

Funcionamiento de Servicios Web con Python*

First Author¹[*PillajoAlexander*], Second Author²[*SimbaEdwin*], Third Author³[*ToasaSteeven*], and Fourth Author⁴[*YauliKatherine*]

Universidad de las Fuerzas Armadas "ESPE" comunicacion@espe-el.espe.edu.ec
<https://espe-el.espe.edu.ec/>

Abstract. Segn Servicios Web y computacion Grid. Los servicios web son aplicaciones modulares basadas en Internet que realizan una tarea comercial especifica y se ajustan a un formato tecnico particular. El formato tecnico garantiza que cada uno de estos servicios comerciales autonomos es una aplicacion que se integrar facilmente con otros servicios para crear un proceso empresarial completo. Esta interoperabilidad permite a las empresas publicar, descubrir y agregar dinmicamente una gama de servicios web a traves de Internet para crear productos innovadores, procesos comerciales y cadenas de valor ms facilmente. El termino Web Services describe una forma estandarizada de integrar aplicaciones WEB mediante el uso de XML, SOAP, WSDL y UDDI sobre los protocolos de la Internet. XML es usado para describir los datos, SOAP se ocupa para la transferencia de los datos, WSDL se emplea para describir los servicios disponibles y UDDI se ocupa para conocer cules son los servicios disponibles.

Keywords: Web(World Wide Web) · XML(Wireless Markup Language) · SOAP(Simple Object Access Protocol) · UDDI(Universal Description, Discovery and Integration).

1 Introduccion

1.1 Funcionamiento de los Servicios Web

Los Web Services permiten a distintas aplicaciones, de diferentes orgenes, comunicarse entre ellos sin necesidad de escribir programas costosos, esto porque la comunicacin se hace con XML. Esta comunicacion debe poder efectuarse entre un programa X, que corre en el computador Alfa, y otro programa Y, que corre en el computador Beta.

Funcionamiento.- Para que esto ocurra el programador a cargo de X debe conocer de Y. A su vez el programador a cargo de Y debe conocer de X, por lo menos en los que se refiere al intercambio de datos. Esto hace que si no hay acuerdo entre el programador de X y el programador de Y, no hay comunicacin posible. La magia de los Web Services est en que el programador de X puede crear

* Ing.Abg.Darwin Alulema.Msc.

un Web Service para transferir datos sin necesidad de conocer al programador Y, ni a los programas que ste tiene a cargo.

1. Service Provider genera el WSDL describiendo el Web Service y registra el WSDL en el directorio UDDI o Service Registry.
2. El Service Requestor o la aplicacin del cliente requiere un Web Service y se pone en contacto con el UDDI para localizar el Web Service.
3. El cliente, basndose en la descripcin descrita por el WSDL, enva un request para un servicio particular al Web Service Listener, que se encarga de recibir y enviar los mensajes en formato SOAP.
4. El Web Service analiza el mensaje SOAP del request e invoca una operacin particular en la aplicacin para procesar el request. El resultado se escribe de nuevo en SOAP en forma de respuesta y se enva al cliente.
5. El cliente analiza el mensaje de respuesta SOAP y lo interpreta o genera un error si ha habido alguno.

```
#!/usr/bin/python
import os
import MySQLdb
os.environ['PYTHON_EGG_CACHE'] = '/tmp/'
from soaplib.wsgi_soap import SimpleWSGISoapApp
from soaplib.service import soapmethod
from soaplib.serializers.primitive import String, Integer, Array

class LoginService(SimpleWSGISoapApp):

    @soapmethod(Integer, _returns=Array(String))
    def getUserData(self, id):
        result = []
        db = MySQLdb.connect('192.168.0.12', 'root', 'password', 'db')
        cursor = db.cursor()
        try:
            cursor.execute('SELECT * FROM usuario WHERE id_usuario = %s', (id))
            lista = cursor.fetchone()
            if (lista == None):
                result = ['0']
            else:
                for n in lista:
                    r=str(n)
                    result.append(r)
        except:
            result = ['-1']
        db.close()

        return result
=====
application=LoginService() █
```

Fig. 1. Programa en Python

1.2 Aporte Científico

1. El desarrollo de la web 2.0 facilita la creatividad, la colaboración y el intercambio entre los usuarios, permitiendo a los científicos, estudiantes y profesionales de la medicina organizarse de manera eficiente y gestionar la información y los recursos que son fundamentales en el cambio rápido en torno a la investigación biomédica de hoy [1].
2. La web 2.0 ayuda a tener acceso a diversas informaciones, permitiendo aumentar nuestra creatividad y productividad científica; además, mejora nuestra lectura crítica al poder tener mayor acceso a trabajos de investigación.
3. Proporciona una interfaz bien definida, ocultando la implementación real. El servicio puede estar implementado en cualquier tecnología, por ejemplo COBOL. La tecnología real de implementación, y los detalles de esta, no son importantes y no deben ser visibles al consumidor del servicio.
4. Promueven la interoperabilidad: La interacción entre un proveedor y un solicitante de servicio está diseñada para que sea completamente independiente de la plataforma y el lenguaje. Esta interacción requiere un documento WSDL para definir la interfaz y describir el servicio, junto con un protocolo de red (generalmente HTTP).

2 Identificación del objeto propenso a fallas en el servicio web

Los servicios web (Alonso, 2003) han agregado un nuevo nivel de Funcionalidad para el servicio web actual dando un primer paso hacia una perfecta integración del software distribuido. Componentes que utilizan muchos estándares web [1]. Los servicios web son muy complejos y variados. Por lo tanto, la gestión de objetos grupo (MOF) [2] define un lenguaje y marco abstractos para la especificación, construcción y gestión de tecnología natural. meta-modelos.

3 Conclusiones

1. Es suficiente un lenguaje de programación con soporte de XML y HTTP para comenzar. En el servidor, se necesita un servidor Web y un motor de servlets. El cliente y el servidor se pueden implementar en entornos distintos. El servicio Web puede habilitar una aplicación existente sin escribir una sola línea de código.
2. Los servicios Web se basan en un conjunto conciso de estándares abiertos basados en XML diseñados para promover la interoperabilidad entre servicios Web y clientes en una variedad de plataformas de sistemas y lenguajes de programación.
3. El servicio web solo va a ser consumido por clientes desarrollados por la misma organización que lo creó.
4. Se pueden agregar servicios Web sencillos para formar servicios Web más complejos, ya sea, utilizando técnicas de flujo de trabajo o llamando a servicios Web de una capa inferior desde una implementación de servicio.

References

1. Holmes KL, Dubinsky EK. Integration of Web 2.0 technologies in the translational research environment. *Med Ref Serv Q* 2009; 28: 309-35.
2. Object Management Group Inc. (OMG). (2011). Meta object facility (MOF) specification v2.4.1.
3. Niclas Ohlsson, Hans Alberg, Predicting Fault-Prone Software Modules in Telephone Switches, *IEEE Trans. SE*, Vol.22, NO. 12, pp 886-894, Dec. 1996.
4. Khosgoftaar T. M., Allen E. B., Early Quality Prediction: A Case Study in Telecommunications, *IEEE Software*, vol. 13, no. 1, pp. 65-71, January 1996.
5. Hempel G, Neef M, Rotzoll D, Heinke W. Study of medicine 2.0 due to Web 2.0?! -risks and opportunities for the curriculum in Leipzig. *GMS Z Med Ausbild* 2013; 30: Doc11.