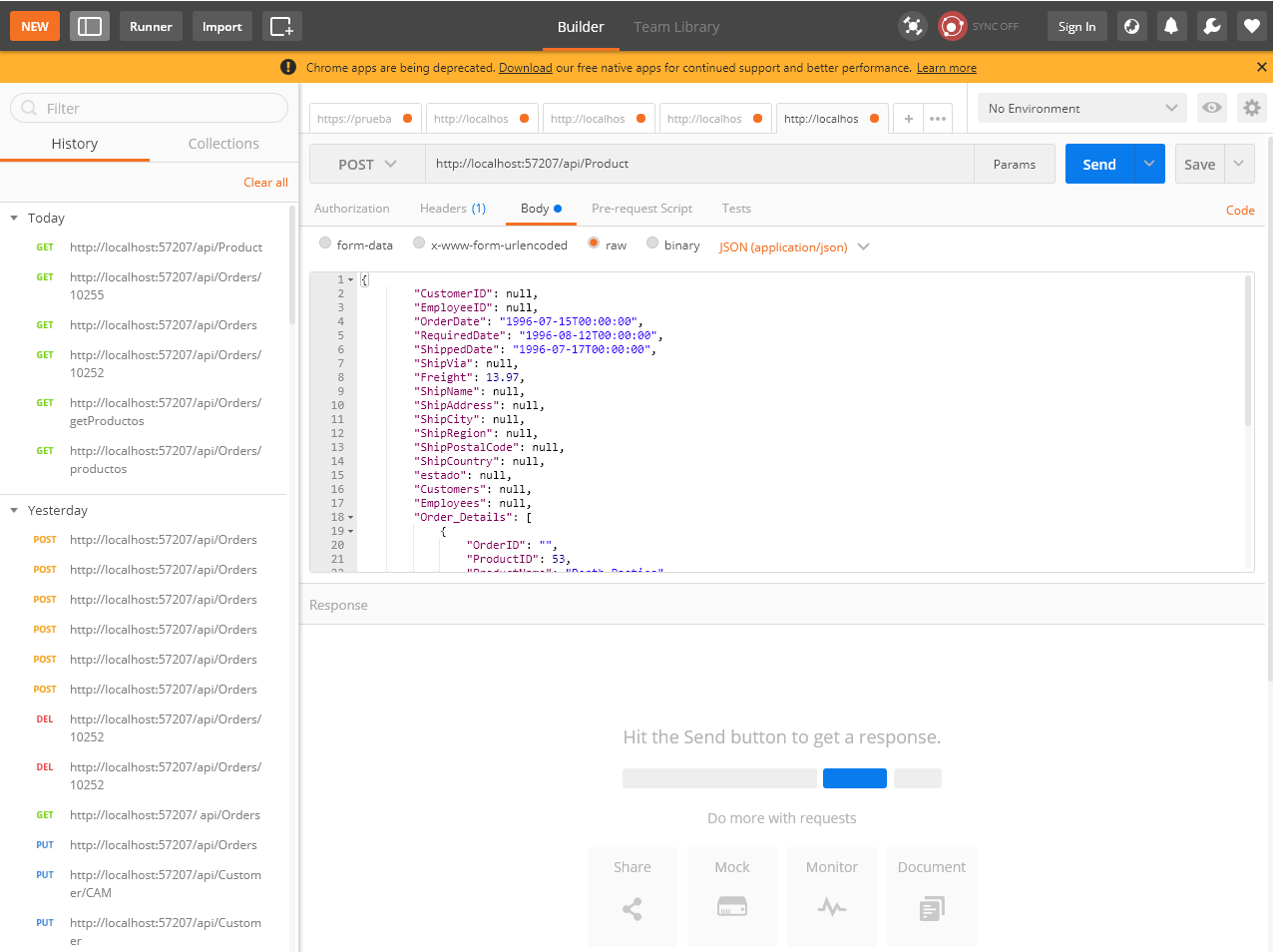
1. Una vez finalizado el paso anterior, nos iremos al controlador de productos, y este controlador estará vacío, allí agregaremos las siguientes líneas de código :



1. Aquí nos saldrá un error, ya que en la clase “ProductDAO”, no se tiene creado el método que especificamos, entonces iremos a la clase y agregamos lo siguiente:



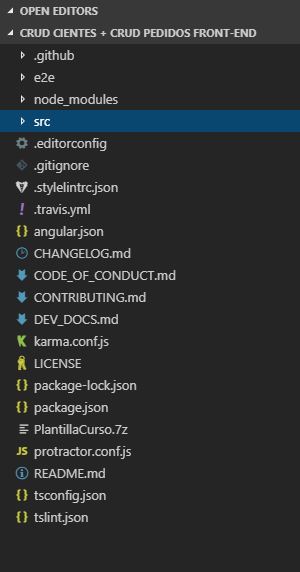
1. Después de realizar todos estos pasos, lo siguiente será probar nuestro servicio, utilizando el Postman o cualquier otra herramienta que nos proporcione la capacidad de probar nuestras peticiones:



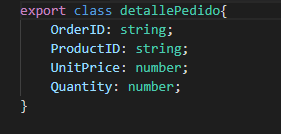
**Angular**

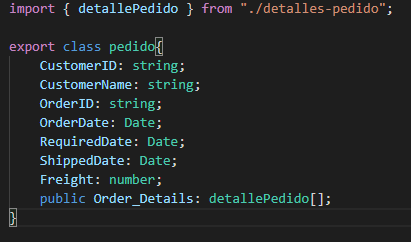
**Pasos o recomendaciones**

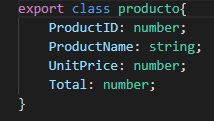
1. Para empezar, lo primero que se va a hacer es abrir la plantilla proporcionada en el curso:



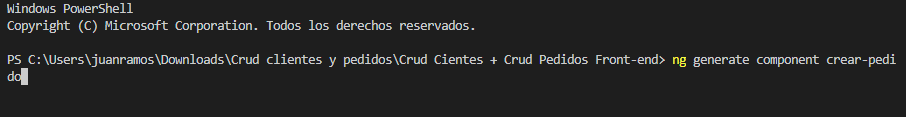
1. En la carpeta de model, vamos a crear las clases de pedido, detallePedido y productos, estas clases se verán de la siguiente manera:



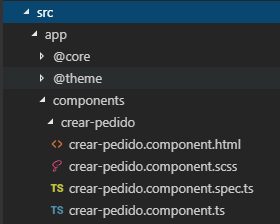




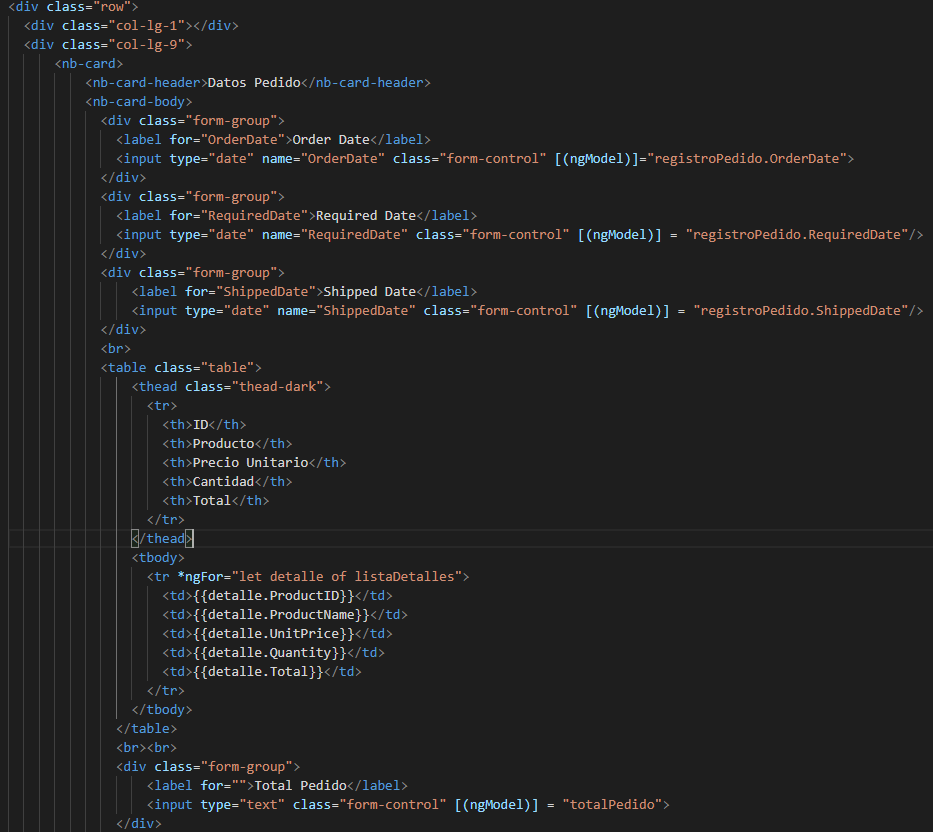
1. En la carpeta componentes, se creará un nuevo componente con el nombre “crear-pedido”, ejecutando el siguiente comando:

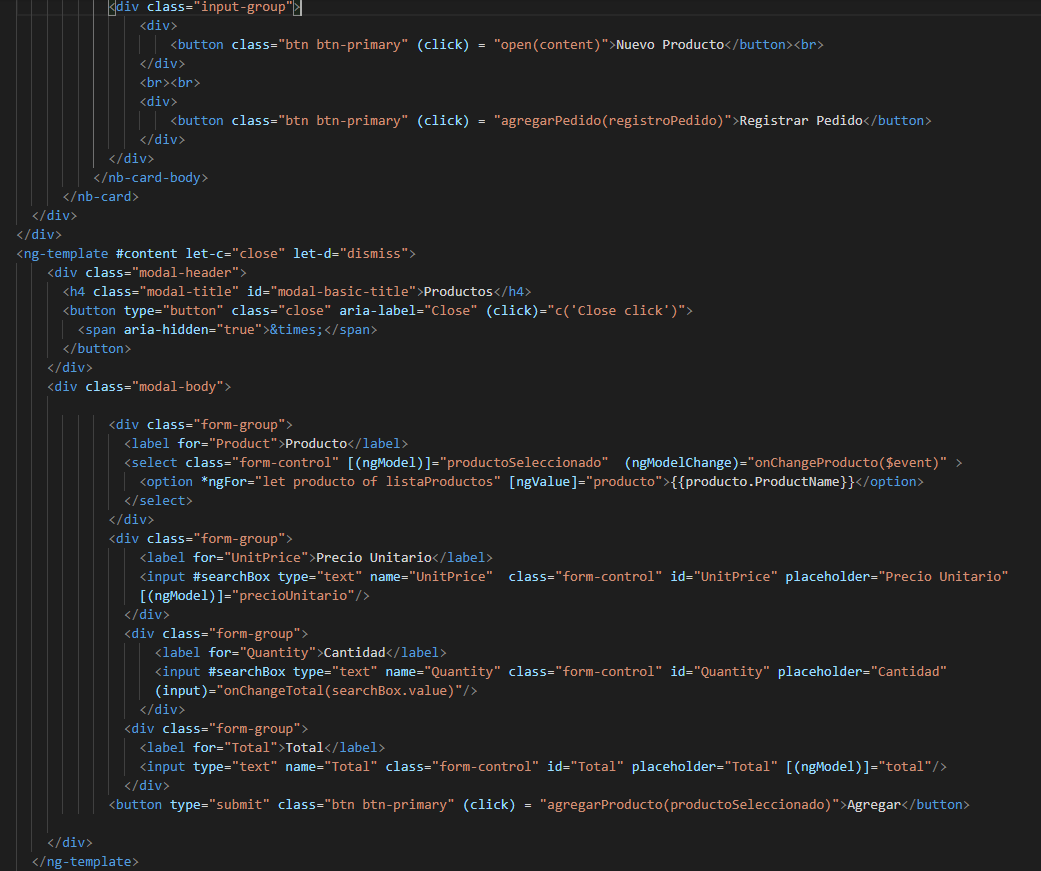


1. Después de crear el componente, el proyecto quedara de la siguiente manera:



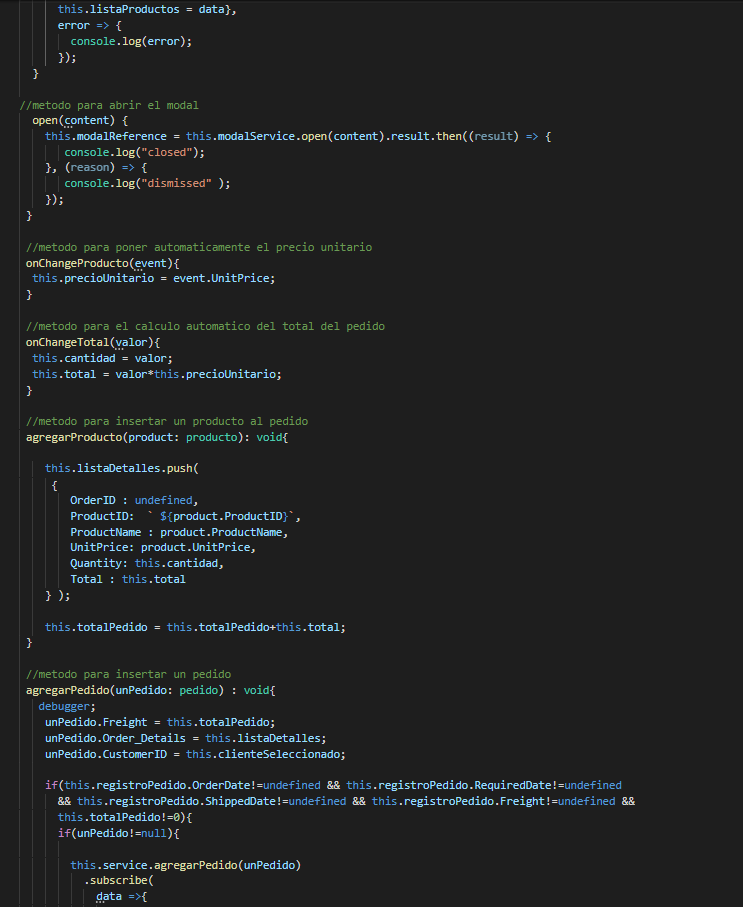
1. En el HTML de este componente vamos a poner el siguiente código, para crear el formulario de registro de pedidos, el código fuente del formulario se puede consultar en los anexos de documento:

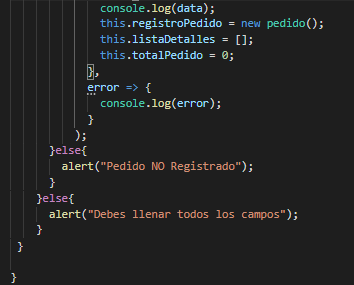




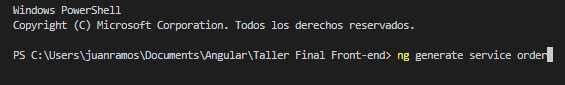
1. En el TypeScript del componente escribiremos las siguientes líneas de código, estableciendo la lógica de negocio y el acceso a los datos:



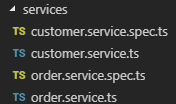




1. Después de agregar este código vamos a crear un servicio para el control de los pedidos, a este servicio le pondremos el siguiente nombre “order”, y ejecutaremos el siguiente comando:



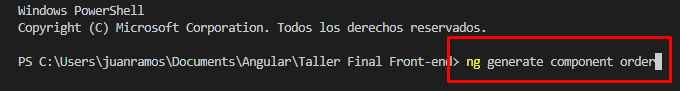
1. Después de crear el servicio, vamos a agregarlo en la carpeta “services”, esto se vera de la siguiente manera:

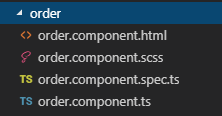


1. En el order.service.ts de nuestro proyecto crearemos los métodos que especificamos en el método anterior, recordar que se expondrán los archivos de este proyecto:

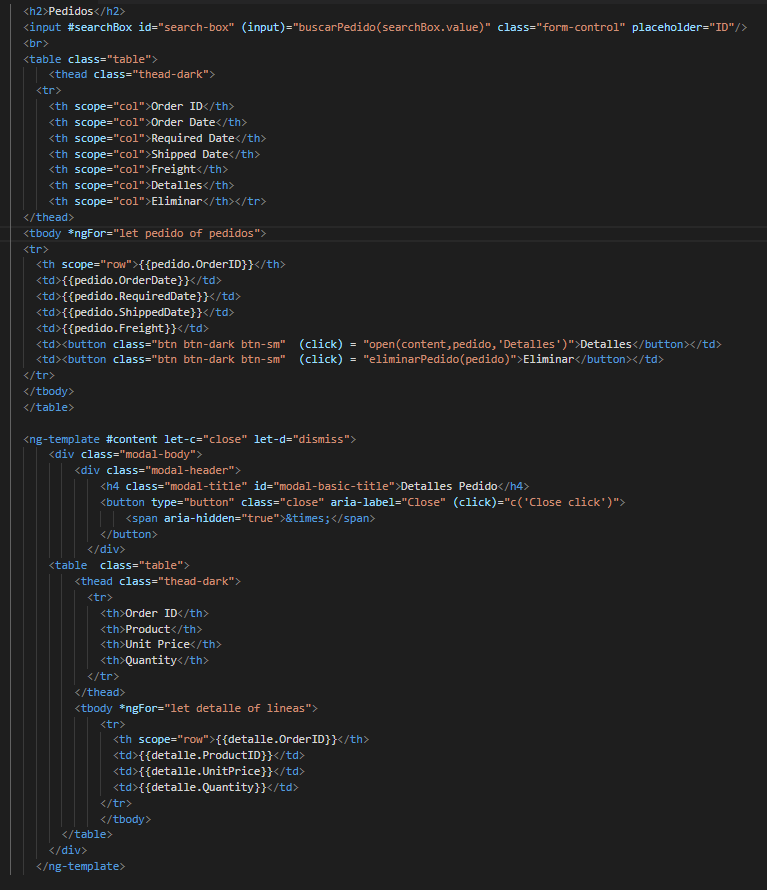


1. Después de realizar los pasos anteriores, vamos a crear un nuevo componente con el nombre “order” o el que ustedes quieran establecer, lo agregamos a la carpeta “components”, ejecutaremos el comando especifico. Este componente nos ayudará a visualizar la lista de pedidos y tendrá dos acciones, una para visualizar los detalles de los pedidos y otra para eliminarlo, este componente se creará en base al comando anteriormente ejecutado y se vera de esta manera:



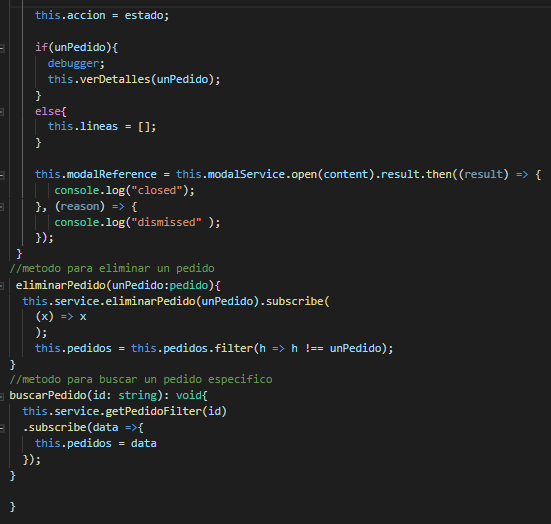


1. En el html de este componente se creará el siguiente código:

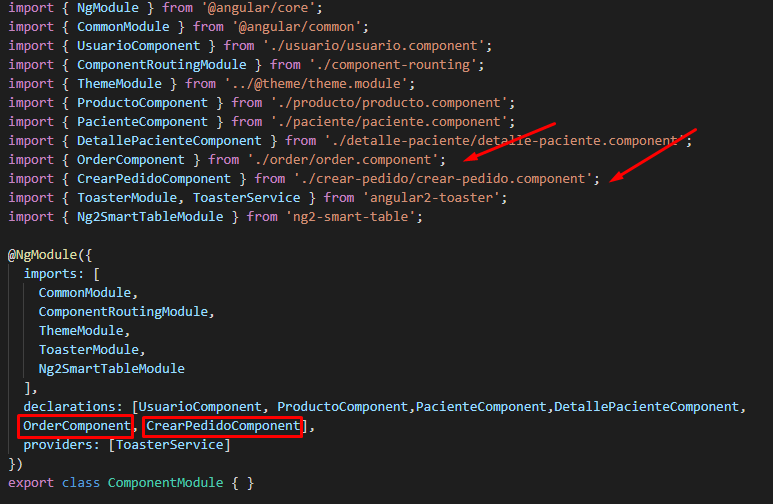


1. En el TypeScript de este componente escribiremos lo siguiente:

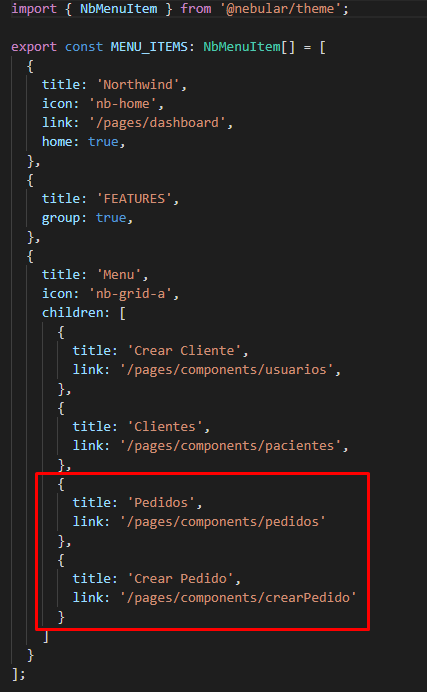




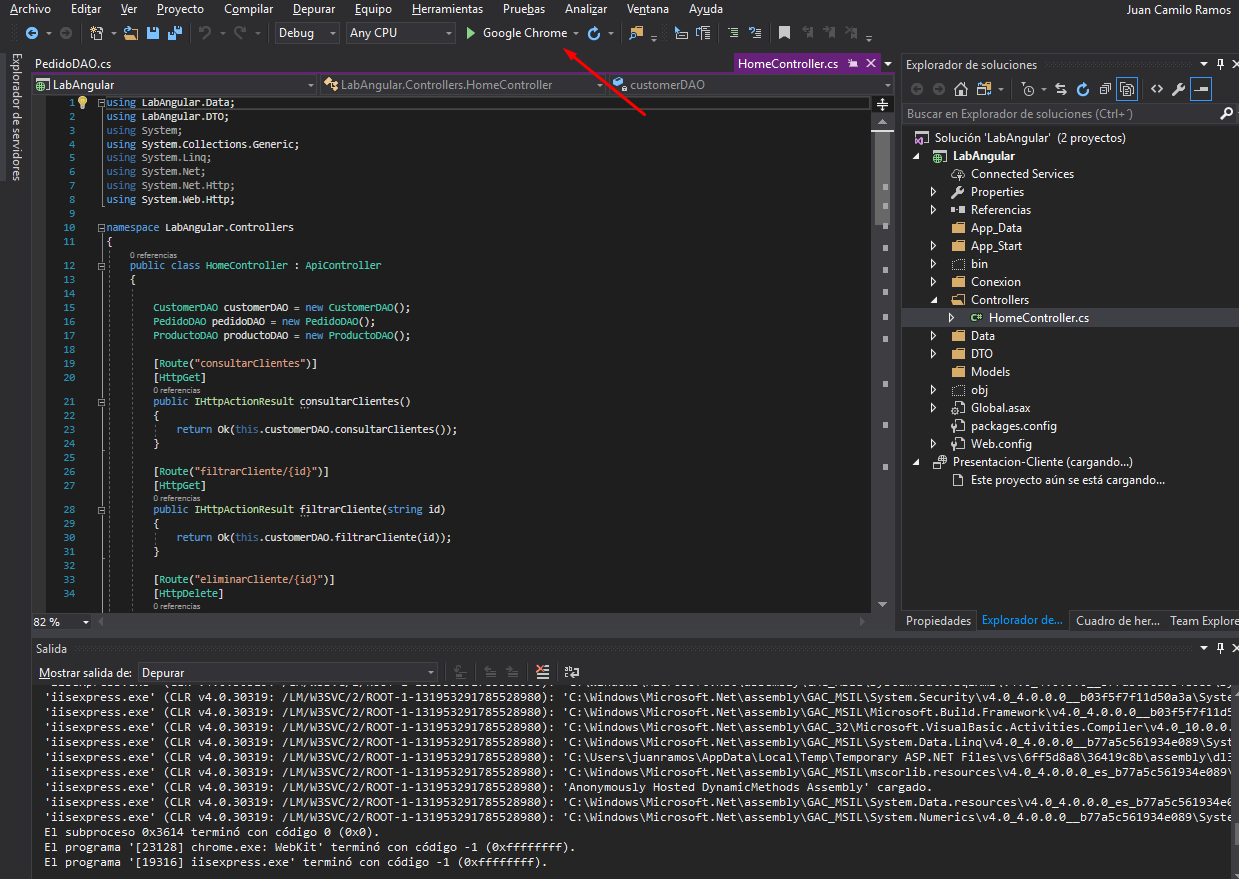
1. Después de crear los componentes y la lógica de cada uno de ellos, se debe establecer en el menú de este proyecto las rutas de acceso a estos componentes
2. En el archivo “component.module.ts”, vamos a importar los componentes “order” y “crear-pedido”, y después los llamaremos en el “declarations” de este módulo:



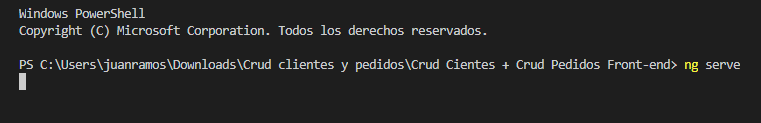
1. Y en la carpeta “pages”, buscaremos el archivo “pages-menu.ts” y a este le agregamos el siguiente menú:



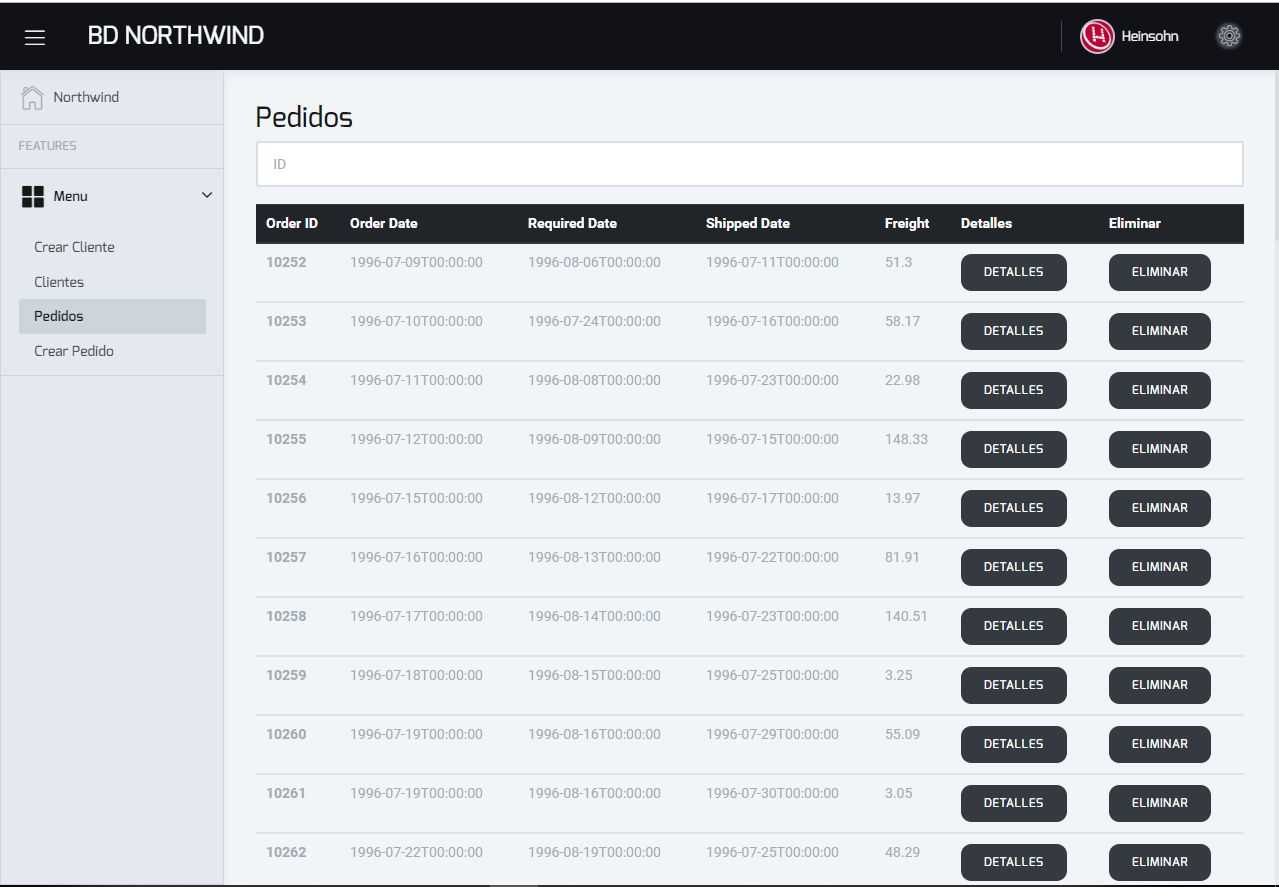
1. Una vez finalizado este paso, ejecutaremos nuestro servicio:

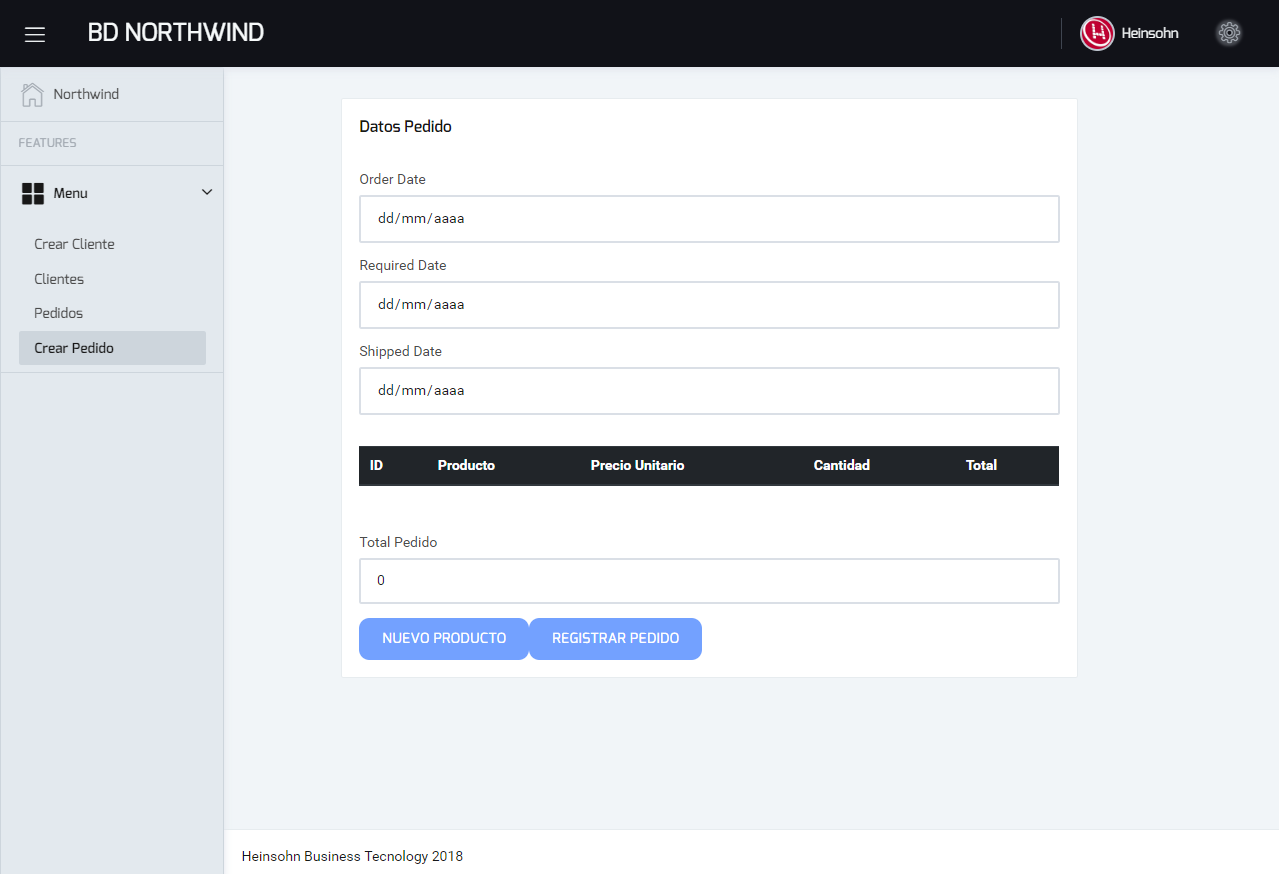


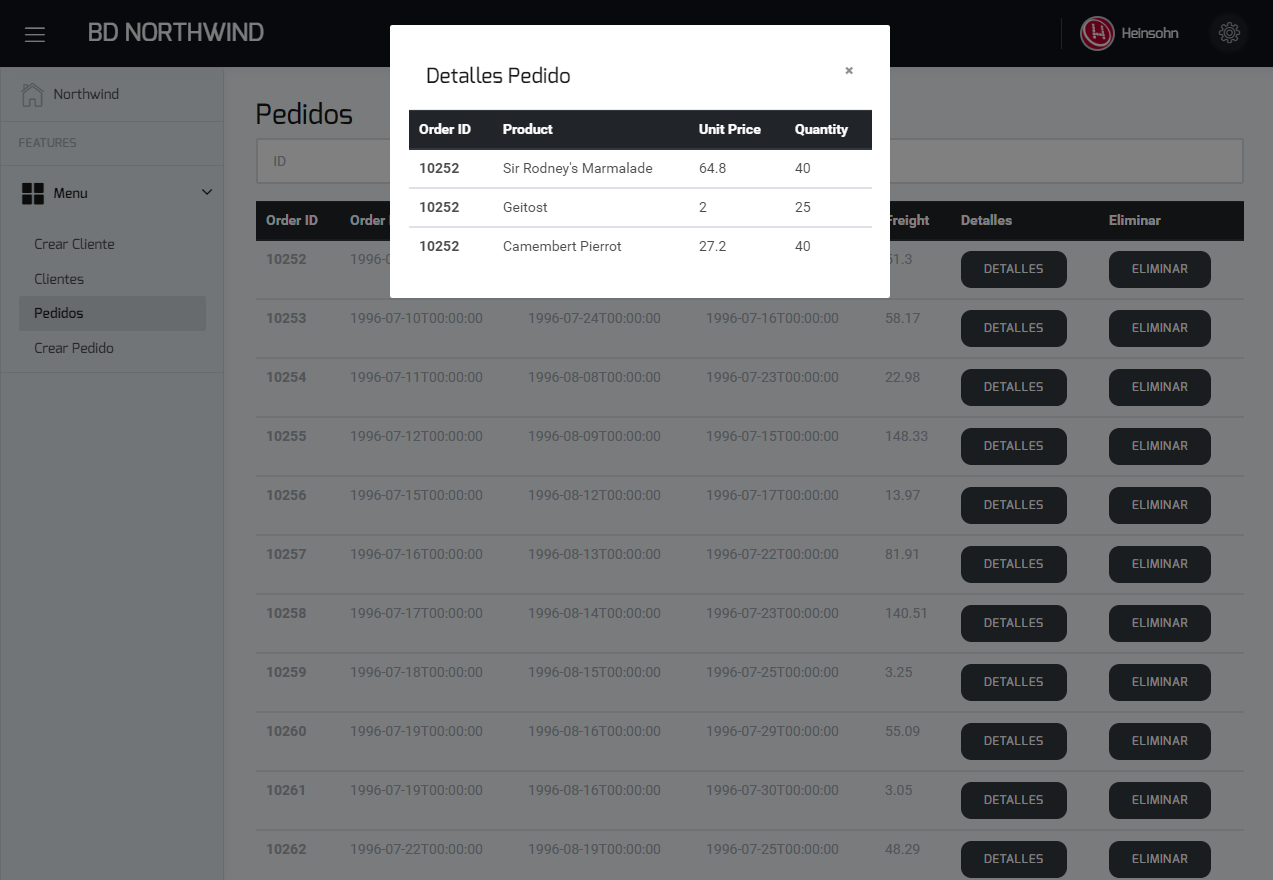
1. Después de ejecutar nuestro servicio, ejecutaremos nuestro proyecto angular con el siguiente comando:



1. Este será el resultado de nuestro proyecto:







1. Fin del ejercicio, si tienen duda con alguno de los documentos que se deben crear, recuerden que se expondrá el proyecto con el código fuente para que se guíen en la creación de este CRUD.