



.NET Core
com Entity Framework SQL Server e MySQL
Tratando CORS

Profª Patrícia Gagliardo de Campos
Disciplina: Desenvolvimento para Internet III

Sumário

| | |
|----------------------------------|---|
| 1. Introdução..... | 3 |
| 2. O que é CORS? | 3 |
| 3. Configurando a aplicação..... | 3 |
| 4. Fonte | 5 |

1. Introdução

Na etapa anterior, concluímos os métodos GET, POST, DELETE e PUT da aplicação. O próximo passo será integrar ao *front-end* **React** para que os dados dos alunos que são exibidos na página não sejam dados estáticos, mas dados que sejam obtidos de um banco de dados.

Vamos precisar subir os dois servidores:

- ASP.Net que vai rodar na porta 5000 e 5001 de localhost
- React que vai rodar na porta 300 e 3001 de localhost

Em ambientes de produção, normalmente essas aplicações rodam em máquinas diferentes e, conseqüentemente, possuem endereços distintos para acessá-los. No nosso caso, estamos em um ambiente de desenvolvimento, as duas aplicações rodarão na mesma máquina e teremos que fazer alguns ajustes para contornar o problema de **Cross-origin resource sharing (CORS ou compartilhamento de recursos de origem cruzada)**.

2. O que é CORS?

CORS - Cross-Origin Resource Sharing (Compartilhamento de recursos com origens diferentes) é um mecanismo que usa cabeçalhos adicionais HTTP para informar a um navegador que permita que um aplicativo Web seja executado em uma origem (domínio) com permissão para acessar recursos selecionados de um servidor em uma origem distinta. Um aplicativo Web executa uma requisição *cross-origin* HTTP ao solicitar um recurso que tenha uma origem diferente (domínio, protocolo e porta) da sua própria origem.¹

Acesse os site no final do arquivo e leia mais sobre o CORS. Essa política dos navegadores ocorre para qualquer tipo de aplicação, tanto ASP.Net como Node.js, por exemplo. Aqui, veremos como resolvemos isso no ASP.Net.

3. Configurando a aplicação

Existem várias maneiras de resolver o problema do CORS, mas no nosso caso vamos adicionar na aplicação algumas configurações para que o navegador seja informado e autorize requisições da mesma máquina. Para isso, vamos primeiramente adicionar o serviço de CORS no arquivo **Startup.cs** e na máquina onde está rodando a aplicação está permitindo acesso à aplicação do servidor “x” que é onde está rodando o *front-end* (aplicação em React).

Program.cs

```
using Microsoft.EntityFrameworkCore;
using ProjetoEscola_API.Data;

var builder = WebApplication.CreateBuilder(args);

//Allow CORS
var MyAllowSpecificOrigins = "_myAllowSpecificOrigins";

// Add services to the container.

//Allow CORS
```

¹ Fonte: Cross-Origin Resource Sharing (CORS) - https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Controle_Acesso_CORS

```

builder.Services.AddCors(options =>
{
    options.AddPolicy(MyAllowSpecificOrigins, builder => {
        builder.WithOrigins("http://localhost").AllowAnyOrigin().AllowAnyMethod().AllowAnyHeader();
        builder.AllowAnyOrigin().AllowAnyMethod().AllowAnyHeader();
        builder.SetIsOriginAllowed(origin => new Uri(origin).Host == "localhost");
        builder.SetIsOriginAllowed(origin => true);
    });
});

```

```

builder.Services.AddControllers();
// Learn more about configuring Swagger/OpenAPI at
https://aka.ms/aspnetcore/swashbuckle
builder.Services.AddEndpointsApiExplorer();
builder.Services.AddSwaggerGen();

// Add DbContext
builder.Services.AddDbContext<EscolaContext>(options =>
{
    options.UseSqlServer(builder.Configuration.GetConnectionString("StringConexaoSQLServer"));
});

var app = builder.Build();

// Configure the HTTP request pipeline.
if (app.Environment.IsDevelopment())
{
    app.UseSwagger();
    app.UseSwaggerUI();
}

//app.UseHttpsRedirection();

//Allow CORS
app.UseCors(MyAllowSpecificOrigins);

app.UseAuthorization();

app.MapControllers();

app.Run();

```

4. Fonte

- **Habilitar solicitações entre origens (CORS) em ASP.NET Core**

<https://docs.microsoft.com/pt-br/aspnet/core/security/cors?view=aspnetcore-6.0>

- **Cross-Origin Resource Sharing (CORS)**

https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Controle_Acesso_CORS