

HTTP Y APIs



¿Qué es una API?

Una API (Application Programming Interface) permite la interacción entre aplicaciones, facilitando la comunicación con un servidor, estas suelen usar métodos HTTP para la interacción.

Características

Interoperabilidad : Permite que aplicaciones diferentes se comuniquen entre sí.

Modularidad: Ayudan a decidir funcionalidades complejas en módulos más pequeños que puedan ser reutilizados.

Acceso controlado : Que las APIs permitan que los usuarios solo puedan acceder a algunos datos.

Facilita el desarrollo : Usar APIs permiten a los desarrolladores concentrarse en su lógica de negocio y reutilizar funcionalidades ya creadas por otros.

TIPOS DE APIS

APIs de Web: Usada para comunicación entre aplicaciones usando protocolos como HTTP

APIS de Sistema: Usadas por SO para interactuar con el hardware.

APIs de Librerías: Facilitan el uso de funcionalidades específicas de librerías o frameworks de programación.

¿HTTP?

(HyperText Transfer Protocol), protocolo que define cómo los mensajes son transmitidos a través de una página web. Principal protocolo para la transferencia de datos entre un cliente y un servidor en la web.

MÉTODOS

HTTP tiene varios métodos que determinan la acción que se va a realizar, como :

- **GET** : Solicita datos del servidor.
- **POST** : Enviar Datos al servidor.
- **PUT** : Actualizar un recurso existente en el servidor.
- **DELETE**: Elimina un recurso existente en el servidor.
- **PATCH**: Realiza actualizaciones parciales en un recurso.

URL

Dirección única que identifica un recurso en la web.

CÓDIGOS DE ESTADO HTTP

Los servidores devuelven códigos de estado HTTP para indicar el resultado de una solicitud:

2xx: Indica éxito. Ejemplos: **200 OK**, **201 Created**.

3xx: Redirecciones. Ejemplo: **301 Moved Permanently**.

4xx: Errores del cliente. Ejemplos: **404 Not Found**, **401 Unauthorized**.

5xx: Errores del servidor. Ejemplo: **500 Internal Server Error**.

HEADERS HTTP

Contienen información adicional sobre la solicitud de respuesta, algunos son:

Content-Type: Indica el tipo de contenido que se está enviando o recibiendo (por ejemplo, **application/json**).

Authorization: Se usa para enviar credenciales, como tokens de autenticación.

User-Agent: Proporciona información sobre el navegador o cliente que realiza la solicitud.

REQUEST Y RESPONSE

Request: Solicitud que el cliente envía al servidor, que incluye el método HTTP, la URL, y si quiere, un body de datos.

Response: Respuesta de servidor incluyendo un código de estado, headers, y cuerpo de la respuesta

FLUJO BÁSICO DE UNA SOLICITUD

El Cliente (Navegador o Aplicación) envía una solicitud HTTP a un servidor mediante una URL específica.

El Servidor procesa la solicitud, busca el recurso y genera una respuesta.

El Servidor devuelve una respuesta HTTP al cliente con un código de estado, headers, y posiblemente datos (HTML, JSON, XML, etc.).

El Cliente muestra la respuesta o la utiliza según sea necesario (mostrar una página web, procesar datos de una API, etc.).

APIS RESTFUL Y HTTP

Sitio arquitectónico para diseñar APIS, basado en HTTP:

- Usar HTTP y sus métodos.
- Usar URL para acceder a recursos.
- Enviar datos en forma de JSON O XML.
- Ser stateless, lo que significa que el servidor no guarda estado entre las solicitudes.

Un ejemplo de una API rest sería:

GET: <https://api.ejemplo.com/users>: Obtiene la lista de usuarios.

POST: <https://api.ejemplo.com/users>: Crea un nuevo usuario.

PUT: <https://api.ejemplo.com/users/123>: Actualiza el usuario con ID 123.

DELETE: <https://api.ejemplo.com/users/123>: Elimina el usuario con ID 123.