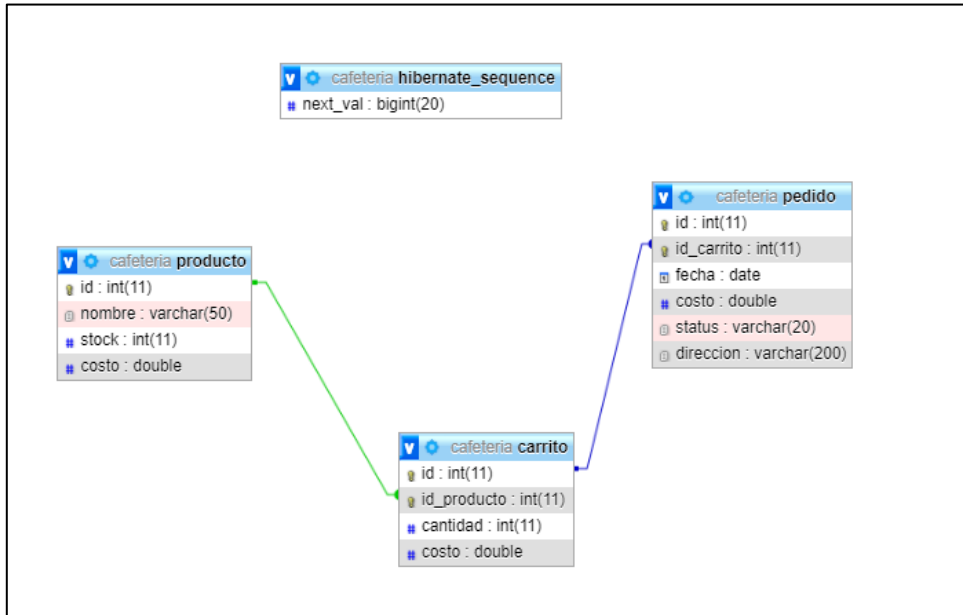


Este servicio web está concebido para trabajar en conjunto con otros servicios web. En si este servicio web será el encargado de la gestión de pedidos de una cafetería, está relacionado directamente con el carrito de compras y tabla de productos.

En un principio, las relaciones de la base de datos para las pruebas se ven de la siguiente manera:



Para poder probar el servicio web se debe de importar la base de datos que se encuentra en el repositorio de GitHub (https://github.com/RafaelUV18/Tec_In-Pedidos/tree/master) en el archivo cafetería.slq. Para hacer esto se debe de seguir estos pasos:

1. En phpMyAdmin crear una base datos llamada cafetería.

Bases de datos

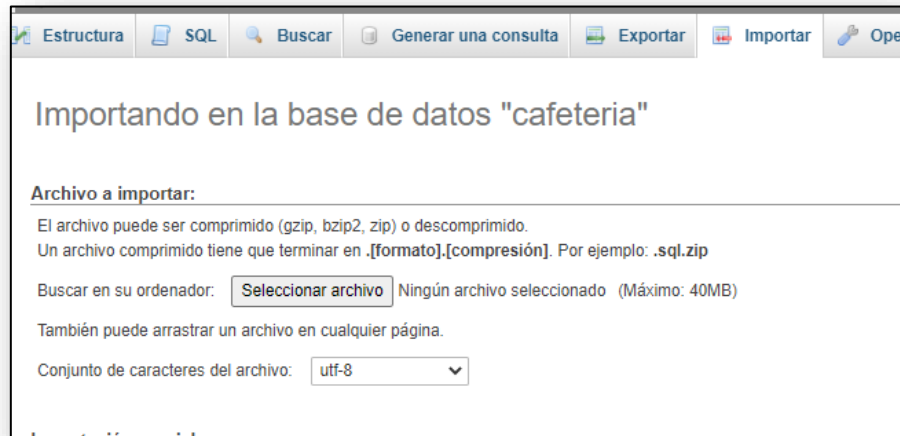
Crear base de datos

cafeteria utf8mb4_general_ci Crear

Base de datos	Cotejamiento	Acción
<input type="checkbox"/> bd-taco-uv	utf8mb4_general_ci	Seleccionar privilegios
<input type="checkbox"/> bduv	utf8mb4_general_ci	Seleccionar privilegios
<input type="checkbox"/> cafeteria	utf8mb4_general_ci	Seleccionar privilegios
<input type="checkbox"/> information_schema	utf8_general_ci	Seleccionar privilegios
<input type="checkbox"/> mensajes_2021	utf8mb4_general_ci	Seleccionar privilegios
<input type="checkbox"/> mysql	utf8mb4_general_ci	Seleccionar privilegios
<input type="checkbox"/> performance_schema	utf8_general_ci	Seleccionar privilegios
<input type="checkbox"/> phpmyadmin	utf8_bin	Seleccionar privilegios
<input type="checkbox"/> sistema1	utf8mb4_general_ci	Seleccionar privilegios
<input type="checkbox"/> test	latin1_swedish_ci	Seleccionar privilegios

Total: 10

2. Dentro de esa base de datos importar el archivo cafeteria.sql.

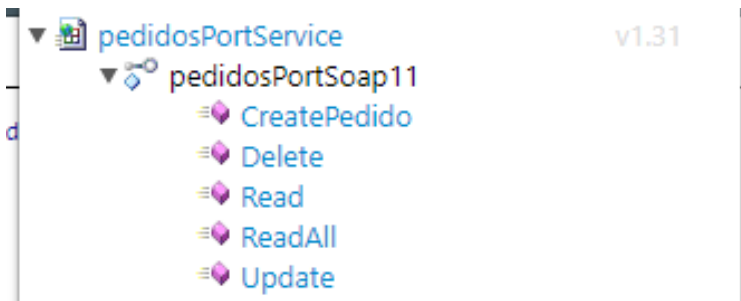


De igual manera se deben de cambiar algunos valores del archivo application.properties dependiendo de la configuración que se tenga en la computadora que vaya a ejecutar el SW. Por defecto se tienen estos valores:

```
application.properties X
src > main > resources > application.properties
1 spring.jpa.hibernate.ddl-auto=update
2 server.port=8081
3 spring.datasource.url=jdbc:mysql://${MYSQL_HOST:localhost}:3306/cafeteria?serverTimezone=UTC
4 spring.datasource.username=root
5 spring.datasource.password=
```

Una vez que el SW este corriendo el URL para acceder es el siguiente:
<http://localhost:8081/ws/cafeteria/pedidos.wsdl>

El servicio web de Pedidos posee 5 operaciones:



1. CreatePedido:

```
POST http://localhost:8081/ws/pedidos
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Body>
    <CreatePedidoRequest xmlns="http://cafeteria/pedidos">
      <id_carrito>[int]</id_carrito>
      <fecha>[string]</fecha>
      <costo>[decimal]</costo>
      <status>[string]</status>
      <direccion>[string]</direccion>
    </CreatePedidoRequest>
  </Body>
</Envelope>
```

Esta operación crea un pedido, recibe 5 argumentos:

- El primero de ellos (id_carrito) necesita que se inserte el id del carrito de compras al que se le relacionó el pedido.
- En la fecha se insertará la fecha en que se realizó el pedido, el formato correcto para escribir la fecha es el siguiente: YYYY-MM-DD (ejemplo 2021-04-20).
- En el costo, se ingresará el total a pagar de los productos, por ello se debería de obtener el total previamente con el carrito de compras.
- En el estatus de pedido, se deberá de ingresar el estado inicial del pedido que se tenga considerado.
- En la dirección, se deberá de ingresar una cadena de texto con la dirección a la que se enviará el pedido.

Respuesta:

```
POST http://localhost:8081/ws/pedidos
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:CreatePedidoResponse xmlns:ns2="http://cafeteria/pedidos">
      <ns2:mensaje>Pedido Agregado con éxito</ns2:mensaje>
      <ns2:idPedidoCreado>10</ns2:idPedidoCreado>
    </ns2:CreatePedidoResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Esta operación regresa un mensaje y un número con el id del pedido creado.

2. Read

Esta operación recibe un argumento con el id del pedido que se quiere recuperar.

```
POST http://localhost:8081/ws/pedidos

<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Body>
    <ReadRequest xmlns="http://cafeteria/pedidos">
      <id>[int]</id>
    </ReadRequest>
  </Body>
</Envelope>
```

Respuesta:

```
POST http://localhost:8081/ws/pedidos

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:ReadResponse xmlns:ns2="http://cafeteria/pedidos">
      <ns2:id>10</ns2:id>
      <ns2:id_carrito>3</ns2:id_carrito>
      <ns2:fecha>2021-04-25</ns2:fecha>
      <ns2:costo>250.0</ns2:costo>
      <ns2:status>Preparando</ns2:status>
      <ns2:direccion>Enriquez</ns2:direccion>
    </ns2:ReadResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Esta operación devuelve la información del pedido que se buscó.

3. ReadAll

Esta operación devuelve la información de todos los pedidos que se tiene registrados. No pide ningún argumento.

```
POST http://localhost:8081/ws/pedidos

<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Body>
    <ReadAllRequest xmlns="http://cafeteria/pedidos"/>
  </Body>
</Envelope>
```

```

POST http://localhost:8081/ws/pedidos
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:ReadAllResponse xmlns:ns2="http://cafeteria/pedidos">
      <ns2:pedido>
        <ns2:id>2</ns2:id>
        <ns2:id_carrito>2</ns2:id_carrito>
        <ns2:fecha>2021-04-17</ns2:fecha>
        <ns2:costo>49.98</ns2:costo>
        <ns2:status>Nice</ns2:status>
        <ns2:direccion>Diamante</ns2:direccion>
      </ns2:pedido>
      <ns2:pedido>
        <ns2:id>7</ns2:id>
        <ns2:id_carrito>1</ns2:id_carrito>
        <ns2:fecha>2021-04-20</ns2:fecha>
        <ns2:costo>160.98</ns2:costo>
        <ns2:status>Inicío</ns2:status>
        <ns2:direccion>Venustiano</ns2:direccion>
      </ns2:pedido>
      <ns2:pedido>
        <ns2:id>9</ns2:id>
        <ns2:id_carrito>3</ns2:id_carrito>
        <ns2:fecha>2021-04-19</ns2:fecha>
        <ns2:costo>7996.0</ns2:costo>
        <ns2:status>Entregado</ns2:status>
        <ns2:direccion>Diamante</ns2:direccion>
      </ns2:pedido>
      <ns2:pedido>
        <ns2:id>10</ns2:id>
        <ns2:id_carrito>3</ns2:id_carrito>
        <ns2:fecha>2021-04-25</ns2:fecha>
        <ns2:costo>250.0</ns2:costo>
        <ns2:status>Preparando</ns2:status>
        <ns2:direccion>Enriquez</ns2:direccion>
      </ns2:pedido>
    </ns2:ReadAllResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4. Update

Esta operación actualiza el estatus de un pedido, requiere de 2 argumentos, el id del pedido y el nuevo estatus que se quiere actualizar.

```

POST http://localhost:8081/ws/pedidos
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Body>
    <UpdateRequest xmlns="http://cafeteria/pedidos">
      <id>[int]</id>
      <status>[string]</status>
    </UpdateRequest>
  </Body>
</Envelope>

```

Respuesta:

```

POST http://localhost:8081/ws/pedidos
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:UpdateResponse xmlns:ns2="http://cafeteria/pedidos">
      <ns2:mensaje>Staus del pedido actualizado</ns2:mensaje>
      <ns2:idPedidoActualizado>10</ns2:idPedidoActualizado>
    </ns2:UpdateResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Esta operación devuelve un mensaje y el id del pedido que se actualizó.

5. Delete

Esta operación elimina un pedido, requiere del id del pedido.

```
POST http://localhost:8081/ws/pedidos
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Body>
    <DeleteRequest xmlns="http://cafeteria/pedidos">
      <id>[int]</id>
    </DeleteRequest>
  </Body>
</Envelope>
```

Respuesta:

```
POST http://localhost:8081/ws/pedidos
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:DeleteResponse xmlns:ns2="http://cafeteria/pedidos">
      <ns2:mensaje>Pedido Eliminado con éxito</ns2:mensaje>
      <ns2:idPedidoEliminado>10</ns2:idPedidoEliminado>
    </ns2:DeleteResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Esta operación devuelve un mensaje y el id del pedido que se eliminó.