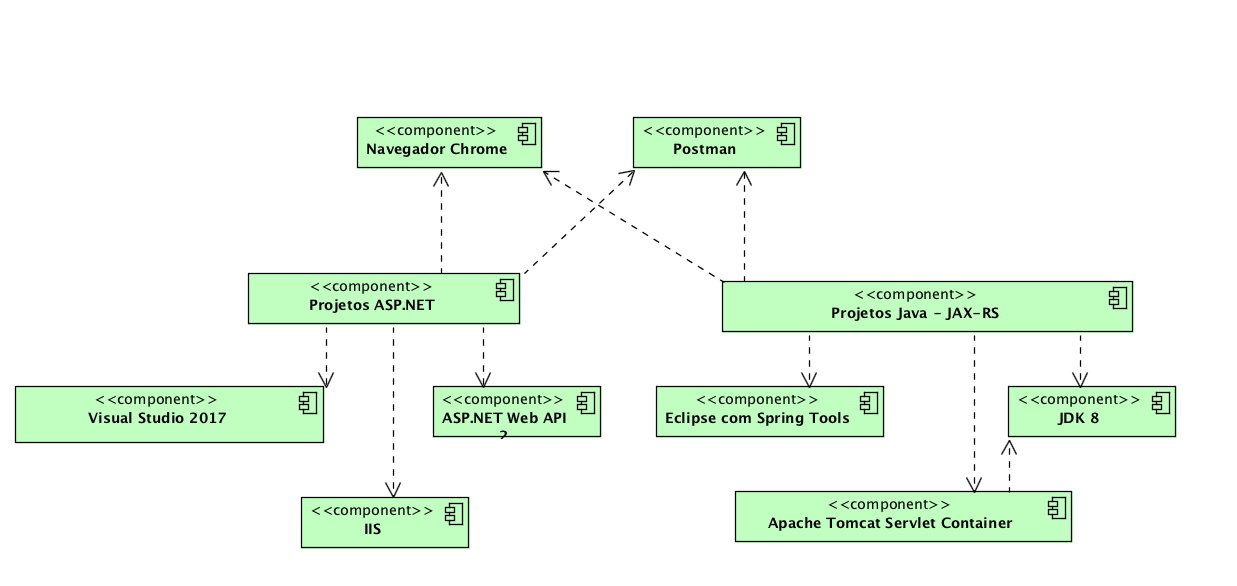
|  |  |
| --- | --- |
| **_Brasão DEC** | **PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**  **PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO**  **DIRETORIA DE EDUCAÇÃO CONTINUADA** |

**­Curso/Oferta:** Desenvolvimento de Web Full Stack  
**Disciplina:** API e Web Services   
**Professor:** Marco Mendes   
**Data:** 23/05/19

# Atividade Aula 02

**PARTE 1 – Laboratório**

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. (Livro Texto Arquitetura de Sistemas Web)

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

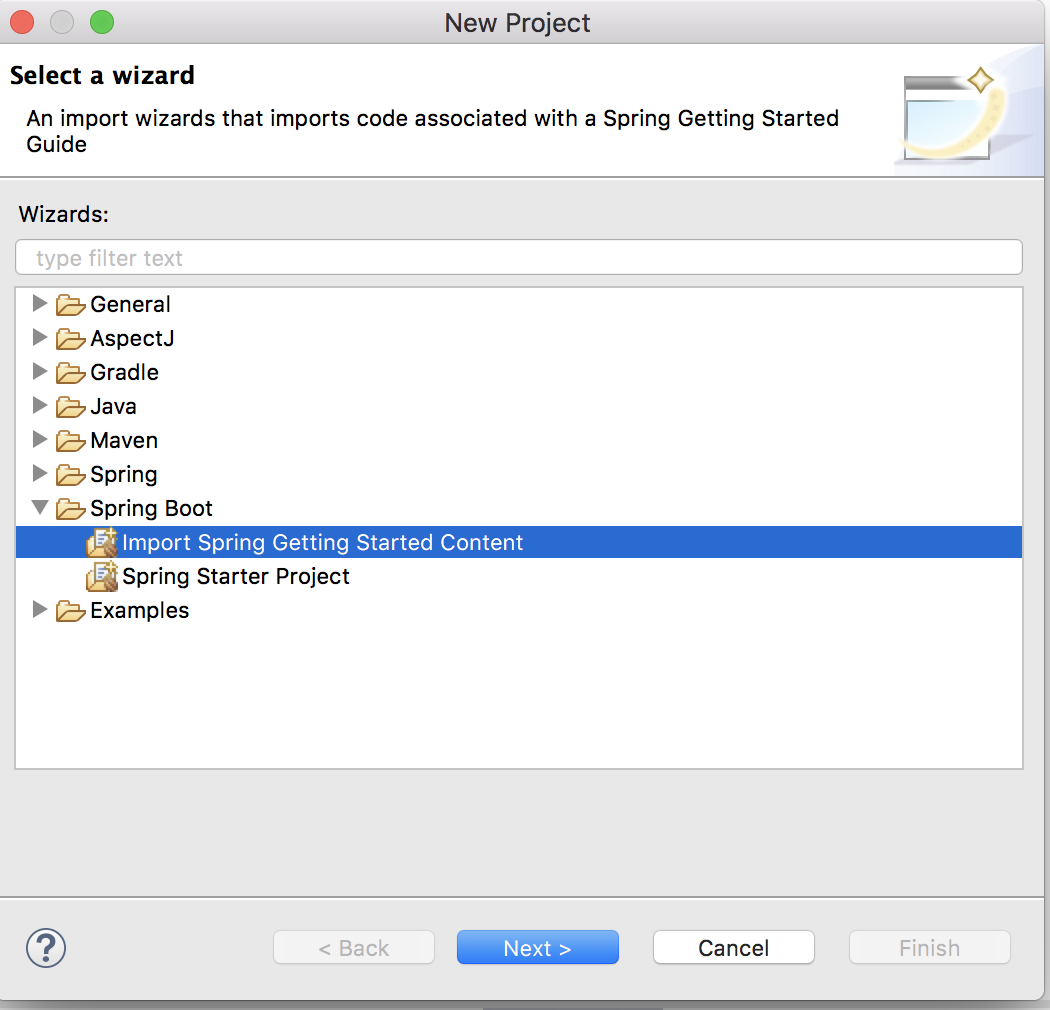
**Apêndice A - Fundamentação –ASP.NET Web API**

1. Execute o tutorial disponibilizado em https://docs.microsoft.com/pt-br/aspnet/web-api/overview/getting-started-with-aspnet-web-api/tutorial-your-first-web-api. Este tutorial apresenta o passo da criação de uma aplicação com uma API mínima a tecnologia ASP.NET Core.
2. 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-2.2>

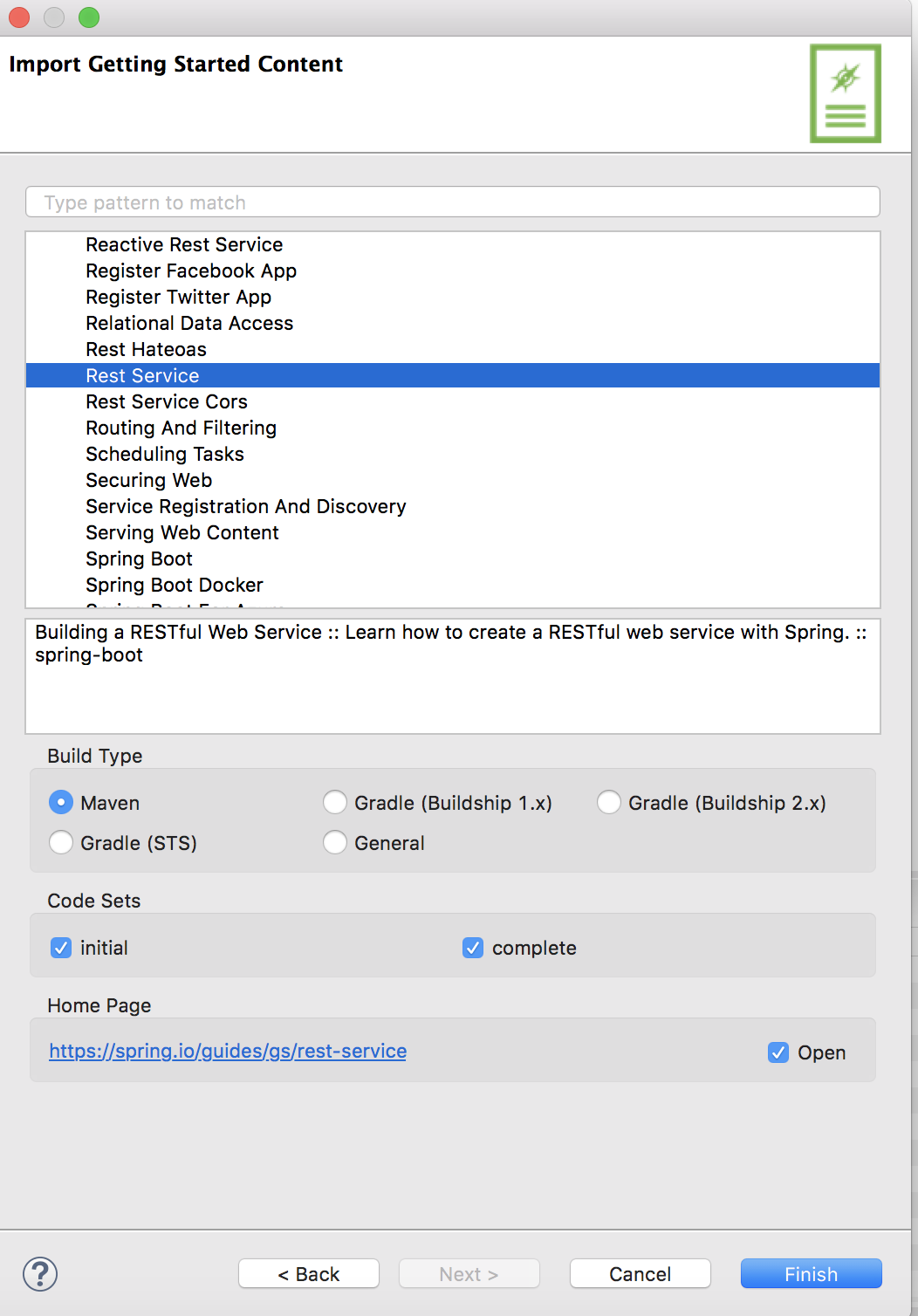
Execute o apêndice B para se inteirar de como operar com a tecnologia JAX-RS em Java

**Apêndice B - Fundamentação – JAX-RS**

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

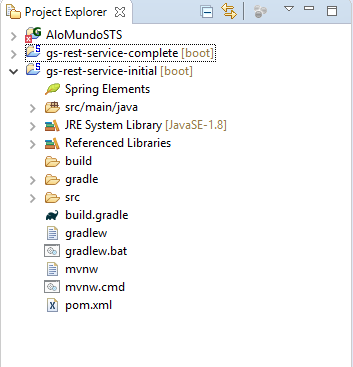


Clique na opção “Import Spring Getting Started Content”. A tela a seguir será mostrada.



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

1. 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/>

**PARTE 2 – Questões de Fixação**

**Questões de fixação sobre HTTP**

1. Que métodos HTTP podem ser usados para suportar a inserção de objetos no servidor HTTP? Justifique a sua resposta.
2. O que é idempotência em métodos HTTP? Qual o seu efeito prático no desenho de apli- cações?
3. Quais são os métodos HTTP que suportam cache?
4. Qual a diferença entre o método POST e o método PUT no protocolo HTTP?

**Questões sobre estilos arquiteturais de API**

1. O que é o estilo arquitetural de APIs? E que vantagens ele oferta para organizações?
2. Forneça três exemplos de tecnologias que podem ser usadas para implementar o conceito de APIs.
3. Descreva, em suas palavras, o modelo de maturidade de Richardson.