

# Importando e Estruturando o Projeto Spring Boot

## Análise Inicial na IDE

Rômulo C. Silvestre

September 8, 2025

# Importação e Verificação de Dependências

- 1 **Abrir o Projeto:** Importar a pasta `api` na IDE.
- 2 **Resolver Dependências:** Aguardar o Maven baixar as dependências listadas no `pom.xml`.
  - ▶ *Spring Web, DevTools, Lombok.*
  - ▶ *spring-boot-starter-test* é adicionado automaticamente.
- 3 **Solução de Problemas:** Se houver falhas, usar a opção *"Reload All Maven Projects"*.

# Analizando o pom.xml

## Onde está o Spring Boot? A Tag <parent>

O projeto herda do `spring-boot-starter-parent`. Isso define a versão principal do Spring Boot e gerencia as versões das dependências "filhas".

## Gerenciamento Simplificado de Dependências

Graças à herança do <parent>, não é necessário especificar a versão para os *starters* do Spring Boot. O Maven já sabe qual versão usar.

```
<artifactId>spring-boot-starter-web</artifactId>
```

# Estrutura de Diretórios Gerada

## src/main

- java/med/voll/api
  - ▶ `ApiApplication.java`
    - ★ Classe principal com `@SpringBootApplication` e método `main`.
- resources
  - ▶ `application.properties`: Arquivo principal de configurações.
  - ▶ `static, templates`: Para web (não usados na API Rest).

## src/test

- java/med/voll/api
  - ▶ `ApiApplicationTests.java`
    - ★ Classe de exemplo para testes automatizados.

## Raiz do Projeto

- `pom.xml`: Arquivo de configuração do Maven.

# Conclusão e Próximos Passos

## Resumo

O Spring Initializr nos entrega um projeto pronto para rodar, com toda a estrutura de diretórios e configurações do Maven já preparadas.

## Próximo Passo

Com a estrutura compreendida, o projeto está pronto para ser executado pela primeira vez e para iniciarmos o desenvolvimento, como um "Hello, World".