

Prueba Técnica API

Nombre de la prueba: TiendaAPI

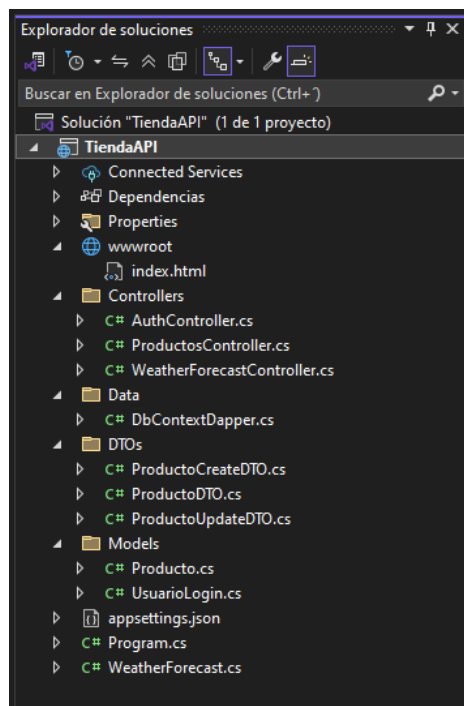
Herramientas utilizadas:

- ASP.NET Core 7 – API REST
- PostgreSQL 15
- Dapper (Micro ORM)
- JWT (Json Web Token)
- Swagger
- HTML + JavaScript

Descripción:

Esta API permite gestionar productos en una tienda, incluyendo su creación, consulta, actualización y eliminación. Cuenta con autenticación segura mediante tokens JWT y conexión a una base de datos PostgreSQL usando Dapper. Incluye un frontend sencillo para probar la API desde el navegador.

Estructura de la TiendaAPI:



Funcionalidades implementadas

CRUD de productos

Productos		^
GET	/api/Productos	✓ 🔒
POST	/api/Productos	✓ 🔒
PUT	/api/Productos/{id}	✓ 🔒
DELETE	/api/Productos/{id}	✓ 🔒

- GET /api/productos → Listar productos
- POST /api/productos → Crear nuevo producto
- PUT /api/productos/{id} → Actualizar producto
- DELETE /api/productos/{id} → Eliminar producto

Autenticación JWT

Auth ^

POST /auth/login ⌵

Usuario quemado:

```
{
  "usuario": "admin",
  "contrasena": "1234"
}
```

Token:

[illegible]

Validaciones en POST y PUT

```
[HttpPost]
0 referencias
public async Task<ActionResult> CrearProducto([FromBody] ProductoCreateDTO productoDTO)
{
    // Validaciones
    if (productoDTO.Precio < 0)
        return BadRequest("El precio no puede ser negativo.");

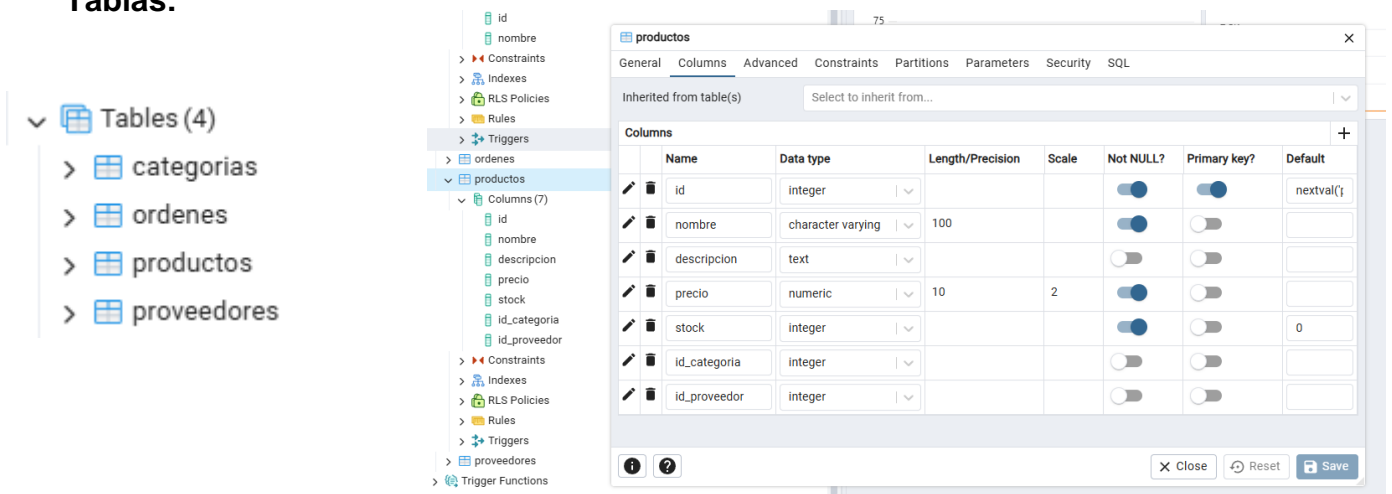
    if (productoDTO.Stock < 0)
        return BadRequest("El stock no puede ser negativo.");

    if (string.IsNullOrEmpty(productoDTO.Nombre))
        return BadRequest("El nombre del producto es obligatorio.");
}
```

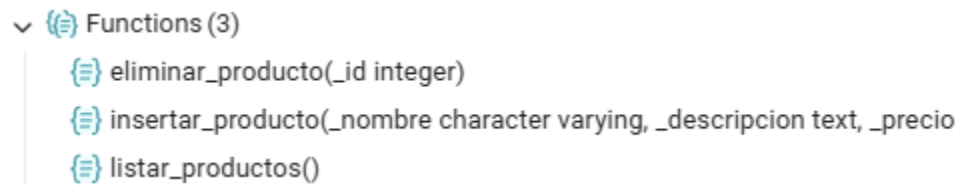
Base de Datos

Nombre de la base de datos: tienda_db

Tablas:



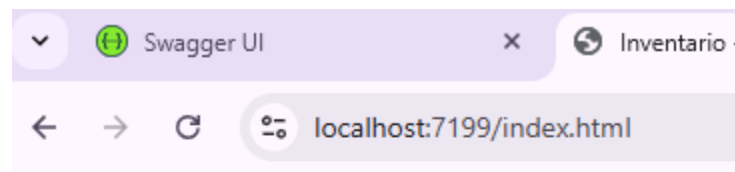
Funciones creadas:



Manejo de errores:



HTML Básico



Listado de Productos

Cargar productos

- Panquecitos - \$2500
- JABON ARIEL - \$12000
- Gomas - \$1500
- Curry - \$16000
- Lavaloz - \$10000
- Suavitel - \$35000
- Salsa - \$0