**Proyecto Integrador Aclaraciones:**

La base de datos es facturación Las tablas son las siguientes:

Tabla cliente, con los campos

Entero largo id; PK

Texto cuit; UNIQUE

Texto razonSocial;

Tabla pedido\_venta, con los campos

Entero largo id; PK

Entero largo idcliente; (Cliente)

Date fechaPedido;

Entero largo nroComprobante; UNIQUE

Texto formaPago;

Texto observaciones;

Decimal totalPedido;

Tabla pedido\_venta\_detalle, con los campos

Entero largo id; PK

Entero largo idpedidoventa;(PedidoVenta)

Entero largo idproducto; (Producto)

Entero cantidad;

Decimal subtotal;

Tabla producto, con los campos

Entero largo id; PK

Text codigoProducto; UNIQUE

Entero largo denominacion;

Decimal precioVenta;

EL archivo facturación\_base.sql crea la base de datos con Datos en la tabla clientes y producto(NO AGREGAR MAS).

Las tablas pedido\_venta y pedido\_venta\_detalle están vacias.

SI deseo agregar datos utilizar luego de que por primera vez cargue la base de datos facturación\_base.sql ; utilizar el ejemplo\_pedidos\_y\_detalles.sql luego de eso. Se recomienda asi para que no hayan errores con las primary keys, ya que son autogeneradas.

Para reiniciar la base de datos y las claves autoincrementadadas eliminar completamente la base de datos y volverla a cargar. Primero facturación\_base.sql y luego si lo desearamos cargar los ejemplos.

No modificar el archivo facturación\_base.sql / SI SE plantea MODIFICAR consultar solo por el cambio de generación de numero de comprobante hablado mas adelante

Notas de la API

Primero comando: npx tsc

Luego: npm run dev

GET

* La api debe hacer un get de todos los pedidos.-
* La api debe hacer un get de un pedido por el ID.(me sirve para en el front para obtener el id en la grilla si o requiero para algo)-
* La api debe hacer un get de un pedido por el nroComprobante.-
* La api debe hacer un get de un pedido por un rango de fechas. –

POST

* Al hacer un alta(post) de un pedido tengo que verificar que el cliente exista en la base de datos, hago la logica y cargar datos a la tabla pedido\_venta(en este caso no agrego datos a pedido\_venta\_detalle porque esa logica se encarga el Front, yo recibo información ya procesada, el front luego de insertar el pedido recibe el id del pedido insertado y utiliza las apis de detalle para insertar en la tabla venta\_pedido\_detalle

PUT

* Al hacer la modificacion de un pedido(put) primero verifico sie el pedido exista, que el cliente nuevo que le estoy pasando exista y que el nroComprobante no exista en otro pedido. Aca de vuelta el front me va a pasar la informacion ya procesada y correcta, es decir si aun pedido le modifico la cantidad de productos, le agrego mas, le saco o elmino productos, los cálculos del nuevo total son realizados en el front y el put solo recibe la info para modificar la tabla pedido, las modificaciones de la tabla pedido\_venta detalle se realizaran en el front donde utilizaran la api de detalles para hacer las correspondientes modificaciones de esa tabla. Ya sea modificar un detalle o eliminarlo si elimine ese producto del pedido.

DELETE

* Al hacer un delete tengo que primero eliminar el pedido\_venta\_detalle con el correspondiente Id, y luego ahi eliminar el pedido\_venta aquí no es necesario que se haga lógica en el front ya que al pasarle a la api el Id a eliminar se realiza sobre este método.

Se pueden hacer pruebas con insomnia o con postman.

LISTAS DE APIS:

GET

<http://localhost:3000/api/pedidos/>

<http://localhost:3000/api/pedido/:id>

ejemplo de uso de id <http://localhost:3000/api/pedidos/2> me devuelve un json del id 2

<http://localhost:3000/api/pedidos/comprobante/:nroComprobante>

ejemplo de uso de id <http://localhost:3000/api/pedidos/comprobante/1002> me devuelve un json de nroCompronante 1002

<http://localhost:3000/api/pedidosXFecha?fechaInicio=&fechaFin=>

ejemplo de uso de fechas

http://localhost:3000/api/pedidosXFecha?fechaInicio=2023-01-01&fechaFin=2024-11-09

<http://localhost:3000/api/pedidos/generarPDF/:id> me genera un pdf con toda la información del pedido

<http://localhost:3000/api/detalle/:id> me devuelve un solo detalle del id solicitado

<http://localhost:3000/api/detalle/pedido/:idpedidoventa> me devuelve los detalles donde coincida los id del pedido

<http://localhost:3000/api/clientes/>

<http://localhost:3000/api/clientes/:id>

<http://localhost:3000/api/productos/>

<http://localhost:3000/api/productos/:id>

POST

<http://localhost:3000/api/pedidos/insertar/>

<http://localhost:3000/api/detalle/insertar/>

PUT

<http://localhost:3000/api/pedidos/modificar/:id>

<http://localhost:3000/api/detalle/modificar/:id> este solo modifica la cantidad y el subtotal.

DELETE

<http://localhost:3000/api/pedidos/eliminar/:id>

<http://localhost:3000/api/detalle/eliminar/:id> Esta api solo se utiliza en el caso de que al hacer una modificación de un pedido eliminemos un producto de ese pedido, entonces en la tabla pedido\_venta\_detalle ese detalle no existe mas

**FRONTEND**

Ejemplo de front:



(use otro proyecto y lo modifique para simular la vista no se guien por los campos en las columnas) no tiene que ser 100% igual solo que debe poseer todas las funcionalidades.

El front Debe tener las siguientes cosas.

* Presente en una Grilla todos los pedidos de venta existentes donde pueda buscar todos los pedidos
* Permita la busqueda de un pedido de venta mediante su número de comprobante mediante un input(number).
* Dos campos de Date Fecha inicio y Fecha Fin para buscar un mediante un rango de fechas. La Api debe recibir esa información de la siguiente manera para recibir la información: <http://localhost:3000/api/pedidosXFecha?fechaInicio=2023-01-01&fechaFin=2024-11-09>

La grilla debe ser de la siguiente forma(ejemplo)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Numero Comprobante | Cliente | Fecha | Forma de Pago | Observaciones | Total | Acciones |
| 1007 | Carlos Fernandez | 2024-11-05 | Transferencia | Agregar seguro de envío | $ 52000 | BtnModificar  BtnEliminar  BtnPDF |

Notas Grilla:

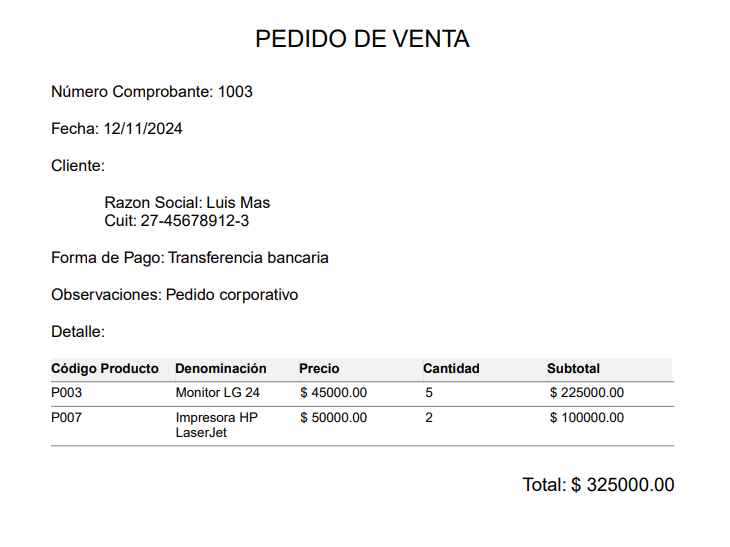
El cliente lo obtenes por su ID lo buscamos en la base de datos(se puede usar la api de clientes) y obtenemos la razón social

En acciones el btn Modificar le pasamos el Id de ese pedido para poder abrir un formulario de modificación.

El btn eliminar nos hace un confirm de si deseamos eliminar ese pedido (también le pasamos el Id).

El btn PDF debe generar un PDF al pasarle el id del pedido a la api <http://localhost:3000/api/pedidos/generarPDF/:id>

Ejemplo PDF:



Para generar el PDF debemos utilizar la api que genera el pdf automáticamente y lo descarga.

**FORMULARIO PARA NUEVO PEDIDO**

* El formulario de Ingresar Un nuevo Pedido tiene que:
* Solicitar numero de Comprobante( se puede obviar si se modifica en la base de datos que sea un valor generado asi no hay errores de carga de nros Duplicados ya que es de tipo UNIQUE tema a discutir).
* La fecha hay que pedirla con un input DATE.
* El cliente lo seleccionamos de una lista de clientes(se puede utilizar de una api)
* Forma de Pago Seleccionamos ahí(Se pueden Cargar opciones en el select si asi lo desearamos).ejmplo Transferencia Efectivo TarjCredito TarjDebito etc etc
* Observaciones es un simple text Area para escribir.
* Ahora vamos con la carga de productos: tenemos que mostrarle la lista de productos de la base de datos(se puede usar una api) y debemos podemos seleccionar los productos que queramos y la cantidad de ellos.

Debemos también poder eliminar un producto si lo seleccionamos y dsp no lo queremos(NO HABLO DE ELMINAR PRODUCTOS DE LA BASE DE DATOS) sino del formulario

* Cuando ya seleccionamos los productos y la cantidad. En base a esto debemos calcular el total del pedido y mostrarlo en el formulario.
* Cuando ya tenemos todo listo y hacemos el submit del pedido. Debemos primero usando la api:

<http://localhost:3000/api/pedidos/insertar/>

Insertar un nuevo pedido (aca ya asumimos que la lógica fue hecha correctamente y el idcliente es el correcto seleccionado del de formulario y el totalPedido esta correctamente calculado.

Ademas con el Id del pedido generado al hacer el submit se debe agregar a la tabla pedido\_venta\_detalle los detalles del pedido es decir con el idpedidoventa, mediante la api:

<http://localhost:3000/api/detalle/insertar>/

las veces que sea necesario según la cantidad de productos que agregue. Con el id de los productos solicitados, la cantidad de productos solicitados, el subtotal que es la cantidad \* el precioventa del producto.

FORMULARIO DE MODIFICACION.

* Es similar al formulario de Nuevo Pedido, pero usa la api:

<http://localhost:3000/api/pedidos/modificar/:id>

En donde le pasamos el id por la api para hacer la modificación (PUT) del pedido.

Tambien debe hacer la misma lógica para calcular totales y pueden ocurrir distintos casos:

* Si modificamos la cantidad de un producto. El totalPedido se va a modificar y en tabla pedido\_venta\_detalle esto se tiene que ver reflejado en el cambio de cantidad y del subtotal utilizando la api del PUT de detalle

<http://localhost:3000/api/pedidos/modificar/:id>

* En el caso que en un pedido eliminamos un producto completamente también debe calcular los nuevos totales y esto también se tiene que ver reflejado en la tabla pedido\_venta\_detalle ya que ese detalle debe ser eliminado mediante la api

<http://localhost:3000/api/detalle/eliminar/:id>