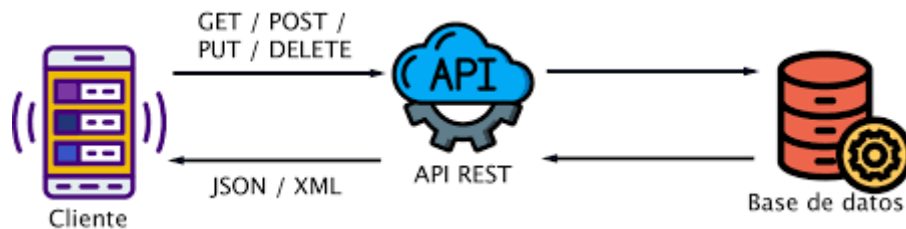


**REST** (Representational State Transfer) es un estilo de arquitectura para diseñar servicios web. Permite que distintos sistemas se comuniquen entre sí usando **protocolo HTTP**, de manera sencilla, ligera y escalable.

En una comunicación REST, un cliente (por ejemplo, una app o un navegador) se comunica con un servidor para enviar o recibir datos, normalmente en formato **JSON** o **XML**.

REST se usa para:

- Crear **API RESTful** que permiten a aplicaciones acceder a datos o servicios.
- Conectar sistemas diferentes, como apps móviles con servidores web.
- Automatizar procesos entre dispositivos en sistemas IoT.
- Crear servicios web escalables y fáciles de mantener.



**Características:**

- Usa **HTTP** como protocolo de comunicación.
- Utiliza los **métodos HTTP**:
  - GET → Obtener datos
  - POST → Enviar datos nuevos
  - PUT → Actualizar datos existentes
  - DELETE → Eliminar datos
- Es **sin estado**: cada petición lleva toda la información necesaria (no se guarda el estado entre peticiones).
- Usa **URLs** para identificar recursos.



#### Ejemplos de comunicación REST:

##### 1. Cliente solicita los datos de temperatura

###### Petición HTTP (GET):

GET /api/temperatura HTTP/1.1

Host: midominio.com

###### Respuesta del servidor (JSON):

```
{  
  "sensor_id": 101,  
  "valor": 24.7,  
  "unidad": "°C",  
  "fecha": "2025-04-18T15:30:00"  
}
```

##### 2. Cliente envía un nuevo dato de temperatura (ESP32)

###### Petición HTTP (POST):

POST /api/temperatura HTTP/1.1

Content-Type: application/json

```
{  
  "sensor_id": 101,  
  "valor": 25.1  
}
```

###### Respuesta del servidor:

```
{  
  "mensaje": "Dato guardado correctamente",  
  "estado": 201  
}
```

### **3. Cliente actualiza un dato (PUT)**

PUT /api/temperatura/101 HTTP/1.1

Content-Type: application/json

```
{  
  "valor": 26.3  
}
```

### **4. Cliente elimina un dato (DELETE)**

DELETE /api/temperatura/101 HTTP/1.1