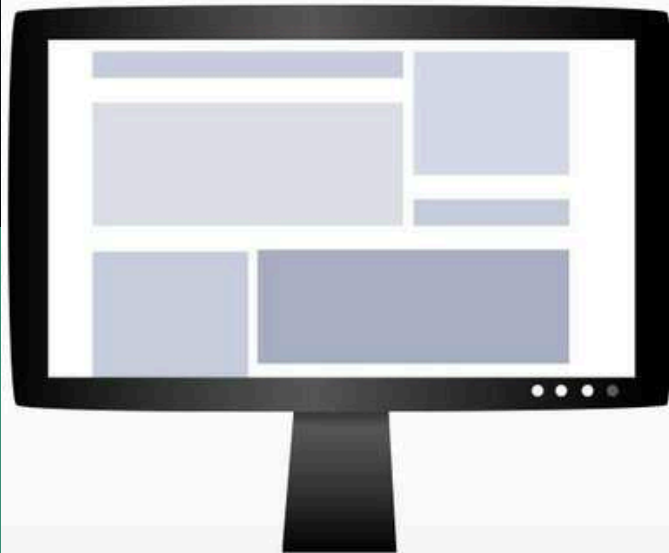




# BACKEND VS FRONTEND

## Front End



VS

## Back End



### *Frontend*

- Lo que ve e interactúa el usuario.
- Tecnologías: HTML, CSS, JS, React, Flutter.
- Ejemplo: Formularios, botones, diseño de la página.

### *Backend*

- La parte oculta que procesa datos y reglas de negocio.
- Tecnologías: Java, Python, C#, PHP, Node.js.
- Ejemplo: Guardar información, validar usuarios, conectarse a la base de datos.

# ¿QUÉ ES UNA API?

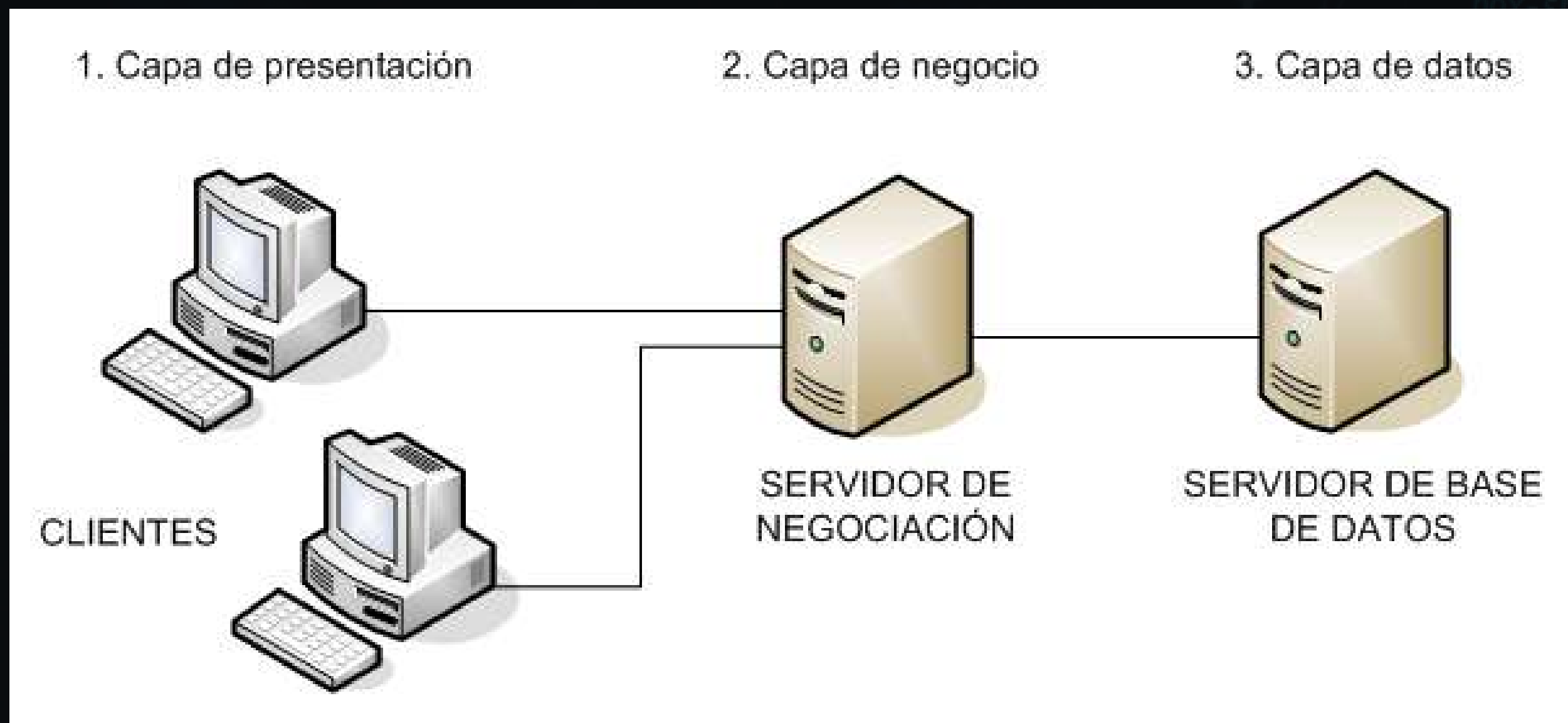


Es un conjunto de reglas que permite que dos aplicaciones se comuniquen entre sí.



# ARQUITECTURA DEL SOFTWARE MULTICAPA

- Modelo que divide una aplicación en capas independientes.





# API VS WEBSERVICE



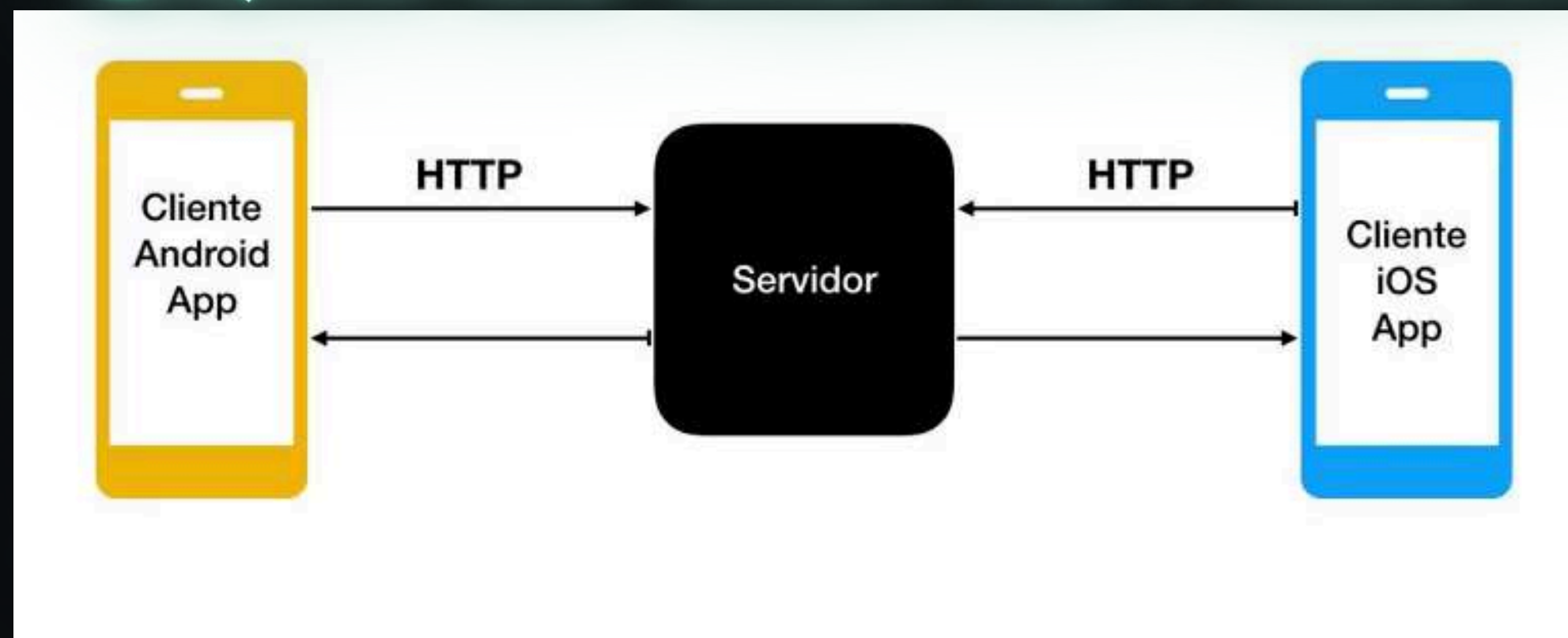
## **API**

- Puede comunicarse en distintos formatos (JSON, XML, etc.), no necesita estar en la web.

## **WebService**

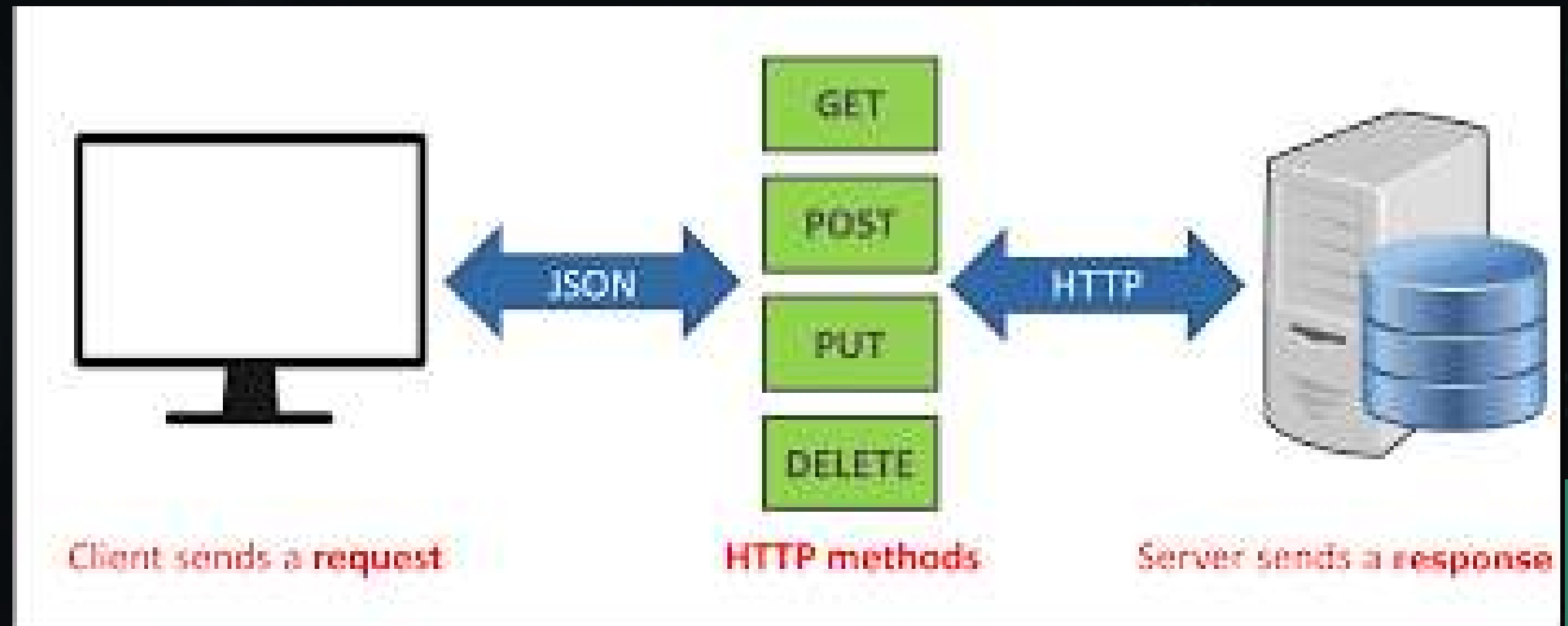
- Es un tipo de API que siempre funciona en la web y usa protocolos como SOAP.

# ¿QUÉ ES UNA API REST?



Es un tipo de API web que sigue el estilo arquitectónico REST, utilizando los verbos HTTP (GET, POST, PUT, DELETE) para operaciones sobre recursos identificados con URLs.

# ¿QUÉ ES UNA API HTTP?



Es un "mensajero" o un "intermediario" que permite que dos programas se comuniquen a través de Internet, usando el lenguaje web (HTTP)



# ¡GRACIAS!



401

Unauthorized

400

Bad Request

404

Page Not Found

403

Forbidden Error

503

Service Unavailable

```
box h2{
margin: 0 0 30px;
padding: 0;
color: #fff;
text-align: center;
}
box h3{
margin: 0 0 10px;
padding: 0;
color: #fff;
text-align: center;
}
box input{
```