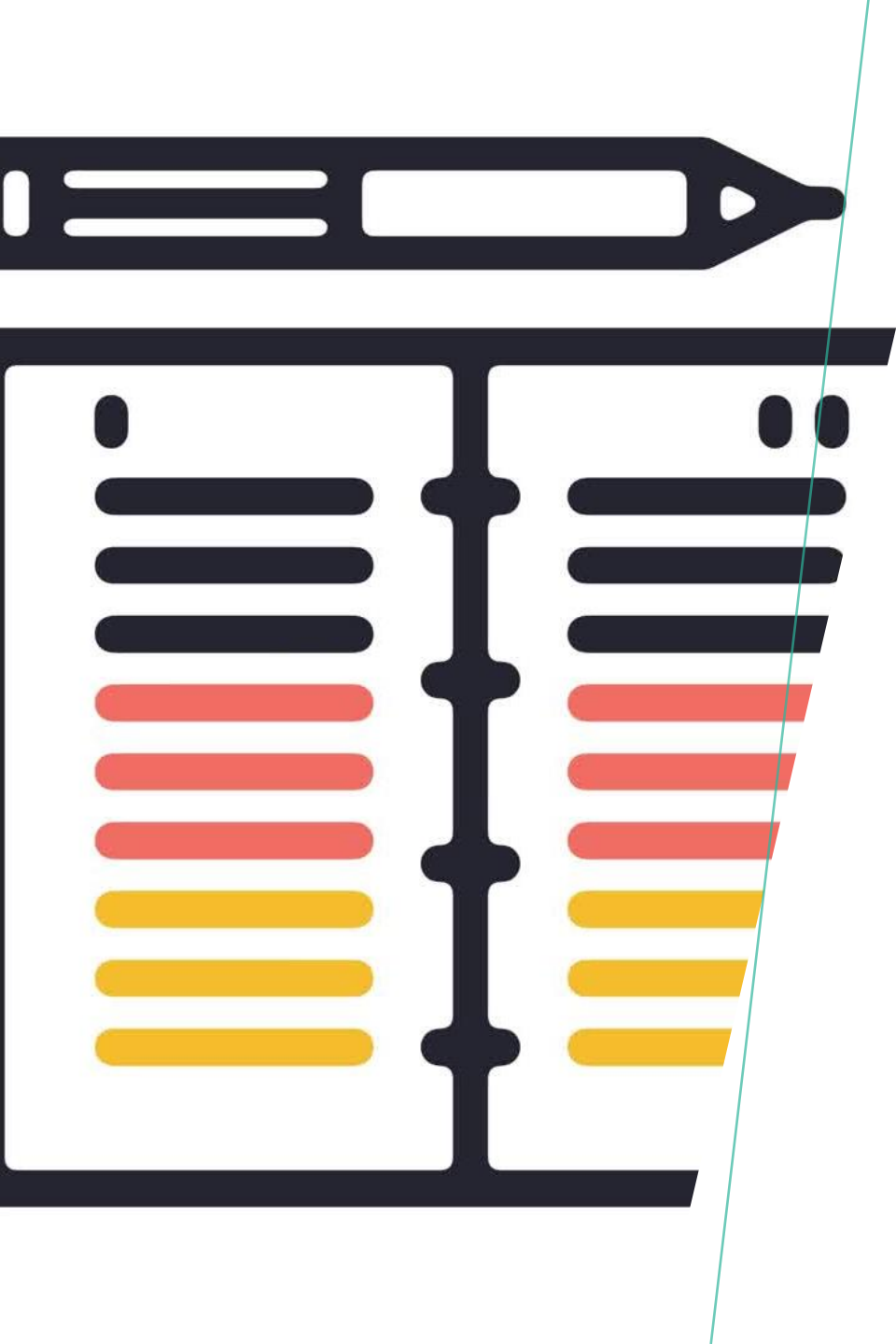


PROGRAMACIÓN WEB

CLASE 9





AGENDA

- Programación Web
- API Rest
- Herramientas Backend
- Dudas o Comentarios

```
if (n) {  
  if (a) {  
    for (; o > i; i++)  
      if (r = t.apply(e[i], n), r === !1) break  
  } else  
    for (i in e)  
      if (r = t.apply(e[i], n), r === !1) break  
} else if (a) {  
  for (; o > i; i++)  
    if (r = t.call(e[i], i, e[i]), r === !1) break  
} else  
  for (i in e)  
    if (r = t.call(e[i], i, e[i]), r === !1) break;  
return e  
},  
trim: b && !b.call("\uffeff\u00a0") ? function(e) {  
  return null == e ? "" : b.call(e)  
} : function(e) {  
  return null == e ? "" : (e + "").replace(C, "")  
},  
makeArray: function(e, t) {
```

PROGRAMACIÓN WEB

¿QUÉ ES?, TIPOS DE PROGRAMACIÓN WEB



¿QUÉ ES?

Se trata del proceso por el cual se diseña y desarrolla una página web; por medio de una serie de lenguajes que permiten dar forma al proyecto que se va a desarrollar.

Los lenguajes de programación web se caracterizan porque dan la posibilidad de enlazar a través de hipervínculos a otros sitios, de esta forma mejora el posicionamiento y calidad del contenido que se le muestra al usuario.

TIPOS DE PROGRAMACIÓN WEB

- Frontend
- Backend





FRONTEND

Es la parte de un sitio que interactúa directamente con los usuarios.

En otras palabras, se le puede definir como los atributos del proyecto al que el usuario accede de manera directa y con la cual va a interactuar.

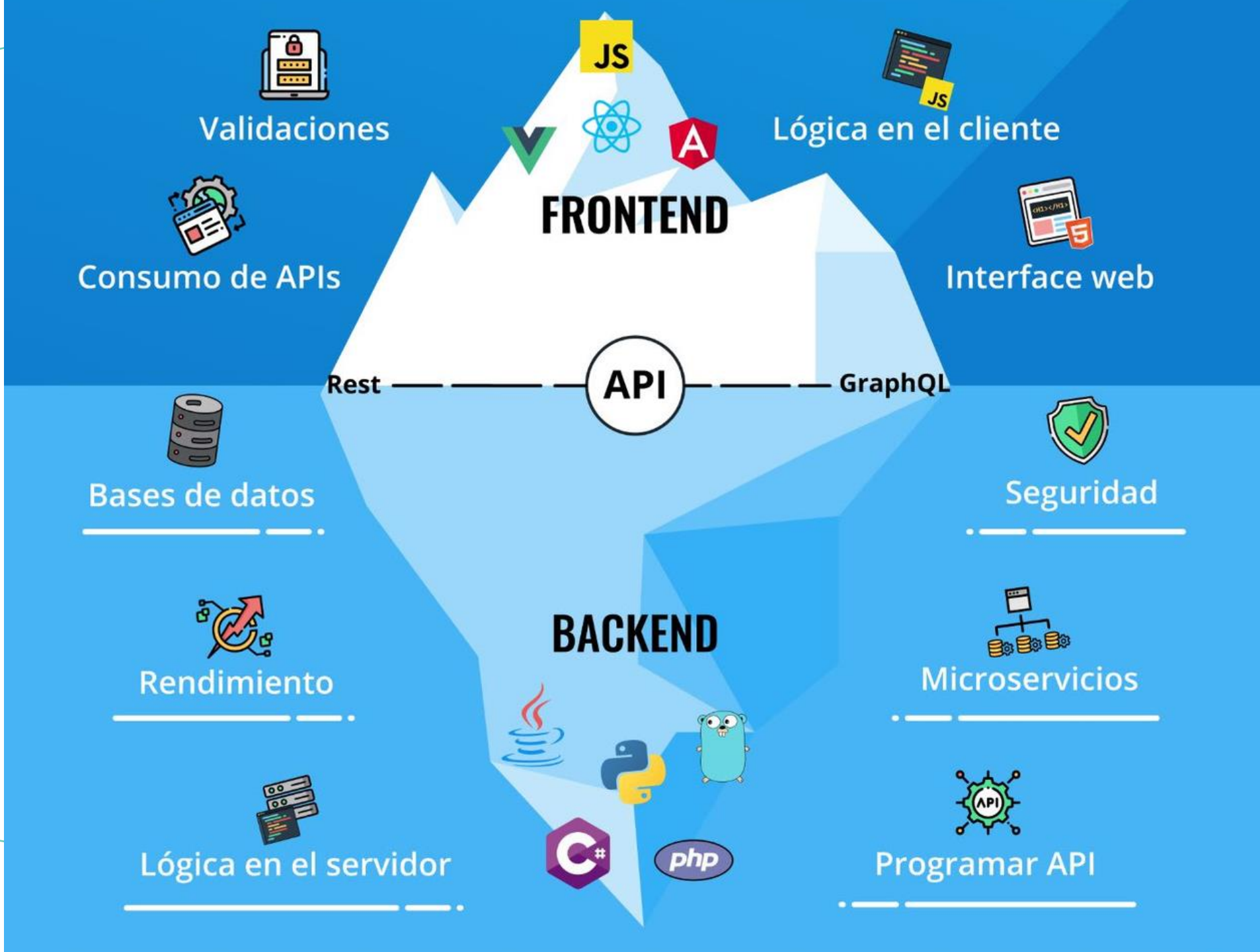
El desarrollador se encarga de crear todos los componentes visuales que conforman el sitio, dando formato a las imágenes, agregando componentes, animaciones y finalizan con la interactividad.



BACKEND

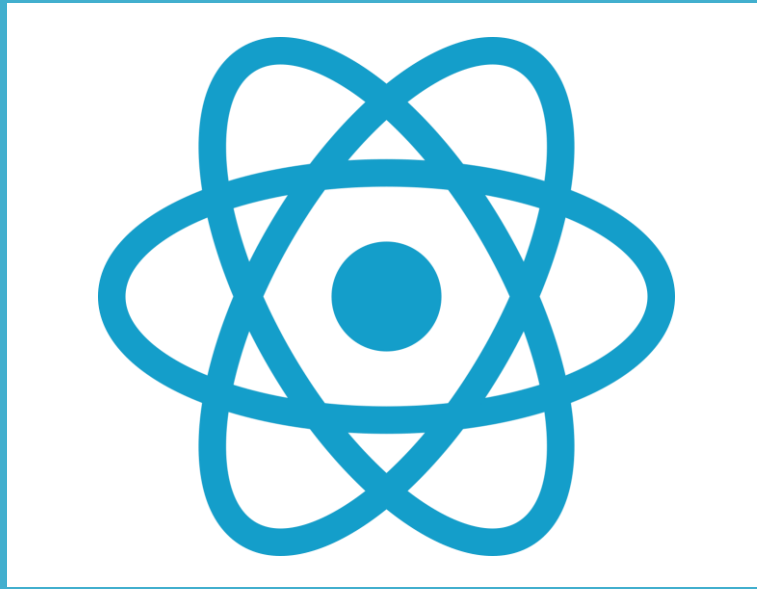
El Backend es la capa de datos a la cual el usuario no tiene acceso, siendo esta la capa que la cuál está detrás de los elementos visuales de la aplicación web, y la cual se encarga del correcto funcionamiento de los componentes.

Aquí el desarrollador está encargado de que toda la lógica de la web funcione adecuadamente y que el desempeño se adecúe a las necesidades del usuario.

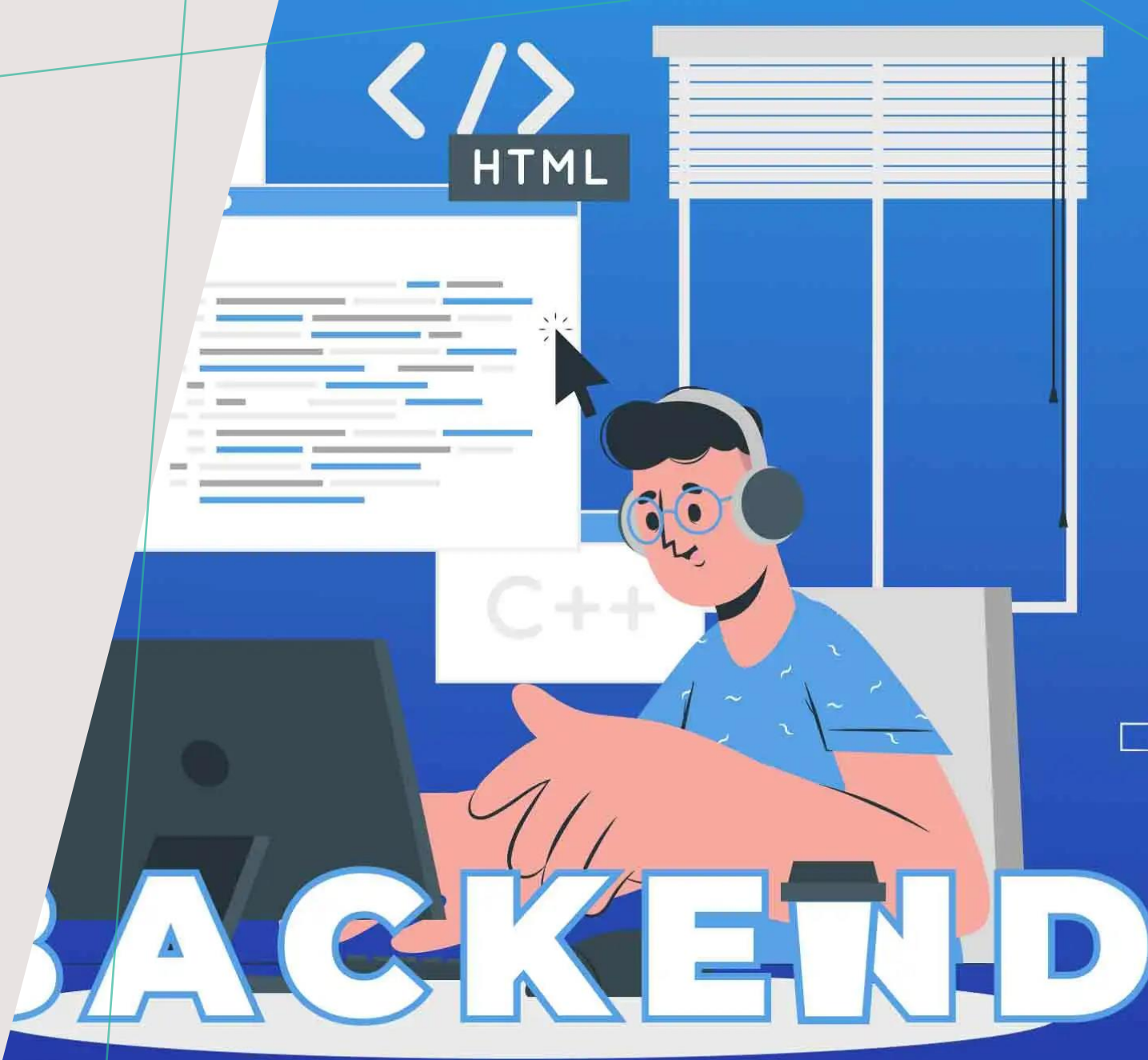




HERRAMIENTAS PARA PROGRAMACIÓN WEB



*DONDE
PARTIR*



Backend Solutions

django



firebase



nest



Flask



Back4App



RAILS

Express



JS



laravel



CakePHP

METEOR

API REST

¿QUÉ ES?



An abstract graphic on the left side of the slide. It features a light gray grid with various colored squares (black, green, blue, purple, pink, teal) and lines of different colors (gray, teal) connecting them, creating a network-like structure. The squares have a slight 3D effect with shadows.

¿QUÉ ES?

Una API REST (o API basada en REST) es una interfaz de programación que sigue los principios del diseño REST (Representational State Transfer) para permitir que diferentes sistemas o aplicaciones se comuniquen entre sí a través de HTTP. Es una de las formas más comunes de crear APIs debido a su flexibilidad y facilidad de uso.



API

API (Interfaz de programación de aplicaciones). Es un conjunto de requisiciones que permite la comunicación de datos entre aplicaciones.

Para eso, la API utiliza requisiciones HTTP responsables de las operaciones básicas necesarias para la manipulación de datos.

REST

Rest, que es la abreviación de Representational State Transfer, es un conjunto de restricciones que se utilizan para que las solicitudes HTTP cumplan con las directrices definidas en la arquitectura.

Básicamente, las restricciones determinadas por la arquitectura Rest son:

- Cliente-servidor
- Sin estado
- Caché
- Interfaz uniforme

*COMPONENTES
DE UNA API REST*

Recursos

Operaciones (HTTP)

Formato de datos

Stateless

Códigos de estado HTTP

RECURSOS

Los datos que se gestionan en la API (digamos en el caso de una plataforma de ventas: usuarios, productos, pedidos) se representan como "recursos". Cada recurso tiene una URL única que lo identifica.

- <http://localhost:5000/api/admin/usuarios>
- <http://localhost:5000/api/admin/usuario=?300>



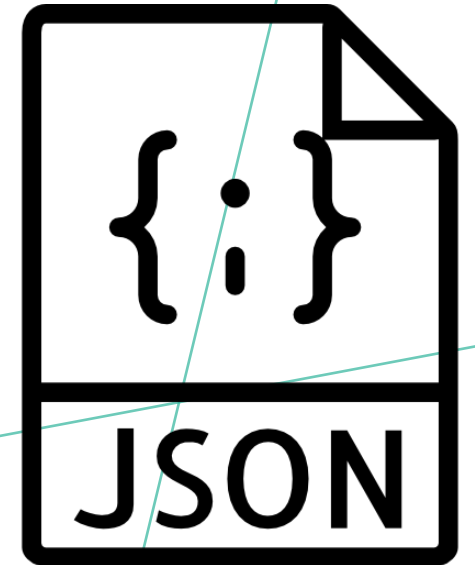
OPERACIONES (MÉTODOS HTTP)

- GET: Obtención datos (lectura)
- POST: Crear recursos
- PUT/PATCH: Actualizar recurso existente
- DELETE: Eliminar recurso

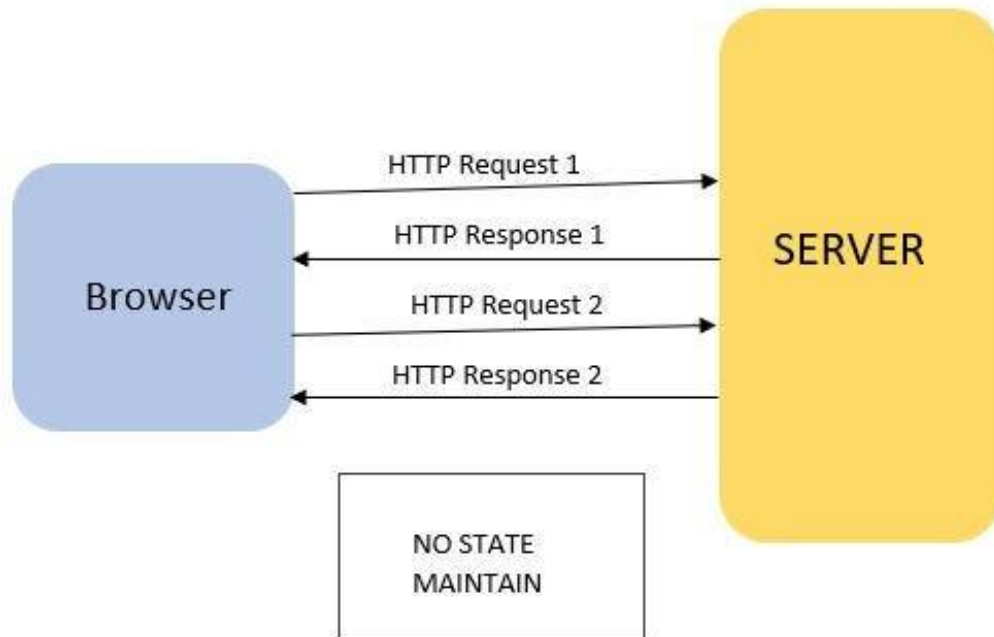
SAFE METHODS NO ACTION ON SERVER	{	GET	HTTP/1.1 MUST IMPLEMENT THIS METHOD
		HEAD	INSPECT RESOURCE HEADERS
MESSAGE WITH BODY SEND DATA TO SERVER	{	PUT	DEPOSIT DATA ON SERVER — INVERSE OF GET
		POST	SEND INPUT DATA FOR PROCESSING
		PATCH	PARTIALLY MODIFY A RESOURCE
		TRACE	ECHO BACK RECEIVED MESSAGE
		OPTIONS	SERVER CAPABILITIES
		DELETE	DELETE A RESOURCE — NOT GUARANTEED

FORMATO DE DATOS

Los datos suelen enviarse y recibirse en formato JSON o XML, aunque JSON es el estándar más popular actualmente.



STATELESS (*SIN ESTADO*)



Cada solicitud enviada al servidor debe ser independiente y contener toda la información necesaria (tokens). El servidor no guarda el estado entre solicitudes.

CÓDIGOS DE ESTADO HTTP

Indican los resultados de las solicitudes:

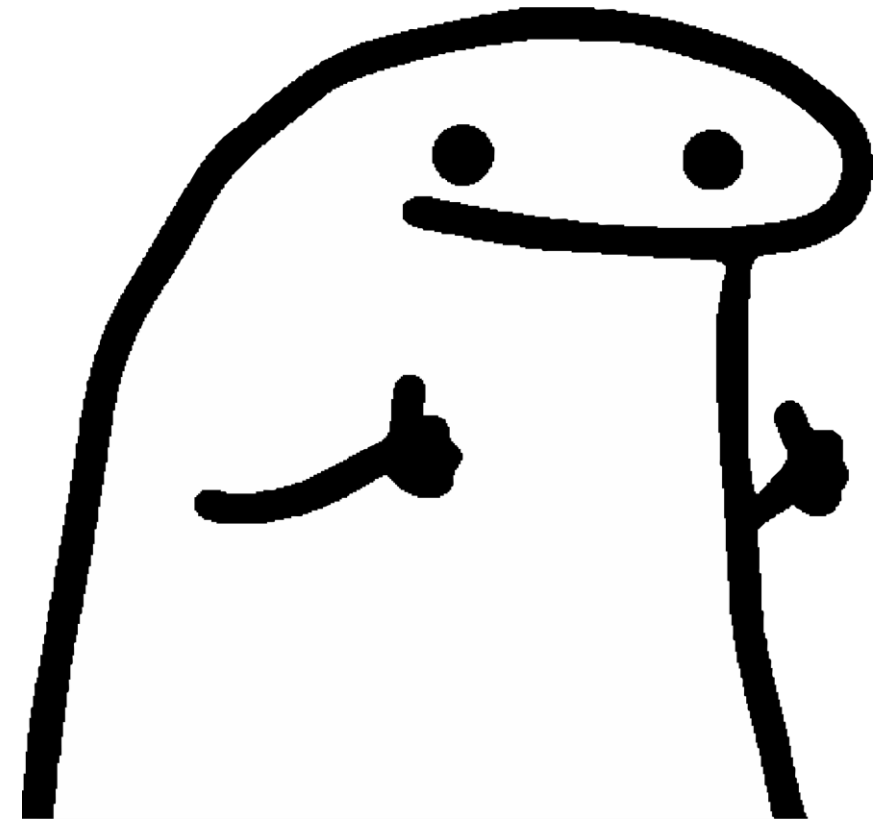
- 200 "Ok": Solicitud exitosa.
- 201 "Created": Recurso creado.
- 400 "Bad Request": Error en el envío de la información.
- 404 "Not Found": Recurso no encontrado.
- 500 "Internal Server Error": Error en el servidor.

●	1XX	Códigos informativos	El servidor ha recibido la petición y procederá con ella.
●	2XX	Códigos de éxito	El servidor ha recibido, entendido y procesado la solicitud correctamente.
●	3XX	Códigos de redirección	El servidor ha recibido la solicitud, pero hay una redirección a alguna otra parte (o, en raras ocasiones, alguna acción adicional que debe completarse).
●	4XX	Códigos de error de cliente	El servidor no puede encontrar (o alcanzar) la página o la web. Se trata de un error del lado de la web.
●	5XX	Códigos de error de servidor	El cliente ha realizado una solicitud válida, pero el servidor ha fallado al completarla.

<https://es.semrush.com/blog/codigos-de-estado-http/>

¿POR QUÉ USAR API REST?

- **Escalabilidad:** Permite manejar muchas solicitudes de forma eficiente.
- **Interoperabilidad:** Puede ser utilizada por diferentes tecnologías y lenguajes de programación.
- **Flexibilidad:** Los clientes pueden consumir los recursos según sus necesidades.
- **Simplicidad:** Utiliza HTTP, un protocolo ampliamente conocido y fácil de implementar.



ASP
NET



Java



spring®

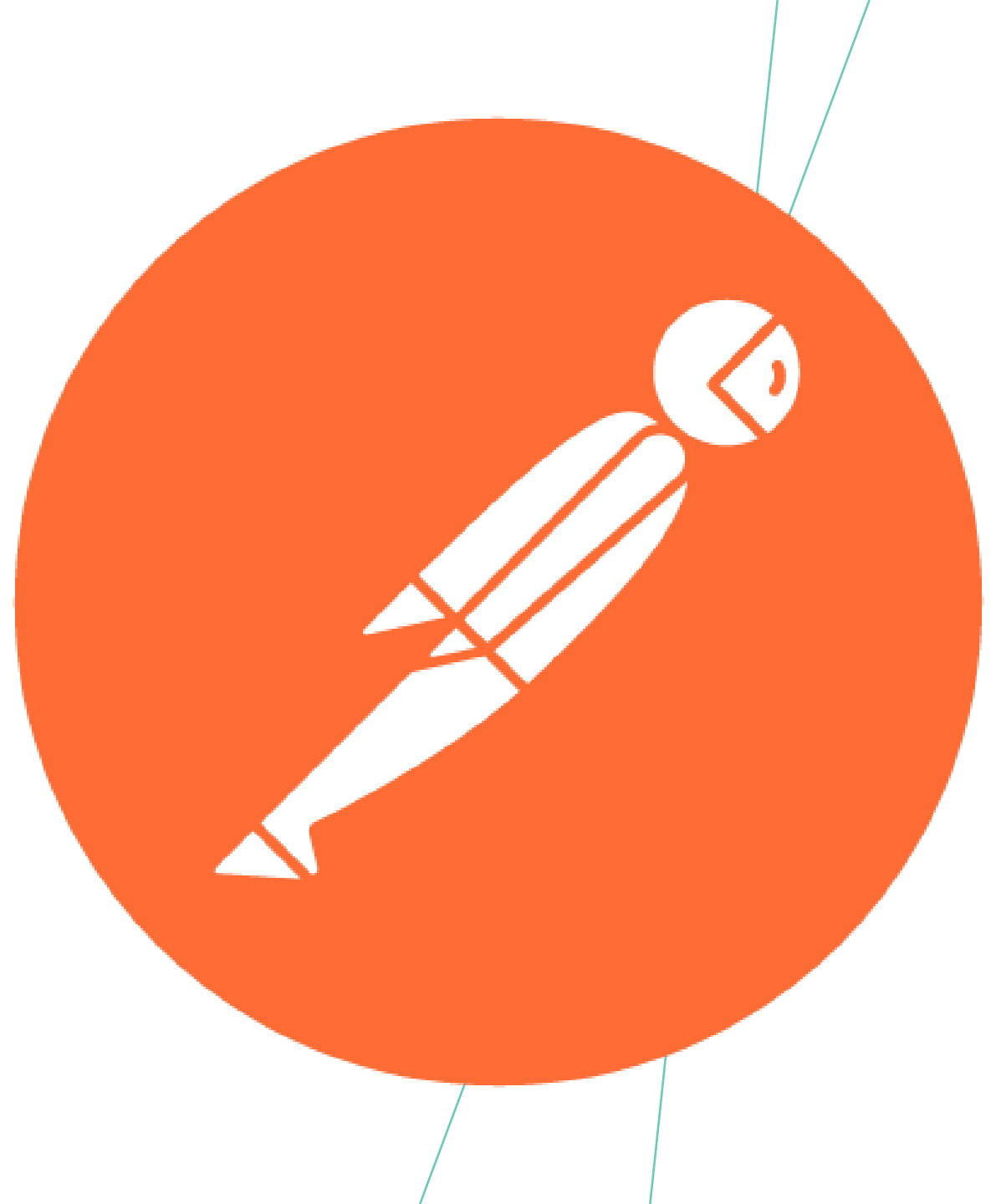


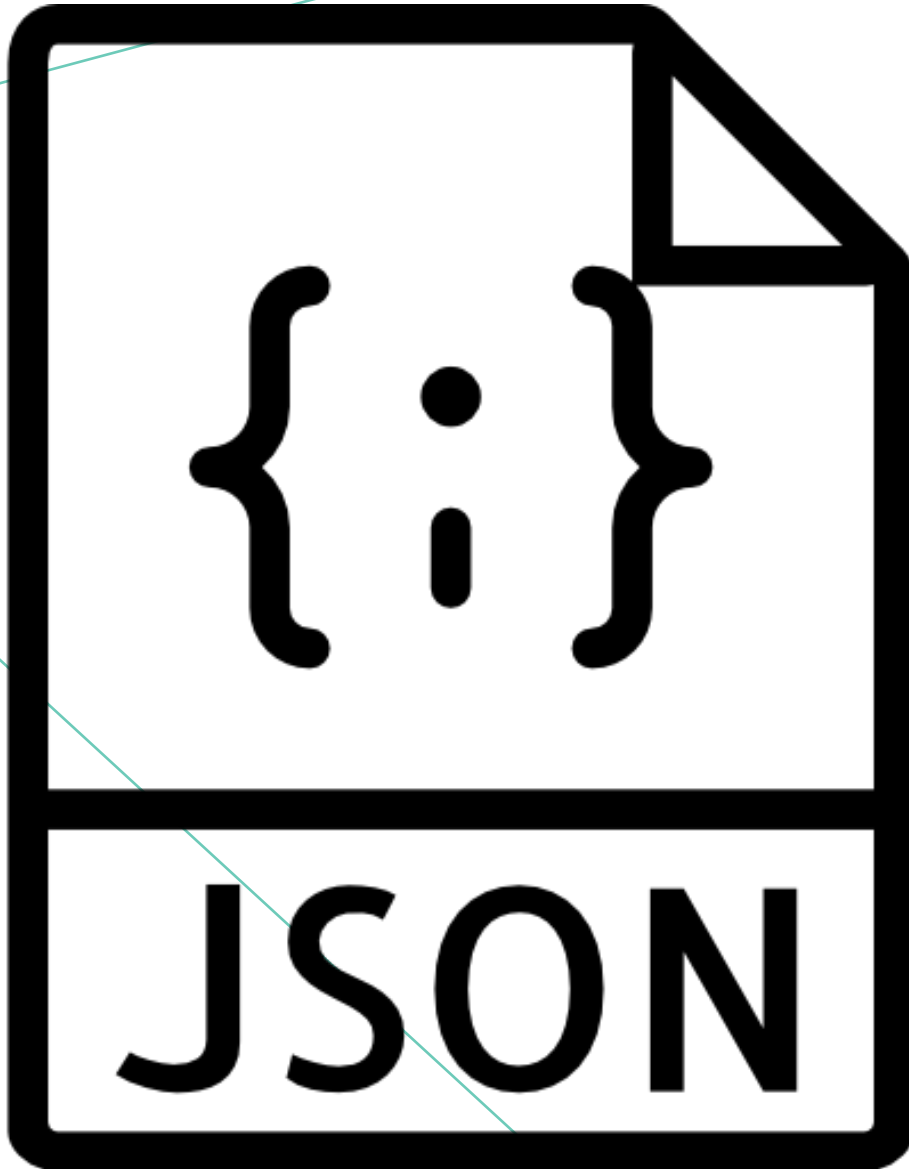
python™

HERRAMIENTAS PARA BACKEND

POSTMAN

Postman es una herramienta que se utiliza, sobre todo, para el testing de API REST, aunque también admite otras funcionalidades que se salen de lo que engloba el testing de este tipo de sistemas.



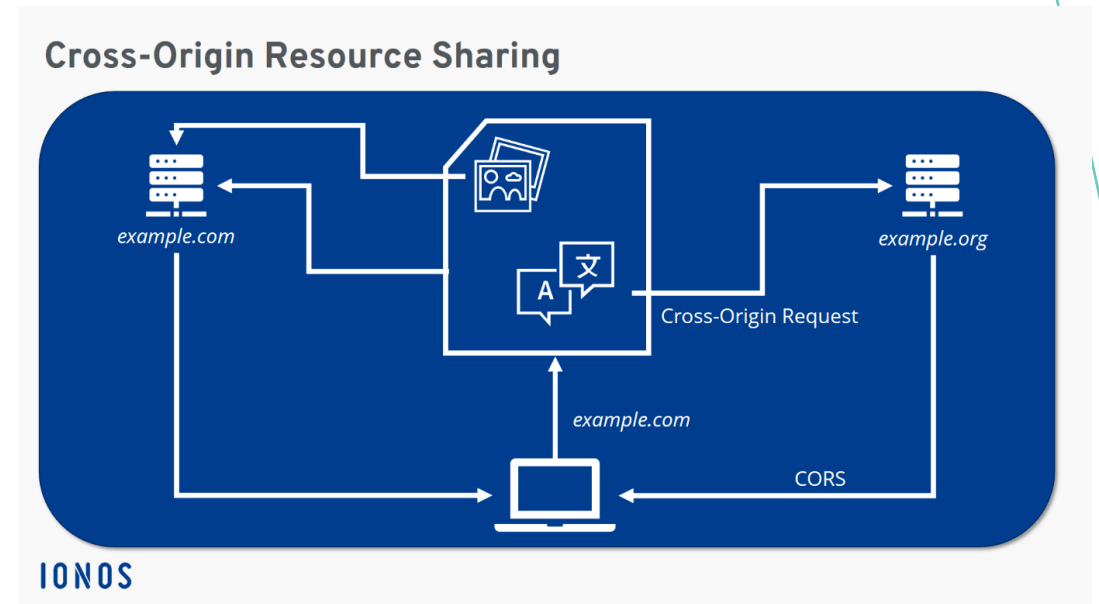


JSON

JSON (JavaScript Object Notation) es un formato de texto pensado para el intercambio de datos. Su sintaxis está basada originalmente en la sintaxis de JavaScript, pero realmente es independiente de cualquier lenguaje de programación.

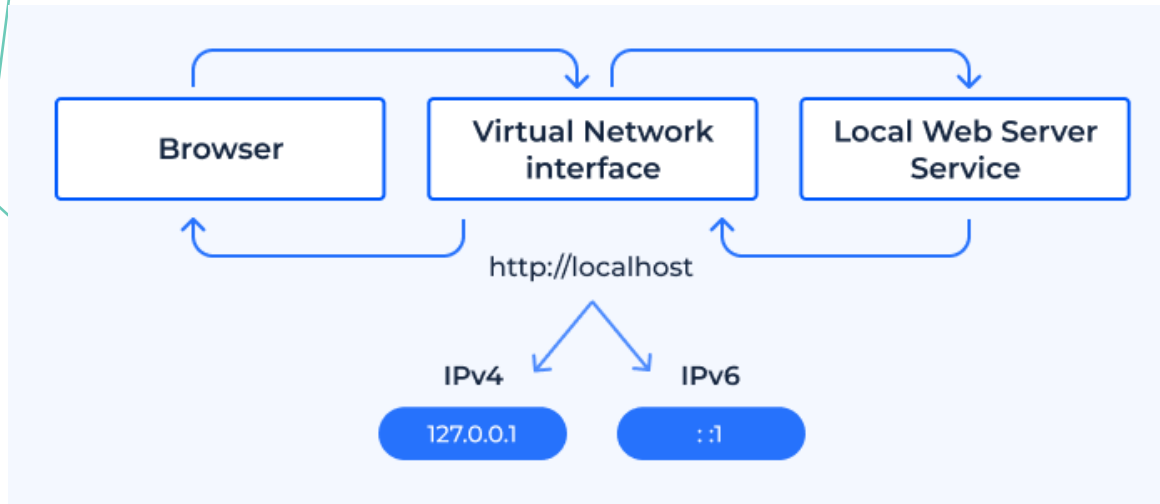
CORS

Cuando se abre una página web, cargar datos de servidores ajenos está, en teoría, estrictamente prohibido. Sin embargo, puede haber excepciones: si los administradores de ambas webs han acordado trabajar juntos, no hay por qué evitar el intercambio. En estos casos, el llamado cross-origin resource sharing (CORS) regula la colaboración.

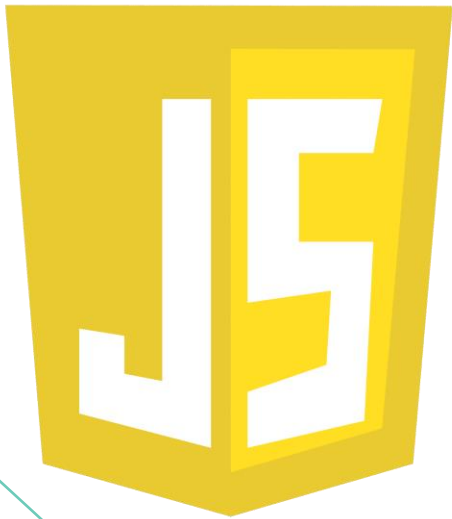


LOCALHOST

En Hardware, en el contexto de redes TCP/IP, localhost es un nombre reservado que tienen todas las computadoras, routers o dispositivos independientemente de que dispongan o no de una tarjeta de red ethernet.



JavaScript

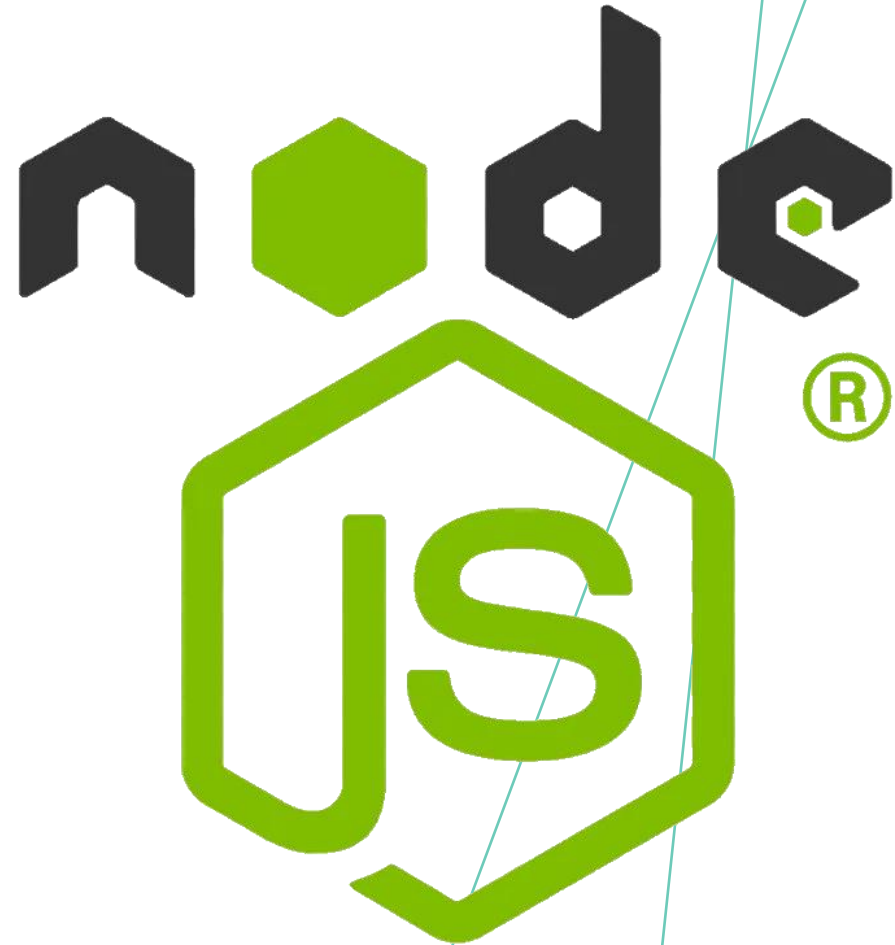


JAVASCRIPT

JavaScript (JS) es un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo (just-in-time) con funciones de primera clase. Si bien es más conocido como un lenguaje de scripting (secuencias de comandos) para páginas web, y es usado en muchos entornos fuera del navegador, tal como Node.js, Apache CouchDB y Adobe Acrobat JavaScript es un lenguaje de programación basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa (por ejemplo programación funcional).

NODEJS

Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google.



NPM

npm responde a las siglas de Node Package Manager o manejador de paquetes de node, es la herramienta por defecto de JavaScript para la tarea de compartir e instalar paquetes. Tal como reza su documentación, npm se compone de al menos dos partes principales.

- Un repositorio online para publicar paquetes de software libre para ser utilizados en proyectos Node.js
- Una herramienta para la terminal (command line utility) para interactuar con dicho repositorio que te ayuda a la instalación de utilidades, manejo de dependencias y la publicación de paquetes.





SPRING BOOT

Es un framework de Java basado en Spring Framework, diseñado para simplificar el desarrollo de aplicaciones Java modernas, especialmente aplicaciones web y servicios REST. Proporciona una configuración automática y predeterminada que permite a los desarrolladores crear aplicaciones rápidamente con una cantidad mínima de configuración manual.



Gritos de dolor

*DUDAS O
COMENTARIOS*