

Estructura del Backend

El backend del proyecto se desarrollará en Apache Netbeans.

La estructura principal del proyecto estará compuesta por 4 carpetas principales:

- **Config** → Contendrá el archivo que realiza la conexión con mysql.
 - **Controller** → Es el
 - **Model** → Crearemos las clases para cada una de las tablas existentes en nuestra base de datos así como sus setters y getters, además de sus funciones métodos requeridos.
 - **Rest** → Se encarga de la creación del archivo que conecta el backend con el front desarrollado en ionic en vscode.
-

En nuestra base de datos contamos con 7 tablas que se traducen a 7 clases. Para cada clase se crearan 3 archivos (controller, model y rest)

1. Rol
 2. User
 3. Platillo
 4. Pedidos
 5. Clientes
 6. Inventario
 7. Proveedores
-

Que esperamos poder realizar con el CRUD (Lo resaltado en morado no conlleva foreign keys):

▼ Rol

- ✓ Obtener todos los roles (**GET:** <http://localhost:8080/smartMenu/api/roles>)
- ✓ Obtener rol por id (**GET:** <http://localhost:8080/smartMenu/api/roles/1>)

▼ User

- ✓ Obtener todos los usuarios (**GET:** <http://localhost:8080/smartMenu/api/users>)
- ✓ Obtener usuario por ID (**GET:** <http://localhost:8080/smartMenu/api/users/4>)
- ✓ Obtener usuarios por rol (**GET:** <http://localhost:8080/smartMenu/api/users/rol/1>)
- ✓ Crear un nuevo usuario (**POST:** <http://localhost:8080/smartMenu/api/users>)
- ✓ Modificar usuario (**PUT:** <http://localhost:8080/smartMenu/api/users/5>)
- ✓ Eliminar Usuario por ID (**DELETE:** <http://localhost:8080/smartMenu/api/users/5>)

▼ Platillo

- ✓ Mostrar todos los platillos existentes (GET: <http://localhost:8080/smartMenu/api/platillos>)
- ✓ Crear platillos (POST: <http://localhost:8080/smartMenu/api/platillos>)
- ✓ Actualizar platillo (PUT: <http://localhost:8080/smartMenu/api/platillos/4>)
- ✓ Eliminar platillos (DELETE: <http://localhost:8080/smartMenu/api/platillos/7>)
- ✓ Obtener platillos disponibles (GET: <http://localhost:8080/smartMenu/api/platillos/disponibles>)
- ✓ Obtener platillo por id (GET: <http://localhost:8080/smartMenu/api/platillos/4>)

▼ Pedidos

- ✓ Obtener todos los pedidos (sin detalles) (GET: <http://localhost:8080/smartMenu/api/pedidos>)
- ✓ Obtener un pedido completo por ID (con detalles de platillos) (GET: <http://localhost:8080/smartMenu/api/pedidos/2>)
- ✓ Actualizar estado de un pedido (PUT: <http://localhost:8080/smartMenu/api/pedidos/2/estado>)
- ✓ Eliminar un pedido (DELETE: <http://localhost:8080/smartMenu/api/pedidos/31>)
- ✓ Obtener pedidos por estado (GET: <http://localhost:8080/smartMenu/api/pedidos/estado/en-preparacion>)
- ✓ Crear pedidos (POST: <http://localhost:8080/smartMenu/api/pedidos>)

▼ Clientes

- ✓ Mostrar todos los clientes (GET: <http://localhost:8080/smartMenu/api/clientes>)
- ✓ Obtener clientes por ID (GET: <http://localhost:8080/smartMenu/api/clientes/33>)
- ✓ Crear cliente (POST: <http://localhost:8080/smartMenu/api/clientes>)
- ✓ Modificar cliente (PUT: <http://localhost:8080/smartMenu/api/clientes/33>)
- ✓ Eliminar Cliente (DELETE: <http://localhost:8080/smartMenu/api/clientes/31>)

▼ Inventario

- ✓ Obtener todos los insumos (GET: <http://localhost:8080/smartMenu/api/inventario>)
- ✓ Obtener insumo por id (GET: <http://localhost:8080/smartMenu/api/inventario/1>)
- ✓ Crear nuevo insumo (POST: <http://localhost:8080/smartMenu/api/inventario>)
- ✓ Actualizar insumo (PUT: <http://localhost:8080/smartMenu/api/inventario/51>)
- ✓ Actualizar cantidad de un insumo (PUT: <http://localhost:8080/smartMenu/api/inventario/51/cantidad>)
- ✓ Eliminar insumo (DELETE: <http://localhost:8080/smartMenu/api/inventario/51>)

▼ Proveedores

- ✓ Mostrar todos los proveedores
- ✓ Agregar proveedor
- ✓ Modificar proveedor
- ✓ Eliminar proveedor

baby