



Consumo de datos desde una API REST

API REST

Implementar un aplicativo web que consume datos desde una API REST utilizando la librería Axios para dar solución a una problemática.

- Unidad 1: Introducción a Componentes Web y Vue Js
- Unidad 2: Binding de formularios
- Unidad 3: Templates y rendering en Vue
- Unidad 4: Manejo de eventos y reutilización de componentes
- Unidad 5: Consumo de datos desde una API REST



Te encuentras aquí



¿Qué aprenderás en esta sesión?

- *Verifica el funcionamiento de una API Rest publicada en un servidor utilizando una herramienta cliente.*

¿Tienes idea de cómo
se pueden comunicar
2 aplicaciones?

Frontend ↔ Backend



/* API REST */

¿Qué es una API REST?

Concepto de API

Una **API** es un **conjunto de propiedades y/o métodos** que a través de una capa de abstracción nos entregan información o la posibilidad de ejecutar funciones de una entidad.

Por sus siglas **A**pplication **P**rogramming **I**nterface, utilizamos la programación para encapsular datos y métodos permitiéndole a un usuario acceder a éstas a través de una interfaz.

Veamos un ejemplo nativo con JavaScript:

¿Qué es una API REST?

Concepto de API

Si ejecutamos en una consola `new Date()` obtendremos como respuesta la fecha actual en el siguiente formato:

Sat Nov 12 2022 11:13:31 GMT-0300 (hora de verano de Chile)

Aunque parezca un String ordinario, es el resultado de un método dentro de la API **Date**.

¿Qué es una API REST?

Concepto de API

Si escribimos **new Date()** . veremos cómo obtenemos una lista de sugerencias de muchos diferentes métodos.

Todos y cada uno de estos métodos **pertenece** a la API **Date**.

Aunque no sepamos cómo funciona internamente, podemos ejecutarlos y obtener sus resultados.

```
> new Date().__defineGetter__  
getDate  
getDay  
getFullYear  
getHours  
getMilliseconds  
getMinutes  
getMonth  
getSeconds  
getTime  
getTimezoneOffset  
getUTCDate  
getUTCDay  
getUTCFullYear  
getUTCHours  
getUTCMilliseconds  
getUTCMinutes  
getUTCMonth  
getUTCSeconds  
getYear
```


¿Qué es una API REST?

Concepto de REST

REST es una arquitectura de software que utiliza el protocolo HTTP para comunicar 2 aplicaciones distintas que quieran intercambiar información.

Esta arquitectura utiliza diferentes métodos para indicar la intención de la comunicación.

- **GET**
- **POST**
- **PUT**
- **DELETE**

Cada uno de estos verbos serán interpretados por una de las aplicaciones y conllevarán a la ejecución de alguna función.

¿Qué es una API REST?

Concepto de REST

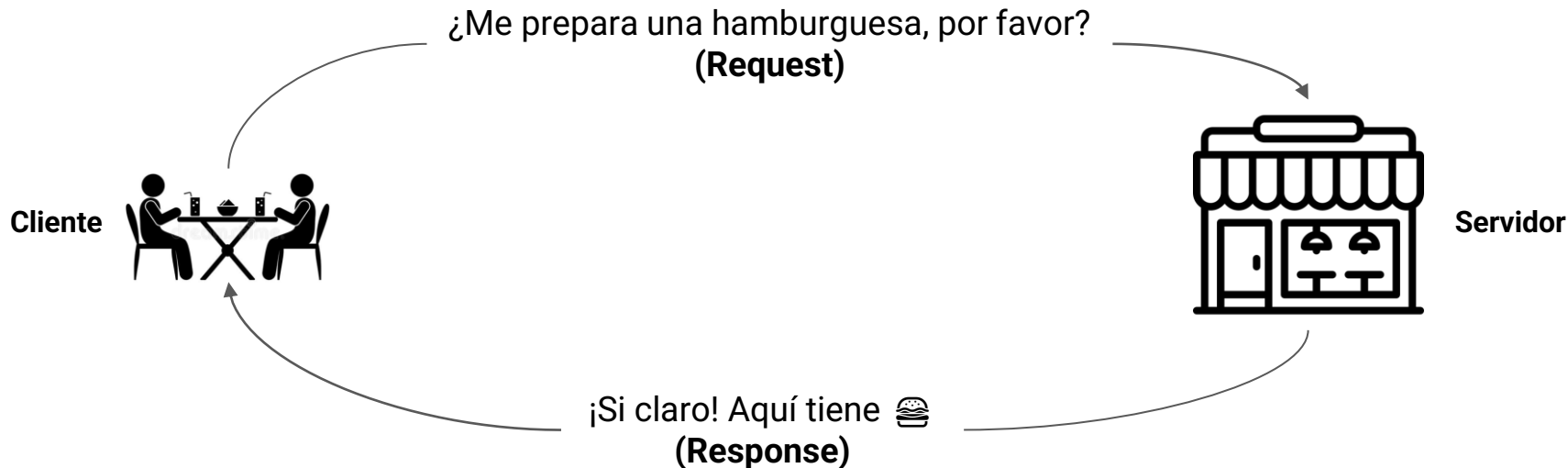
Podemos asociar cada uno de estos métodos a las operaciones **CRUD**.

- POST → **CREATE**
- GET → **READ**
- PUT → **UPDATE**
- DELETE → **DELETE**

¿Qué es una API REST?

Concepto de REST

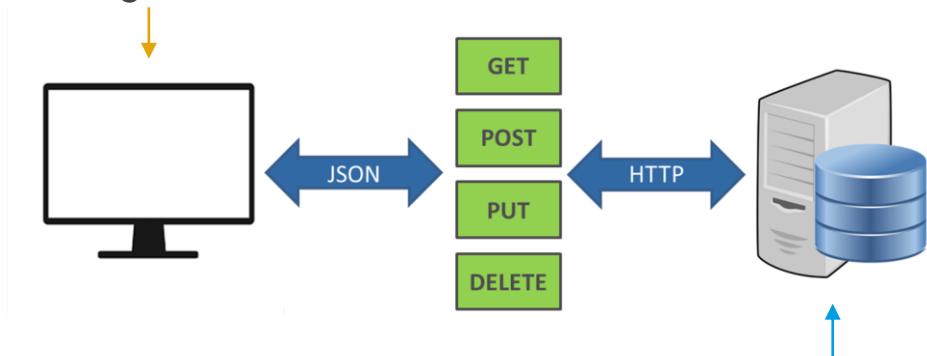
Pudiéramos comparar esta arquitectura con el proceso de pedidos en un restaurante:



¿Qué es una API REST?

API REST

Si unimos ambos conceptos asignando el sujeto de **cliente** a una aplicación **frontend** donde los usuarios interactúan y **servidor** a una aplicación **backend** donde se alojan los datos obtenemos el siguiente diagrama.

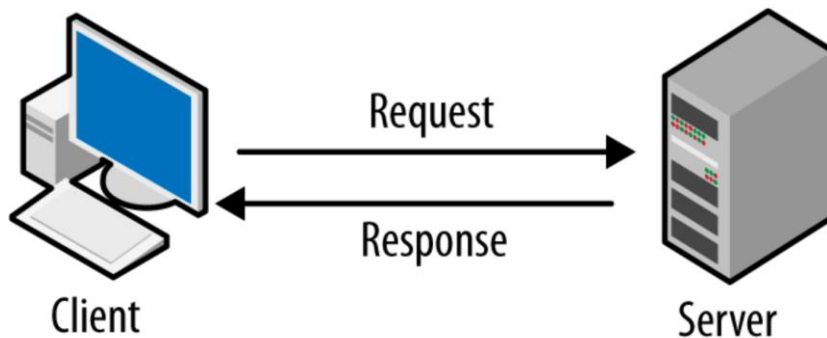


*La información que se envía u obtiene en la comunicación comúnmente será en formato **JSON**.*

API REST

Composición de un request/response

Las comunicaciones en las API REST se producen con la existencia de un **request** de parte de la aplicación cliente.




Posterior al **request**, el servidor procesa la consulta y devuelve un **response** al cliente.

API REST

Composición de un request/response

Tanto el request como el response están compuestos por varios componentes, 3 de ellos son el **header**, **body** y **status**.

Name	×	Headers	Preview	Response	Initiator	Timing	Cookies
 api	▼ General						
Request URL: https://randomuser.me/api							
Request Method: GET							
Status Code: 🟢 200							
Remote Address: 172.67.144.77:443							
Referrer Policy: strict-origin-when-cross-origin							
▶ Response Headers							(14)
▶ Request Headers							(19)

API REST

header

El **header** incluye los metadatos asociadas a la consulta o respuesta.

Entre los diferentes metadatos podemos encontrarnos con:

- **method** → GET, POST, PUT, DELETE...
- **query strings** → ?param=123
- **params** → /usuarios/:id
- **tokens** → Authorization: Bearer <token>

API REST

Body

El **body** contiene la información en bruto que se está trasladando en la comunicación.

Está relacionado al **payload** de la consulta o respuesta.

Típicamente optamos por enviar y recibir datos en formato JSON.

```
1 // 20221112130521
2 // https://mindicador.cl/api
3
4 {
5   "version": "1.7.0",
6   "autor": "mindicador.cl",
7   "fecha": "2022-11-12T16:00:00.000Z",
8   "uf": {
9     "codigo": "uf",
10    "nombre": "Unidad de fomento (UF)",
11    "unidad_medida": "Pesos",
12    "fecha": "2022-11-12T03:00:00.000Z",
13    "valor": 34707.78
14  },
15   "ivp": {
16     "codigo": "ivp",
17     "nombre": "Indice de valor promedio (IVP)",
18     "unidad_medida": "Pesos",
19     "fecha": "2022-11-12T03:00:00.000Z",
20     "valor": 35371.05
21   },
22   "dolar": {
23     "codigo": "dolar",
24     "nombre": "Dólar observado",
```


API REST

Status

El **status** se refiere al estado de la comunicación.

Éste se determina por un número que representa una situación determinada.

- **200:** OK
- **201:** Recurso creado
- **400:** Consulta mal realizada
- **404:** Recurso no encontrado
- **500:** Error del servidor

API REST

Web services

En la web contamos con diferentes servicios que podemos usar para complementar y agregar funcionalidades a nuestras aplicaciones.

Por ejemplo, si necesitáramos en una aplicación utilizar la geolocalización para indicar un punto geográfico o la ruta entre 2 puntos podemos ocupar el servicio de Google Maps.



Google maps posee su propia API y documentación para ser utilizada por cualquier cliente.

Como éste servicio existen muchos otros que podemos usar gratuitamente ó a través de suscripciones de pago.

¿Qué concepto te
costó más entender?



Investiga en internet otro
Web Service como el de
Google Maps y compártelo
con tus compañeros





Próxima sesión...

- *Verifica el funcionamiento de una API Rest publicada en un servidor utilizando una herramienta cliente.
(continuación)*

{desafío}
latam_

*Academia de
talentos digitales*

