orientações

# git bash

**$ git clone ‘url do repositório do git hub’**

*//Este comando clona o repositório em uma determinada pasta.*

**$ git add .**

*//Comando para adicionar todos os arquivos na fila de comit*

**$ git status**

*//Comando para mostrar os arquivos que foram adicionado ou que sofreram mudanças*

**$ git commit -m ‘Starting development in REST’**

*//Comando para salvar commit dando o nome conforme descrição acima entre aspas simples*

**$ git push**

*//Comando para enviar atualizações para o repositório no git hub*

# git ignore file

Este arquivo serve para impedir que arquivos de compilação sejam salvos no repositório do git hub. Para obter este arquivo, basta acessar o link <https://raw.githubusercontent.com/github/gitignore/master/VisualStudio.gitignore>, copiar seu conteúdo, colar em um bloco de notas e salva-lo com o nome “.gitignore” na pasta raiz do projeto.

# git hub credentials

**$ git config –-global credential.helper wincred**

*Comando para salvar as credenciais no windows. Assim não é necessário entrar com as crdenciais toda vez que é executado o comando* ***git push****.*

*Obs. É necessário que o git bash esteja em modo Adm.*

# dependências para persistencia de dados

**Pomelo.EntityFrameworkCore.MySql**

*Dependência para implementar o MySql na aplicação.*

# versionamento de endpoints

**Microsoft.AspNetCore.Mvc.Versioning**

*Dependência necessária para o versionamento. O link para mais detalhes é o seguinte:* [*https://github.com/Microsoft/aspnet-api-versioning/wiki/versioning-via-the-url-path#aspnet-core*](https://github.com/Microsoft/aspnet-api-versioning/wiki/versioning-via-the-url-path#aspnet-core)*.*

# migrations

**Evolve**

**MySql.Data**

*Haverá no fim das configurações erro de ambiguidade devido a dependência adicionada anteriormente chamada Pomelo....*

*Basta abrir o arquivo da solução.csproj e adicionado o código abaixo:*

*<Target Name="ChangeAliasesOfStrongNameAssemblies" BeforeTargets="FindReferenceAssembliesForReferences;ResolveReferences">*

*<ItemGroup>*

*<ReferencePath Condition="'%(FileName)' == 'MySqlConnector'">*

*<Aliases>MySqlConnectorAlias</Aliases>*

*</ReferencePath>*

*</ItemGroup>*

*</Target>*

# contente negotiation

**Microsoft.AspNetCore.Mvc.Formatters.Xml**

*Dependência necessária para retorno de requisição em formato XML*

services.AddMvc(options => {

options.RespectBrowserAcceptHeader = true;

options.FormatterMappings.SetMediaTypeMappingForFormat("xml", MediaTypeHeaderValue.Parse("text/xml"));

options.FormatterMappings.SetMediaTypeMappingForFormat("json", MediaTypeHeaderValue.Parse("application/json"));

})

.AddXmlSerializerFormatters()

# provedor de hyperlink nas requisições

**Tapioca.HATEOAS**

*Dependência necessária para prover hyperlinks nas requições como por exemplo, em um usuário ter um hyperlink para update de dados*