

"Elementos de Solicitud HTTP"

Integrantes:

- "Díaz Rossini Juan José",
- "Gallo Genaro"
- "Navarro Victor Leandro"

-
- Métodos HTTP
 - Camino (Path)
 - Versión de HTTP
 - Cabeceras (headers)
 - Cuerpo (body)
-

Elementos de una solicitud HTTP

1. Métodos HTTP

Los métodos HTTP indican la acción que el cliente quiere realizar sobre un recurso en el servidor. Cada método tiene un propósito específico y define cómo se manipulan los datos.

Ejemplos comunes:

- **GET:** obtener información (por ejemplo, consultar una lista de productos).
- **POST:** enviar datos al servidor para crear un recurso nuevo (registrar un usuario).
- **PUT:** actualizar un recurso existente (editar datos de perfil).
- **DELETE:** eliminar un recurso (borrar un producto de un catálogo).

En JavaScript, librerías como Axios o la API nativa `fetch()` permiten enviar solicitudes usando estos métodos.

2. Camino (Path)

El path es la ruta específica de un recurso dentro del servidor y se coloca en los endpoints de la petición. Permite acceder a información puntual.

Ejemplo: En la URL `https://ejemplo.com/productos/10`, el path es `/productos/10`, lo que generalmente apunta al producto con ID 10.

En las APIs modernas, los paths se diseñan de forma clara y semántica para facilitar su uso por los desarrolladores.

3. Versión de HTTP

La versión de HTTP indica qué estándar del protocolo se está utilizando en la comunicación. Define la manera en que se transmiten los datos entre cliente y servidor.

Ejemplos:

- **HTTP/1.1**: ampliamente usado, soporta múltiples cabeceras y conexiones persistentes.
- **HTTP/2**: más eficiente, permite multiplexar varias solicitudes en una sola conexión.
- **HTTP/3**: basado en QUIC, mejora la seguridad y la velocidad de respuesta.

4. Cabeceras (Headers)

Las cabeceras son pares clave: valor que aportan información adicional sobre la solicitud. Pueden incluir datos de autenticación, tipo de contenido o configuración de idioma.

Ejemplos:

- **Content-Type**: application/json → indica que el cuerpo de la solicitud está en formato JSON.
- **Authorization**: Bearer <token> → se utiliza para autenticar al usuario.
- **Accept-Language**: es-AR → especifica el idioma preferido de la respuesta.

Gracias a las cabeceras, el servidor entiende mejor el contexto de la petición.

5. Cuerpo (Body)

El cuerpo de la solicitud contiene los datos que el cliente envía al servidor. No siempre está presente (por ejemplo, en GET suele estar vacío), pero es esencial en métodos como POST o PUT.

Ejemplo de cuerpo en formato JSON para registrar un usuario:

```
{
  "nombre": "fulano",
  "edad": 20,
  "email": "fulanito@mail.com"
}
```

En JavaScript, se envía un body con fetch() o Axios incluyendo un objeto JSON en la configuración de la petición.