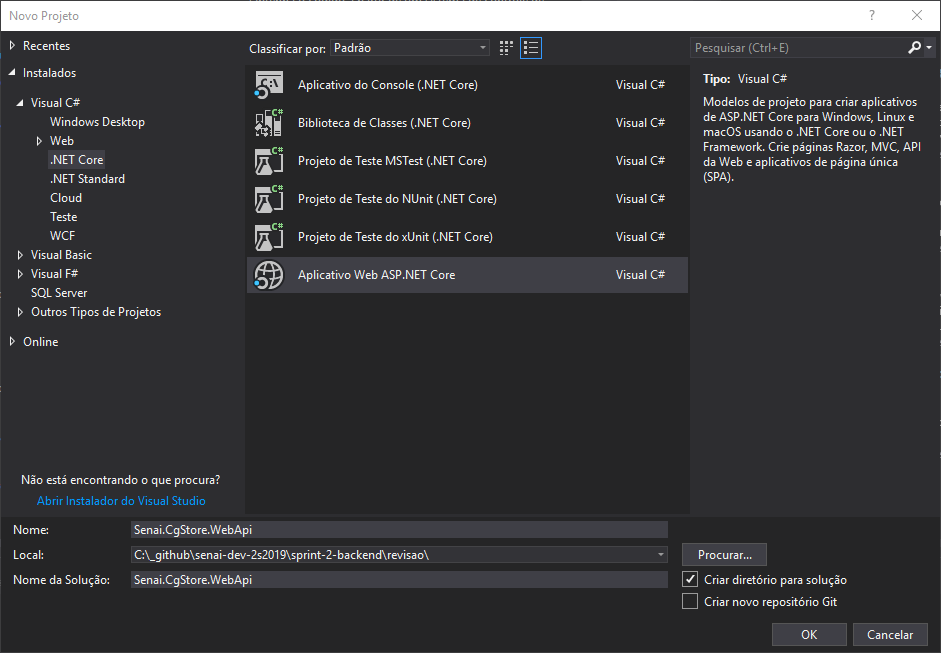
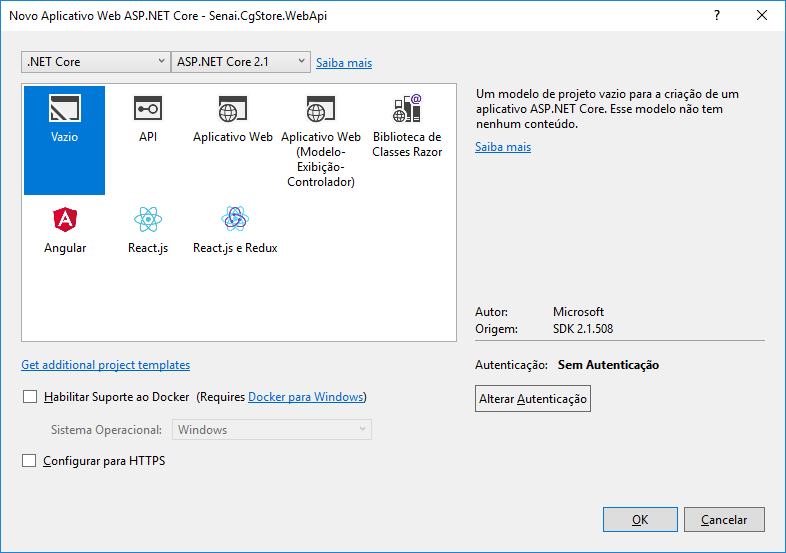
Executar o script para a criação do banco de dados e tabelas:

cgstore\_revisao.sql

Criar Projeto





Gerenciar Pacotes do NuGet e instalar as dependências:

*Token - JWT*

System.IdentityModel.Tokens.Jwt (5.5.0) - criar e validar o jwt

Microsoft.AspNetCore.Authentication.JwtBearer (2.1.1) - integrar a parte de autenticação

*EFCore*

Microsoft.EntityFrameworkCore.SqlServer (2.1.11)

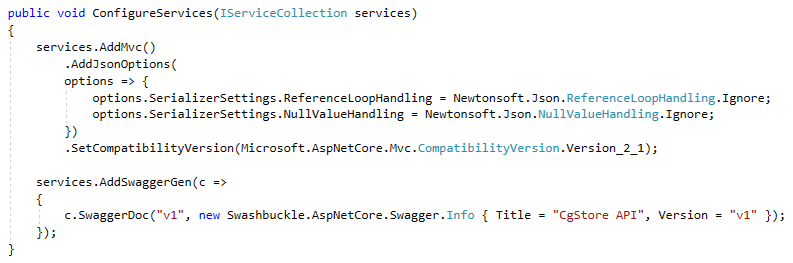
Microsoft.EntityFrameworkCore.SqlServer.Design (1.1.6)

Microsoft.EntityFrameworkCore.Tools (2.1.11)

*Swagger*

Swashbuckle.AspNetCore (4.0.1)

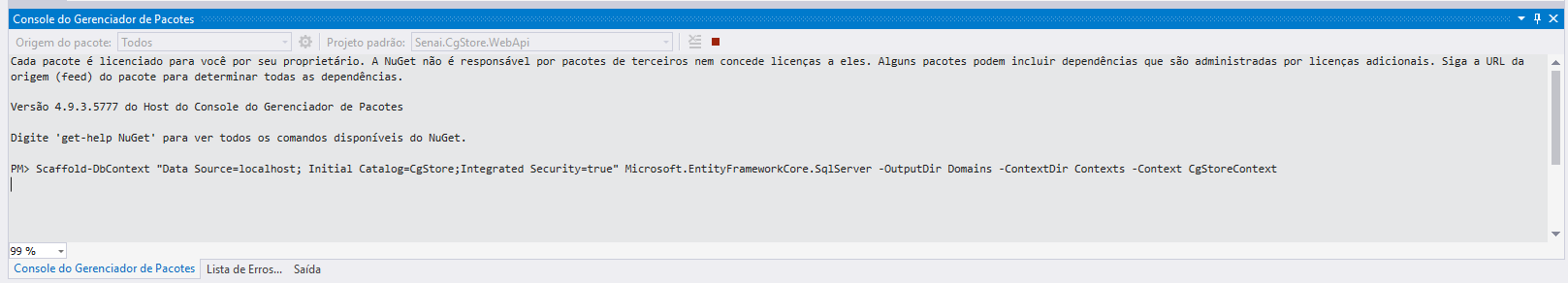
Configurar o startup o MVC e o Swagger.



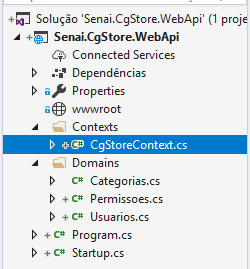


Como estamos utilizando o EFCore, vamos realizar o Scaffold para dar início aos modelos da nossa aplicação.

Scaffold-DbContext "Data Source=localhost; Initial Catalog=CgStore;Integrated Security=true" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Domains -ContextDir Contexts -Context CgStoreContext



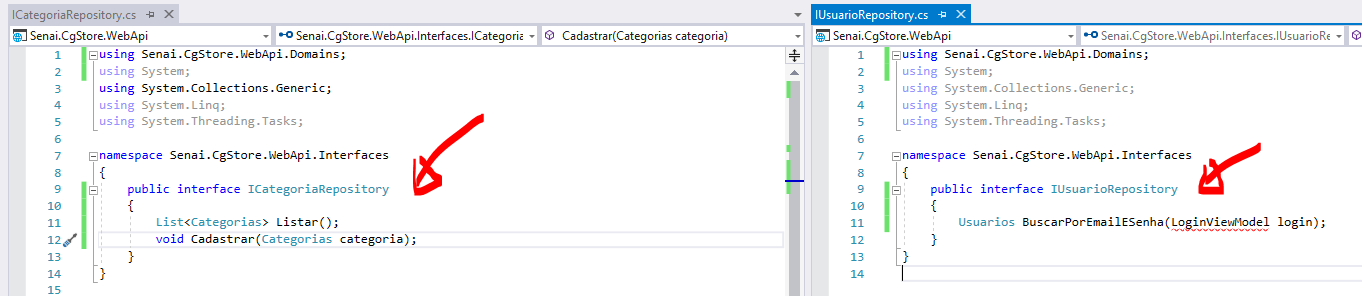
Solução com o projeto.



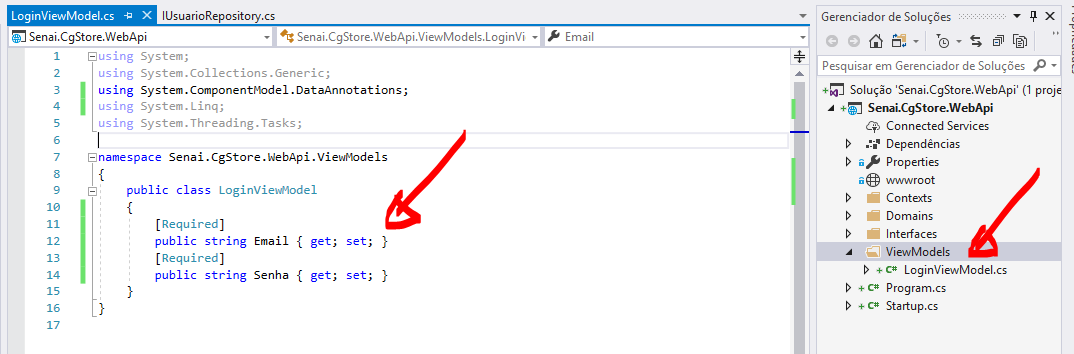
Criar as interfaces antes de criar os repositórios, bem como antes de criar os controllers. Para já definir o que será utilizado.

ICategoriaRepository.cs

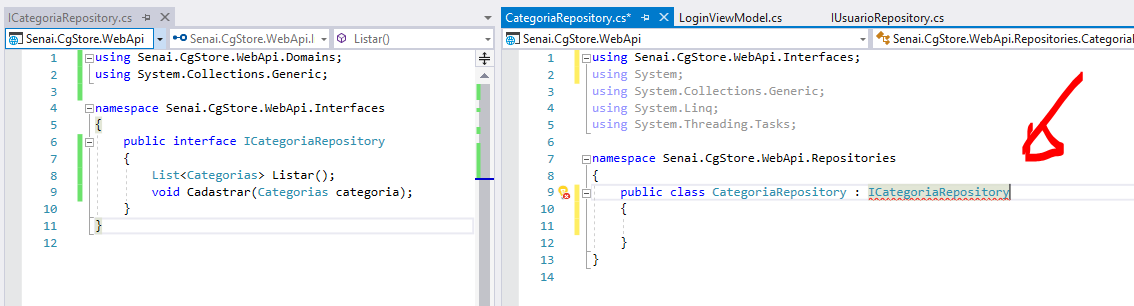
IUsuarioRepository.cs

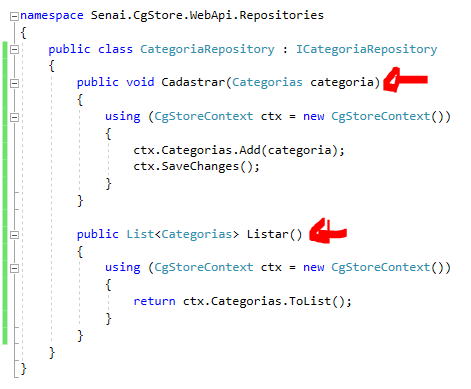


Criar o LoginViewModel e realizar o using na interface.



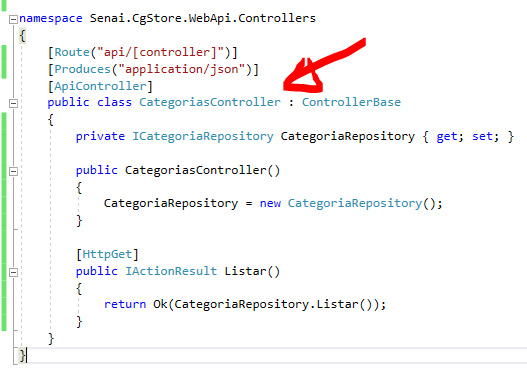
Criar uma classe CategoriaRepository e realizar a chamada a interface.





Criar o Controller de Categorias para realizar a listagem.

CategoriasController.cs



Lembrar de: launchSettings.json, alterar a opção do navegador para falso.

Lembrar de incluir os comentários no controller e no repositório

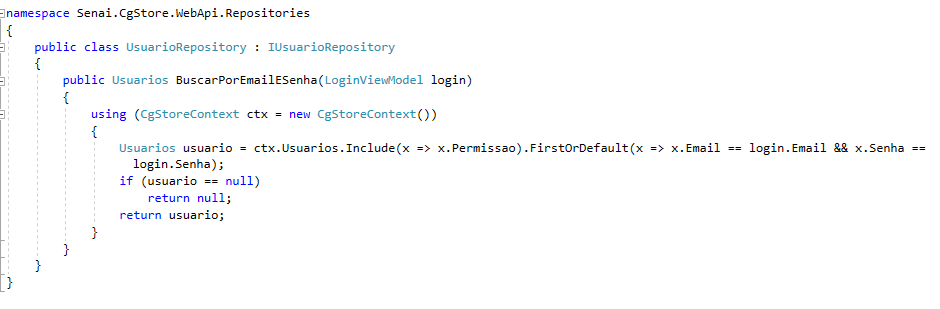
Postman

Criar uma coleção: Senai.CgStore e uma pasta Categorias.

Criar uma requisição Categorias.Listar.

Após apresentar a lista, vamos criar o login para gerar o token e restringir o acesso ao cadastro de novas categorias.

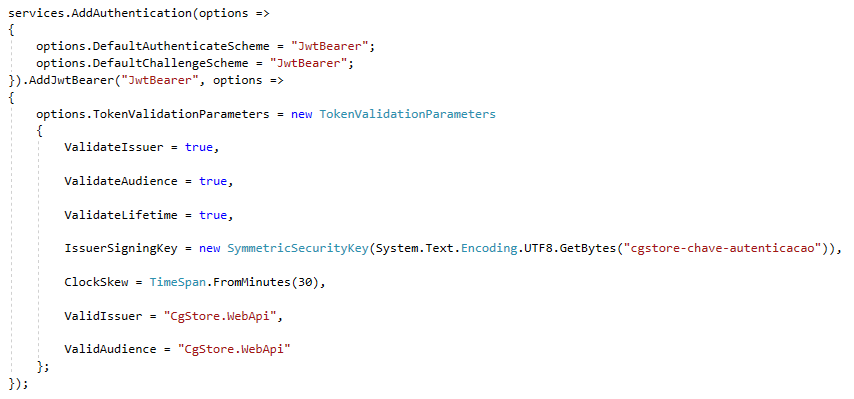
Mas antes disso, vamos criar o repositório para ter acesso ao banco de dados.



Para configurar toda a parte do token e gerá-lo, precisamos de duas configurações.

Primeira: Configurar o startup.cs.

ConfigureServices



Configure



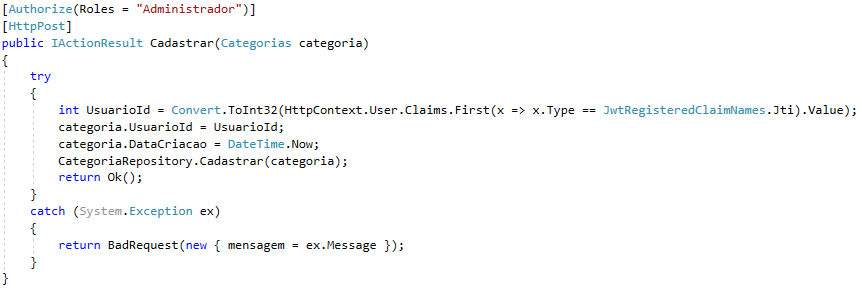
Após isto, precisamos criar o controller para gerar o token.

LoginController.cs



Validar no postman e no jwt.io se o token está válido com as informações solicitadas.

Realizar o cadastro no CategoriasController.cs.



Validar o Swagger.

Na inclusão de um novo item, como BuscarPorId, primeiro atualizar a interface e depois realizar a chamada no repositório.

Depois, incluir no controller.

