



UNJu | Universidad
Nacional de Jujuy



Argentina
programa
4.0

Unidad 3

Introduccion a la Programacion WEB con Java



Unidad 3 | Programación WEB con Java



- Introducción a JAVA Web.
Arquitectura REST. Métodos HTTP. GET. POST.
- Otros métodos
- JSP (Java Server Pages)
- Presentación del TP Final



Unidad 3 | Programación WEB con Java



Ya estamos en el último módulo del curso

¡No te detengas ahora!

Mantén la determinación y disciplina.

*¡Adelante demostraremos al mundo que somos imparables
en nuestro deseo de **aprender** y **crecer**!*





UNJu | Universidad
Nacional de Jujuy



Argentina
programa
4.0

Arquitectura REST

Métodos HTTP. GET y POST

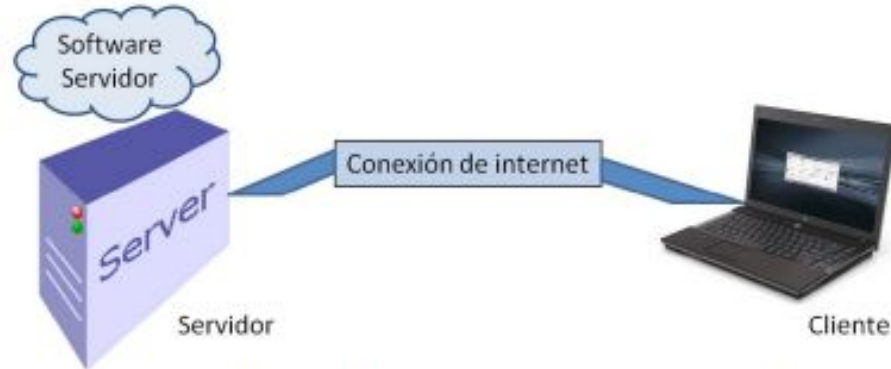


¿Que es una aplicación WEB?

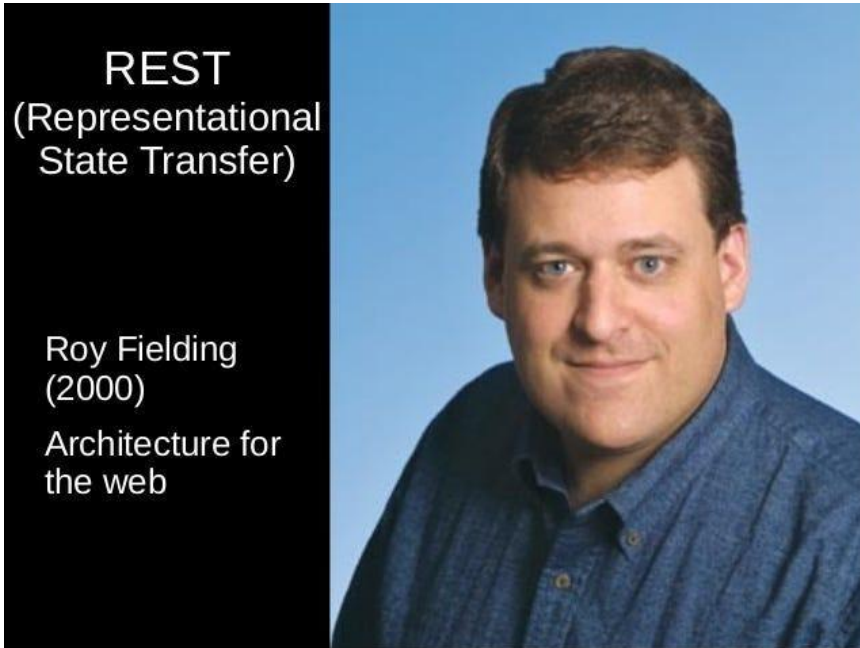


Una aplicación Web es un conjunto de páginas que funcionan en internet, estas páginas son las que el usuario ve a través de un navegador.

Las aplicaciones Web se almacenan en un servidor, el cual es una computadora que se encarga de que éstas sean accesibles a través de internet.



Una aplicación web funciona con una computadora-servidor y una o varias computadoras-cliente conectadas a través de internet



La arquitectura REST (Representational State Transfer) es un estilo arquitectónico utilizado en el diseño de aplicaciones web que se basa en principios y restricciones específicas para facilitar la comunicación entre sistemas distribuidos.

Arquitectura REST | RECURSOS



En la arquitectura REST, todo es considerado como un recurso. Cada recurso tiene una identificación única llamada **URI** (Uniform Resource Identifier), que se utiliza para acceder y manipular el recurso.

- Las URI recibirán nombres que no deben implicar una acción, es decir, se debe evitar colocar verbos en ellas.
- Deben ser únicas, no debemos tener más de una URI para identificar un mismo recurso.
- Deben ser independiente de formato, es decir, no debe representar ninguna extensión:
- Deben mantener una jerarquía lógica. La jerarquía es el criterio por el que se ordena los elementos.



Arquitectura REST | Métodos HTTP



Los métodos HTTP (GET, POST, PUT, DELETE, etc.) se utilizan para definir las operaciones que se pueden realizar sobre los recursos.

El proceso de comunicación según HTTP



Arquitectura REST | Conceptos



Representaciones

Los recursos pueden tener diferentes representaciones, como JSON, XML, HTML, etc.

Estado de la Aplicación

La comunicación entre el cliente y el servidor en REST es stateless. El estado de la aplicación se mantiene en el cliente o en el servidor, pero no en el protocolo.



Arquitectura REST | Conceptos



Interfaz Uniforme

REST sigue una interfaz uniforme que simplifica el diseño y la interacción entre los componentes de la aplicación.

Sin Estado (Stateless)

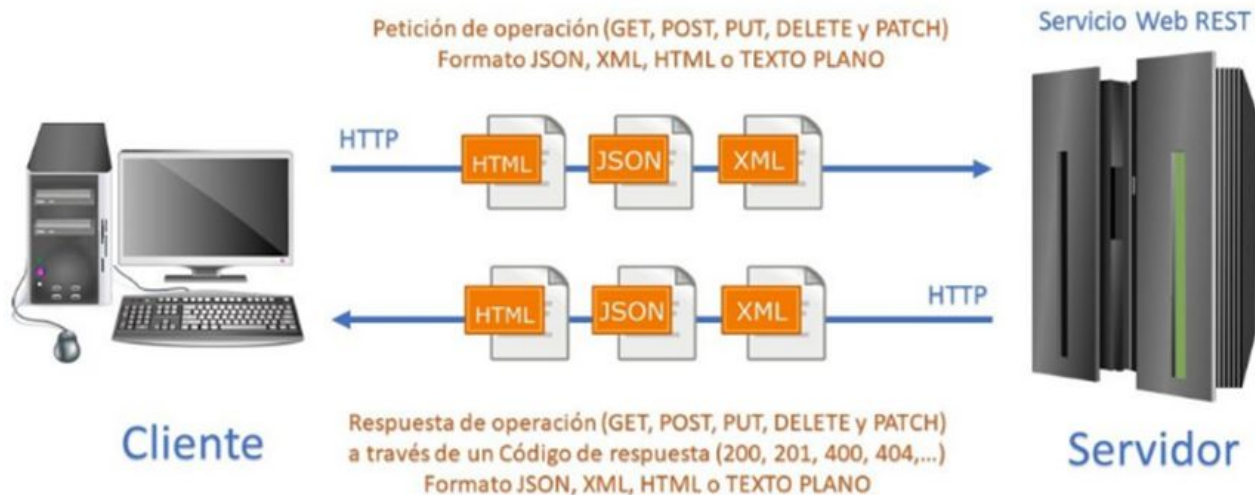
Las interacciones entre el cliente y el servidor en REST son independientes y no almacenan información de estado en el servidor entre solicitudes.



Arquitectura REST | Sistemas Distribuidos



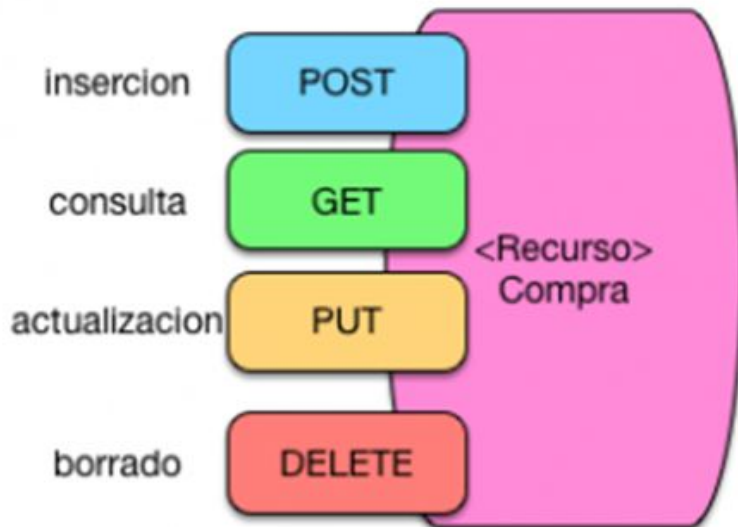
REST está diseñado para ser utilizado en sistemas distribuidos, lo que significa que puede escalarse fácilmente y puede ser utilizado en aplicaciones que se ejecutan en diferentes servidores y plataformas.



Métodos HTTP | GET POST



Para poder interactuar con los recursos de la arquitectura REST utilizaremos, los siguientes métodos HTML:



Métodos HTTP | GET



Realiza una petición a un recurso específico. No permite el envío de datos a excepción si dichos datos se envían como parámetro en la Url que realiza la petición.

```
<form action="www.misitioweb/index.html" method="get">  
    Nombre: <input type="text" name="nombre"><br>  
    Apellido: <input type="text" name="apellido"><br>  
    <input type="submit" value="Enviar">
```



Métodos HTTP | POST



Envia datos al servidor por medio del cuerpo (body) y nada por la Url como se emplea en el método GET. El tipo de cuerpo de solicitud se define en la cabecera Content-Type. Semánticamente se utiliza para registrar información, similar al INSERT de datos a nivel de base de datos.

```
<form action="www.misitioweb/index.html" method="post">
```

```
Nombre: <input type="text" name="nombre"><br>
```

```
Apellido: <input type="text" name="apellido"><br>
```

```
<input type="submit" value="Enviar">
```

```
</form>
```



Métodos HTTP | PUT



- Es similar al método de petición POST
- Puede ser ejecutado varias veces y tiene el mismo efecto
- Semánticamente es semejante a un UPDATE de datos a nivel de base de datos.



Métodos HTTP | DELETE



- Este método de petición permite eliminar un recurso específico.
- puede ser ejecutado varias veces
- tiene el mismo efecto similar al PUT y GET.
- Semánticamente se utiliza para eliminar información existente, es semejante a un DELETE de datos a nivel de base de datos.

