

# ContactManager\_API

---

Este proyecto es una API RESTful para la gestión de contactos desarrollada en ASP.NET Core 6, utilizando Entity Framework Core y autenticación JWT.

## Requisitos

- .NET 6 SDK
- SQL Server (local o remoto)

## Dependencias NuGet principales

Proyecto principal (**ContactManagerAPI**)

- **Microsoft.AspNetCore.Authentication.JwtBearer**  
Permite la autenticación y validación de tokens JWT en la API.
- **Microsoft.EntityFrameworkCore**  
ORM para trabajar con bases de datos relacionales usando el enfoque Code First.
- **Microsoft.EntityFrameworkCore.SqlServer**  
Proveedor de Entity Framework Core para SQL Server.
- **Microsoft.EntityFrameworkCore.Design**  
Herramientas para migraciones y diseño de base de datos.
- **Microsoft.IdentityModel.Tokens**  
Utilidades para la validación y manejo de tokens JWT.
- **System.IdentityModel.Tokens.Jwt**  
Permite la creación y validación de JWT.
- **Swashbuckle.AspNetCore**  
Genera documentación Swagger/OpenAPI para la API.
- **xunit**  
(Solo para pruebas, puede estar presente por error en el proyecto principal).

Proyecto de pruebas (**ContactManagerAPI.Tests**)

- **xunit**  
Framework de pruebas unitarias.
- **xunit.runner.visualstudio**  
Permite ejecutar pruebas xUnit desde Visual Studio y la línea de comandos.
- **Microsoft.NET.Test.Sdk**  
Infraestructura para ejecutar pruebas en .NET.

- **Microsoft.AspNetCore.Mvc.Testing**  
Facilita pruebas de integración para aplicaciones ASP.NET Core.
- **Microsoft.EntityFrameworkCore.InMemory**  
Proveedor de EF Core para pruebas en memoria (sin base de datos real).
- **Moq**  
Framework para crear objetos simulados (mocks) en pruebas unitarias.
- **Castle.Core**  
Dependencia interna de Moq para la creación de proxies en pruebas.
- **coverlet.collector**  
Herramienta para medir la cobertura de código en las pruebas.

## Configuración

- La cadena de conexión y los parámetros de JWT se encuentran en `appsettings.json`.
- Para ejecutar migraciones de base de datos, asegúrate de tener instalado el paquete `Microsoft.EntityFrameworkCore.Design`.

## Ejecución

1. Restaura los paquetes NuGet:

```
dotnet restore
```

2. Aplica las migraciones (si es necesario):

```
dotnet ef database update
```

3. Ejecuta la API:

```
dotnet run --project ContactManagerAPI
```

4. (Opcional) Ejecuta las pruebas:

```
dotnet test
```

## Swagger

La documentación interactiva de la API estará disponible en `/swagger` cuando la aplicación esté en modo desarrollo.

**Nota:**

Si agregas o quitas dependencias, actualiza este README para reflejar los cambios.