

Informe de clase 8: Arquitecturas de datos

Juan Diego Quintero - Angie Rodríguez - Diego Baquero

November 10, 2023

Abstract

El presente informe da cuenta de los temas vistos durante la sesión de clase que tuvo como tema central la arquitectura digital para el tratamiento de datos.

1 Introducción

En la era de la información y la conectividad, el intercambio de datos y la comunicación entre aplicaciones desempeñan un papel crucial en el funcionamiento de nuestro mundo digital. La gran revolución web es la altísima capacidad de comunicar.

En este informe, exploraremos conceptos fundamentales relacionados con la arquitectura REST (Representational State Transfer) y cómo consumir servicios a través de una API (Interfaz de Programación de Aplicaciones). La arquitectura REST es un enfoque ampliamente utilizado en el diseño de sistemas web que se basa en principios y restricciones específicas para crear servicios web eficientes, escalables y fáciles de mantener. Para comprender mejor este concepto, abordaremos los siguientes puntos:

- Concepto de servidor-cliente.
- Qué es una API y sus tipos.
- Qué es REST y RESTful.
- Qué es HTTP, sus verbos más utilizados, métodos y recursos.
- Formatos JSON.
- Qué es Postman y sus códigos de respuesta.

Al explorar estos conceptos y temas, obtendremos una comprensión más profunda de cómo las aplicaciones se comunican a través de la web y cómo se diseñan servicios web efectivos y eficientes. Con esta base, estaremos preparados para abordar los desafíos y oportunidades en el mundo de la integración de aplicaciones y la programación web.

La infraestructura digital es vital en las sociedades actuales, tanto para su bienestar como para su desarrollo. Servicios esenciales, por ejemplo del campo de la salud o de la justicia, utilizan métodos digitales para comunicarse entre entidades interesadas. En

Colombia, varias iniciativas del Estado buscan fomentar el uso de datos, y por tanto, el robustecimiento de la infraestructura digital. Es el caso del Open Data, uno de los más elevados estadios de desarrollo del productos y servicios con base en datos.

2 Conceptos de clase

2.1 Concepto de servidor-cliente

El concepto de servidor-cliente es fundamental en la arquitectura de internet y en las comunicaciones en red. En esta configuración, hay dos roles principales:

- **Servidor:** Es una computadora o dispositivo que proporciona recursos, servicios o datos a través de una red. Responde a las solicitudes de los clientes y cumple con las peticiones que se le hacen.
- **Cliente:** Es la entidad que solicita recursos o servicios al servidor. Puede ser una aplicación o dispositivo que busca información o realiza acciones en el servidor.

2.2 ¿Qué es un API?

API significa Interfaz de Programación de Aplicaciones en inglés (Application Programming Interface). Es un conjunto de reglas y protocolos que permiten que diferentes aplicaciones se comuniquen entre sí. Las API definen los métodos y formatos de datos que las aplicaciones pueden utilizar para solicitar y compartir información.

- **APIs públicas:** Son accesibles para cualquier persona o desarrollador y se utilizan para proporcionar servicios o datos de manera abierta a través de Internet.
- **APIs privadas:** Son restringidas y solo están disponibles para un conjunto limitado de usuarios o aplicaciones, a menudo para uso interno o en entornos de empresas.
- **APIs remotas:** Son aquellas que se encuentran en servidores o sistemas externos y se acceden a través de la red. Pueden estar en la nube o en otros servidores remotos.
- **APIs locales:** Estas APIs se encuentran en la misma máquina o sistema que la aplicación que las utiliza, lo que significa que no es necesario el acceso a través de una red.

2.3 ¿Qué es REST y RESTful?

- **REST (Representational State Transfer):** Es un estilo de arquitectura de software que se basa en una serie de principios y restricciones para diseñar sistemas web. Se utiliza para crear servicios web que sean simples, escalables y fáciles de mantener.
- **RESTful:** Se refiere a un servicio web que cumple con los principios y restricciones de REST.

2.4 ¿Qué es HTTP?

Hypertext Transfer Protocol es un protocolo de comunicación utilizado para la transferencia de datos y se utiliza para la comunicación entre un cliente (como un navegador web) y un servidor web. HTTP permite que los clientes realicen solicitudes de recursos (como páginas web, imágenes, videos o datos) a través de la red, y los servidores responden a esas solicitudes proporcionando los recursos solicitados.

- **Métodos HTTP:** son verbos utilizados para indicar la acción que se debe realizar en un recurso. Algunos métodos comunes son GET (obtener), POST (crear), PUT (actualizar) y DELETE (borrar).
- **Buenas prácticas:** Las buenas prácticas incluyen el uso adecuado de estos métodos según su semántica y la seguridad.
- **Estados más comunes (códigos de respuesta HTTP):** a) 200 OK - Solicitud exitosa. b) 201 Created - Recurso creado exitosamente. c) 204 No Content - Solicitud exitosa sin contenido en la respuesta. d) 400 Bad Request - Error en la solicitud del cliente. e) 401 Unauthorized - Falta de autenticación o credenciales incorrectas.

2.5 Formatos JSON

JSON (JavaScript Object Notation) es un formato de intercambio de datos ligero y legible por humanos. Se utiliza comúnmente en las respuestas de las API para estructurar datos en forma de objetos y arrays.

2.6 ¿Qué es Postman y sus códigos de respuesta?

- **Postman:** Es una herramienta que permite probar y desarrollar APIs. Facilita la creación y el envío de solicitudes HTTP a APIs y muestra las respuestas, lo que es útil para la depuración y la prueba de servicios web.
- Los códigos de respuesta de Postman son los mismos que los códigos de respuesta HTTP.

3 Bibliografía adicional

Durante la clase, el profesor presentó una serie de videos que explicaban de forma más intuitiva lo mencionado en este documento. Uno de ellos es el siguiente: <https://www.youtube.com/watch?v=aoOKxUOhg18>

Con esta información se pueden ampliar los conocimientos acá presentados y crear nuevas preguntas que den cuenta de la gran capacidad de estos sistemas de comunicación masiva.

4 Conclusiones

Hemos explorado una serie de conceptos clave que abarcan desde la arquitectura subyacente hasta las herramientas prácticas utilizadas en el proceso, comenzamos por desglosar

el concepto de servidor-cliente, que es la base de la comunicación en la web, donde los servidores proporcionan recursos y los clientes realizan solicitudes para obtenerlos. Luego, profundizamos en qué es una API y cómo se clasifican, ya sea como públicas o privadas, y si están alojadas en servidores remotos o locales.

La arquitectura REST y su derivado RESTful se presentaron como enfoques para diseñar servicios web eficientes y escalables, basados en principios y restricciones específicas. Comprender cómo funciona HTTP, los métodos HTTP más utilizados y los códigos de respuesta comunes es esencial para el intercambio de datos en la web. Además, destacamos la importancia de los formatos de datos como JSON, que son ampliamente utilizados en las respuestas de las API para estructurar información de manera legible por humanos y máquinas.

La herramienta Postman, que permite probar y desarrollar APIs, se presentó como una solución útil para depurar y probar servicios web. A través de Postman, los desarrolladores pueden enviar solicitudes HTTP y analizar las respuestas.