



**REST API**

# ¿Qué es una API REST?

Una **API REST (Application Programming Interface - Representational State Transfer)** o “Interfaz de programación de aplicaciones - Transferencia de Estado Representacional” es una forma de permitir que diferentes sistemas o aplicaciones se comuniquen entre sí a través de la web. REST es un estilo arquitectónico para diseñar servicios web que se centran en recursos y utiliza HTTP como protocolo de comunicación.

## Recursos

**En REST, todo se trata como un recurso.**

Un recurso puede ser cualquier cosa que se pueda nombrar y manejar en la web, como usuarios, productos, pedidos, etc. Cada recurso es identificado por una URL única.

# Verbos HTTP

REST utiliza los verbos HTTP para operar con los recursos.  
Los verbos más comunes son:

- **GET:** Recuperar información sobre un recurso.
- **POST:** Crear un nuevo recurso.
- **PUT:** Actualizar un recurso existente.
- **PATCH:** Actualizar solo los campos seleccionados de un recurso existente.
- **DELETE:** Eliminar un recurso.

# Representaciones

Cuando accedes o modificas un recurso, lo haces mediante una representación del mismo, que generalmente está en formato **JSON** o **XML**.

La representación es simplemente una forma de describir el recurso en un formato que tanto el cliente como el servidor puedan entender.

## Estateless (sin estado)

Cada solicitud de la **API REST** debe contener toda la información necesaria para que el servidor pueda entenderla y procesarla. El servidor no debe almacenar ninguna información sobre el estado del cliente entre solicitudes. Esto hace que las interacciones sean más simples y escalables.

## Código de estado HTTP:

Las API REST utilizan códigos de estado HTTP para indicar el resultado de la solicitud:

200	OK	La solicitud fue exitosa.
201	Created	Un nuevo recurso fue creado exitosamente.
400	Bad Request	La solicitud tiene un error y no puede ser procesada.
401	Unauthorized	La autenticación es requerida y ha
404	Not Found	El recurso solicitado no fue encontrado. fallado o no ha sido proporcionada.
500	Internal Server Error	Error en el servidor

## **Uniform Interface (Interfaz Uniforme):**

REST tiene un conjunto uniforme de reglas y convenciones para interactuar con los recursos, lo que hace que sea predecible y más fácil de entender.



## Ejemplo de una API REST

Imagina que tienes una API REST para gestionar usuarios:

- **GET /usuarios** Recupera una lista de todos los usuarios.
- **GET /usuarios/1** Recupera los detalles del usuarios con ID 1.
- **POST /usuarios** Crea un nuevo usuarios.
- **PUT /usuarios/1** Actualiza la información del usuario con ID 1.
- **PATH /usuarios/1** Actualiza campos del usuario con ID 1
- **DELETE /usuarios/1** Elimina el usuarios con ID 1.

Con esta estructura, cada recurso (en este caso, los libros) tiene una **URL única** y se puede interactuar con ellos usando los verbos HTTP adecuados.