

Oscar Ariel Quintana Merino

TID41M

Tecnologías de la información: Desarrollo de Software Multiplataforma

Aplicaciones Web Orientadas a Servicios

Reporte de investigación de las características y lenguajes utilizados en las aplicaciones web orientadas a servicios.

Docente: Addai Guerrero

# **Servicios Web en la Nube**

## **Concepto:**

## **Servicios web:**

Los servicios web son componentes esenciales en el ámbito de la tecnología de la información y la comunicación. Estos son sistemas autónomos y modulares que se utilizan para mejorar la flexibilidad y la integración de procesos empresariales al permitir la comunicación entre aplicaciones que de otro modo no podrían interactuar.

Los servicios web pueden incluir el propio servicio o el cliente que accede al servicio.

Los servicios web son aplicaciones web que permiten aumentar la flexibilidad de los procesos empresariales al integrarse con aplicaciones que de otra forma no se comunicarían.

Los servicios web reflejan el enfoque SOA (service-oriented architecture - arquitectura orientada a servicios) en la programación. Este enfoque está basado en la creación de aplicaciones detectando e implementando los servicios de red disponibles o invocando las aplicaciones disponibles para que realicen una tarea.

Los servicios web dependen de las tecnologías de transporte existentes (como HTTP) y las técnicas de codificación de datos estándar (como XML, Extensible Markup Language) para invocar la implementación.

* **Web Services Description Language (WSDL)**
  + WSDL es el archivo basado en XML que describe el servicio web. La solicitud de servicio web utiliza este archivo para enlazarse con el servicio.
  + WSDL (Web Services Description Language) es un lenguaje de descripción basado en XML (Extensible Markup Language). Este lenguaje se remitió a W3C (World Wide Web Consortium) como el estándar del sector para describir servicios web. La potencia de WSDL se deriva de dos principios básicos de arquitectura: la posibilidad de describir un conjunto de operaciones empresariales y la posibilidad de separar la descripción en dos unidades básicas.
* **SOAP**
  + SOAP es el protocolo basado en XML que utiliza la solicitud de servicio web para invocar el servicio.
  + SOAP es una especificación para el intercambio de información estructurada en un entorno distribuido descentralizado. Como tal, representa la principal forma de comunicación entre los tres actores clave de una arquitectura orientada a los servicios (SOA): el proveedor de servicios, el solicitante de servicios y el intermediario de servicios. El principal objetivo de su diseño es ser simple y ampliable.
* **REST**
  + REST o "Representational State Transfer" (Transferencia de Estado Representacional) es un estilo arquitectónico para el diseño de sistemas de software distribuidos, como aplicaciones web. Fue propuesto por Roy Fielding en su tesis doctoral en 2000 y se ha convertido en un enfoque ampliamente utilizado para crear servicios web y APIs (Interfaces de Programación de Aplicaciones).
  + A diferencia de SOAP y WSDL, que son protocolos y lenguajes de descripción específicos para servicios web, REST es un enfoque más simple y basado en principios fundamentales de la web.
  + En REST, todo se considera un recurso, que puede ser cualquier cosa, desde un objeto físico hasta un concepto abstracto.
  + En lugar de utilizar operaciones específicas como "getCustomer" o "createOrder" como en SOAP, REST utiliza operaciones estándar de HTTP, como GET (obtener), POST (crear), PUT (actualizar) y DELETE (eliminar), para interactuar con los recursos.
* **Microservicios**
  + Los microservicios son un enfoque arquitectónico para diseñar sistemas de software en el que una aplicación se divide en una colección de servicios independientes y pequeños, cada uno de los cuales se encarga de una función o característica específica.
  + Cada microservicio es una entidad autónoma que se ejecuta de manera independiente y se comunica con otros microservicios a través de interfaces bien definidas, generalmente a través de protocolos de comunicación como HTTP o gRPC.
  + El concepto de microservicios se basa en la idea de dividir una aplicación monolítica en componentes más pequeños y autónomos, donde cada microservicio se encarga de una tarea particular. Estos servicios suelen comunicarse entre sí a través de interfaces bien definidas, como API REST o mensajes, y pueden ser desarrollados y desplegados de manera independiente. Cada microservicio se ejecuta en su propio proceso o contenedor, lo que proporciona un alto grado de aislamiento y escalabilidad.

# Fuentes de Consulta:

* <https://www.ibm.com/docs/es/was/9.0.5?topic=services-web>
* <https://aws.amazon.com/es/microservices/>
* https://restfulapi.net/