



High performance. Delivered.

# Practical Application Development with Spring BOOT

Module 2:Spring Boot - Componentes





- Revisão do Spring Essentials
- Spring BooT Overview
- Spring BooT Introduction;
- Spring BooT Components;
- Spring BooT Features;
- Funcionamento;
- Construção de aplicação;



## **Module Objectives**



- Ao final deste modulo você saberá:
  - O que é Spring Boot 2.0.
  - Instalar o STS.
  - Desenvolver uma pequena aplicação web.



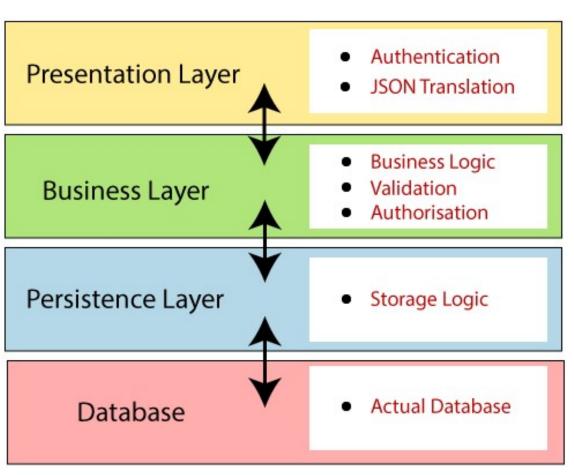
### **Spring Boot**

## Spring Boot

2448696768207065726661726d616e63652e20444656c6976657265642e27486967682070657265672667726667726667726667726667726772677267726772677266772







A camada de apresentação lida com as solicitações HTTP, converte o parâmetro JSON em objeto e autentica a solicitação e a transfere para a camada de negócios.

A camada de negócios lida com toda a lógica de negócios. Ele consiste em classes de serviço. Ele também executa autorização e validação.

A camada de persistência contém toda a lógica de armazenamento e converte os objetos de negócios de e para as linhas do banco de dados.

Na camada de banco de dados, as operações CRUD (criar, recuperar, atualizar, excluir) são realizadas.



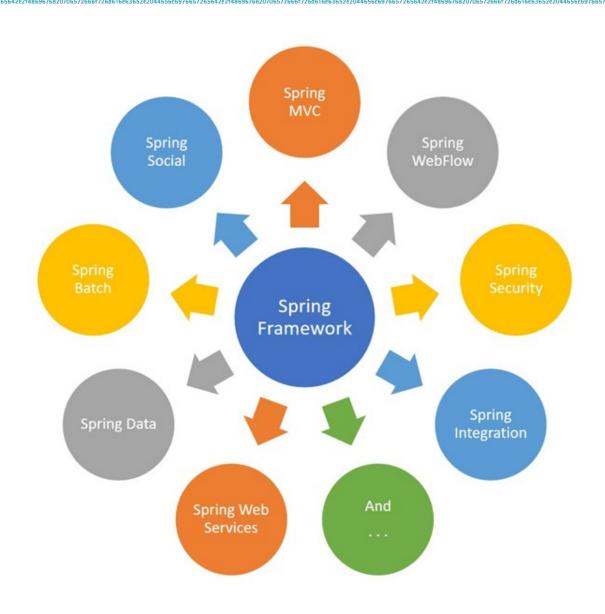


#### Os componentes-base do Spring Boot

- O Spring Boot consiste, ao final, de um template préconfigurado para desenvolvimento e execução de aplicações baseadas no Spring;
- O Spring Boot possui 3 principais componentes:
  - Spring Boot Starter;
  - Spring Boot AutoConfigurator;
  - Spring Boot Actuator

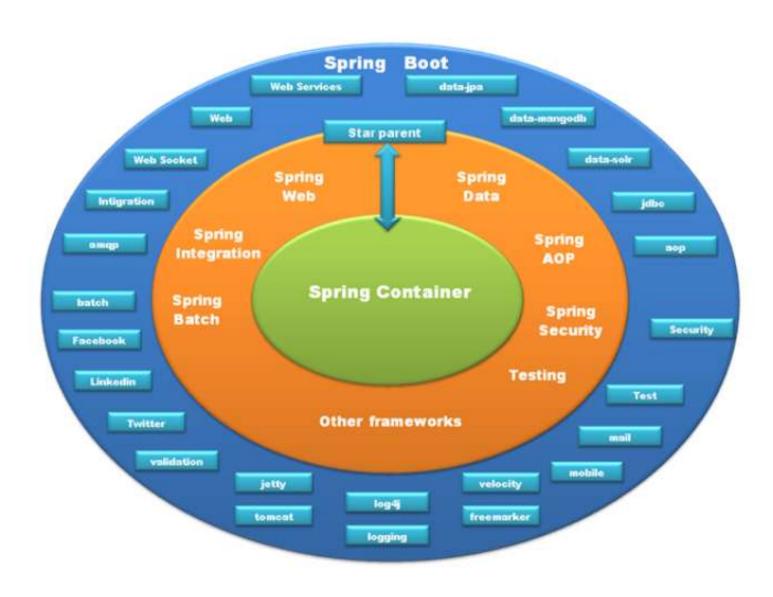


## **Componentes do Spring Boot**



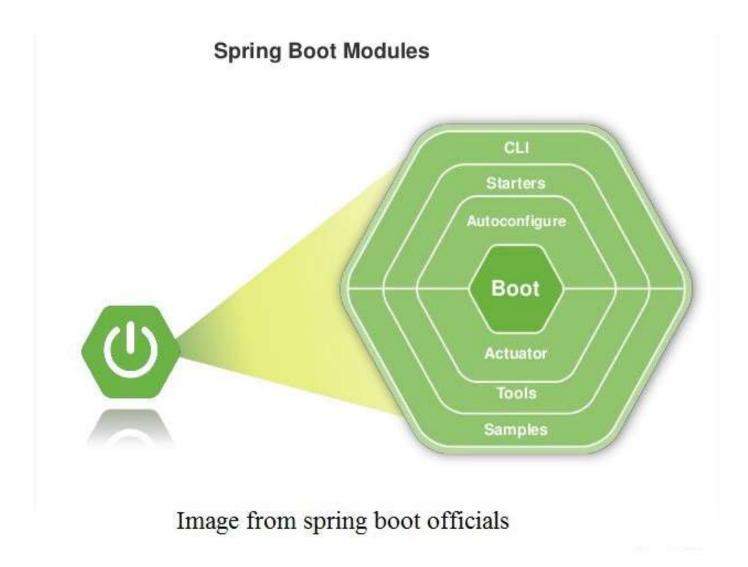


## **Componentes do Spring Boot**













#### O Spring Boot Starter

- O Spring Boot Starter é o coração de uma aplicação;
- Sua função principal é combinar as várias dependências advindas de um projeto Spring Boot em uma única dependência, retirando-se a necessidade de configuração de múltiplas dependências;





#### O Spring Boot AutoConfigurator

- É responsável por gerenciar o processo de configuração de uma aplicação;
- Fornecendo as configurações-padrão e fazendo a fusão destas com as possíveis configurações personalizadas;
- Pode ser visto através da utilização da tradicional annotation @SpringBootApplication, que fica acima do método de inicialização da aplicação;

Por exemplo: por padrão, o Spring Boot AutoConfigurator define o servidor web em localhost na porta 8080





#### **O Spring Boot Actuator**

- As principais funções deste componente são o provisionamento de endpoints;
- A obtenção de métricas da aplicação;





#### **Convention over Configuration**

- Ao invés de informar ao Spring Boot onde estão os HTMLs, deixamos todos eles em um único local que é uma convenção;
- Convensão: Regras previamente estabelecidas;



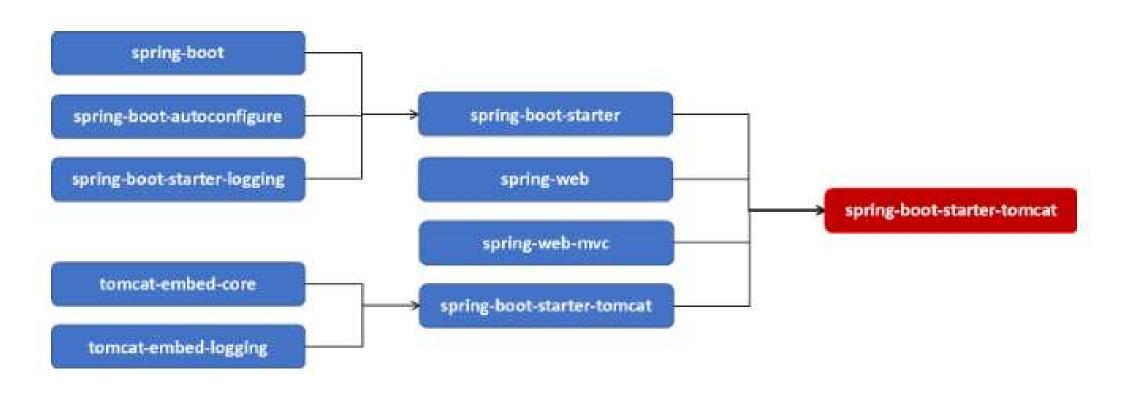


#### **INITIALIZR**

- O spring initializr gera um projeto base para que você já comece o desenvolvimento das regras de negócio da sua aplicação.
- Fornece uma interface web bem simples para o usuário.
- Podendo gerar seu projeto a partir de uma estrutura de configurações pré-moldadas.
- São configurações de versões do java/spring boot, grupo/nome do projeto, série de lista de dependências e etc.

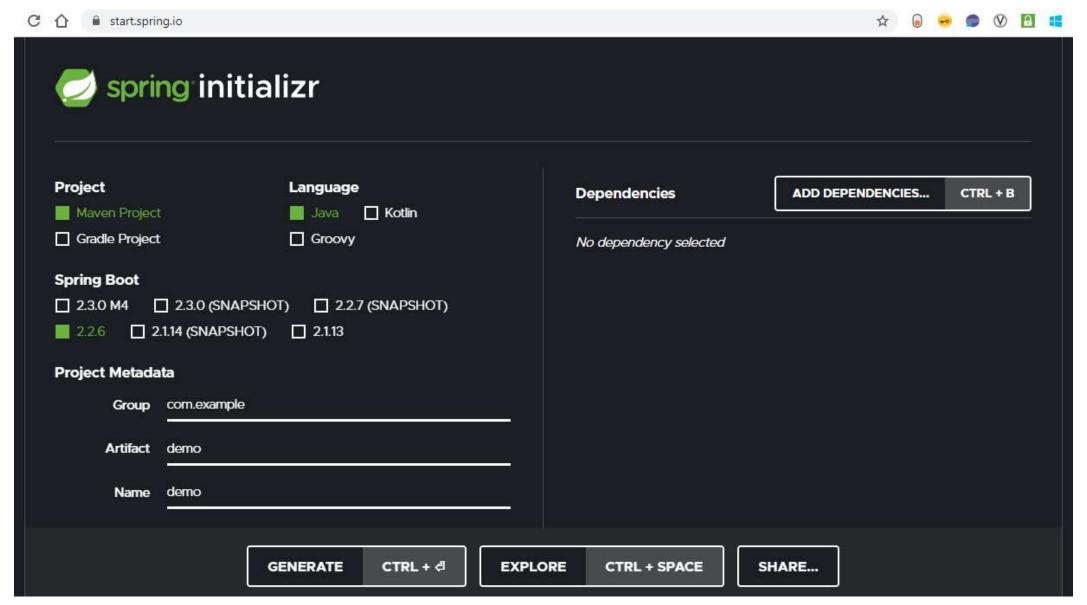










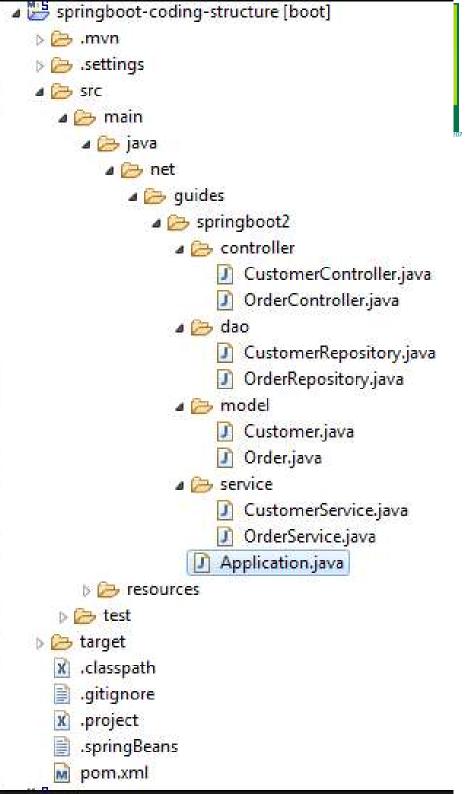


### Estrutura do Projeto

#### Estrutura tipica:

- Controller
- Dao
- Model
- Service

 Não utilize a "default" Package;







 What questions or comments do you have?

