

[Inicio](#) [Desarrollo web »](#) [Sistemas »](#) [Cursos »](#) [Videos](#) [Sobre mi](#) [Contacto](#) [Newsletter](#)

API Rest con Slim

[Inicio](#)[Desarrollo web](#)[API Rest con Slim](#) Like 0

Vamos a ver como crear un API Rest con Slim. REST, REpresentational State Transfer, es una arquitectura de desarrollo web basada en el estándar HTTP.

REST nos permite crear servicios y aplicaciones que pueden ser usadas por cualquier dispositivo o cliente que entienda HTTP, por lo que es muy simple y flexible de usar.

REST cuenta con muchos métodos para utilizar, en concreto los métodos disponibles en el protocolo http. Los mas utilizados son:

- GET: Para consultar y conseguir registros
- POST: Para crear registros
- PUT: Para editar registros
- DELETE: Para eliminar registros

Veamos un ejemplo de un API REST con Slim Framework. Para ir comprobando el funcionamiento utilizaremos Postman que podemos instalar en nuestro navegador Chrome.

```
1 <?php
2
3 //Cargamos el framework
4 require_once 'vendor/autoload.php';
5
6 $app = new \Slim\Slim();
7
8 //Creamos la conexión a la base de datos con MySQLi
9 $db = new mysqli('localhost', 'root', '', 'pruebas');
10
11 /*
12 * Ruta/Controlador Get, le decimos que use las variables $db, $app
13 * GET para CONSEGUIR
14 */
15 $app->get('/users', function () use($db, $app) {
16
17     $select = $db->query("select * from users;");
18     $users=array();
19     while($fila=$select->fetch_assoc()){
20         $users[]=$fila;
21     }
22
23     echo json_encode($select->fetch_assoc());
24 });
25
26 //POST para INSERTAR
27 $app->post('/users', function () use($db, $app) {
28     //Request recoge variables de las peticiones http
29     $request = $app->request;
30
31     $insert = $db->query("INSERT INTO users VALUES(NULL,
32                                     '{$request->post("email")}',
33                                     '{$request->post("password")}',
34                                     '{$request->post("nombre")}',
35                                     '{$request->post("apellido")}'
36                                     );");
37
38     if ($insert) {
39         $result = array("status" => "true", "message" => "Usuario creado
40     } else {
41         $result = array("status" => "false", "message" => "Usuario NO cre
42     }
43     echo json_encode($result);
44 });
45
46 //PUT para ACTUALIZAR
47 $app->put('/users/:id', function ($id) use($db, $app) {
48
49     $request = $app->request;
50
51     $sql = "UPDATE users SET
52                                     email = '{$request->params("email")}',
53                                     password = '{$request->params("password")}'";
```

```

53         nombre = '{$request->params("nombre")}',
54         apellido = '{$request->params("apellido")}'
55         WHERE id=$id";
56
57     $update = $db->query($sql);
58
59     if ($update) {
60         $result = array("status" => "true", "message" => "Usuario modificado");
61     } else {
62         $result = array("status" => "false", "message" => "Usuario NO modificado");
63     }
64     echo json_encode($result);
65 });
66
67 //DELETE para BORRAR
68 $app->delete('/users/:id', function ($id) use($db, $app) {
69
70     $request = $app->request;
71
72     $sql = "DELETE FROM users WHERE id=$id";
73
74     $delete = $db->query($sql);
75
76     if ($delete) {
77         $result = array("status" => "true", "message" => "Usuario eliminado");
78     } else {
79         $result = array("status" => "false", "message" => "Usuario NO eliminado");
80     }
81     echo json_encode($result);
82 });
83
84 $app->run();
85 ?>

```

Si hacemos una petición GET nos devolverá un objeto json con el resultset.

Por ejemplo si en postman elegimos POST, rellenamos el “formulario” y enviamos, nos devuelve que ha creado el usuario.

Normal Basic Auth Digest Auth OAuth 1.0 No environment

http://localhost/slim/index.php/users

form-data x-www-form-urlencoded raw

| | | | |
|----------|-------------------------|------|---|
| email | info@victorroblesweb.es | Text | ✕ |
| password | secret | Text | ✕ |
| nombre | victor | Text | ✕ |
| apellido | robles | Text | ✕ |
| Key | Value | Text | ✕ |

Send Preview Add to collection

Body Headers (7) STATUS 200 OK TIME 41 ms

Pretty Raw Preview

```
{
  "status": "true",
  "message": "Usuario creado correctamente"
}
```

Si hacemos una petición PUT, tendremos que elegir x-www-form-urlencoded para enviar los datos, nos modificará el usuario correctamente.

Normal Basic Auth Digest Auth OAuth 1.0 No environment

http://localhost/slim/index.php/users/20

form-data x-www-form-urlencoded raw

| | | |
|----------|-------------------------|---|
| email | info@victorroblesweb.es | ✕ |
| password | secret99 | ✕ |
| nombre | Victor | ✕ |
| apellido | robles | ✕ |
| Key | Value | |

Send Preview Add to collection

Body Headers (7) STATUS 200 OK TIME 37 ms

Pretty Raw Preview

```
{ "status": "true", "message": "Usuario modificado correctamente" }
```

Y lo mismo con DELETE.

Ya tendremos el API REST disponible para usar desde cualquier aplicación externa. Esto habría que darle seguridad de alguna forma, pero como ejemplo de momento nos vale.

Slim es muy similar a Silex, otro microframework Restful para PHP basado en Symfony. Si queremos construir APIs REST más complejas podemos utilizar frameworks como Laravel o Symfony2.

☐ ¿Quieres más? **Apúntate gratis al curso Crear un API RESTful con Slim PHP y usarla con AJAX**

Más información:

[Conceptos sobre APIs REST](#)

<http://www.slimframework.com/>

[Introducción a API REST y Framework Slim](#)

También te gustará:



Instalar Slim Framework



Crear un API RESTful con Slim PHP y usarla con AJAX



Crear servicios en Symfony3



Hacer consultas y conseguir el usuario dentro de un formulario de Symfony3



Crear un módulo básico en Drupal 8

Like 0



Autor: Victor

Desarrollador web - Formador online - Blogger

Compartir este post

Poner un comentario

Tu dirección de correo electrónico no será publicada. Los campos obligatorios están marcados con *

Comentario

Nombre *

Correo electrónico *

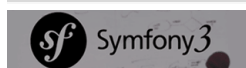
Web

Enviar comentario

Buscador

Buscar

Entra a mis cursos online



Recibe contenido exclusivo ↓

Pon tu email y tendrás

¡Apuntarme!

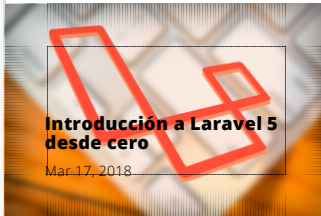
Sígueme y ve las novedades



Seguir a @victorobs

Me gusta 1465

Novedades en Desarrollo web



Curso gratuito de HTML: Crear mi primera página web

Mar 10, 2018



Curso de Laravel 5 desde cero + APIs RESTful y webapps en Angular

Feb 12, 2018



Configurar el acceso CORS en NodeJS

Ene 31, 2018

Categorías

- ▶ Cursos (35)
- ▶ Desarrollo web (305)
 - ▶ Angular (66)
 - ▶ Codeigniter (4)
 - ▶ Drupal 8 (5)
 - ▶ JavaScript (108)
 - ▶ Laravel 5 (20)
 - ▶ NodeJS (17)
 - ▶ PHP (175)
 - ▶ PHP POO (12)
 - ▶ Symfony2 (55)
 - ▶ Symfony3 (35)
 - ▶ VueJS 2 (5)
 - ▶ Yii Framework (20)
 - ▶ Zend Framework 2 (36)
- ▶ Promociones (6)
- ▶ Sin categoría (2)
- ▶ Sistemas (66)

- Hardware (10)
- Linux (44)
 - Shell Script (5)
- Windows (25)
- Videos (12)

Desarrollado por **Victor Robles** © 2018

Victor Robles - Política de cookies

Este sitio utiliza cookies para mejorar la experiencia de usuario. Si continuas navegando considero que aceptas el uso de cookies.

[Close](#)