

**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**  
**FACULTAD DE INGENIERÍA**  
**INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 2**  
**VACACIONES DICIEMBRE 2019**



**“TAREA 1”**

NOMBRE: AUDRIE ANNELISSE DEL CID OCHOA  
CARNÉ: 201801263

GUATEMALA, 18 DE DICIEMBRE DE 2019

## **Tipos de Casos de uso**

### **Casos de uso esenciales:**

Son casos de uso abstractos, sin compromiso con el diseño. Se usa en fase de análisis.

La mayor parte de estos casos de uso sirven como base para los casos de uso reales. En la fase de diseño estos casos de uso disminuyen o se modifican.

### **Casos de uso reales:**

Describe el diseño concreto del caso de uso a partir de una tecnología de entrada y salida, y su implementación global . Se basan en los casos de uso esenciales.

## **¿Cuáles son las diferencias entre los casos de uso esenciales y los casos de uso reales?**

<b>CASOS DE USO ESENCIALES</b>	<b>CASOS DE USO REALES</b>
Es abstracto	Es concreto
No está comprometido con el diseño	Esta comprometido con el diseño
Se usa en fase de análisis	Se usa en fase de diseño

# Web Services

## ¿Qué es un web service?

Es un método de comunicación entre dos aparatos electrónicos en una red. Es una colección de protocolos abiertos y estándares los cuales se usan para intercambiar datos entre aplicaciones o sistemas. Los Web Services pueden ser usados por aplicaciones escritas en varios lenguajes de programación que funcionan en plataformas diferentes para intercambiar información a través de una red.

La base de comunicación entre web services es el lenguaje XML y protocolo HTTP.

## ¿Cómo funciona un Web Service?

El Service Provider genera el WSDL describiendo el Web Service y registrando el WSDL en el directorio UDDI. Luego el service request o aplicación del cliente requiere un web service y se pone en contacto con el UDDI para localizar el Web Service.

El cliente basándose en la descripción descrita por el WSDL envía un reques para un servicio particular al Web Service Listener, que se encarga de recibir y enviar los mensajes en formato SOAP. El web Service analiza el mensaje SOAP del reques e invoca una operación particular en la aplicación para procesar el reques. El resultado se describe de nuevo en el SOAP en forma de respuesta y se envía al cliente.

Por último el cliente analiza el mensaje de respuesta SOAP y lo interpreta o genera un error en caso de haber uno.

## Componentes de los Web services:

- SOAP: Simple Object Access Protocol

Protocolo escrito en XML para intercambio de información entre aplicaciones. Es un formato para enviar mensajes diseñado para servir de comunicación en internet.

- WSDL: Web Services Description Language

Es un lenguaje basado en XML para describir los servicios web y como acceder a ellos. Es un formato estándar para describir un web service.

- UDDI: Universal Description, Discovery and Integration

Es un estándar de XML para describir, publicar y encontrar servicios web. Es un directorio donde las compañías pueden registrar y buscar servicios de web.

### **Arquitectura de los Web Services:**

- Service Discovery: Centraliza servicios de web en un directorio común de registro y provee una funcionalidad sencilla para publicar y buscar. Manejado por el UDDI
- Service Description: Dado que los web services se autodescriben, al localizar uno este proporcionará la información sobre que operaciones soporta y como activarlo. Manejado por WSDL.
- Service Invocation: Mensajes entre el cliente y el servidor. El SOAP especifica como se debe formatear los mensajes request para el servidor y como se debería formatear los mensajes de respuesta.
- Transport: Forma en el que se transmiten los mensajes entre el servidor y el cliente. El protocolo HTTP es el que se encarga de esto.

### **Componentes de los servidores en una aplicación Web Service:**

- Web Service: Realiza las operaciones. Puede estar escrito en lenguaje Java y los clientes invocan las operaciones enviando mensajes SOAP.
- SOAP Engine: Debido a que el Web Service no interpreta SOAP requests, se crean SOAP response y el SOAP engine se encarga del manejo de estos mensajes.
- Application Server: Permite funcionar al SOAP engine, proporciona un espacio libre para aplicaciones que han de ser acreditadas por múltiples clientes.
- HTTP Server: Se usa en caso de que la aplicación server carezca de funcionalidad. Maneja mensajes HTTP.

## Referencias

- <https://diego.com.es/introduccion-a-los-web-services>
- [file:///C:/Users/Audrie/Downloads/libro%20uml%20\(1\).pdf](file:///C:/Users/Audrie/Downloads/libro%20uml%20(1).pdf)