

SERVICIOS WEB

(extraído de <https://www.codigonaranja.com/2018/restful-web-service>)

¿Que es un web service?

Primero definamos que es un web service; un web service es un programa, diseñado para el intercambio de información máquina a máquina, sobre una red. Entonces esto hace que una computadora o programa pueda solicitar y recibir información de otra computadora o programa. A quien solicita la información se le llama cliente y a quien envía la información se le llama servidor.

¿Qué es API?

La palabra viene de Application Programming Interface, y no es más que un programa que permite que otros programas se comuniquen con un programa en específico, por ejemplo Facebook. A diferencia de los web services, las API no necesariamente deben comunicarse entre una red, pueden usarse entre dos aplicaciones en una misma computadora.

¿REST vs RESTful?

¿Qué es rest?, es una arquitectura para aplicaciones basadas en redes (como Internet), sus siglas significan REpresentational State Transfer y por otro lado RESTful web service o RESTful api, son programas basados en REST. Pero muchas veces se usan como sinónimos (REST y RESTful).

¿Que hace que un web service sea REST?

Usualmente los RESTful web service tienen estas características:

- Esta asociados a información
- Permiten listar, crear, leer, actualizar y borrar información
- Para las operaciones anteriores necesitan una URL y un método HTTP para accederlas
- Usualmente regresan la información en formato JSON.
- Retornan códigos de respuesta HTML, por ejemplo 200, 201, 404, etc

Los métodos HTTP que usan son los siguientes

Cuando solicitamos una página web, podemos hacer por diferentes métodos, el más común es el GET, es el que usamos cuando digitamos una dirección en nuestro navegador, en ocasiones utilizamos POST, cuando enviamos un formulario con datos, pero las aplicaciones pueden usar otros métodos como PATCH, PUT, etc.

- Listar y leer: Usan el método GET
- Crear: Usan el método POST
- Actualizar: Usan el método PATCH para actualizar y PUT para reemplazar.
- Borrar: Usan el método DELETE

Algunos códigos de estado HTML

Cuando se recibe una página HTML, también se recibe un código de estado HTML (solo uno), en los web service RESTful, estos se usan para saber el estado de la ejecución del servicio, y estos son algunos de los usados:

- 200 (Ok), cuando una operación fue exitosa.
- 201 (Created), cuando se creó un registro (recuerdan? con el método POST)
- 403 (Forbidden), cuando intentamos leer un registro para el que no tenemos acceso, por ejemplo los datos del perfil de otro usuario.
- 404 (Not found), cuando intentamos leer un registro que no existe.

EJEMPLO DE SERVICIO RESTful

(extraído de <https://www.codigonaranja.com/2018/crear-restful-web-service-php>)

Para este ejemplo vamos a asumir que tenemos un blog y necesitamos un servicio RESTful para acceder a los artículos y editarlos, entonces nuestro servicio va a realizar estas operaciones:

- Listar todos los posts (artículos de un blog)
- Mostrar un post
- Agregar un post
- Eliminar un post
- Actualizar uno o más campos de un post

Tanto la base de datos como los ficheros que incluyen el modelo están en el documento comprimido que acompaña esta explicación. Aquí se analiza solo el controlador principal

Código de PHP para el web service RESTful

Ahora vamos a crear 3 archivos de PHP

- config.php: En este archivo se encuentra la configuración para la base de datos
- utils.php: Funciones que hacen más fácil interactuar con la base de datos
- post.php: El código propio del web service RESTful

Este es el archivo de configuración, debes asegurarte de que tiene todos los parámetros correctos para que pueda conectarse a tu base de datos

```
<?php
include "modelo.php";
$dbConn = new bd();
/*
    listar todos los posts o solo uno
*/
if ($_SERVER['REQUEST_METHOD'] == 'GET')
{
    if (isset($_GET['id']))
    {
        //Mostrar un post
        $post= new Post ($_GET['id'],'',' ',' ',' ');
        header("HTTP/1.1 200 OK");
        echo json_encode($post->buscar($dbConn->link));
        exit();
    }
    else {
        //Mostrar lista de post
        header("HTTP/1.1 200 OK");
        echo json_encode( Post::getall($dbConn->link) );
        exit();
    }
}

// Crear un nuevo post
if ($_SERVER['REQUEST_METHOD'] == 'POST')
```

```

{
    $post= new Post ('',$_POST['title'],$_POST['status'],
$_POST['content'],$_POST['user_id']);
    if($postId=$post->insertar($dbConn->link))
    {
        header("HTTP/1.1 200 OK");
        echo json_encode("Insertado el post numero $postId");
        exit();
    }
}

//Borrar
if ($_SERVER['REQUEST_METHOD'] == 'DELETE')
{
    $post= new Post ($_GET['id'],'','','','');
    $post->borrar($dbConn->link);
    header("HTTP/1.1 200 OK");
    echo json_encode("Borrado el post numero ".$_GET['id']);
    exit();
}

//Actualizar
if ($_SERVER['REQUEST_METHOD'] == 'PUT')
{
    $post= new Post ($_GET['id'],'','','','');
    $post->modificar($dbConn->link,$_GET);
    echo json_encode($post->buscar($dbConn->link));
    header("HTTP/1.1 200 OK");
    exit();
}

//En caso de que ninguna de las opciones anteriores se haya  
ejecutado
header("HTTP/1.1 400 Bad Request");

?>

```

Ahora voy a explicar un poco el último archivo, como ya sabes en RESTful la acción a ejecutar depende mucho del método HTTP que utilizas, entonces utilizo la variable `$_SERVER['REQUEST_METHOD']`, para determinar si se está haciendo una petición GET, POST, DELETE o PUT.

Con la sentencia `header("HTTP/1.1 200 OK")`, puedo enviar códigos de respuesta, para informar si el servicio se ejecutó con éxito o con algún error.

Con la sentencia `json_encode` de php, se transforma la salida de `echo` en un formato JSON, lo cual es necesario en los servicios RESTful.