



# Criação, documentação e testes unitários de **ASP.NET Web API**

Equipe KPL HUB

# Ferramentas e requisitos

**Visual Studio 2013**

**Nuget (atualizado)**

**Conhecimentos em C#**

**Noções de HTTP e RESTful**

Pacotes:

ASP.NET MVC 4

Swashbuckle

# WEB API

ASP.NET WEB API é um framework .NET para construir serviços HTTP  
Veio para facilitar a criação de aplicações RESTful no framework .NET

**Mais informações:**

<http://www.asp.net/web-api>

[http://www.tutorialspoint.com/restful/restful\\_introduction.htm](http://www.tutorialspoint.com/restful/restful_introduction.htm)

## Swagger

Interface para API RESTful

Swagger basicamente um framework para descrever, visualizar e consumir APIs RESTful

Swashbuckle implementa swagger para Web API

**Mais informações:**

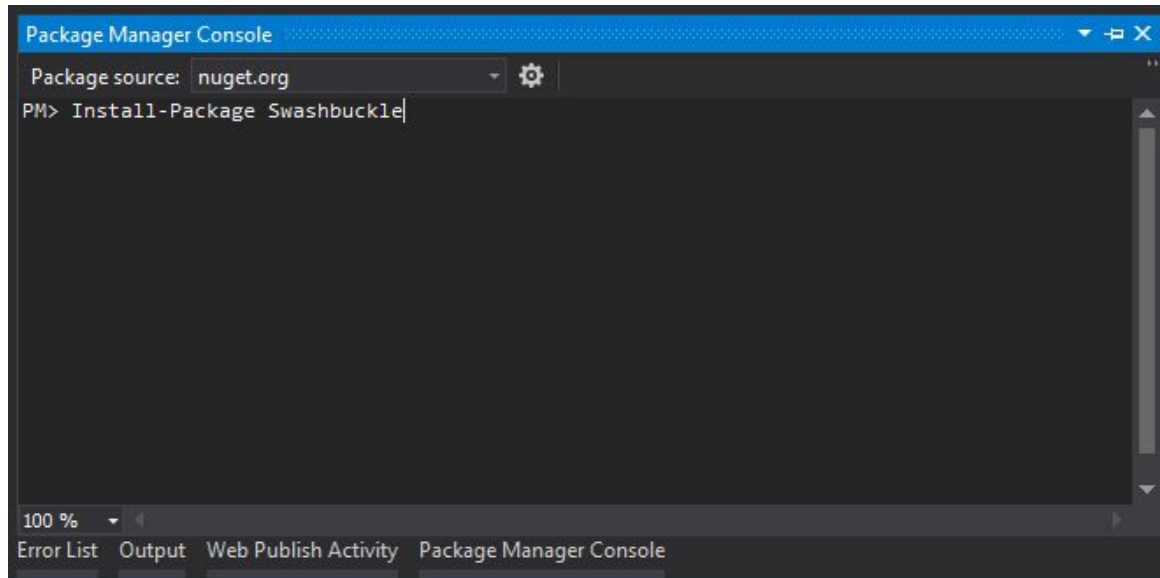
<http://swagger.io/>

<https://github.com/domaindrivendev/Swashbuckle>

# 1. Instalação

- a. Criar Projeto (Web > Asp.net WebApplication > Empty > WebApi > Unit Tests);
- b. Atualizar pacotes NuGet;
- c. Instalar Swashbuckle;
- d. Habilitar documentação XML;
- e. Configurar Swashbuckle;
- f. Criar controller e classes com comentários;
- g. Testar.

# Instalação Swashbuckle



# XML Doc

Output

Output path:

☒ XML documentation file:

☐ Register for COM interop

Generate serialization assembly:

# Configuração

```
public class SwaggerConfig
{
    public static void Register()
    {
        var thisAssembly = typeof(SwaggerConfig).Assembly;
        GlobalConfiguration.Configuration.EnableSwagger(c =>
        {
            c.SingleApiVersion("v1", "WebApiApresentacao");
            c.IncludeXmlComments(ReturnXmlPath());
        }).EnableSwaggerUi(c => { });
    }
    private static string ReturnXmlPath()
    {
        return string.Format(@"{0}\bin\WebApiApresentacao.XML", System.AppDomain.CurrentDomain.BaseDirectory);
    }
}
```

# Class

```
/// <summary>
/// Model Class
/// </summary>
public class ApresentacaoModel
{
    /// <summary>
    /// Constructor
    /// </summary>
    /// <param name="_id">Id</param>
    /// <param name="_value">Value</param>
    public ApresentacaoModel(string _id, string _value)
    {
        id = _id;
        value = _value;
    }

    /// <summary>
    /// Id
    /// </summary>
    public string id { get; set; }
    /// <summary>
    /// The value
    /// </summary>
    public string value { get; set; }
}
```



# Controller

```
/// <summary>
/// Controller Apresentacao
/// </summary>
public class ApresentacaoController : ApiController
{
    /// <summary>
    /// Get all
    /// </summary>
    /// <remarks>Get all classes</remarks>
    /// <response code="200">OK</response>
    /// <response code="400">Bad request</response>
    /// <response code="500">Internal Server Error</response>
    /// <returns>All classes</returns>
    [ResponseType(typeof(ApresentacaoModel[]))]
    public IHttpActionResult Get()
    {
        var retorno = new ApresentacaoModel[] { new ApresentacaoModel("1", "value1"), new ApresentacaoModel("2", "value2") };
        return ResponseMessage(Request.CreateResponse(System.Net.HttpStatusCode.OK, retorno));
    }
}
```

## 2. Customização

- a. Alteração da rota;
- b. Adicionar página customizada;
- c. Alterar página Index.html;
- d. Alterar configurações.

# Rota e configurações

```
public static void Register()
{
    var thisAssembly = typeof(SwaggerConfig).Assembly;

    GlobalConfiguration.Configuration
        .EnableSwagger(c =>
        {
            c.SingleApiVersion("v1", "WebApiApresentacao");
            c.IncludeXmlComments(ReturnXmlPath());
        }).EnableSwaggerUi("Help/{*assetPath}", c =>
        {
            c.CustomAsset("ui", thisAssembly, "WebApiApresentacao.Help.index.html");
        });
}
```



# THANKS!

## Any questions?

You can find me at @username & user@mail.me



## CREDITS

Special thanks to all the people who made and released these awesome resources for free:

- ▶ Presentation template by [SlidesCarnival](#)
- ▶ Photographs by [Death to the Stock Photo](#) ([license](#))