



# TEMA: APIs & WEB SERVICES





# Resultados de aprendizaje.

**Desarrollar API's para aplicaciones móviles según requerimientos del usuario.**

- Definir que es una API.
- Comprender la importancia del uso de las API en las aplicaciones móviles.
- Identificar los elementos claves de una API al momento de su desarrollo.
- Comprender los tipos mas comunes de API existentes y su forma de uso.
- Lenguajes de programación para el desarrollo de API.
- Conocer la terminología de trabajo con las APIs.



# EJEMPLOS

**Caso 1:** Supongamos que se desea desarrollar una aplicación tipo UBER (Transporte, seguimiento o ubicación de vehículos, entre otros).

**¿Qué se necesita?** Mapas y geolocalización.

**¿Cómo harías los mapas desde cero?**



*“Imagínate por un momento desarrollar toda la tecnología necesaria para crear la aplicación desde cero”*

**Solución:** Lo correcto sería conectarnos a los mapas de Google (Google maps) y usar los mapas de ellos. Se aprovecha el desarrollo de terceros para la creación de nuevas aplicaciones que necesiten hacer uso de mapas.



# EJEMPLOS

**Caso 2:** Supongamos que se desea desarrollar una aplicación tipo e-commerce (Comercio electrónico, comercio por Internet o comercio en línea). Compra y venta de productos o de servicios a través de internet.

**¿Qué se necesita?** Realizar cobros por Internet utilizando diferentes medios de pagos.

**¿Cómo harías para cobrar por Internet?**



*“Imagínate por un momento desarrollar toda la tecnología necesaria para crear la aplicación desde cero”*

**Solución:** Lo correcto sería conectarnos a una pasarela de pago como PayPal, Pagadito, Wompi, entre otros.



# EJEMPLOS

**Piensa por un momento...**



**¿Qué otros ejemplos de desarrollo de aplicaciones podrías plantear?**

**Nota:** En cualquiera de los casos se aprovecha el desarrollo que han realizado otras empresas para tú aplicación. Por acá vamos dando entrada al concepto de APIs.



# Definición de API



- API es el acrónimo inglés “Application Programming Interface”, es decir, “Interfaz de Programación de Aplicaciones”.
- Una «interfaz» es la forma en que dos aplicaciones o servicios se comunican entre sí. Lo hacen exponiendo al resto de aplicaciones el **conjunto de servicios** disponibles en cada una y cómo se deben acceder.

*A la serie de estos servicios, se le denomina API. Por eso, estas sirven para que una aplicación pueda interactuar con otra.*



# Ventajas de las APIs

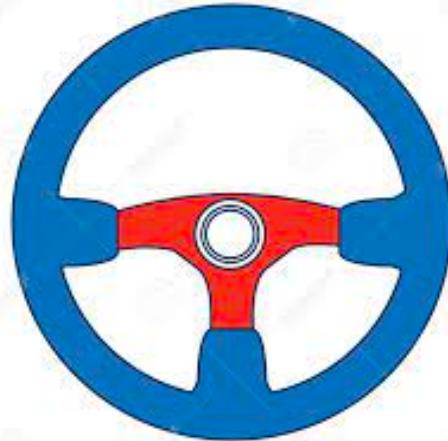


- Permiten la reutilización de código, reduciendo los tiempos y costes asociados al desarrollo de aplicaciones.
- Incrementan la interoperabilidad entre aplicaciones, con el ahorro de costes que esto supone, ya que las máquinas se comunican información entre sí más rápido que los humanos.
- Permiten la integración con otras aplicaciones.
- Hacer que una aplicación pueda conectarse con otras le da muchísimo más poder en comparación con una simple aplicación aislada.
- Permite la interacción con dos o más aplicaciones para consumir datos de una misma fuente de información.
- Permiten un desarrollo más eficiente de aplicaciones ya que se pueden utilizar APIs de terceros para crear nuevas aplicaciones en menor tiempo.
- Permite que una aplicación sea más adaptada a las necesidades de los usuarios según el propósito para el que fue desarrollada la API.



# Terminología con API.

**1. INTERFAZ.** Capa de abstracción para que dos sistemas se comuniquen.

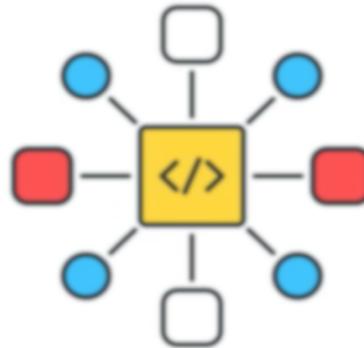


Ejemplo de **interacción con una capa de abstracción**. Mover el volante para que el vehículo gire en cierta dirección. No se necesita saber la mecánica que está por detrás para que haga una tarea concreta.

*Lleva en mente...una API es como una INTERFAZ.*



# Terminología con API.



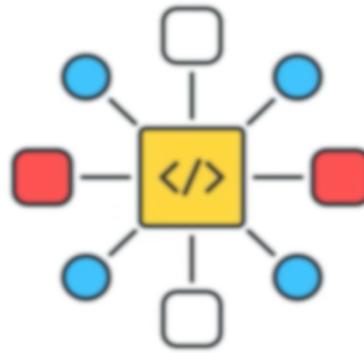
**API: Application Programming Interface o “interfaz de programación de aplicaciones”.**

En el contexto de las API, la palabra aplicación se refiere a cualquier software con una función distinta. La interfaz puede considerarse como un contrato de servicio entre dos aplicaciones.

Es una pieza de código que permite a diferentes aplicaciones comunicarse entre sí y compartir información y funcionalidades. Una API es un intermediario entre dos sistemas, que permite que una aplicación se comunique con otra y pida datos o acciones específicas.



# Terminología con API.

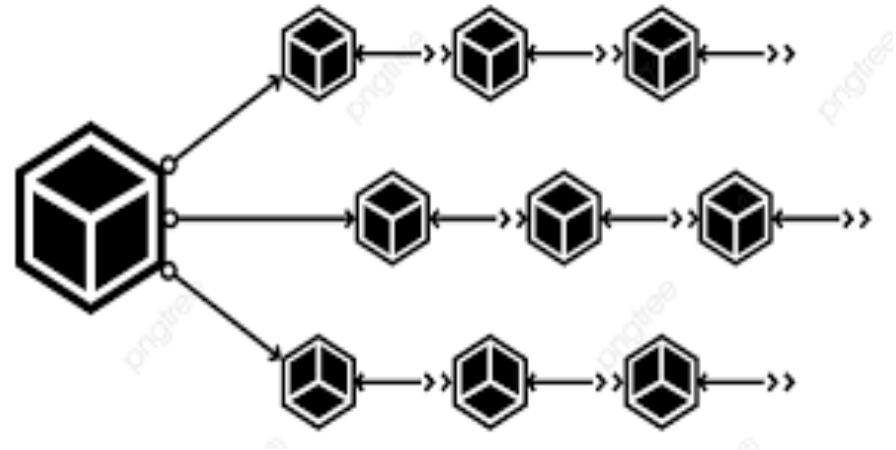
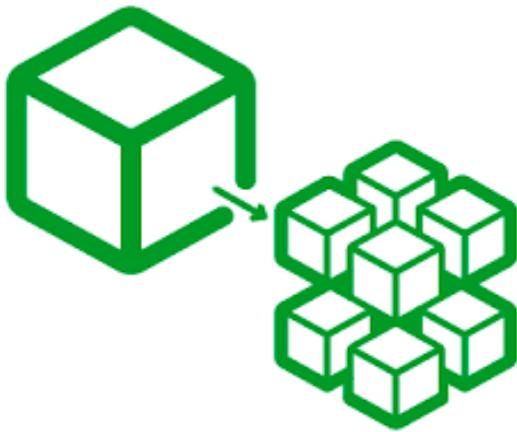


**API:** Es una Interfaz para que se comuniquen aplicaciones o programas de software y de esta manera poder compartir datos entre ellos.



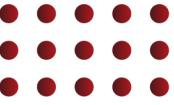


# ARQUITECTURA DE SOFTWARE

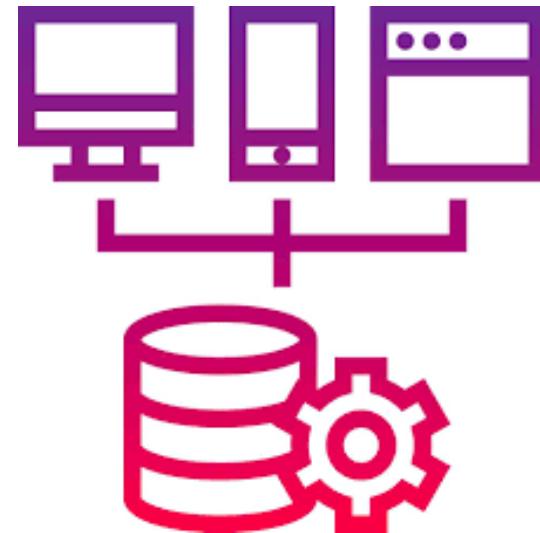


Es la forma en que está diseñado un sistema, como están organizados sus componentes, como se comunican entre ellos, que funciones cumplen, entre otros.





# SERVICIO WEB



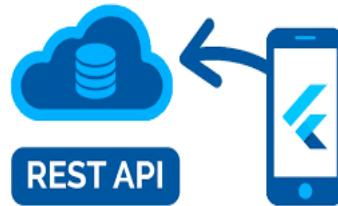
Es un servicio que permite la comunicación entre equipos que se encuentren en una red. Estos sistemas siguen un estandar de comunicación como lo es el protocolo HTTP (Mismo protocolo para navegar por Internet). Es la base para el tema de las **APIs remotas**, es decir, comunicación entre programas o aplicaciones que esten en distintos lugares (paises del mundo).





# SERVICIO REST

***REST: Transferencia de Estado Representacional o Representational State Transfer***



**REST:** Es una arquitectura.

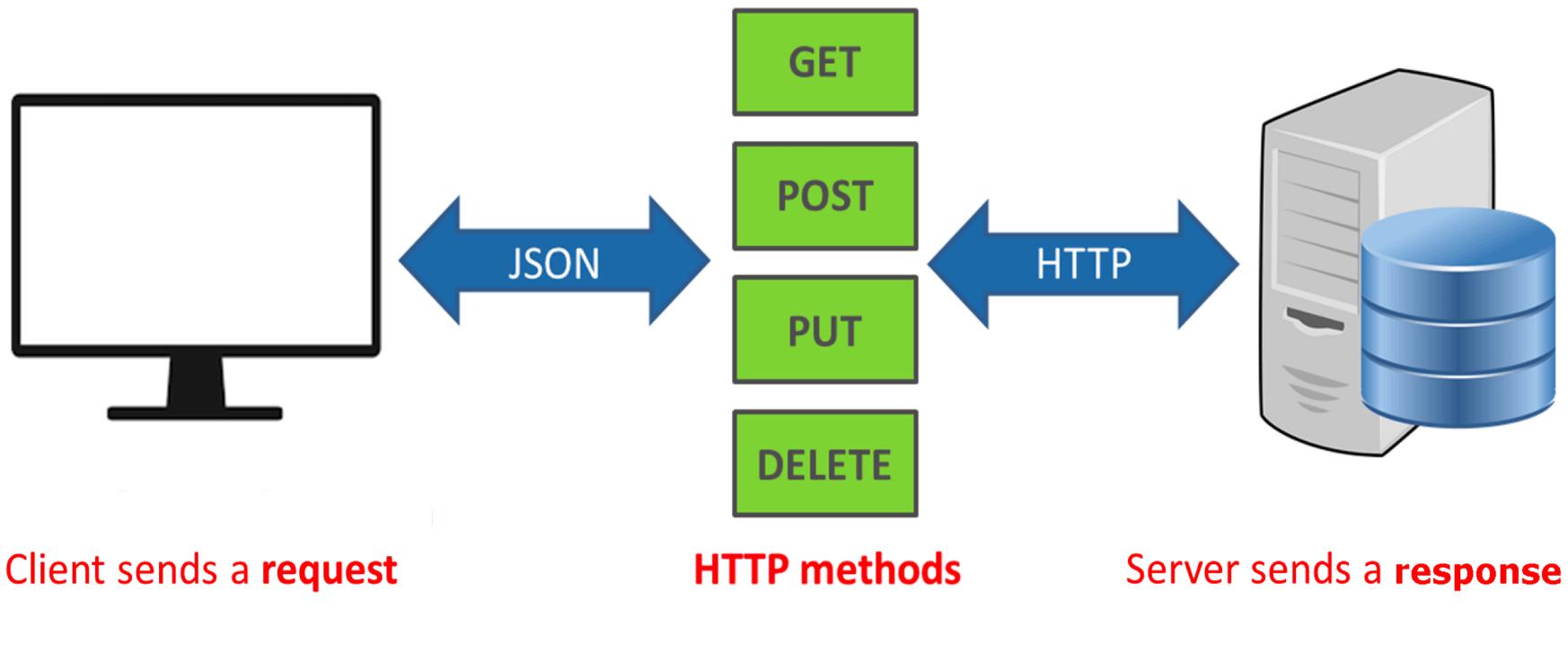
La transferencia de estado representacional o REST es un estilo de arquitectura software para sistemas hipermedia distribuidos como la ***World Wide Web (www o Internet)***.

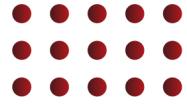
**REST** es una interfaz para conectar varios sistemas basados en el protocolo HTTP y nos sirve para obtener y generar datos y operaciones, devolviendo esos datos en formatos muy específicos, como **XML, JSON** y a veces simplemente en **texto plano**.

*Las APIs pueden ser de diferente tipo y la API REST es un tipo.*

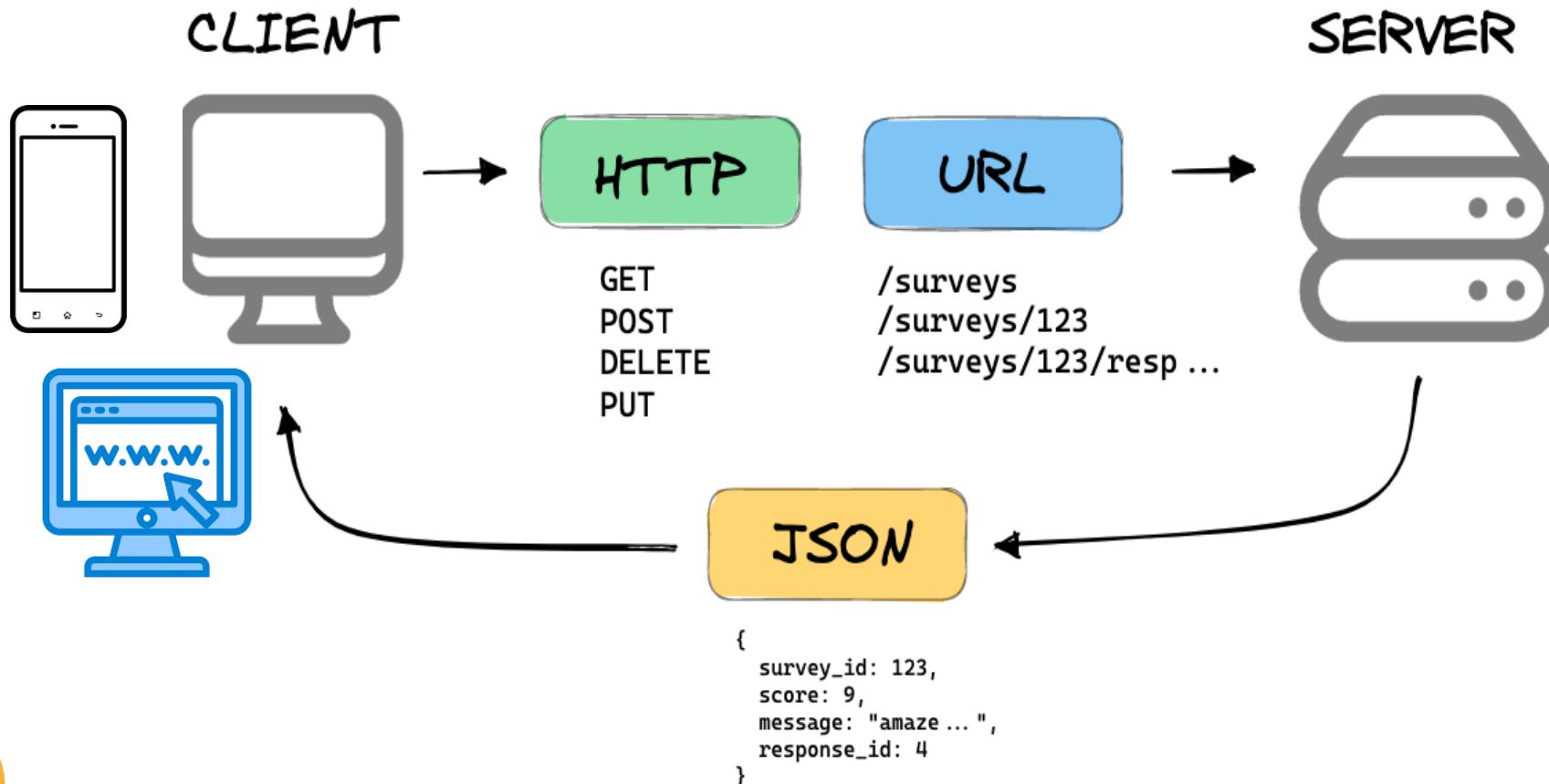
# SERVICIO REST

Ejemplo de peticiones HTTP.





# Como funciona una REST API





# Formatos para enviar y recibir datos entre aplicaciones



```
<Company>
  <Id>1</Id>
  <Name>EDteam</Name>
  <Sector>Education</Sector>
</Company>
```

XML

Extensible Markup Language



```
[
  {
    "id"      : 1,
    "name"    : "EDteam",
    "sector"  : "Education",
  }
]
```

JSON

JavaScript Object Notation

*Las APIs pueden ser públicas o privadas.*



# TIPOS DE APIs

## API Pùblicas:

Todo mundo puede acceder y hacer uso de estas para consultar información.

## API Privadas:

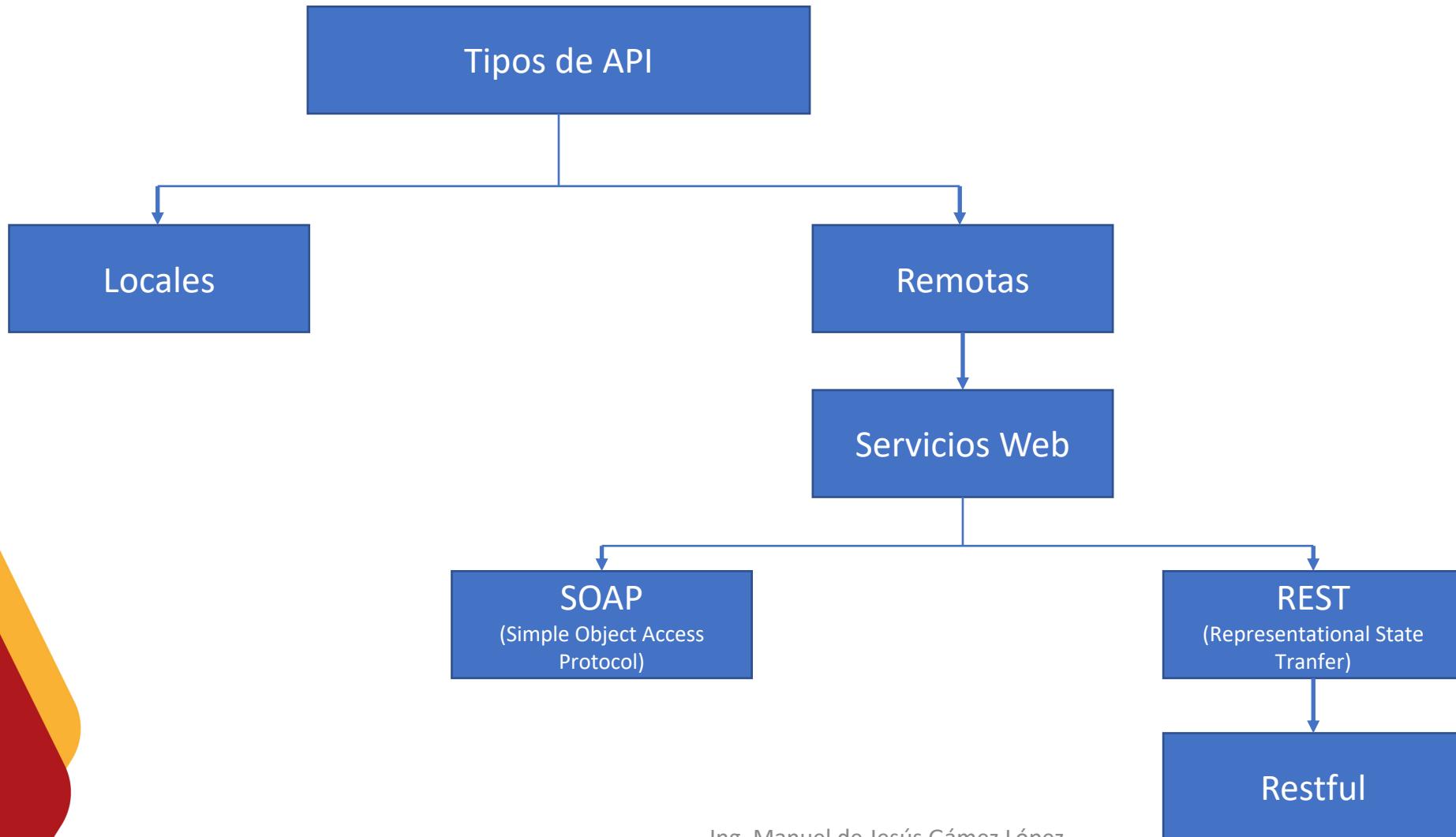
Una API privada requiere una autenticación. Normalmente cuando te autenticas la primera vez el servidor te devuelve un token. ***El token es un objeto que contiene todos los datos de la autenticación.*** Sucede que cada vez que se solicite información adicional, el servidor revisará si el token está vigente para ya no pedir una nueva autenticación y de esta manera dar paso a la petición entrante para posteriormente generar una respuesta.

## El formato más común para los token en API REST es JWT.

JSON Web Token (JWT) es un estàndar para transmitir informaciòn de forma segura en internet, por medio de archivos en formato JSON, que es un tipo de archivo de texto plano con el cual se pueden crear parámetros y asignarles un valor.



# TIPOS DE APIs





# DESARROLLAR UNA API REST

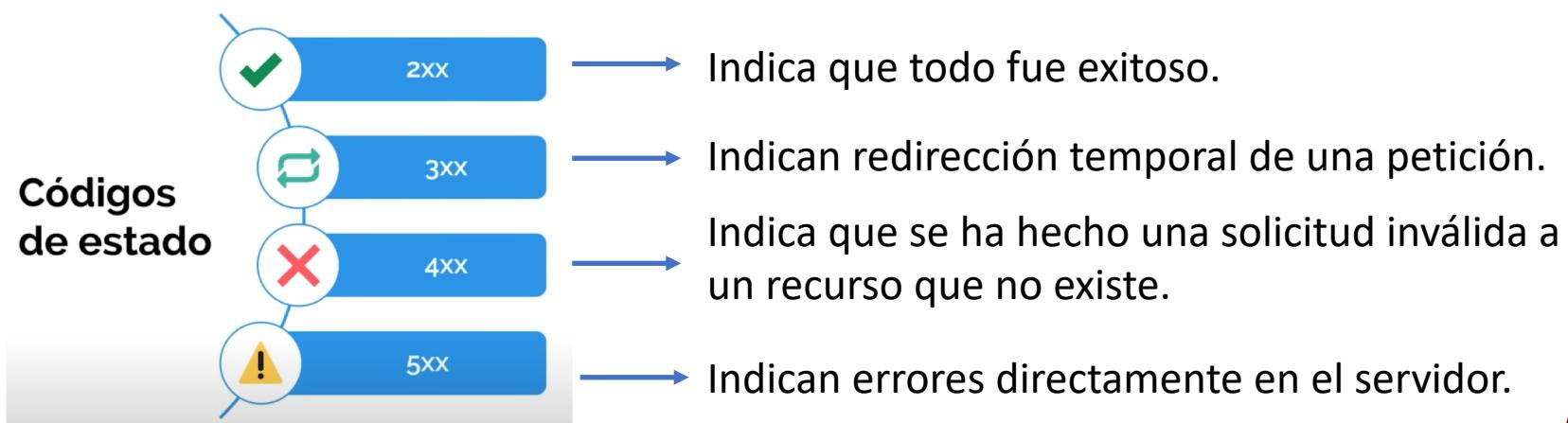
*Puntos clave a considerar al momento de crear una API.*

- **Consultar recursos (URI).** Los famosos endpoints o URL completa. La URI permite consultar directamente un recurso.

Por ejemplo: <https://servicestechology.com.sv/data>

Es un identificador único que permite buscar y encontrar cualquier recurso que se necesite consumir por medio de la URI desde la aplicación.

- **Códigos de estado:** Cuando se solicita información a través de una API el servidor puede contestar con distintos códigos. Estos códigos permiten saber qué paso con una petición.

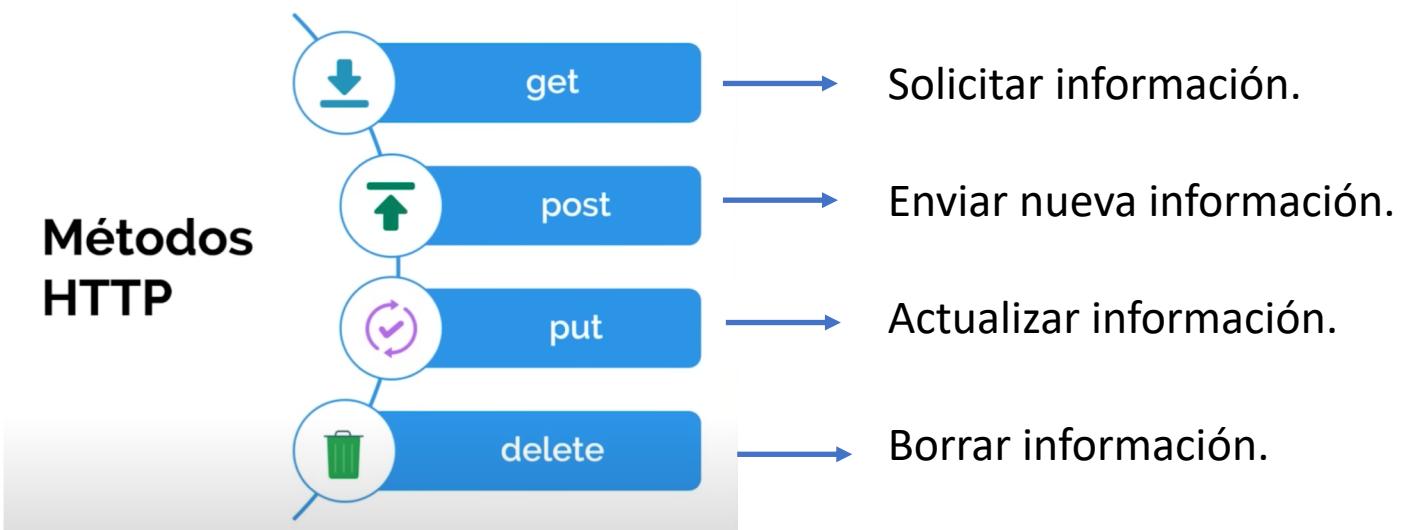




# DESARROLLAR UNA API REST

*Puntos clave a considerar al momento de crear una API.*

- **Métodos HTTP:** Estos métodos permiten interactuar con la API.



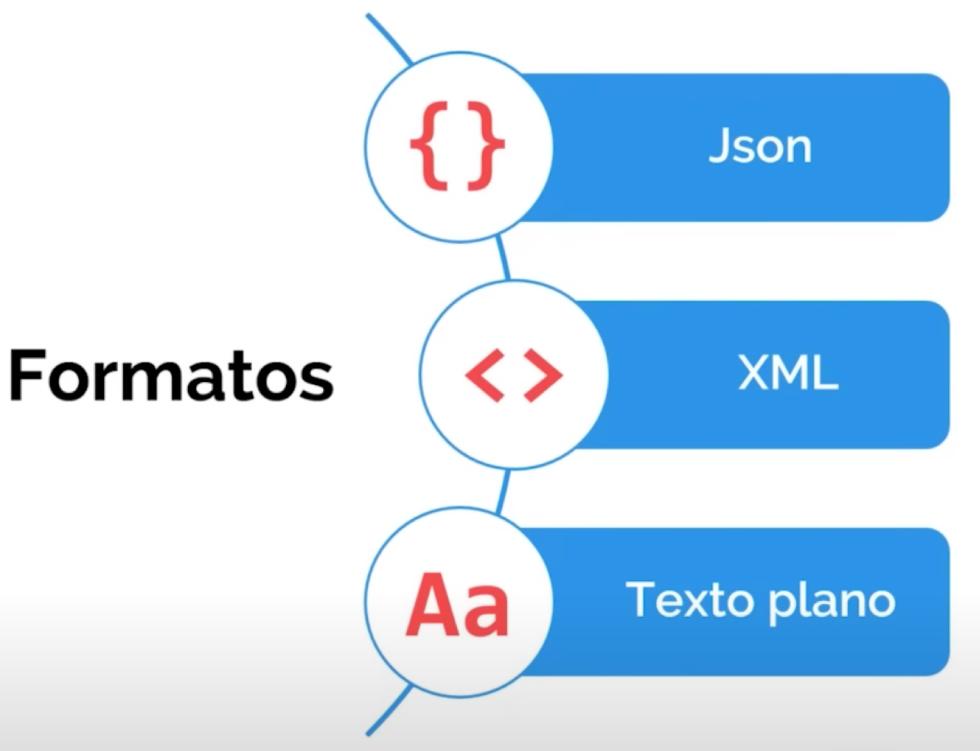
*Nota: Esto es equivalente al típico CRUD con las base de datos, pero no es lo mismo, Pero es equivalente.*



# DESARROLLAR UNA API REST

*Puntos clave a considerar al momento de crear una API.*

- **Formatos:** Las API pueden devolver la información en distinto formato.





# DESARROLLAR UNA API REST

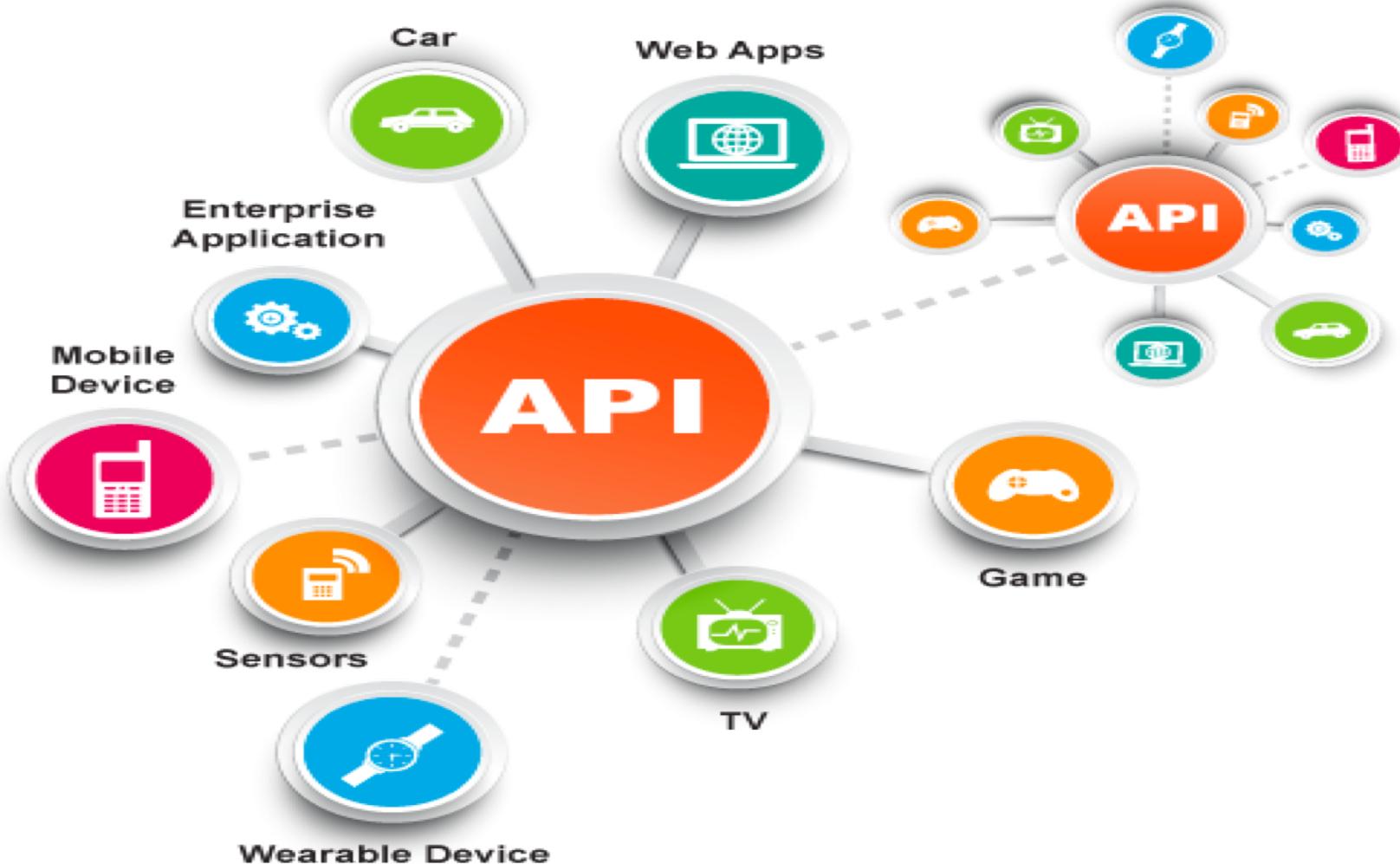
*Puntos clave a considerar al momento de crear una API.*

- **Buenas prácticas:** HATEOAS son las siglas de "Hypermedia As The Engine Of Application State" Especifica que las API REST deben proveer de información suficiente al cliente para interactuar con el servidor.
    - Significa que la API se autodescribe, cada recurso tiene información de cual seria el recurso siguiente o de la cantidad de recursos totales que hay.
    - Significa protección de la API privada.
    - Significa hacer pruebas de funcionalidad de la API. Asegurarte que todo funcione correctamente.
    - Significa documentar claramente el proceso para que otros desarrolladores puedan integrar la API con otras aplicaciones de manera fácil.
- 

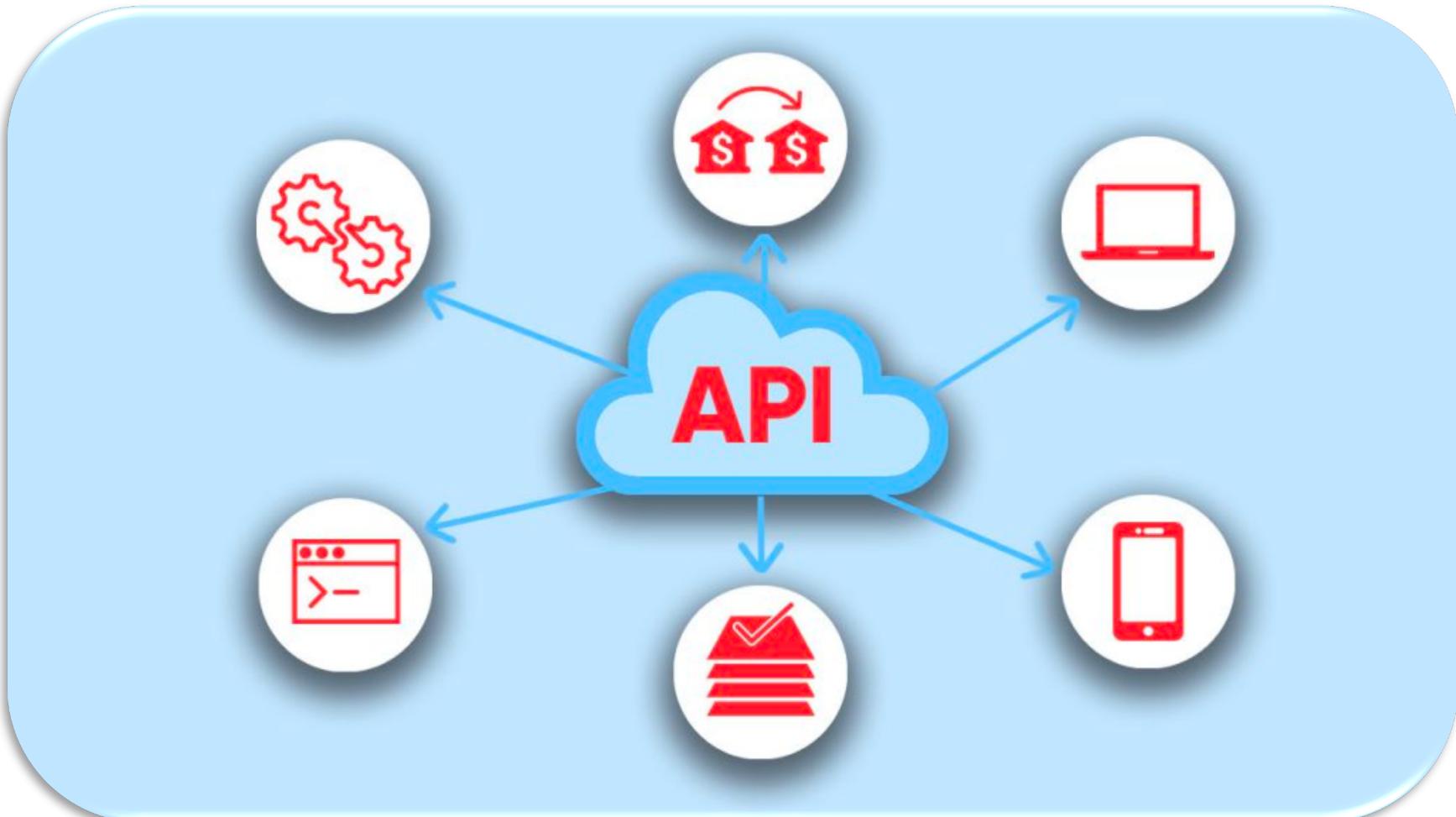
## Buenas prácticas

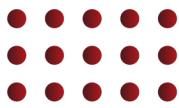
*Nota: Cuando se crea una API la intención de la misma es para que alguien la consuma, ya que es una interfaz o sistema de comunicación entre aplicaciones.*

# EJEMPLOS DE INTERACCIÓN DE APLICACIONES CON API.



# EJEMPLOS DE INTERACCIÓN DE APLICACIONES CON API.





# API / WEB SERVICE.

#	Web Services.	API
1	Interacción entre dos máquinas a través de una red.	Interacción entre dos API.
2	Utiliza SOAP, REST y XML como medio de comunicación.	Puede utilizar cualquier medio de comunicación.
3	Todos los servicios web son API.	Todas las API no son servicios web.
4	Un servicio Web siempre necesita una red para su funcionamiento.	API no necesita una red para su funcionamiento.
5	Los WebServices son servicios disponibles a través de Internet. Puede llamar a estos servicios y obtener información en su aplicación sin saber cómo funciona. Por ejemplo, los servicios web meteorológicos le brindan información sobre el clima de la ciudad.	API es una colección de clases que le brindan algunas funciones, como la API de Google que brinda la búsqueda de Google.

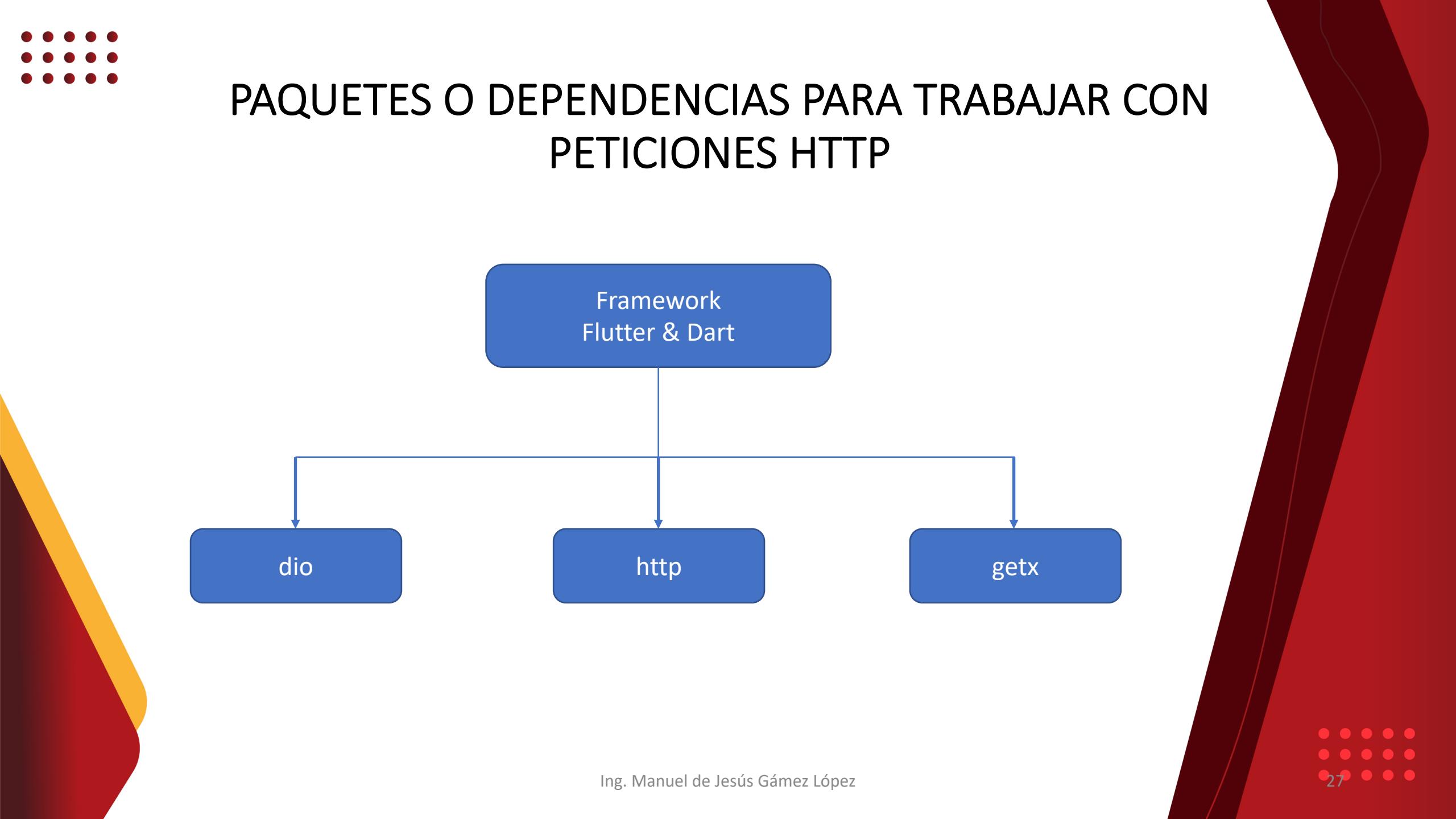


# HERRAMIENTA PARA TESTEO DE APIs Y WEB SERVICE

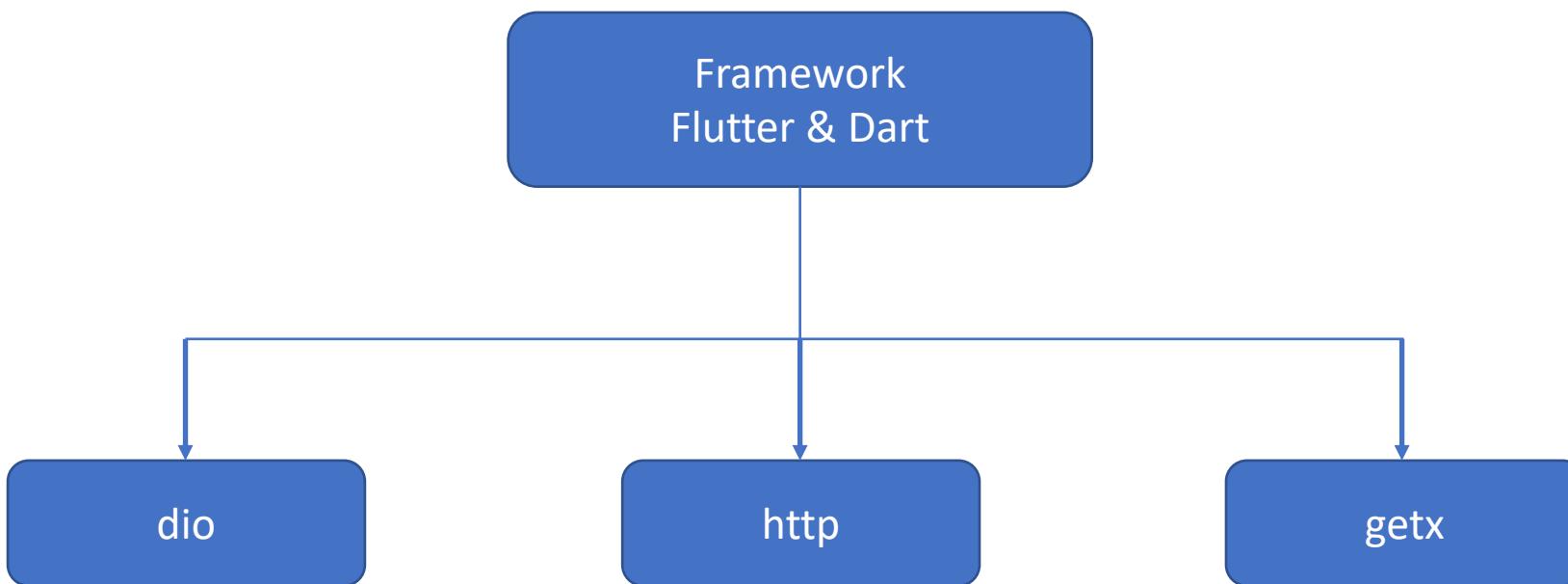


**Postman** es una aplicación que nos permite realizar pruebas API. Es un cliente HTTP que nos da la posibilidad de testear 'HTTP requests' a través de una interfaz gráfica de usuario, por medio de la cual obtendremos diferentes tipos de respuesta que posteriormente deberán ser validados.

**Postman** es una plataforma API para desarrolladores.



# PAQUETES O DEPENDENCIAS PARA TRABAJAR CON PETICIONES HTTP





# Conclusiones

- Las API son interfaces para que programas de software se comuniquen entre sí y comparten datos bajo ciertos estándares.
- Existen diferentes tipos de API desarrolladas por terceros, las cuales pueden ser consumidas por otros desarrolladores según la documentación proporcionada para el desarrollador de una API determinada.
- Las API Permiten tener una interfaz o sistema de comunicaciones para que las aplicaciones de software se comuniquen entre sí.
- Existen diferentes factores a considerar al momento de desarrollar una API.
- La importancia de las APIs es que las aplicaciones tengan un medio o canal de comunicación para interactuar e intercambiar información, y, además, permiten hacer una aplicación mucho más enriquecida y se puede aprovechar el desarrollo de otras empresas en los proyectos.





# Investigar

## API REST Vs API RESTful

## API Vs Web Services