



Introdução ao Ecossistema Spring



Prof. MSc. Kelson Almeida

Agenda

- Introdução
 - Produtividade ?!
 - O que é o ecossistema spring?
- Módulos
 - Spring Boot
 - Spring Data
 - Spring MVC
 - Spring Security
 - Spring Cloud
- Outros módulos...

Produtividade?

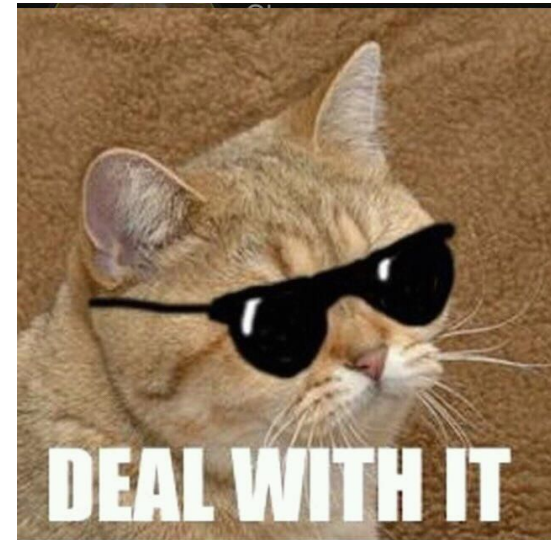
- Agilidade em Desenvolvimento de Software.
- Entregas contínuas.
 - Mercado exige rapidez e eficiência nas entregas de software.
- No desenvolvimento moderno as equipes não tem tempo a perder com configurações e excesso de códigos que poderiam ter sido evitados.
- *Frameworks* nos auxiliam nesses objetivos.



Antigamente...



Atualmente utilizando o ecossistema spring no meu projeto....



Ecosystem Spring

- Ecosystem de desenvolvimento de aplicações **Java**.
 - Com compatibilidade para **Kotlin** e **Groovy**.
- No início o projeto Spring Framework era um único aglomerado de várias funcionalidades.
 - Com o grande crescimento da plataforma, suas funcionalidades foram divididas em módulos, cada um com a sua especialidade.
 - Esses módulos podem ser utilizados em conjunto ou até mesmo com outros Frameworks que não façam parte do ecossistema spring.

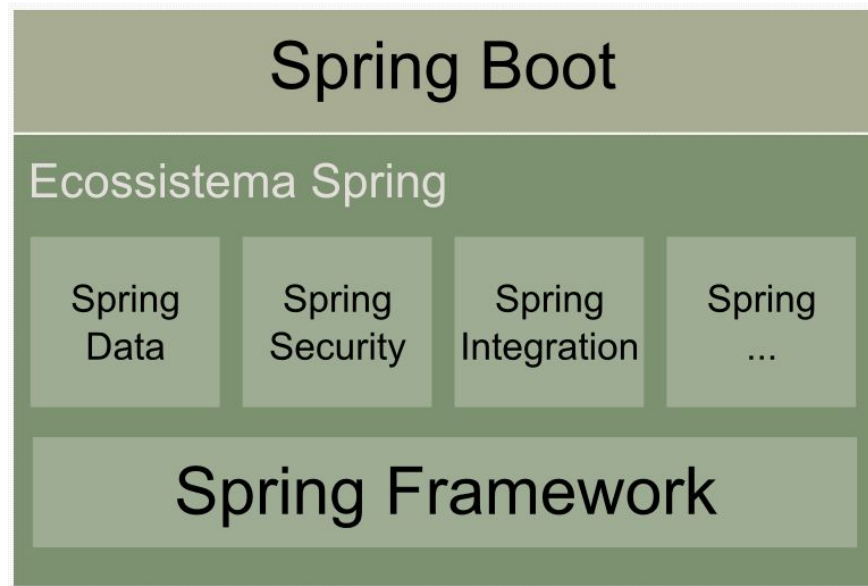


Módulos



- **Spring Boot**

- Acaba com as longas e improdutivas horas de **configurações do projeto**;
- Pré-configuração dos componentes necessários para as aplicações modernas;
- Simplificando o desenvolvimento de aplicações **Java**, **Kotlin** ou **Groovy**;
- Ganho na **produtividade**;
- Gerenciar **dependências**;
- Excelente ferramenta para arquitetura de **microsserviços**.



“Dificuldade” para criar um novo projeto spring boot... (https://start.spring.io)

The screenshot shows the Spring Initializr web application. The interface is divided into several sections:

- Project:** Includes radio buttons for **Maven Project** (selected) and **Gradle Project**.
- Language:** Includes radio buttons for **Java** (selected), **Kotlin**, and **Groovy**.
- Spring Boot:** Includes radio buttons for versions: 3.0.0 (SNAPSHOT), 3.0.0 (M3), 2.7.2 (SNAPSHOT), **2.7.1** (selected), 2.6.10 (SNAPSHOT), and 2.6.9.
- Project Metadata:** Includes input fields for Group (com.example), Artifact (demo), Name (demo), Description (Demo project for Spring Boot), and Package name (com.example.demo). It also includes a **Packaging** section with radio buttons for **Jar** (selected) and **War**.
- Dependencies:** A section on the right with a button **ADD DEPENDENCIES... CTRL + B**. It lists several dependencies: **Spring Data JPA** (SQL), **Spring Security** (SECURITY), and **Eureka Discovery Client** (SPRING CLOUD DISCOVERY). Each dependency has a brief description.
- Footer:** Includes a **GENERATE CTRL + G** button, an **EXPLORE CTRL + SPACE** button, and a **SHARE...** button.

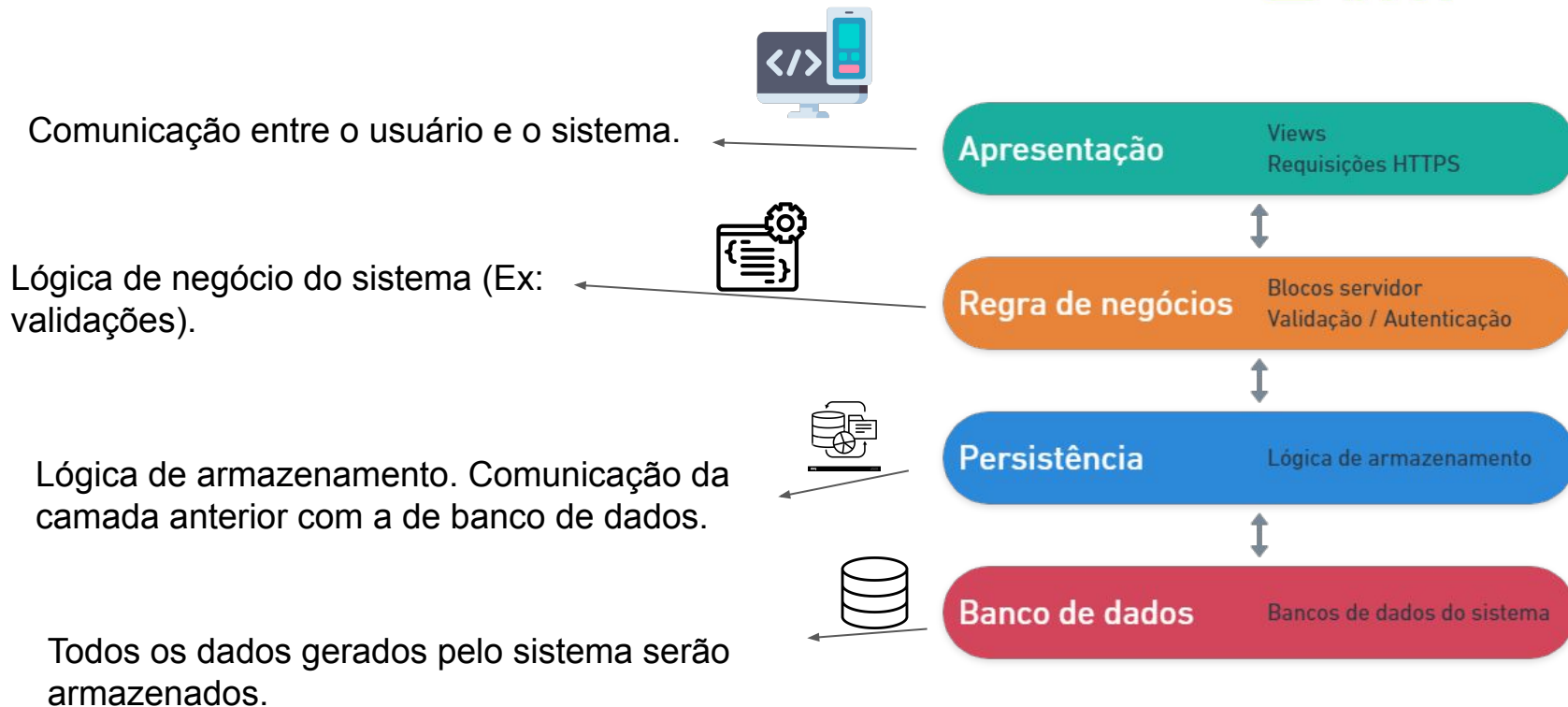
Annotations on the image:

- An arrow points from the text "Gera um .zip com todo o projeto já organizado e suas respectivas libs pré-instaladas." to the **GENERATE** button.
- An arrow points from the text "Adicionando outros módulos do ecossistema no meu projeto." to the **ADD DEPENDENCIES...** button.

Gera um .zip com todo o projeto já organizado e suas respectivas libs pré-instaladas.

Adicionando outros módulos do ecossistema no meu projeto.

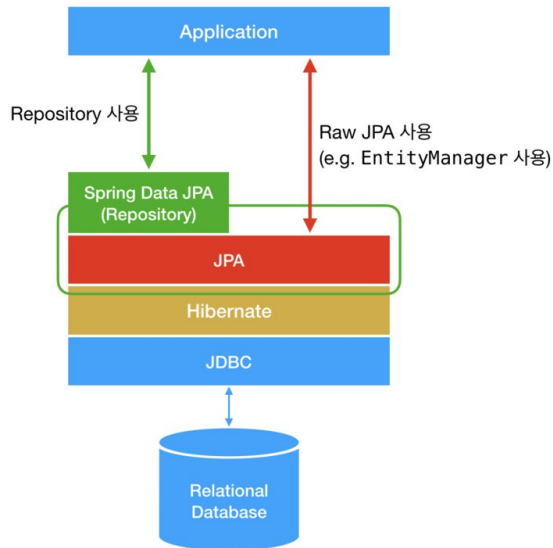
Divisão em Camadas



Módulos



- **Spring Data**
- Acesso e manipulação de dados no ecossistema spring;
- Facilidade em configurar e utilizar o Banco de Dados;
- Compatibilidade com os mais diversos BDs
 - Relacional ou não



```

<dependency>
<groupId>org.springframework.boot</groupId> <artifactId>spring-boot-
</dependency>
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
<groupId>org.mariadb.jdbc</groupId>
<artifactId>mariadb-java-client</artifactId>
<scope>runtime</scope>
</dependency>

```

```

public interface PessoaRepository extends JpaRepository<P
}

```

```

spring.datasource.url=jdbc:mariadb://localhost:3306/banco
spring.datasource.username=root
spring.datasource.password=root
spring.datasource.driver-class-name=org.mariadb.jdbc.Driver
spring.jpa.hibernate.ddl-auto=update
spring.jpa.generate-ddl=true

```

```

PessoaRepository repository;
Pessoa pessoa = pessoaRequest;
repository.save(pessoa);

```

```

@Entity
public class Pessoa {

    @Id
    private String id = UUID.randomUUID().toString();

    private String nomeCompleto;
    private String email;

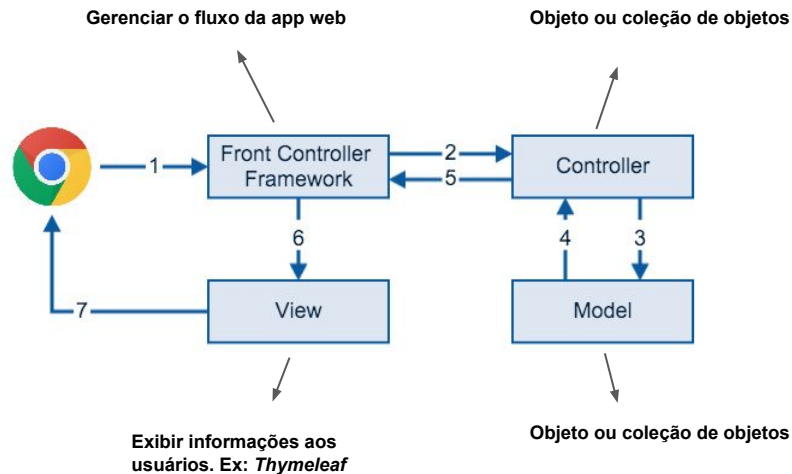
    public Pessoa() {
    }

    public Pessoa(String id, String nomeCompleto, String email) {
        super();
        this.id = id;
        this.nomeCompleto = nomeCompleto;
        this.email = email;
    }
}

```

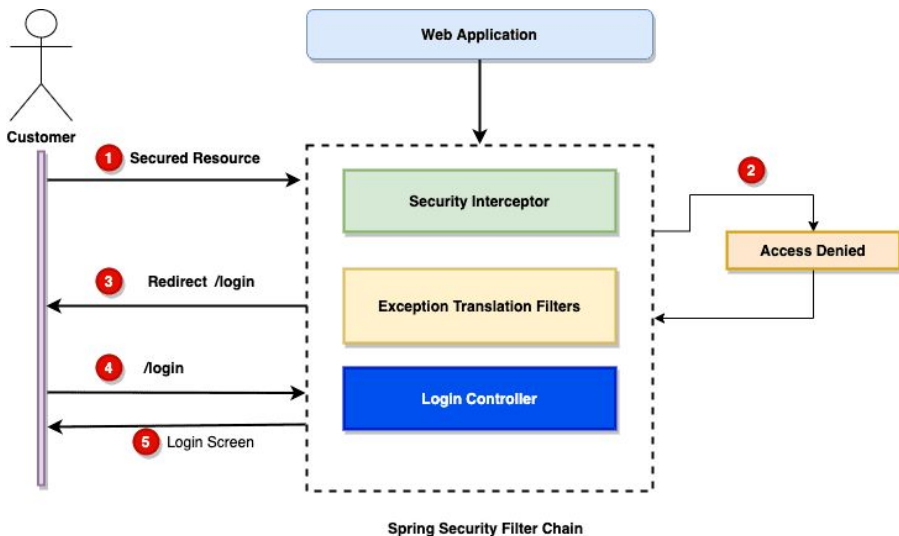
Módulos

- **Spring MVC**
- Framework para desenvolvimento web
- Model View Controller
 - Padrão de Projeto dividido em 3 camadas
- Pode levar mais tempo no desenvolvimento (Comparado ao Spring Boot)



Módulos

- **Spring Security**
- Solução oficial de segurança em aplicações Spring Boot
- Autenticação
 - Basicamente o login na aplicação.
- Autorização
 - No processo após a autenticação, o usuário autenticado tem permissão para tal recurso?
- Armazenamento de Senha
 - Algoritmos de criptografias para evitar que as senhas sejam armazenadas em texto puro.



Módulos



- **Spring Cloud**
- Stack completa para **microserviços**
- Fornecer uma integração completa entre o Spring Boot e o projeto **Netflix OSS**.
 - Netflix OSS é um conjunto de frameworks e bibliotecas que a Netflix escreveu para resolver alguns problemas em sistemas distribuídos em escalas.
- Hoje o OSS é sinônimo de dev microserviços em ambiente em **nuvem**.



Outros módulos

- Spring Web Flow
 - Gerencia a navegação em aplicações web. Definir passos dados pelo usuário na aplicação.
- Spring Batch
 - Implementações de tarefas automáticas executadas em periodicidade pré-definida.
- Spring Roo
 - Ferramenta baseada em texto para o desenvolvimento rápido de aplicações.

Nas próximas aulas...

- Navegaremos mais a fundo em cada um dos módulos desse belo ecossistema :)