11ª Emisión

DIPLOMADO

Desarrollo de Sistemas con Tecnología Java

Módulo 7

Persistencia con Spring Data

Dr. Omar Mendoza González

omarmendoza564@aragon.unam.mx

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
Dirección General de Cómputo y de Tecnologías de información y Comunicación
Dirección de Docencia en TIC

DGTIC

50 AÑOS Educación Continua 1971 - 2021

REST

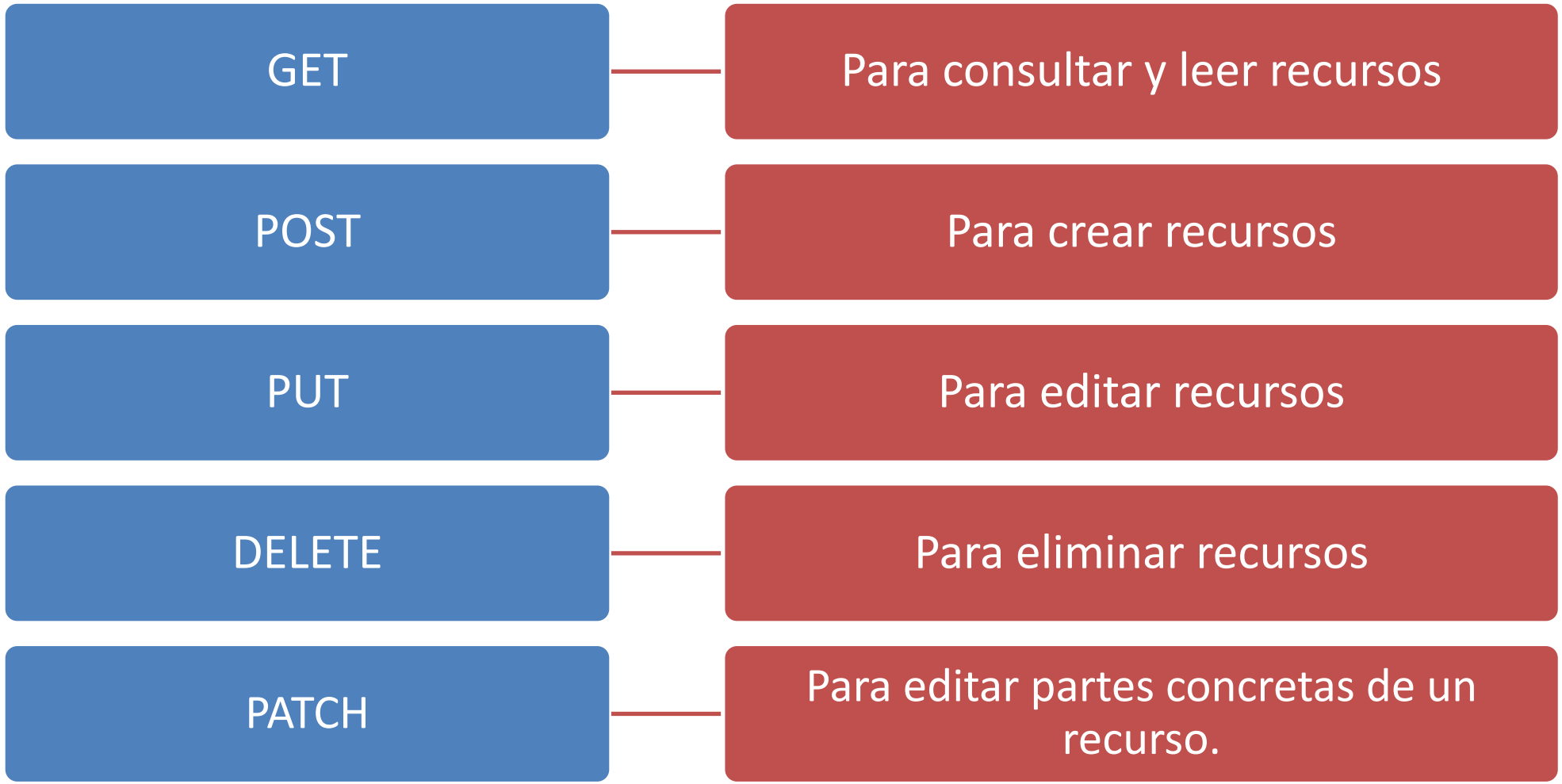
- REST, REpresentational State Transfer, es un tipo de arquitectura de desarrollo web que se apoya totalmente en el estándar HTTP.

- REST permite crear servicios y aplicaciones que pueden ser usadas por cualquier dispositivo o cliente que entienda HTTP.

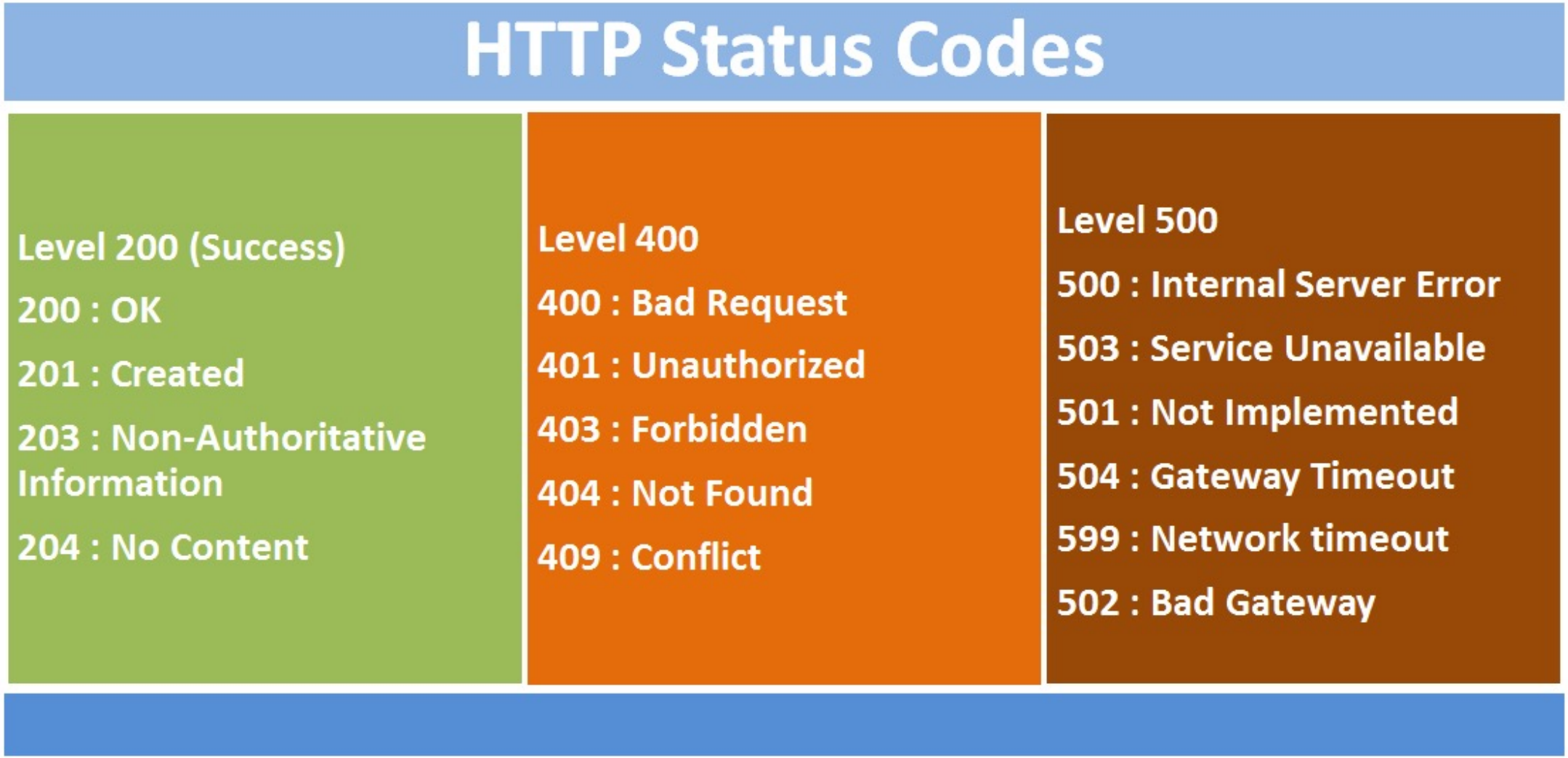
REST

- Características
  - Independencia de plataforma.
  - Independencia de lenguaje de programación.
  - Basado en estándares.
  - Puede ser usado fácilmente en presencia de firewall.
  - Protocolo cliente/servidor sin estado
  - Sólo utiliza HTTP

Verbos HTTP


GET
Para consultar y leer recursos
POST
Para crear recursos
PUT
Para editar recursos
DELETE
Para eliminar recursos
PATCH
Para editar partes concretas de un recurso.

Códigos de estado HTTP



Spring Data REST

- Spring Data REST es parte del proyecto general Spring Data y facilita la creación de servicios web REST sobre los repositorios de Spring Data.

- Se basa en dichos repositorios, analiza el modelo de dominio que se tenga definido en las entidades y expone estos recursos HTTP controlados por hipermedia, es decir, bajo la arquitectura HATEOAS (Hypermedia as the Engine of Application State)

Spring Data REST

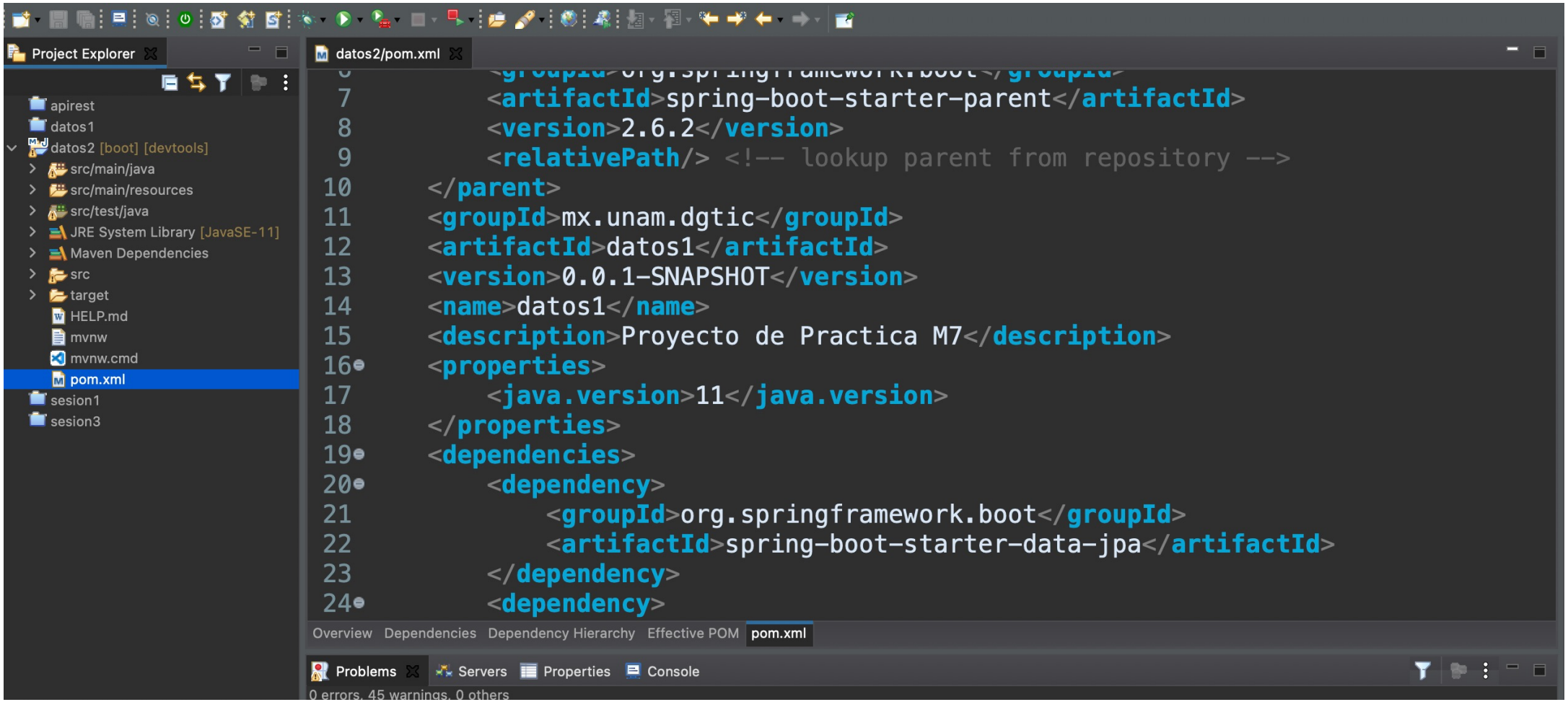
- Aprovecha los standards HYPERMEDIA & Internet

- HAL (Hypertext Application Language,)
- ALPS (Application-Level Profile Semantics)
- JSON Schema
- URI Templates (RFC 6570) text/uri-list mediatype (RFC 2483)
- profile link relation (RFC 6906)

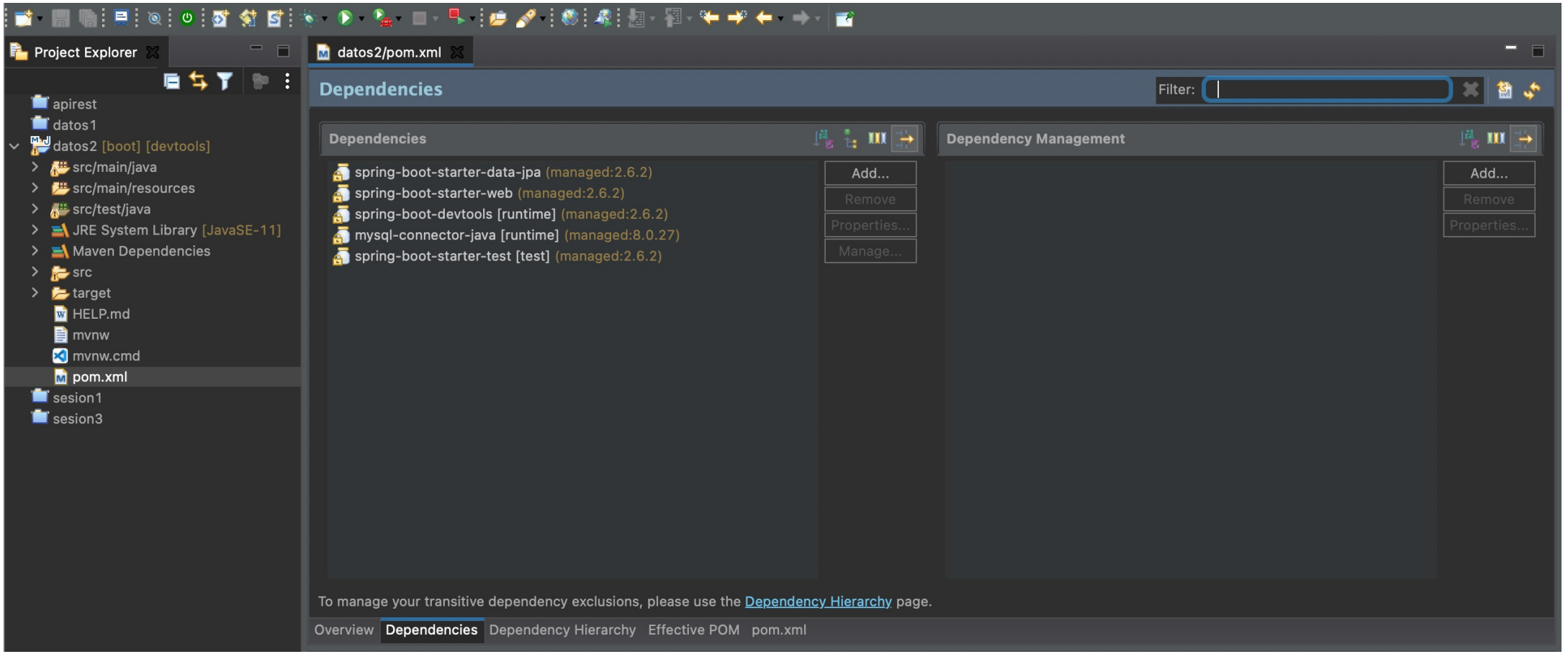
Json


```
{
    "matricula": "7B",
    "nombre": "Gema",
    "paterno": null,
    "fnac": "2001-03-20T06:00:00.000+00:00",
    "estatura": 1.53,
    "calificaciones": [
        {
        "id": 7,
        "materia": "BDII",
        "calificacion": 8
        },
        {
        "id": 8,
        "materia": "POO",
        "calificacion": 5
        }
    ]
}
```

pom.xml


Project Explorer
apirest
datos1
datos2 [boot] [devtools]
src/main/java
src/main/resources
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
sesion1
sesion3
datos2/pom.xml
<artifactId>spring-boot-starter-parent</artifactId>
<version>2.6.2</version>
<relativePath/> <!-- lookup parent from repository -->
</parent>
<groupId>mx.unam.dgtic</groupId>
<artifactId>datos1</artifactId>
<version>0.0.1-SNAPSHOT</version>
<name>datos1</name>
<description>Proyecto de Practica M7</description>
<properties>
<java.version>11</java.version>
</properties>
<dependencies>
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-data-jpa</artifactId>
</dependency>
<dependency>
Overview
Dependencies
Dependency Hierarchy
Effective POM
pom.xml
Problems
Servers
Properties
Console
0 errors, 45 warnings, 0 others

pom.xml


Project Explorer
apirest
datos1
datos2 [boot] [devtools]
src/main/java
src/main/resources
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
sesion1
sesion3
datos2/pom.xml
Dependencies
Filter:
Dependencies
spring-boot-starter-data-jpa (managed:2.6.2)
spring-boot-starter-web (managed:2.6.2)
spring-boot-devtools [runtime] (managed:2.6.2)
mysql-connector-java [runtime] (managed:8.0.27)
spring-boot-starter-test [test] (managed:2.6.2)
Add...
Remove
Properties...
Manage...
Dependency Management
Add...
Remove
Properties...
To manage your transitive dependency exclusions, please use the Dependency Hierarchy page.
Overview
Dependencies
Dependency Hierarchy
Effective POM
pom.xml

pom.xml

https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-data-rest

Spring Boot Starter Data REST » 2.6.2

Starter for exposing Spring Data repositories over REST using Spring Data REST

License	Apache 2.0
Organization	Pivotal Software, Inc.
HomePage	https://spring.io/projects/spring-boot
Date	(Dec 22, 2021)
Files	pom (2 KB) jar (4 KB) View All
Repositories	Central
Used By	228 artifacts

Maven | Gradle | Gradle (Short) | Gradle (Kotlin) | SBT | Ivy | Grape | Leiningen | Buildr

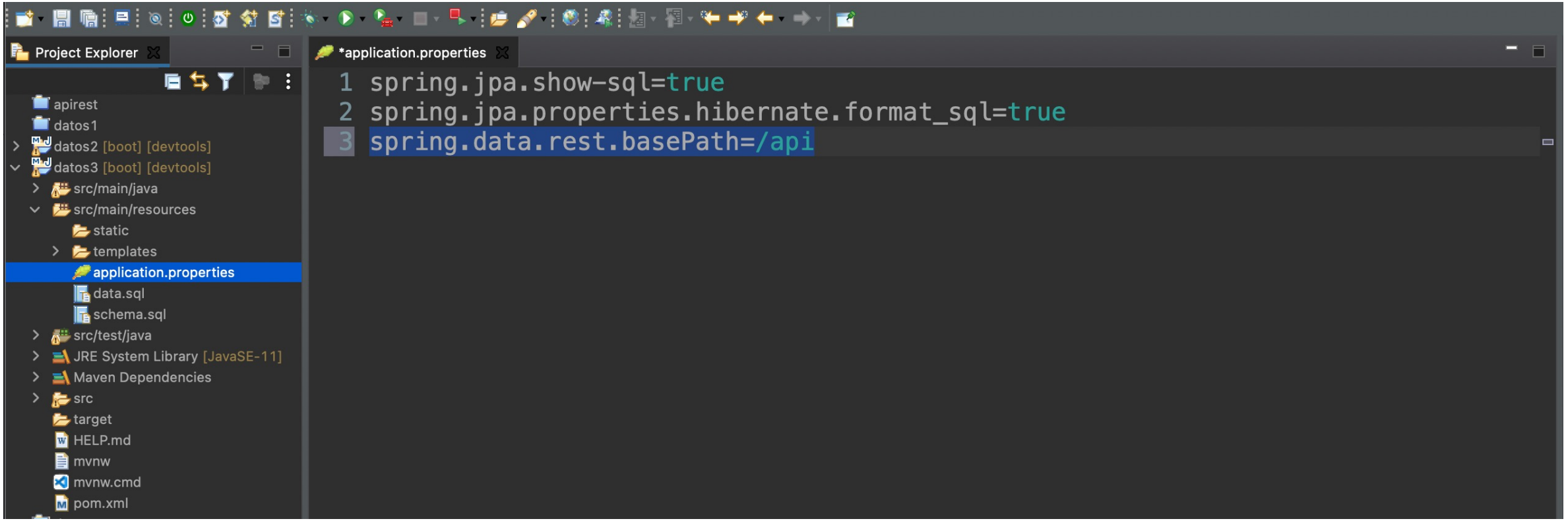
```
<!-- https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-data-rest -->
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-rest</artifactId>
    <version>2.6.2</version>
</dependency>
```

Include comment with link to declaration

pom.xml

```
<!--
https://mvnrepository.com/artifact/org.springframework.boot/spring-
boot-starter-data-rest -->
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-rest</artifactId>
</dependency>
```

application.properties



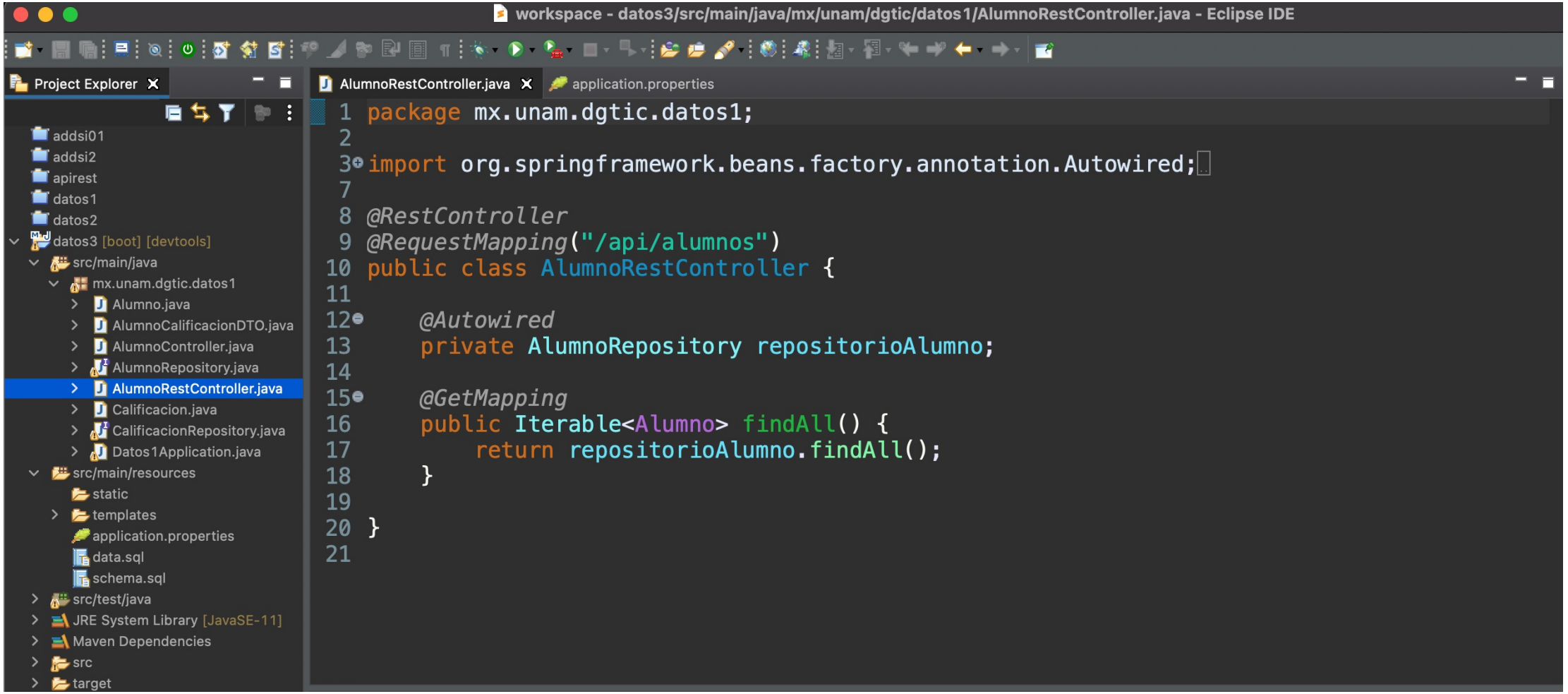
@RestController

- La anotación @RestController permite que los métodos del controlador se expongan como API RESTful

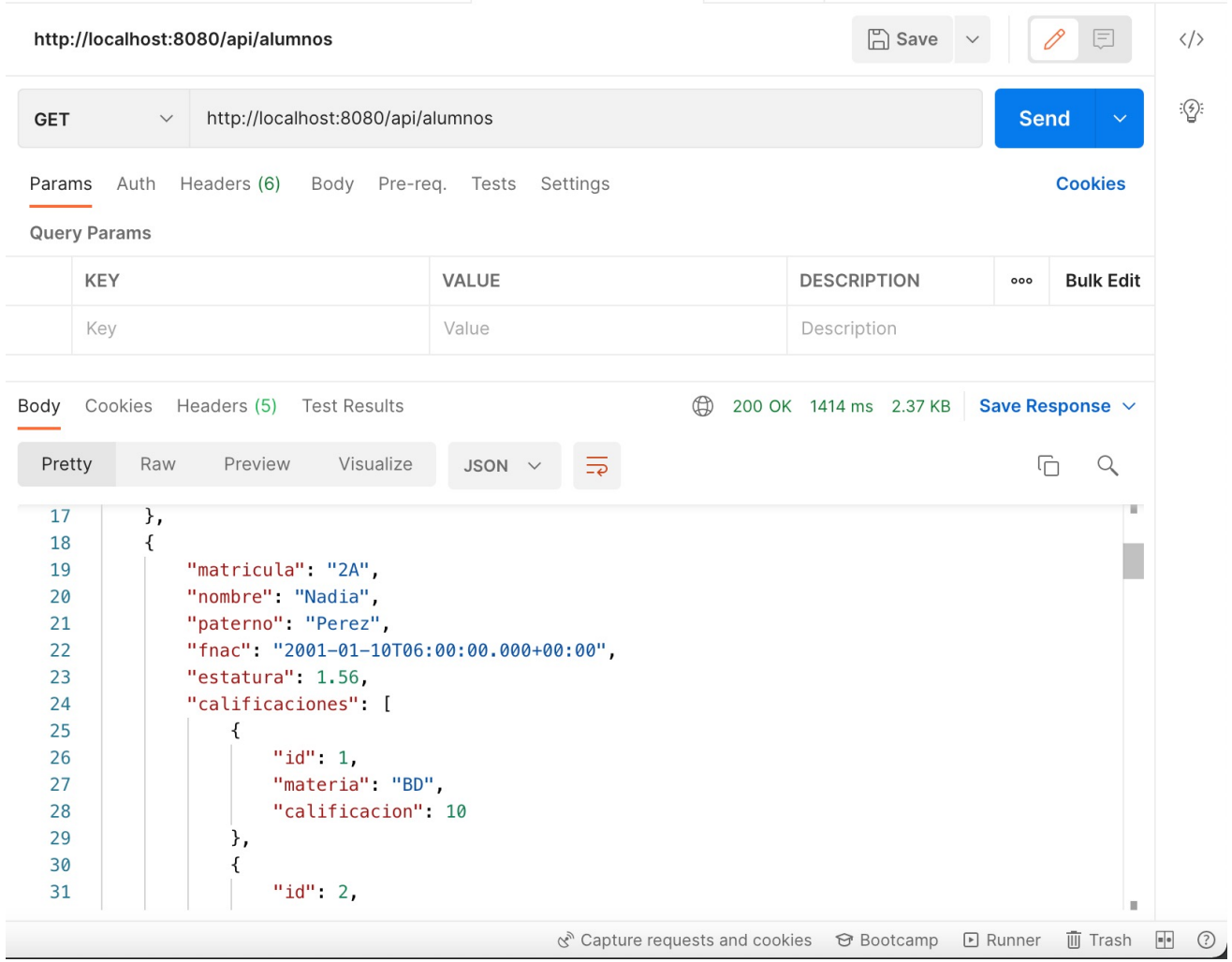
- Es una versión especializada del controlador.

- Incluye las anotaciones *@Controller* y *@ResponseBody* y simplifica la implementación del controlador

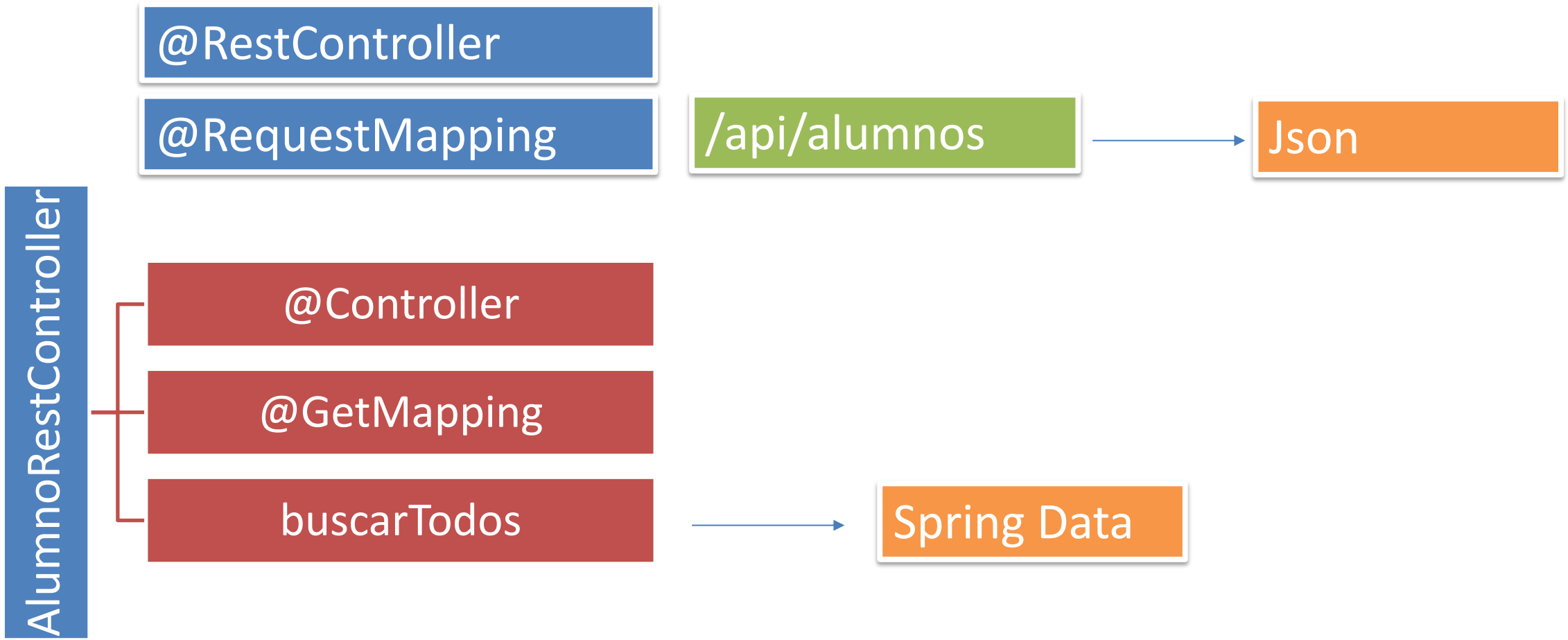
AlumnoRestController



http://localhost:8080/api/alumnos



Spring REST GET



pom.xml

https://mvnrepository.com/artifact/io.rest-assured/rest-assured

REST Assured » 4.4.0

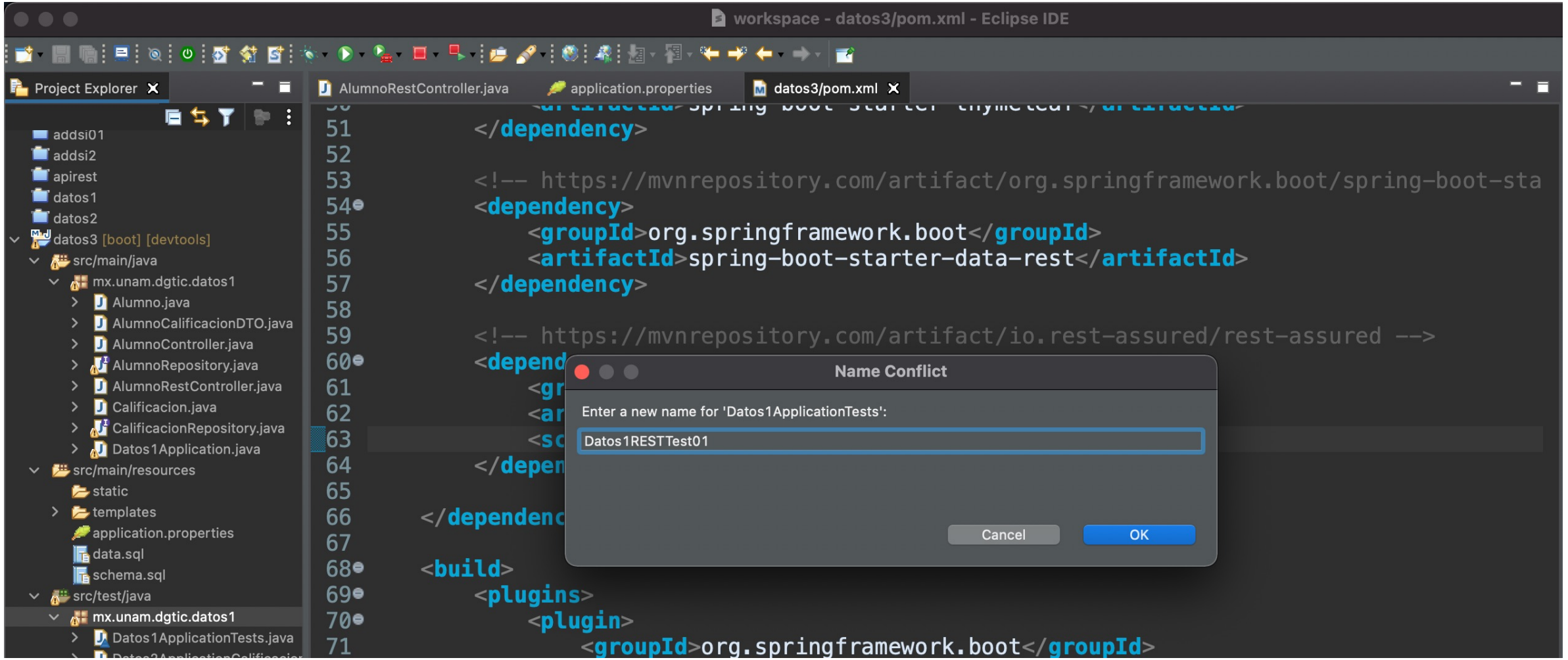
Java DSL for easy testing of REST services

License	Apache 2.0
Categories	Testing Frameworks
HomePage	http://code.google.com/p/rest-assured
Date	(May 23, 2021)
Files	bundle (680 KB) View All
Repositories	Central
Used By	1,803 artifacts
Vulnerabilities	Vulnerabilities from dependencies: CVE-2019-10172

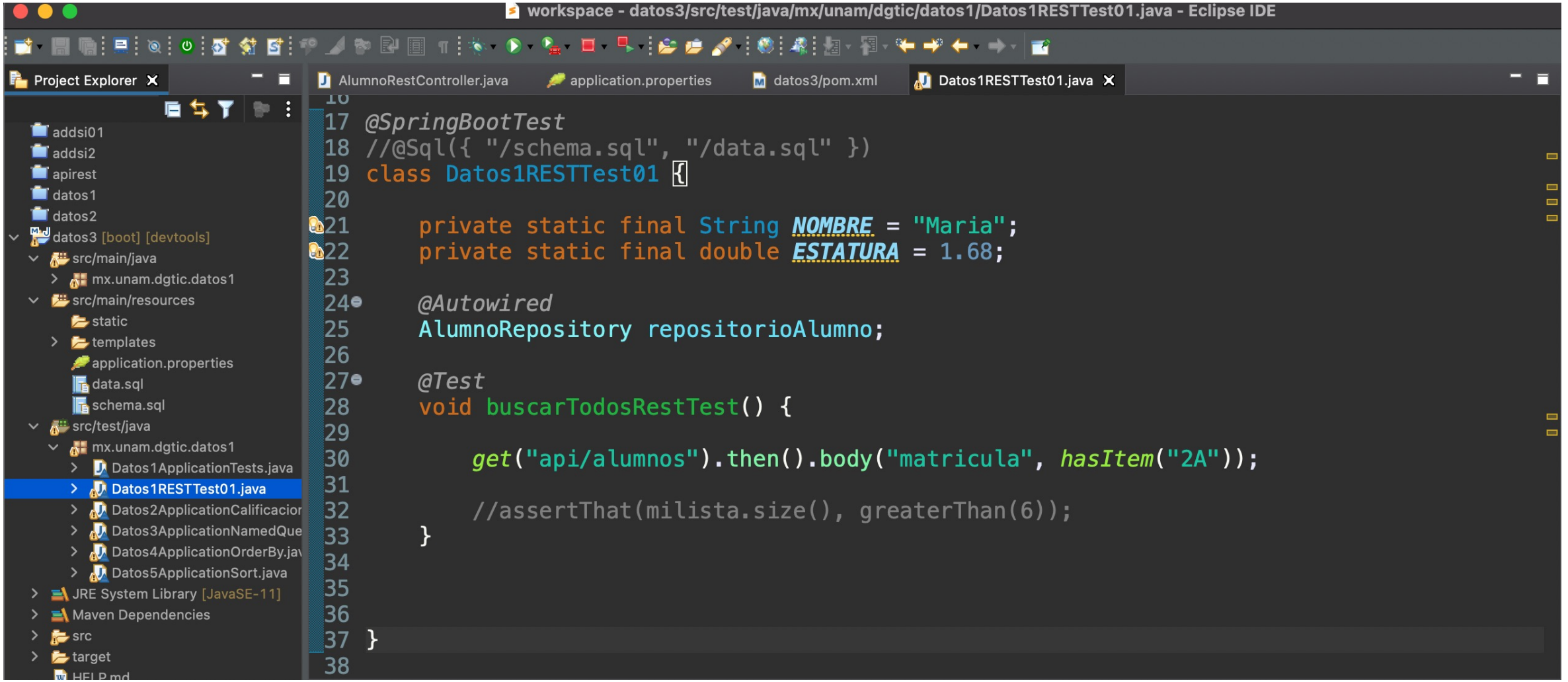
Maven | Gradle | Gradle (Short) | Gradle (Kotlin) | SBT | Ivy | Grape | Leiningen | Buildr

```
<!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured -->
<dependency>
    <groupId>io.rest-assured</groupId>
    <artifactId>rest-assured</artifactId>
    <version>4.4.0</version>
    <scope>test</scope>
</dependency>
```

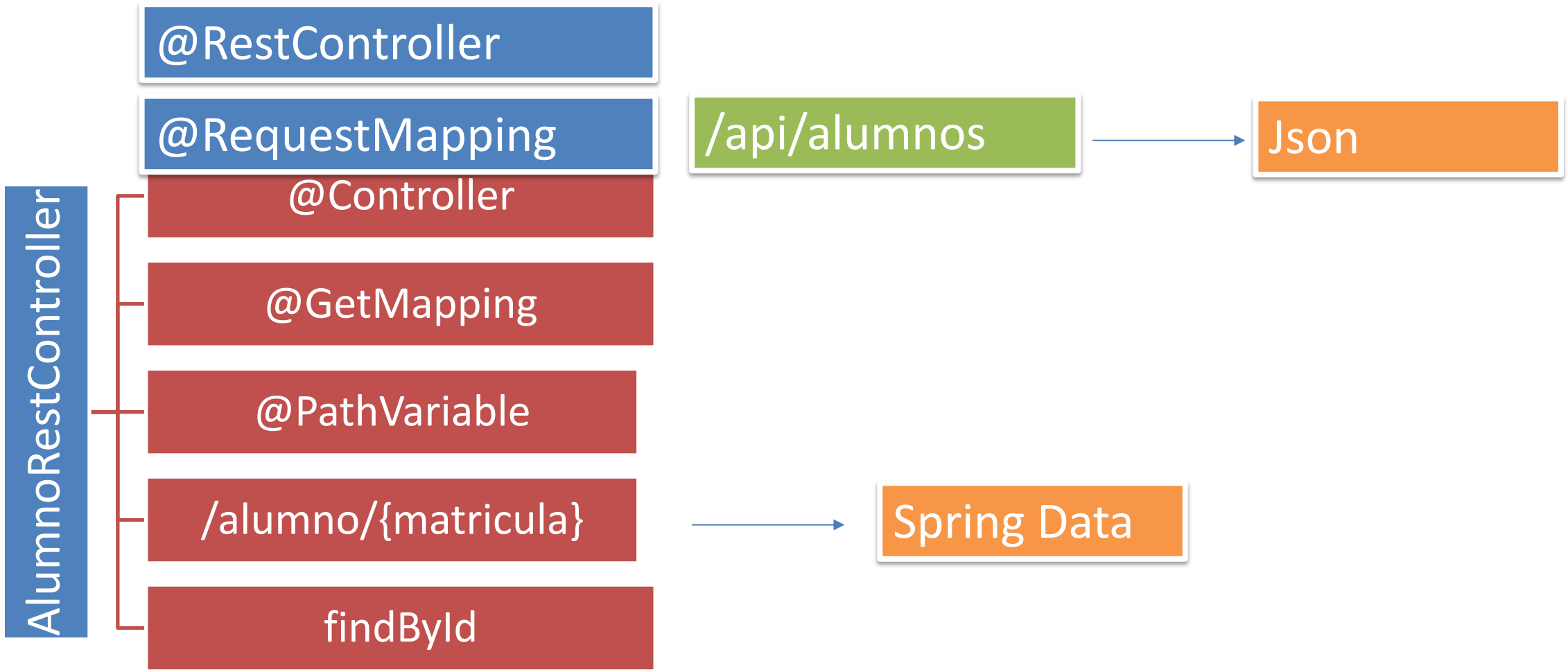
Agregar nueva prueba



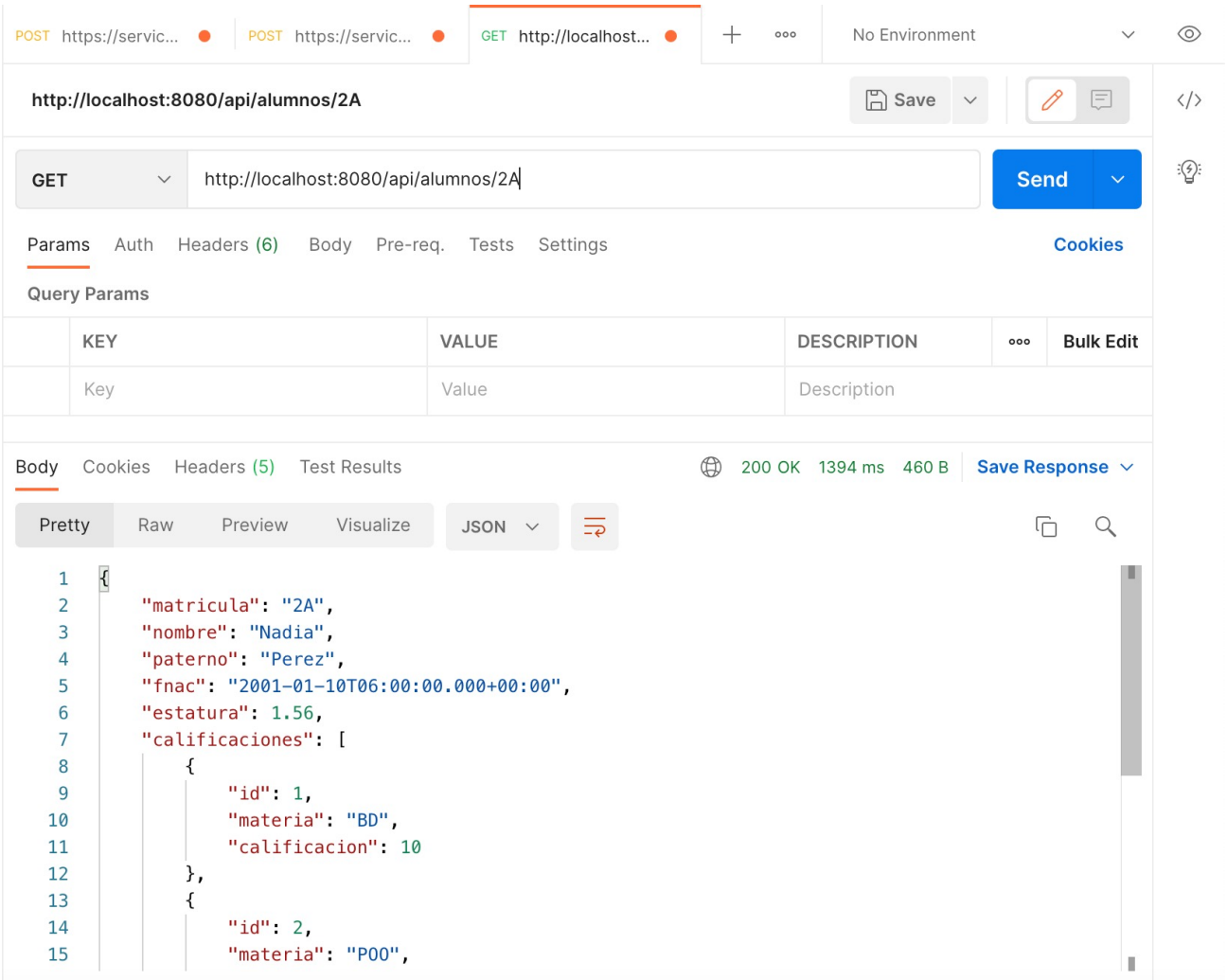
Lanzar la aplicación y Lanzar la Prueba Unitaria



Spring REST GET/{ID}



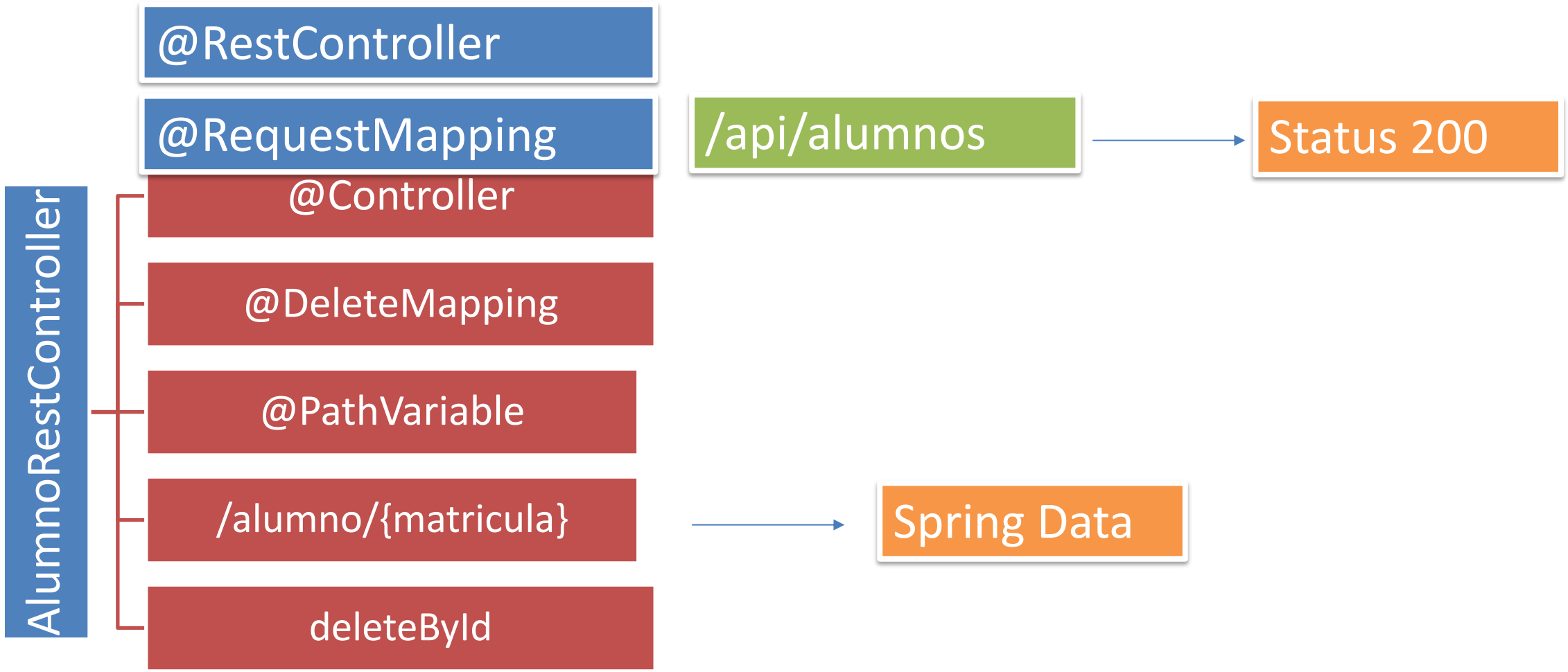
http://localhost:8080/api/alumnos/2A



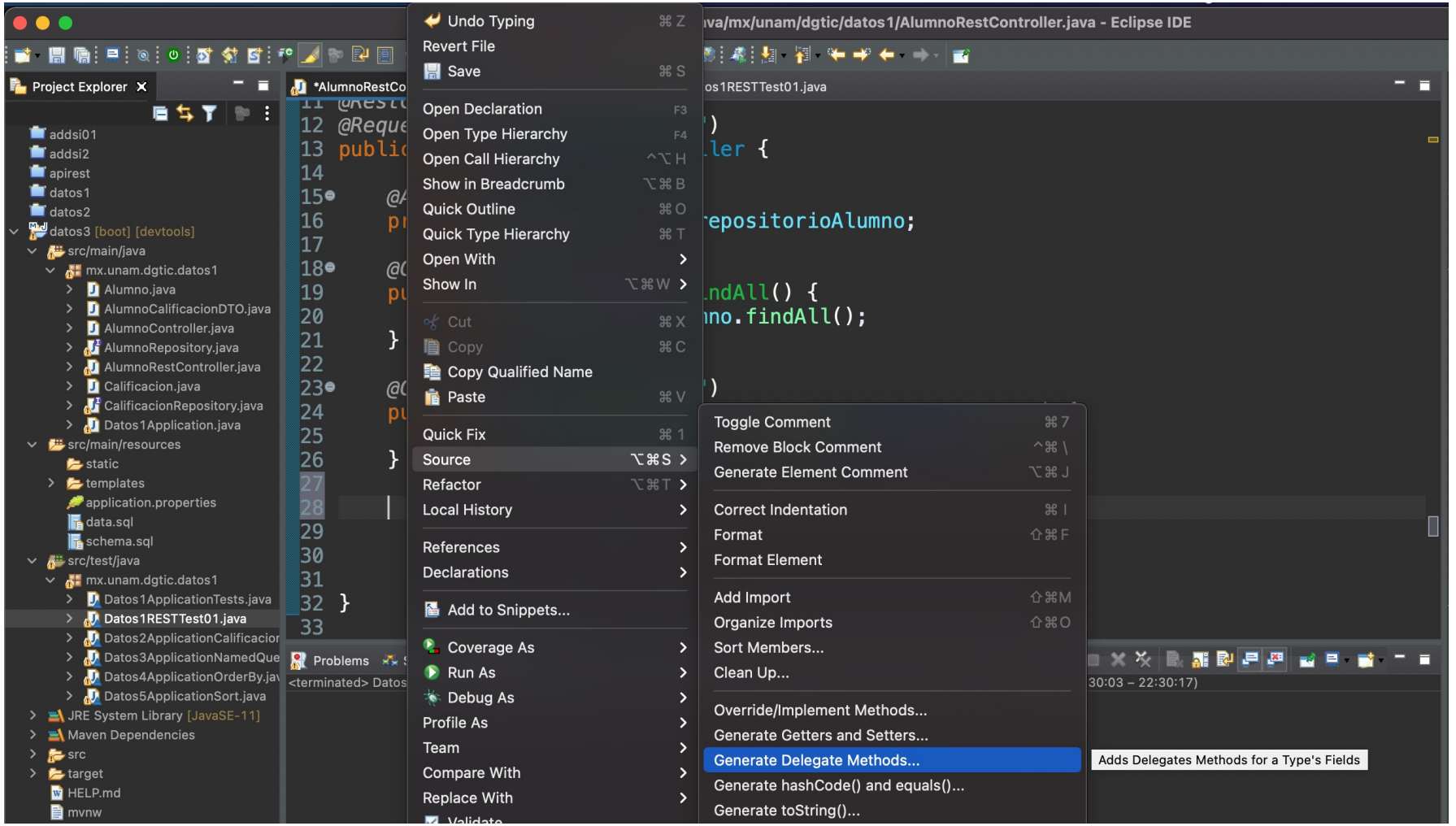
Lanzar la aplicación y Lanzar la segunda UT

```
@Test
void buscarUnoRestTest() {

    get("api/alumnos/2A").then().body("nombre", equalTo("Nadia"));

}
```

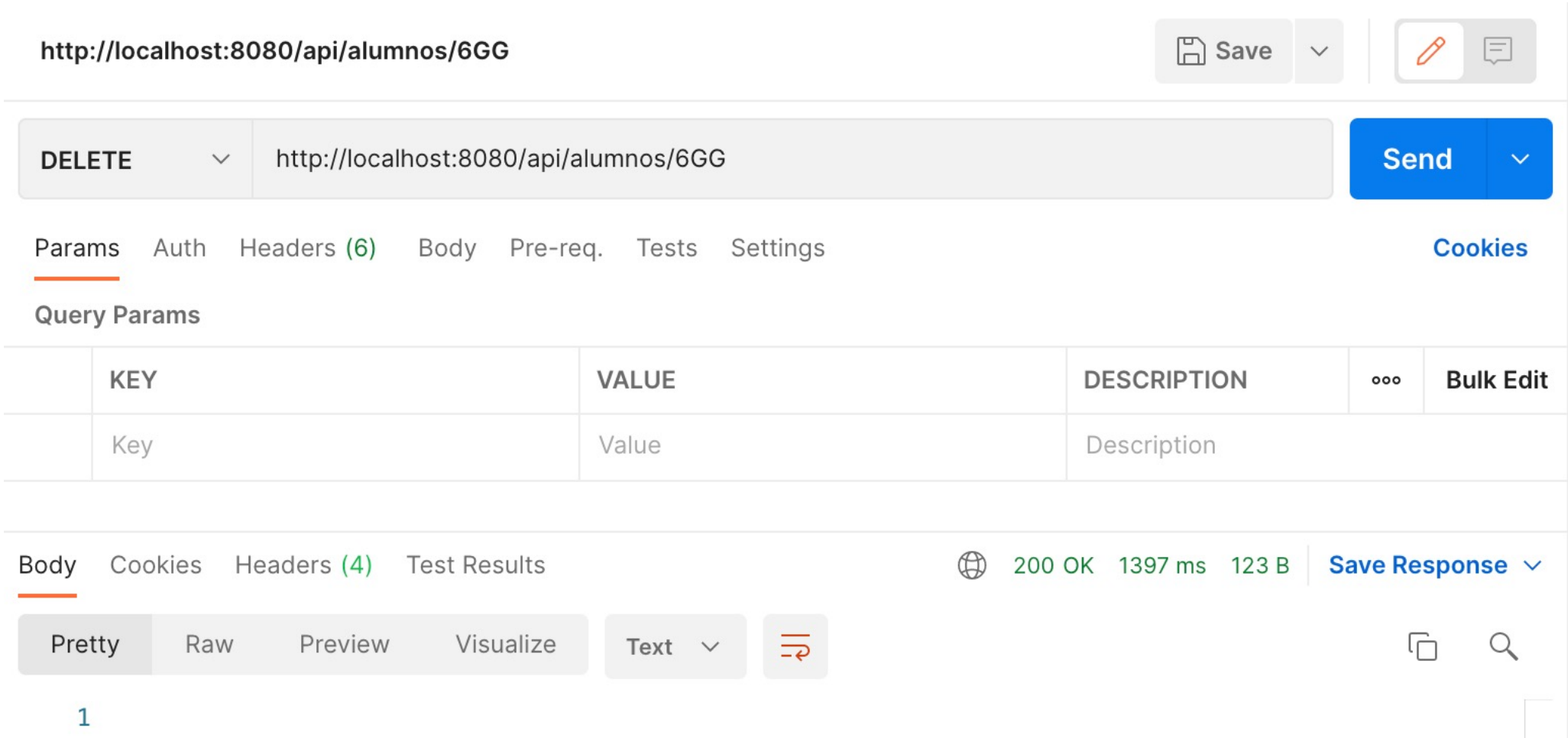
Spring REST DELETE/{ID}



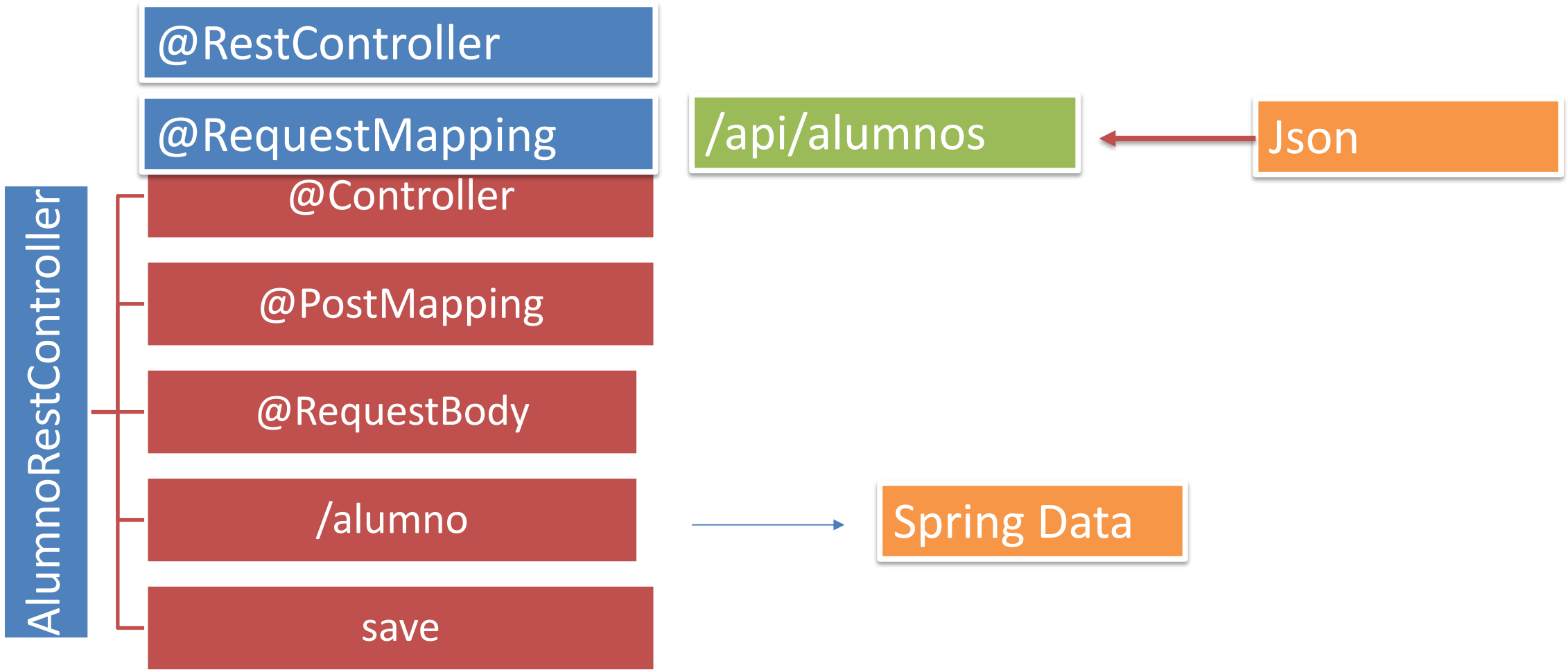
Delete


Undo Typing ⌘Z
Revert File
Save ⌘S
Open Declaration F3
Open Type Hierarchy F4
Open Call Hierarchy ^⌥H
Show in Breadcrumb ⌥⌘B
Quick Outline ⌘O
Quick Type Hierarchy ⌘T
Open With
Show In ⌥⌘W
Cut ⌘X
Copy ⌘C
Copy Qualified Name
Paste ⌘V
Quick Fix ⌘1
Source ⌥⌘S
Refactor ⌥⌘T
Local History
References
Declarations
Add to Snippets...
Coverage As
Run As
Debug As
Profile As
Team
Compare With
Replace With
Toggle Comment ⌘7
Remove Block Comment ^⌘\
Generate Element Comment ⌥⌘J
Correct Indentation ⌘I
Format ⇧⌘F
Format Element
Add Import ⇧⌘M
Organize Imports ⇧⌘O
Sort Members...
Clean Up...
Override/Implement Methods...
Generate Getters and Setters...
Generate Delegate Methods...
Generate hashCode() and equals()...
Generate toString()...
Adds Delegates Methods for a Type's Fields
Project Explorer
addsi01
addsi2
apirest
datos1
datos2
datos3 [boot] [devtools]
src/main/java
mx.unam.dgtic.datos1
Alumno.java
AlumnoCalificacionDTO.java
AlumnoController.java
AlumnoRepository.java
AlumnoRestController.java
Calificacion.java
CalificacionRepository.java
Datos1Application.java
src/main/resources
static
templates
application.properties
data.sql
schema.sql
src/test/java
mx.unam.dgtic.datos1
Datos1ApplicationTests.java
Datos1RESTTest01.java
Datos2ApplicationCalificacior
Datos3ApplicationNamedQue
Datos4ApplicationOrderBy.jav
Datos5ApplicationSort.java
JRE System Library [JavaSE-11]
Maven Dependencies
src
target
HELP.md
mvnw
Problems
<terminated> Datos
/mx/unam/dgtic/datos1/AlumnoRestController.java - Eclipse IDE
os1RESTTest01.java
repositorioAlumno;
ndAll() {
no.findAll();
30:03 – 22:30:17)

DELETE


http://localhost:8080/api/alumnos/6GG
Save
DELETE
http://localhost:8080/api/alumnos/6GG
Send
Params
Auth
Headers (6)
Body
Pre-req.
Tests
Settings
Cookies
Query Params
KEY
VALUE
DESCRIPTION
Bulk Edit
Key
Value
Description
Body
Cookies
Headers (4)
Test Results
200 OK
1397 ms
123 B
Save Response
Pretty
Raw
Preview
Visualize
Text
1

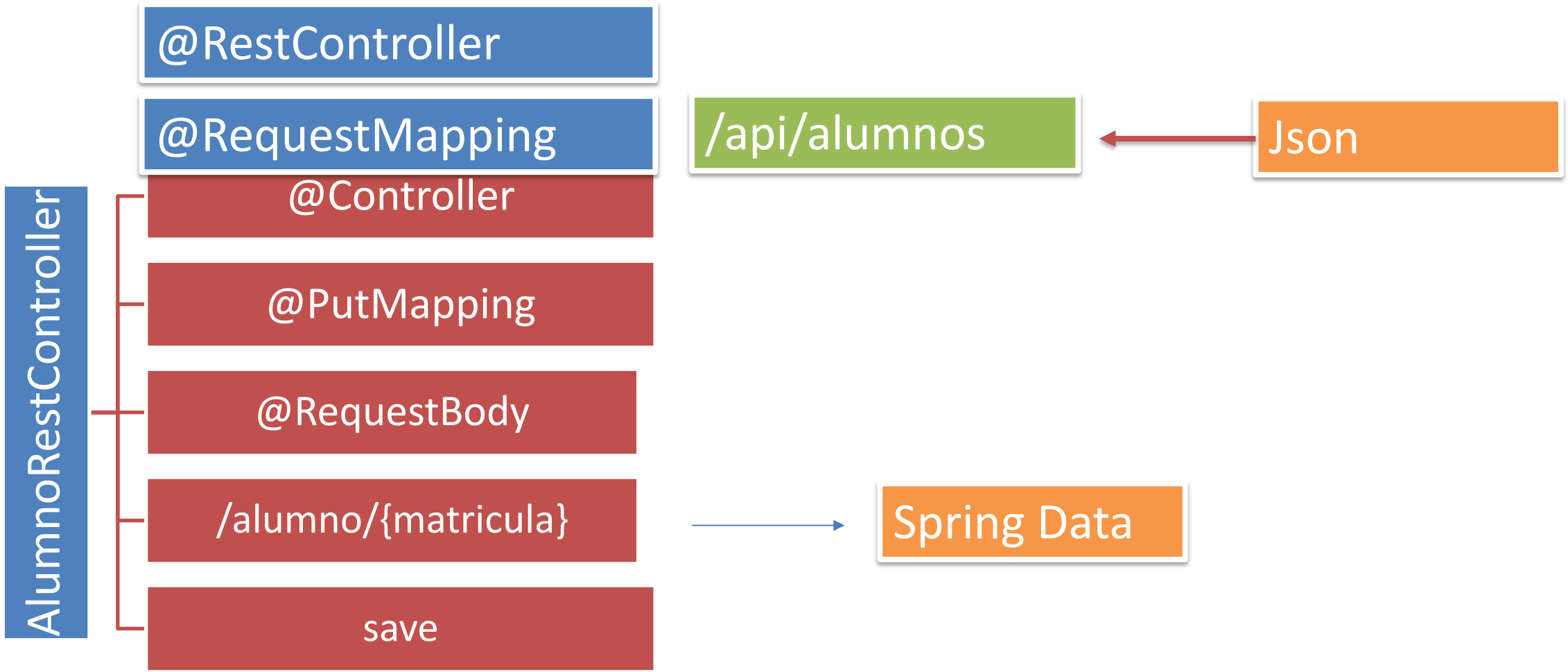
Spring REST POST



POST REST Test

```java
@Test
void insertarUnoRestTest() throws JsonProcessingException {

    Alumno alumno = new Alumno("7GG", "Daniela", "Rios", new Date(), 1.62);
    ObjectMapper mapeador = new ObjectMapper();
    String objetoJsonString = mapeador.writeValueAsString(alumno);

    System.out.println(objetoJsonString);

    given().header("Content-type", "application/json")
    .and().body(objetoJsonString)
    .post("api/alumnos").then().statusCode(200);

    get("api/alumnos/7GG").then().body("nombre", equalTo("Daniela"));

}
```

Spring REST PUT



Contacto

Dr. Omar Mendoza González

omarmendoza564@aragon.unam.mx