

Punto de partida

¡Bienvenidos a la semana 6!

La demanda de pruebas de API está aumentando día a día, ya que ayuda a encontrar errores en las primeras etapas del ciclo de vida del desarrollo de software y a evitar errores importantes en etapas posteriores. Podemos realizar las pruebas de la API de forma manual y programada, es decir, automatizada.

Hemos cubierto las pruebas manuales de la API utilizando Postman. Ahora es el momento de aprender sobre la automatización de las pruebas de la API utilizando una biblioteca Java llamada Rest Assured.

API

Application Program Interface

Es un conjunto de rutinas y normas establecidas y documentadas por una aplicación X, para que otras aplicaciones puedan utilizar las funcionalidades de esta aplicación X, sin necesidad de conocer los detalles de la implementación del software, es decir:

Es la comunicación eficaz y normalizada entre dos sistemas.

Terminología:

HTTP

Es el principal protocolo de comunicación de los sistemas web. Este protocolo nos provee de los siguientes métodos para interactuar con una API.

GET

Recupera los datos.

POST

Envía los datos.

PUT

Altera los datos.

DELETE

Apaga/elimina los datos.

¿Qué es Rest Assured?

Rest-Assured es una **biblioteca Java de código abierto** que se utiliza para probar y validar las API de tipo REST. Los lenguajes dinámicos como Groovy y Ruby son útiles y sencillos para realizar pruebas de API que eran más difíciles en Java. Rest Assured se encarga de aportar simplicidad a la hora de realizar pruebas de APIs utilizando Ruby y Groovy en Java.

Rest Assured es un DSL de Java para simplificar las pruebas de los servicios basados en REST contruidos sobre HTTP Builder. REST Assured admite cualquier método HTTP, pero tiene soporte explícito para POST, GET, PUT, DELETE, OPTIONS, PATCH y HEAD, e incluye la especificación y validación, por ejemplo, de parámetros, cabeceras, cookies y cuerpo fácilmente.

Además, puede utilizarse para validar y verificar la respuesta a estas solicitudes.



Utilizar Rest-Assured



Ventajas

- Es de código abierto, es decir, gratuito.
- La configuración inicial es sencilla antes de llegar a cualquier punto final.
- Fácil análisis y validación de la respuesta en JSON y XML.
- Sigue las palabras clave de BDD como given(), when(), then(), lo que hace que el código sea legible y permite una codificación limpia. Esta función está disponible a partir de la versión 2.0.
- Se pueden comprobar en tiempo real las cabeceras, las cookies, el tipo de contenido, etc.
- Puede integrarse fácilmente con otras librerías Java como TestNG, Junit como Test Framework y Extent Report y Allure Report para la elaboración de informes.



Desventajas



Utilizar Rest-Assured



Ventajas

Desventajas

- No admite la comprobación de las API SOAP de forma explícita.
- Requiere buenos conocimientos de programación en Java.
- No hay informes incorporados. Serenity BDD es una buena opción en este caso.

