



# Curso Flask

Introducción  
Protocolo HTTP

# Presentación

José Domingo Muñoz Rodríguez

[www.josedomingo.org](http://www.josedomingo.org)

@Pledin\_JD

## Descripción general

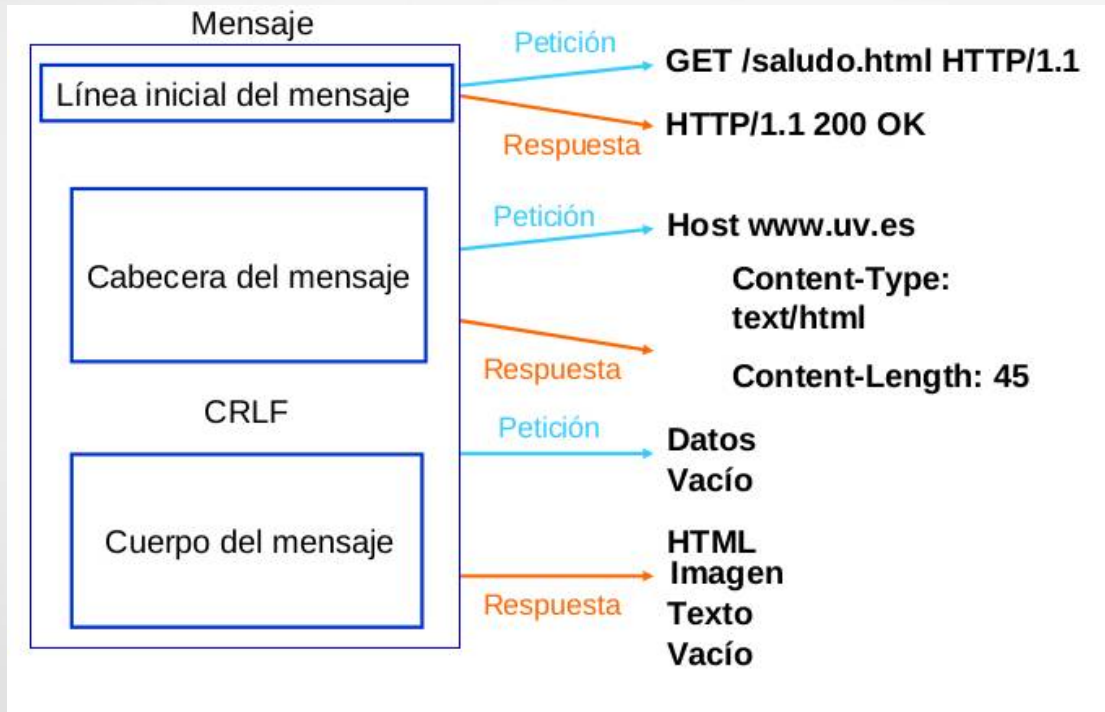
Protocolo de comunicaciones estándar que comunica servidores, proxies y clientes. Permite la transferencia de documentos web, sin importar cual es el cliente o cual es el servidor.

- ▶ Es un protocolo basado en el esquema petición/respuesta.

## Funcionamiento del protocolo

El usuario escribe una URL, indicando el protocolo, servidor y recurso que quiere obtener, el servidor procesa dicha información y devuelve un mensaje de respuesta, normalmente con el HTML de la página que se va a visualizar,...

# Funcionamiento del protocolo



## Métodos de envío de los datos

- ▶ GET: Solicita un documento al servidor.
- ▶ HEAD: Similar a GET, pero sólo pide las cabeceras HTTP.
- ▶ POST: Manda datos al servidor para su procesado.
- ▶ PUT: Almacena el documento enviado en el cuerpo del mensaje.
- ▶ DELETE: Elimina el documento referenciado en la URL.

# Código de estados

- **1xx:** Mensaje informativo.
- **2xx:** Exito
  - 200 OK
  - 201 Created
  - 202 Accepted
  - 204 No Content
- **3xx:** Redirección
  - 300 Multiple Choice
  - 301 Moved Permanently
  - 302 Found
  - 304 Not Modified
- **4xx:** Error del cliente
  - 400 Bad Request
  - 401 Unauthorized
  - 403 Forbidden
  - 404 Not Found
- **5xx:** Error del servidor
  - 500 Internal Server Error
  - 501 Not Implemented
  - 502 Bad Gateway
  - 503 Service Unavailable



## Cabeceras

- ▶ Server
- ▶ Age
- ▶ Cache-control
- ▶ Content-Encoding
- ▶ Expires
- ▶ Location
- ▶ Set-Cookie



# Cookies

Las cookies son información que el navegador guarda en memoria o en el disco duro dentro de ficheros de texto.

- Guarda información de la sesión.
- Comercio electrónico.
- Carrito de la compra.
- Personalización de páginas
- Idiomas
- Seguimiento de las visitas a un Web
- Carteles publicitarios
- Almacenamiento del login y password

# Sesiones

HTTP es un protocolo sin manejo de estados. Las sesiones son fundamentales en las aplicaciones Web. Permiten:

- Definir varios estados distintos en la aplicación.
- Colocar las solicitudes y respuestas dentro de un contexto más amplio.
- Los clientes y servidores intercambian información sobre el estado de la aplicación.