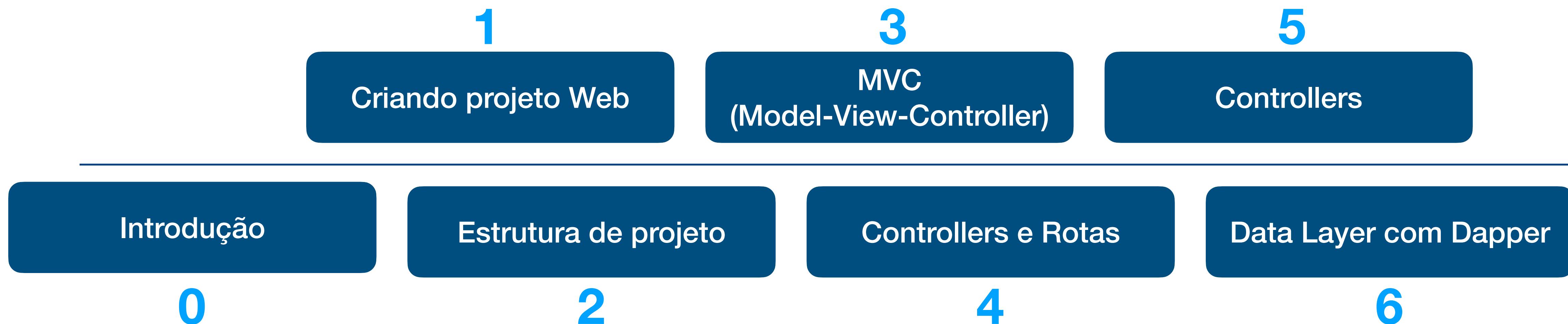


Treinamento C#/.Net

Básico/Intermediário

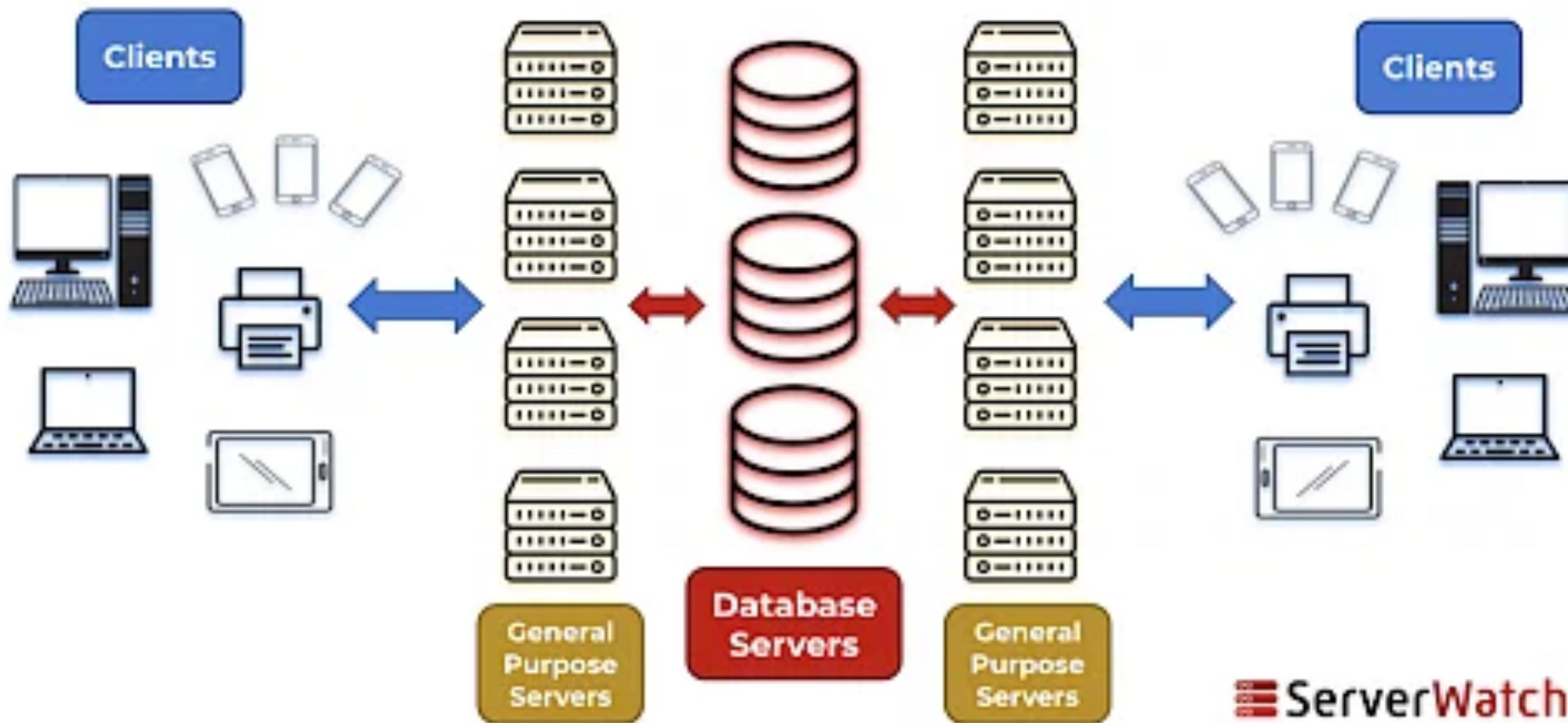
Lucas Baggio Figueira

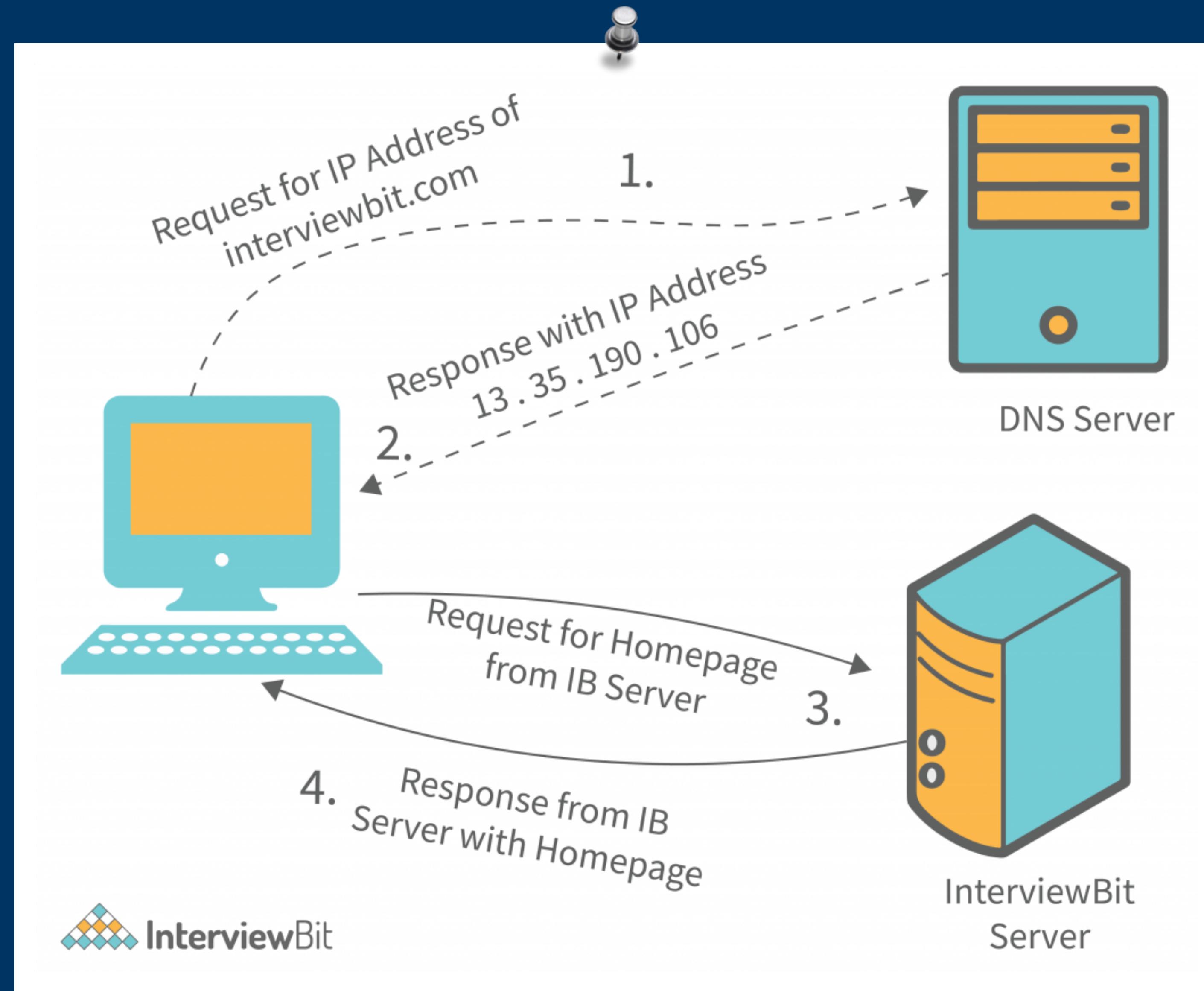
Sumário

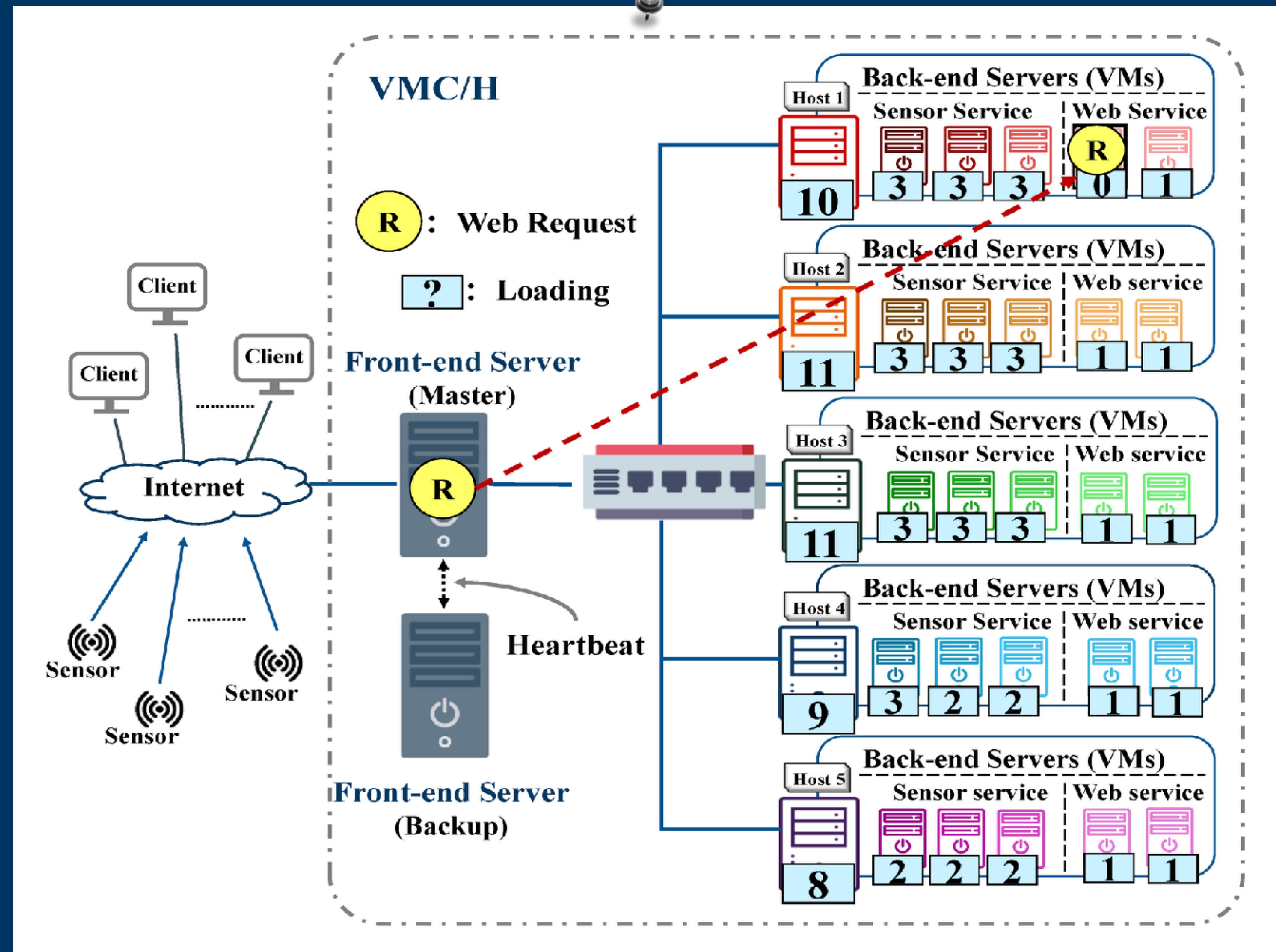


Introdução

The Client-Server Model



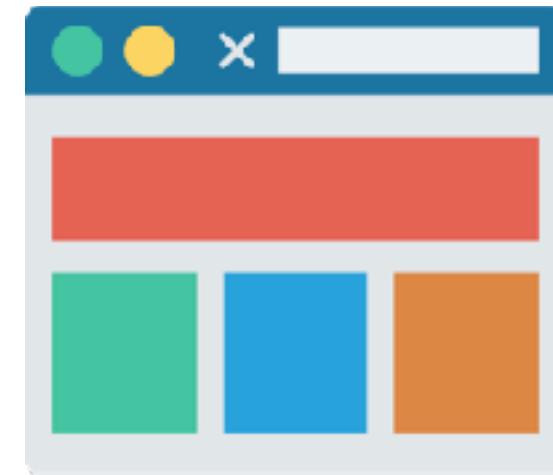




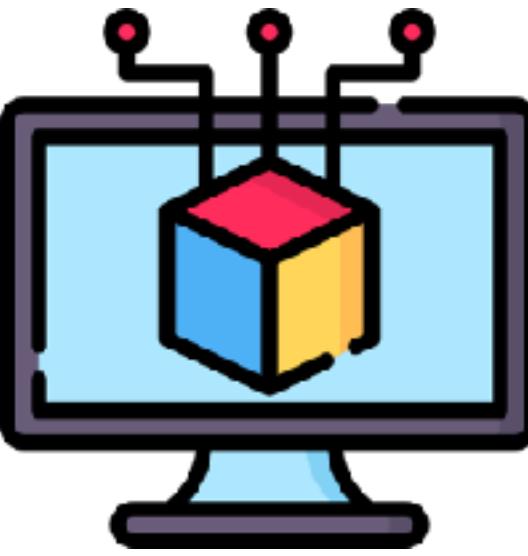
Aplicação Web

Conjunto de recursos computacionais que são executados sobre a estrutura da internet.

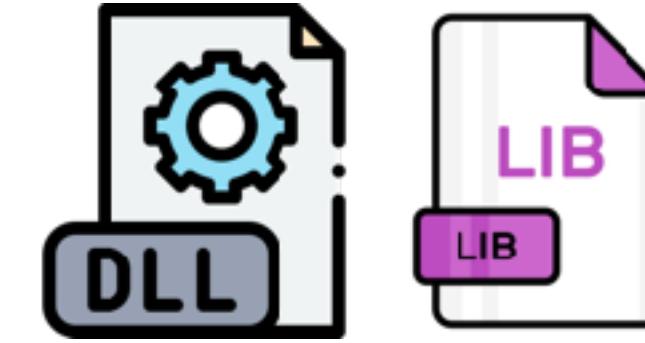
Conjunto de recursos computacionais



Páginas Web



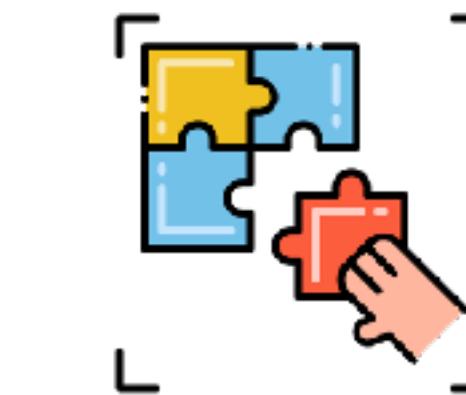
Programas



Bibliotecas



Banco de Dados

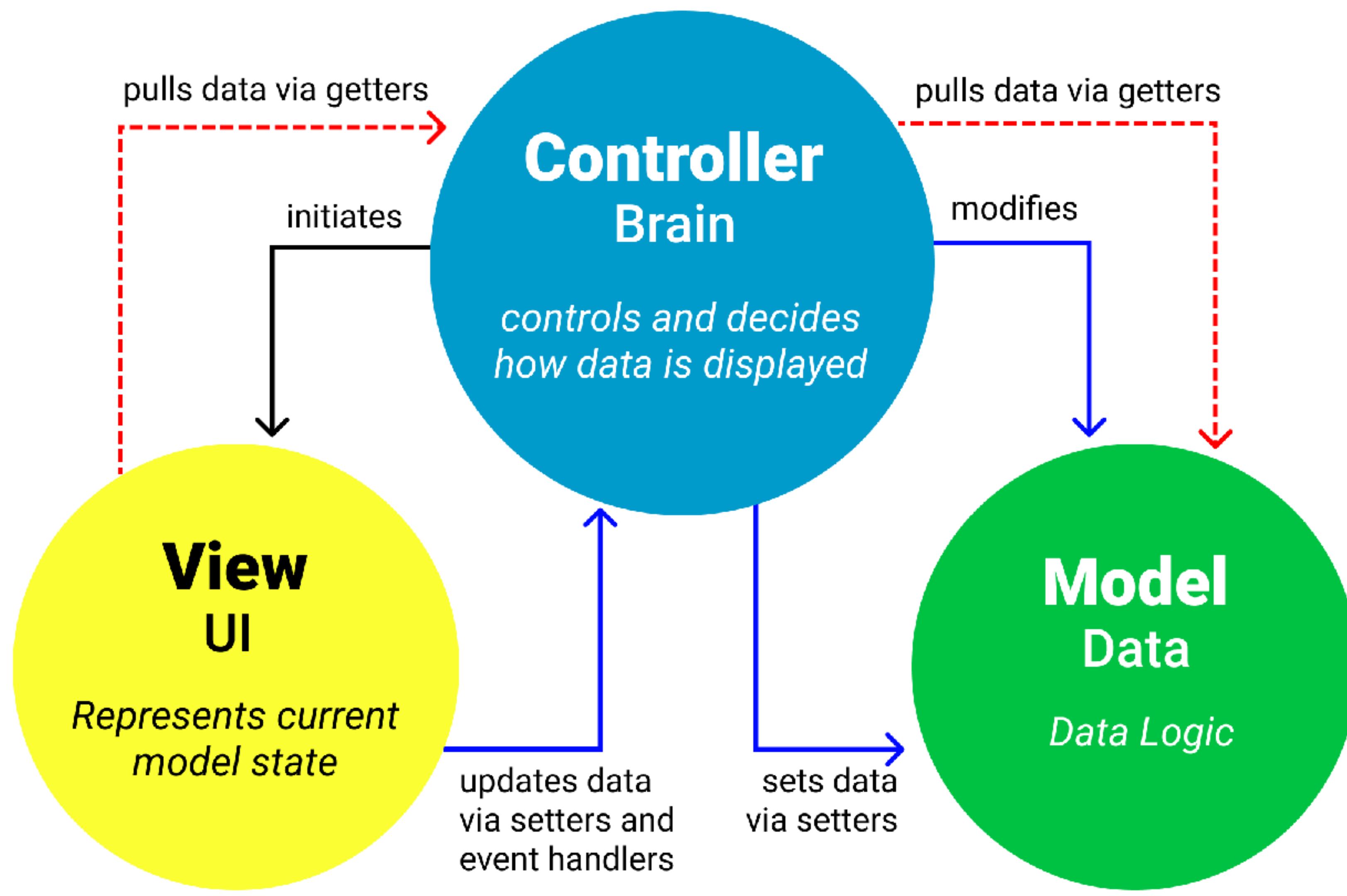


Frameworks



Sistemas Operacionais

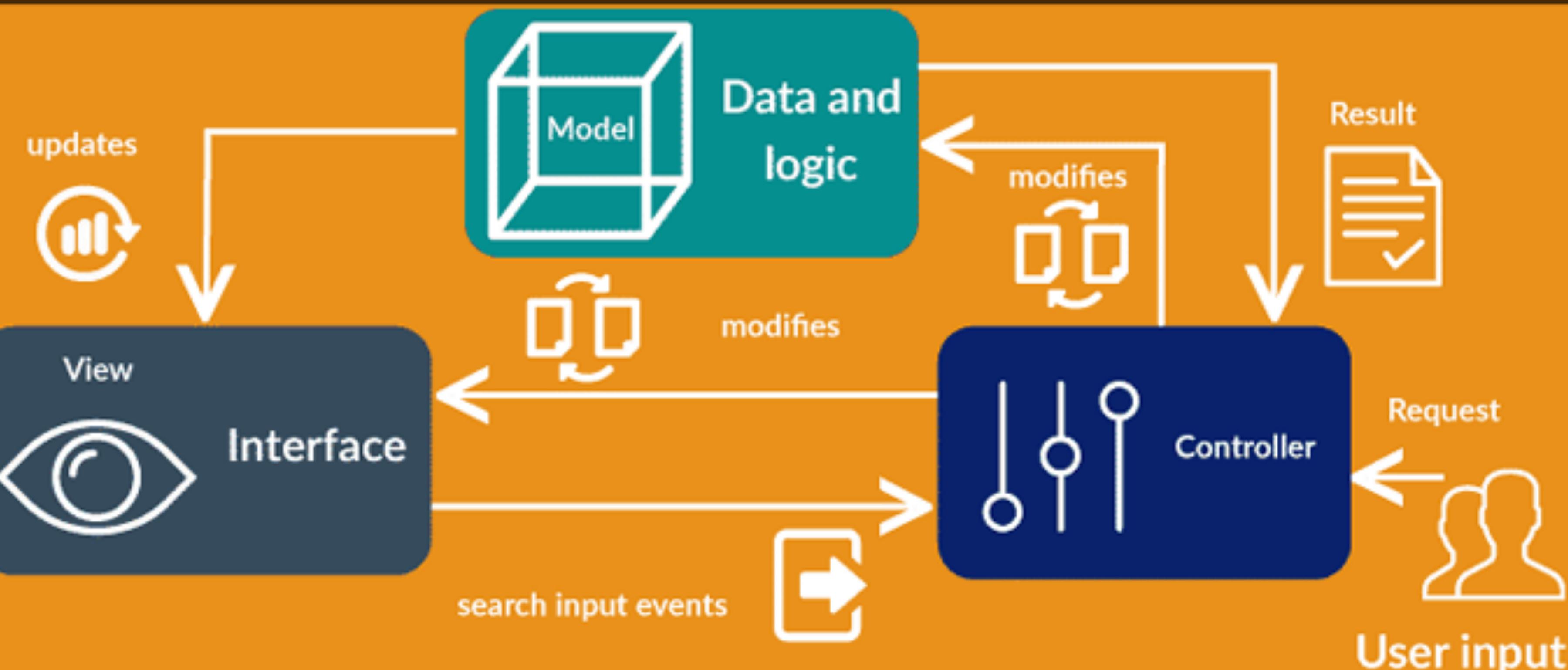
MVC Architecture Pattern

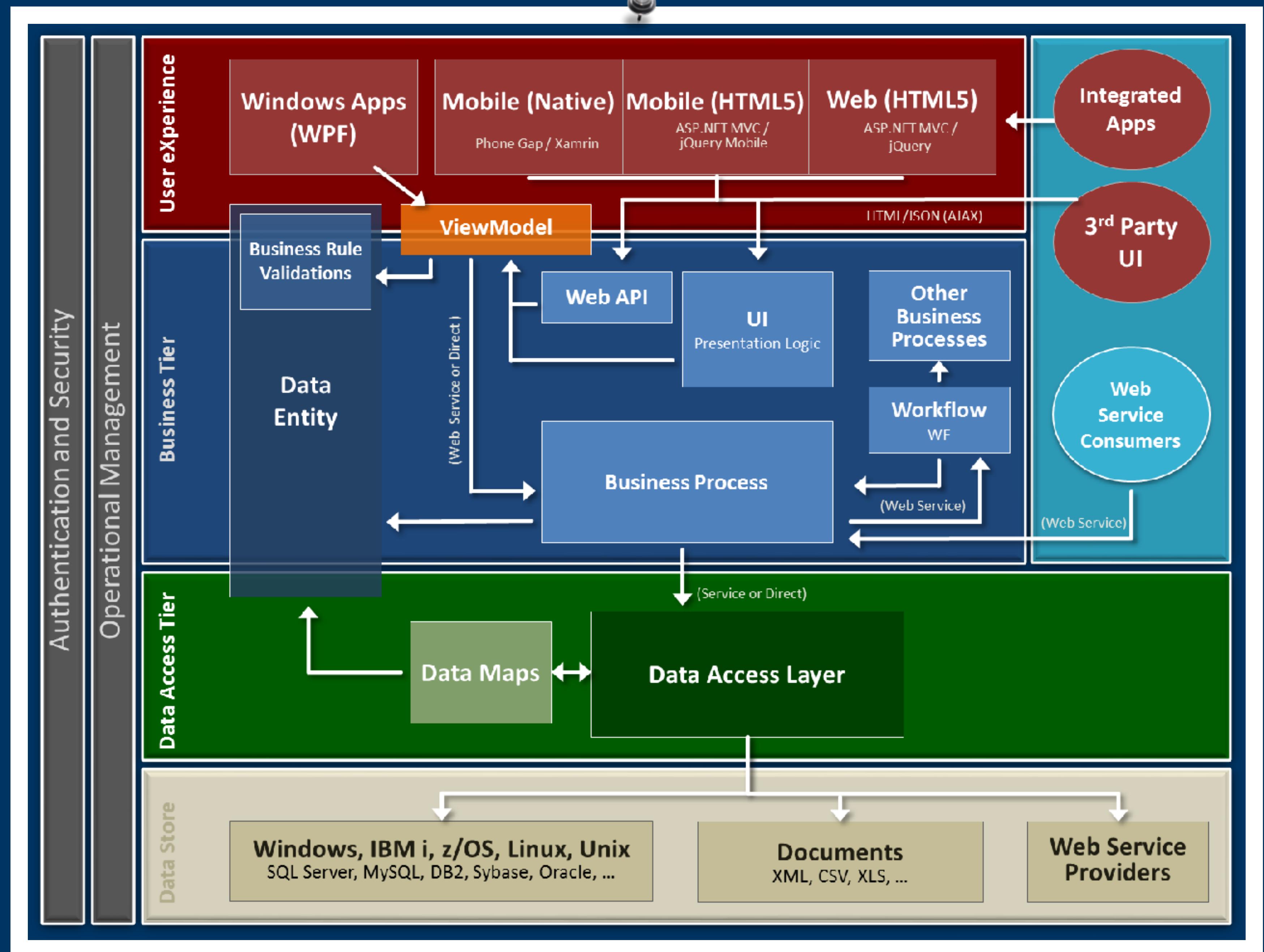


Padrão MVC

O padrão de design model-view-controller especifica que um aplicativo consiste em um **modelo de dados, informações de apresentação e controle e orquestração do fluxo de informação**.

ASP.NET MVC FRAMEWORK





Camadas

View

User Experience

Controller

Business Logic

Model

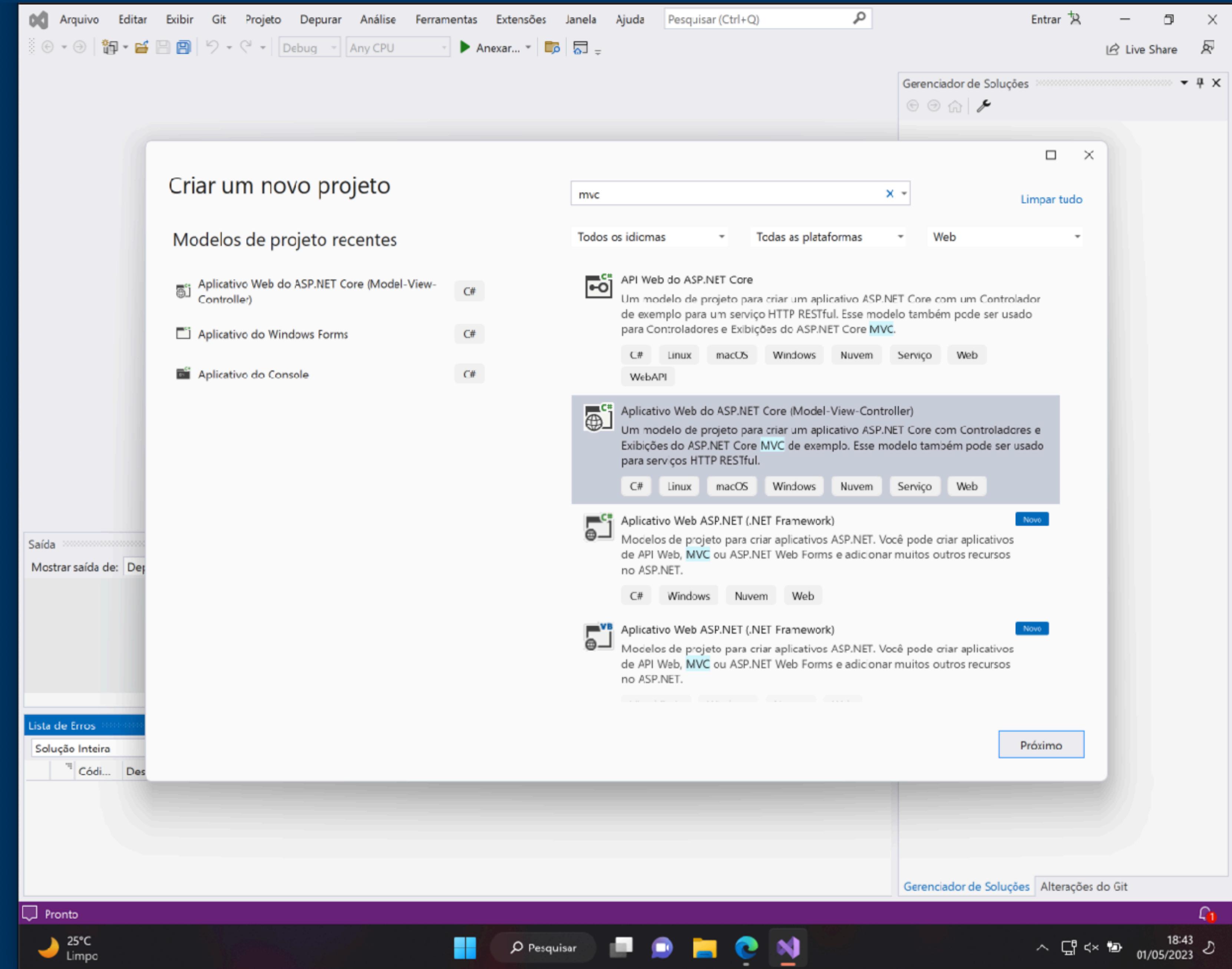
Data Access

Camada voltada para interagir com o usuário.

Camada voltada para todas as regras de negócio do domínio.

Camada voltada para gerenciar o armazenamento de informações

Criando um projeto
ASP.Net Core MVC



mvc

Limpar tudo

Todos os idiomas

Todas as plataformas

Web

C# API Web do ASP.NET Core
Um modelo de projeto para criar um aplicativo ASP.NET Core com um Controlador de exemplo para um serviço HTTP RESTful. Esse modelo também pode ser usado para Controladores e Exibições do ASP.NET Core MVC.

C# Linux macOS Windows Nuvem Serviço Web

C# WebAPI

Aplicativo Web do ASP.NET Core (Model-View-Controller)
Um modelo de projeto para criar um aplicativo ASP.NET Core com Controladores e Exibições do ASP.NET Core MVC de exemplo. Esse modelo também pode ser usado para serviços HTTP RESTful.

C# Linux macOS Windows Nuvem Serviço Web

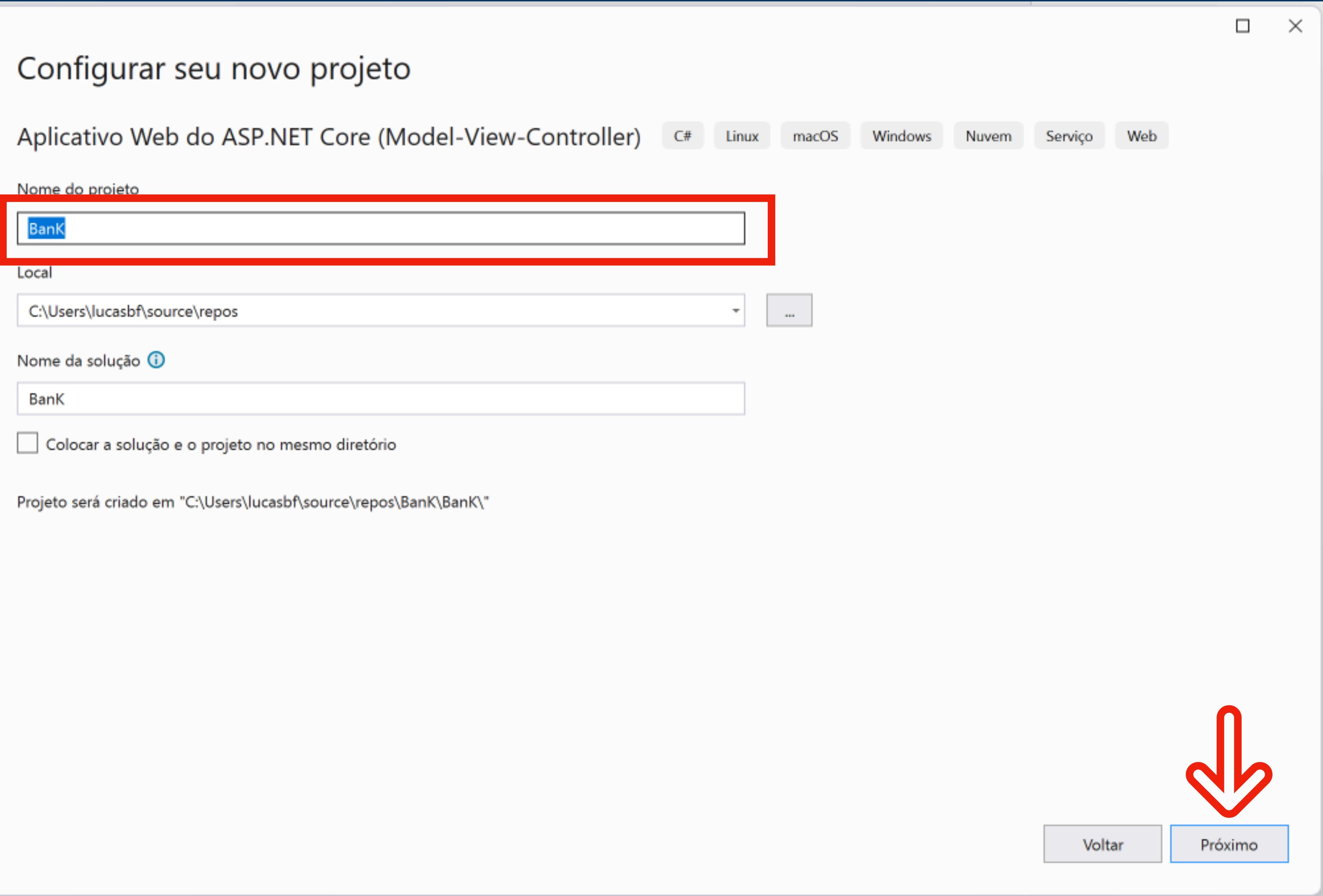
Aplicativo Web ASP.NET (.NET Framework)
Modelos de projeto para criar aplicativos ASP.NET. Você pode criar aplicativos de API Web, MVC ou ASP.NET Web Forms e adicionar muitos outros recursos no ASP.NET.

C# Windows Nuvem Web

Aplicativo Web ASP.NET (.NET Framework)
Modelos de projeto para criar aplicativos ASP.NET. Você pode criar aplicativos de API Web, MVC ou ASP.NET Web Forms e adicionar muitos outros recursos no ASP.NET.

Novo

Próximo



Informações adicionais

Aplicativo Web do ASP.NET Core (Model-View-Controller)

C#

Linux

macOS

Windows

Nuvem

Serviço

Web

Estrutura 

.NET 6.0 (Suporte de Longo Prazo)

Tipo de autenticação 

Nenhum

Configurar para HTTPS 

Habilitar o Docker 

Sistema Operacional do Docker 

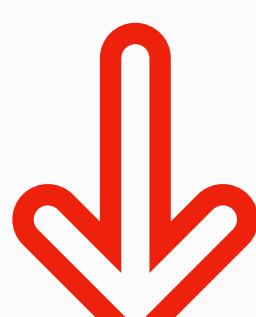
Linux

Não use instruções de nível superior 



Voltar

Criar



Arquivo Editar Exibir Git Projeto Compilação Depurar Teste Análise Ferramentas Extensões Janela Ajuda Pesquisar (Ctrl+Q) BanK - Live Share

BanK: Visão geral X

Visão geral

Serviços Conectados

Publicar

ASP.NET Core

Saiba mais sobre a plataforma .NET, crie seu primeiro aplicativo e estenda-o para a nuvem.

{} Compilar Seu Aplicativo

Conectar ao Azure

Saiba seu IDE

Pesquisar documentos, amostras e tutoriais

Publicar seu site no Azure

Ver nosso guia de produtividade

Introdução ao ASP.NET no Azure

Escrever código mais rápido

Arquitetura do aplicativo .NET

Saída

Mostrar saída de: Dependências de Serviço

Lista de Erros

Solução Inteira 0 Erros 0 Avisos 0 Mensagens Compilação + IntelliSense Pesquisar na Lista de Erros

Cód... Descrição Projeto Arquivo Li... Estado de Supressão

C:\Users\lucasbf\source\repos\BanK\BanK\BanK.csproj restaurado (em 1,16 sec). Adicionar ao Controle do Código-Fonte Selecionar Repositório 1 25°C Limpo Pesquisar 18:51 01/05/2023

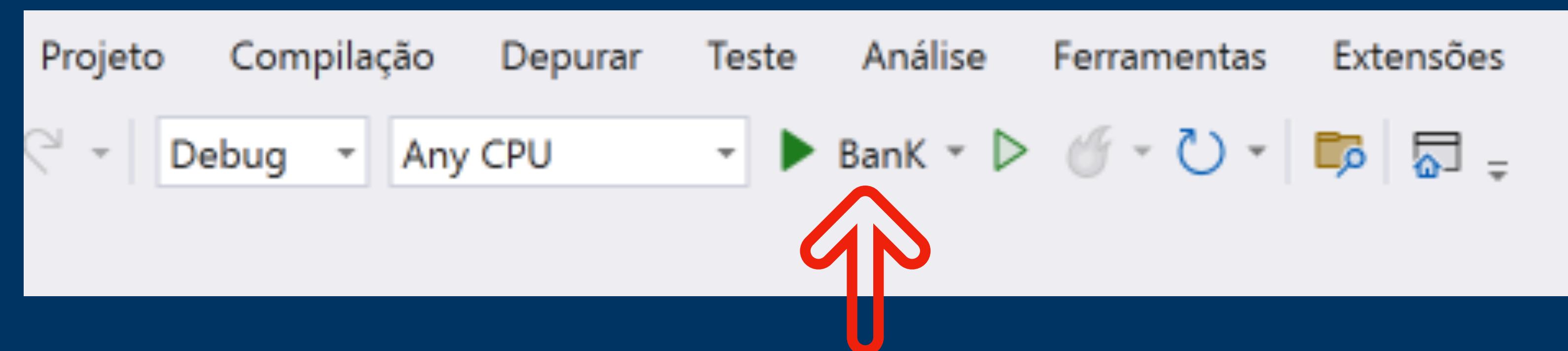
Gerenciador de Soluções Alterações do Git

Solução 'BanK' (1 de 1 projeto)

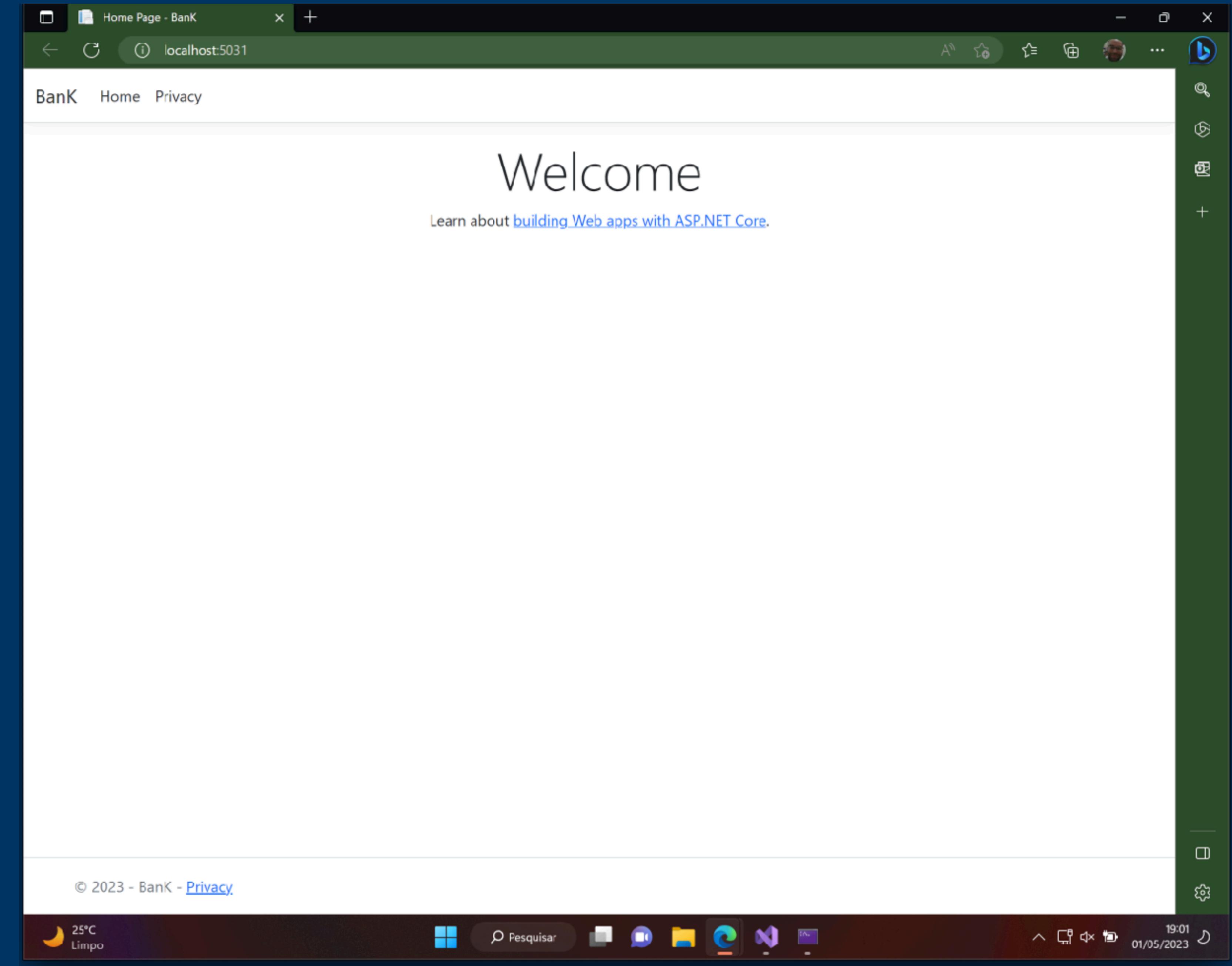
BanK

- Connected Services
- Dependências
- Properties
- wwwroot
- Controllers
- Models
- Views
- appsettings.json
- Program.cs

Rodar para Testar



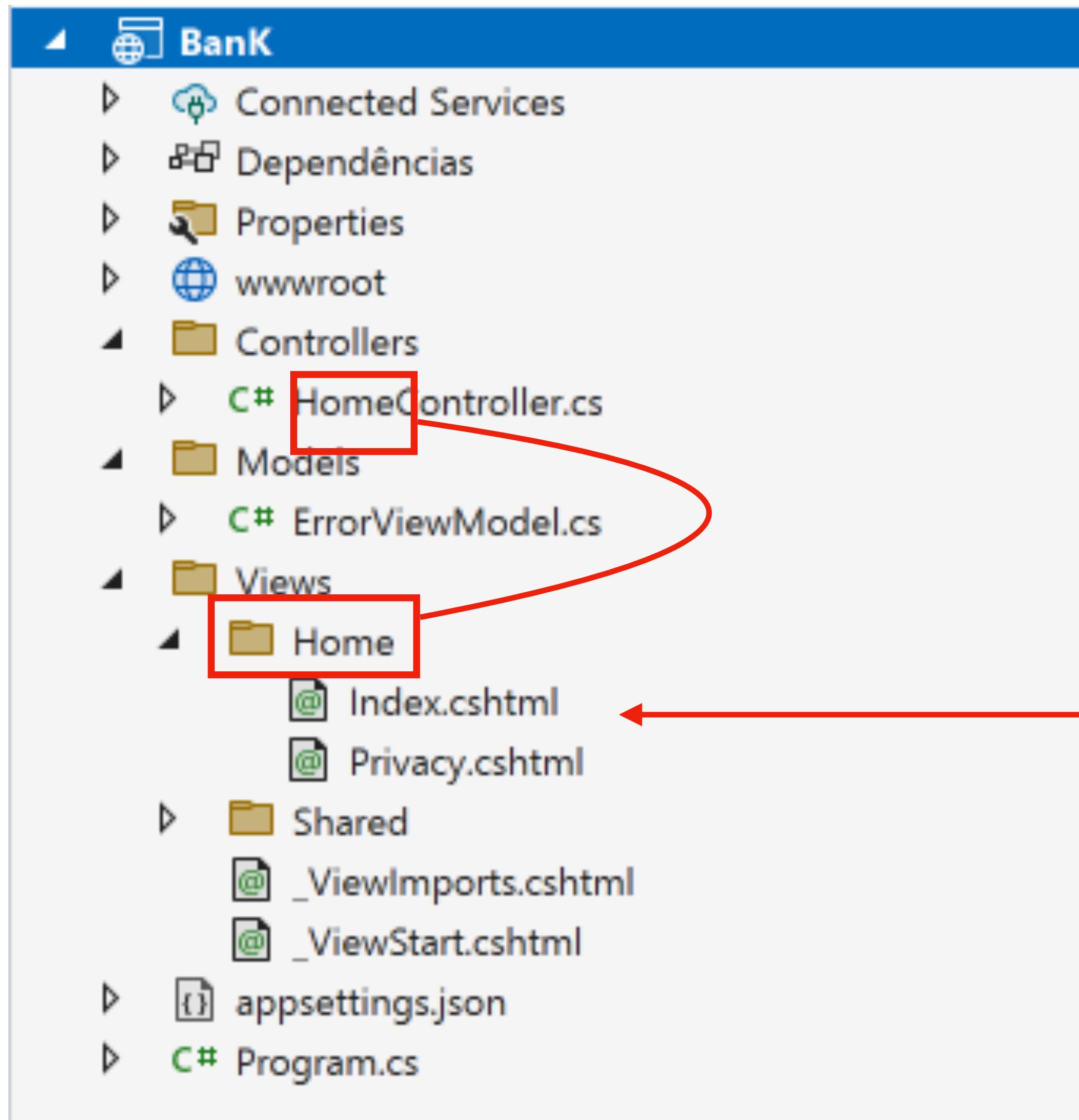
Conferindo



Estrutura do Projeto

Estrutura

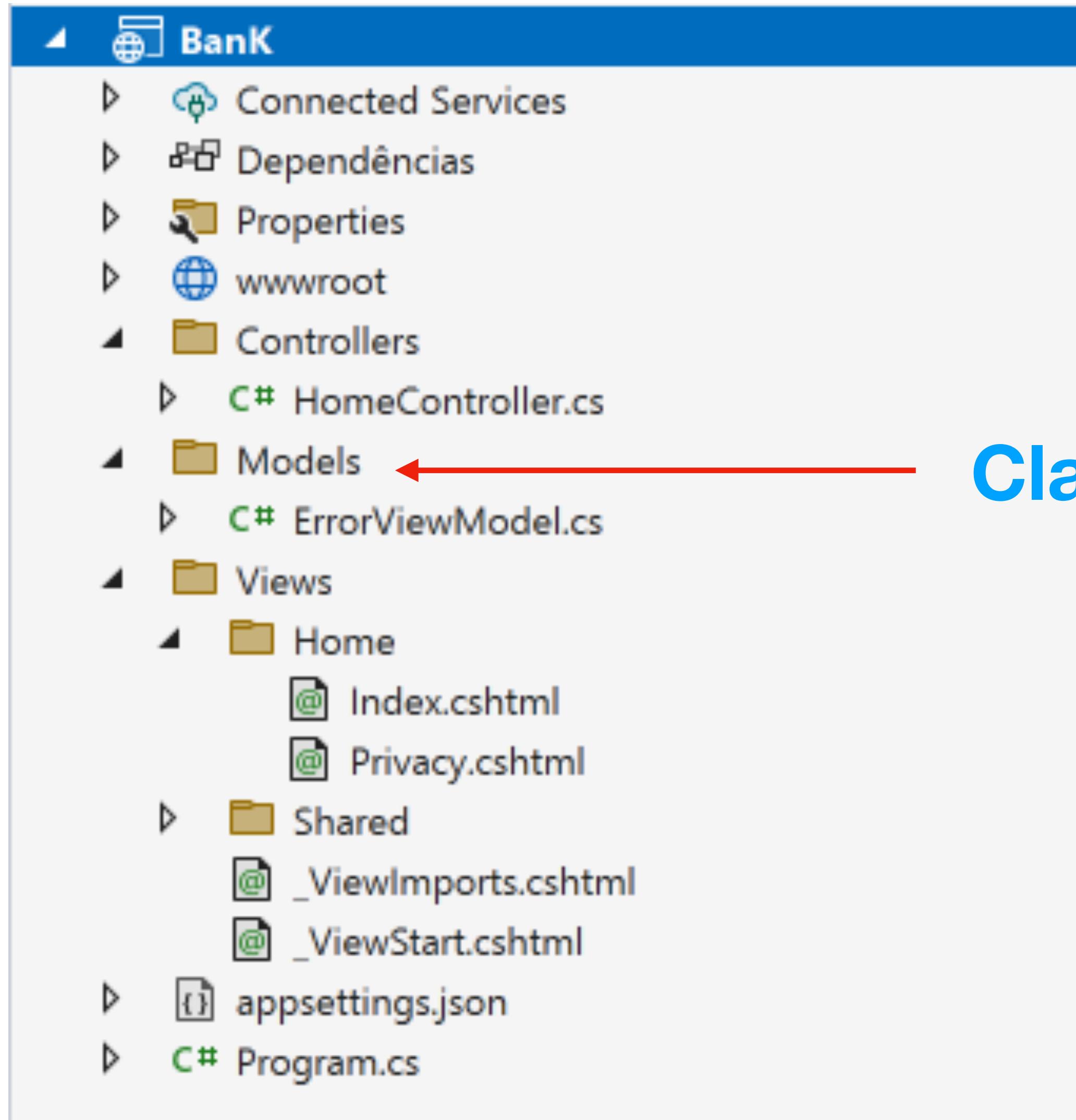
Aplicação Web ASP.Net MVC



Views (Exibições) referentes ao controlador em questão

Estrutura

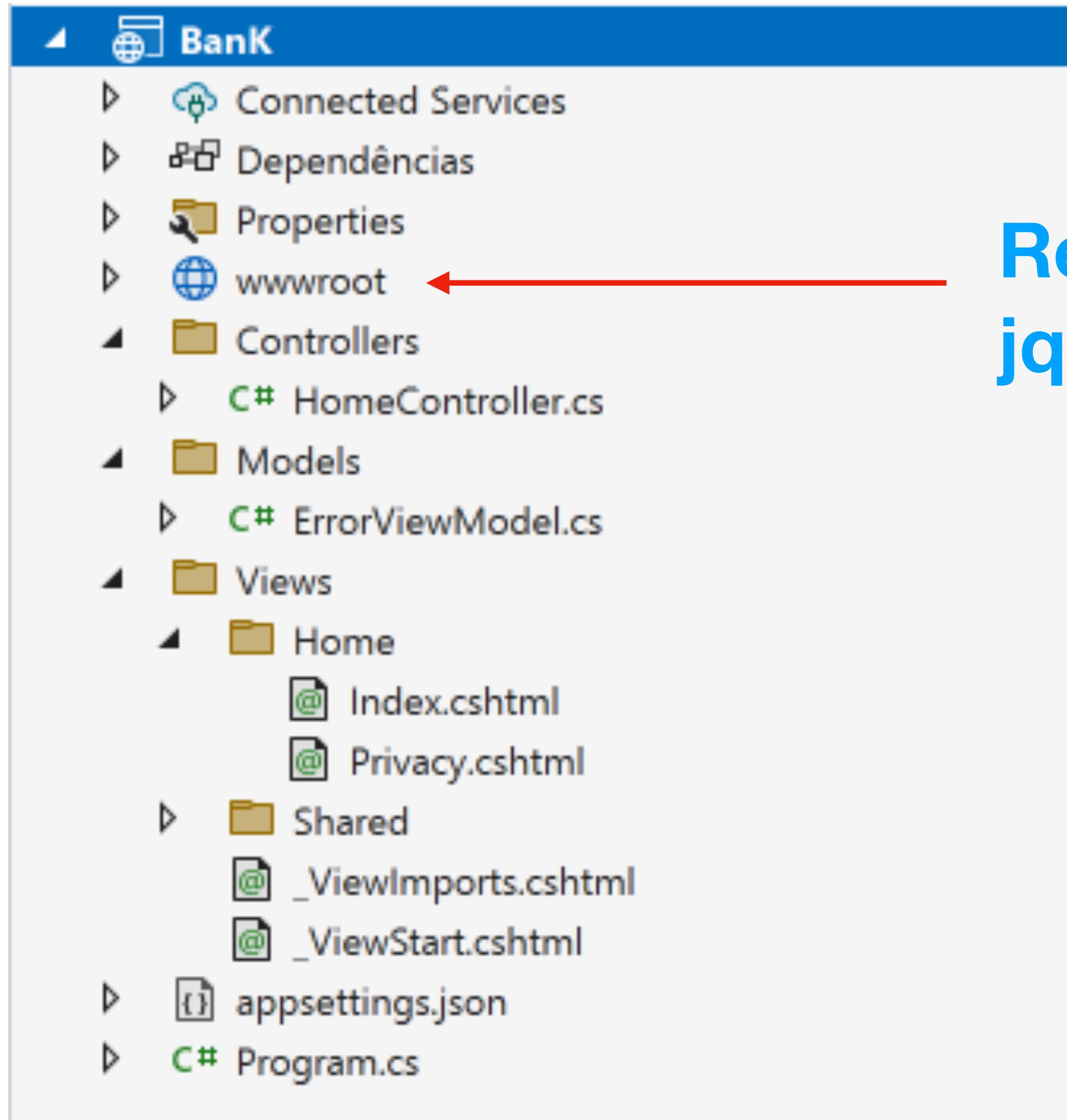
Aplicação Web ASP.Net MVC



Classes de Modelo de Dados

Estrutura

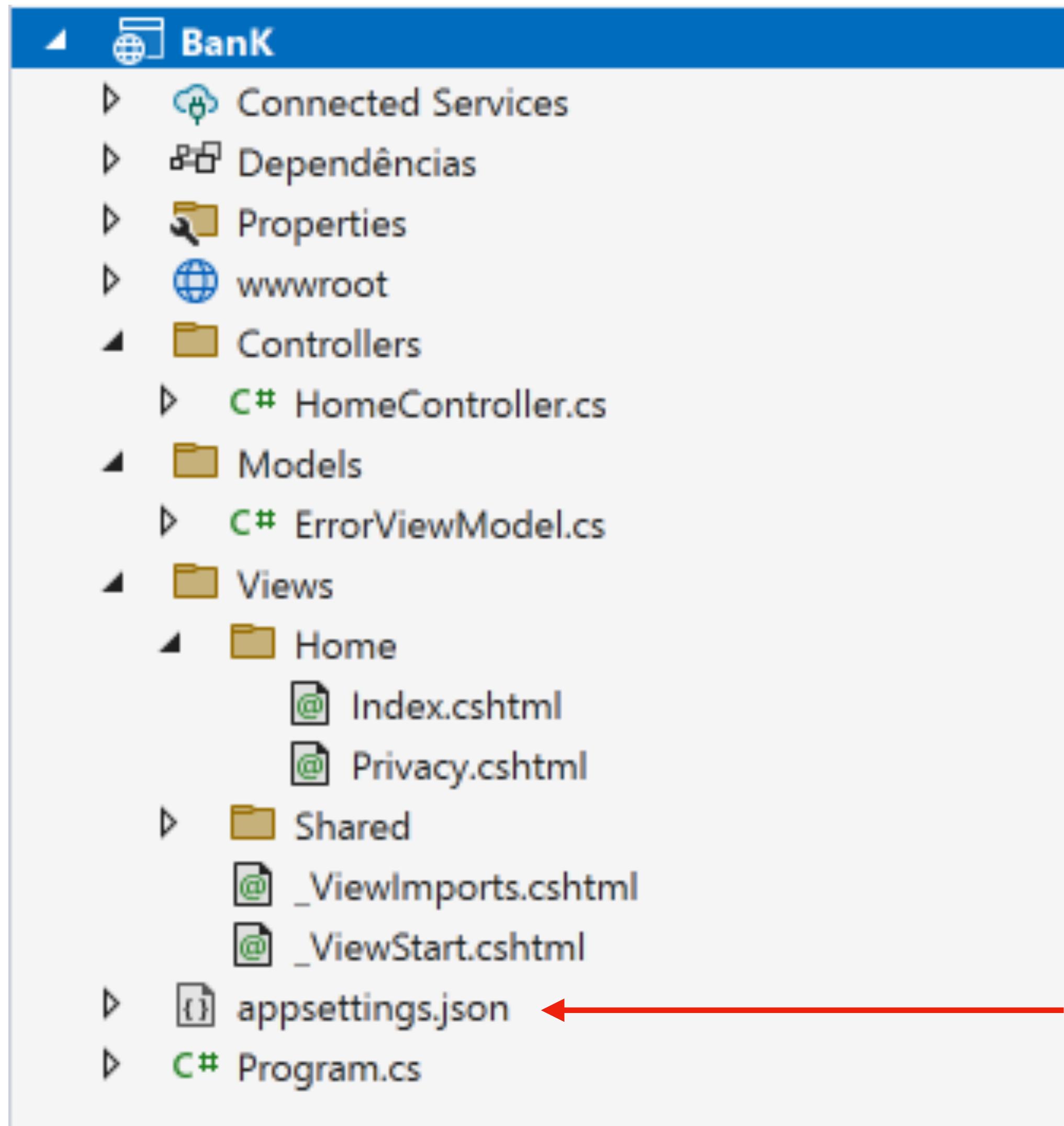
Aplicação Web ASP.Net MVC



Recursos referentes ao site (css, jquery, bootstrap, ...)

Estrutura

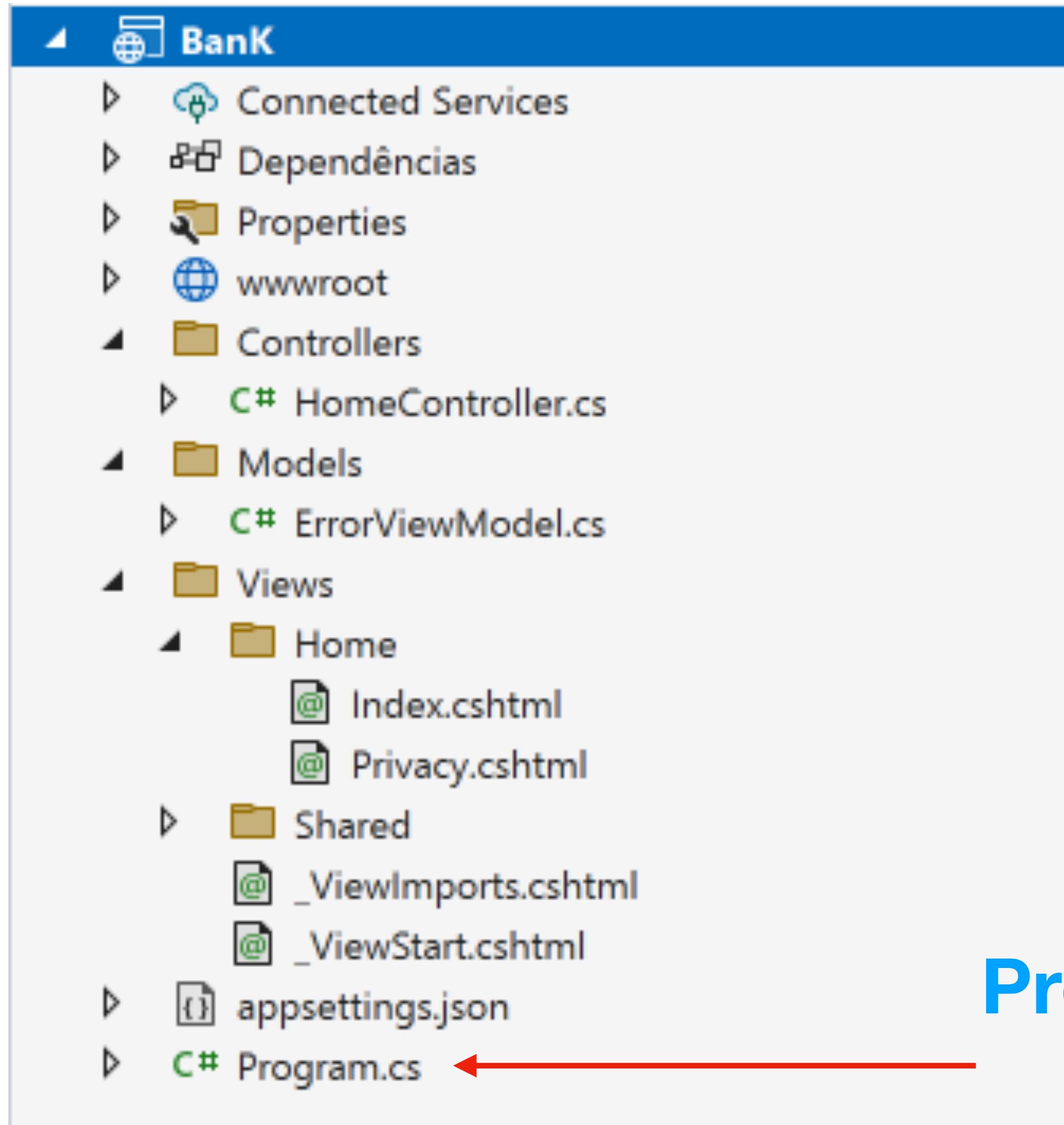
Aplicação Web ASP.Net MVC



Configurações da solução

Estrutura

Aplicação Web ASP.Net MVC



Programa principal (Main)
Vamos conferir no Visual Studio

Model-View-Controller

View

Páginas cshtml

```
1  @{
2      ViewData["Title"] = "Home Page";
3  }
4
5  <div class="text-center">
6      <h1 class="display-4">Welcome</h1>
7      <p>Learn about <a href="https://docs.microsoft.com/aspnet/core">building Web apps with ASP.NET Core</a>.</p>
8  </div>
9
```

Bloco C# (identificado por @)

View

Páginas cshtml

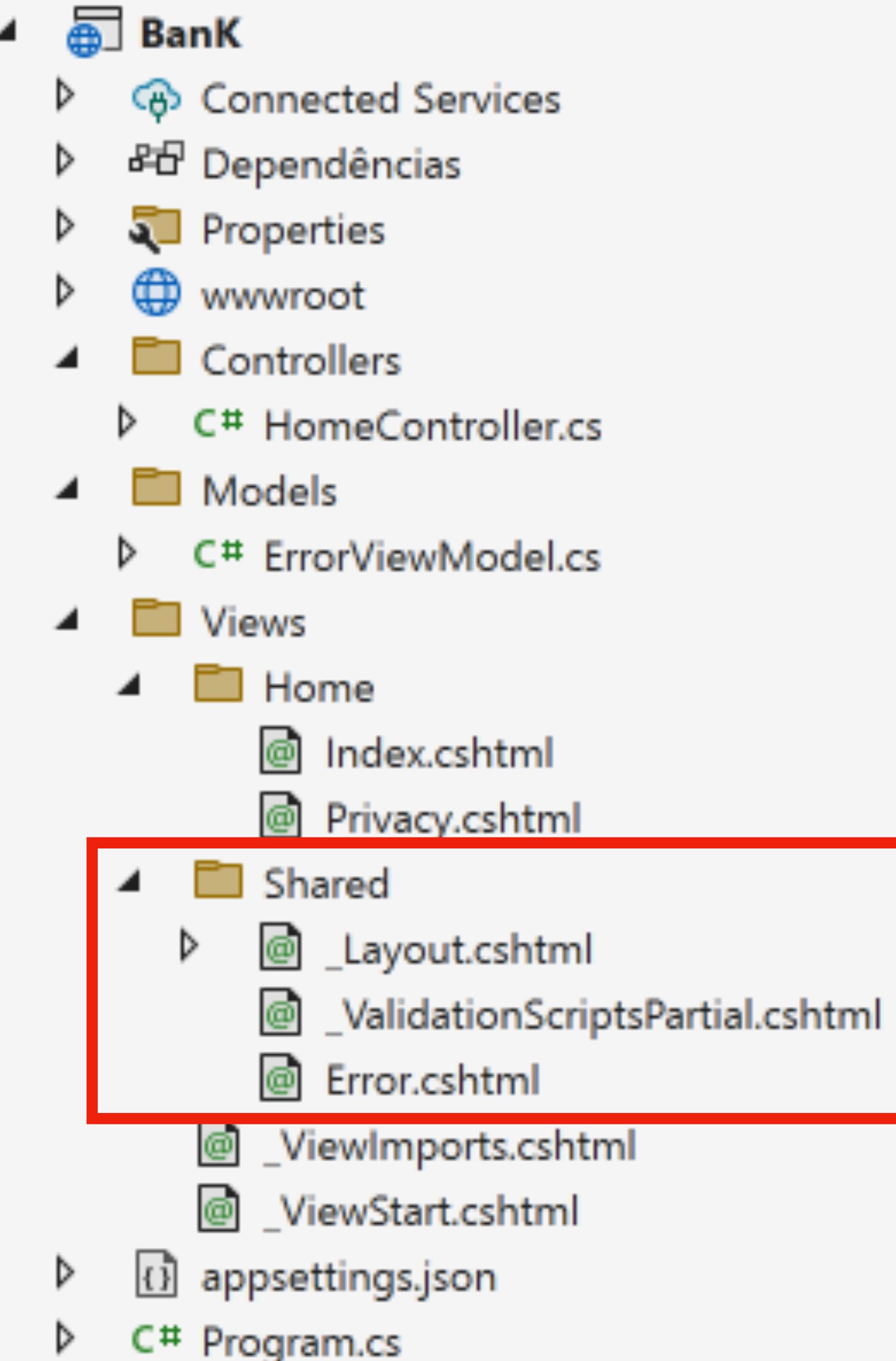
```
1  @{
2      ViewData["Title"] = "Home Page";
3  }
4
5  <div class="text-center">
6      <h1 class="display-4">Welcome</h1>
7      <p>Learn about <a href="https://docs.microsoft.com/aspnet/core">building Web apps with ASP.NET Core</a>.</p>
8  </div>
9
```

Bloco C# (identificado por @)

Essa não é uma página HTML:

1. Ela será processada para se tornar uma página HTML
2. Ela possui um conteúdo parcial, ou seja, não está completa

**Se ela não está completa, onde
está o restante da página?**



Vamos conferir no Visual Studio

Controllers e Rotas

Controller

Objeto C# derivado da classe Controller

```
1  using BanK.Models;
2  using Microsoft.AspNetCore.Mvc;
3  using System.Diagnostics;
4
5  namespace BanK.Controllers
6  {
7      public class HomeController : Controller
8      {
9          private readonly ILogger<HomeController> _logger;
10
11         public HomeController(ILogger<HomeController> logger)
12         {
13             _logger = logger;
14         }
15     }
```

**Como o controller, com várias views,
consegue selecionar a correta?**

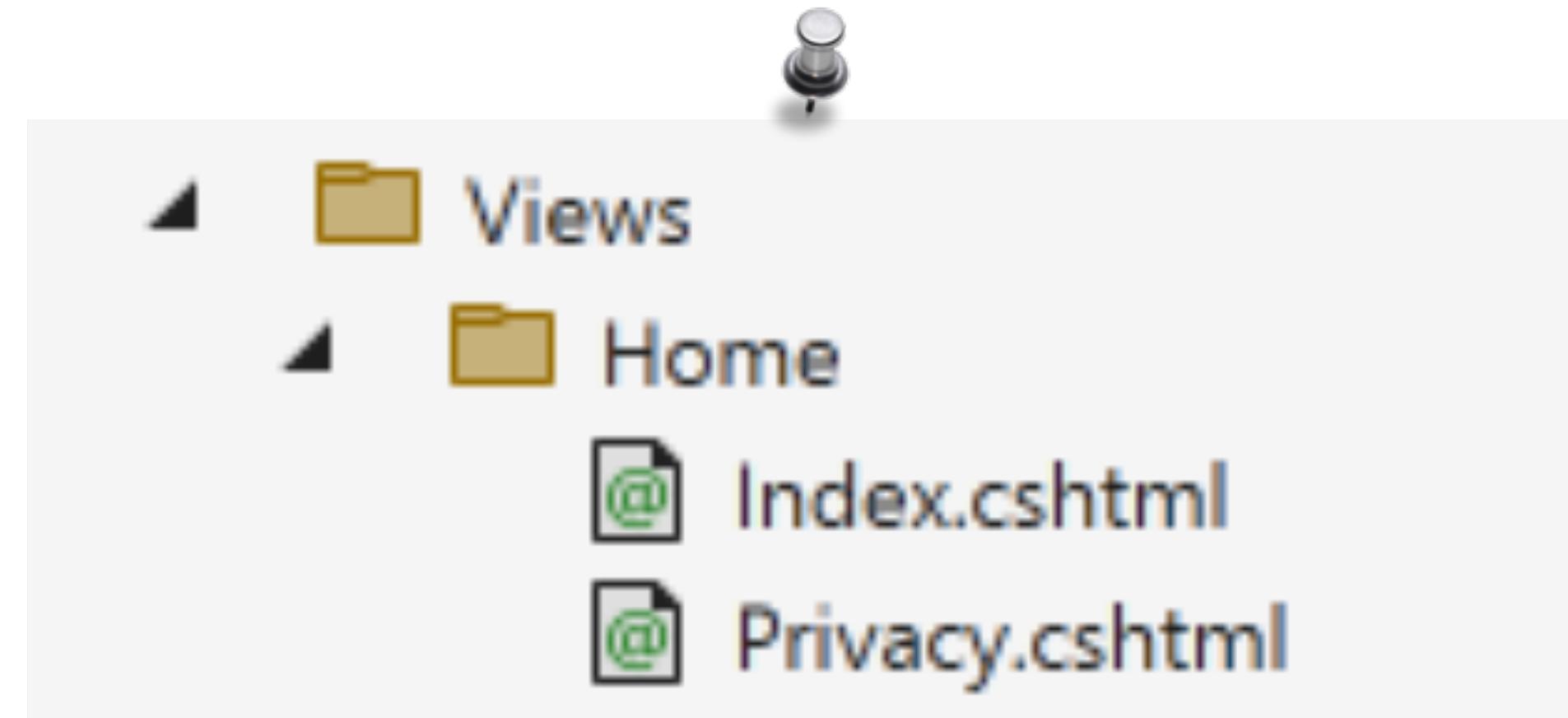
```
public class HomeController : Controller
{
    private readonly ILogger<HomeController> _logger;
    0 referências
    public HomeController(ILogger<HomeController> logger)
    {
        _logger = logger;
    }

    0 referências
    public IActionResult Index()
    {
        return View();
    }

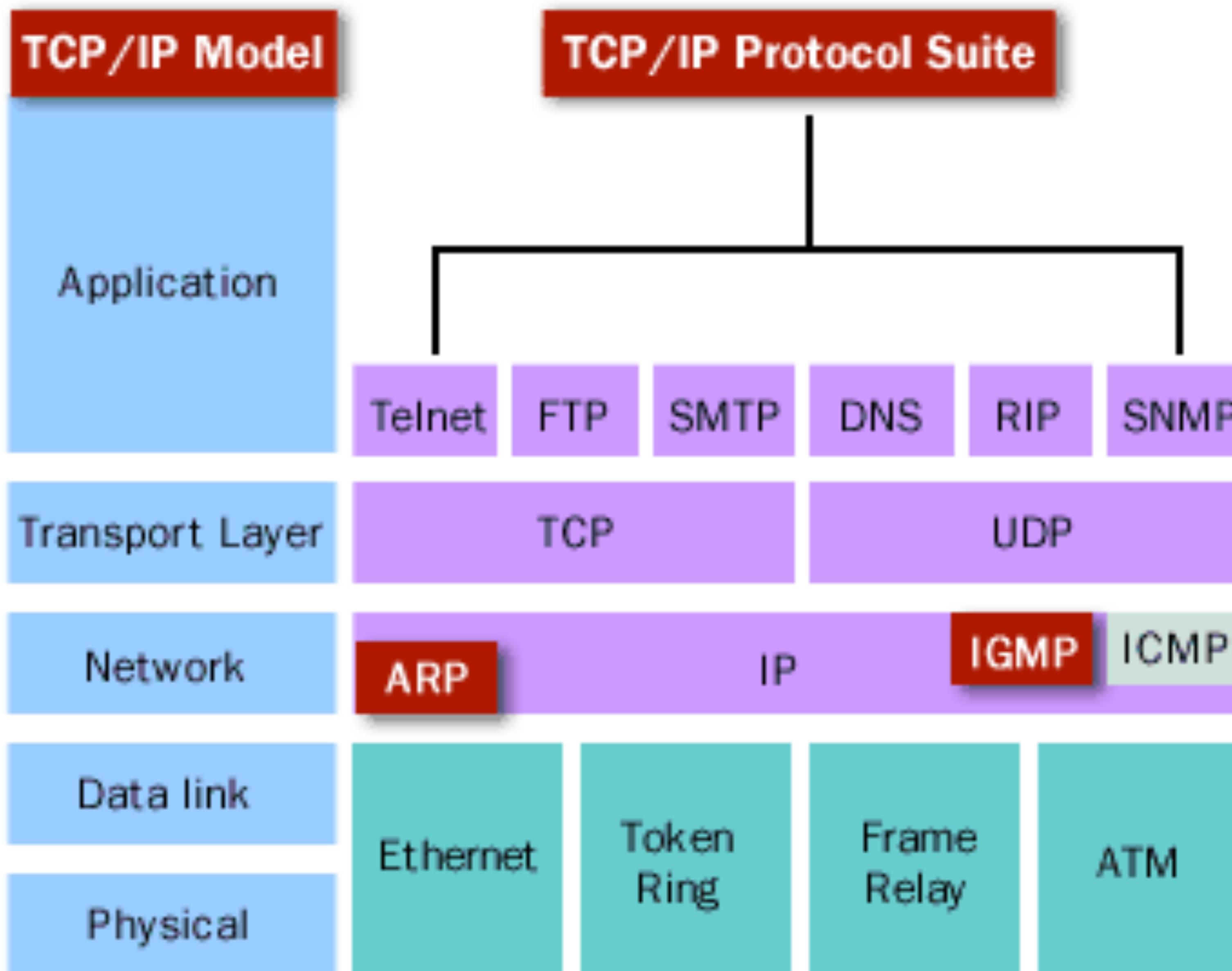
    0 referências
    public IActionResult Privacy()
    {
        return View();
    }

    [ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore = true)]
    0 referências
    public IActionResult Error()
    {
        return View(
            new ErrorViewModel
            { RequestId = Activity.Current?.Id ?? HttpContext.TraceIdentifier });
    }
}
```

Ações são métodos de um controller, que retornam uma exibição nomeada de maneira fluente ao método



**Mas e como possível chamar o
método (ação) no controller?**



[path]/CONTROLLER/ACTION

Program.cs

```
19 app.MapControllerRoute(  
20     name: "default",  
21     pattern: "{controller=Home}/{action=Index}/{id?}");  
22
```

Controller *Action method*
http://localhost:1234/home/index/100 *Id parameter value*

Controller *Action method*
http://localhost:1234/home/index

```
public class BooksController : Controller
{
    // eg: /books
    // eg: /books/1430210079
    [Route("books/{isbn?}")]
    public ActionResult View(string isbn)
    {
        if (!String.IsNullOrEmpty(isbn))
        {
            return View("OneBook", GetBook(isbn));
        }
        return View("AllBooks", GetBooks());
    }

    // eg: /books/lang
    // eg: /books/lang/en
    // eg: /books/lang/he
    [Route("books/lang/{lang=en}")]
    public ActionResult ViewByLanguage(string lang)
    {
        return View("OneBook", GetBooksByLanguage(lang));
    }
}
```

Roteamento Explícito

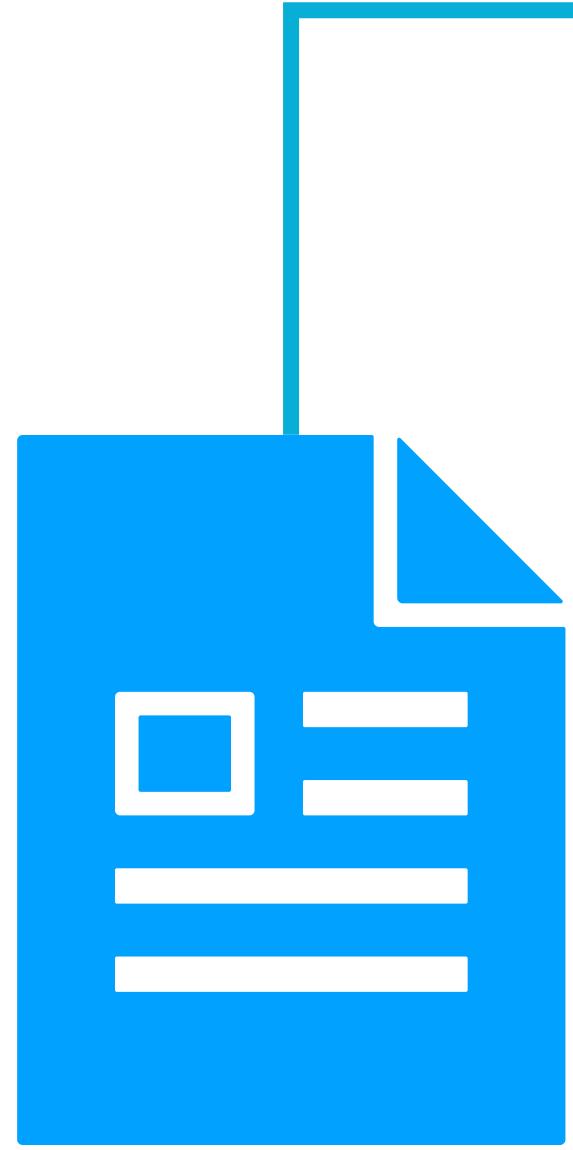
```
app.MapControllerRoute(name: "blog",
    pattern: "blog/{*article}",
    defaults: new { controller = "Blog", action = "Article" });

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");
```

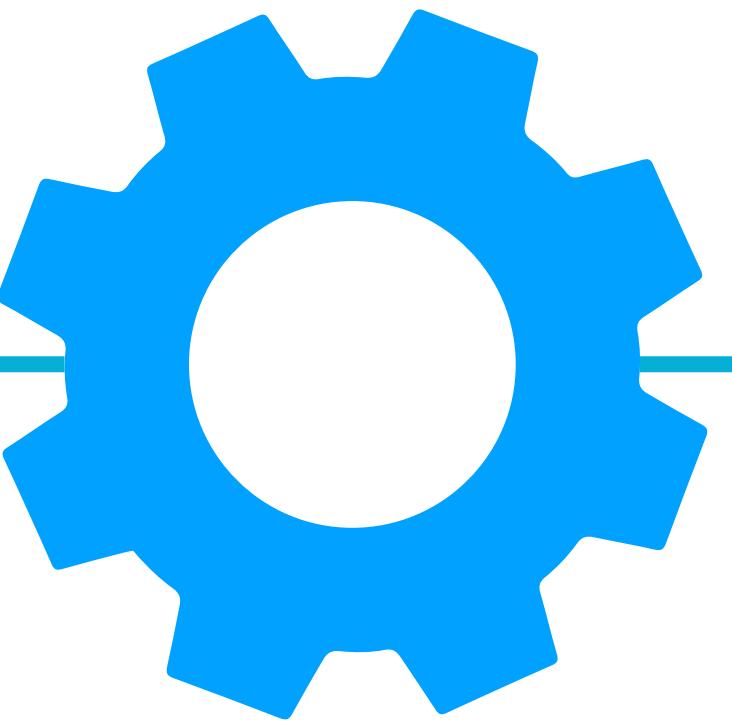
Documentação

Routing to controller actions in ASP.NET Core

Controllers

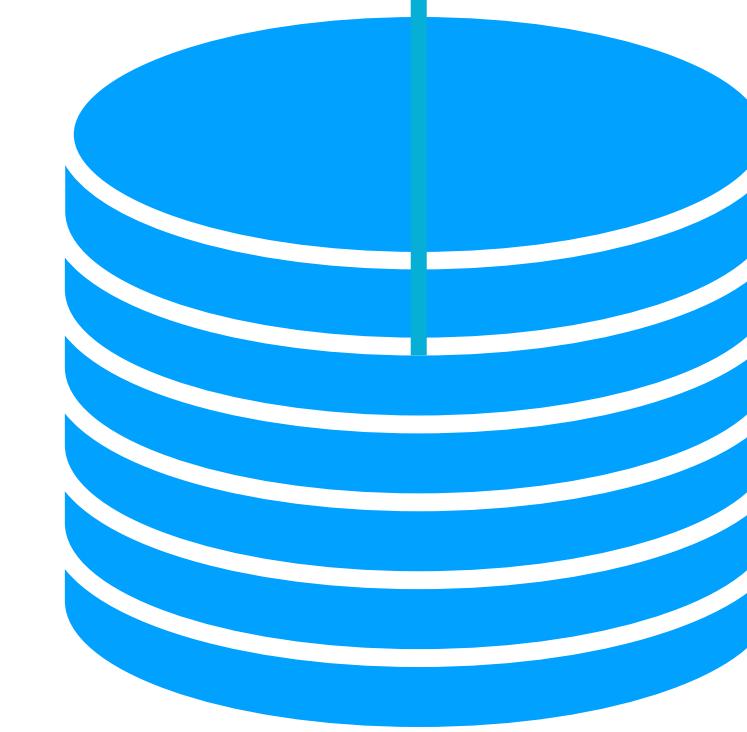


View



Controller

Fazem a orquestração de todo o fluxo de informação, bem como da lógica do negócio.



Model

O controller expõe e estabelece as funcionalidades necessárias ao funcionamento lógico do negócio.

Mas o que isso quer dizer
exatamente?

**Vamos construir uma aplicação a
partir do BD (Database First)**

Al's ToyBarn ContactApp



1. Criar o Banco de Dados

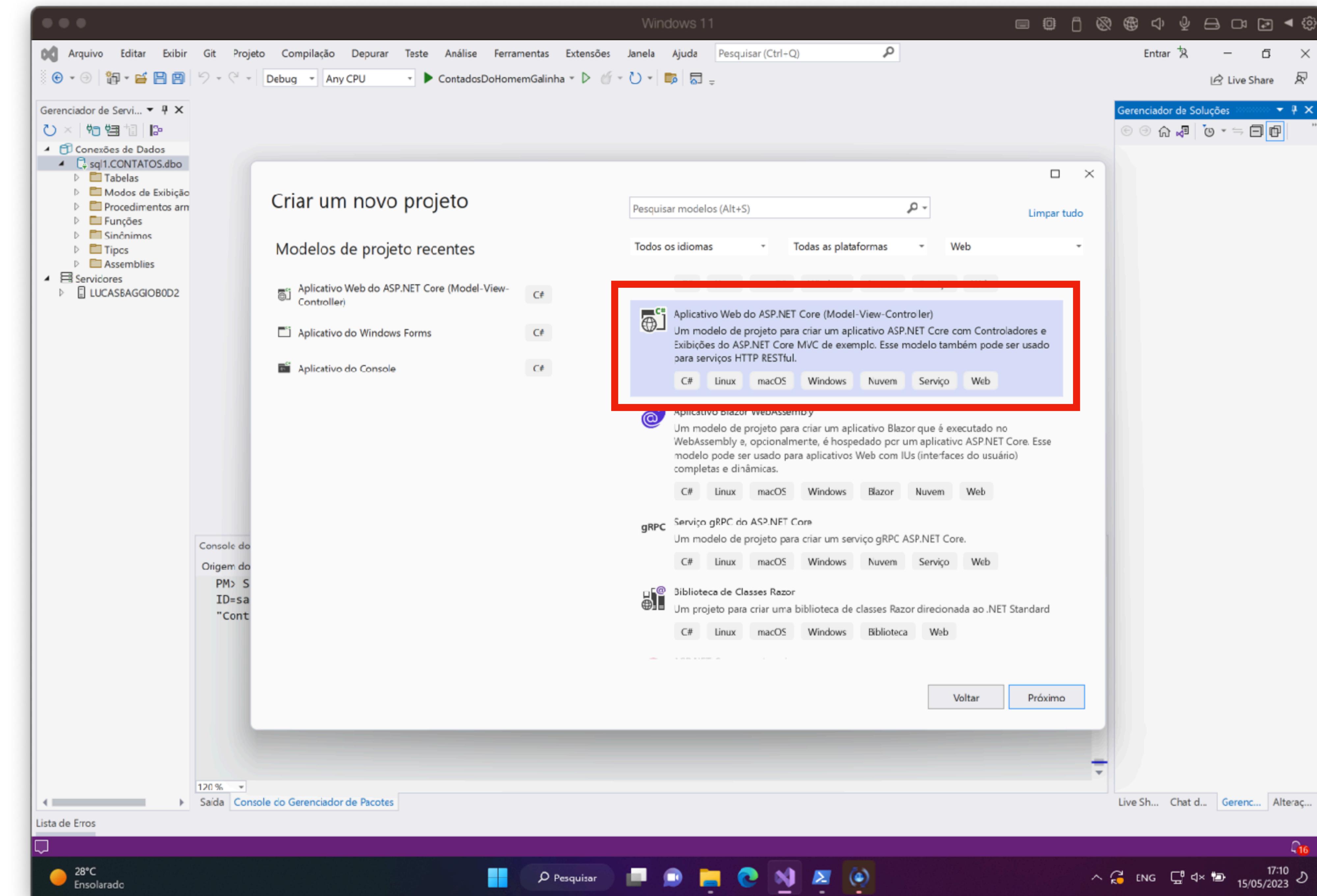


```
1 USE master;
2 CREATE DATABASE CONTATOS;
3 USE CONTATOS;
4 CREATE TABLE Cliente (
5     Id INT IDENTITY,
6     Nome VARCHAR(100) NOT NULL,
7     Endereco VARCHAR(200),
8     CONSTRAINT PK_Cliente PRIMARY KEY (Id)
9 );
10
11 CREATE TABLE Contato (
12     Id INT IDENTITY,
13     Perfil VARCHAR(100),
14     Tipo VARCHAR(10),
15     IdCliente INT,
16     CONSTRAINT PK_Contato PRIMARY KEY (Id, IdCliente),
17     CONSTRAINT FK_Contato_Cliente FOREIGN KEY (IdCliente)
18         REFERENCES Cliente (Id)
19 );
```

```
USE master;
CREATE DATABASE CONTATOS;
USE CONTATOS;
CREATE TABLE Cliente (
    Id INT IDENTITY,
    Nome VARCHAR(100) NOT NULL,
    Endereco VARCHAR(200),
    CONSTRAINT PK_Cliente
        PRIMARY KEY (Id)
);
CREATE TABLE Contato (
    Id INT IDENTITY,
    Perfil VARCHAR(100),
    Tipo VARCHAR(10),
    IdCliente INT,
    CONSTRAINT PK_Contato
        PRIMARY KEY (Id, IdCliente),
    CONSTRAINT FK_Contato_Cliente
        FOREIGN KEY (IdCliente)
            REFERENCES Cliente (Id)
);
```

2. Criar o projeto ASP.Net Core (MVC)





3. Sincronizar classes de Modelo com o Banco de Dados





Object Relationship Management

Classes \leftrightarrow Tabelas

Atributos \leftrightarrow Colunas

Objetos \leftrightarrow Tuplas



Framework que realiza a sincronização entre classes de modelo e o banco de dados, por meio de um configuração de anotação (Data Annotation) ou nomeação fluente (Fluente API).

PROBLEMA

```
[Table("Cliente")]
public partial class Cliente
{
    public Cliente()
    {
        Contatos = new HashSet<Contato>();
    }

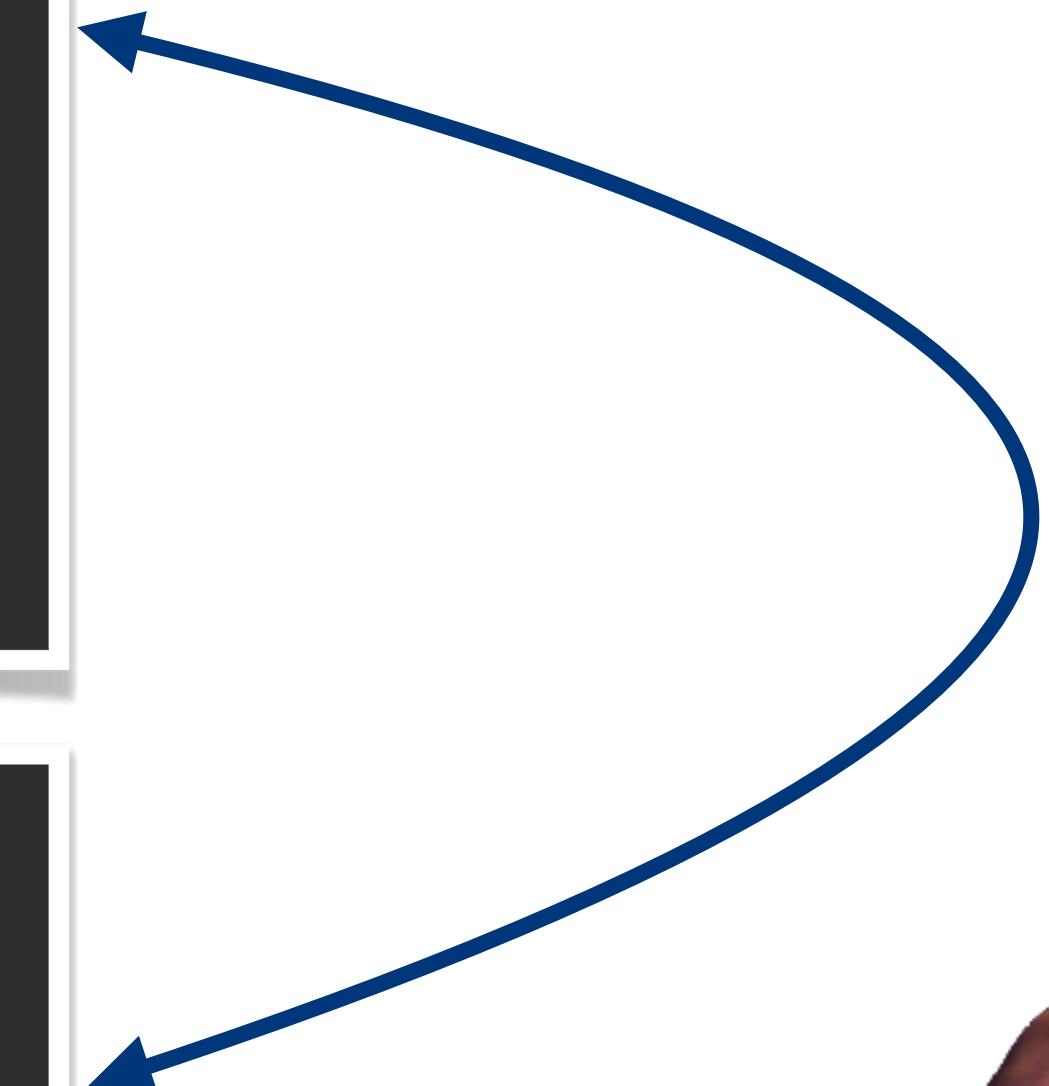
    [Key]
    public int Id { get; set; }
    [StringLength(100)]
    [Unicode(false)]
    public string Nome { get; set; } = null!;
    [StringLength(200)]
    [Unicode(false)]
    public string? Endereco { get; set; }

    [InverseProperty(nameof(Contato.IdClienteNavigation))]
    public virtual ICollection<Contato> Contatos { get; set; }
}
```

```
CREATE TABLE Cliente (
    Id INT IDENTITY,
    Nome VARCHAR(100) NOT NULL,
    Endereco VARCHAR(200),
    CONSTRAINT PK_Cliente PRIMARY KEY (Id)
);
```

Retrabalho

Erros entre anotação
e o RIR/RIE



PROBLEMA

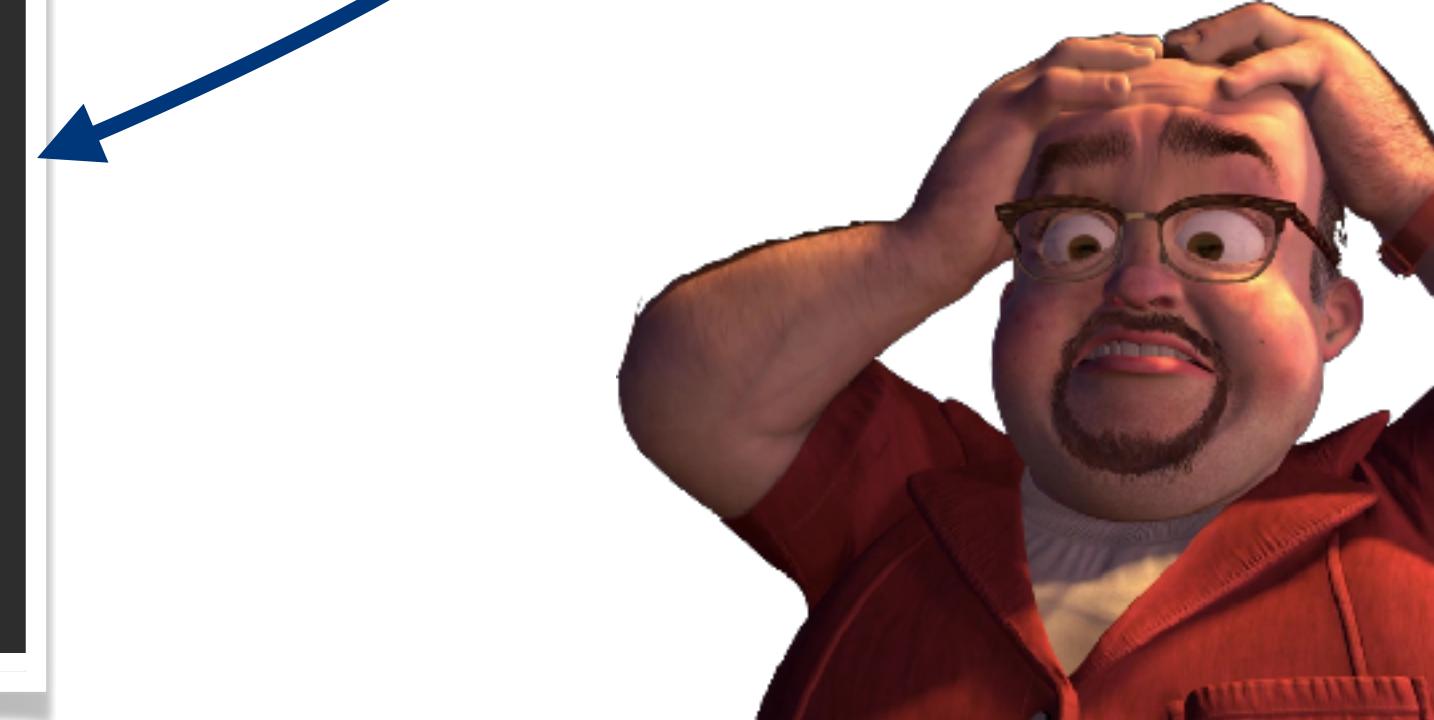
```
[Table("Contato")]
public partial class Contato
{
    [Key]
    public int Id { get; set; }
    [StringLength(100)]
    [Unicode(false)]
    public string? Perfil { get; set; }
    [StringLength(10)]
    [Unicode(false)]
    public string? Tipo { get; set; }
    [Key]
    public int IdCliente { get; set; }

    [ForeignKey(nameof(IdCliente))]
    [InverseProperty(nameof(Cliente.Contatos))]
    public virtual Cliente IdClienteNavigation { get; set; } = null!;
}
```

Retrabalho

Erros entre anotação
e o RIR/RIE

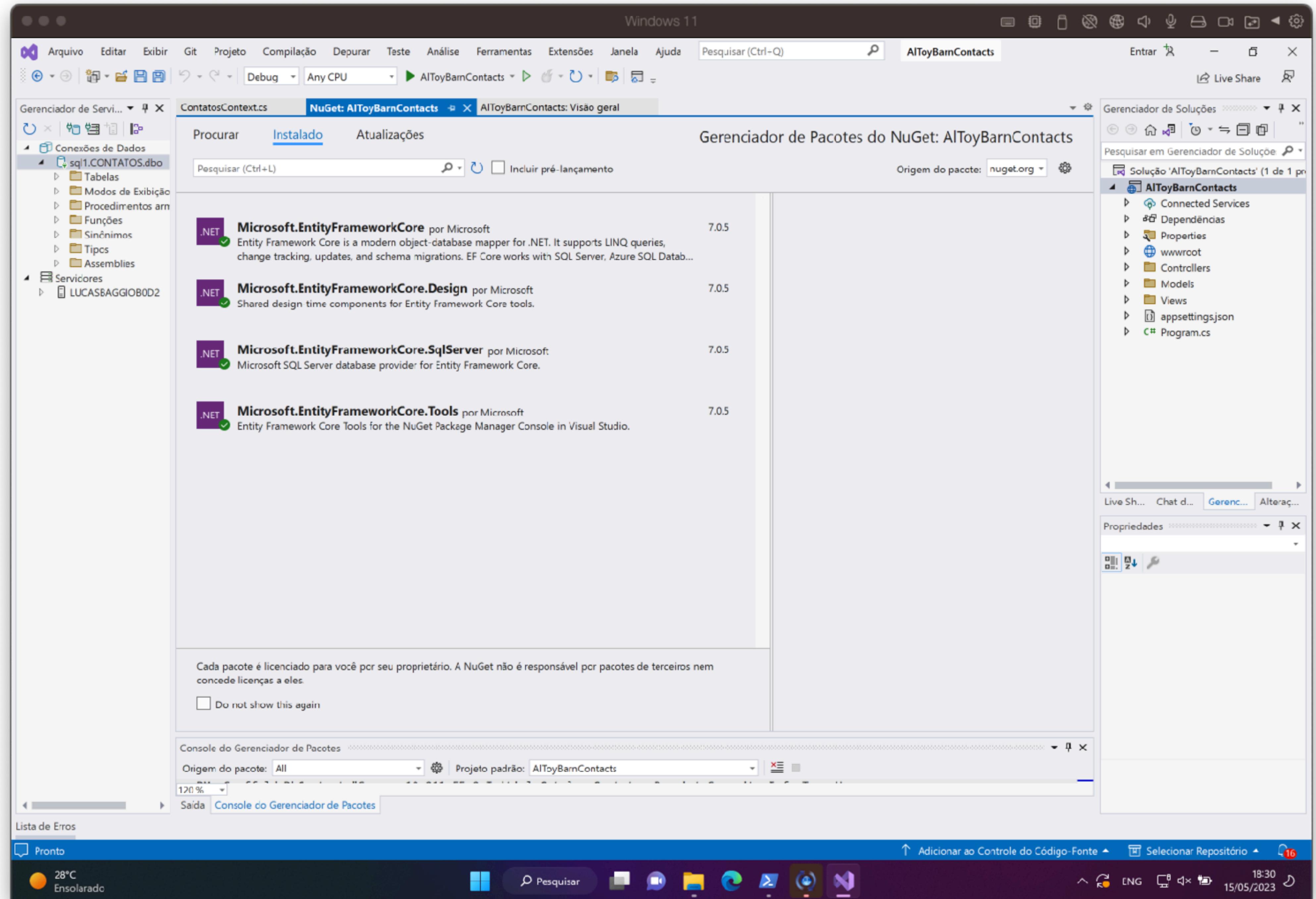
```
CREATE TABLE Contato (
    Id INT IDENTITY,
    Perfil VARCHAR(100),
    Tipo VARCHAR(10),
    IdCliente INT,
    CONSTRAINT PK_Contato PRIMARY KEY (Id, IdCliente),
    CONSTRAINT FK_Contato_Cliente FOREIGN KEY (IdCliente)
        REFERENCES Cliente (Id)
);
```



3. Automação com Scaffold-DbContext



Instalar pacotes



Scaffold-DbContext

String de Conexão (cada um tem a sua)

```
"Server=10.211.55.2;Initial Catalog=Contatos;Persist Security Info=True;User ID=sa;Password=BK0_eh_f0d@;TrustServerCertificate=true"
```

`Microsoft.EntityFrameworkCore.SqlServer`

Driver do EF para o Banco de Dados

`-OutputDir Models`

Diretório onde as classes de modelo serão criadas

`-Context "ContatosContext"`

Nome da classe que fará o acesso às tabelas (tuplas)

`-DataAnnotations`

Gerar anotações de maneira explícita, não fluente.

`-NoOnConfiguring`

Não configurar a conexão com a string dada

`-Force`

Sobrescrever quaisquer arquivos anteriormente gerados

Windows 11

Arquivo Editar Exibir Git Projeto Compilação Depurar Teste Análise Ferramentas Extensões Janela Ajuda Pesquisar (Ctrl-Q) Entrar Live Share

Debug Any CPU AlToyBarnContacts: Visão geral

Gerenciador de Servi... X ContatosContext.cs X NuGet: AlToyBarnContacts

AlToyBarnContacts

ContatosContext.cs

```
1 using System;
2 using System.Collections.Generic;
3 using Microsoft.EntityFrameworkCore;
4
5 namespace AlToyBarnContacts.Models;
6
7 public partial class ContatosContext : DbContext
8 {
```

Conexões de Dados

- sql1.CONTATOS.dbo
 - Tabelas
 - Modos de Exibição
 - Procedimentos armados
 - Funções
 - Síncronos
 - Tipos
 - Assemblies
- Servidores
- LUCASBAGGIOB0D2

Não foi encontrado nenhum problema

Gerenciador de Soluções

Solução 'AlToyBarnContacts' (1 de 1 projeto)

- AlToyBarnContacts
 - Connected Services
 - Dependências
 - Properties
 - wwwroot
 - Controllers
 - Models
 - Views
 - appsettings.json
 - Program.cs

Console do Gerenciador de Pacotes

Origem do pacote: All Projeto padrão: AlToyBarnContacts

```
PM> Scaffold-DbContext "Server=10.211.55.2;Initial Catalog=Contatos;Persist Security Info=True;User ID=sa;Password=BK0_eh_f0d@;TrustServerCertificate=true" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models -Context "ContatosContext" -DataAnnotations -NoOnConfiguring -Force
Build started...
Build succeeded.
PM>
```

Lista de Erros

Pronto Adicionar ao Controle do Código-Fonte Seleccionar Repositório 16

28°C Ensolarado

Pesquisar

18:32 15/05/2023

Cliente.cs

The screenshot shows the Microsoft Visual Studio 2022 interface with the following details:

- Title Bar:** Windows 11, Arquivo, Editar, Exibir, Git, Projeto, Compilação, Depurar, Teste, Análise, Ferramentas, Extensões, Janela, Ajuda, Pesquisar (Ctrl+Q), AIToyBarnContacts, Entrar, Live Share.
- Solution Explorer:** Shows the solution 'AlToyBarnContacts' with its projects, files, and properties.
- Code Editor:** Displays the 'Cliente.cs' file content. The code defines a partial class 'Cliente' with properties 'Id' (primary key), 'Nome' (string), and 'Endereco' (string?). It also includes a navigation property 'Contatos' of type 'ICollection<Contato>'.
- Status Bar:** Shows the status bar with various icons and text, including '120 %', 'Não foi encontrado nenhum problema', 'Ln: 1 Car: 1 SPC CRLF', and a commit count of '16'.
- Taskbar:** Shows the taskbar with the VS icon, search bar, pinned icons for File Explorer, Task View, Task Manager, and others, and system status icons.

```
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel.DataAnnotations;
4 using System.ComponentModel.DataAnnotations.Schema;
5 using Microsoft.EntityFrameworkCore;
6
7 namespace AlToyBarnContacts.Models;
8
9 [Table("Cliente")]
10 public partial class Cliente
11 {
12     [Key]
13     public int Id { get; set; }
14
15     [StringLength(100)]
16     [Unicode(false)]
17     public string Nome { get; set; } = null!;
18
19     [StringLength(200)]
20     [Unicode(false)]
21     public string? Endereco { get; set; }
22
23     [InverseProperty("IdClienteNavigation")]
24     public virtual ICollection<Contato> Contatos { get; set; } = new List<Contato>();
25
26 }
```

Cliente.cs

```
7  namespace AlToyBarnContacts.Models;
8
9  [Table("Cliente")]
10 public partial class Cliente
11 {
12     [Key]
13     public int Id { get; set; }
14
15     [StringLength(100)]
16     [Unicode(false)]
17     public string Nome { get; set; } = null!;
18
19     [StringLength(200)]
20     [Unicode(false)]
21     public string? Endereco { get; set; }
22
23     [InverseProperty("IdClienteNavigation")]
24     public virtual ICollection<Contato> Contatos { get; set; } = new List<Contato>();
25 }
26
```

Contato.cs

The screenshot shows the Microsoft Visual Studio interface with the following details:

- Title Bar:** Windows 11, Arquivo, Editar, Exibir, Git, Projeto, Compilação, Depurar, Teste, Análise, Ferramentas, Extensões, Janela, Ajuda, Pesquisar (Ctrl-Q), AIToyBarnContacts, Entrar.
- Toolbar:** Standard icons for file operations, search, and navigation.
- Code Editor:** The main window displays the `Contato.cs` file content. The code defines a partial class `Contato` within the `AlToyBarnContacts.Models` namespace. It includes properties `Id`, `Perfil`, `Tipo`, and `IdCliente`, each annotated with attributes like `[Key]`, `[StringLength]`, and `[ForeignKey]`. The code editor shows 4 references for the `Contato` class.
- Solution Explorer:** Shows the project structure for `AIToyBarnContacts`, including `Connected Services`, `Dependências`, `Properties`, `wwwroot`, `Controllers`, `Models` (containing `Cliente.cs`, `Contato.cs`, `ContatosContext.cs`, `ErrorViewModel.cs`), `Views`, `appsettings.json`, and `Program.cs`.
- Properties Window:** Available on the right side of the interface.
- Status Bar:** Shows the current file is `Contato.cs`, line 7, character 36, and includes links for `Adicionar ao Controle do Código-Fonte` and `Selecionar Repositório`.
- Bottom Bar:** Shows system status (weather, battery, network) and date/time (19:02, 15/05/2023).

Contato.cs

```
7  namespace AlToyBarnContacts.Models;
8
9 [PrimaryKey("Id", "IdCliente")]
10 [Table("Contato")]
11 4 referências
12  public partial class Contato
13 {
14     [Key]
15     1 referência
16     public int Id { get; set; }
17
18     [StringLength(100)]
19     [Unicode(false)]
20     0 referências
21     public string? Perfil { get; set; }
22
23     [StringLength(10)]
24     [Unicode(false)]
25     0 referências
26     public string? Tipo { get; set; }
27
28     [Key]
29     0 referências
30     public int IdCliente { get; set; }
31
32     [ForeignKey("IdCliente")]
33     [InverseProperty("Contatos")]
34     1 referência
35     public virtual Cliente IdClienteNavigation { get; set; } = null!;
36 }
```

ContatosContext.cs

The screenshot shows the Microsoft Visual Studio interface with the following details:

- Title Bar:** Windows 11, Arquivo, Editar, Exibir, Git, Projeto, Compilação, Depurar, Teste, Análise, Ferramentas, Extensões, Janela, Ajuda, Pesquisar (Ctrl-Q), AIToyBarnContacts, Entrar, Live Share.
- Toolbar:** Standard Visual Studio toolbar with icons for file operations, search, and navigation.
- Code Editor:** The main window displays the `ContatosContext.cs` file. The code defines a partial class `ContatosContext` that inherits from `DbContext`. It includes properties for `Cientes` and `Contatos`, and overrides the `OnModelCreating` method to configure relationships between `Contato` and `Cliente`.
- Solution Explorer:** Shows the project structure for `AIToyBarnContacts`, including files like `Contato.cs`, `Cliente.cs`, `ContatosContext.cs`, `ErrorViewModel.cs`, and `Program.cs`.
- Properties Window:** Shows the properties for selected items in the Solution Explorer.
- Task List:** Shows a single reference to the `OnModelCreatingPartial` method.
- Status Bar:** Shows the date and time (20:13, 15/05/2023), weather (28°C, Ensolarado), and system status.

```
namespace AIToyBarnContacts.Models.ContatosContext
{
    public partial class ContatosContext : DbContext
    {
        public ContatosContext(DbContextOptions<ContatosContext> options)
            : base(options)
        {
        }

        public virtual DbSet<Cliente> Clientes { get; set; }

        public virtual DbSet<Contato> Contatos { get; set; }

        protected override void OnModelCreating(ModelBuilder modelBuilder)
        {
            modelBuilder.Entity<Contato>(entity =>
            {
                entity.Property(e => e.Id).ValueGeneratedOnAdd();

                entity.HasOne(d => d.IdClienteNavigation).WithMany(p => p.Contatos)
                    .OnDelete(DeleteBehavior.ClientSetNull)
                    .HasConstraintName("FK_Contato_Cliente");
            });

            OnModelCreatingPartial(modelBuilder);
        }

        partial void OnModelCreatingPartial(ModelBuilder modelBuilder);
    }
}
```

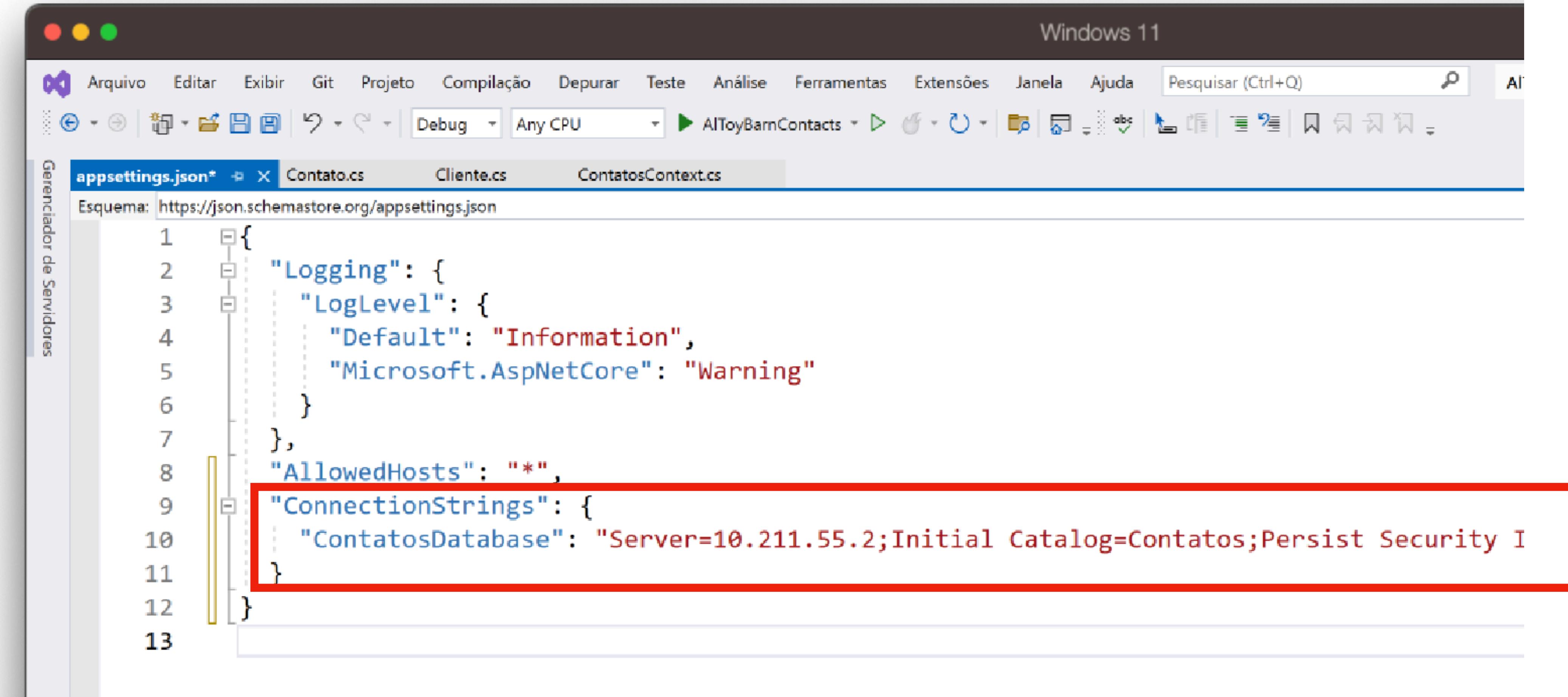
ContatosContext.cs

```
0 referências  
14     public virtual DbSet<Cliente> Clientes { get; set; }  
15  
16     public virtual DbSet<Contato> Contatos { get; set; }  
17  
18     protected override void OnModelCreating(ModelBuilder modelBuilder)  
19     {  
20         modelBuilder.Entity<Contato>(entity =>  
21         {  
22             entity.Property(e => e.Id).ValueGeneratedOnAdd();  
23  
24             entity.HasOne(d => d.IdClienteNavigation).WithMany(p => p.Contatos)  
25                 .onDelete(DeleteBehavior.ClientSetNull)  
26                 .HasConstraintName("FK_Contato_Cliente");  
27         });  
28  
29         OnModelCreatingPartial(modelBuilder);  
30     }  
--
```

Precisamos acertar a string de
conexão no projeto asp.net



appsettings.json

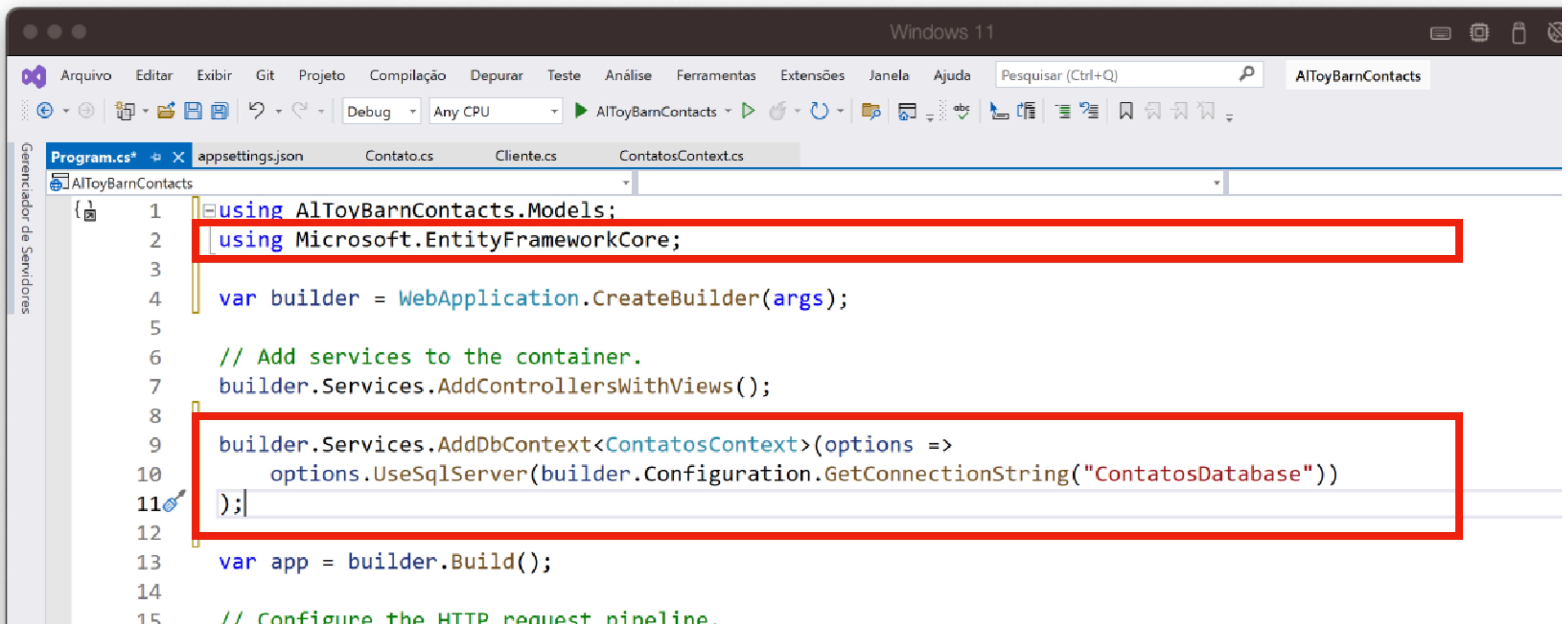


The screenshot shows the Visual Studio IDE interface on Windows 11. The title bar reads "Windows 11". The menu bar includes "Arquivo", "Editar", "Exibir", "Git", "Projeto", "Compilação", "Depurar", "Teste", "Análise", "Ferramentas", "Extensões", "Janela", "Ajuda", and "Pesquisar (Ctrl+Q)". The toolbar below the menu bar includes icons for file operations, search, and project navigation. The tabs at the top of the code editor are "appsettings.json*", "Contato.cs", "Cliente.cs", and "ContatosContext.cs". The "appsettings.json*" tab is selected. The code editor displays the following JSON configuration:

```
1  {
2    "Logging": {
3      "LogLevel": {
4        "Default": "Information",
5        "Microsoft.AspNetCore": "Warning"
6      }
7    },
8    "AllowedHosts": "*",
9    "ConnectionStrings": {
10      "ContatosDatabase": "Server=10.211.55.2;Initial Catalog=Contatos;Persist Security I
11    }
12  }
```

The "ConnectionStrings" section is highlighted with a red rectangular box.

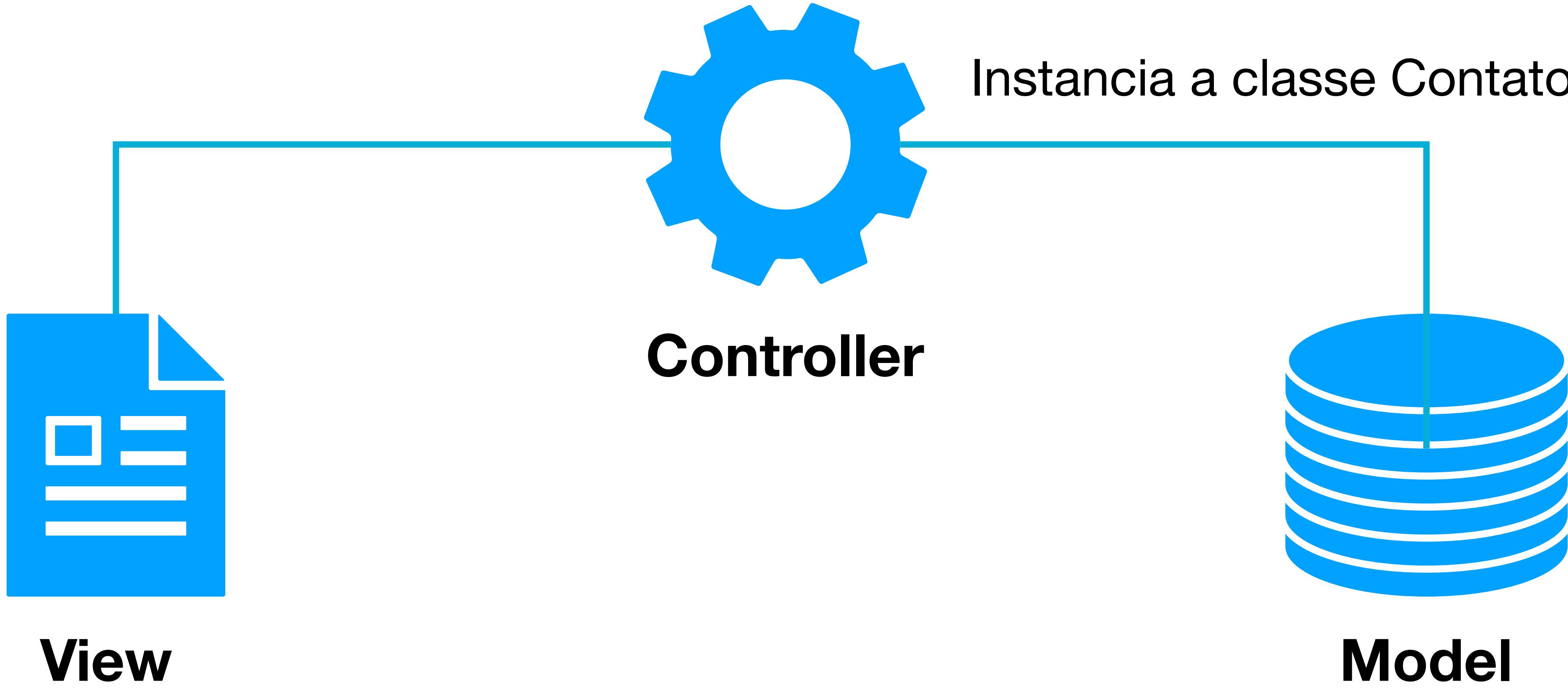
Program.cs



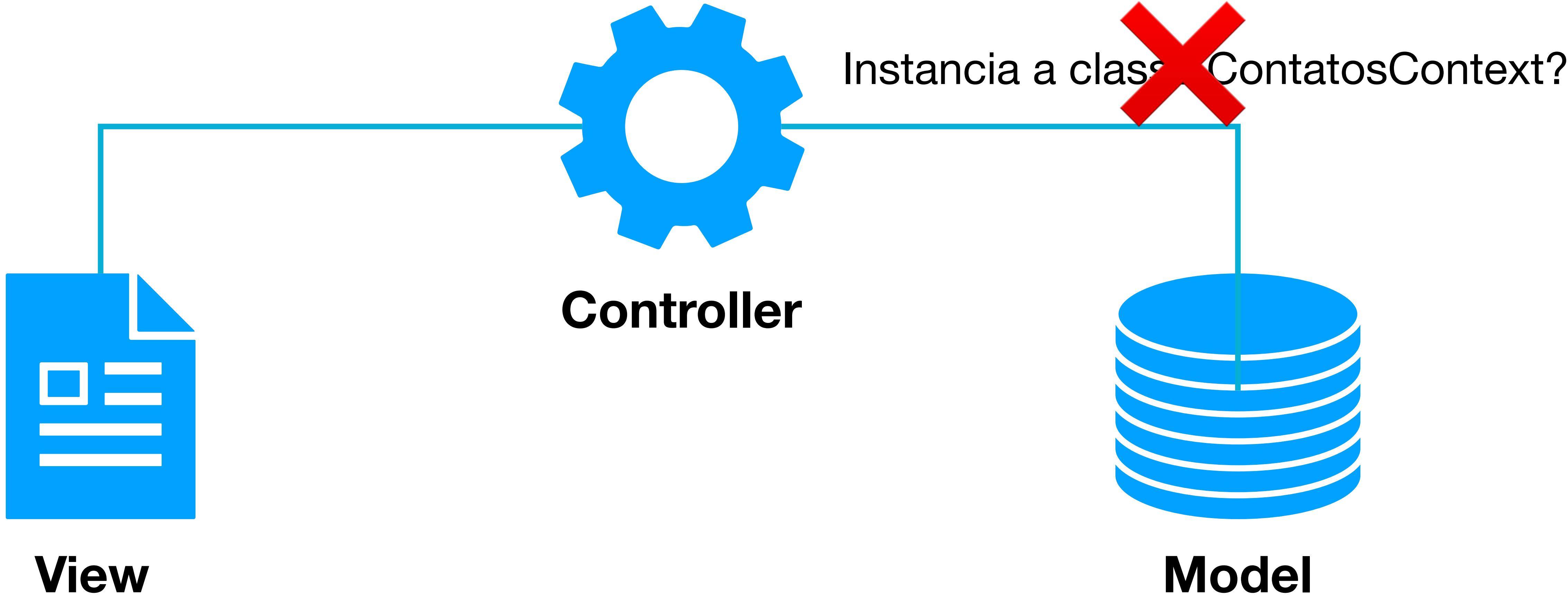
The screenshot shows the Windows 11 desktop environment with the Visual Studio IDE open. The title bar reads "Windows 11". The menu bar includes "Arquivo", "Editar", "Exibir", "Git", "Projeto", "Compilação", "Depurar", "Teste", "Análise", "Ferramentas", "Extensões", "Janela", "Ajuda", and a search bar "Pesquisar (Ctrl+Q)". The toolbar contains various icons for file operations like Open, Save, and Build. The solution explorer on the left shows a project named "AlToyBarnContacts". The code editor displays the "Program.cs" file with the following content:

```
1  using AlToyBarnContacts.Models;
2  using Microsoft.EntityFrameworkCore;
3
4  var builder = WebApplication.CreateBuilder(args);
5
6  // Add services to the container.
7  builder.Services.AddControllersWithViews();
8
9  builder.Services.AddDbContext<ContatosContext>(options =>
10    options.UseSqlServer(builder.Configuration.GetConnectionString("ContatosDatabase"))
11  );
12
13  var app = builder.Build();
14
15  // Configure the HTTP request pipeline.
```

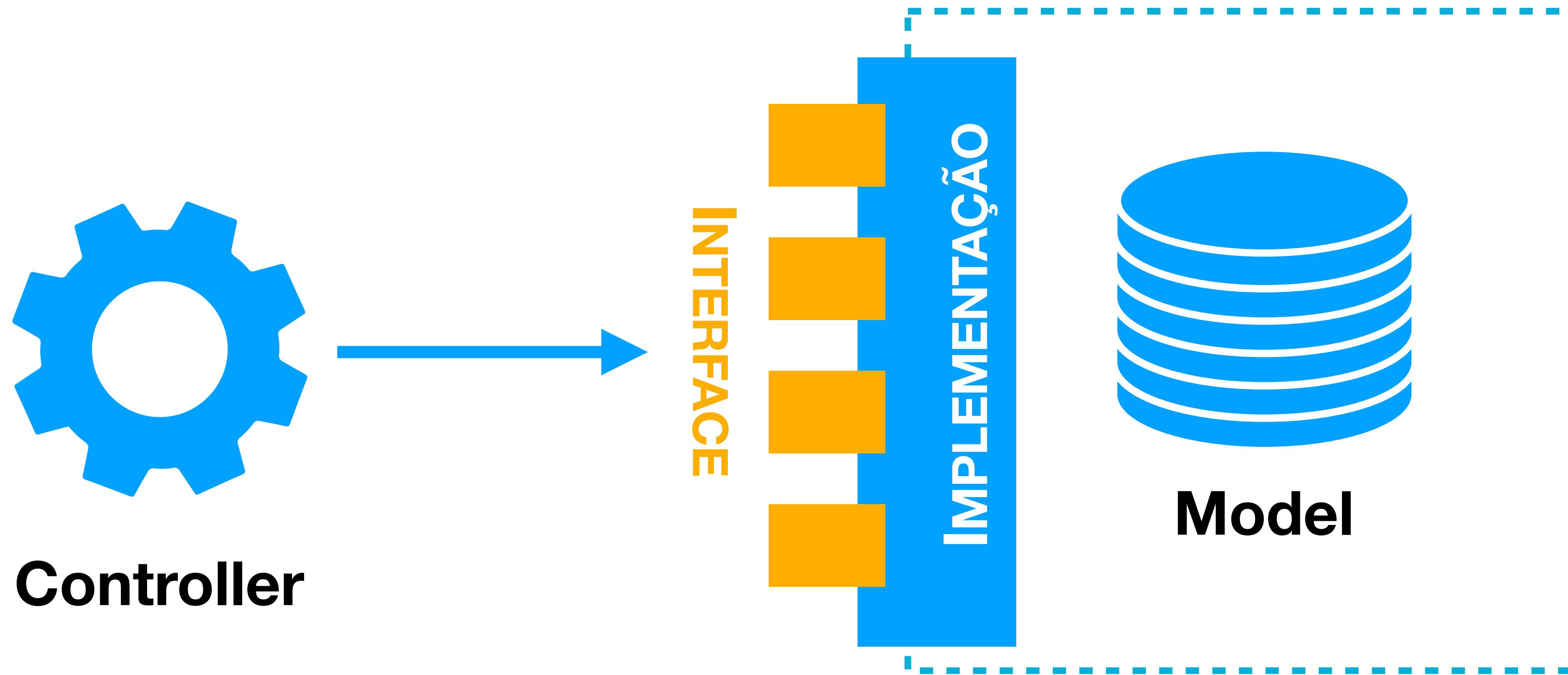
Two sections of the code are highlighted with red boxes: the "using" statements at the top and the DbContext registration in the builder services.



**Uma classe DBContext é a
ponte de comunicação entre o
ASP.Net e o Banco de Dados**



Não vamos depender de implementações, e sim de abstrações

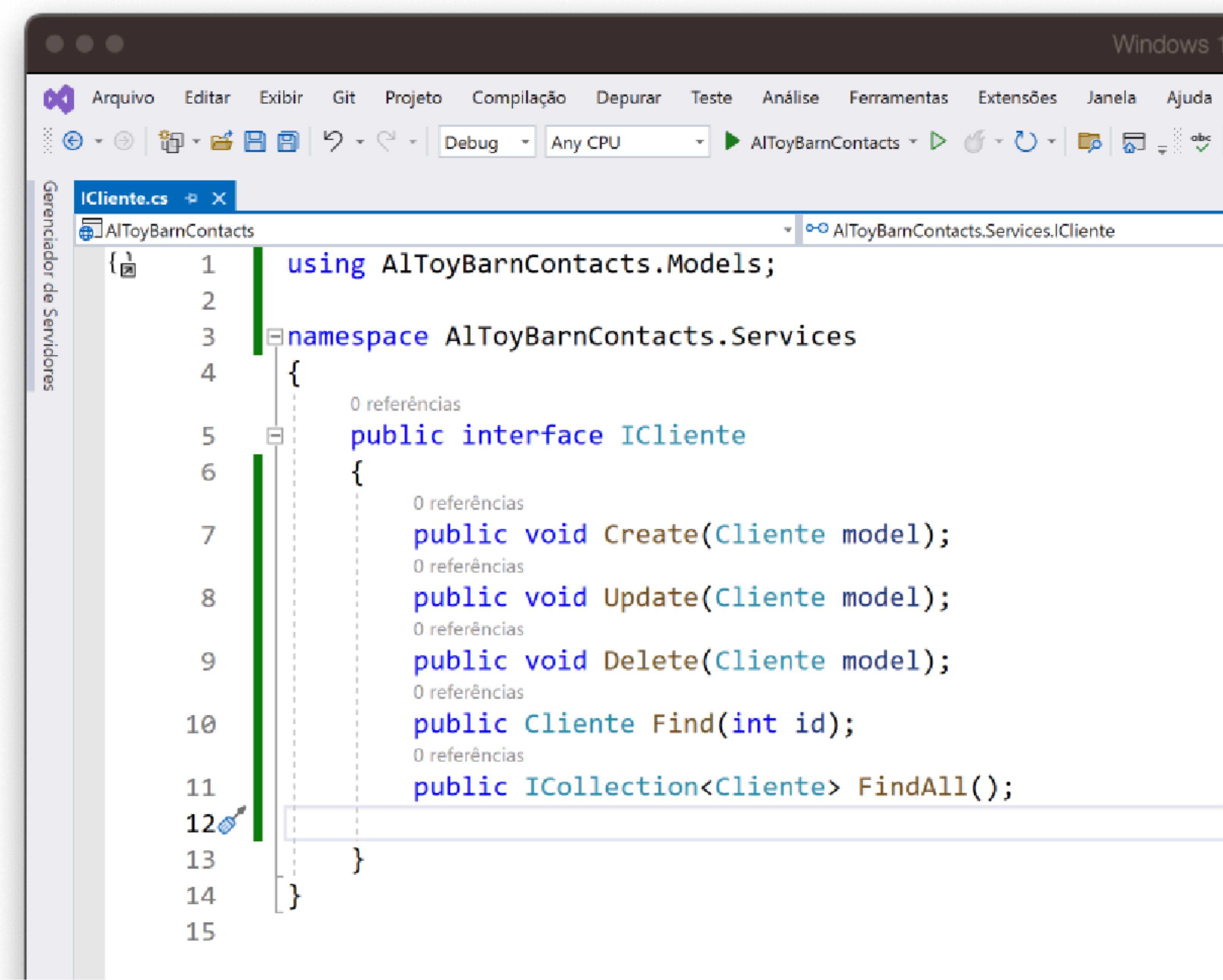


O Controller depende exclusivamente da Interface.

Vamos criar a interface então.



Services\IClienteService.cs



The screenshot shows the Microsoft Visual Studio IDE interface with the title bar "Windows 11". The menu bar includes "Arquivo", "Editar", "Exibir", "Git", "Projeto", "Compilação", "Depurar", "Teste", "Análise", "Ferramentas", "Extensões", "Janela", and "Ajuda". The toolbar has icons for file operations like Open, Save, and Build. The status bar shows "Debug Any CPU" and the project name "AlToyBarnContacts". The code editor window is titled "ICliente.cs" and displays the following C# interface definition:

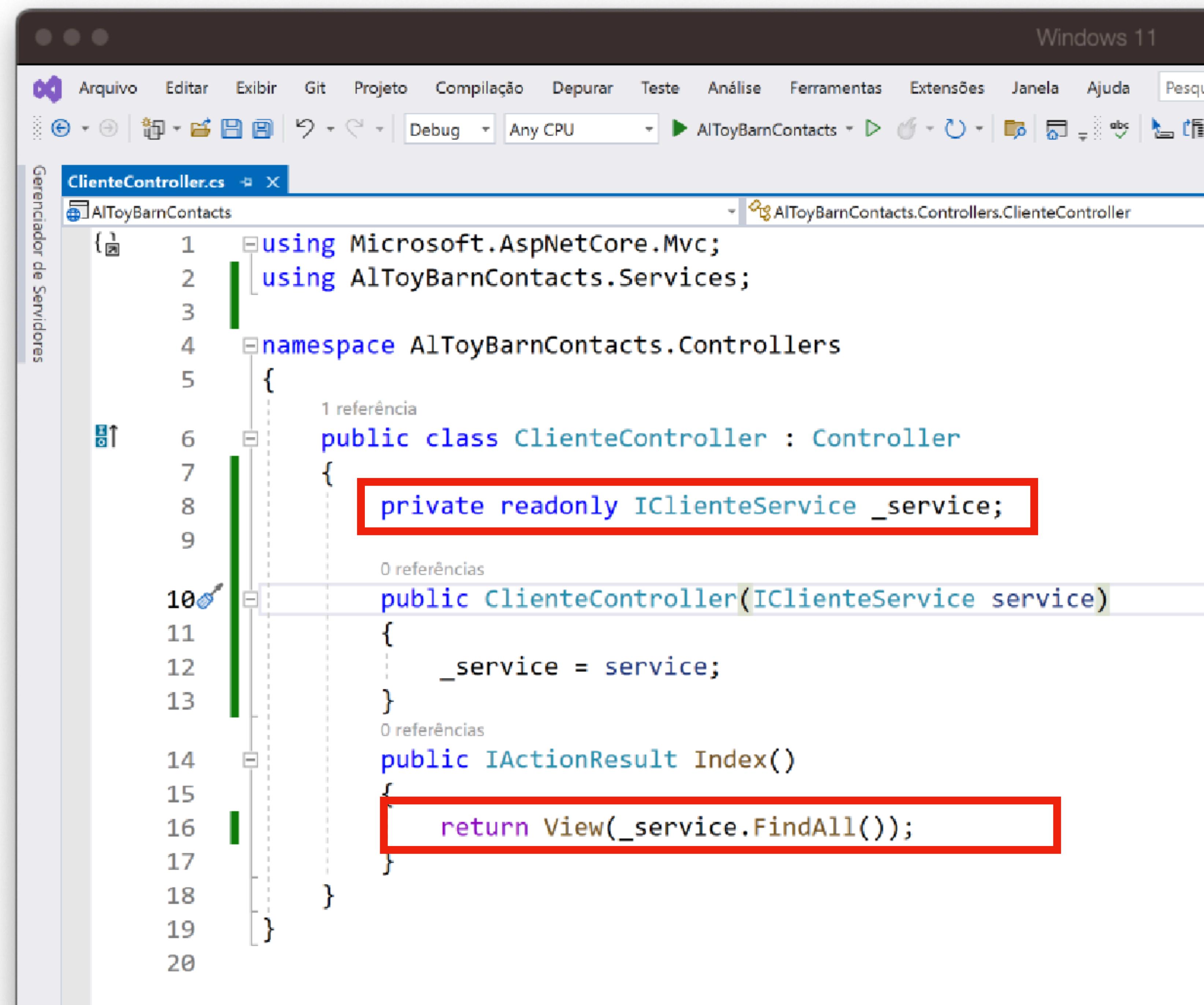
```
1  using AlToyBarnContacts.Models;
2
3  namespace AlToyBarnContacts.Services
4  {
5      public interface ICliente
6      {
7          public void Create(Cliente model);
8          public void Update(Cliente model);
9          public void Delete(Cliente model);
10         public Cliente Find(int id);
11         public ICollection<Cliente> FindAll();
12     }
13 }
14
15 }
```

A vertical green line highlights the opening brace of the interface definition on line 4.

Agora o Controller dependerá da interface.



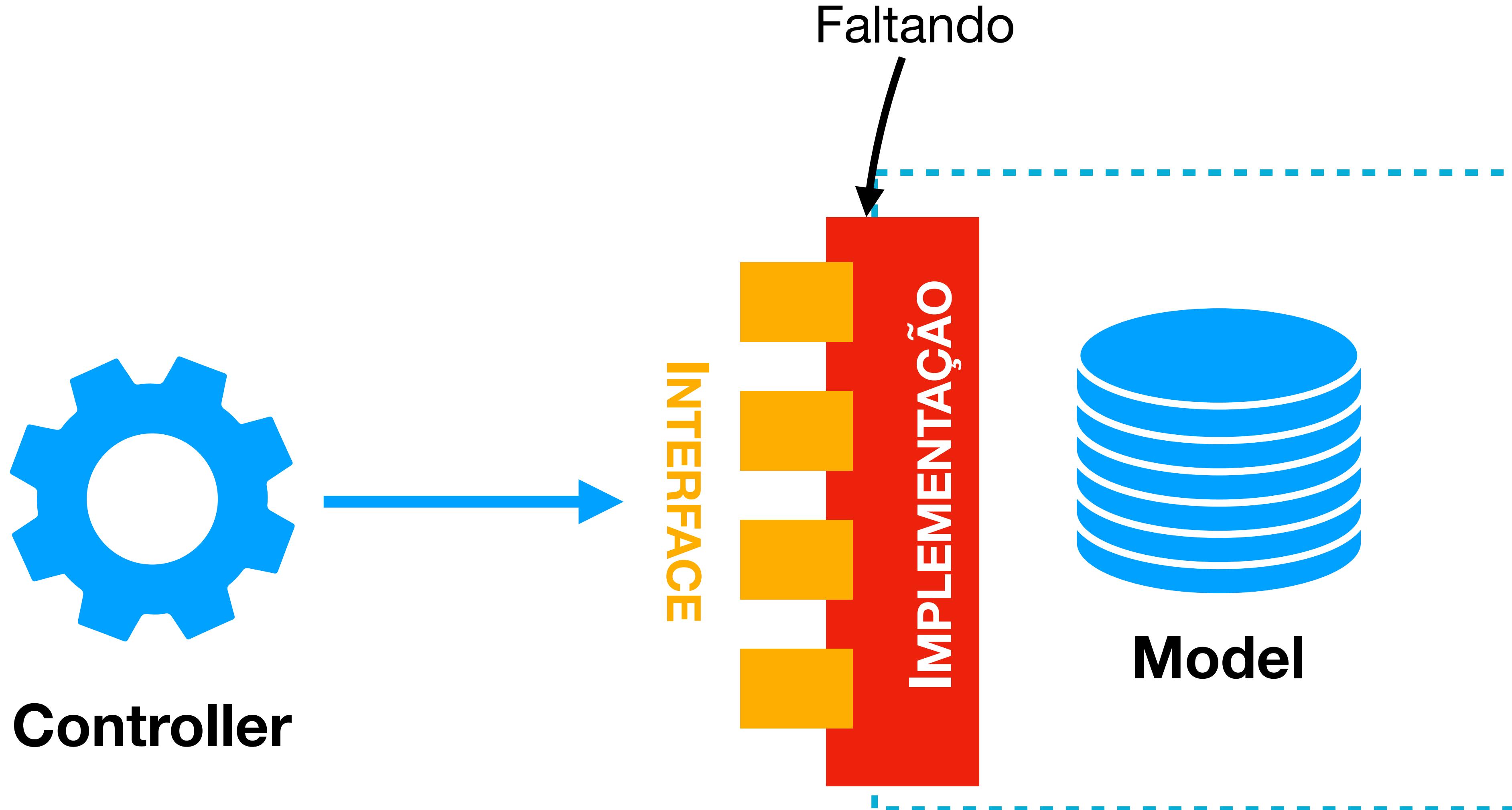
Controller\ClienteController.cs



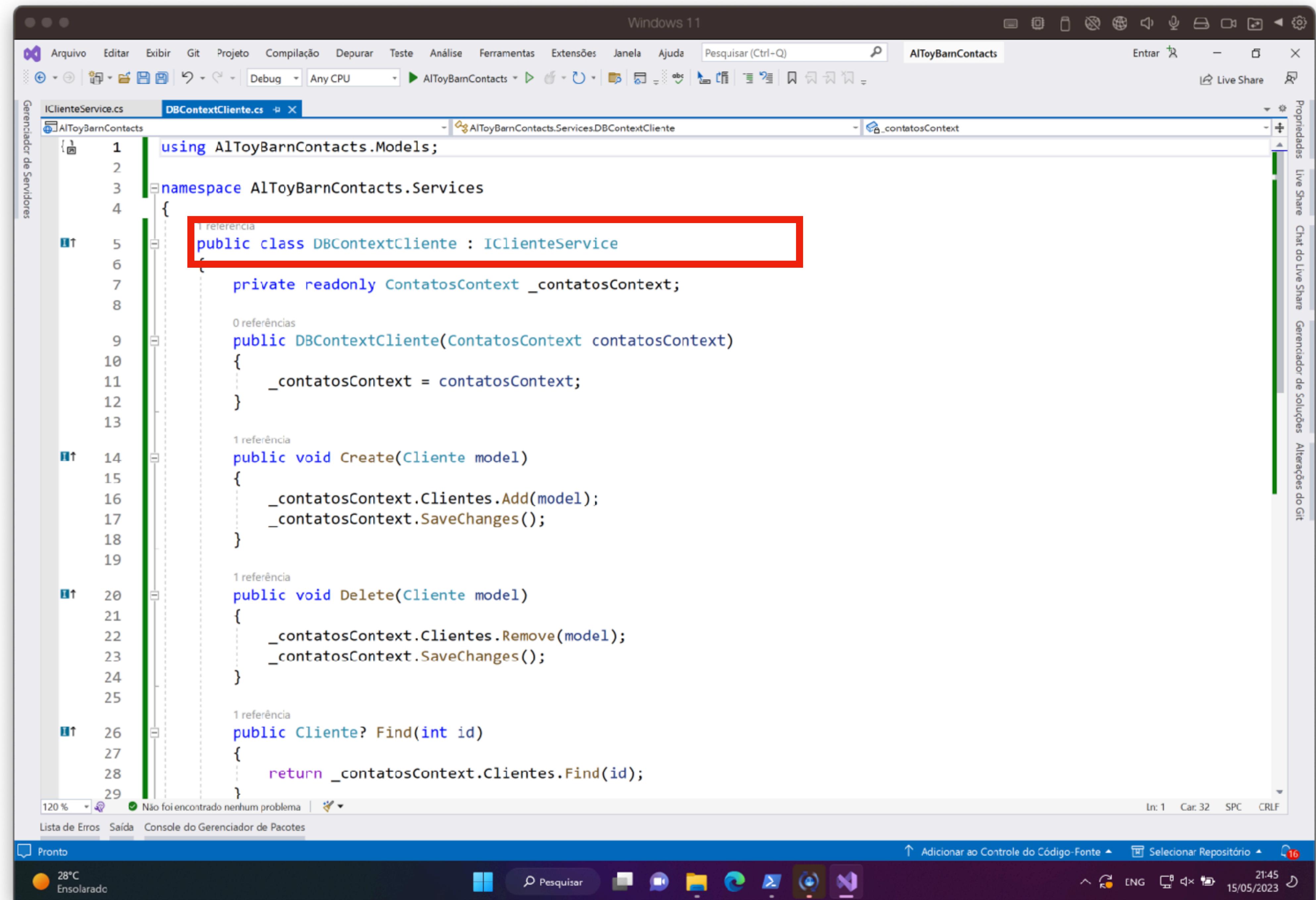
The screenshot shows the Visual Studio IDE interface with the following details:

- Title Bar:** Windows 11
- Menu Bar:** Arquivo, Editar, Exibir, Git, Projeto, Compilação, Depurar, Teste, Análise, Ferramentas, Extensões, Janela, Ajuda, Pesqu.
- Toolbar:** Includes icons for New, Open, Save, Build, Run, and others.
- Project Explorer:** Shows the project structure with "AlToyBarnContacts" selected.
- Solution Explorer:** Shows the file "AlToyBarnContacts.Controllers.ClienteController".
- Code Editor:** Displays the "ClienteController.cs" code. The code uses the Microsoft.AspNetCore.Mvc namespace and references a service from the "AlToyBarnContacts.Services" assembly.
- Annotations:** Two specific lines of code are highlighted with red boxes:
 - Line 8: `private readonly IClienteService _service;`
 - Line 16: `return View(_service.FindAll());`
- Vertical Ruler:** A green vertical bar on the left side of the editor indicates the current line of code being edited.

```
1  using Microsoft.AspNetCore.Mvc;
2  using AlToyBarnContacts.Services;
3
4  namespace AlToyBarnContacts.Controllers
5  {
6      public class ClienteController : Controller
7      {
8          private readonly IClienteService _service;
9
10         public ClienteController(IClienteService service)
11         {
12             _service = service;
13         }
14
15         public IActionResult Index()
16         {
17             return View(_service.FindAll());
18         }
19     }
20 }
```



Services\DbContextCliente.cs



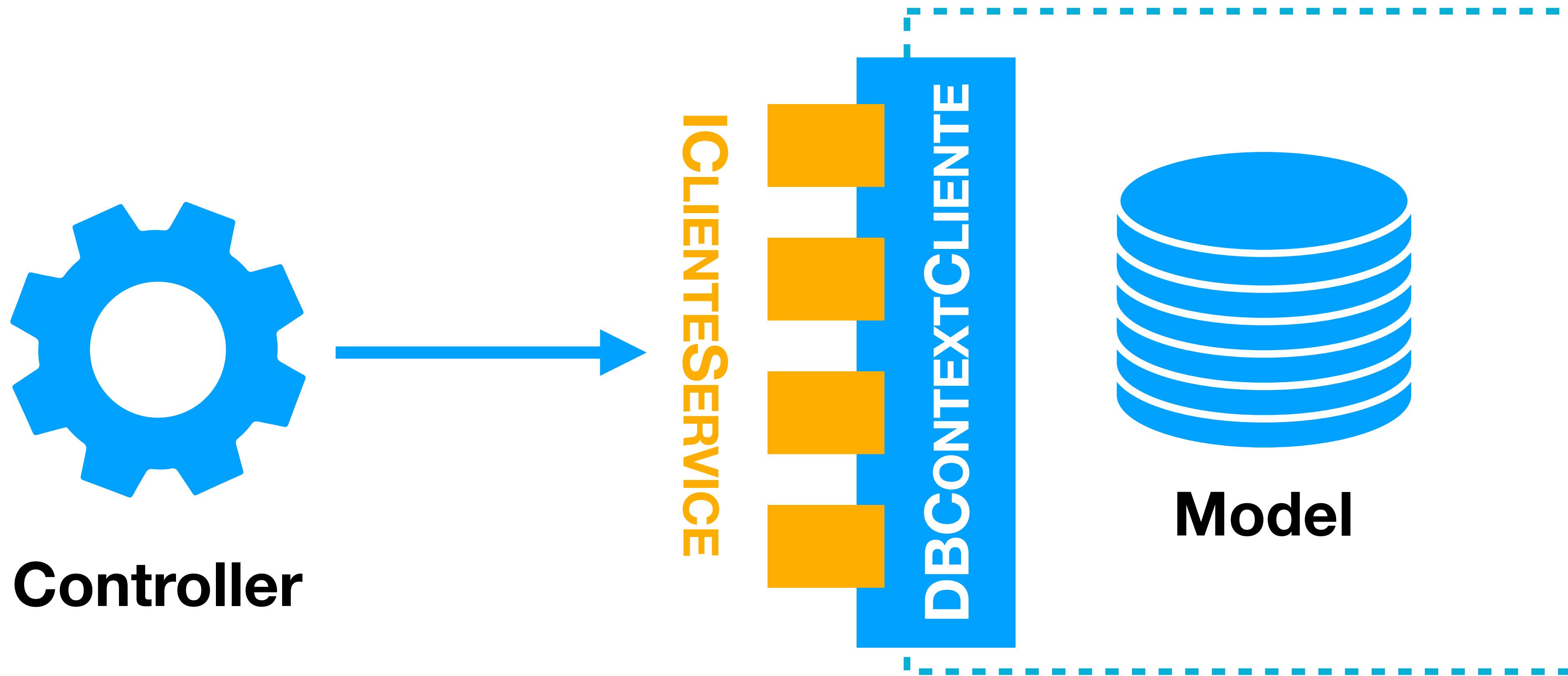
The screenshot shows the Microsoft Visual Studio IDE interface. The title bar indicates it's running on Windows 11. The current project is "AlToyBarnContacts". The code editor displays the file "DbContextCliente.cs" which implements the interface "IClienteService". The class contains methods for creating, deleting, and finding clients. A red box highlights the class definition.

```
1  using AlToyBarnContacts.Models;
2
3  namespace AlToyBarnContacts.Services
4  {
5      public class DbContextCliente : IClienteService
6      {
7          private readonly ContatosContext _contatosContext;
8
9          public DbContextCliente(ContatosContext contatosContext)
10         {
11             _contatosContext = contatosContext;
12         }
13
14         public void Create(Cliente model)
15         {
16             _contatosContext.Clientes.Add(model);
17             _contatosContext.SaveChanges();
18         }
19
20         public void Delete(Cliente model)
21         {
22             _contatosContext.Clientes.Remove(model);
23             _contatosContext.SaveChanges();
24         }
25
26         public Cliente? Find(int id)
27         {
28             return _contatosContext.Clientes.Find(id);
29         }
30     }
31 }
```

The status bar at the bottom shows the following information: 120% zoom, no errors found, and file statistics: In: 1 Car: 32 SPC CRLF.

O **ClienteController** deve receber uma instância de
DBContextCliente, o qual implementa a interface
IClienteService





Princípio da Inversão de Dependência
Dependa de abstrações e não de implementações.

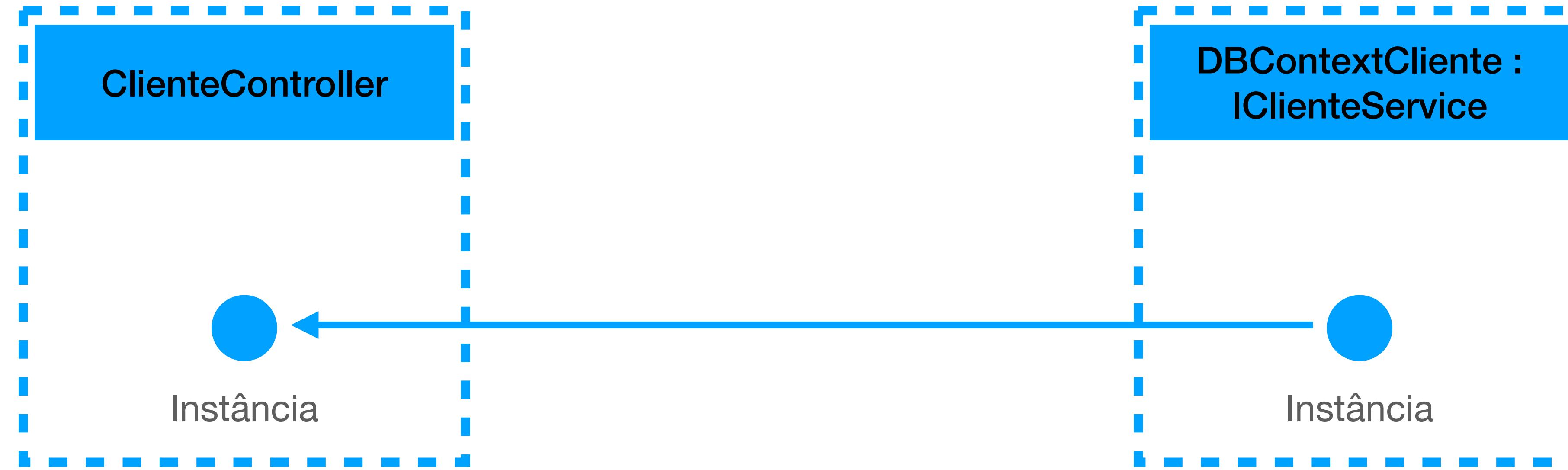
Uma dúvida ...



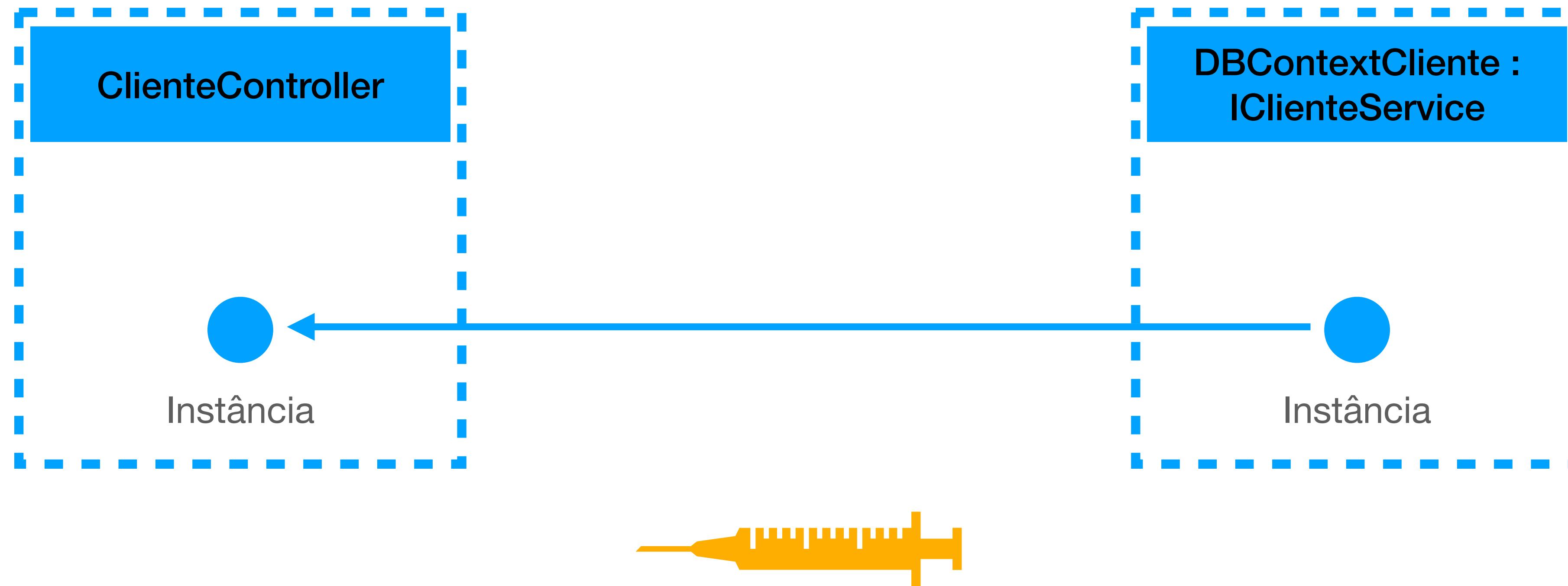
A screenshot of the Microsoft Visual Studio IDE interface. The title bar says "Windows 11". The menu bar includes "Arquivo", "Editar", "Exibir", "Git", "Projeto", "Compilação", "Depurar", "Teste", "Análise", "Ferramentas", "Extensões", "Janela", "Ajuda", and "Pesquisa". The toolbar has icons for file operations like Open, Save, and Print. A status bar at the bottom shows "Any CPU" and "Debug". The main window displays the code for "ClienteController.cs" under the project "AlToyBarnContacts". The code uses dependency injection to inject an instance of `IClienteService` into the controller's constructor.

```
1  using Microsoft.AspNetCore.Mvc;
2  using AlToyBarnContacts.Services;
3
4  namespace AlToyBarnContacts.Controllers
5  {
6      public class ClienteController : Controller
7      {
8          private readonly IClienteService _service;
9
10         public ClienteController(IClienteService service) ←
11         {
12             _service = service;
13         }
14
15         public IActionResult Index()
16         {
17             return View(_service.FindAll());
18         }
19     }
20 }
```

Como o controller vai receber uma
instância do DBContextCliente?



Para a instância de ClienteController ser criada é necessário **FORNECER** uma instância que implementa IClienteService, no caso DBContextCliente



Injeção de Dependência

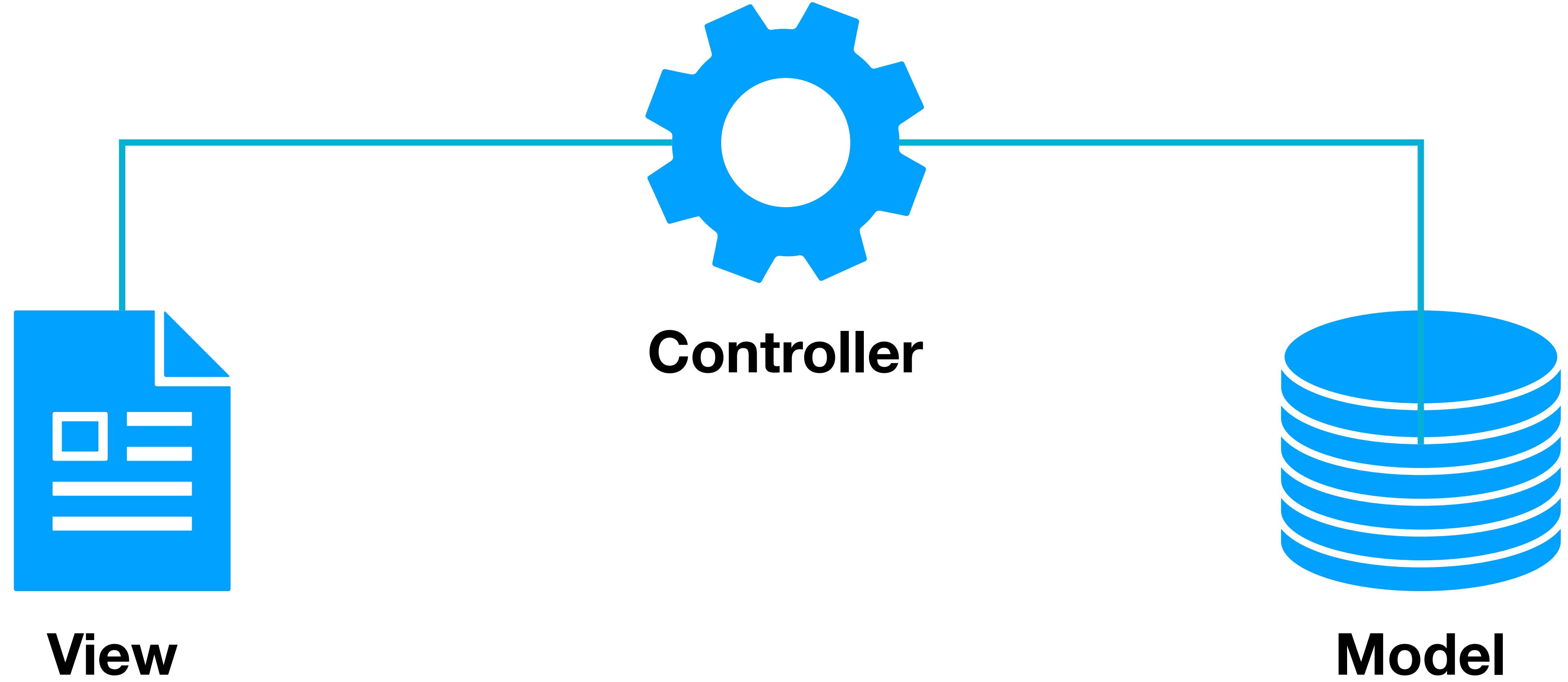
Program.cs

The screenshot shows the Windows 11 desktop environment with the Visual Studio IDE open. The title bar reads "Windows 11". The menu bar includes Arquivo, Editar, Exibir, Git, Projeto, Compilação, Depurar, Teste, Análise, Ferramentas, Extensões, Janela, Ajuda, and Pesquisar (Ctrl+Q). The toolbar below has icons for Undo, Redo, Save, Print, and others. The status bar shows "Any CPU" and "AlToyBarnContacts". The code editor window displays the "Program.cs" file under the project "AlToyBarnContacts". The code uses C# syntax to configure a WebApplication builder, adding services and a database context. A red rectangular box highlights the line "builder.Services.AddScoped<IClienteService, DBContextCliente>();".

```
1  using AlToyBarnContacts.Models;
2  using AlToyBarnContacts.Services;
3  using Microsoft.EntityFrameworkCore;
4
5  var builder = WebApplication.CreateBuilder(args);
6
7  // Add services to the container.
8  builder.Services.AddControllersWithViews();
9
10 builder.Services.AddScoped<IClienteService, DBContextCliente>();
11
12 builder.Services.AddDbContext<ContatosContext>(options =>
13     options.UseSqlServer(builder.Configuration.GetConnectionString("ContatosDatabase"))
14 );
15
16 var app = builder.Build();
```

**Não confunda inversão de dependência
com injeção de dependência.**





**Formam a comunicação visual
(semiótica) do usuário com o
sistema.**

Controller\ClienteController.cs

The screenshot shows the Microsoft Visual Studio IDE interface on Windows 11. The title bar reads "Windows 11". The menu bar includes "Arquivo", "Editar", "Exibir", "Git", "Projeto", "Compilação", "Depurar", "Teste", "Análise", "Ferramentas", "Extensões", "Janela", "Ajuda", and "Pesqu". The toolbar has icons for file operations like Open, Save, and Build. The status bar shows "Debug Any CPU". The current file is "ClienteController.cs" under the project "AlToyBarnContacts". The code editor displays the following C# code:

```
1  using Microsoft.AspNetCore.Mvc;
2  using AlToyBarnContacts.Services;
3
4  namespace AlToyBarnContacts.Controllers
5  {
6      public class ClienteController : Controller
7      {
8          private readonly IClienteService _service;
9
10         public ClienteController(IClienteService service)
11         {
12             _service = service;
13         }
14
15         public IActionResult Index()
16         {
17             return View(_service.FindAll());
18         }
19     }
20 }
```

A red rectangular box highlights the line "return View(_service.FindAll());". A blue arrow points from the text "Refere-se a uma View chamada Index, pois não há nome explicitamente indicado, logo segue-se o nome do método." to this highlighted line.

Refere-se a uma View
chamada Index, pois não há
nome explicitamente
indicado, logo segue-se o
nome do método.

Controller\ClienteController.cs

The screenshot shows the Microsoft Visual Studio IDE interface with the following details:

- Title Bar:** Windows 11
- Menu Bar:** Arquivo, Editar, Exibir, Git, Projeto, Compilação, Depurar, Teste, Análise, Ferramentas, Extensões, Janela, Ajuda, Pesqu.
- Toolbar:** Includes icons for file operations, search, and navigation.
- Project Explorer:** Shows the project structure with "AlToyBarnContacts" selected.
- Toolbox:** Standard .NET development tools.
- Status Bar:** Debug, Any CPU, AlToyBarnContacts, Controller.
- Code Editor:** The file "ClienteController.cs" is open. The code is as follows:

```
1  using Microsoft.AspNetCore.Mvc;
2  using AlToyBarnContacts.Services;
3
4  namespace AlToyBarnContacts.Controllers
5  {
6      public class ClienteController : Controller
7      {
8          private readonly IClienteService _service;
9
10         public ClienteController(IClienteService service)
11         {
12             _service = service;
13         }
14
15         public IActionResult Index()
16         {
17             return View(_service.FindAll());
18         }
19     }
20 }
```

A red rectangular box highlights the entire body of the controller's constructor and its corresponding Index action method. A blue arrow points from the bottom right of this highlighted area towards the explanatory text on the right.

A View recebe uma lista
com todos os clientes

Vamos criar a View Index.cshtml

Windows 11

Arquivo Editar Exibir Gt Projeto Compilação Depurar Teste Análise Ferramentas Extensões Janeia Ajuda Pesquisar (Ctrl+Q) Entrar Live Share

Index.cshtml Program.cs

Gerenciador de Servidores

1 `@*`
2 For more information on enabling MVC for empty projects, visit <https://go.microsoft.com/fwlink/?>
3 `*@`
4 `@{`
5 `}`
6

Gerenciador de Soluções

Solução 'AlToyBarnContacts' (1 de 1 projeto)
AlToyBarnContacts
Connected Services
Dependências
Properties
wwwroot
Controllers
C# ClienteController.cs
C# HomeController.cs
Models
Services
C# D8ContextCliente.cs
C# IClienteService.cs
Views
Cliente
Index.cshtml
Home
Shared
_ViewImports.cshtml
_ViewStart.cshtml
appsettings.json
Program.cs

Propriedades Live Share Chat do Live Share Gerenciador de Soluções Alterações do Git

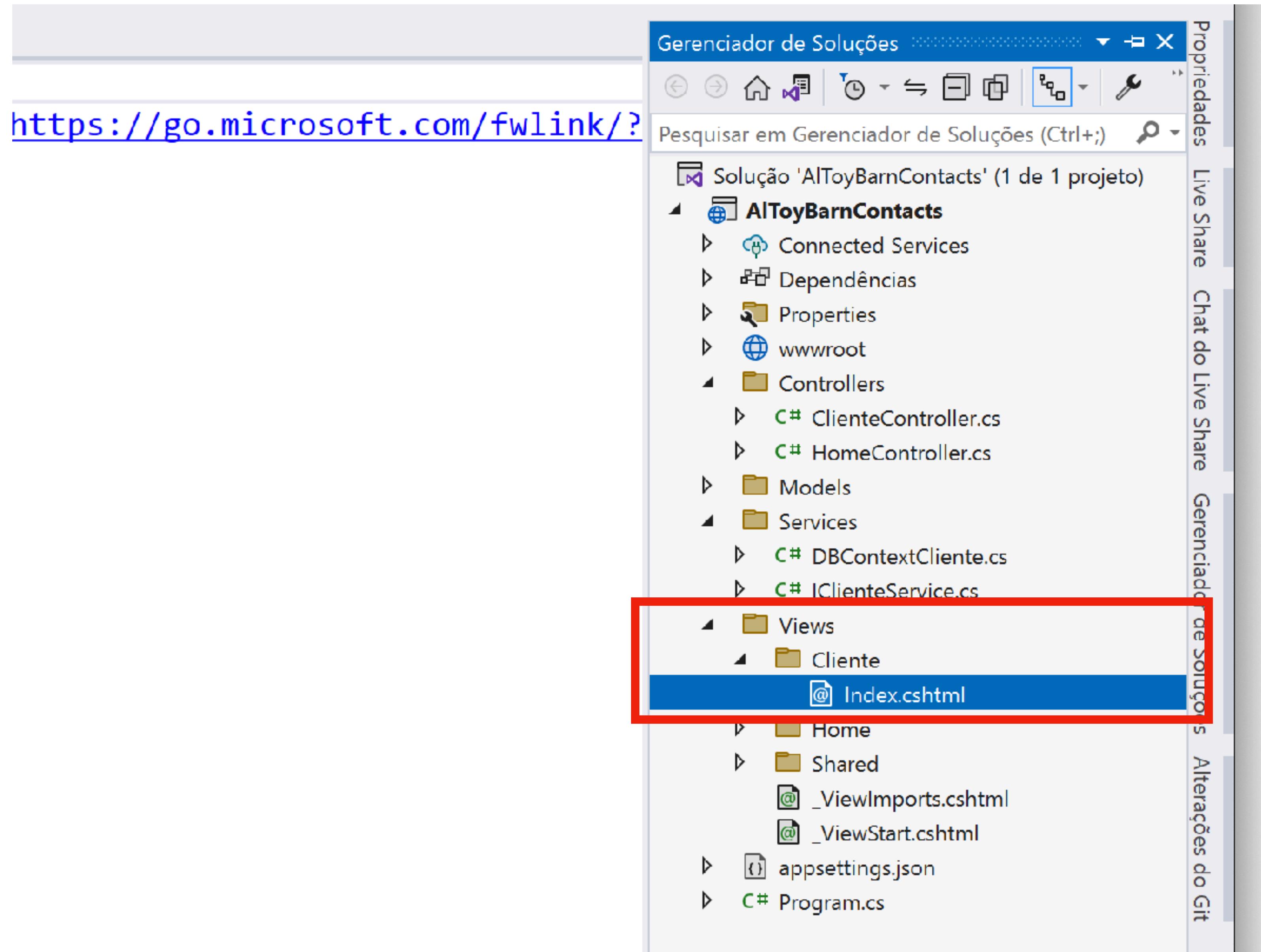
120% Não foi encontrado nenhum problema

Lista de Erros Saída Console da Gerenciador de Pacotes

Front Adicionar ao Controle do Código-Fonte Seleccionar Repositório 16

28°C Ensolarado Pesquisar

ENG 09:03 16/05/2023



Windows 11

Arquivo Editar Exibir Gt Projeto Compilação Depurar Teste Análise Ferramentas Extensões Janeia Ajuda Pesquisar (Ctrl+Q) Entrar Live Share

Index.cshtml* Index.cshtml Program.cs

Gerenciador de Servidores

```
1 @model IEnumerable<Contatos.Models.Cliente>
2 @{
3     ViewData["Title"] = "Home Page";
4 }
5
6 <h1>Clientes</h1>
7
8 <p>
9     <a asp-action="Cria">Novo Cliente</a>
10 </p>
11
12 <table class="table">
13     <thead>
14         <tr>
15             <th>@Html.DisplayNameFor(model => model.Id)</th>
16             <th>@Html.DisplayNameFor(model => model.Nome)</th>
17             <th>@Html.DisplayNameFor(model => model.Endereco)</th>
18             <th>Ações</th>
19         </tr>
20     </thead>
21     <tbody>
22         @foreach (var item in Model)
23         {
24             <tr>
25                 <td>@Html.DisplayFor(modelItem => item.Id)</td>
26                 <td>@Html.DisplayFor(modelItem => item.Nome)</td>
27                 <td>@Html.DisplayFor(modelItem => item.Endereco)</td>
28                 <td>
29                     <a asp-action="Editar" asp-route-id="@item.Id">Editar</a> |
30                     <a asp-action="Detalhar" asp-route-id="@item.Id">Detalhar</a> |
31                     <a asp-action="Remover" asp-route-id="@item.Id">Remover</a>
32                 </td>
33             </tr>
34         }
35     </tbody>
36 </table>
```

Informação recebida via argumento
View(_service.FindAll())

Propriedades Live Share Chat do Live Share Gerenciador de Soluções Alterações do Git

120 % Não foi encontrado nenhum problema

Ln: 15 Car: 2 SPC CRLF

Lista de Erros Saída Console da Gerenciador de Pacotes

Item(ns) salvo(s)

Adicionar ao Controle do Código-Fonte Seleccionar Repositório 16

28°C Ensolarado

Pesquisar

09:30 16/05/2023

Windows 11

Arquivo Editar Exibir Gt Projeto Compilação Depurar Teste Análise Ferramentas Extensões Janeia Ajuda Pesquisar (Ctrl+Q) AlloyBarnContacts Entrar Live Share Propriedades Live Share Chat do Live Share Gerenciador de Soluções Alterações do Git

Index.cshtml* Index.cshtml Program.cs

```
5 <h1>Clientes</h1>
6
7 <p>
8   <a asp-action="Cria">Novo Cliente</a>
9 </p>
10
11 <table class="table">
12   <thead>
13     <tr>
14       <th>@Html.DisplayNameFor(model => model.Id)</th>
15       <th>@Html.DisplayNameFor(model => model.Nome)</th>
16       <th>@Html.DisplayNameFor(model => model.Endereco)</th>
17       <th>Ações</th>
18     </tr>
19   </thead>
20   <tbody>
21     @foreach (var item in Model)
22     {
23       <tr>
24         <td>@Html.DisplayFor(modelItem => item.Id)</td>
25         <td>@Html.DisplayFor(modelItem => item.Nome)</td>
26         <td>@Html.DisplayFor(modelItem => item.Endereco)</td>
27         <td>
28           <a asp-action="Editar" asp-route-id="@item.Id">Editar</a> |
29           <a asp-action="Detalhar" asp-route-id="@item.Id">Detalhar</a> |
30           <a asp-action="Remover" asp-route-id="@item.Id">Remover</a>
31         </td>
32       </tr>
33     }
34   </tbody>
35 </table>
```

120 % Não foi encontrado nenhum problema

Lista de Erros Saída Console da Gerenciador de Pacotes

Item(ns) salvo(s) Adicionar ao Controle do Código-Fonte Seleccionar Repositório 16

28°C Ensolarado Pesquisar ENG 09:29 16/05/2023