

Curso/Oferta: Arquitetura de Software Distribuído – Oferta 11

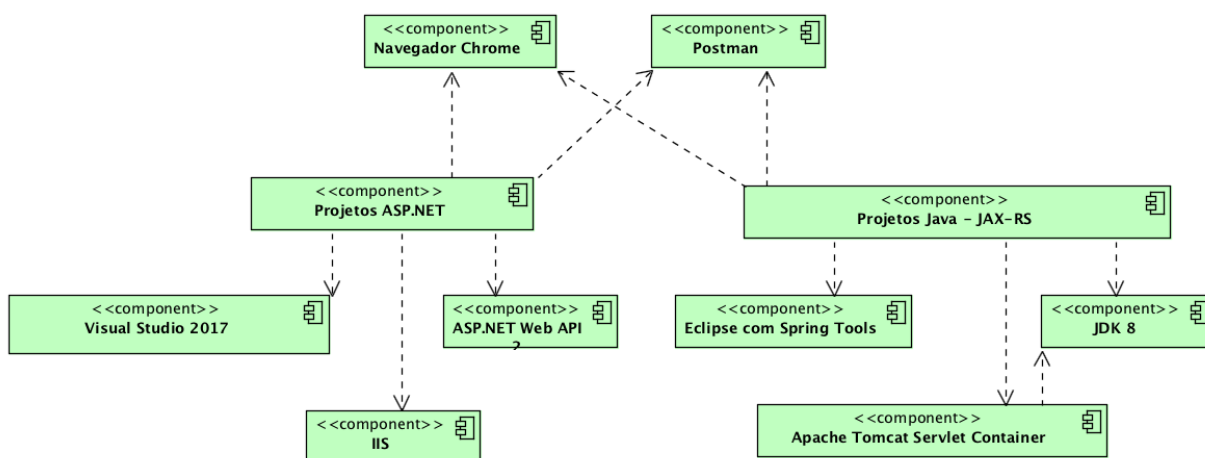
Disciplina: Arquitetura de Backend e Microserviços

Professor: Marco Mendes

Data: 04/04/18

Atividade Aula 02

Nesta aula iremos explorar o **estilo arquitetural de APIs** nas tecnologias Java EE e .NET. O seu ambiente de desenvolvimento, disponibilizado em uma máquina virtual, contará com os seguintes componentes arquiteturais representados no diagrama UML abaixo mostrado.



Estudo Dirigido

O seu objetivo é implementar uma API para a gestão de uma livraria virtual. Uma livraria virtual deve ter pelo menos funcionalidades para:

- Postar comentários para livros. (resenhadas)
- Pesquisa de livros por critérios diversos.
- Manipular um carrinho de compras
- Realizar pedidos
- Acompanhamento o status das entregas realizadas.
- Cadastrar livros

Como benchmarking, explore o sítio da Amazon aqui - <https://www.amazon.com.br>

A sua API deve seguir as 11 regras apresentadas no capítulo 3 da apostila do curso (ver as páginas 36 a 38)

Antes de avançar para a implementação do seu código, faça o desenho da sua API apropriadamente.

A sua implementação deve ser realizada em ASP.NET Web API/C# ou JAX-RS Java.

Por simplicidade, você não deve se preocupar com persistência de banco de dados ou regras de negócio. Você pode simular o funcionamento através de mocks.

E se você ainda não conhece as tecnologias básicas de montagem de API em Java ou .NET, não se preocupe. Teremos roteiros nos apêndices A e B para você se orientar.

Entregáveis da aula de hoje

- Desenho da sua API
- Código fonte (Java ou .NET) da API da gestão de livros publicado no GitHub.
- Justificativas que as 11 regras apresentadas no capítulo 3 foram observadas.

Poste no nosso ambiente o texto da sua entrega juntamente com o link para o seu código no GitHub.

Execute o apêndice A para se inteirar de como operar com a tecnologia ASP.NET Web API

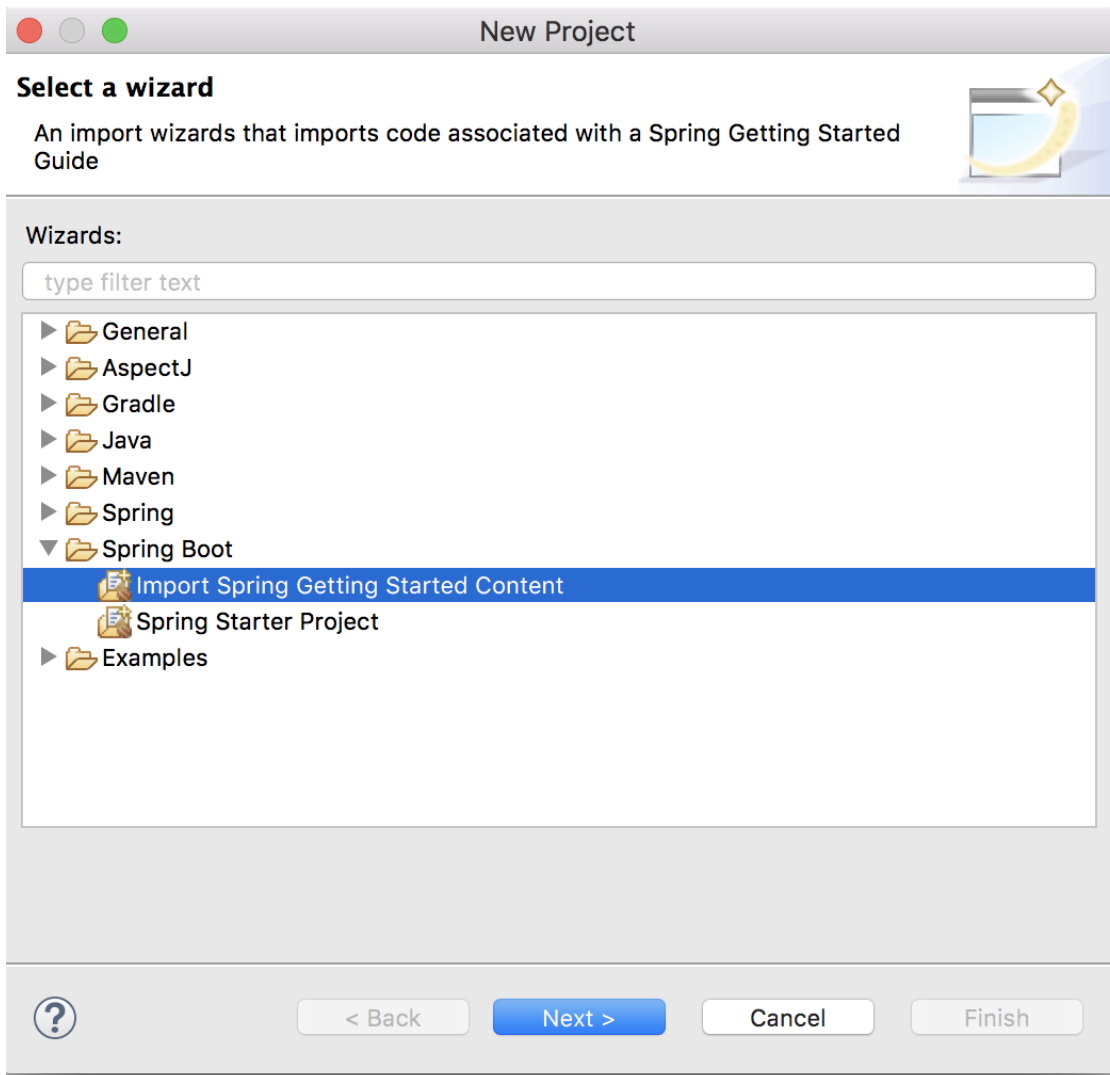
Apêndice A - Fundamentação –ASP.NET Web API

1. Execute o tutorial disponibilizado em
<https://docs.microsoft.com/pt-br/aspnet/core/web-api/?view=aspnetcore-2.1> ou
<https://docs.microsoft.com/pt-br/aspnet/core/web-api/?view=aspnetcore-2.2>
2. Este tutorial apresenta o passo da criação de uma aplicação com uma API mínima a tecnologia ASP.NET Core 2.1.
3. Vamos agora observar como uma API pode ser publicada com o uso do Swagger e o plugin Swashbuckle dentro do Visual Studio. Para isso execute o tutorial disponibilizado em
<https://docs.microsoft.com/pt-br/aspnet/core/tutorials/getting-started-with-swashbuckle?view=aspnetcore-2.1&tabs=visual-studio-code%2Cvisual-studio-xml>

Execute o apêndice B para se inteirar de como operar com a tecnologia JAX-RS em Java

Apêndice B - Fundamentação – JAX-RS

Após abrir o Eclipse, abra um novo projeto e selecione a opção mostrada abaixo.



Clique na opção “Import Spring Getting Started Content”. A tela a seguir será mostrada.

Import Getting Started Content

Type pattern to match

- Reactive Rest Service
- Register Facebook App
- Register Twitter App
- Relational Data Access
- Rest Hateoas
- Rest Service**
- Rest Service Cors
- Routing And Filtering
- Scheduling Tasks
- Securing Web
- Service Registration And Discovery
- Serving Web Content
- Spring Boot
- Spring Boot Docker

Building a RESTful Web Service :: Learn how to create a RESTful web service with Spring. :: spring-boot

Build Type

☒ Maven ☐ Gradle (Buildship 1.x) ☐ Gradle (Buildship 2.x)
☐ Gradle (STS) ☐ General

Code Sets

☒ initial ☒ complete

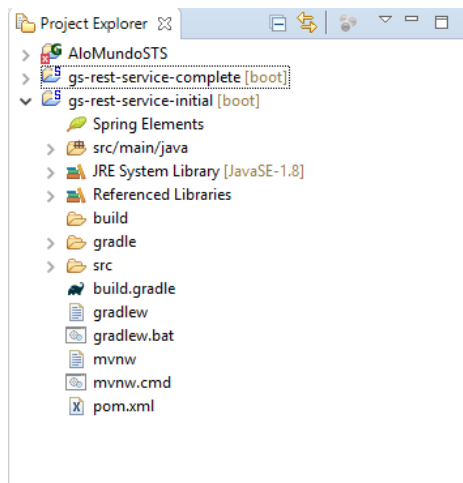
Home Page

<https://spring.io/guides/gs/rest-service> ☒ Open

? < Back Next > Cancel Finish

Esta tela permite que você possa escolher projetos exemplos para explorar conceitos Java EE e em particular neste laboratório conceitos JAX-RS (RESTful).

4. Importe o projeto exemplo chamado Rest-Service a partir da tela acima. Este projeto irá criar um projeto completo (gabarito) e um projeto vazio na área do explorador de projetos, conforme a figura acima.



Após importado o projeto, siga os passos do guia que será exibido na área central da tela <https://spring.io/guides/gs/rest-service/>

Apêndice C – Introdução ao GitHub

Execute esse tutorial para aprender o básico da operação do Git com o repositório do GitHub