

Manual de uso para el aplicativo BillingApp.

Descripción

En este manual, se explica paso a paso como utilizar el software de facturación solicitado, teniendo en cuenta su instalación y configuración para un correcto funcionamiento.

Recursos necesarios

- Tener un entorno SQL para poder importar la base de datos compartida.
- Descargar la solución “BillingApp” que se compartió mediante el repositorio de Github.

Nota: Si no se cuenta con un IDE que pueda interpretar una solución hecha en .NET Framework 4.8, se recomienda instalar Visual Studio 2019.

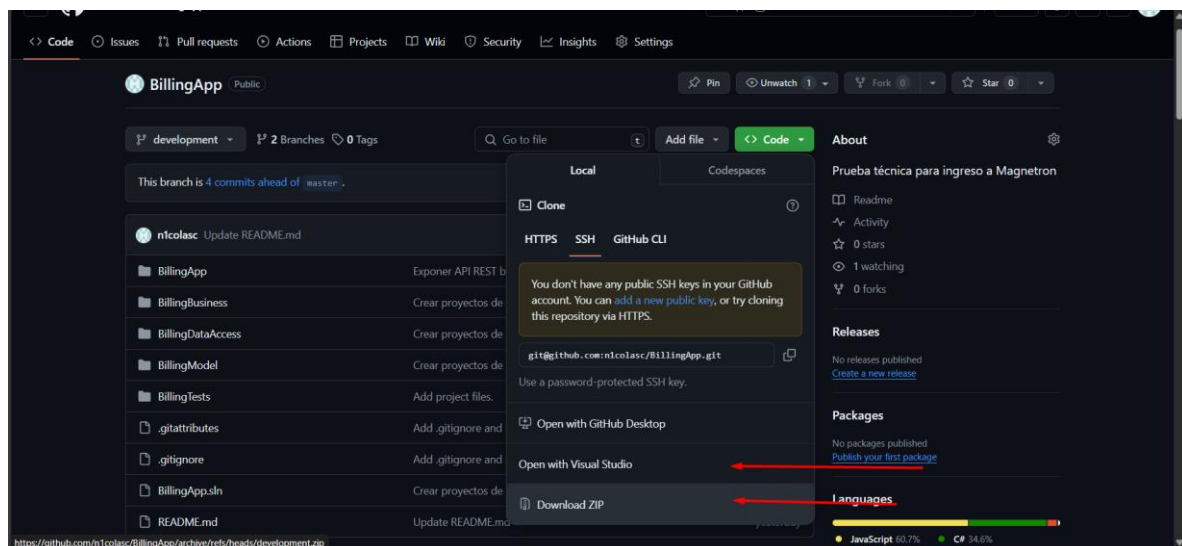
- Instalar Postman para consumir los API-REST

Descargar proyecto de Github

Para descargar el proyecto desde github, se puede acceder a la siguiente url:

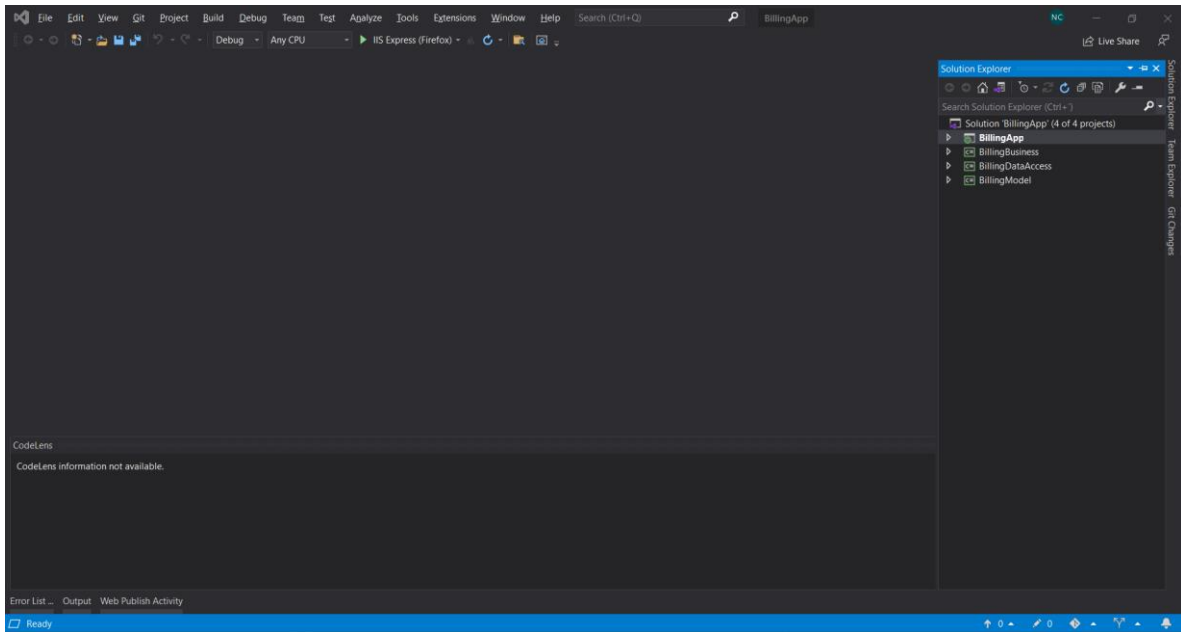
[n1colasc/BillingApp at development \(github.com\)](https://github.com/n1colasc/BillingApp)

Nota: Se dejó el Branch de “master” intacto para poder hacer las comparativas de “inicio” y “final”.

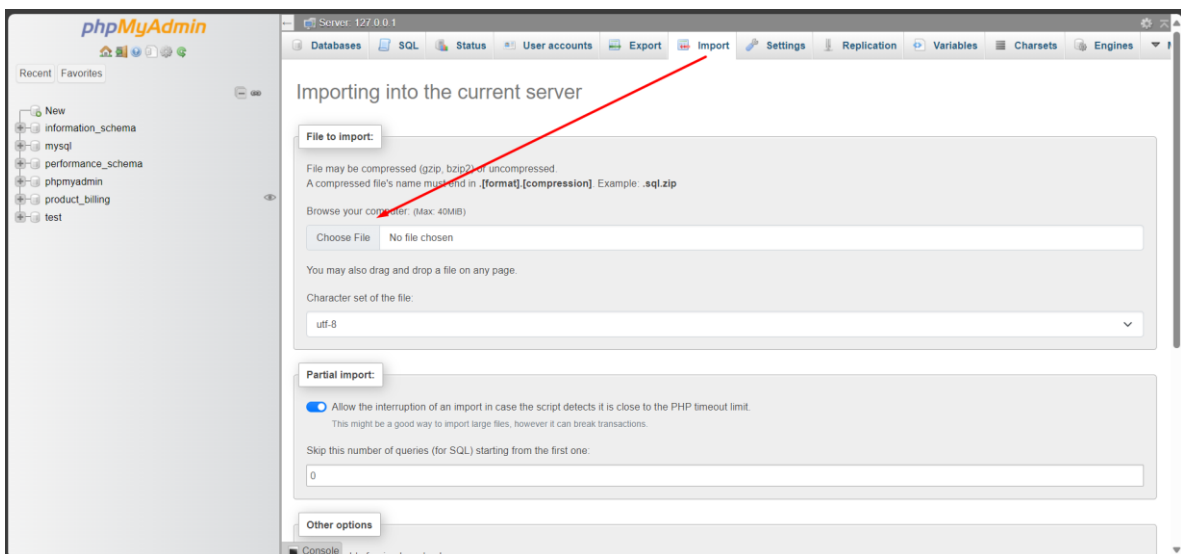


Levantamiento del proyecto

Una vez abierta la solución en el IDE seleccionado, se podrá levantar el Web Service para exponer los servicios solicitados en un entorno de desarrollo (localhost).



Para la base de datos, en el ejemplo se ha usado el software xampp con ayuda de phpmyadmin, lo cual facilita la importación de la base de datos y su respectivo uso.



Consumiendo los servicios

Una vez el proyecto esté arriba, se podrá ver a primera vista el listado de las APIs expuestas, donde se evidencian sus modelos de entrada y salida. Esto nos ayudará a saber qué debemos poner en cada servicio para poder consumirlo

ASP.NET Web API Help Page

Introduction

Provide a general description of your APIs here.

Main

API	Description
POST api/principal/v1/execute	No documentation available.

Productos

API	Description
DELETE api/productos/v1/execute	No documentation available.
GET api/productos/v1/execute	No documentation available.
POST api/productos/v1/execute	No documentation available.
PATCH api/productos/v1/execute	No documentation available.

Persona

API	Description
DELETE api/persona/v1/execute	No documentation available.
GET api/persona/v1/execute	No documentation available.

[Help Page Home](#)

POST api/principal/v1/execute

Request Information

URI Parameters

None.

Body Parameters

[MainRequest](#)

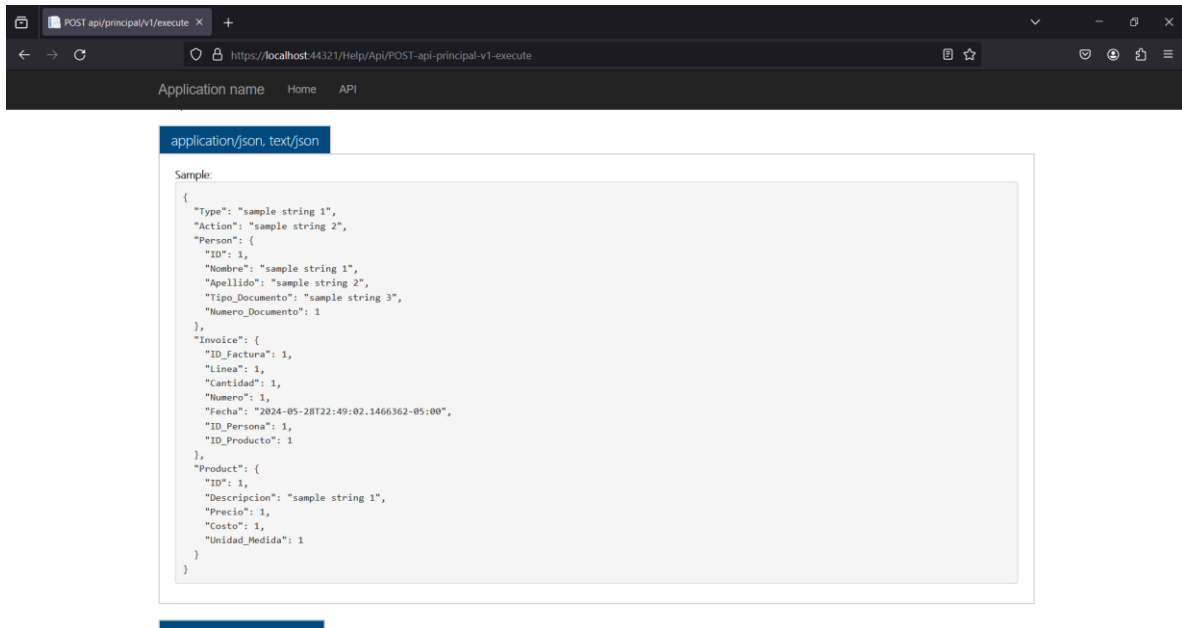
Name	Description	Type	Additional information
Type		string	None.
Action		string	None.
Person		PersonRequest	None.
Invoice		InvoiceRequest	None.
Product		ProductsRequest	None.

Request Formats

[application/json, text/json](#)

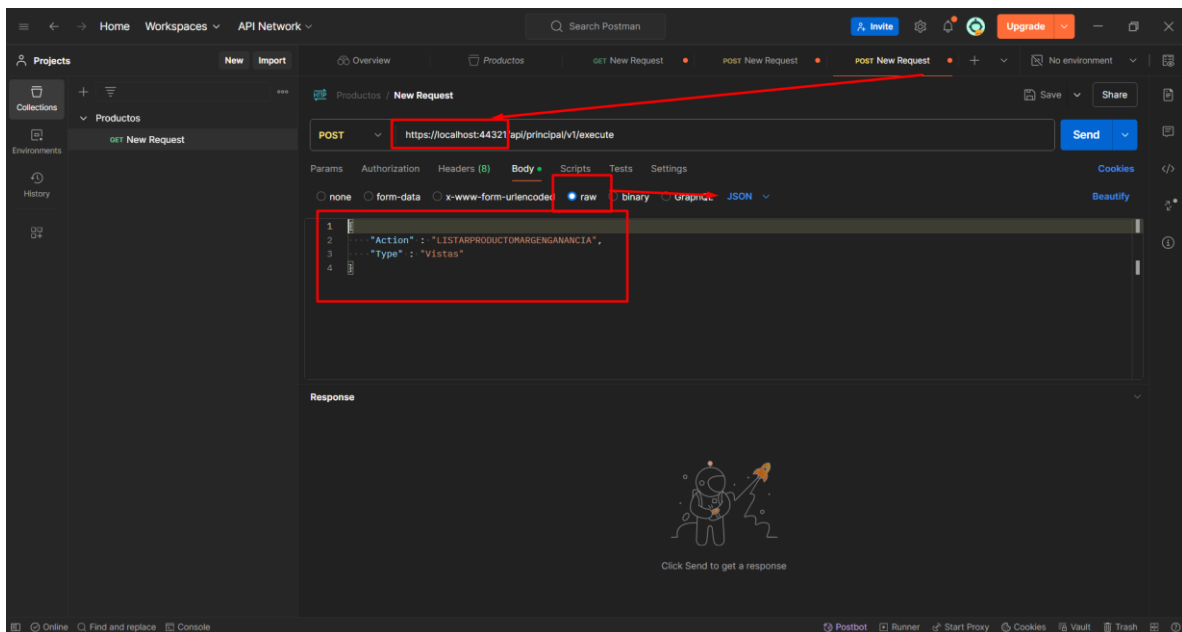
Sample:

```
{
```



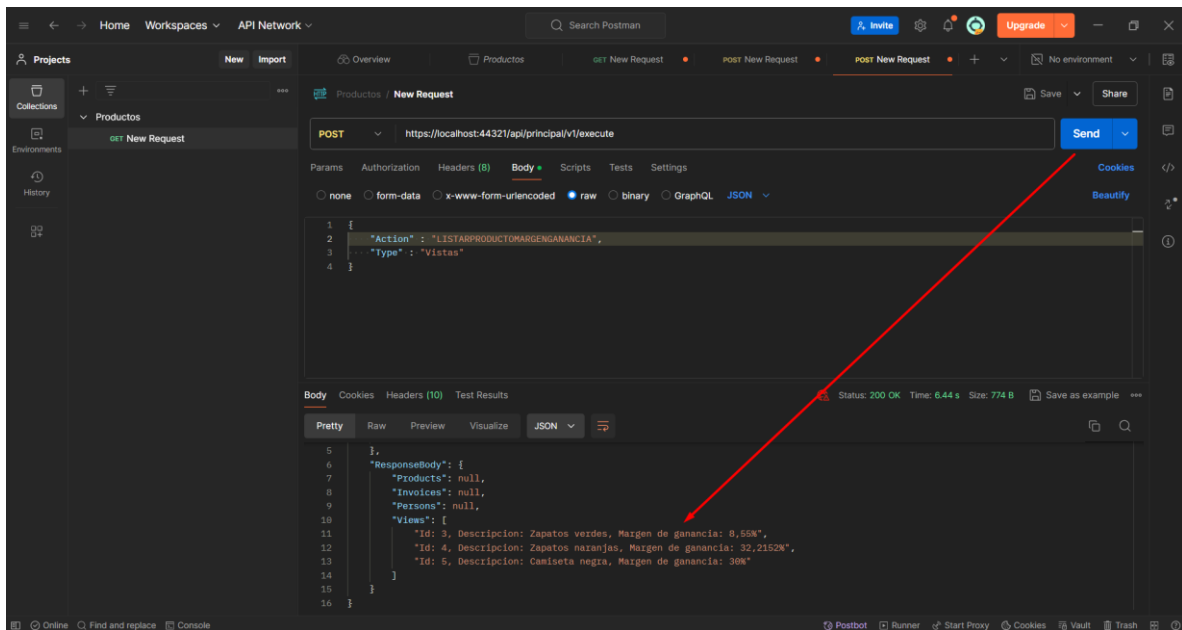
Para poder consumir las API-REST expuestas, haremos uso de Postman, el cual nos ayudará a comunicarnos con el servidor que se levantó anteriormente.

En este caso, haremos clic en el + para abrir una nueva pestaña, se configurará el método HTTP según el servicio, y se digitará el endpoint según lo que se quiera probar. Por último, en la pestaña de “Body” seleccionamos “raw” y formato JSON



Como se muestra en la imagen, haremos uso del servicio **[POST] /api/principal/v1/execute** con un ejemplo de consumo de vista, específicamente el requerimiento:

Liste los productos y el margen de ganancia de cada uno según su facturación.



De esta forma, y guiándonos con la documentación expuesta por el mismo Web Service, podremos consumir todos los servicios del listado.

- CRUD Productos
- CRUD Persona
- CRUD Facturas
- CRUD Principal (Unifica las funcionalidades anteriores y permite consultar las vistas)
 - "ListarPersonasTotalFacturado",
 - "ListarPersonaProductoMasCaro",
 - "ListarProductosCantidadFacturadaDESC",
 - "ListarProductosSegunUtilidad",
 - "ListarProductoMargenGanancia"

Dockerización

Para este proyecto, se utilizó la función integrada de Visual Studio 2019, el cual nos permite publicar la solución en una imagen.

Nota: El archivo "Dockerfile" autogenerado por Visual Studio se encuentra dentro del proyecto "BillingApp".

