

Validação e Persistência de Dados

Adicionando Dependências e Configurando o Banco de Dados

Rômulo C. Silvestre

September 8, 2025

1. Adicionando Novas Dependências ao pom.xml

Objetivo

Incluir os módulos necessários para validação, persistência (JPA), migração (Flyway) e conexão com o banco (MySQL).

Principais Dependências Adicionadas

- `spring-boot-starter-validation`
- `spring-boot-starter-data-jpa`
- `flyway-mysql`
- `mysql-connector-j`

Ação Obrigatória

Após alterar o `pom.xml`, clique em **"Reload All Maven Projects"** na IDE para baixar as novas bibliotecas.

2. Configurando a Conexão com o Banco (DataSource)

O Problema

Após adicionar as dependências, a aplicação falha ao iniciar por não saber como se conectar ao banco de dados.

A Solução: `application.properties`

Arquivo: `src/main/resources/application.properties`

```
# URL de conexao com o banco de dados
spring.datasource.url=jdbc:mysql://localhost:3306/
    vollmed_api

# Usuario do banco de dados
spring.datasource.username=root

# Senha do banco de dados
spring.datasource.password=root
```

3. Criando o Banco de Dados Manualmente

O Último Passo da Configuração

O Spring Boot gerencia a conexão, mas **não cria o banco de dados** (schema) automaticamente.

Ação Manual Necessária

É preciso conectar-se ao MySQL e executar o comando de criação.

Comando SQL

```
mysql> create database vollmed_api;
```

Com o banco criado, a aplicação inicia com sucesso e está pronta para a próxima etapa: validações e persistência de dados.