

## 1.0.1. API REST Java + Spring Boot

### Enunciado

Se requiere implementar un API REST de acuerdo a la especificación [OAS](#) del archivo adjunto swagger.yaml. Se requiere que el desarrollo cumpla con los siguientes requerimientos.

Requerimientos	
Lenguaje	Java
Framework	Spring Boot
Versionado	Git

Además debe contener, en la raíz del proyecto, un archivo README.md explicando como ejecutar el proyecto. El resto de las definiciones quedan a criterio del candidato.

### Librerías útiles

Descripción	
Mapeo de objetos	<ul style="list-style-type: none"><li>MapStruct</li></ul>
Base de datos en memoria	<ul style="list-style-type: none"><li>H2</li></ul>
ORM	<ul style="list-style-type: none"><li>Hibernate</li></ul>
Acceso a datos	<ul style="list-style-type: none"><li>Spring Data JPA</li></ul>

### Herramientas

Descripción	Recomendadas	Alternativas
IDE	<ul style="list-style-type: none"><li>IntelliJ</li></ul>	<ul style="list-style-type: none"><li>Eclipse</li><li>Spring Tools</li></ul>
Alojamiento de código	<ul style="list-style-type: none"><li>Bitbucket</li><li>GitHub</li><li>Gitlab</li></ul>	
Gestión de dependencias	<ul style="list-style-type: none"><li>Gradle</li></ul>	<ul style="list-style-type: none"><li>Maven</li></ul>
Editor de código	<ul style="list-style-type: none"><li>Visual Studio Code</li></ul>	

### Otras herramientas útiles

Descripción	Herramienta
Scaffolding	<ul style="list-style-type: none"><li>Spring Initializr</li></ul>

Visualización OAS	<ul style="list-style-type: none"> <li>• <a href="#">Swagger Editor</a></li> </ul>
Pruebas de APIs REST	<ul style="list-style-type: none"> <li>• <a href="#">Postman</a></li> <li>• <a href="#">Insomnia</a></li> </ul>

## Otros enlaces útiles

Enlace	Descripción
<a href="https://swagger.io/specification/">https://swagger.io/specification/</a>	Especificación OAS
<a href="https://openapi.tools/">https://openapi.tools/</a>	Herramientas OAS
<a href="https://www.baeldung.com/">https://www.baeldung.com/</a>	Tutoriales y ejemplos implementaciones Java/Spring Boot

## Características opcionales

A continuación se listan características no requeridas

Autenticación	Los mecanismos mas comunes de autenticación en APIs REST son autenticación <a href="#">Basic</a> y autenticación Bearer con JWT(JSON Web Token)
Autorización	El mecanismo mas común de autorización en APIs REST es JWT ya que es stateless
Despliegue	No se requiere desplegar el servicio en un servidor pero si se desea hacerlo hay algunas plataformas que ofrecen el servicio gratuitamente por tiempo limitado y/o para proyectos personales como AWS, Heroku o GCP
Pruebas unitarias	<a href="#">La herramienta mas utilizada es JUnit. Además se suele utilizar Mockito para crear mocks</a>
Base de datos	Es suficiente con utilizar una base de datos en memoria como H2 o SQLite pero opcionalmente se puede implementar con un motor <a href="#">PostgreSQL</a> , <a href="#">SQL Server</a> o <a href="#">MySQL</a>
Versionado de Scripts SQL	<a href="#">Flyway</a> o <a href="#">Liquibase</a>
Contenerizado	Hoy en día es bastante común el contenerizado de servicios mediante <a href="#">Docker</a>
Validación de datos	JSR 380 - Bean Validation 2.0. Ejemplo <a href="https://www.baeldung.com/javax-validation">https://www.baeldung.com/javax-validation</a>

## Entrega

Enviar el enlace del repositorio público por correo.

## Evaluación

La resolución de los desafíos será evaluada por personal del equipo de Arquitectura del NBCH. Opcionalmente se puede solicitar a los candidatos una call en la que deberán explicar la resolución del ejercicio.