

¿Qué es un Servicio Rest?

Conociendo el potencial de Internet y de que se podía enviar información entre una y otra computadora, se creó una especie de arquitectura para este tipo de comunicación y esto era a través de lo que se conocía antes servicios web.

¿Qué son los servicios Web?

Pues el servidor base o el servidor donde estaba toda la información, o una computadora donde estaba toda la información del sistema. Podía crear servicios web para que los demás consuman estos servicios.

Sin embargo, a medida que fueron creciendo los sistemas web, bueno, esto era más sencillo porque ya no necesitabas un sistema de escritorio, sino simplemente construían un sistema web. Este ya estaba alojado en un servidor, tal cual los proyectos que se han hecho durante el curso

Solamente necesitaba un navegador para poder acceder al sistema y esta empieza a utilizar tu base de datos y ya no es necesario hacer esta comunicación entre XML y todo esto de aquí con un sistema web.

Pero ¿qué pasa si ahora necesito comunicar un sistema con otro independiente?

Mediante los servicios REST

Quiere decir que de aquí va a crear un servicio a los servicios web, solo que esta vez va a ser un servicio REST. Y la característica de este servicio REST. Es que es sencillo de implementar y ya no va a trabajar con formatos XML sino con archivos de formato JSON

¿Qué es JSON?

JSON es un formato de texto muy sencillo. Cuando me refiero a un formato de texto es, por ejemplo, si yo quería mostrar dentro de mi HTML. O bueno, si yo quería que en mi navegador se pinte algo, se muestre un esquema que hacíamos, escribíamos dentro de un archivo, un formato de esquemas, y esto lo hacíamos con HTML. Sabíamos cómo ubicar las cosas para que esto se muestre en el navegador.

Bien, pues un JSON es algo similar. Vamos a nosotros formar un texto, esta vez un texto algo así, con cabecera, con todo lo necesario cabecera, el cuerpo.

Pero es un texto que luego una aplicación, ya sea en JavaScript, ya sea una aplicación de Flutter, de Android, la que sea, va tomar todo ese texto de ahí y va a entender su formato, sabe que es un formato de tipo Jameson, va a entender y va a transformar esa información en objetos o clases que luego va a procesar para interactuar con nuestra base de datos.

Arquitectura REST - Servicios Web

Django REST nos pide en su arquitectura ciertos requisitos

- Tener bien separado lo que es el cliente del servidor

Es decir todo aquel proceso donde sea consulta de datos o guardar datos directamente interactuando con la BD tiene que ser un proceso exclusivo del servidor.

Sin embargo todo aquello donde se muestre una interfaz que interactúa con el usuario tiene que ser exclusivamente del cliente, lo que se conoce como frontend y el backend

- Peticiones o procesos sin estado es decir por ejemplo si se ha creado un servicio para que lo consuma el front ya sea de listar clientes, pues éste servicio no debe depender de ningún otro servicio o API

Característica Esencial ==>

- * Tiene una interfaz uniforme, es decir que todos van a acceder al servicio de la misma forma

- * Tiene un Sistema de Capas (desarrollo por capas) ==> hay que respetar las capas y el patrón de diseño

Métodos HTTP

GET- POST- PUT- DELETE

Códigos de Estado que responde REST

- 20X (correcto) ya sea 202, 200 , etc quiere decir que el proceso fue exitoso
- 30X (Caché, redirección) 301, 300, etc quiere decir que la info ya fue enviada varias veces y está en cache y no es necesario hacerlo de nuevo
- 40X (No se encontró) 404, 400 Que no ha encontrado nada/servicio
- 50X (error del servidor) 502, 501 Error del servidor, en caso de que por ejemplo en un listado no se puso correctamente el atributo de un modelo





