



Academind

Acad

FULL STACK DEVELOPER BOOTCAMP

Desarrollamos
{ talento }



API REST

API REST

API es la abreviatura de “Interfaz de Programación de Aplicaciones” (Application Programming Interface en inglés). Es una “llave de acceso” a funciones que podemos utilizar de un servicio web provisto por un tercero, dentro de nuestra propia aplicación web, de manera segura y confiable

API REST

La Transferencia de Estado Representacional (Representational State Transfer) o REST es un estilo de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web. El término se originó en el año 2000, en una tesis doctoral sobre la web escrita por Roy Fielding, uno de los principales autores de la especificación del protocolo HTTP y ha pasado a ser ampliamente utilizado por la comunidad de desarrollo.

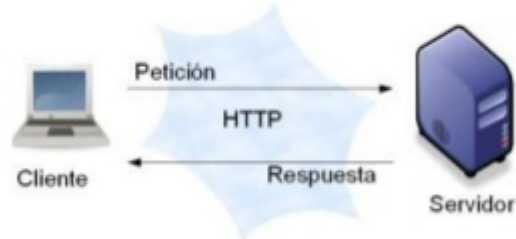
API REST

Digamos que es cualquier interfaz entre sistemas que use HTTP para obtener datos o generar operaciones sobre esos datos en todos los formatos posibles como Xml o JSON.

¿Qué es una API REST?



Arquitectura Cliente - Servidor



El cliente hace peticiones al servidor usando el protocolo HTTP. El servidor le responde a través del mismo protocolo.

API REST

¿Pero sólo se pueden crear api Rest?

Por desgracia no y digo por desgracia ya que existen otros estandares pero cual más complejo por ejemplo SOAP(Simple Object Access Protocol)

API REST



Ventajas de una API REST respecto a otras arquitecturas.

API REST

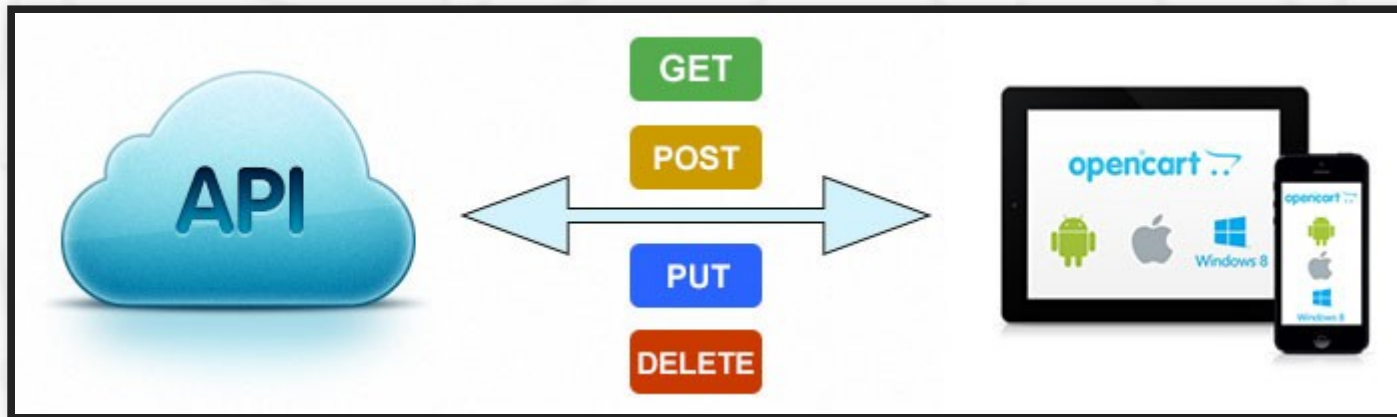
¿Como funciona?

Utilizamos el estandar HTTP

- Post: Para crear un recurso
- Get: Para obtener recursos
- Put: Para modificar un recurso.
- Delete: Para eliminar un recurso.

En definitiva podemos realizar un **C.R.U.D (Create, Read, Update, Delete)**

API REST



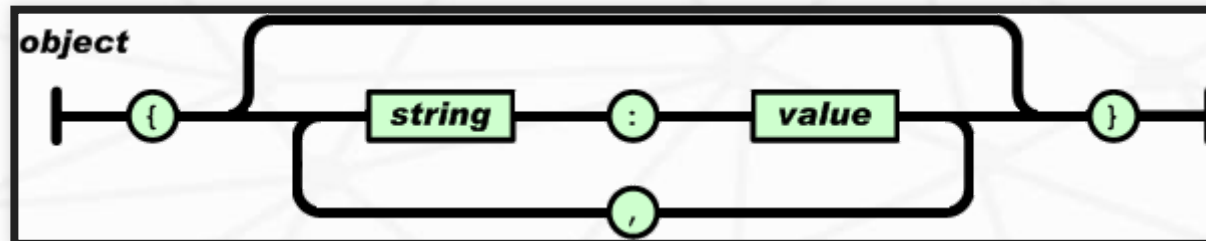
API REST

Los objetos en REST siempre se manipulan a partir de la URI (EndPoints) Recuperar todos los **cursos**. Recuperar un **curso particular**.

API REST

¿Como funciona?

Pues esto se lo debemos a Douglas Crockford, creador del estandar JSON. **JavaScript Object Notation**



API REST

¿Como se envia el JSON?

Mediante una petición POST a un endpoint determinado y en pasando en el body un json con la información que queremos registrar.

peticion

```
{  
  "provider":0,  
  "nickname": "xavi12",  
  "password": "jrodriguez"  
}
```

API REST

¿Como modificamos un elemento?

Mediante una petición PUT a un endpoint determinado y en pasando en el body un json con la información que queremos modificar.

Petición

```
{  
  "provider":0,  
  "nickname": "xavi12",  
  "password": "jrodriguez"  
}
```


API REST

¿Y cómo eliminamos?

Accediendo a una uri y utilizando DELETE para poder eliminar un elemento en particular podemos pasar por la uri un parámetro que puede ser un id de dicho elemento.

API REST

Dado que estamos sobre el protocolo http, también debemos devolver un código http.

- 2XX Peticiones correctas:
- 200 ok (Respuesta correcta)
- 204 No Content(No existe el contenido).
- 205 Reset Content(Para indicar que se ha eliminado).
- 4XX Peticiones incorrectas:
- 401 Unauthorized (No tienes acceso.)
- 404 Not Found (No localizado).

API REST

Seguridad

El primer problema es que debemos tener cifrada la conexión porque es abierta para todos los usuarios, para ello deberíamos montar nuestra conexión mediante SSL. Para que se encuentre cifrada.

HTTP vs HTTPS

User



Insecure connection

Normal HTTP (80)



User



Encrypted Connection

Secure HTTPS (443)



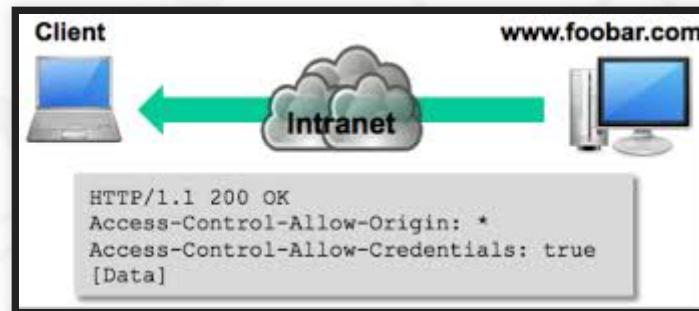
SSL Certificate



API REST

Seguridad

Otro punto que nos puede dar muchos dolores de cabeza es CORS Son una cabecera que permite el acceso a dicha página.



API REST

Seguridad

Como securizar algunas de las peticiones:

Mediante JWT → JSON web Tokens, que es otro estandar podemos encontrar más información en:

<https://jwt.io/>



Header

```
base64enc({  
  "alg": "HS256",  
  "typ": "JWT"  
})
```

Payload

```
base64enc({  
  "iss": "toptal.com",  
  "exp": 1426420800,  
  "company": "Toptal",  
  "awesome": true  
})
```

Signature

```
HMACSHA256(  
  base64enc(header)  
  + '.' +  
  base64enc(payload)  
  , secretKey)
```

