



# Proyectos de API REST de Spring Boot con ejemplos de código

por Ranga Karanam MVB · 12 y 18 de enero · Zona de integración

La Zona de integración se le brinda en asociación con Cloud Elements . ¿Qué hay debajo de la superficie de una integración API? Descargue The Definitive Guide to API Integrations para comenzar a construir una estrategia de API.

Esta guía lo ayudará a comprender los proyectos API REST con ejemplos de código en GitHub. Tenemos varios artículos que explican estos proyectos. Estos ejemplos de código ayudarán a los principiantes y expertos a aprender y adquirir experiencia en el desarrollo de Servicios RESTful con Spring Boot.

## Proyectos de API REST de Spring Boot - Ejemplos de código en GitHub

Veamos los artículos en las siguientes secciones

- Proyectos que crean la API REST básica
- Agregar pruebas de unidad e integración a servicios RESTful
- Asegurar los servicios RESTful
- Características básicas del servicio RESTful
- Funciones avanzadas del servicio RESTful

## Proyectos que crean la API REST básica

Exploramos un par de opciones para crear Servicios RESTful con Spring Boot:

- Uso de Spring Boot Web Starter
- Usando Spring Data REST Starter

Usando Spring Boot Web Starter, tenemos artículos con

- Un artículo que se centra en los fundamentos de un servicio REST
- Un proyecto que expone una API CRUD de extremo a extremo para un recurso completo. El proyecto usa JPA (Hibernate) para conectarse a una base de datos H2 en memoria.

TÍTULO	CATEGORÍA	URL	GITHUB
Crear servicio REST con Spring Boot	API REST	URL	Código de proyecto en Github
Introducción a Spring Data Rest - Crear API RESTful a velocidad F1	Datos de primavera	URL	Código de proyecto en Github
Crear una API / servicio REST CRUD con Spring Boot, JPA e Hibernate	API REST	URL	Código de proyecto en Github

## Agregar pruebas de unidad e integración a servicios RESTful

Las pruebas de automatización (incluidas las pruebas de unidad e integración) son una parte clave para garantizar que sus servicios y API evolucionen durante un período de tiempo.

Consideremos un StudentController que depende de StudentService.

En la prueba de unidad

- Vamos a simular el StudentService usando Mockito
- Usaremos el framework Mock MVC para lanzar solo StudentController

---

**Una parte clave de las pruebas unitarias es restringir el alcance a un mínimo. En esta prueba unitaria, queremos probar solo los métodos en `StudentController` .**

---

En la prueba de integración:

- Lanzaremos la aplicación Spring Boot completa usando `@SpringBootTest`
- Invocaremos los métodos de servicio usando `TestRestTemplate`
- Afirmaremos los resultados usando un gran marco de afirmación JSON - `org.skyscreamer.jsonassert.JSONAssert`

---

**Una parte clave de las pruebas de integración es probar todas las capas en la aplicación.**

---

TÍTULO	CATEGORÍA	URL	GITHUB
Servicios de descanso de pruebas unitarias con Spring Boot y JUnit	API REST	URL	Código de proyecto en Github
Escritura de pruebas de integración para servicios de descanso con Spring Boot	API REST	URL	Código de proyecto en Github
Spring Boot - Prueba unitaria y burla con Mockito y JUnit	Prueba de unidad de arranque de primavera	URL	Código de proyecto en Github

## Asegurar los servicios RESTful

En el artículo siguiente, implementamos seguridad básica para una API REST.

TÍTULO	CATEGORÍA	URL	GITHUB
Servicios de descanso seguros y aplicaciones web con Spring Boot Security Starter	Fundamentos de Spring Boot	URL	Código de proyecto en Github

## Características básicas del servicio relajante

Las excelentes API REST tienen:

- Excelente manejo de excepciones: le gustaría devolver la respuesta correcta con el estado de respuesta exacto según la situación.
- Validación adecuada: le conviene validar las cosas correctas y devolver un mensaje que ayude al consumidor a comprender lo que falló.
- Documentación actual: un consumidor debe comprender cómo usar su API. La gran documentación hace que su consumidor y su trabajo sean fáciles.

Los artículos a continuación exploran estas características en profundidad.

TÍTULO	CATEGORÍA	URL	GITHUB
Spring Boot Exception (Error) Manejo de servicios RESTful	API REST	URL	Código de proyecto en Github
Implementación de validación para servicios RESTful con Spring Boot	API REST	URL	Código de proyecto en Github
Spring Boot and Swagger - Documentación de	API REST	URL	Código de proyecto en Github

## Características avanzadas de REST API

En el siguiente nivel, las API REST tienen:

- **HATEOAS** - HATEOAS significa "Hipermedia como motor del estado de la aplicación". Cuando se soliciten algunos detalles de un recurso, proporcionará los detalles del recurso, así como los detalles de los recursos relacionados y las posibles acciones que puede realizar en el recurso. Por ejemplo, al solicitar información sobre un usuario de Facebook, un servicio REST puede devolver lo siguiente:
  - Detalles de usuario.
  - Enlaces para obtener sus publicaciones recientes.
  - Enlaces para obtener sus comentarios recientes.
  - Enlaces para recuperar la lista de sus amigos.
- **Negociación de contenido**: ¿por qué JSON debe ser el único formato de intercambio de datos compatible? ¿Qué pasa si su consumidor ama (o está atrapado con) XML? La Negociación de contenido lo ayuda a admitir múltiples formatos de intercambio de datos para su API RESTful.
- **Versiones**: a medida que su API evoluciona con sus necesidades y las de sus consumidores, tendría la necesidad de tener múltiples versiones de la misma API Resouce. ¿Cómo manejas eso?

Los artículos a continuación exploran estas características en profundidad.

Spring Boot - HATEOAS para servicios RESTful	API REST	URL	Código de proyecto en Github
Spring Boot and Content Negotiation - Representaciones XML y JSON	API REST	URL	Código de proyecto en Github

Servicios RESTful de control de versiones	API REST	URL	Código de proyecto en Github
---	----------	-----	------------------------------

## Otras referencias


- Otros excelentes cursos en in28Minutos
- Spring - spring-framework-tutorial-para-principiantes
- Spring MVC - <https://www.youtube.com/watch?v=BjNhGaZDroY>
- Spring Boot - <https://www.youtube.com/watch?v=PSP1-2cN7vM>
- Eclipse - <https://www.youtube.com/watch?v=s4ShbtOHMCA>
- Maven - <https://www.youtube.com/watch?v=oCFWeVgzsqY>
- JUnit - <https://www.youtube.com/watch?v=o5k9NOR9lrI>
- Mockito - <https://www.youtube.com/watch?v=d2KwvXQgQx4>

---

Tu API no es suficiente. Conozca por qué (y cómo) los principales proveedores de SaaS están convirtiendo sus productos en plataformas con integración de API en el libro electrónico, Build Platforms, Not Products from Cloud Elements.

---

Temas: AUTOMATIZACIÓN, INTEGRACIÓN, ARRANQUE DE PRIMAVERA, API DE DESCANSO

Publicado en DZone con el permiso de Ranga Karanam , DZone MVB . [Vea el artículo original aquí.](#)   
Las opiniones expresadas por los contribuidores de DZone son suyas.

## Obtén lo mejor de la integración en tu banda de

# manténgase de entrada.

Manténgase actualizado con el boletín de integración  
bisemanal de DZone. VER UN EJEMPLO

SUSCRIBIR

## Integration Partner Resources

---

Modernizing Application Architectures with Microservices and  
APIs

CA Technologies



API Strategy and Architecture eBook

CA Technologies



O'Reilly Microservice Architecture Book: Aligning Principles,  
Practices and Culture

CA Technologies



The Definitive Guide to API Integrations: Explore API Integrations  
Below the Surface [eBook]

Cloud Elements

















