

Tarea 8 - Programación de una App para una empresa

Samuel Riveira Escudero

Resolver las siguientes cuestiones **sin ChatGPT**, intentando teorizar con lo visto en clase, se pueden resolver en grupo:

1. ¿Qué es una petición HTTP y qué métodos de peticiones existen?

GET: Solicita al servidor una información o recurso concreto.

HEAD: Sólo recibes la información de la cabecera de la página que quieres cargar

POST: Cuando necesita enviar datos al servidor

PUT: Para actualizar información que ya existe en el servidor final

2. Nosotros nos vamos a centrar en GET y POST, por lo tanto, ¿qué partes componen dichas peticiones HTTP?

- Línea de solicitud

Es la primera línea de la petición HTTP y contiene tres elementos principales:

- **Método HTTP:** Indica la acción a realizar, en este caso GET o POST.
- **URI (Uniform Resource Identifier):** Es el recurso al que se desea acceder.
- **Versión del protocolo:** Especifica la versión de HTTP.

- Encabezados HTTP (Headers)

- **Host:** Indica el servidor al que se envía la petición.
- **Content-Type:** Define el tipo de contenido del cuerpo de la petición (solo para POST o métodos con cuerpo).
- **Authorization:** Incluye tokens o credenciales para autenticación.
- **User-Agent:** Informa sobre el cliente que realiza la petición.

- Cuerpo de la petición (Body)

Solo está presente en métodos que permiten enviar datos (como POST).
Contiene los datos que se envían al servidor, típicamente en formatos como JSON o XML.

Tarea 8 - Programación de una App para una empresa

Samuel Riveira Escudero

- Parámetros de consulta (Query Parameters)

Se utilizan en métodos como GET para enviar datos mediante la URL. Los parámetros se añaden al URI después de un ? y están separados por &.

3. ¿Una petición GET tiene body?

Generalmente, **una petición GET no tiene body**. El propósito de una solicitud GET es solicitar recursos del servidor y enviar parámetros en la **URL** mediante una cadena de consulta. Aunque técnicamente el estándar HTTP permite incluir un body en una solicitud GET, no es común y muchos servidores lo ignoran o lo consideran una mala práctica. Según la especificación RFC 7231 de HTTP/1.1

4. ¿Una petición POST tiene body?

Sí, **una petición POST siempre tiene body**. Esto se debe a que su propósito principal es enviar datos al servidor, como archivos, o información

4.1 ¿Qué formato tiene el body de una petición POST?

El formato del **body** de una petición POST depende del encabezado Content-Type. Algunos formatos comunes son:

- **application/x-www-form-urlencoded:**
 - Formato utilizado por formularios HTML tradicionales.
 - Los datos se envían como pares clave-valor codificados.

Ejemplo:

name=Samu&age=20

- **multipart/form-data:**
 - Usado para enviar archivos o datos binarios junto con otros campos.
 - Divide los datos en partes separadas por delimitadores.
 - Ejemplo típico en formularios que envían imágenes o documentos.
- **application/json:**
 - Formato común en aplicaciones modernas, especialmente APIs REST.
 - Los datos se envían como un objeto JSON.

Tarea 8 - Programación de una App para una empresa

Samuel Riveira Escudero

Ejemplo:

```
json
{
  "name": "John",
  "age": 30
}
```

-
- **Texto plano (text/plain):**
 - El contenido se envía como texto sin formato.
- **Otros formatos personalizados:**
 - Por ejemplo, XML (`application/xml`) u otros tipos específicos según la necesidad del servidor.

5. Ahora que ya saben un poco más de HTTP, es el protocolo que vamos a utilizar para conectar la aplicación de Android/KMP/Web con nuestra base de datos. Pero para ello debemos tener una aplicación de tipo API Rest, ¿qué es una API REST?

Es una interfaz que permite la comunicación entre sistemas mediante **HTTP**. Utiliza métodos como **GET**, **POST**, **PUT**, y **DELETE** para interactuar con recursos identificados por **URLs**, generalmente intercambiando datos en formato **JSON**.

6. ¿Qué es un endpoint de una API REST?

Es una **URL** específica dentro de una **API REST** que permite acceder o manipular un recurso, como `/productos` para gestionar productos.

7. ¿Cómo puedo llamar a una petición de una API REST si sabemos que es de tipo HTTP GET? Pon un ejemplo con palabras. Pueden tomar como partida lo que ya hemos hecho en Tkinter con los productos.

Para obtener una lista de productos:

Tarea 8 - Programación de una App para una empresa

Samuel Riveira Escudero

- **Endpoint:** /productos.
- **Acción:** Enviar una solicitud GET.
- **Respuesta:** Una lista de productos en formato JSON.

8. ¿Cómo puedo llamar a una petición de una API REST si es de tipo POST? Pogan un ejemplo inventado con palabras.

Para crear un nuevo producto:

- **Endpoint:** /productos.
- **Acción:** Enviar una solicitud POST con un cuerpo JSON, como:
 { "nombre": "Zapatos", "precio": 49.99 }
- **Respuesta:** Confirmación y datos del nuevo producto creado.