



Desenvolvimento de Serviços com Spring Boot

TP2

Luiz Carlos de Souza Ardo vino Ribeiro

CPF: 155.647.787-23

Santa Catarina, 18 de agosto de 2025

Atividade

1:

Para o projeto, eu escolhi o java 21, por se tratar de uma versão LTS mais recente. Além de ser recomendada oficialmente pelo springboot 3.x, ela traz melhorias significativas de performance e novos recursos, como Virtual Threads, que facilitam a criação de aplicações web mais facilmente. Essa escolha garante longevidade, estabilidade e modernidade ao projeto.

Atividade

2:

Spring

Initializr

Project

☐ Gradle - Groovy

☐ Gradle - Kotlin

☒ Maven

Language

☒ Java

☐ Kotlin

☐ Groovy

Spring Boot

☐ 4.0.0 (SNAPSHOT)

☐ 4.0.0 (M1)

☐ 3.5.5 (SNAPSHOT)

☒ 3.5.4

☐ 3.4.9 (SNAPSHOT)

☐ 3.4.8

Project Metadata

Group

com.CrudTP2

Artifact

demo

Name

demo

Description

Demo project for Spring Boot

Package name

com.CrudTP2.demo

Packaging

☒ Jar

☐ War

Java

☐ 24

☒ 21

☐ 17

Atividade 3:

Dependências

Dependencies

ADD DEPENDENCIES... CTRL + B

Spring Web WEB
Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Spring Data JPA SQL
Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.

H2 Database SQL
Provides a fast in-memory database that supports JDBC API and R2DBC access, with a small (2mb) footprint. Supports embedded and server modes as well as a browser based console application.

Validation I/O
Bean Validation with Hibernate validator.

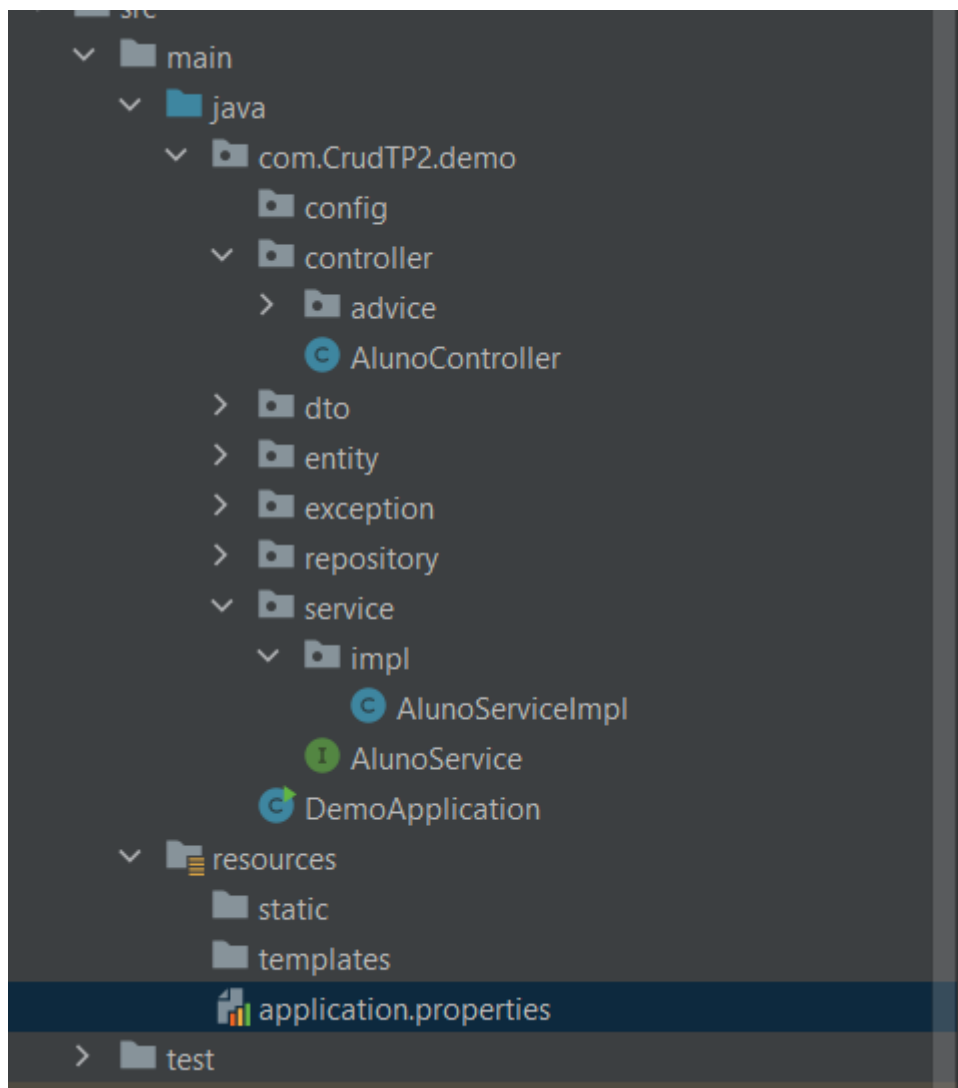
Lombok DEVELOPER TOOLS
Java annotation library which helps to reduce boilerplate code.

Justificativa:

No desenvolvimento deste projeto, selecionei um conjunto de dependências que se complementam e garantem o funcionamento adequado da aplicação. O Spring Web foi

escolhido para possibilitar a criação da API RESTful e o mapeamento de endpoints, que são a base do CRUD. Para a camada de persistência, optei pelo Spring Data JPA, que facilita a interação com o banco de dados ao disponibilizar repositórios prontos e eliminar a necessidade de escrever consultas SQL manuais. Como banco de dados, utilizei o H2, por ser leve, em memória e de fácil configuração, o que torna os testes e a validação da aplicação mais rápidos, além de oferecer um console acessível via navegador. Também incluí o Validation, que permite aplicar regras de validação diretamente nos modelos com anotações como @NotNull e @Email, garantindo que apenas dados consistentes sejam aceitos. Por fim, adicionei o Lombok, para reduzir código repetitivo e melhorar a legibilidade, já que ele gera automaticamente getters, setters e construtores. Dessa forma, o conjunto de dependências oferece uma base completa, eficiente e de fácil manutenção para o desenvolvimento do CRUD proposto.

Evidência final do projeto:



A

Executar comando | Executar selecionado | Auto complete | Limpar | Comando SQL.

```
INSERT INTO ALUNOS (nome, email, curso) VALUES ('João Souza', 'joao@email.com', 'Medicina');
```

```
INSERT INTO ALUNOS (nome, email, curso) VALUES ('João Souza', 'joao@email.com', 'Medicina');  
Número de registros alterados: 1  
(0 ms)
```


HTTP <http://localhost:8080/alunos> Save Share

GET <http://localhost:8080/alunos> Send

Params Authorization Headers (7) **Body** Scripts Tests Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (5) Test Results

JSON Preview Visualize

```
1 [
2   {
3     "id": 1,
4     "nome": "Maria Silva",
5     "email": "maria@email.com",
6     "curso": "Engenharia"
7   },
8   {
9     "id": 2,
10    "nome": "João Souza",
11    "email": "joao@email.com",
12    "curso": "Medicina"
```

HTTP <http://localhost:8080/alunos/1> Save Share

PUT <http://localhost:8080/alunos/1> Send

Params Authorization Headers (9) **Body** Scripts Tests Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON** Beautify

```
1 {
2   "nome": "Luiz Carlos",
3   "email": "luiz.carlos@email.com",
4   "curso": "Ciência da Computação"
5 }
6
```

Body Cookies Headers (5) Test Results

JSON Preview Visualize

```
1 {
2   "id": 1,
3   "nome": "Luiz Carlos",
4   "email": "luiz.carlos@email.com",
5   "curso": "Ciência da Computação"
6 }
```