



UNIVERSIDAD
TECNOLÓGICA
DE PANAMÁ



Facultad de Ingeniería de
Sistemas Computacionales

Kexy Rodríguez

kexy.rodriguez@utp.ac.pa

Desarrollo de Software VII

Desarrollo de Software VII

Contenido

- Definición de Web Services
- Funcionamiento conceptual
- Estándares utilizados

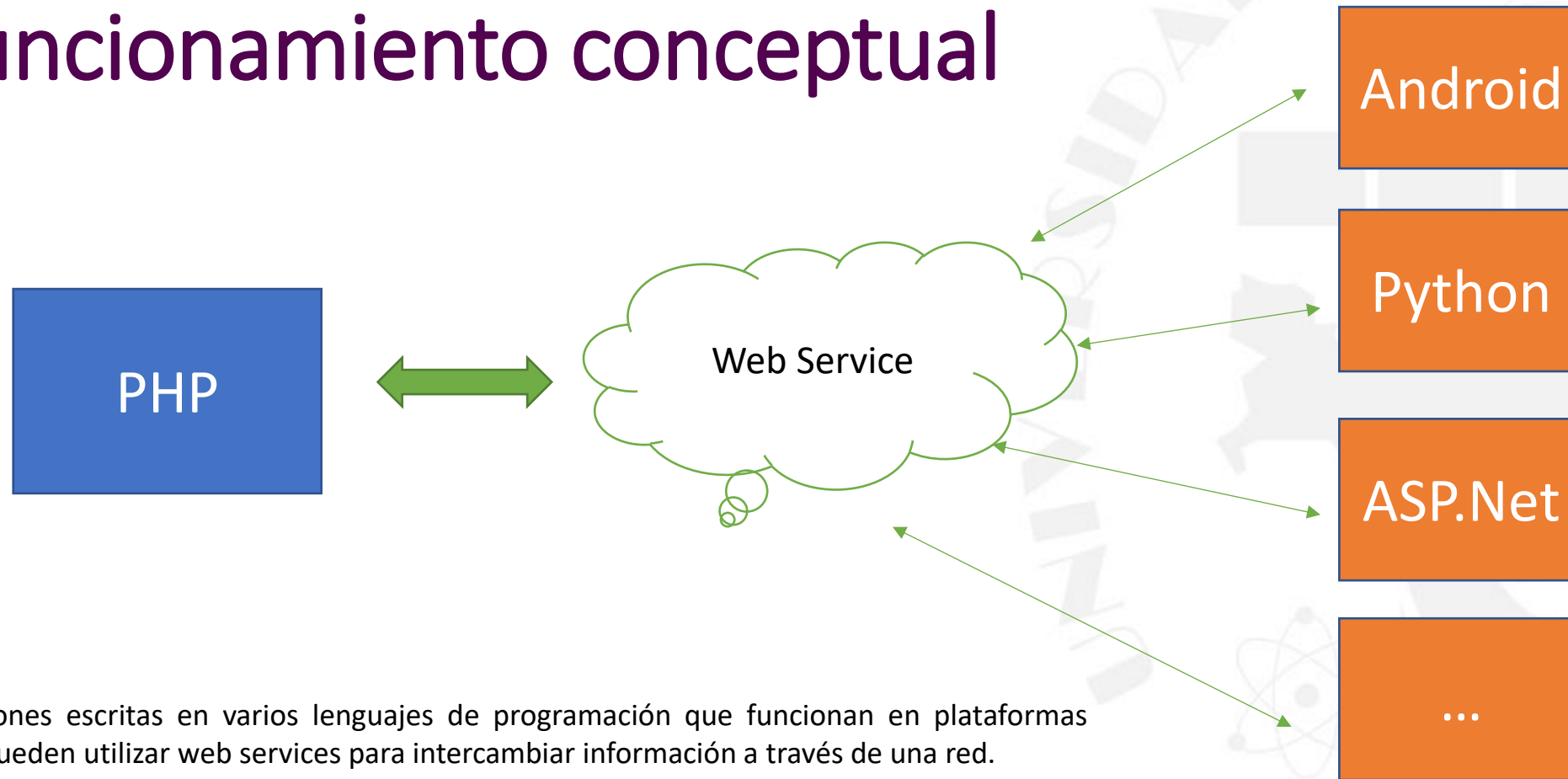


Definición de Web Services

- Un Web Service es un método de comunicación entre dos dispositivos electrónicos en una red. Es una colección de protocolos abiertos y estándares usados para intercambiar datos entre aplicaciones o sistemas. **Las aplicaciones escritas en varios lenguajes de programación que funcionan en plataformas diferentes pueden utilizar web services para intercambiar información** a través de una red. La interoperatividad, por ejemplo, entre Java y Python o Windows y Linux se debe al uso de estándares abiertos (*Diego Lázaró, 2017*).



Funcionamiento conceptual



Las aplicaciones escritas en varios lenguajes de programación que funcionan en plataformas diferentes pueden utilizar web services para intercambiar información a través de una red.

Características

- Ofrecen interoperabilidad entre aplicaciones de software que se ejecutan en distintas plataformas y desarrolladas con distintos marcos de trabajo (framework).
- Pueden ser utilizados para conformar nuevos servicios que ofrecen un nuevo valor agregado.
- Se emplea un mecanismo de intercambio de mensajes entre peticiones y respuestas.



Estándares utilizados

La interoperabilidad se consigue mediante la adopción de estándares abiertos y los responsables de esto son OASIS y W3C.

- Web Services Protocol Stack
- XML (Extensible Markup Language)
- SOAP (Simple Object Access Protocol)
- XML-RPC (XML Remote Producer Call)
- HTTP, FTP o SMTP
- WSDL (Web Services Description Languages)
- UDDI (Universal Description, Discovery and Integration)
- WS-Security (Web Service Security)



Tipos de Web Services que existen

XML-RPC

- Es un protocolo de llamada a procedimiento remoto que usa XML para codificar los datos y HTTP como protocolo de transmisión de mensajes
- Es una especificación y un conjunto de implementaciones que permiten que el software se ejecute en sistemas operativos dispares, que se ejecuten en diferentes entornos para realizar llamadas a procedimientos a través de Internet.
- Es una llamada a procedimiento remoto usando HTTP como transporte y XML como codificación. XML-RPC está diseñado para ser lo más simple posible, al mismo tiempo que permite transmitir, procesar y devolver estructuras de datos complejas.

<https://www.php.net/manual/es/book.xmlrpc.php>



Tipos de Web Services que existen

SOAP (Protocolo simple orientado a objetos)

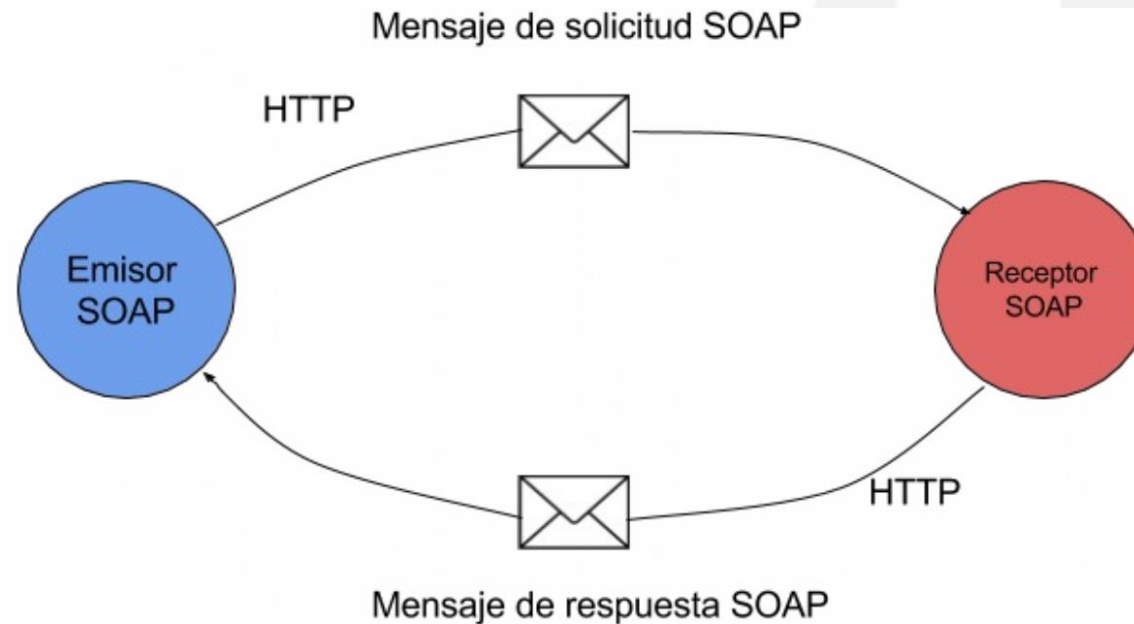
- Es un protocolo ligero genérico sin estado, independiente de la plataforma, basado en XML que utiliza HTTP como medio de transporte y puede utilizarse para desarrollar entornos informáticos complejos distribuidos.
- La aplicación puede comunicarse directamente entre sí a través de Internet mediante SOAP.
- Permite el intercambio de datos entre aplicaciones web heterogéneas.
- SOAP admite RPC como DCOM o CORBA, pero utiliza XML Open Standard para el intercambio de datos entre aplicaciones distribuidas homogéneas o heterogéneas.
- Debido a la independencia de la plataforma, la independencia del idioma y el uso de la comunicación de mensajes, el protocolo SOAP es un mecanismo robusto y estandarizado en redes homogéneas o heterogéneas.
- En la llamada a procedimiento remoto de SOAP, el cliente envía un mensaje de solicitud al servidor. El servidor a su vez envía el mensaje de respuesta al cliente.
- XML es la base de las actividades SOAP. Todos los mensajes SOAP se transmiten en formato XML y es un estándar para la representación e intercambio de datos en forma estructurada entre sistemas.



Tipos de Web Services que existen

SOAP (Protocolo simple orientado a objetos)

- Protocolo estándar para la comunicación entre objetos intercambiando datos XML.



Tipos de Web Services que existen

REST

- REST es el acrónimo de **RE**presentational **St**ate **T**ransfer. Es un estilo arquitectónico para sistemas hipermedia distribuidos y fue presentado por primera vez por Roy Fielding en 2000.
- Como cualquier otro estilo arquitectónico, REST también tiene sus propias 6 restricciones de guía que deben cumplirse si una interfaz necesita ser referida como RESTful .
- Utiliza el protocolo HTTP (Hypertext Transfer Protocol) existente para realizar llamadas entre computadores.



Tipos de Web Services que existen

REST – Características

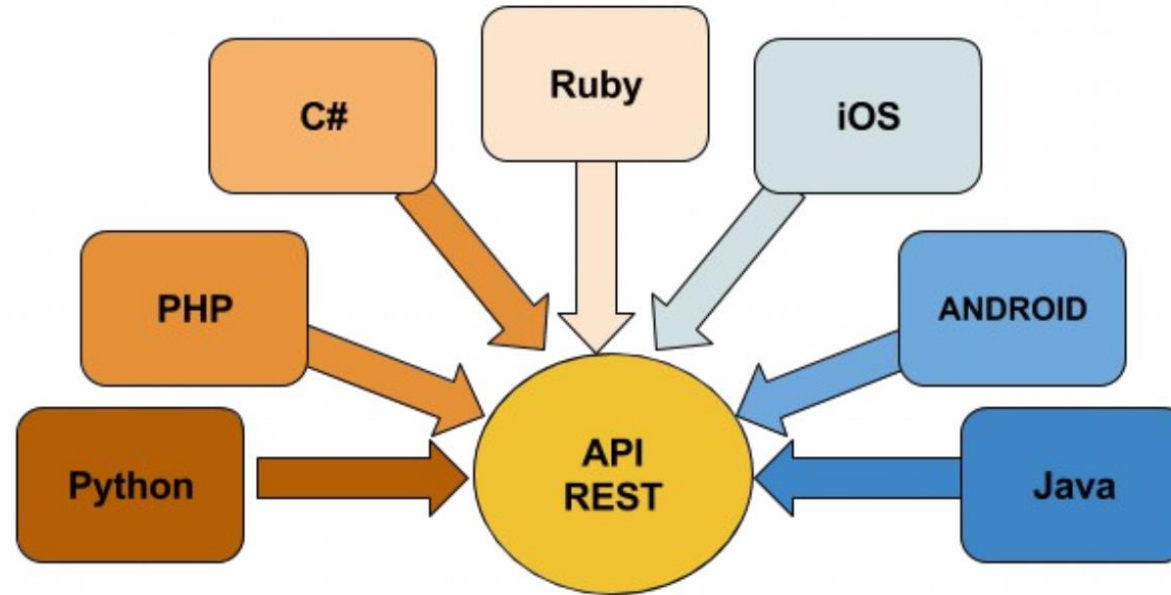
- Independencia de plataforma.
- Independencia de lenguaje de programación.
- Basado en estándares.
- Puede ser usado fácilmente en presencia de firewall.
- Protocolo cliente/servidor sin estado
- Sólo utiliza HTTP
- Métodos HTTP: GET, POST, PUT, DELETE
- Las respuestas a las peticiones se hacen siempre en un lenguaje de intercambio de información, normalmente XML o JSON



Tipos de Web Services que existen

REST – Características

- Lenguajes de programación soportan librerías para peticiones HTTP



Principios rectores de REST

- **Cliente-servidor:** al separar las preocupaciones de la interfaz de usuario de las preocupaciones de almacenamiento de datos, mejoramos la portabilidad de la interfaz de usuario en múltiples plataformas y mejoramos la escalabilidad al simplificar los componentes del servidor.
- **Sin estado:** cada solicitud del cliente al servidor debe contener toda la información necesaria para comprender la solicitud y no puede aprovechar ningún contexto almacenado en el servidor. Por lo tanto, el estado de la sesión se mantiene completamente en el cliente.
- **Almacenable en caché:** las restricciones de caché requieren que los datos dentro de una respuesta a una solicitud estén etiquetados implícita o explícitamente como almacenables en caché o no almacenables en caché. Si una respuesta se puede almacenar en caché, la caché de un cliente tiene derecho a reutilizar esos datos de respuesta para solicitudes posteriores equivalentes.
- **Interfaz uniforme:** al aplicar el principio de generalidad de la ingeniería de software a la interfaz del componente, se simplifica la arquitectura general del sistema y se mejora la visibilidad de las interacciones. Para obtener una interfaz uniforme, se necesitan múltiples restricciones arquitectónicas para guiar el comportamiento de los componentes. REST se define por cuatro restricciones de interfaz: identificación de recursos; manipulación de recursos a través de representaciones; mensajes autodescriptivos; e hipermedia como motor del estado de la aplicación.
- **Sistema en capas:** el estilo del sistema en capas permite que una arquitectura se componga de capas jerárquicas al restringir el comportamiento de los componentes de manera que cada componente no pueda "ver" más allá de la capa inmediata con la que están interactuando.
- **Código a pedido (opcional) :** REST permite ampliar la funcionalidad del cliente descargando y ejecutando código en forma de subprogramas o scripts. Esto simplifica a los clientes al reducir la cantidad de funciones que deben implementarse previamente.

Diferencia entre XML y JSON

JSON y **XML** son dos formatos comunes para intercambiar datos entre diferentes sistemas en la web. La diferencia clave entre JSON y XML es que JSON es un metalenguaje y XML es un lenguaje de marcado.

JSON significa JavaScript Object Notation y ayuda a intercambiar datos entre varias plataformas. JSON se usa principalmente con JavaScript asíncrono (AJAX) y para desarrollar servicios web RESTful. Algunos tipos de datos de JSON son Number, Boolean, String, Array, Object, Null. La sintaxis JSON es utilizar los pares de valores clave y los datos están separados por una coma.

XML significa "Lenguaje de marcado extensible". Tiene una sintaxis basada en etiquetas que es muy similar a HTML, pero no es un reemplazo para HTML. La principal ventaja de XML es que permite a los programadores crear etiquetas por su cuenta. XML es una base de muchas tecnologías web. Algunos de ellos son XHTML, AJAX y servicios web.