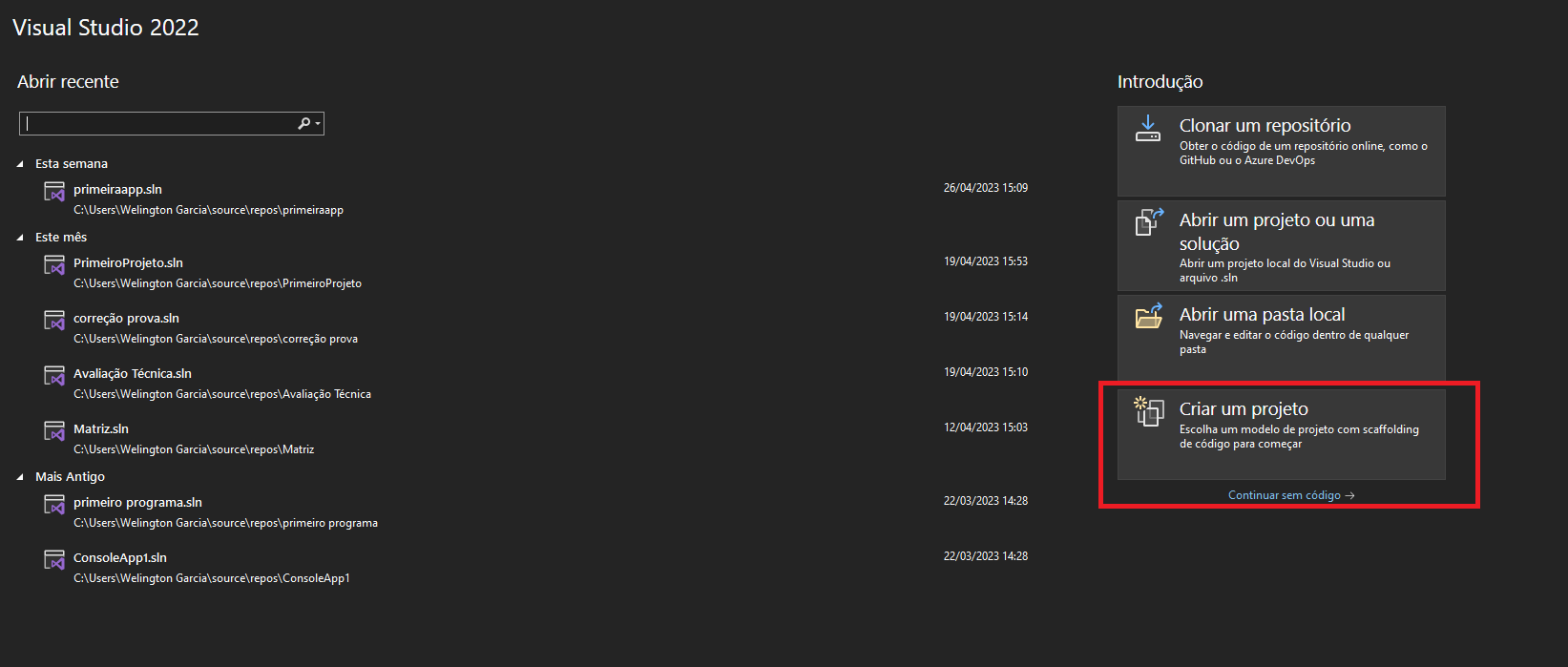
Criando aplicação MVC

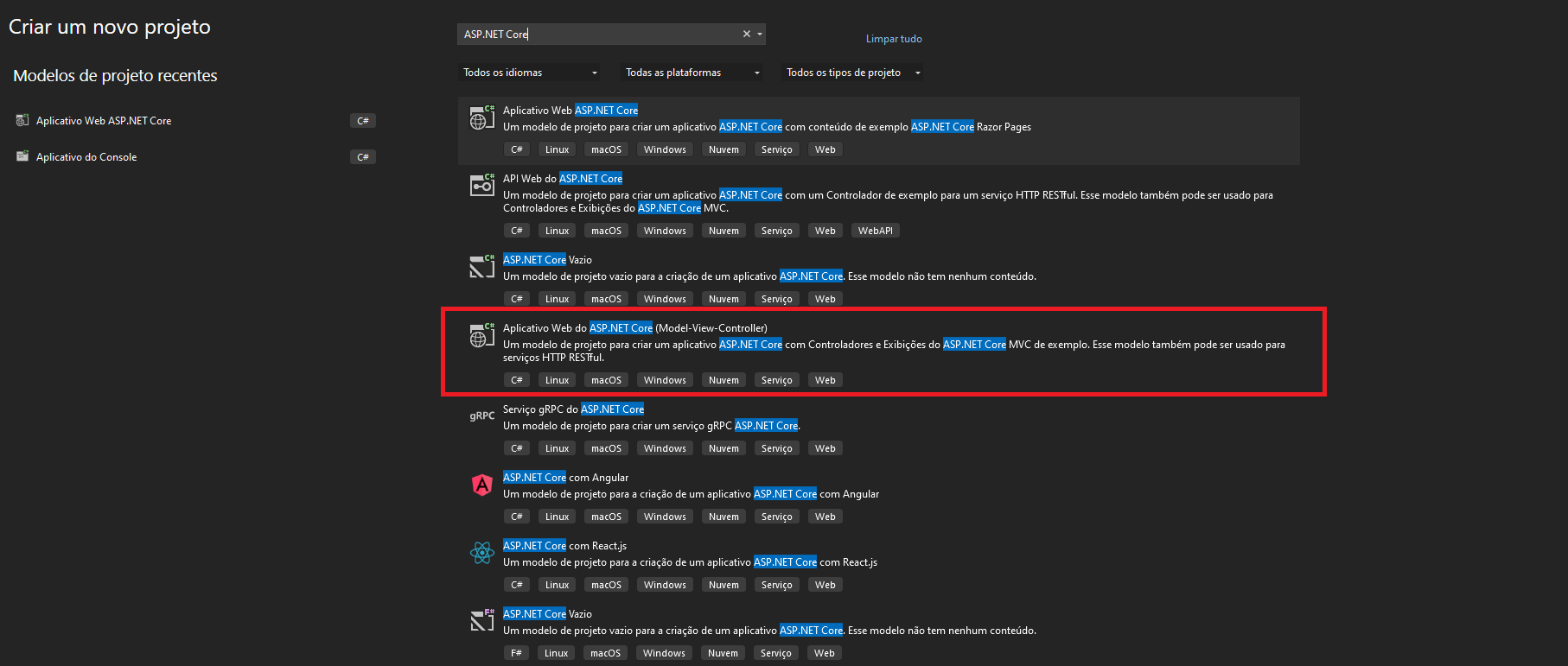
Passo 1

Acesse a opção criar um projeto



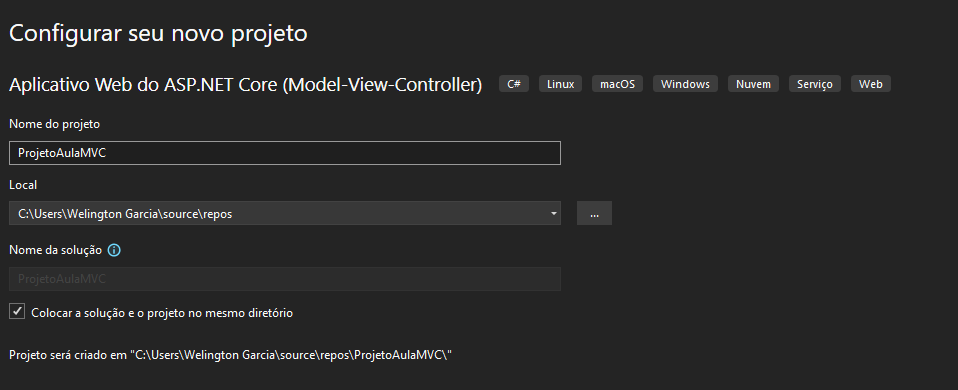
Passo 2

Pesquise por ASP.NET Core e selecione a opção de projeto para desenvolvimento web com MVC, como indicado a imagem abaixo



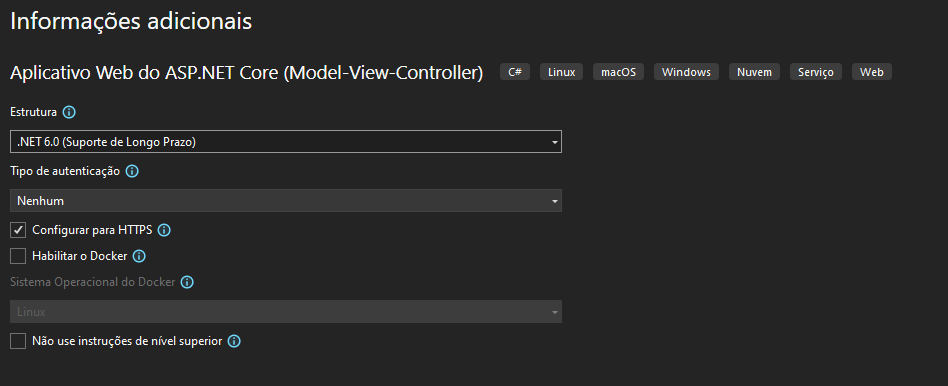
Passo 3

Defina o nome do projeto



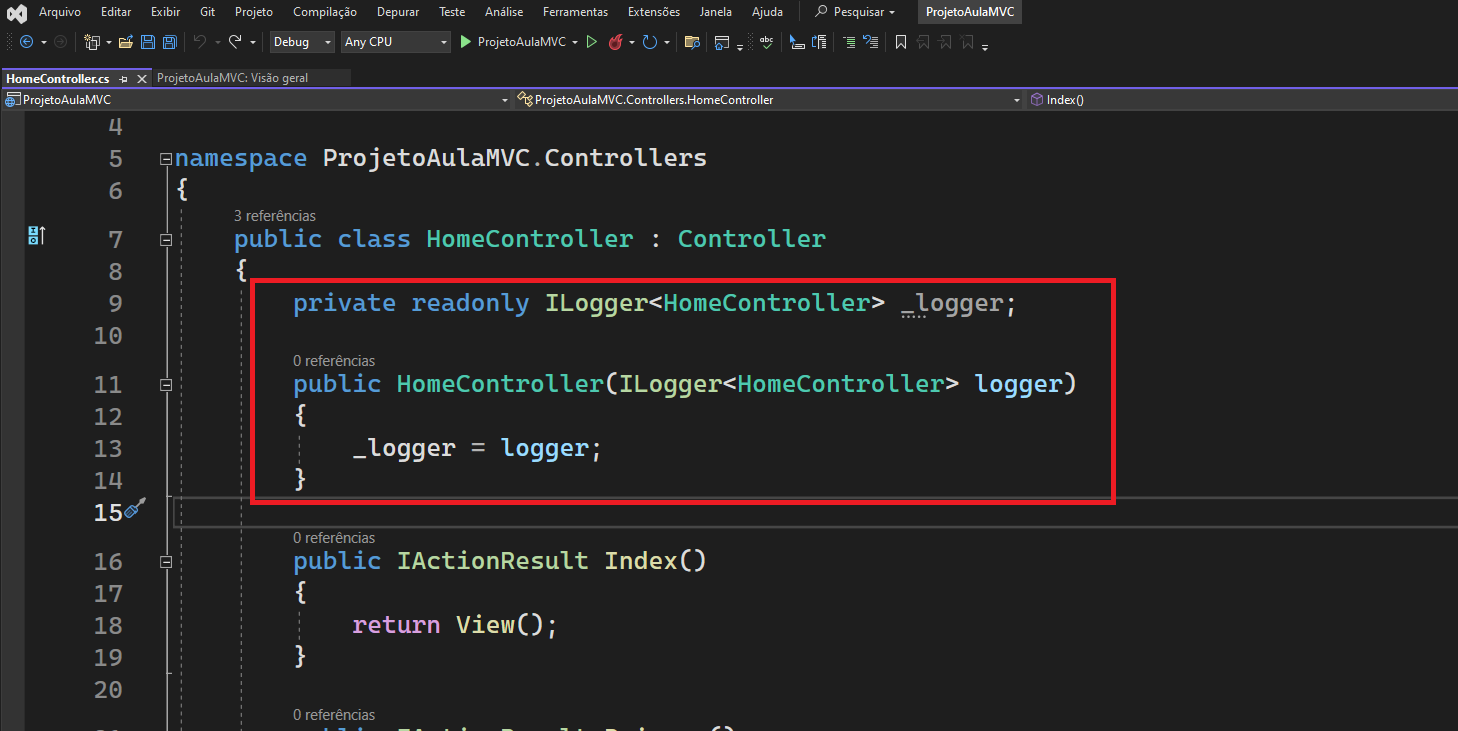
Passo 4

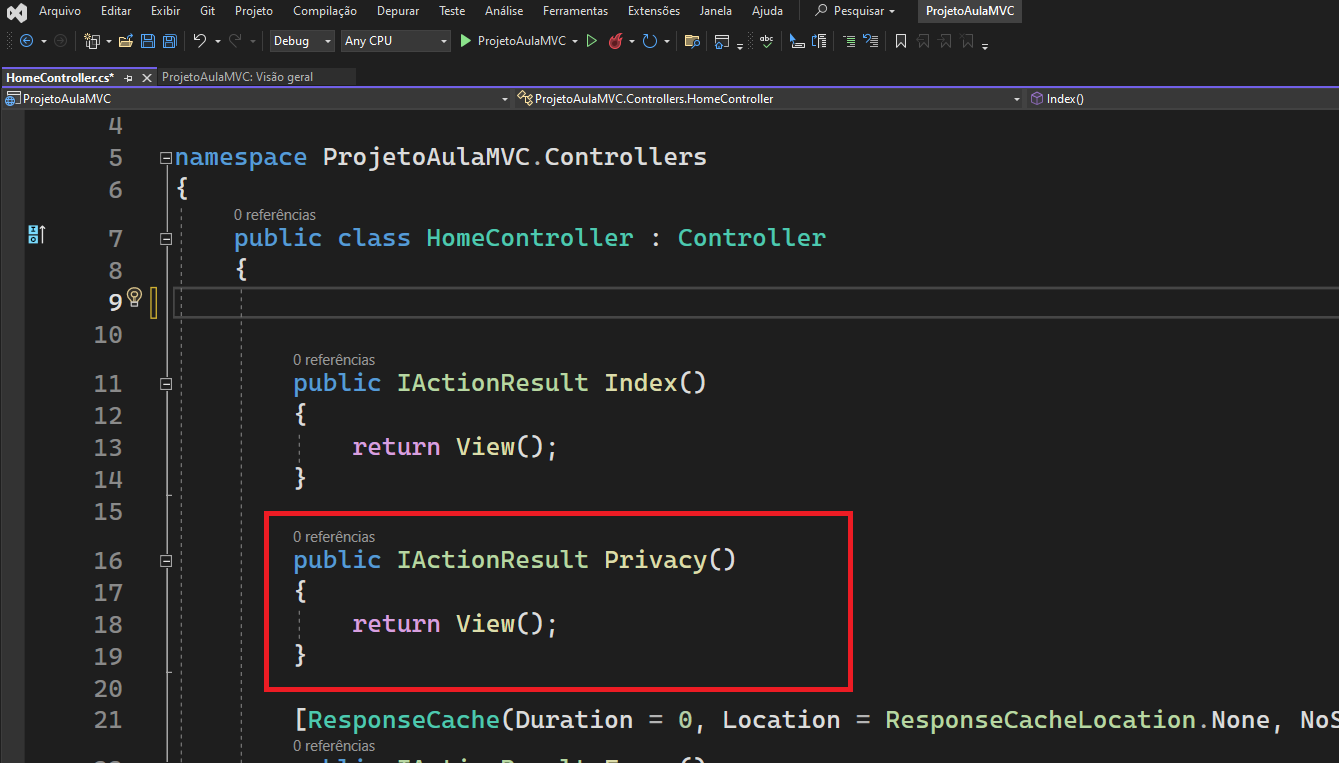
Selecione o Framework .NET 6.0 e termine a criação do projeto



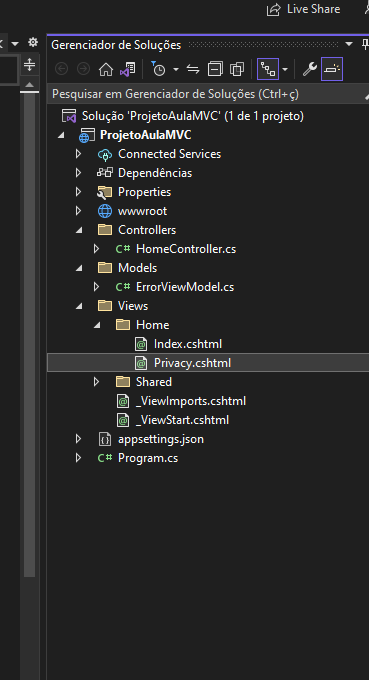
Passo 5

Com o projeto criado acesse dentro de Controllers o arquivo HomeController.cs e apague as seguintes linhas de código.

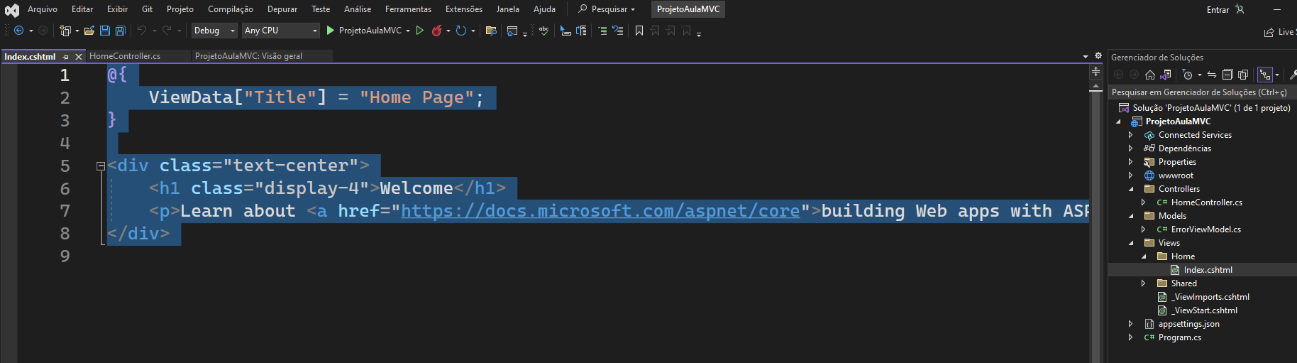




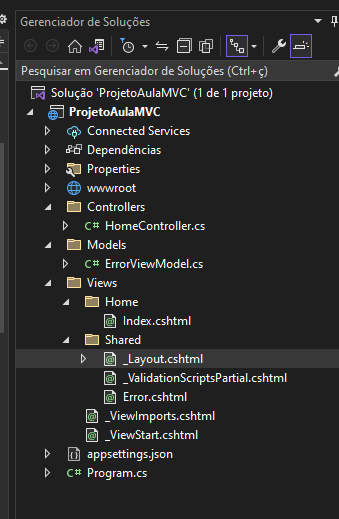
Exclua na seguida o arquivo Privacy na home do projeto



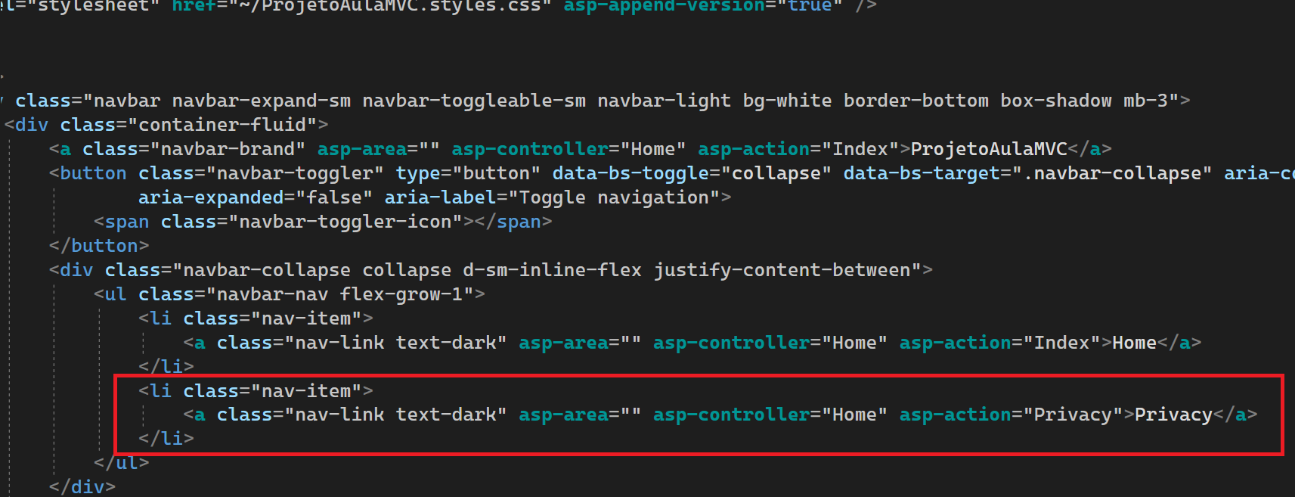
Remova todo o código da Index



Acesse Layout

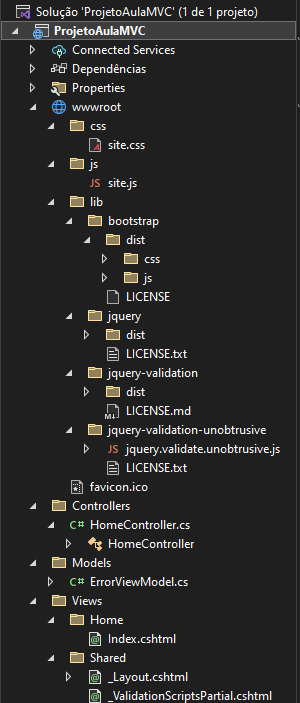


Remova o seguinte liste item



Fazendo essa ação o você estará removendo a opção Privacy do menu da página. Para adicionar mais opção basta adicionar mais itens lists nesta tela.

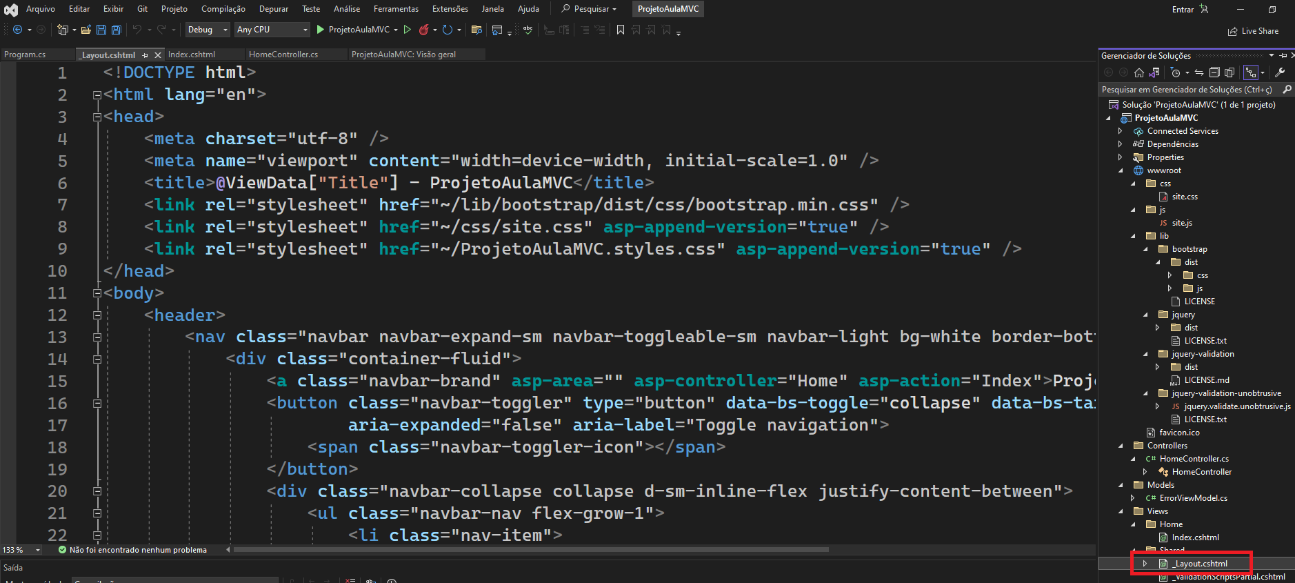
Passo 6

Vamos verificar os arquivos para desenvolvimento. 

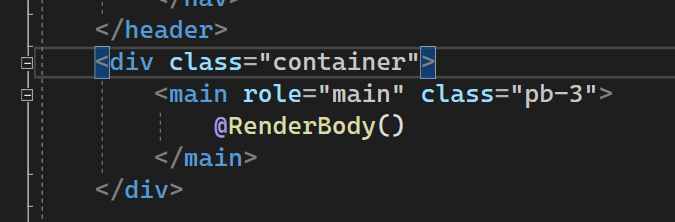
Note que o projeto já traz arquivos do bootstrap instalados, além de vim configurado para o padrão MVC. Encontramos pastas com a divisão para Controle, Modelos e Visões.

Passo 7

Para verificar de onde vem os demais itens da página inicial basta abrir o arquivo \_Layout.cshtml na pasta shared em view, neste arquivo tem a definição de todos os elementos usado na página inicial.



Neste arquivo está a configuração padrão de todas as páginas como cabeçalhos e rodapés e os conteúdos que são mostrados no corpo da página será demostrado através da div contendo a classe container. Como na figura abaixo:



O conteúdo deste arquivo é compartilhado com todas as views do projeto.

**Projetos de Lanches**

Neste tópico vamos criar um carousel para a página inicial do projeto.

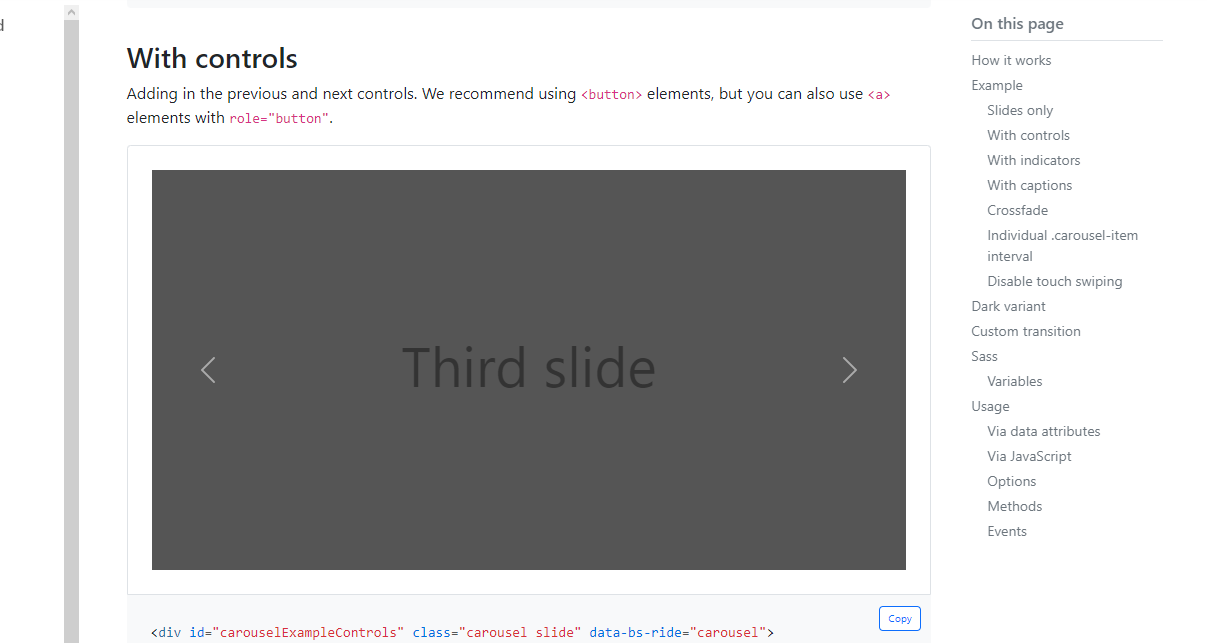
Passo 1

Copie o código de um carousel na documentação do bootstrap este projeto usa o bootstrap 5.1.

Segue o link

<https://getbootstrap.com/docs/5.1/components/carousel/>

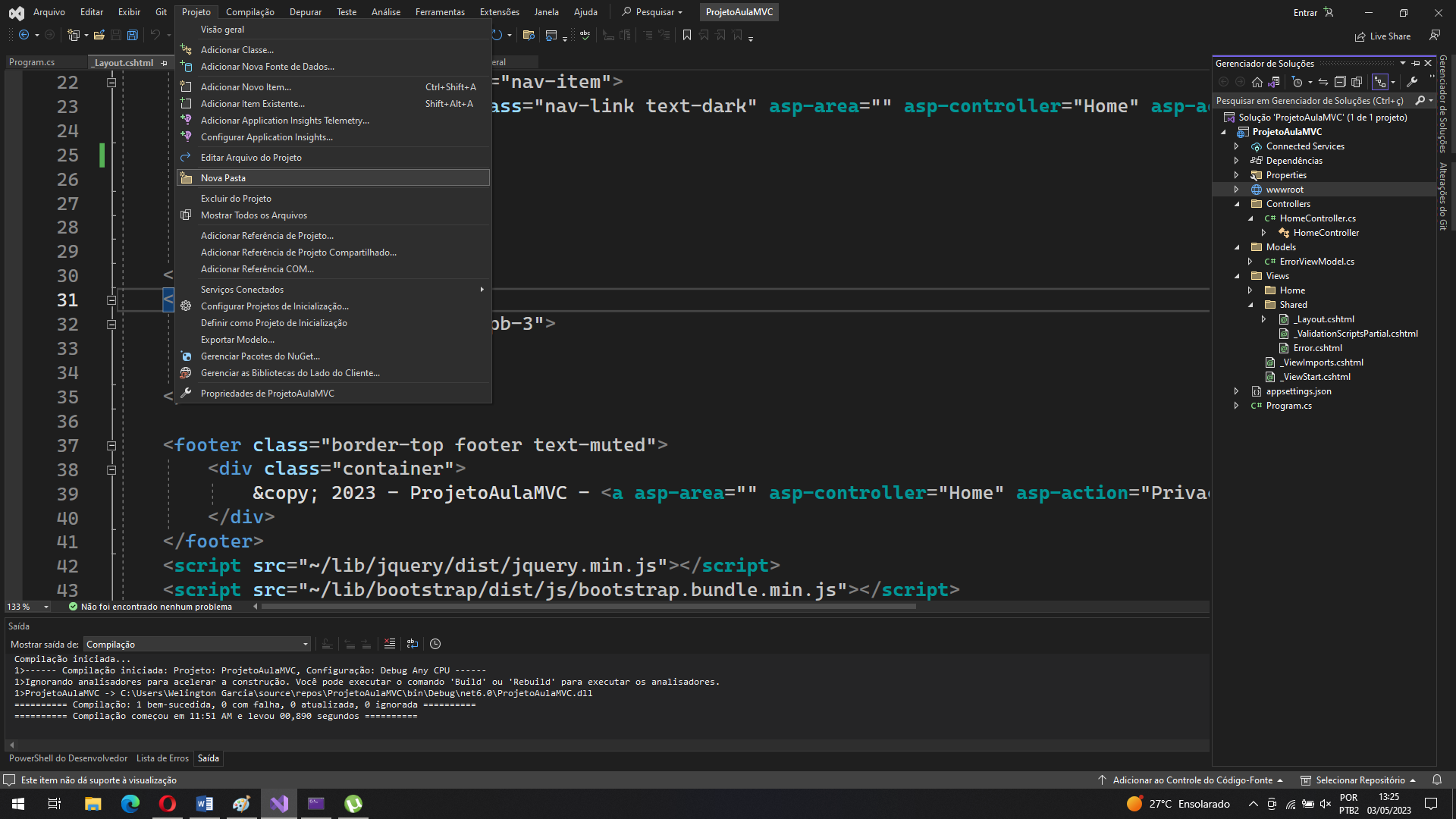
Vamos pegar a versão do carousel com controlers como na imagem abaixo.

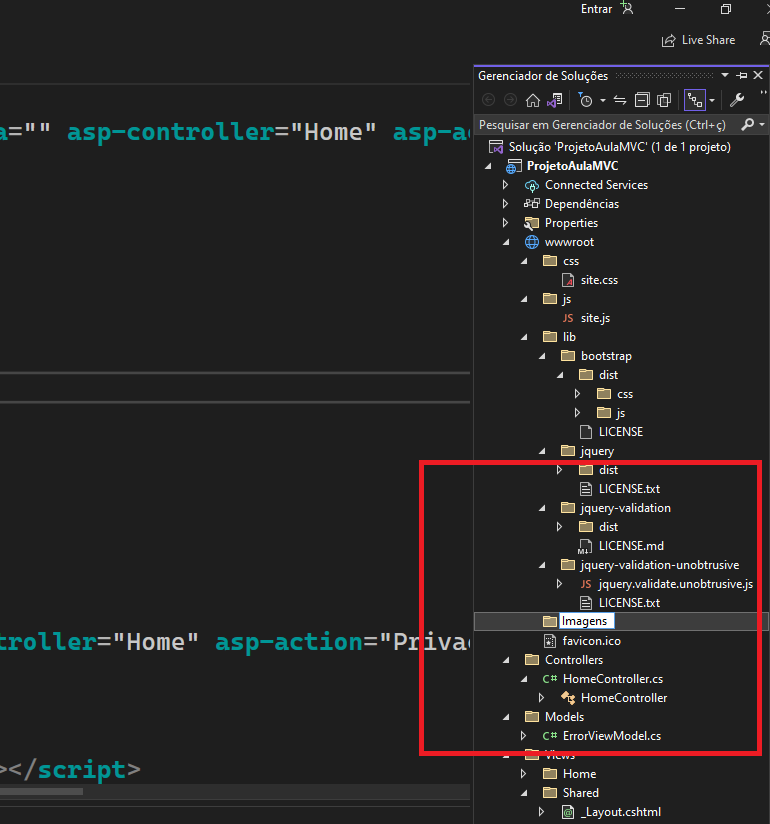


Copie o código e leve para o projeto, da seguinte maneira.

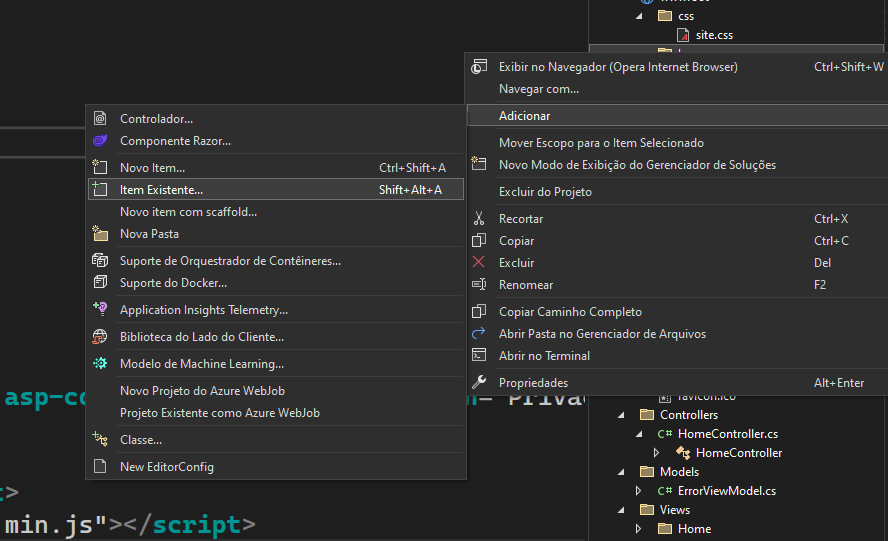
Passo 2

Crie uma para para armazenar as figuras do banner, indo no menu PROJETO -> NOVA PASTA

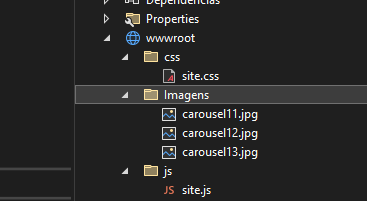


Chame a pasta de Imagens

Vamos adicionar imagens nessa pasta da seguinte maneira:



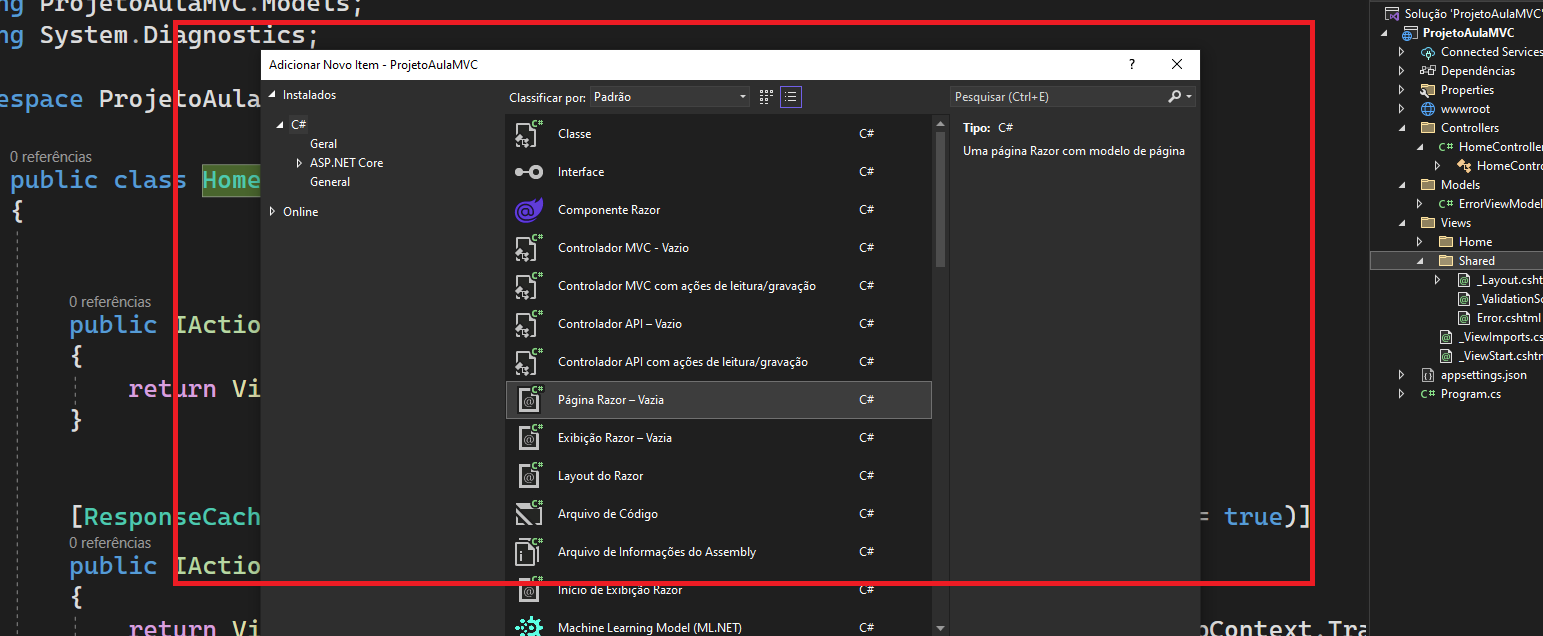
Sobre a pasta imagens clique com o botão direito do mouse e vá em adicionar -> Item Existente e ache as imagens que serão baixadas do Teans. Após isso adicione ao projeto.



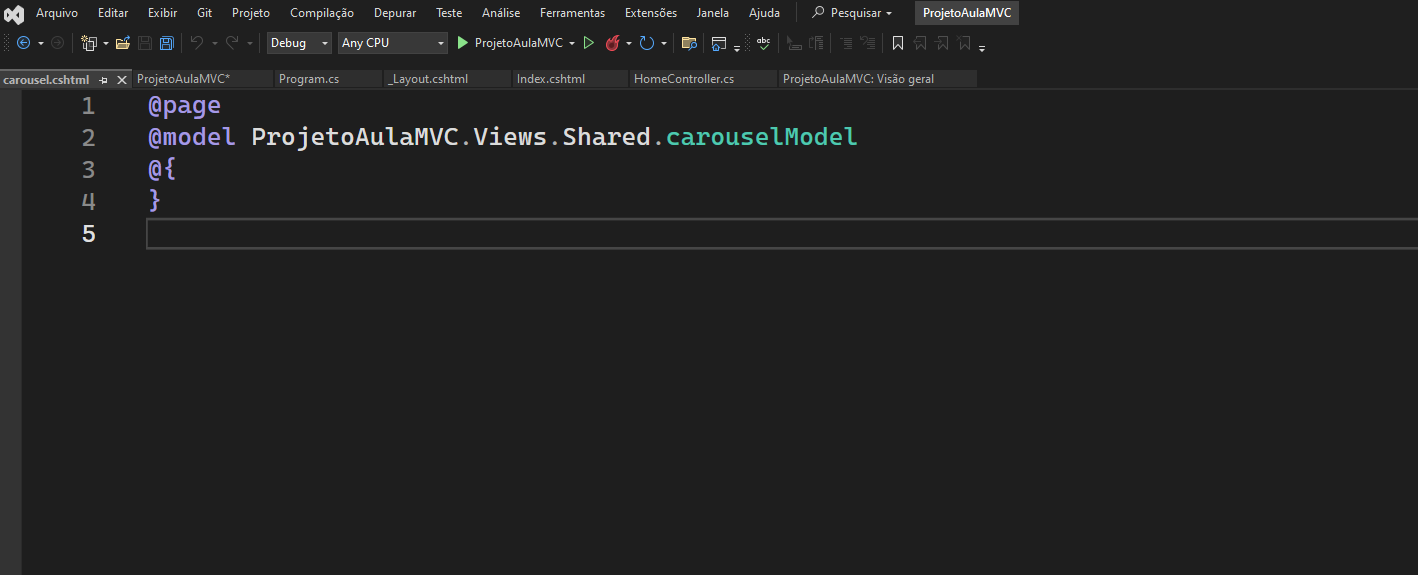
Passo 3

Crie uma view para o código do banner, está view será compartilhada para que outra página do projeto enxergue essa view e use-a. Portanto vamos cria-la como Shared.

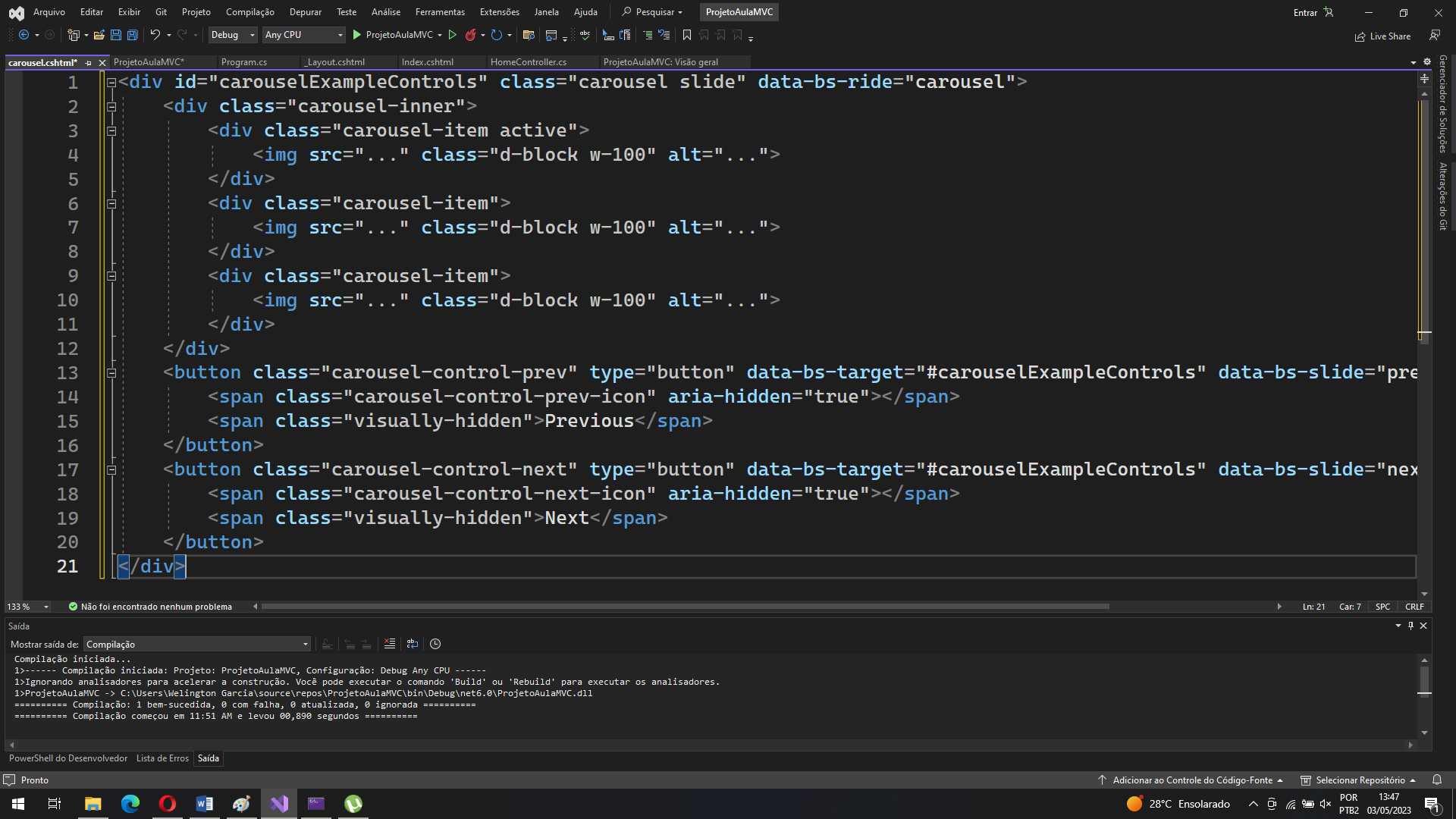
Na pasta shared botão direito do mouse -> adicionar -> componente razor



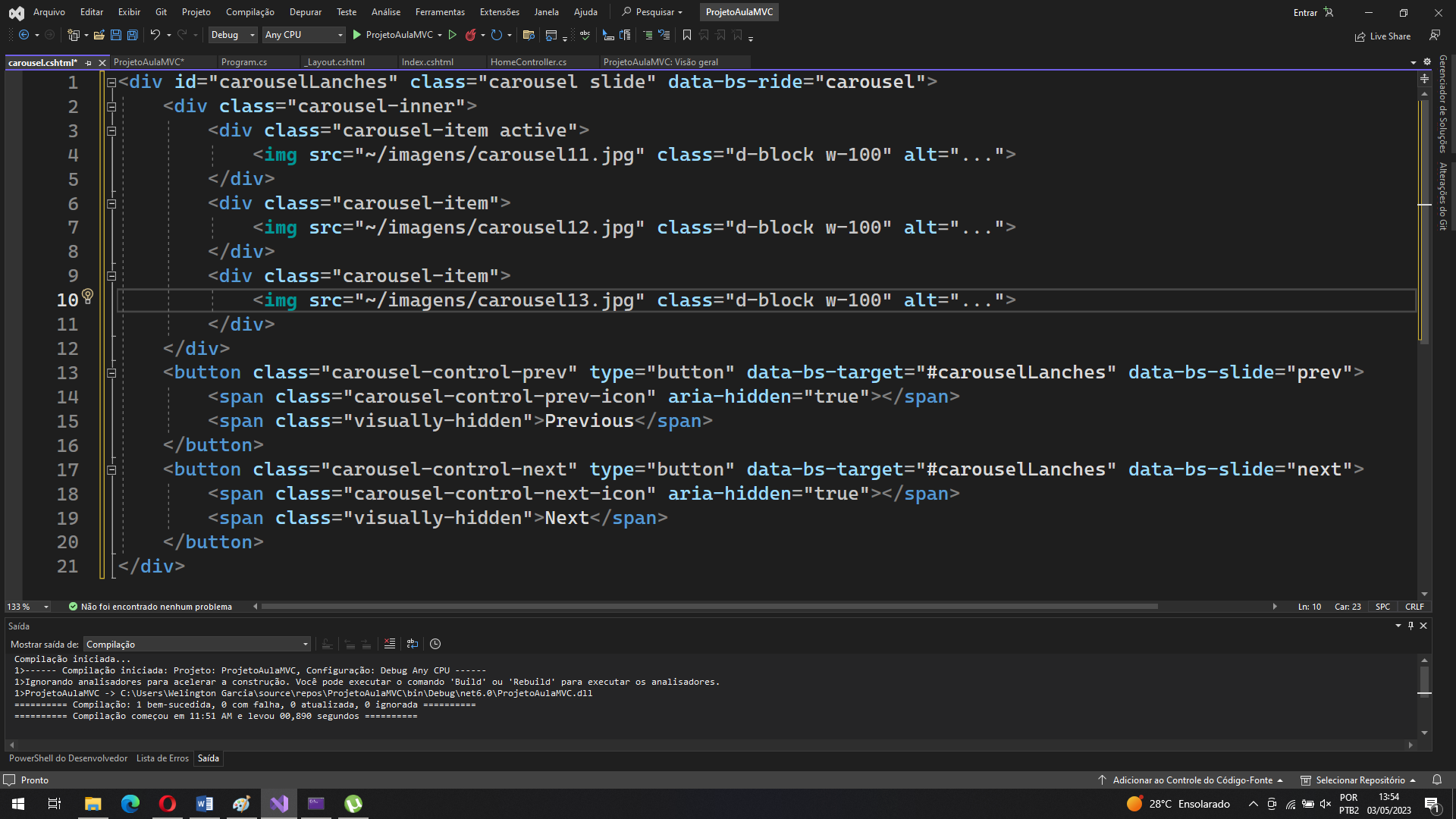
Nesta tela selecione página razor vazia e de o nome de carousel.cshtml



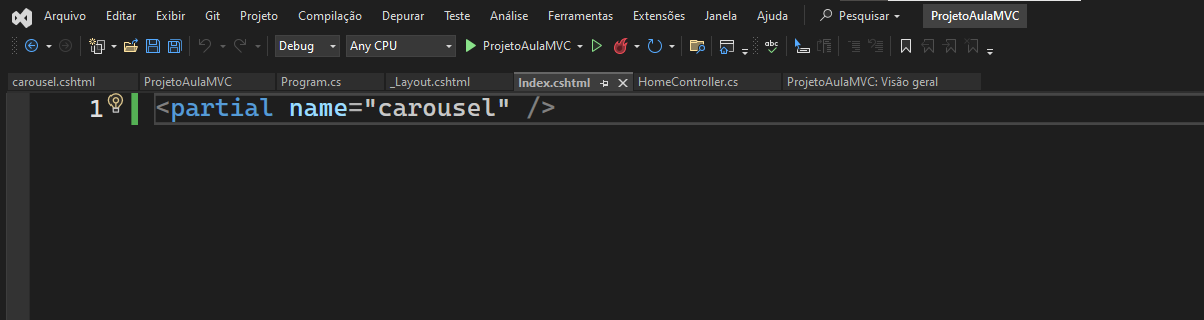
Apague o código da página e copie e cole o código do carousel nesta página.



Set o id do carousel para carouselLanches e altere os data-bs-target, depois altere o caminho das imagens para os arquivos de imagens baixados e anexados ao projeto como na imagem a seguir.



Pare esse carousel ser exibido vou até meu index.cshtml e adiciono o seguinte código:



Após isso execute o projeto.

**Modelo de domínio do site**

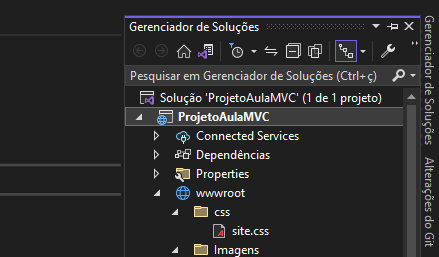
O modelo de domínio é a estrutura das minhas classes, para este projeto vamos ter como uma das principais funcionalidades vender lanches, sendo assim preciso criar uma classe para descrever o Lanche, está classe terá nome, preço, descrição curta, descrição longa, imagem, imagem miniatura, categoria e demais atributos desejados para a venda do lanche. Vamos dar o nome de Lanche.cs a está classe.

Para criar essa classe vamos usar uma ferramenta chamada ORM, está ferramenta da Microsoft é usada para mapeamento de classes em tabelas do banco de dados.

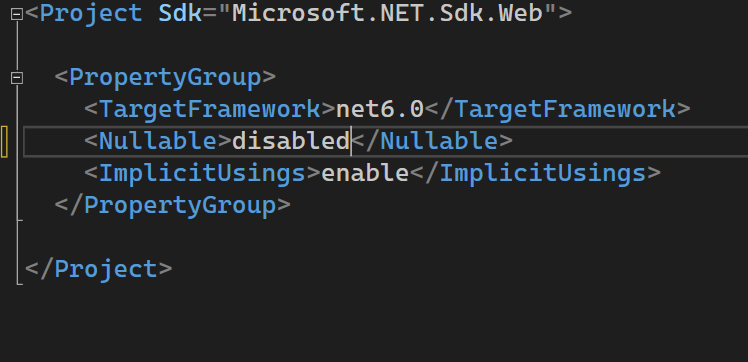
Vamos criar uma classe categorias que irá se relacionar com a classe lanches em um relacionamento 1:N.

**Ajustes no sistema**

Acesse as propriedades do projeto na opção abaixo.

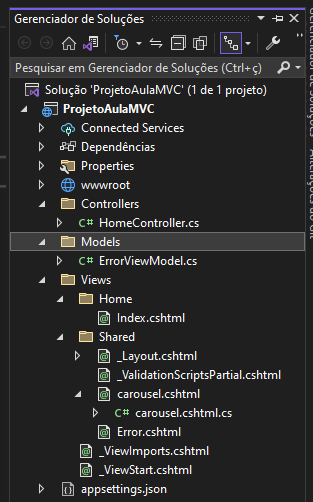
****

Altere as propriedades do projeto no campo Nullable para disable.

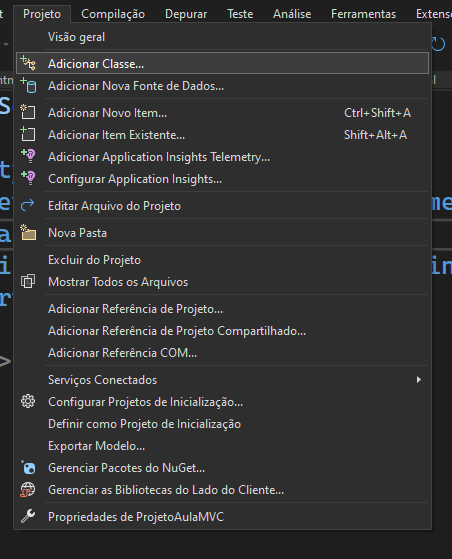


Isso irá desabilitar mensagens desnecessárias no projeto.

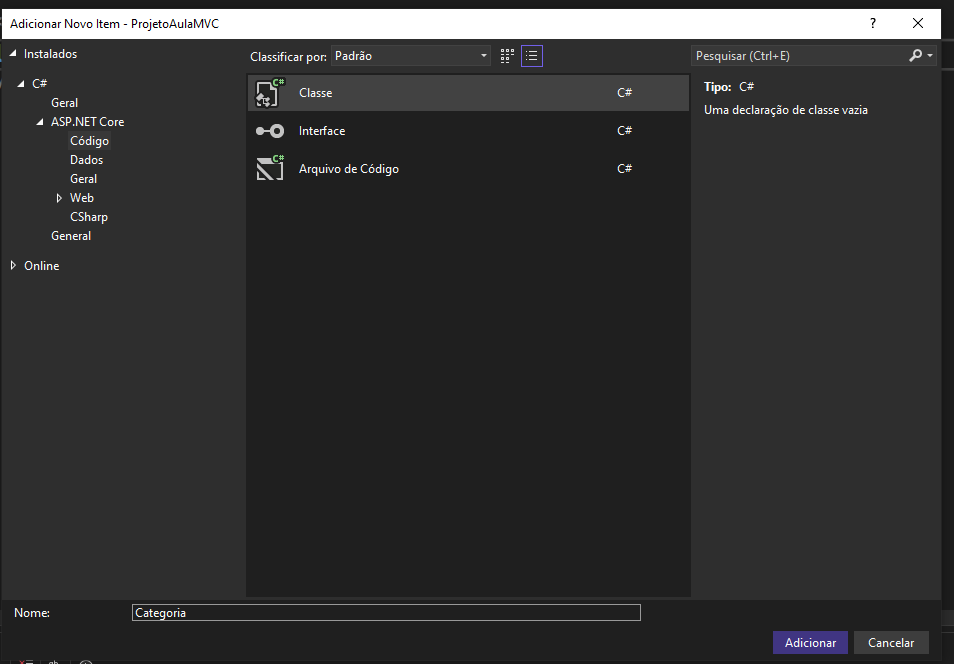
Após isso selecione a pasta models do projeto



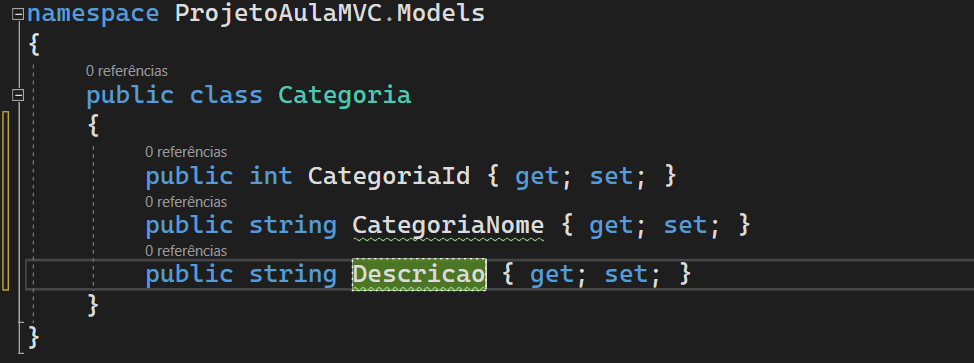
Após isso vá no menu Project e selecione adicionar classe



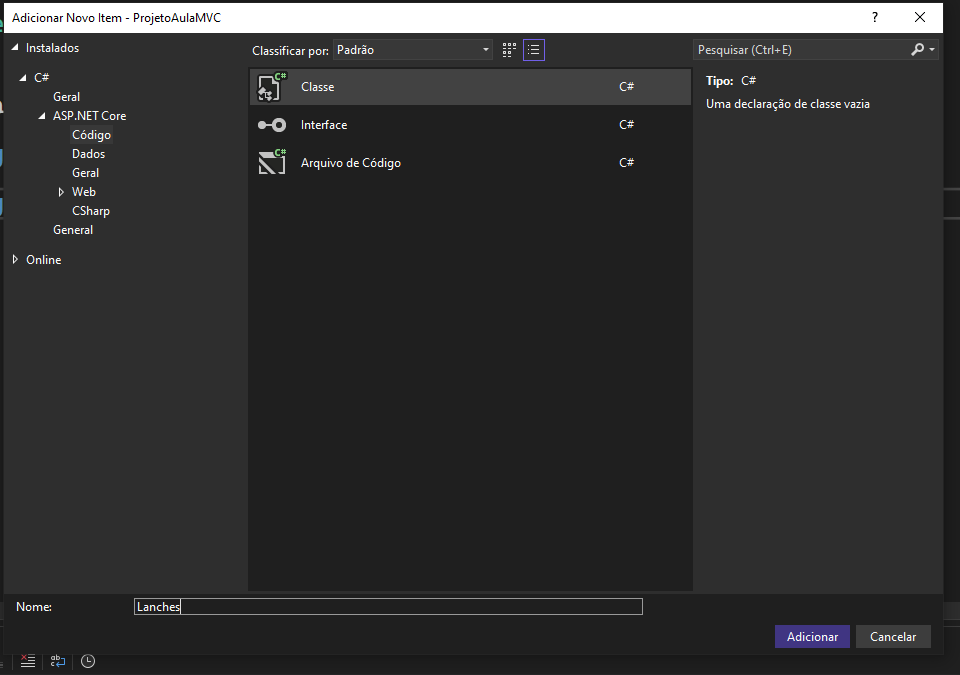
Selecione classe e de o nome de categoria, como na imagem que segue.



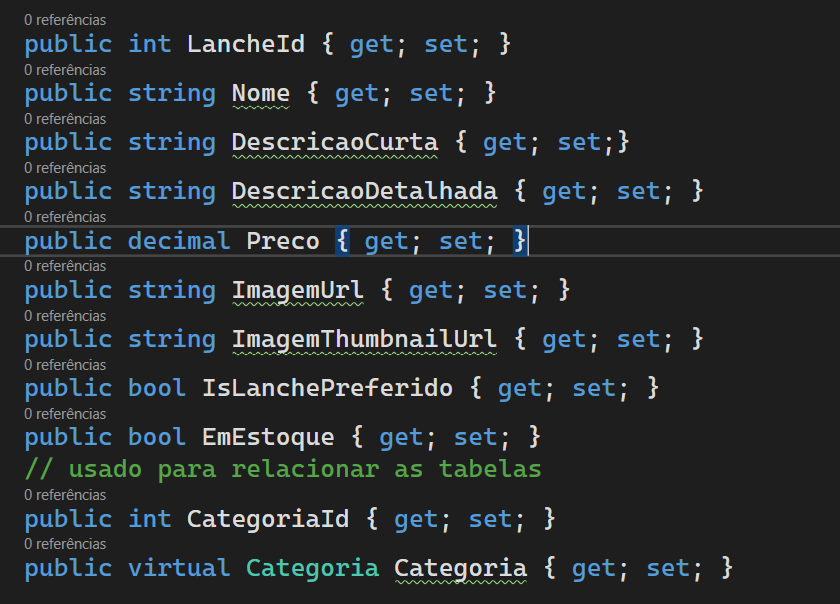
Agora vamos definir as propriedades da classe como na figura abaixo:



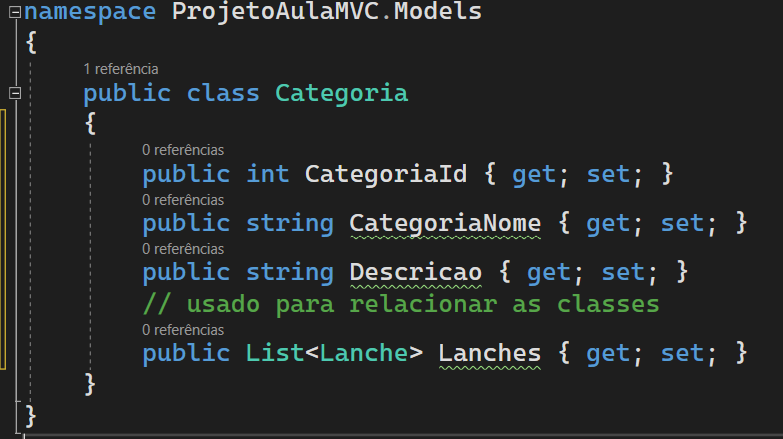
Agora vamos adicionar outra classe chamada Lanches



Preencha como abaixo



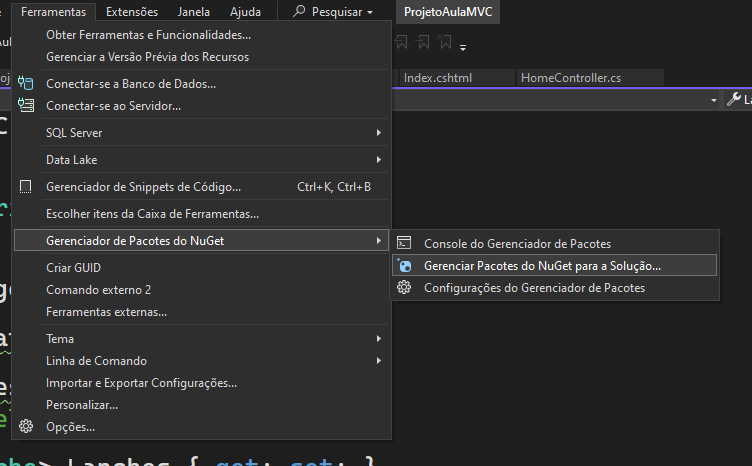
Na classe categoria devemos adicionar:



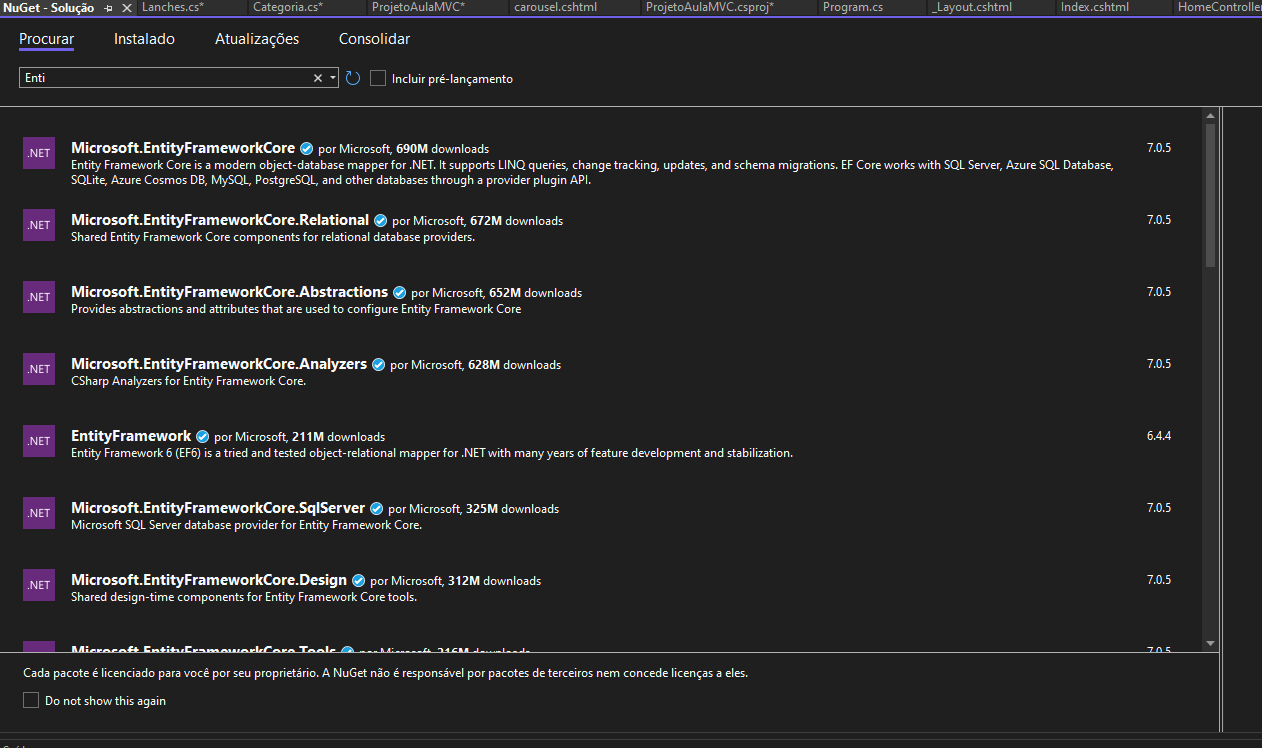
**O Entity Framework Core**

Eleva o nível de abstração eliminando a necessidade de criar o banco na unha.

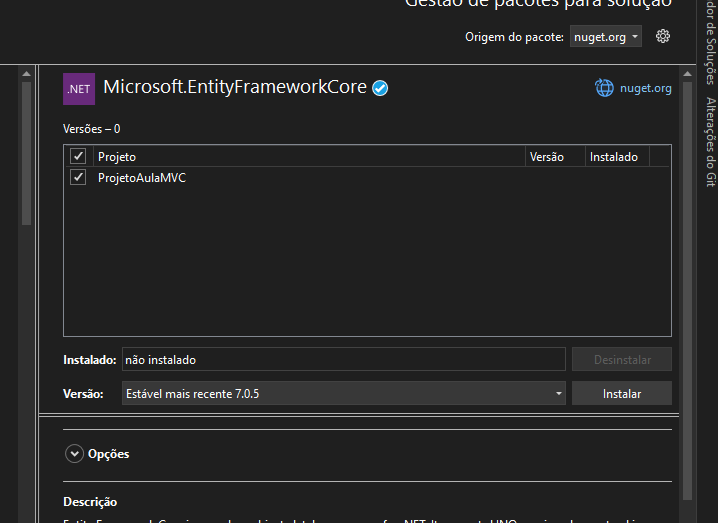
Como configurar acesse:



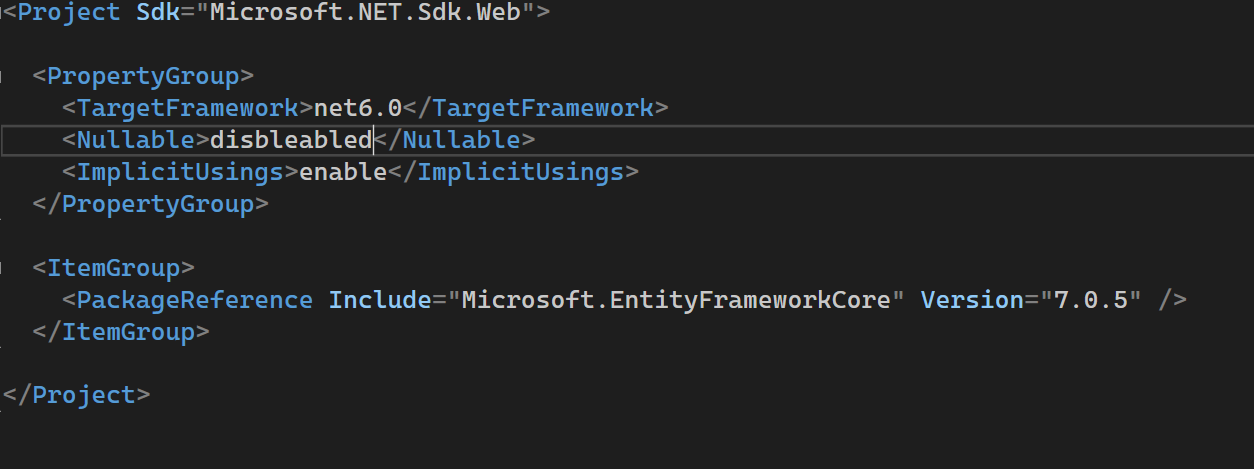
Após isso pesquise:



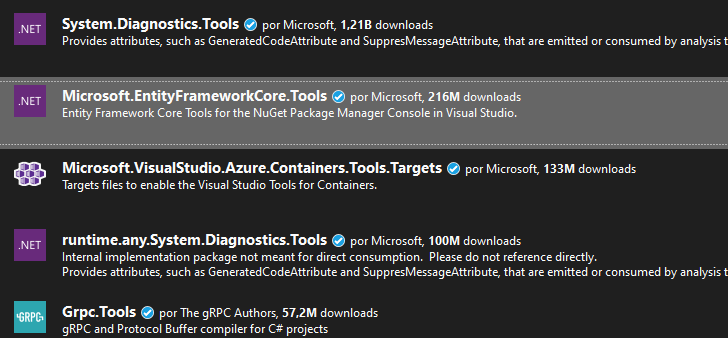
Selecione essas opções e clique em instalar



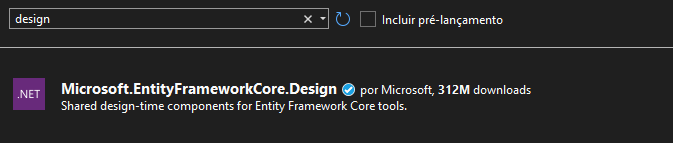
Note as propriedades do projeto



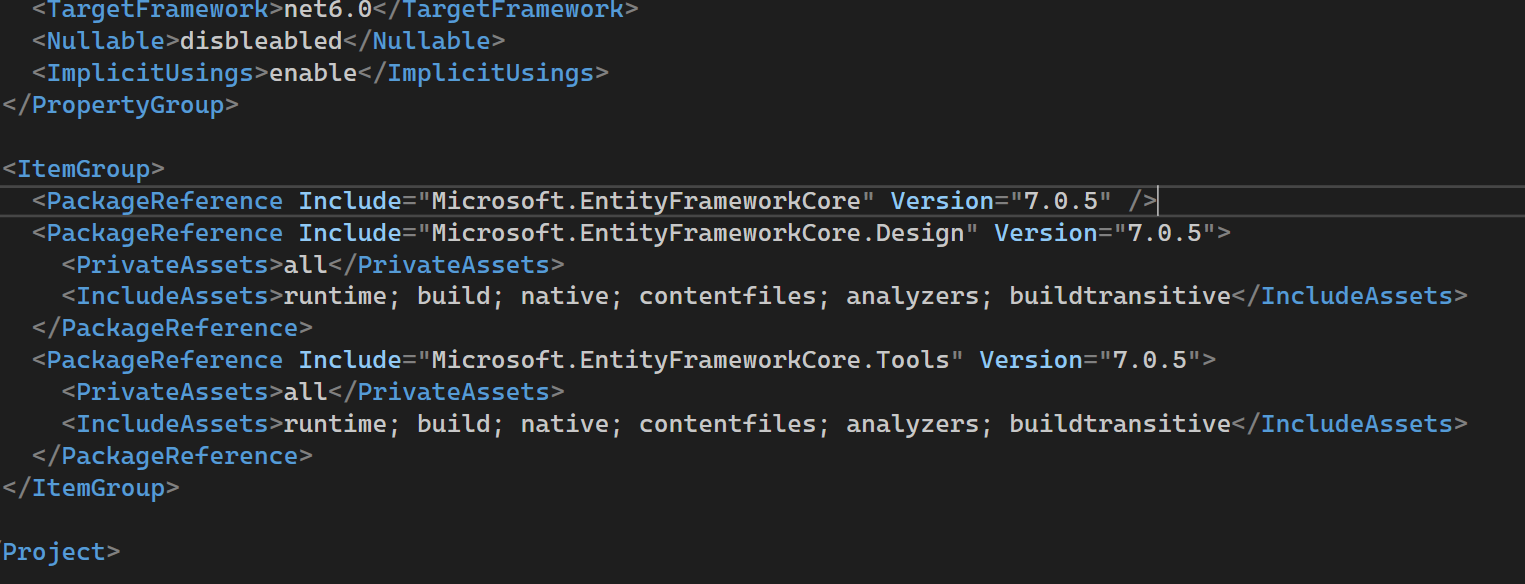
Agora instale o seguinte pacote



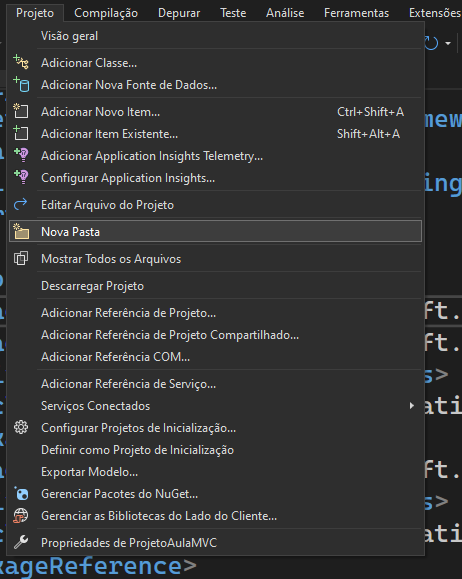
Instale também



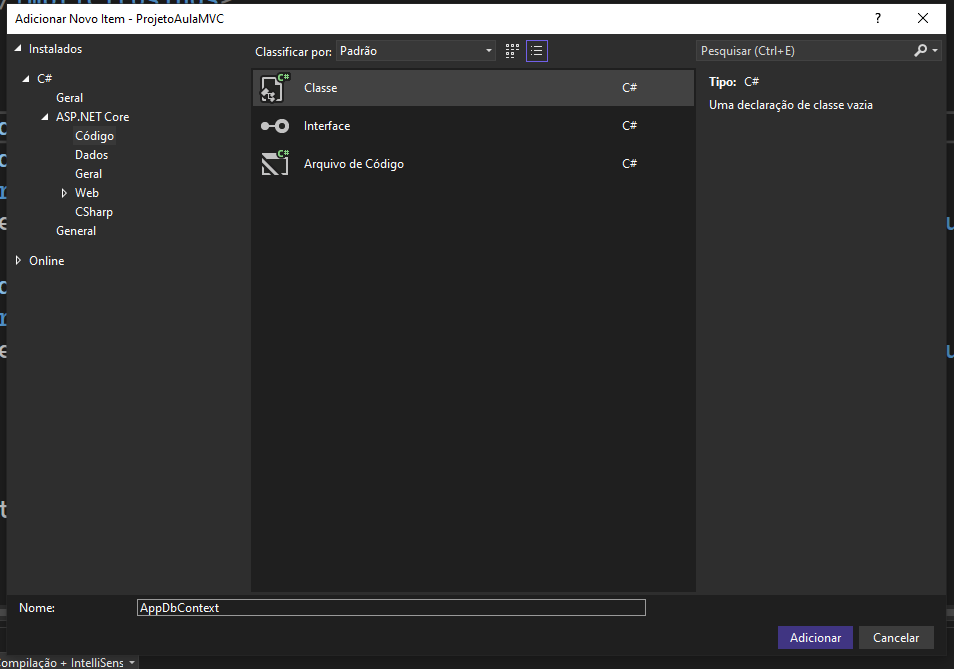
Após isso várias propriedades são adicionadas ao projeto



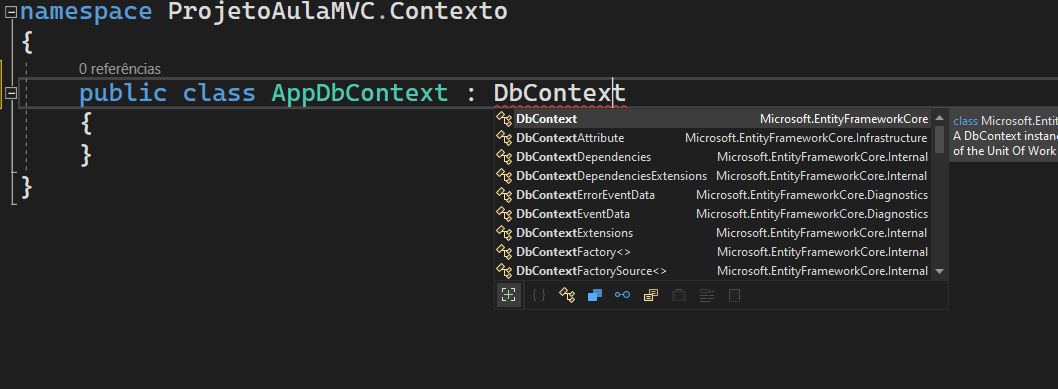
Crie uma nova pasta no projeto chamada Contexto



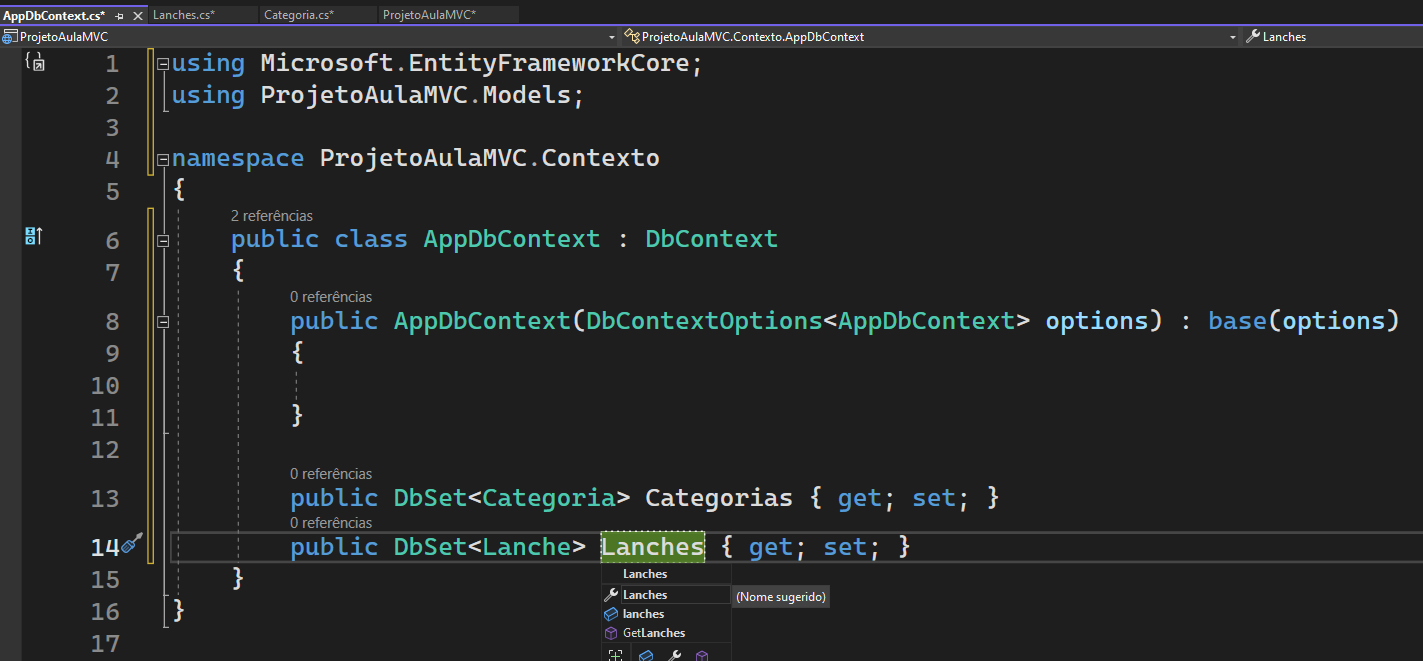
E adicione uma classe chamada AppDbContext



Estenda a classe como abaixo

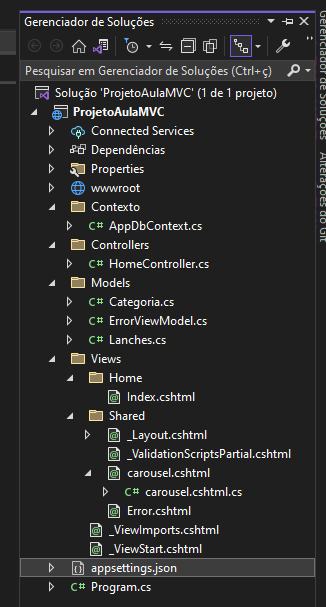


Adicione as seguintes linhas de códigos

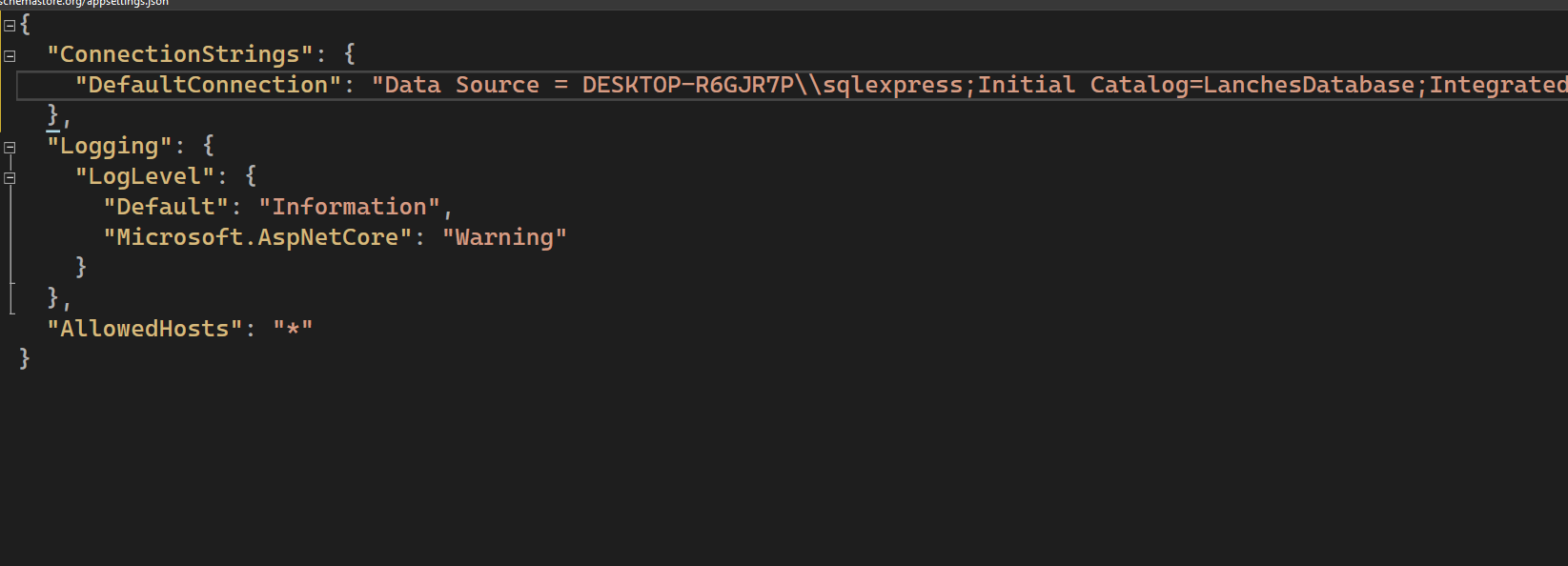


Agora vamos definir a string de conexão:

Vá até o arquivo appsettings.jason



Adicione o seguinte código



Copie e cole o código abaixo

"ConnectionStrings": {

"DefaultConnection": "Data Source = DESKTOP-R6GJR7P\\sqlexpress;Initial Catalog=LanchesDatabase;Integrated Security=True"

},

Agora vamos fazer algumas adições no programa. Vamos digitar a classe Startup.CS

**Conexão com o banco de dados**