

Swagger

¿Qué es Swagger?

Swagger es un conjunto de herramientas de software de código abierto desarrollado por SmartBear Software, diseñado para facilitar el diseño, construcción, documentación y consumo de servicios web RESTful.

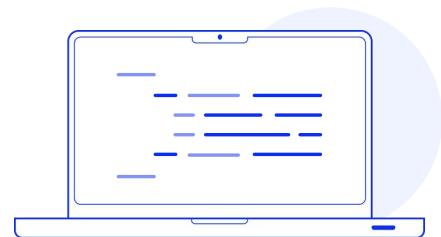
Estas herramientas permiten a los desarrolladores crear API de manera estandarizada, promoviendo buenas prácticas y agilizando el proceso de desarrollo.

Además, simplifican la generación de documentación interactiva y clara, mejorando la comprensión y el uso de las API por parte de los usuarios.

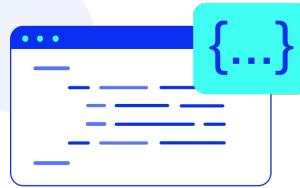


Características de Swagger

- **Documentación interactiva:** genera documentación interactiva de la API que permite a los usuarios ver todos los *endpoints* disponibles, junto con sus métodos HTTP, parámetros de entrada y tipos de datos de respuesta. La interfaz interactiva permite probar los endpoints directamente desde la documentación.
- **Especificación estándar:** define un formato estandarizado para describir las API RESTful, conocido como Swagger Specification (ahora conocido como OpenAPI Specification). Esto asegura que la documentación sea coherente y fácilmente entendible por cualquier desarrollador.



- **Generación de código:** puede generar automáticamente clientes en varios lenguajes de programación, así como servidores esqueléticos, a partir de la especificación de la API. Esto reduce significativamente el tiempo necesario para desarrollar integraciones con la API.
- **Explorador de API:** Swagger UI es una herramienta de exploración de API que permite a los desarrolladores interactuar con la API directamente desde la documentación. Esto facilita la comprensión de cómo funciona la API y cómo se deben realizar las llamadas.
- **Compatibilidad con OpenAPI:** soporta la especificación OpenAPI, que es la evolución del formato Swagger. OpenAPI es ahora mantenido por la OpenAPI Initiative y es ampliamente adoptado en la industria para la documentación de API.



- **Herramientas de validación:** Swagger incluye herramientas para validar las especificaciones de la API y asegurarse de que cumplen con el estándar definido. Esto ayuda a evitar errores e inconsistencias en la documentación.
- **Extensibilidad:** es altamente extensible, lo que permite a los desarrolladores agregar nuevas funcionalidades o personalizar las herramientas existentes para satisfacer necesidades específicas.



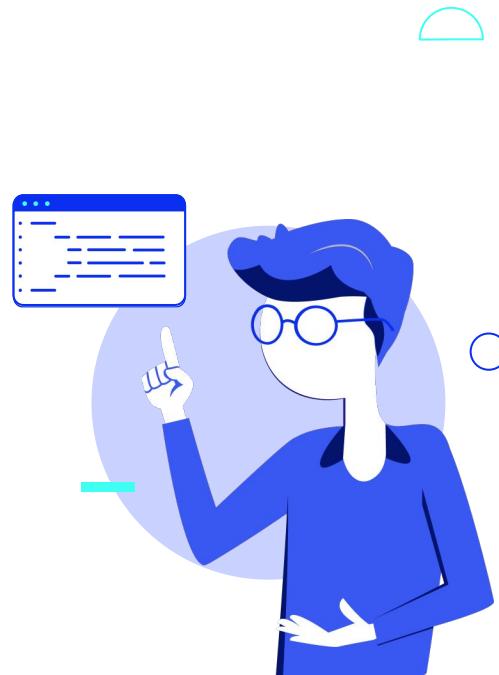
Componentes

Swagger UI	Swagger Editor	Swagger Codegen	Swagger Hub
Una interfaz gráfica basada en web que permite a los usuarios interactuar con la API directamente desde la documentación.	Un editor en línea que permite a los desarrolladores crear y editar especificaciones de API en formato OpenAPI.	Una herramienta que genera clientes de API, servidores esqueléticos y documentación en varios lenguajes de programación a partir de las especificaciones de Swagger.	Una plataforma colaborativa basada en la nube para diseñar, documentar y compartir especificaciones de API.

Documentar API REST en Spring Boot con Swagger

En las siguientes secciones, se explorará cómo integrar y configurar Swagger en una aplicación Spring Boot.

Primero, se verá cómo agregar Swagger al proyecto, y luego, cómo ajustar su configuración según necesidades y requerimientos.



Agregar Swagger a Spring Boot

```
<name>demo</name>
<description>Demo project for Spring Boot</description>
<properties>
<java.version>17</java.version>
</properties>
<dependencies>
<dependency>
<groupId>org.springframework.boot</groupId>
</dependency>
<artifactId>spring-boot-starter-web</artifactId>
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-test</artifactId>
<scope>test</scope>
</dependency>
<groupId>com.google.code.gson</groupId>
<dependency>
```

...



...

```
<artifactId>gson</artifactId>
<version>2.8.8</version>
</dependency>
<!-- Swagger dependencies
<dependency>
<groupId>org.springdoc</groupId>
<artifactId>springdoc-openapi-starter-webmvc-ui</artifactId>
<version>2.1.0</version>
<dependency>
</dependencies>
<build>
<plugins>
<plugin>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-maven-plugin</artifactId>
</plugin>
</plugins>
</build>
<project>
```

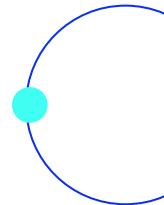
El código presentado en la *slide* anterior, `pom.xml`, se agregan las dependencias necesarias para integrar Swagger en el proyecto mediante la biblioteca `springdoc-openapi`. Esta configuración incluye lo siguiente:



Dependencias principales	Swagger Dependencies
<ul style="list-style-type: none">• <code>spring-boot-starter-web</code>: para desarrollar la API REST.• <code>spring-boot-starter-test</code>: para realizar pruebas.• <code>gson</code>: para la serialización/deserialización de JSON.	<ul style="list-style-type: none">• <code>springdoc-openapi-starter-webmvc-ui</code>: que habilita la interfaz de Swagger UI para explorar la API.• Con esta configuración, podrás acceder a la documentación interactiva de tu API localmente en <code>/swagger-ui.html</code>.

Configurar Swagger en la aplicación

```
import io.swagger.v3.oas.models.OpenAPI;
import io.swagger.v3.oas.models.info.Info; import
org.springframework.context.annotation.Bean
; org.springframework.context.annotation.Configuration; import
@Configuration
public class SwaggerConfig {
@Bean
public OpenAPI customOpenAPI() {
return new OpenAPI()
.info(new Info()
.title("API Documentation")
.version("1.0")
.description("API documentation for the project"));
}
}
```



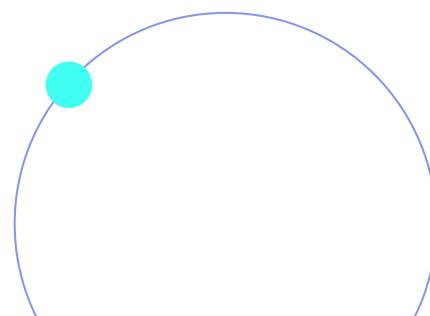
Una vez agregadas las dependencias necesarias, es fundamental configurar Swagger en el proyecto Spring Boot para que genere automáticamente la documentación de la API.

Este código define una clase de configuración que personaliza la instancia de OpenAPI mediante un bean.

En el ejemplo presentado en la slide anterior, se establece un título, versión y descripción personalizados para la documentación de la API.

La anotación `@Configuration` indica que esta clase proporciona configuraciones a Spring, mientras que `@Bean` define el bean de OpenAPI, el cual será utilizado por Swagger para generar la interfaz de documentación.

Este paso permite que la documentación de la API esté disponible de manera clara y accesible en la interfaz de Swagger.



Acceder a la documentación de Swagger

Para acceder a la documentación de Swagger
deberás realizar:

1. Iniciar la aplicación Spring Boot.
2. Abrir un navegador web y acceder a <http://localhost:8080/swagger-ui.html> para ver la interfaz de Swagger UI con la documentación de la API.



App local

Una vez que Swagger ha sido configurado correctamente en la aplicación, **se puede acceder a la interfaz de documentación generada automáticamente.**

Esta interfaz permite visualizar y probar los distintos *endpoints* de la API REST localmente. **En el ejemplo, se muestran los controladores producto-controller y persona-controller**, junto con los métodos HTTP asociados, como GET, POST, PUT y DELETE, que permiten interactuar con los recursos productos y personas.

Swagger no solo documenta los *endpoints*, sino que también proporciona una herramienta interactiva para enviar solicitudes directamente desde la interfaz. De esta manera, facilita las pruebas y el desarrollo de la API.



En la siguiente slide podrás ver cómo quedaría

producto-controller**GET** /productos/{id}**PUT** /productos/{id}**DELETE** /productos/{id}**GET** /productos**POST** /productos**GET** /productos/nombre/{nombre}**persona-controller**