

Guía de Uso y Ejecución de la API ... Mas abajo están los capture de todo

Mi API se llama **Gestión de Usuarios**

La cual se encarga de gestionar usuarios a través de operaciones CRUD utilizando lo que es ASP.NET Core y Entity Framework Core.

Y se estará utilizando Swagger.

Ejecutar la API

Una vez la API esté en ejecución, abre tu navegador y vas a <http://localhost:5001/swagger/index.html> Cuando estes en Swagger, podras probar todos los endpoints para gestionar los usuarios con el CRUD que está ahí.

Pasos para usar correctamente el CRUD

1. Selecciona el endpoint que quieres probar.
2. Haz clic en el botón "Try it out".
3. Ingresa los valores necesarios y haz clic en "Execute".
4. Verifica la respuesta en la sección "Responses".

6. Los errores que se pueden encontrar

SQL Server no se encuentra

Asegúrate de que el servicio SQL Server está en marcha o conectado

400 Bad Request - JSON inválido

Verifica que el JSON esté bien hecho

El correo electrónico ya está en uso

Prueba con otro correo o elimina el usuario duplicado.

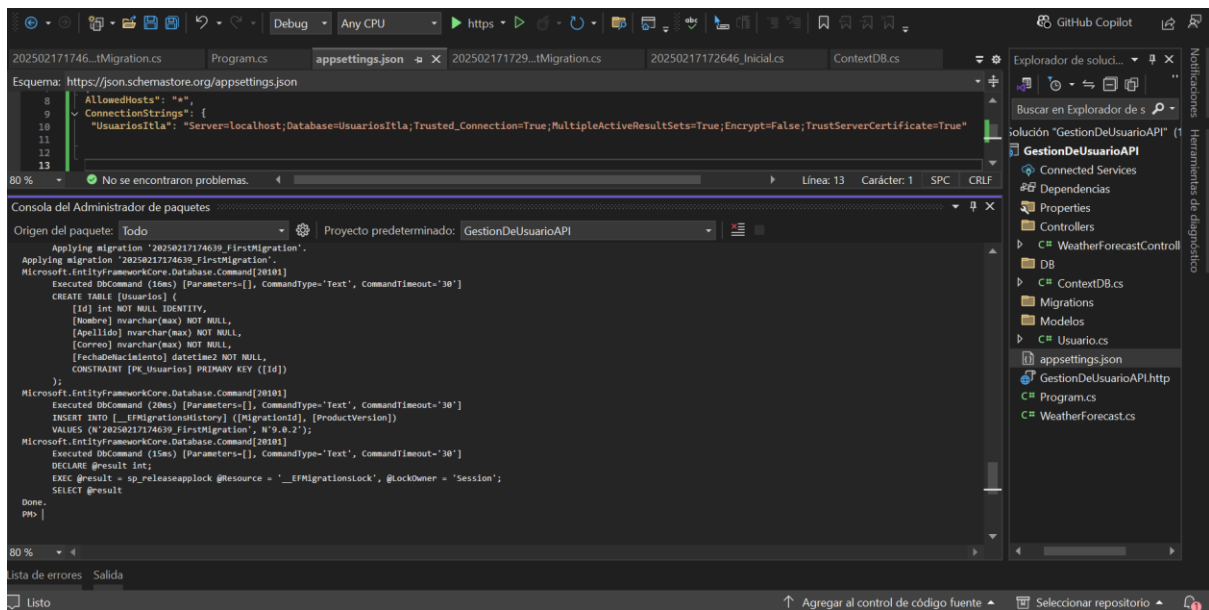
404 No encontrado

Asegúrate de que el usuario con ese ID existe en la base de datos

Con esta API puedes gestionar usuarios de manera eficiente utilizando API en Swagger

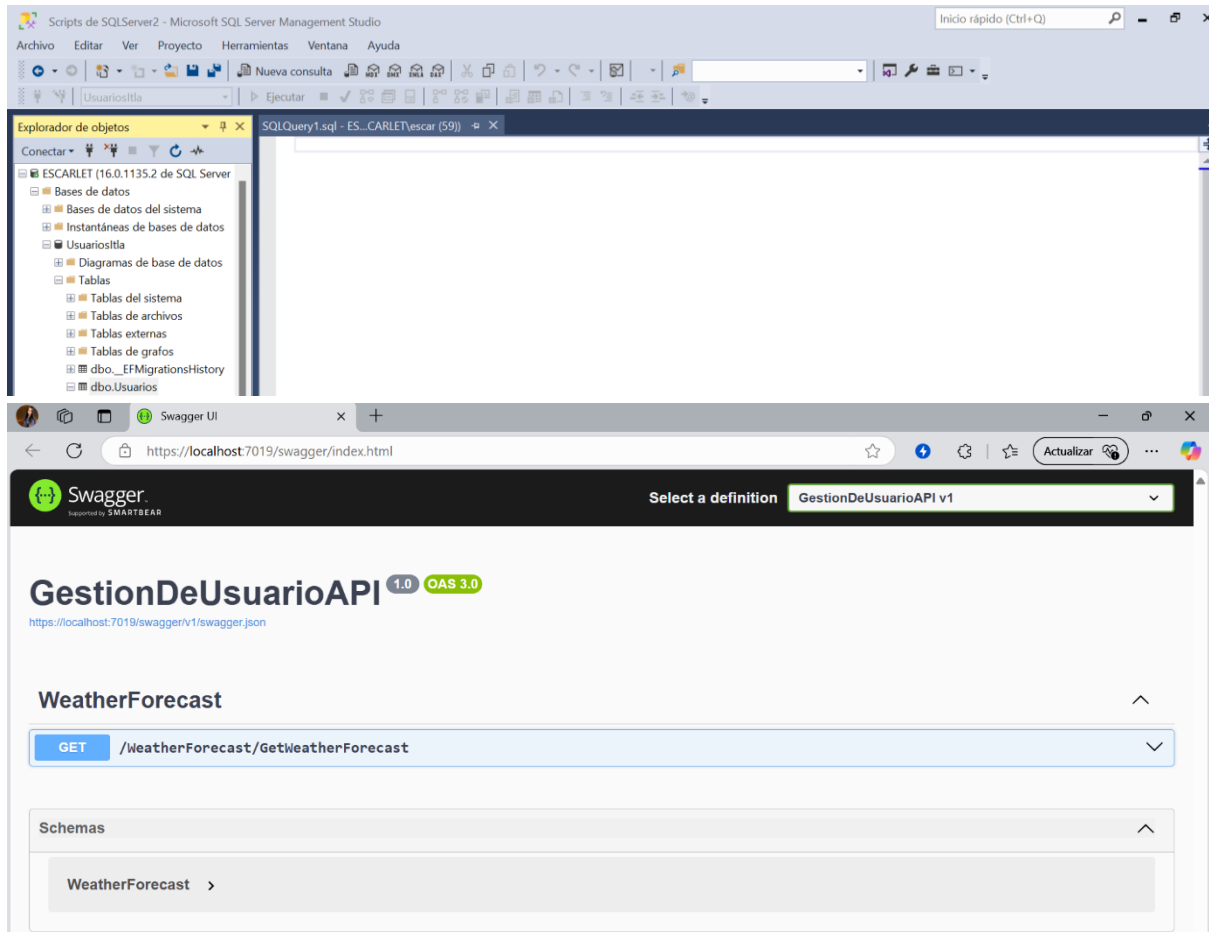
Autor: Escarlet Montero Arias

La correcta migración



```
Esquema: https://json.schemastore.org/appsettings.json
8 AllowedHosts": "*"
9 ConnectionStrings": {
10   "UsuariosItla": "Server=localhost;Database=UsuariosItla;Trusted_Connection=True;MultipleActiveResultSets=True;Encrypt=False;TrustServerCertificate=True"
11 }
12
13
80 % No se encontraron problemas. Línea: 13 Carácter: 1 SPC CRLF

Consola del Administrador de paquetes
Origen del paquete: Todo Proyecto predeterminado: GestionDeUsuarioAPI
Applying migration '20250217174639_FirstMigration'.
Applying migration '20250217174639_FirstMigration'.
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (10ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  CREATE TABLE [Usuarios] (
    [Id] int NOT NULL IDENTITY,
    [Nombre] nvarchar(max) NOT NULL,
    [Apellido] nvarchar(max) NOT NULL,
    [Correo] nvarchar(max) NOT NULL,
    [FechaDeNacimiento] datetime2 NOT NULL,
    CONSTRAINT [PK_Usuarios] PRIMARY KEY ([Id])
  );
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (20ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  INSERT INTO [__EFMigrationsHistory] ([MigrationId], [ProductVersion])
  VALUES (N'20250217174639_FirstMigration', N'9.0.2');
Microsoft.EntityFrameworkCore.Database.Command[20101]
  Executed DbCommand (15ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
  DECLARE @result int;
  EXEC @result = sp_releaseapplock @Resource = '__EFMigrationsLock', @LockOwner = 'Session';
  SELECT @result
Done.
PM>
```



Autor: Escarlet Montero Arias

The top part of the image shows a terminal window with the following output:

```
info: Microsoft.Hosting.Lifetime[14]
  Now listening on: https://localhost:7019
info: Microsoft.Hosting.Lifetime[14]
  Now listening on: http://localhost:5124
info: Microsoft.Hosting.Lifetime[0]
  Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
  Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
  Content root path: C:\Users\escar\Downloads\ITLA\Practica 5\GestionDeUsuarioAPI
```

The bottom part of the image shows a web browser window displaying the Swagger UI for the **GestionDeUsuarioAPI** (version 1.0, OAS 3.0). The URL is <https://localhost:7019/swagger/index.html>. The UI lists the following endpoints:

- Usuarios**
 - GET `/api/Usuarios`
 - POST `/api/Usuarios`
 - GET `/api/Usuarios/{id}`
 - PUT `/api/Usuarios/{id}`
 - DELETE `/api/Usuarios/{id}`
- WeatherForecast**
 - GET `/WeatherForecast/GetWeatherForecast`

ADD NEW USER

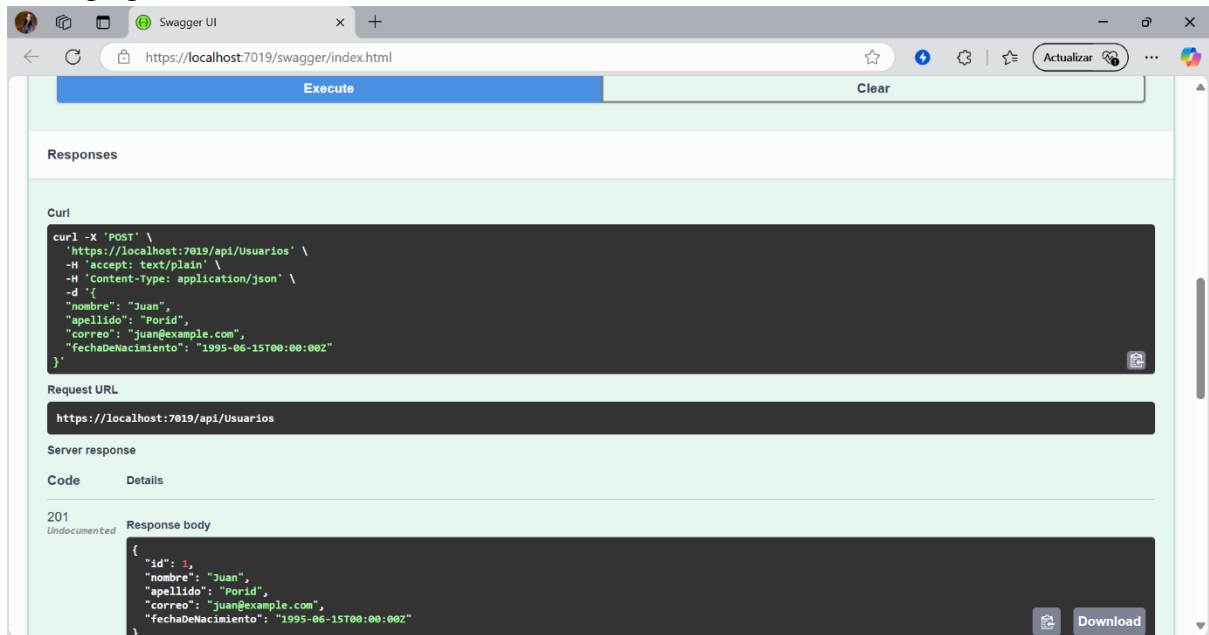
The screenshot shows the Swagger UI for the **Usuarios** section, specifically the **POST /api/Usuarios** endpoint. The interface includes a **Parameters** section with "No parameters" and a **Request body** section with a dropdown menu set to `application/json`. The request body is a JSON object:

```
{
  "nombre": "Juan",
  "apellido": "Perez",
  "correo": "juan@example.com",
  "fechaDeNacimiento": "1995-06-15T00:00:00Z"
}
```

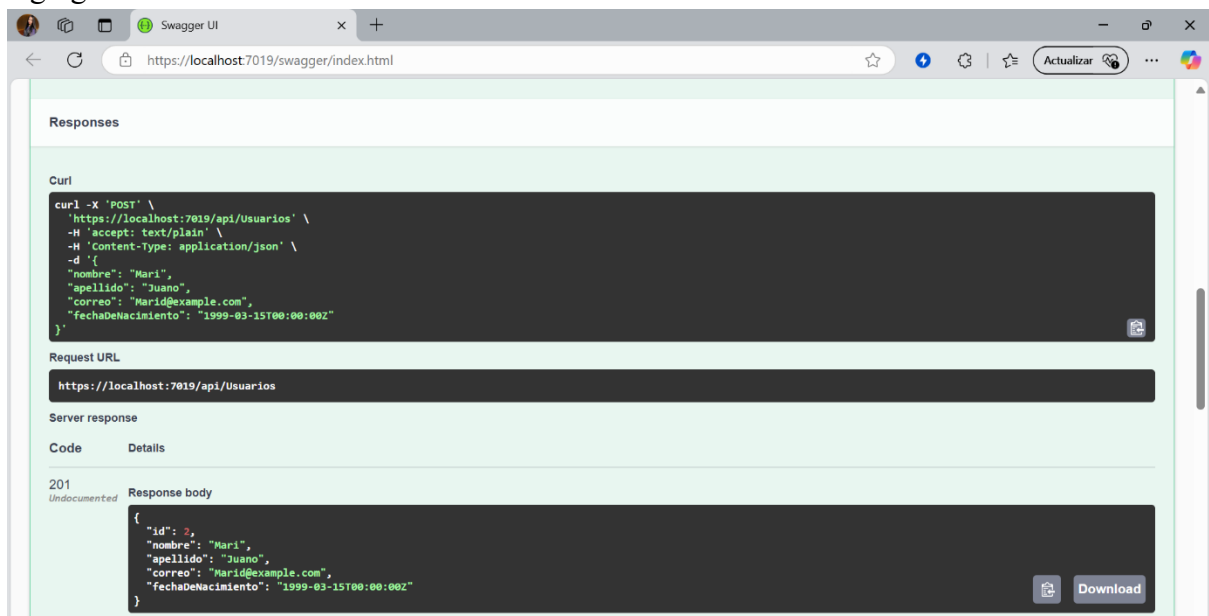
Buttons for **Cancel** and **Reset** are visible in the top right corner of the endpoint details.

Autor: Escarlet Montero Arias

201 Agregado correctamente



Agrego otro mas



Agrego otro con un correo ya existente



Autor: Escarlet Montero Arias

Mostrar Get

The top screenshot shows the Swagger UI interface for the endpoint `GET /api/Usuarios/{id}`. The parameter `id` is required and is set to `2`. The interface includes an `Execute` button and a `Clear` button. Below the parameter field, the `Responses` section is visible, showing the `Curl` command, the `Request URL`, and the `Server response` details.

The bottom screenshot shows the same interface after the `Execute` button has been clicked. The `Server response` section now displays the `Code` `200` and the `Response body` in JSON format:

```
{
  "id": 2,
  "nombre": "Mari",
  "apellido": "Juano",
  "correo": "Mari@example.com",
  "fechaDeNacimiento": "1999-03-15T00:00:00"
}
```

Eliminar

The screenshot shows the Swagger UI interface for the endpoint `DELETE /api/Usuarios/{id}`. The parameter `id` is required and is set to `2`. The interface includes an `Execute` button and a `Cancel` button. Below the parameter field, the `Responses` section is visible, showing the `Curl` command, the `Request URL`, and the `Server response` details.

The `Curl` command is:

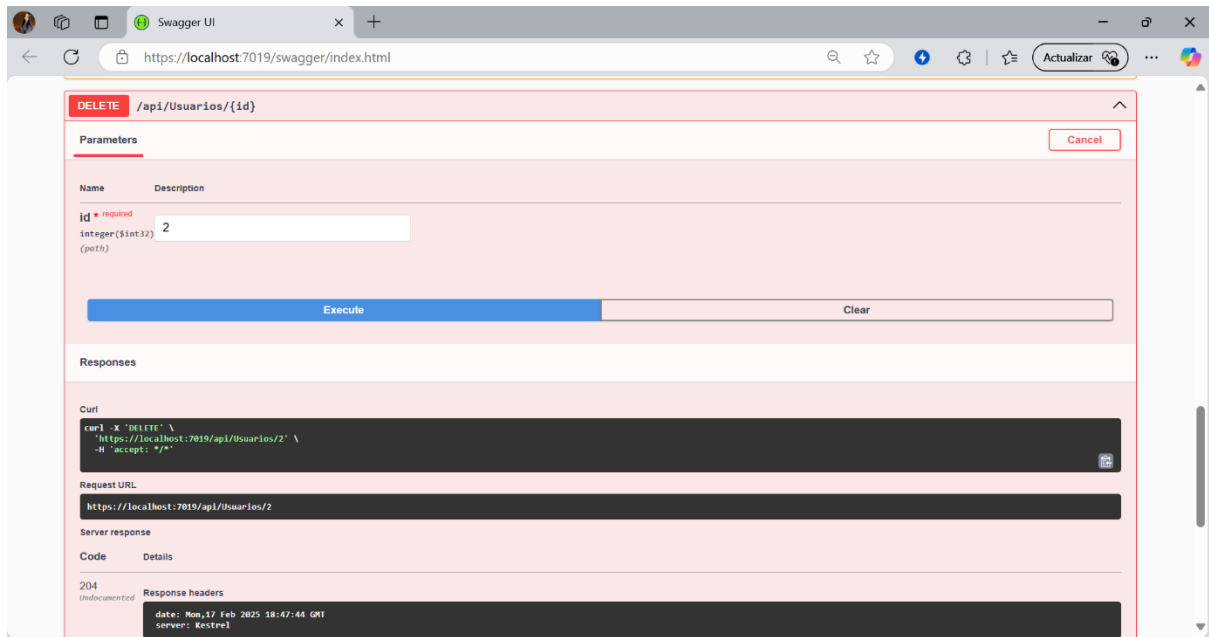
```
curl -X 'DELETE' \
  'https://localhost:7019/api/Usuarios/2' \
  -H 'accept: */*'
```

The `Request URL` is:

```
https://localhost:7019/api/Usuarios/2
```

Autor: Escarlet Montero Arias

Eliminado correctamente



Leer todos

