



## **WEB. Arquitectura, tecnologías y herramientas**



## **Arquitectura cliente-servidor**

- Cliente:
  - Consumidor de recursos (externos)
- Servidor:
  - Comparte recursos y brinda un servicio
- Mayoría de servicios de internet:
  - Email, Web, DNS, etc.
  - Son cliente-servidor

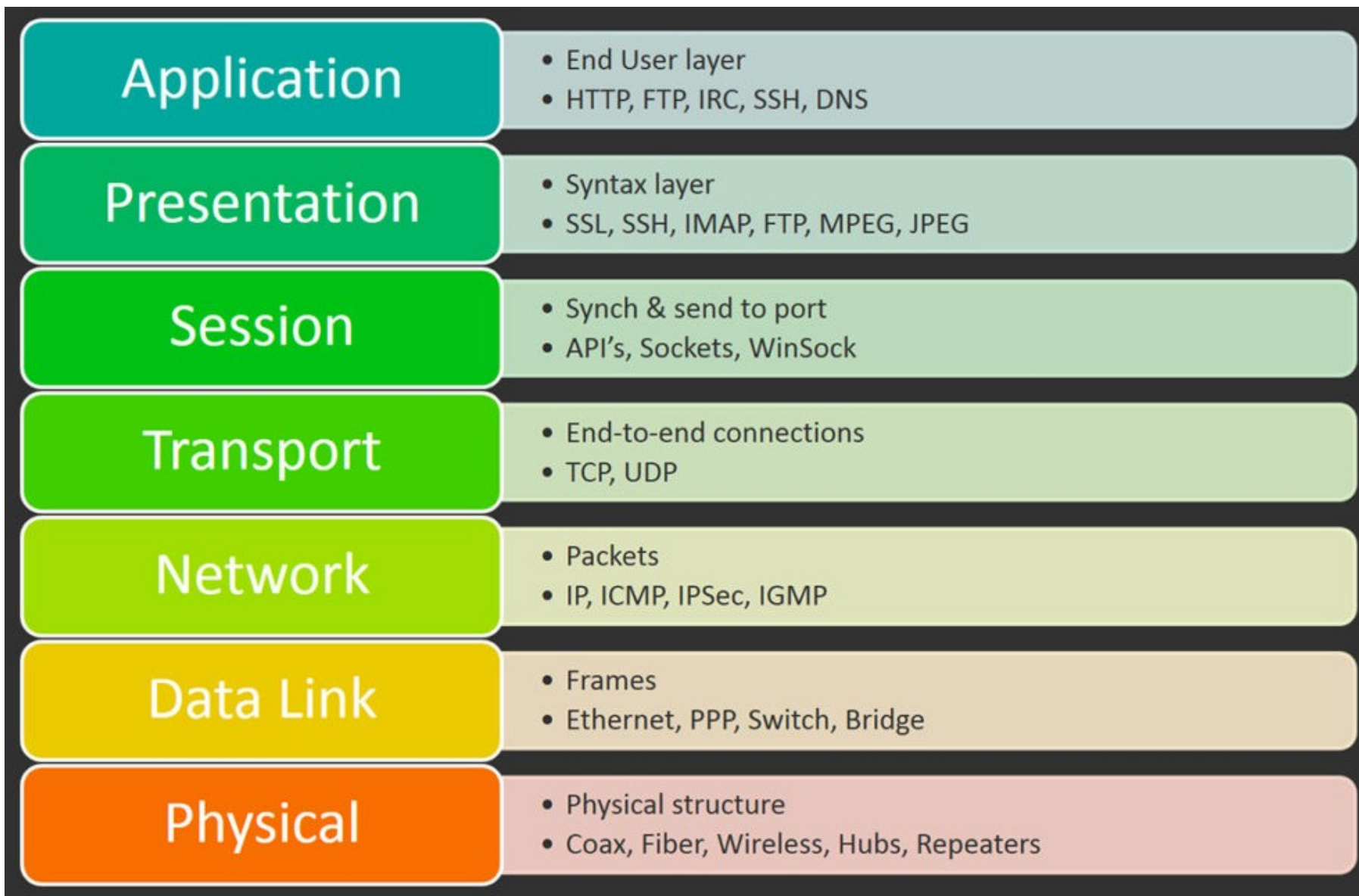


## **Arquitectura cliente-servidor**

- Protocolo TCP/IP
  - Cliente/Servidor
    - Alguien inicia una comunicación
    - Alguien espera una comunicación
- Protocolo HTTP
  - Sobre TCP/IP

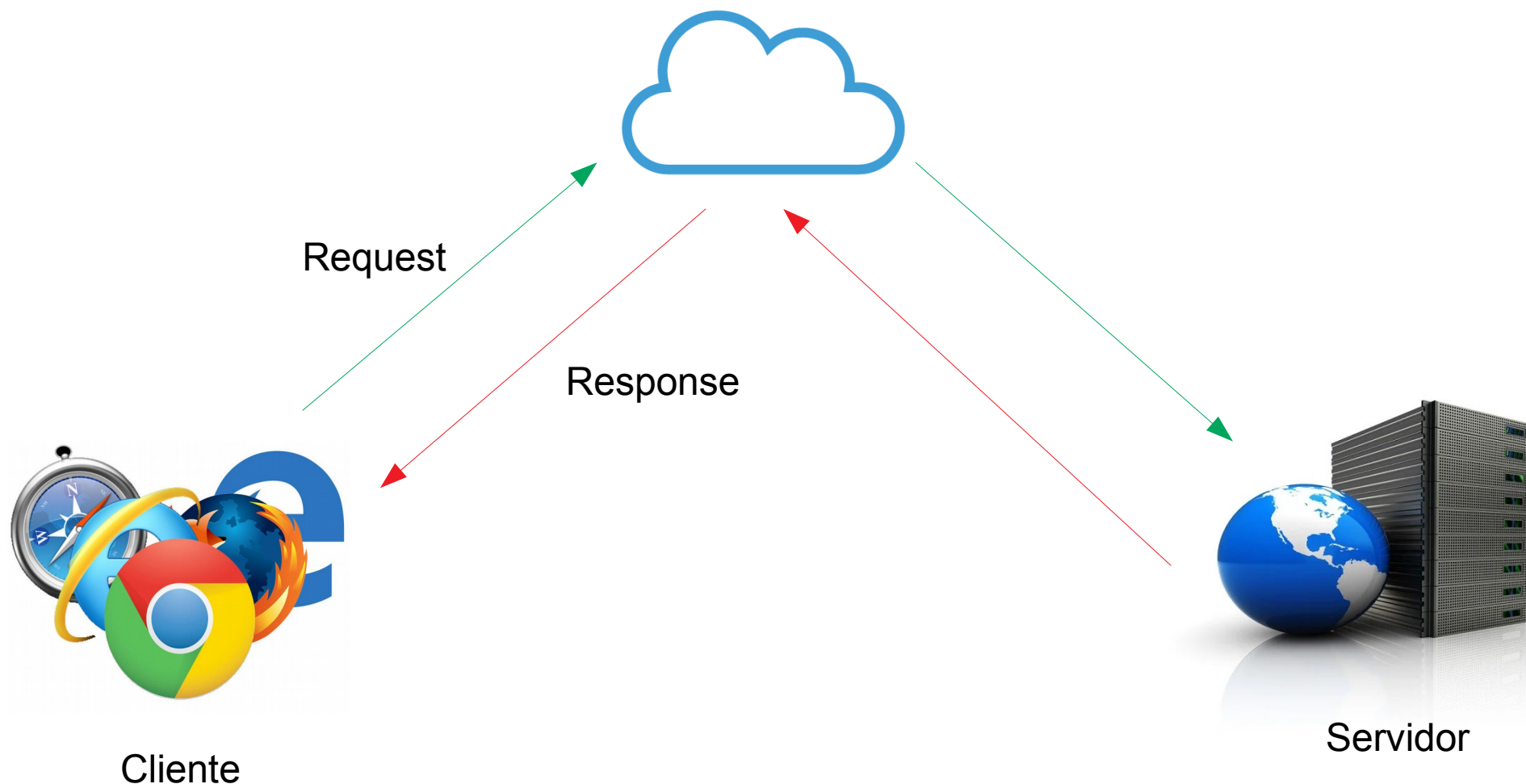


## Modelo de Capas (OSI)



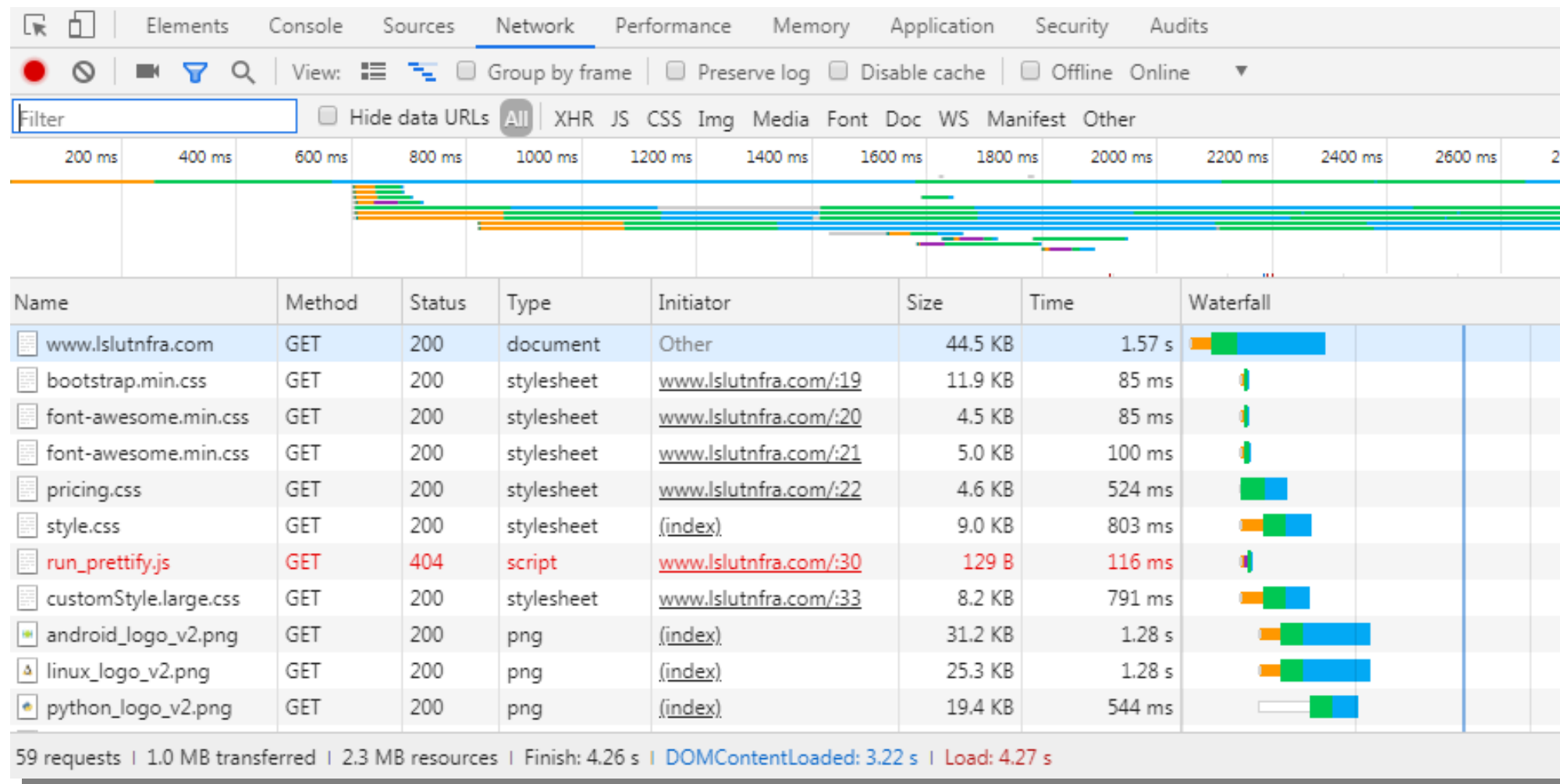


## Arquitectura cliente-servidor





## Arquitectura cliente-servidor







## Arquitectura cliente-servidor

- Protocolo HTTP: Hypertext Transfer Protocol
- Protocolo de request-response
- Sus recursos se identifican con URLs
- Posee un *Header*
- Se transmite texto plano
- Sin estado



## Arquitectura cliente-servidor

- **Request:**
  - Dirección (URL).
  - Versión del protocolo.
  - Campos (Por Ej. "Accept-Language: en").
  - Cuerpo de mensaje (opcional).
  - Método (GET, POST, DELETE, etc.)
    - Definen la acción a realizar
    - Las acciones se definen en el server.





## Arquitectura cliente-servidor

- **Response:**

- Dirección (URL).
- Versión del protocolo.
- Campo estado (200,404, etc.)
- Campos (por Ej. "Content-Type: text/html")
- Cuerpo de mensaje (opcional).



## Tipos de métodos

HTTP method ↕	RFC ↕	Request has Body ↕	Response has Body ↕
GET	<a href="#">RFC 7231</a>	Optional	Yes
HEAD	<a href="#">RFC 7231</a>	Optional	No
POST	<a href="#">RFC 7231</a>	Yes	Yes
PUT	<a href="#">RFC 7231</a>	Yes	Yes
DELETE	<a href="#">RFC 7231</a>	Optional	Yes
CONNECT	<a href="#">RFC 7231</a>	Optional	Yes
OPTIONS	<a href="#">RFC 7231</a>	Optional	Yes
TRACE	<a href="#">RFC 7231</a>	No	Yes
PATCH	<a href="#">RFC 5789</a>	Yes	Yes

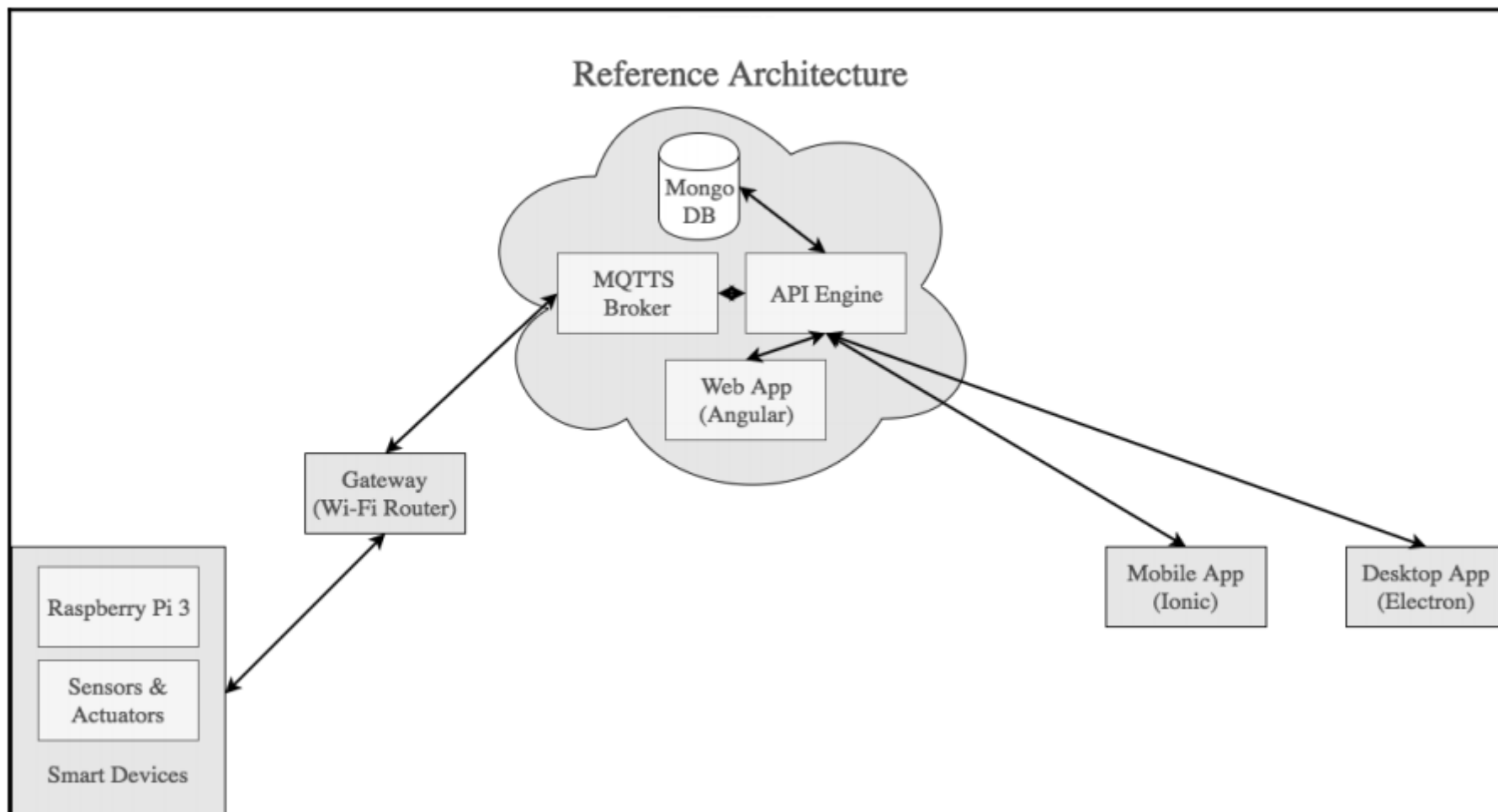


## Parámetros en la URL

- Es posible especificar parámetros como parte de una URL.
- Utilizado con el método GET para especificar la búsqueda de un recurso o ejecución de un servicio
- Formato:  
baseURL/resource?  
parameter\_1=value\_1&parameter\_2=value\_2
- Ejemplo:  
[https://www.paper.com/publication?  
day=1&month=12&year=2019](https://www.paper.com/publication?day=1&month=12&year=2019)



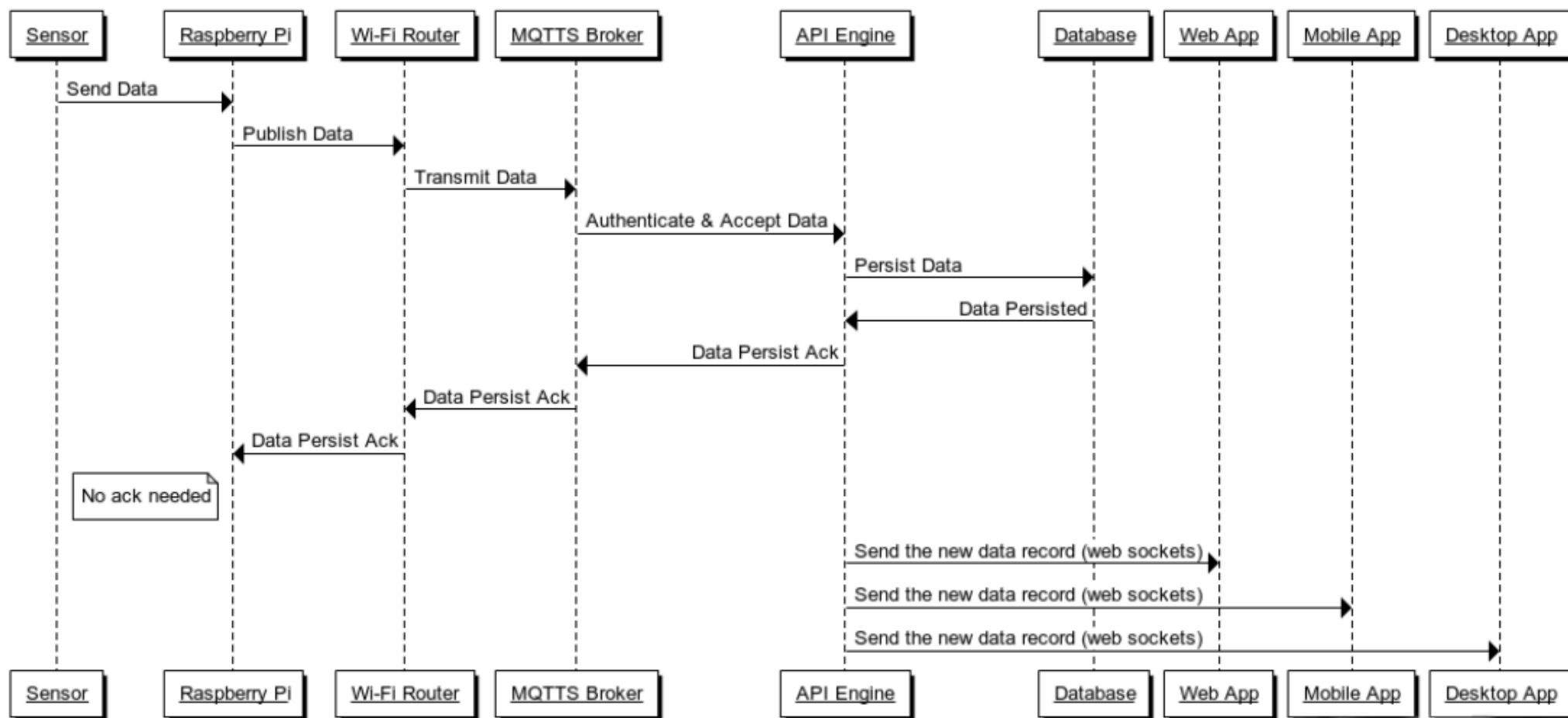
## Arquitectura cliente-servidor: Ejemplo





## Arquitectura cliente-servidor: Ejemplo

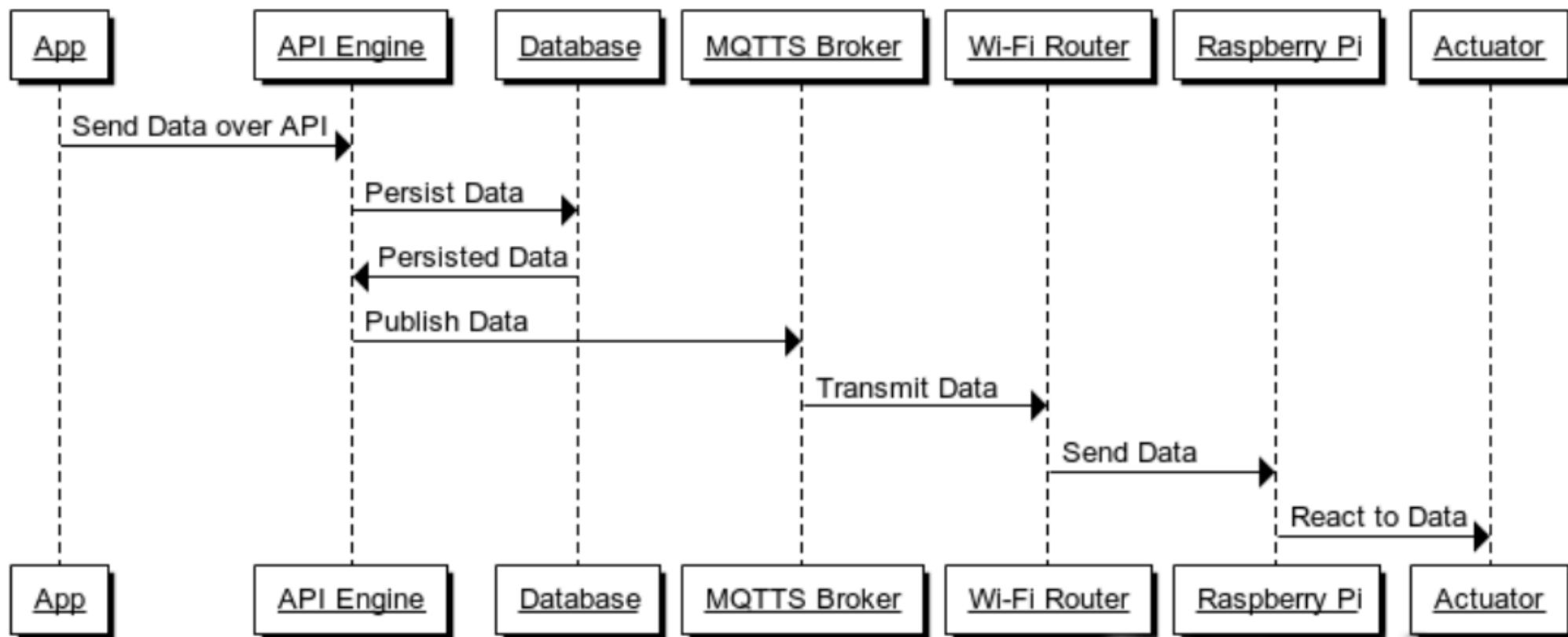
Data Flow from Sensor to Apps





## Arquitectura cliente-servidor: Ejemplo

Data Flow from Apps to Actuators

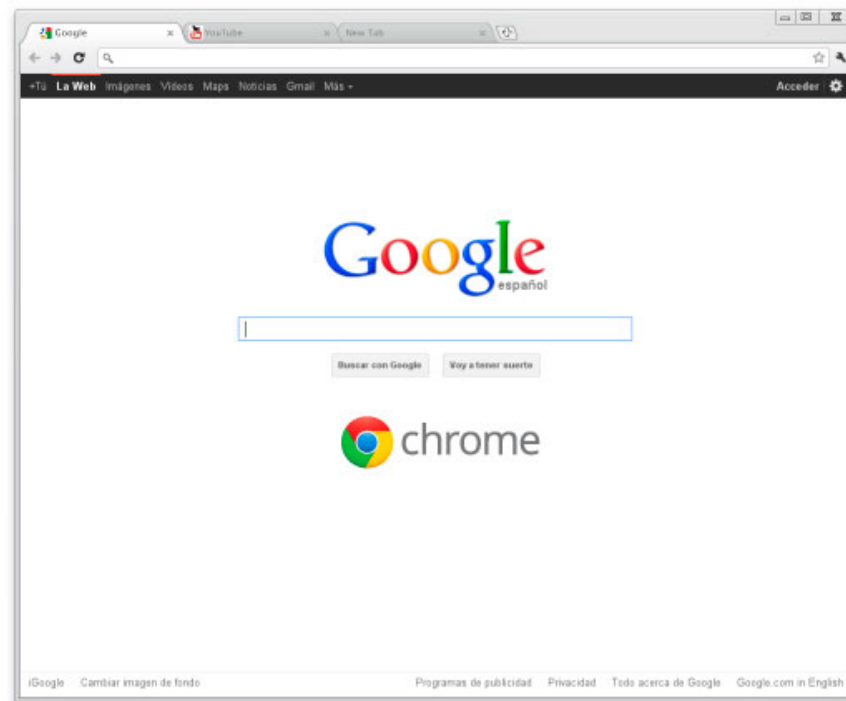




## Tecnologías: Lado cliente

- HTML
- CSS
- Javascript

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <link href="bootstrap.min.css" rel="stylesheet">
    <script src="run_prettify.js"></script>
  </head>
  <body>
    ...
```

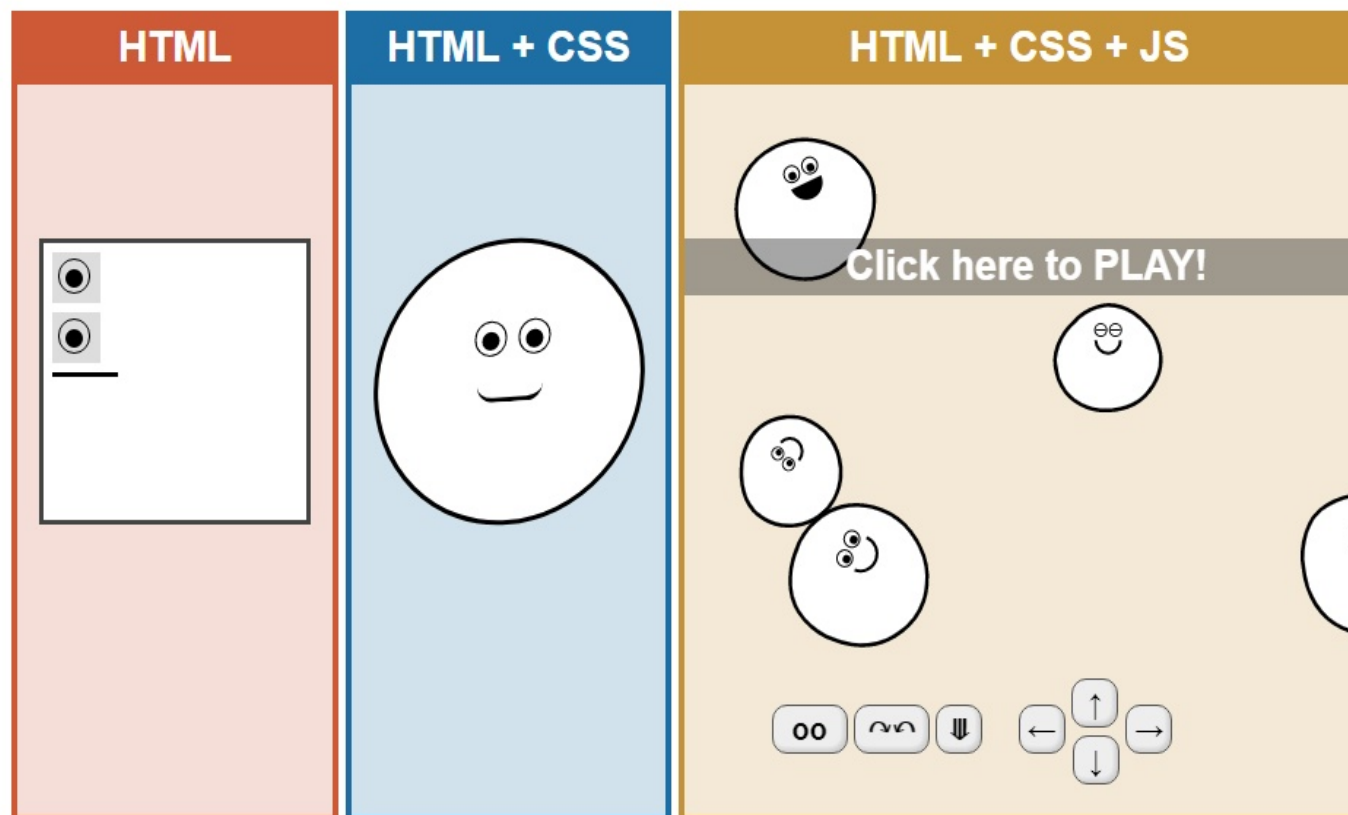






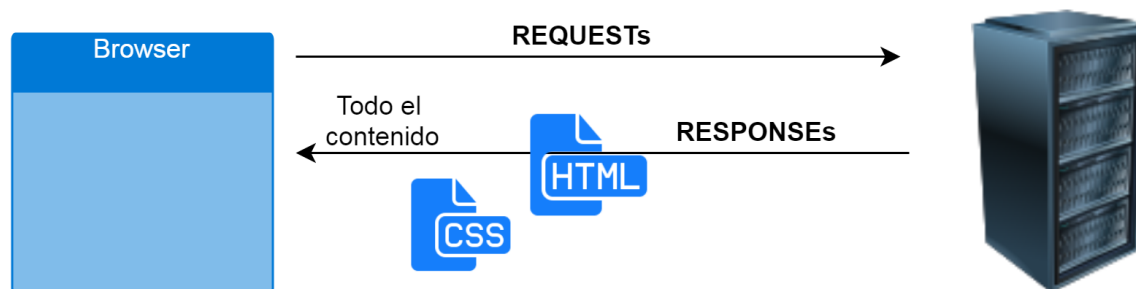
## Tecnologías: Lado cliente

- HTML
- CSS
- Javascript

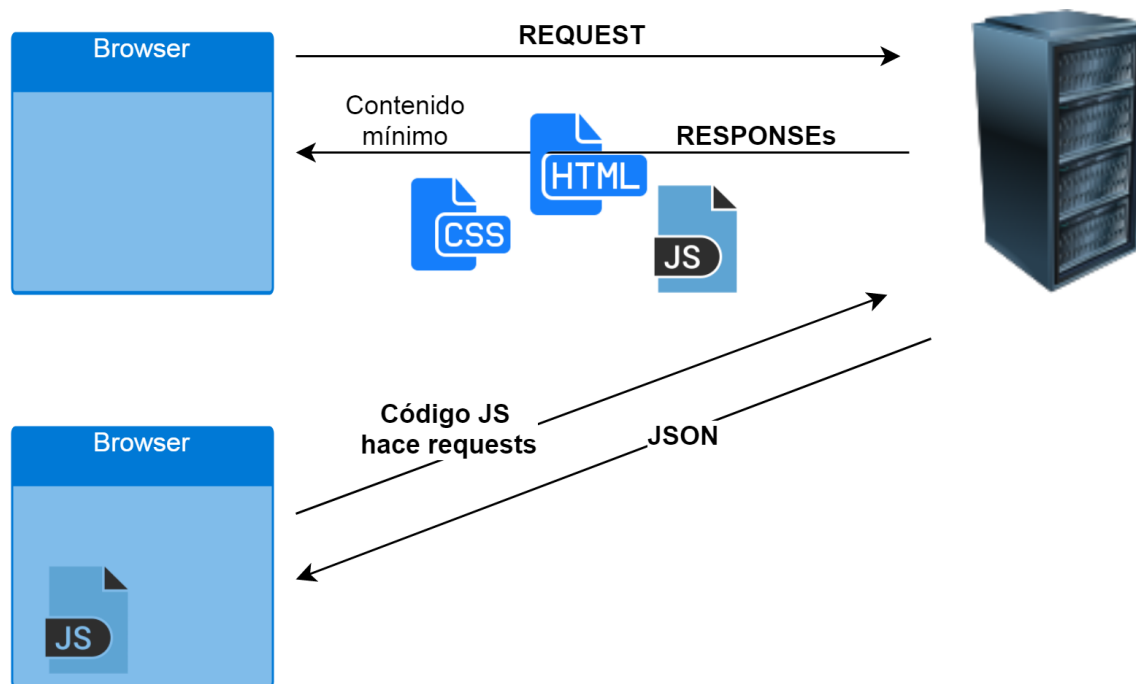




## static/Dynamic web app



## Single Page Application (SPA)





## **Tecnologías: Lado server**

- **Servidores web**
  - NodeJS (Javascript)
  - Tomcat/Jeti/Weblogic (Java)
  - Apache web server (PHP,C/C++,Python)
  - NGINX (contenido estático)
  - Lighttpd (C/C++,Python,PHP)



## Tecnologías: Lado server

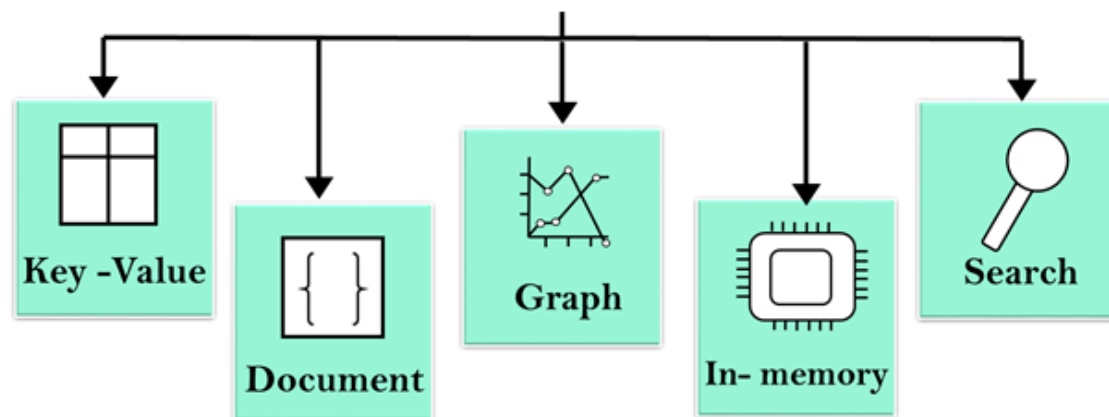
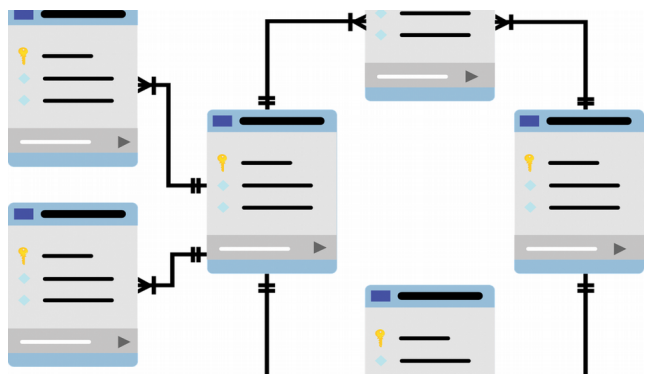
- **Bases de datos**

- Relacionales

- MySQL, MariaDB, Oracle, SQLITE

- No Relacionales

- MongoDB, Cassandra, Redis, Neo4j





## **Tipos de aplicaciones en servidor**

- Servidor de contenido estático (archivos multimedia, archivos js, etc.)
- Servidor de contenido web (HTML, CSS, JS)
- Servicio web RESTful (JSON, XML)
- Servidor websocket
- MQTT broker
- Otros



## Bibliografía

- Fielding, Roy T.; Gettys, James; Mogul, Jeffrey C.; Nielsen, Henrik Frystyk; Masinter, Larry; Leach, Paul J.; Berners-Lee, Tim (June 1999). Hypertext Transfer Protocol – HTTP/1.1
- Practical Internet of Things with JavaScript. Arvind Ravulavaru. 2017. Packt>
- Raspberry Pi for Arduino Users - Building IoT and Network Applications and devices. James R. Strickland. 2018. Apress.
- <https://restfulapi.net>
- [https://en.wikipedia.org/wiki/Hypertext\\_Transfer\\_Protocol](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol)
- <https://en.wikipedia.org/wiki/URL>
-