

# Dojo-Sage ASP.NET Core 2.2 Entity Framework Core

*Marcos Fabricio Rosa*

*05/06/2019*

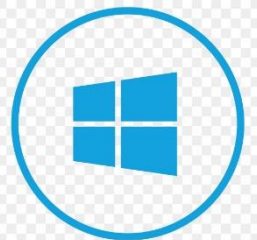
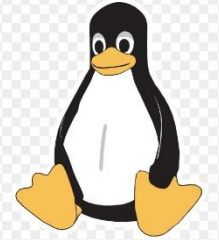
[https://github.com/MarcosRosaSage/curso\\_aspnetcore](https://github.com/MarcosRosaSage/curso_aspnetcore)



# ASP.NET Core – Construído para Web


sage

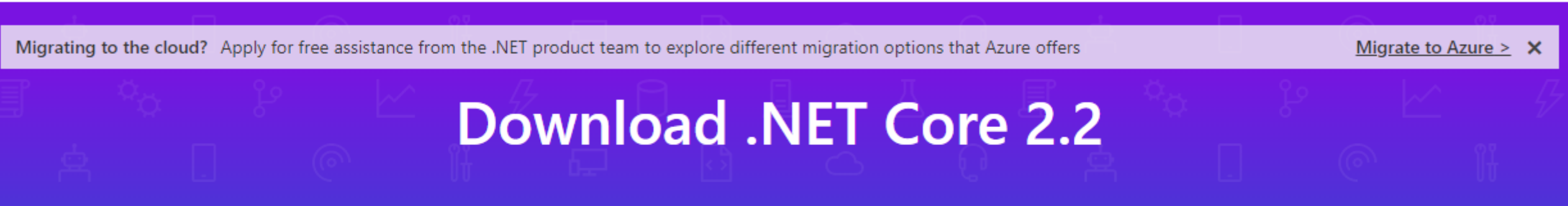
**Open source & cross-platform**  
**MVC, APIs, MicroServices**  
**Rápido, Confiável e Seguro**



<https://dotnet.microsoft.com/download/dotnet-core/2.2>




[.NET](#) > [Downloads](#) > [.NET Core](#) > [.NET Core 2.2](#)

Migrating to the cloud? Apply for free assistance from the .NET product team to explore different migration options that Azure offers
[Migrate to Azure >](#) 



# Download .NET Core 2.2

 Not sure what to download? [See recommended downloads for the latest version of .NET.](#)

Release information	Build apps - SDK 	Run apps - Runtime 
v2.2.5 <div>Current </div>	SDK 2.2.300  Windows	Runtime 2.2.5  Windows

## ASP.NET Core 2.2

**MVC**

**Razor  
Pages**

**Web API**

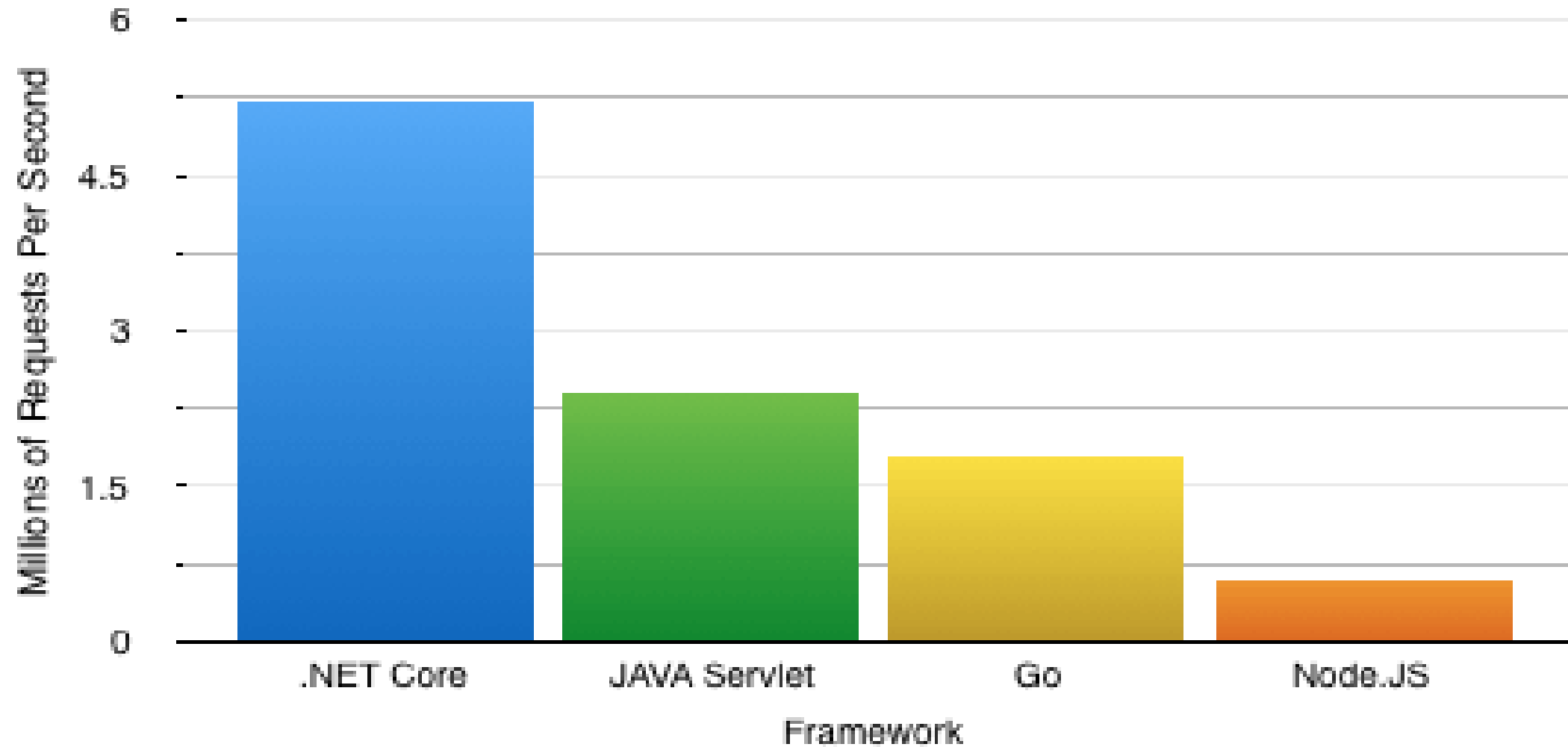
**SignalR**

## .NET Standard

**.NET Framework**

**.NET Core**

# Alto Desempenho

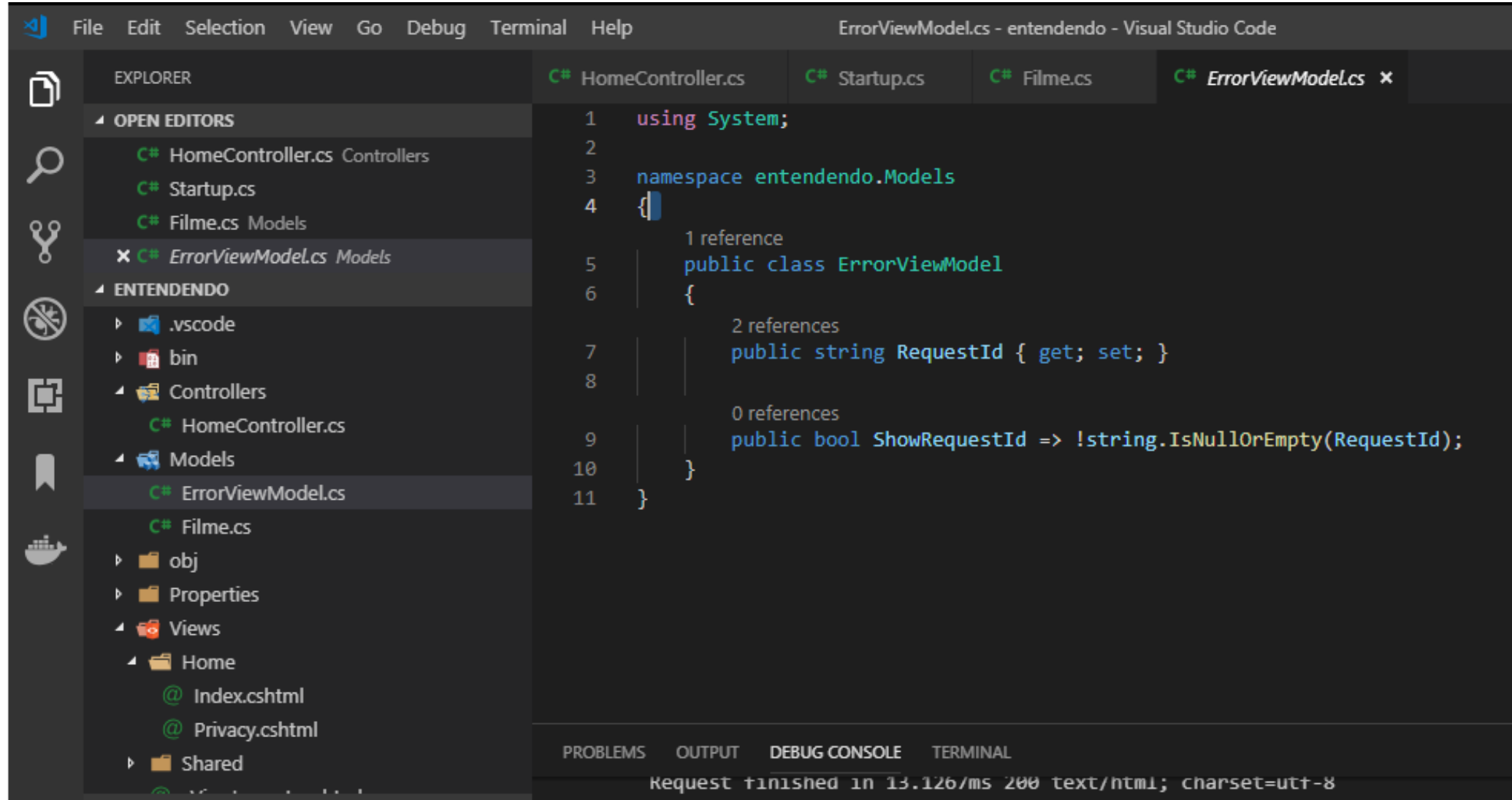


<https://stackify.com/asp-net-core-features/>

# Visual Studio Code

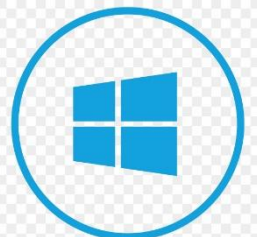
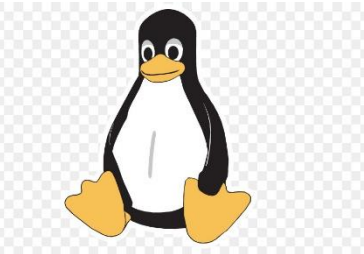
<https://code.visualstudio.com/>

sage



The screenshot displays the Visual Studio Code interface with a dark theme. The Explorer sidebar on the left shows the project structure for 'entendendo', including folders for '.vscode', 'bin', 'Controllers', 'Models', 'Views', and 'Shared'. The 'Models' folder is expanded, showing 'ErrorViewModel.cs' and 'Filme.cs'. The main editor area displays the code for 'ErrorViewModel.cs', which includes a 'using System;' statement, a 'namespace entendendo.Models;' declaration, and a 'public class ErrorViewModel' definition. The class has a 'public string RequestId { get; set; }' property and a 'public bool ShowRequestId => !string.IsNullOrEmpty(RequestId);' method. The bottom status bar shows the output of a request: 'Request finished in 13.126/ms 200 text/html; charset=utf-8'.

```
1 using System;
2
3 namespace entendendo.Models
4 {
5     1 reference
6     public class ErrorViewModel
7     {
8         2 references
9         public string RequestId { get; set; }
10
11         0 references
12         public bool ShowRequestId => !string.IsNullOrEmpty(RequestId);
13     }
14 }
```



# .NET CLI (Interface de linha de comando)



```
C:\aspnetcore>dotnet new
```

```
Uso: new [options]
```

Opções:

```
-h, --help           Exibe a ajuda para esse comando.
-l, --list           Lista os modelos contendo o nome especificado. Se nenhum nome for especificado, listará todos os modelos.
-n, --name           O nome para a saída que está sendo criada. Se nenhum nome for especificado, o nome do diretório atual será usado.
-o, --output         Local para colocar a saída gerada.
-i, --install        Instala uma fonte ou um pacote de modelos.
-u, --uninstall      Desinstala uma fonte ou um pacote de modelos.
--nuget-source       Especifica uma origem do NuGet a ser usada durante a instalação.
--type              Filtra os modelos com base nos tipos disponíveis. Os valores predefinidos são "project", "item" ou "other".
--dry-run           Exibe um resumo do que aconteceria se a linha de comando fornecida fosse executada, se resultasse em uma criação de modelo.
--force             Força o conteúdo a ser gerado, mesmo que isso altere arquivos existentes.
-lang, --language   Filtra os modelos com base na linguagem e especifica a linguagem do modelo a ser criado.
```

Modelos	Nome Curto	Idioma	Marcas
Console Application	console	[C#], F#, VB	Common/Console
Class library	classlib	[C#], F#, VB	Common/Library
Unit Test Project	mstest	[C#], F#, VB	Test/MSTest
NUnit 3 Test Project	nunit	[C#], F#, VB	Test/NUnit
NUnit 3 Test Item	nunit-test	[C#], F#, VB	Test/NUnit
xUnit Test Project	xunit	[C#], F#, VB	Test/xUnit
Razor Page	page	[C#]	Web/ASP.NET
MVC ViewImports	viewimports	[C#]	Web/ASP.NET
MVC ViewStart	viewstart	[C#]	Web/ASP.NET
ASP.NET Core Empty	web	[C#], F#	Web/Empty
ASP.NET Core Web App (Model-View-Controller)	mvc	[C#], F#	Web/MVC
ASP.NET Core Web App	webapp	[C#]	Web/MVC/Razor Pages
ASP.NET Core with Angular	angular	[C#]	Web/MVC/SPA
ASP.NET Core with React.js	react	[C#]	Web/MVC/SPA
ASP.NET Core with React.js and Redux	reactredux	[C#]	Web/MVC/SPA
Razor Class Library	razorclasslib	[C#]	Web/Razor/Library/Razor Class Library
ASP.NET Core Web API	webapi	[C#], F#	Web/WebAPI
global.json file	globaljson		Config
NuGet Config	nugetconfig		Config
Web Config	webconfig		Config
Solution File	sln		Solution



https://www.nuget.org



nuget Packages Upload Statistics Documentation Downloads Blog Sig

## Create .NET apps faster with NuGet

AutoMapper

1.681.198 package versions

20.721.438.988 package downloads

 AutoMapper 8.1.0 

A convention-based object-object mapper.

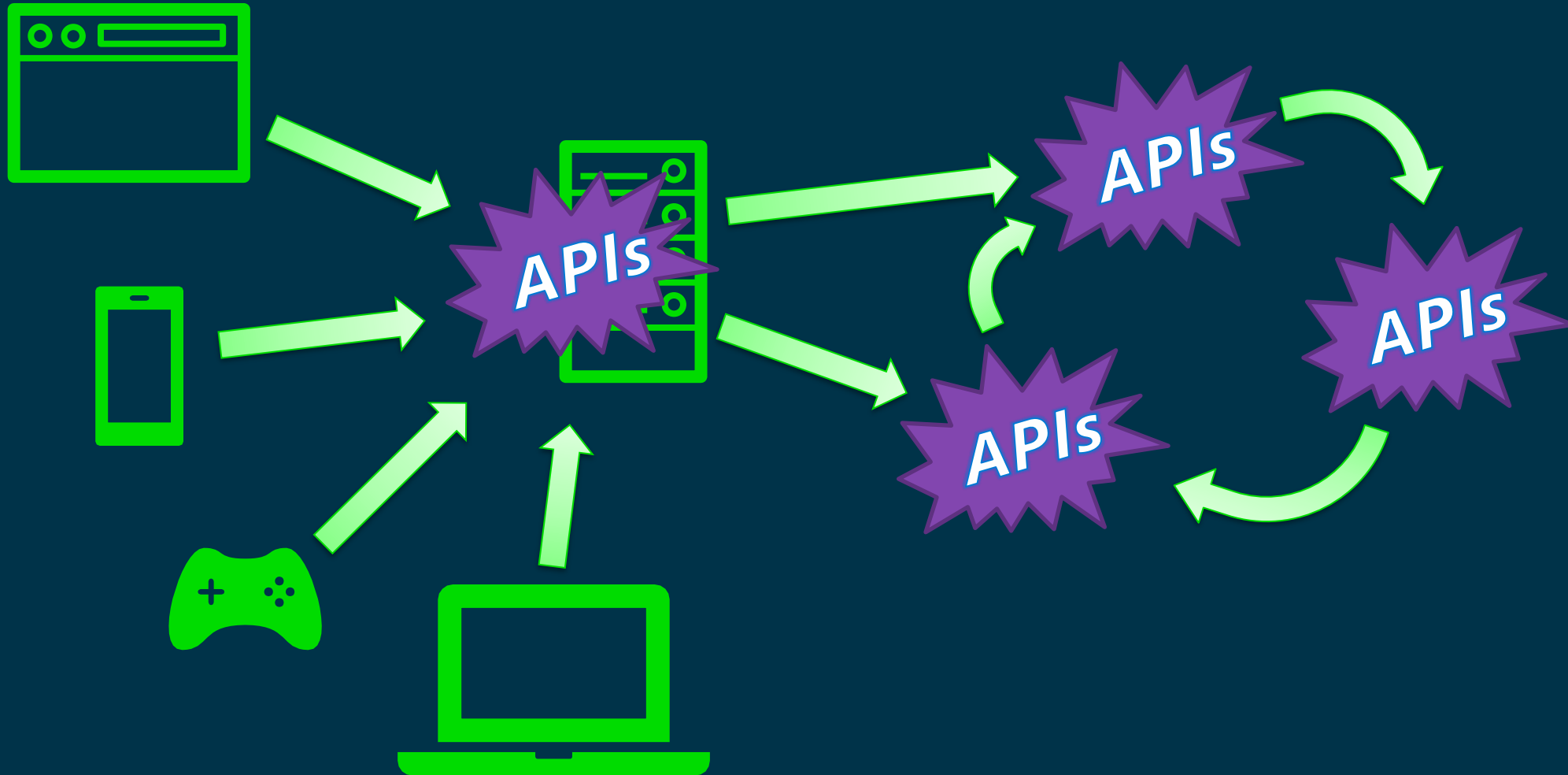
Package Manager **.NET CLI** PackageReference Paket CLI

```
> dotnet add package AutoMapper --version 8.1.0
```



# APIs estão por toda parte!

sage



# VERBOS HTTP



Método	Ação
GET	Responde com informações sobre um recurso específico
POST	Usado para adicionar dados e informações
PUT	Usado para editar dados e informações
DELETE	Usado para remover um registro

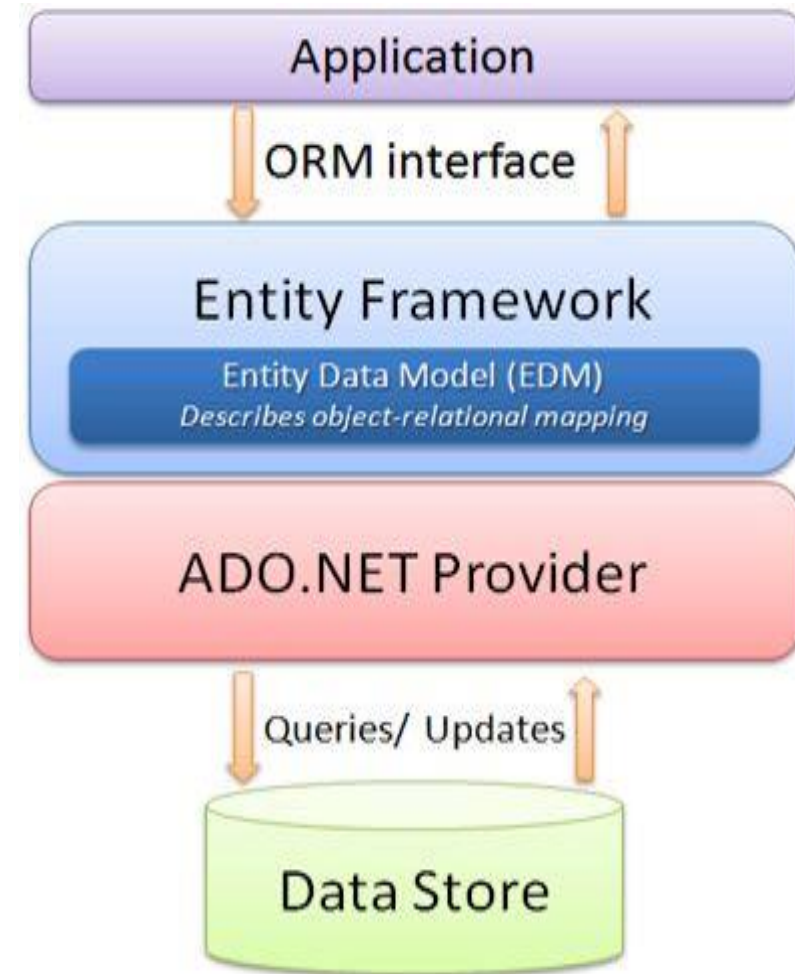
# ROTAS



Method	URI	Name	Action
GET HEAD	/		Closure
GET HEAD	user		App\Http\Controllers\UserController@index
GET HEAD	user/create		App\Http\Controllers\UserController@create
POST	user		App\Http\Controllers\UserController@store
GET HEAD	user/{id}		App\Http\Controllers\UserController@show
GET HEAD	user/{id}/edit		App\Http\Controllers\UserController@edit
PUT	user/{id}		App\Http\Controllers\UserController@update
DELETE	user/{id}		App\Http\Controllers\UserController@delete

# Entity Framework Core

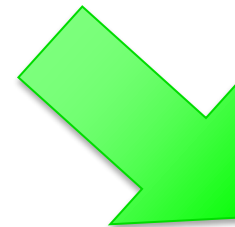
sage



# Entity Framework Core

sage

Entity Framework



Microsoft.EntityFrameworkCore.SqlServer

```
dotnet add package Microsoft.EntityFrameworkCore.SqlServer --version 2.2.4
```



Pomelo.EntityFrameworkCore.MySql 2.2.0

```
dotnet add package Pomelo.EntityFrameworkCore.MySql --version 2.2.0
```

# Banco de Dados (MS-SQL Server e MySQL)

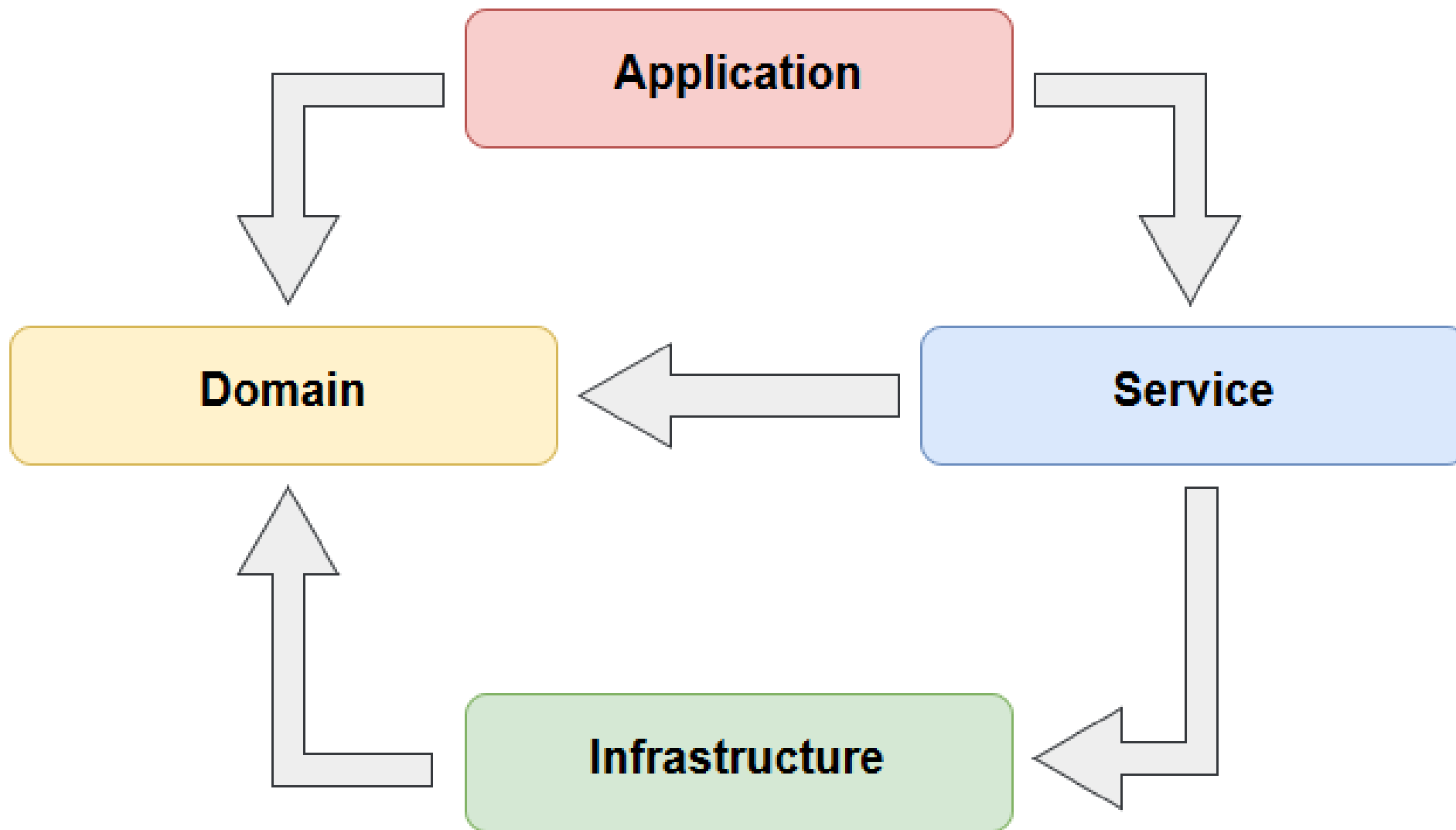
sage



docker

\* Dados do ano fiscal de 2017 que terminam em 30 de setembro de 2017

# Entendendo a Arquitetura Utilizada (Referências) **sage**





# Apresentação da API

# Tipos de Ciclo de Vida – Injeção de Dependencia



<b>TRANSIENTE</b>	Obtém uma nova instância do objeto a cada solicitação
<b>SCOPED</b>	Reutiliza a mesma instancia do objeto durante todo o Request
<b>SINGLETON</b>	Utiliza a mesma instância para toda aplicação ( <b>CUIDADO</b> )

# Obrigado!

[https://github.com/MarcosRosaSage/curso\\_aspnetcore](https://github.com/MarcosRosaSage/curso_aspnetcore)

[https://medium.com/@alexalves\\_85598/criando-uma-api-em-net-core-baseado-na-arquitetura-ddd-2c6a409c686](https://medium.com/@alexalves_85598/criando-uma-api-em-net-core-baseado-na-arquitetura-ddd-2c6a409c686)