

# Atividade 1 — Cliente/Servidor Simples em C#/.NET

Este repositório contém a implementação de uma **API REST em C#/.NET** para gerenciar personagens fictícios, além de instruções para rodar, testar e apresentar evidências conforme solicitado na atividade.

## Objetivo da Atividade

- Compreender a arquitetura cliente-servidor.
- Implementar uma API REST com endpoints básicos (GET, POST).
- Consumir os endpoints via Swagger, VS Code REST Client, Postman ou cliente Console.
- Documentar evidências de funcionamento.

## Estrutura do Projeto

```
/01_Cliente-Servidor Simples_CSharp
├── /APIpersonagens
│   ├── /bin
│   ├── /Controller
│   │   └── PersonagensController.cs
│   ├── /Data
│   │   └── PersonagemStore.cs
│   ├── /Models
│   │   └── APIpersonagens.cs
│   ├── /obj
│   ├── /Properties
│   ├── APIpersonagens.csproj
│   ├── APIpersonagens.http
│   └── Program.cs
├── /ClientePersonagens
│   ├── /bin
│   ├── /obj
│   ├── ClientePersonagens.csproj
│   └── ClientePersonagensAPI.cs
├── /evidencias_imagens
├── /01_Cliente-Servido Simples_CSharp.sln
├── Evidencias.md
└── ReadMe.md
```



## Como rodar a API

```
# na pasta do projeto Web API
dotnet restore
dotnet build
dotnet run
```

A aplicação deve subir em algo como:

```
Now listening on: https://localhost:5066
```

Abra no navegador:

- Swagger: `http://localhost:5066/swagger`
- Endpoint: `http://localhost:5066/personagens`



### Possível erro:

**A path base can only be configured using `IApplicationBuilder.UsePathBase()`**

- Verifique se o `launchSettings.json` não contém caminhos extras (use apenas `http://localhost:PORTA`).
- Zere variáveis de ambiente `ASPNETCORE_URLS` ou `ASPNETCORE_PATHBASE` se estiverem configuradas.



## Endpoints disponíveis

- GET `/personagens` → lista todos os personagens (suporta filtros `universo`, `limit`, `offset`).
- GET `/personagens/{id}` → busca um personagem por ID.
- POST `/personagens` → adiciona novo personagem.

Exemplo de body JSON para o post :

```
{
  "nome": "Satoru Gojo",
  "universo": "Jujutsu Kaisen",
  "poderPrincipal": "Infinito"
}
```

# Testando a API

## 1) Swagger

- Acesse `http://localhost:5087/swagger` .
- Use **Try it out** para executar GET e POST.

## 2) VS Code REST Client

Arquivo APIpersonagens.http

## **\*\* 3) Postman\*\***

- Crie uma Collection “API Personagens”.
- Adicione requests para GET e POST.
- Salve e exporte a Collection (versão 2.1).

## 4) Cliente Console (opcional)

Pode ser implementado em um projeto separado para consumir os endpoints usando `HttpClient` .

## Evidências

As evidências estão em um arquivo separado: [EVIDENCIAS.md](#).

## Conclusão

- API criada com endpoints REST básicos.
- Testada via Swagger, REST Client e Postman.
- Evidências organizadas em `EVIDENCIAS.md` .