

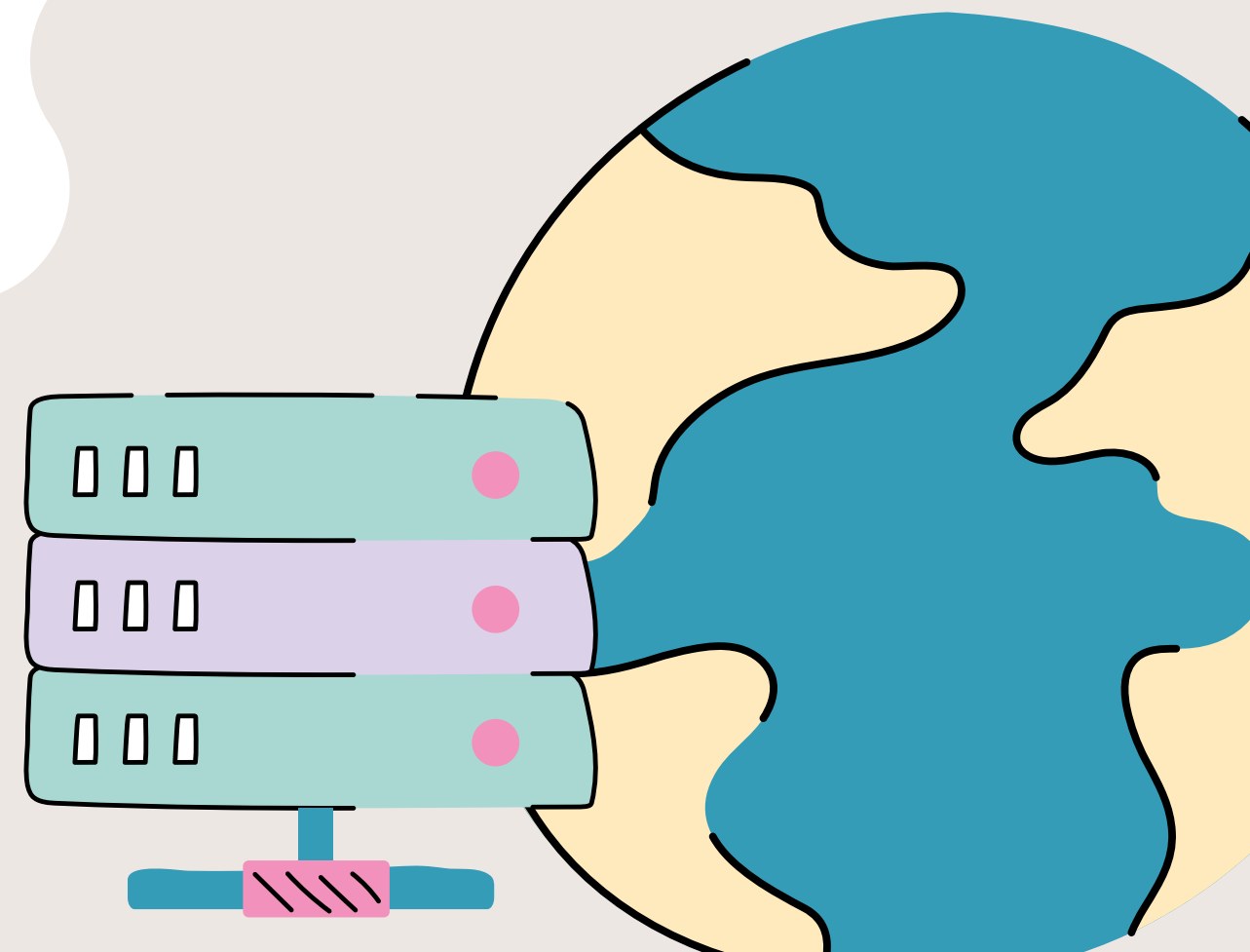


¿Qué son las APIs?

Conexión eficiente entre aplicaciones y servicios



Por Juan Duran



Introducción

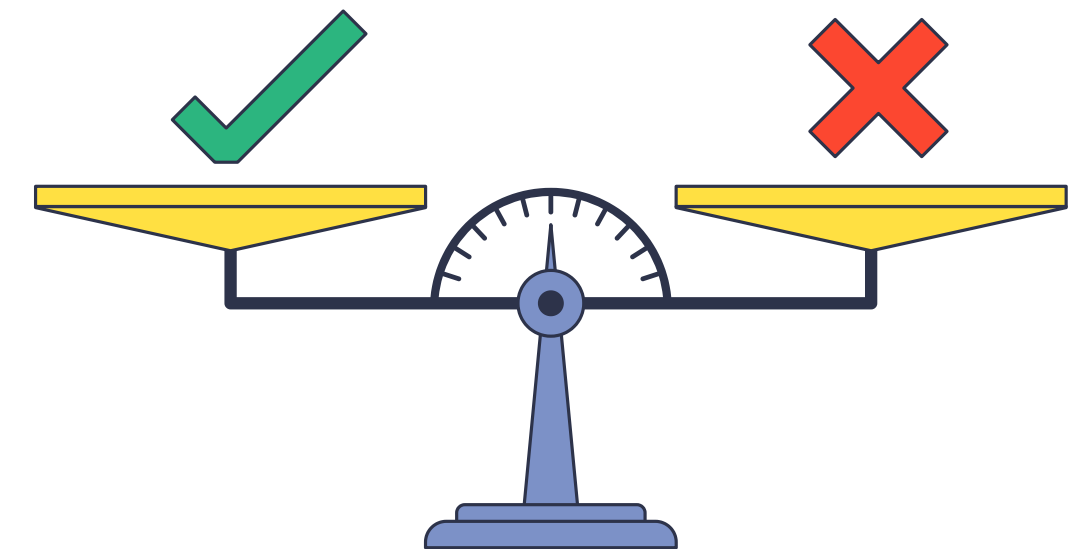
Las **APIs** (Interfaz de Programación de Aplicaciones) son un conjunto de definiciones y **protocolos** que permiten que una aplicación se **comunique** con otra. Actúan como el puente entre diferentes plataformas, servicios y dispositivos, permitiendo que **intercambien información** de manera segura y eficiente. Sin las APIs, muchas de las funcionalidades modernas que damos por sentadas, como las compras en línea, la búsqueda de información o el acceso a datos externos, no serían posibles.

Estas interfaces no solo facilitan la comunicación entre sistemas distintos, sino que también **permiten** a los **desarrolladores integrar** capacidades complejas sin tener que desarrollar todo desde cero. Las APIs son, en esencia, la columna vertebral de la **interactividad** en la web.



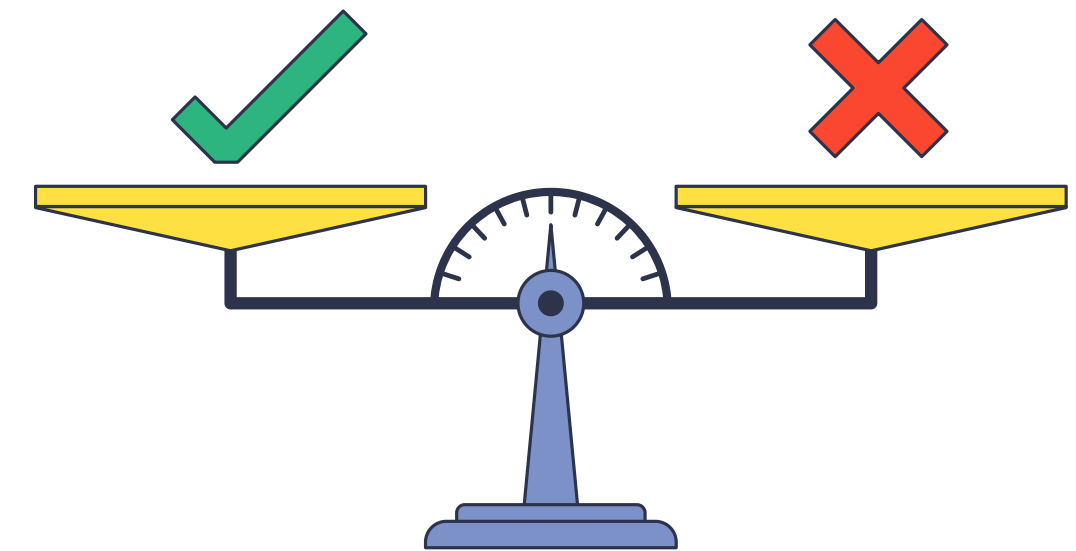
Pros

- **Facilitan la integración** entre plataformas: Las APIs permiten que sistemas desarrollados en diferentes lenguajes o plataformas interactúen entre sí. Esto significa que un desarrollador puede integrar servicios como pagos, análisis de datos o acceso a información sin tener que desarrollar esos servicios internamente.
- **Automatización y eficiencia:** Gracias a las APIs, es posible automatizar tareas complejas. Por ejemplo, obtener datos en tiempo real o realizar transacciones de manera automática. Esto reduce la carga de trabajo manual y minimiza los errores humanos.
- **Escalabilidad:** Las APIs permiten que las aplicaciones crezcan de manera modular. Si se necesita una nueva funcionalidad o servicio, basta con integrar la API correspondiente sin tener que rediseñar todo el sistema.
- **Desarrollo rápido:** Al reutilizar APIs existentes, los desarrolladores pueden agregar nuevas características a sus aplicaciones sin tener que empezar desde cero, lo que acelera el proceso de desarrollo y mejora la productividad.

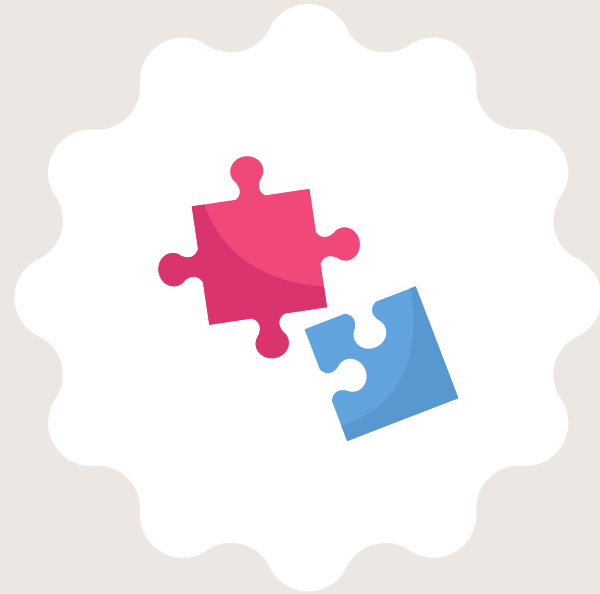


Contras

- **Dependencia de servicios externos:** Las APIs externas dependen de los servicios de terceros, lo que puede ser un riesgo si el proveedor cambia, deja de funcionar o limita el acceso. Esto puede afectar negativamente a las aplicaciones que dependen de estas APIs.
- **Vulnerabilidades de seguridad:** Las APIs pueden ser un punto de entrada para ataques si no se implementan medidas de seguridad adecuadas, como la autenticación y el cifrado de datos. Esto puede exponer datos sensibles o permitir acceso no autorizado.
- **Latencia y rendimiento:** Las llamadas a una API pueden introducir retrasos en el proceso de la aplicación. Si una API no está optimizada, esto puede afectar la experiencia del usuario, especialmente en aplicaciones que requieren respuestas en tiempo real.
- **Costos ocultos:** Algunas APIs externas pueden tener costos asociados dependiendo del número de llamadas o el uso de datos. Esto puede generar gastos imprevistos a medida que el uso de la API crece.

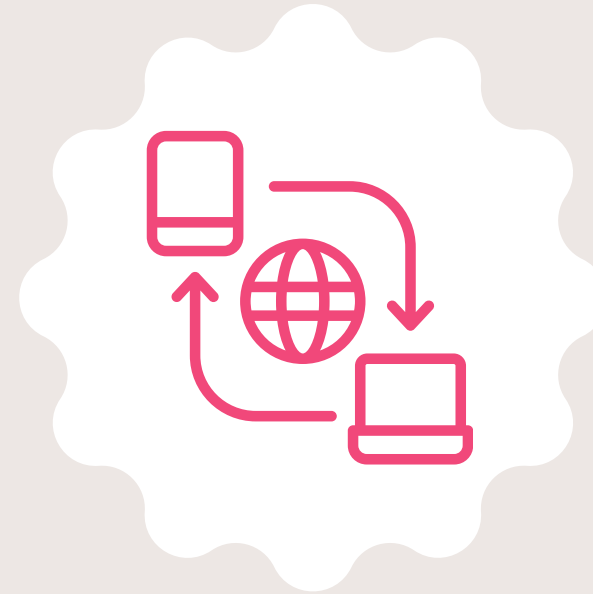


Puntos clave



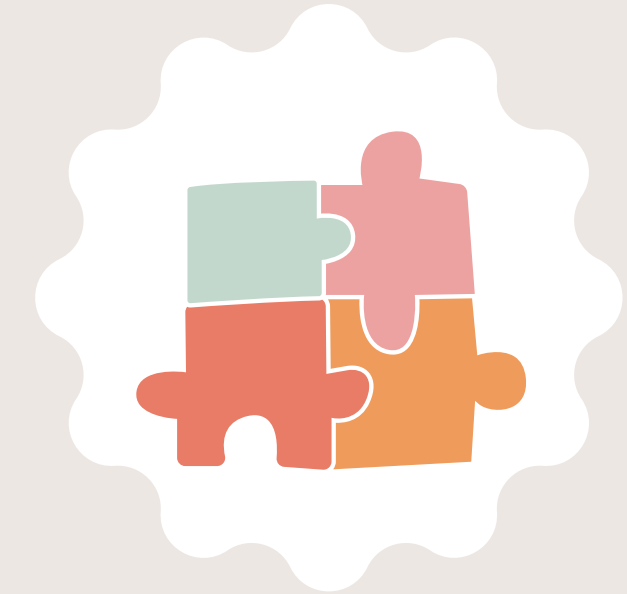
Interoperabilidad

Las APIs son la **solución** para que aplicaciones de diferentes plataformas, lenguajes de programación y dispositivos se **comuniquen** entre sí. Esta capacidad de interoperar es esencial para crear ecosistemas de aplicaciones que se **complementan** y aportan valor a los usuarios.



Acceso a servicios

Gracias a las APIs, los **desarrolladores** pueden **integrar** servicios avanzados sin tener que desarrollar esas funcionalidades desde cero. Por ejemplo, integrar un sistema de pago, obtener datos meteorológicos o realizar análisis de datos complejos puede lograrse fácilmente mediante APIs.



Modularidad

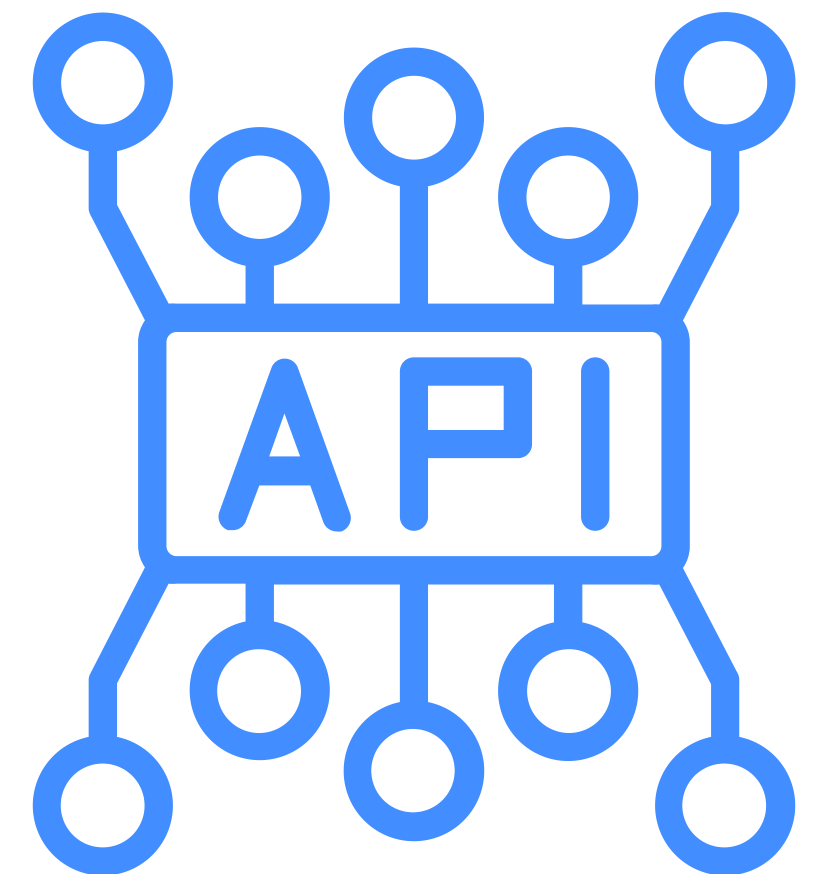
Las APIs permiten **agregar** nuevas **funciones** a las aplicaciones sin tener que reestructurarlas por completo. Esto significa que las aplicaciones pueden evolucionar con el tiempo, integrando nuevos servicios a medida que las necesidades de los usuarios cambian.

¿Cómo funcionan las APIs?

Las **APIs** permiten que las **aplicaciones** se **comuniquen** entre sí enviando solicitudes **HTTP** a través de la web. Cuando una aplicación necesita datos o realizar alguna acción externa, envía una solicitud a una API con los parámetros adecuados. La API procesa la solicitud y devuelve una respuesta, que puede estar en formato **JSON**, **XML** u otro formato estándar.

Las **solicitudes** y **respuestas** siguen un formato predefinido, lo que asegura que tanto el emisor como el receptor entiendan la información intercambiada. Este tipo de comunicación es fundamental para permitir que los sistemas sean **modulares** y **escalables**.

Imagina que quieres **obtener información** sobre el clima en tu ciudad. En lugar de almacenar esa información en tu propia base de datos, tu aplicación hace una solicitud a una API de pronóstico meteorológico. La API obtiene la información de un servicio de meteorología y la **devuelve** a tu aplicación, que la **muestra** al usuario.



Tipos de APIs

- **APIs públicas:** Son accesibles por cualquier desarrollador sin restricciones. Estas APIs suelen ser usadas por grandes plataformas como Google, Twitter o Facebook para permitir la integración de sus servicios con otras aplicaciones. Un ejemplo es la API de Google Maps, que permite integrar mapas en aplicaciones de terceros.
- **APIs privadas:** Son utilizadas internamente por empresas para permitir que sus diferentes sistemas y aplicaciones se comuniquen entre sí. Estas APIs no están disponibles al público y se limitan al uso de la organización que las crea.
- **APIs de socios:** Son utilizadas por empresas para permitir el acceso a sus sistemas a socios comerciales. Estas APIs suelen requerir autenticación o claves de acceso para garantizar que solo los usuarios autorizados puedan interactuar con ellas.
- **APIs RESTful y SOAP:** REST es un estilo arquitectónico que se basa en los métodos estándar de HTTP (GET, POST, PUT, DELETE) y es más sencillo de usar y menos pesado que SOAP, que es un protocolo más complejo y utilizado generalmente en sistemas más antiguos.

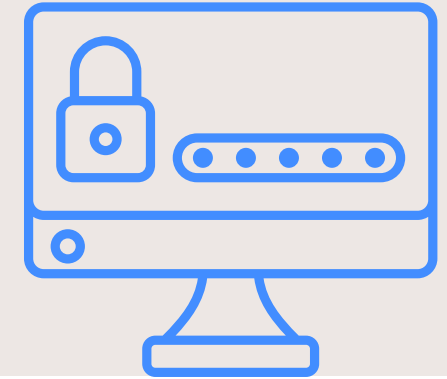


Seguridad en APIs

Debido a que las **APIs** pueden manejar datos sensibles o interactuar con sistemas importantes, la **seguridad** es un aspecto crucial. Los métodos comunes de autenticación incluyen:

- **OAuth:** Es un estándar para autorización, donde una aplicación solicita permisos al usuario para acceder a ciertos datos en su nombre, sin que el usuario tenga que compartir su contraseña. Por ejemplo, cuando inicias sesión en una aplicación usando tu cuenta de Google o Facebook, estás utilizando OAuth.
- **Claves API:** Son códigos únicos que identifican a los usuarios y permiten su autenticación para acceder a una API. Las claves API deben ser protegidas adecuadamente para evitar el acceso no autorizado.
- **Autenticación Básica:** Involucra el uso de un nombre de usuario y una contraseña para acceder a la API. Aunque es más simple, no es tan seguro como otras formas de autenticación.

Además de la autenticación, el cifrado de los datos es fundamental. Usar HTTPS en lugar de HTTP garantiza que los datos enviados entre el cliente y el servidor estén protegidos contra posibles ataques.

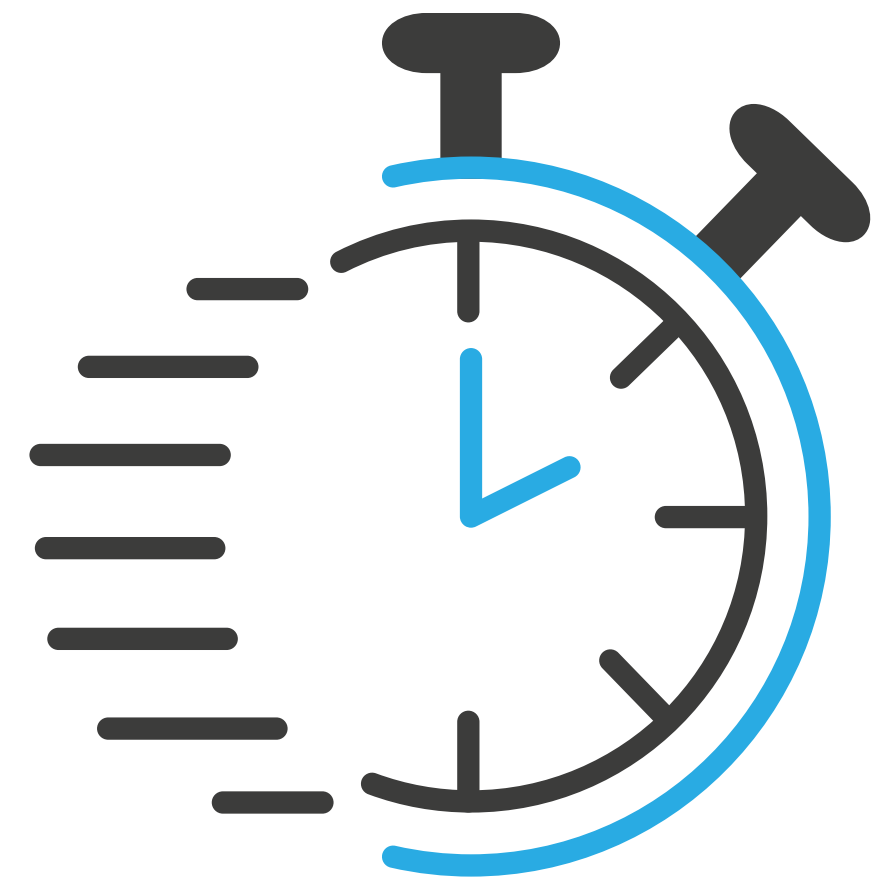


API rate limiting y buenas prácticas

El **rate limiting** es una técnica que **limita** la cantidad de **solicitudes** que un usuario o aplicación puede hacer a una **API** en un período de **tiempo** determinado. Esto evita el uso excesivo de recursos y asegura que el **servicio** no se **sobrecargue**. Las buenas prácticas de rate limiting incluyen establecer **límites** por **usuario**, por **IP** y por tipo de servicio.

Además, para asegurar que una API sea escalable y fácil de mantener, es importante:

- **Versionar** la API: Para evitar que los cambios en la API rompan la funcionalidad de las aplicaciones que la utilizan.
- **Documentación** clara y completa: La API debe tener una documentación detallada que explique cómo utilizarla, qué endpoints están disponibles y cómo manejar errores.



APIs divertidas para probar

JokeAPI: Una API que te proporciona chistes en varias categorías.

URL: <https://jokeapi.dev/>

Cat Facts: Esta API te proporciona datos curiosos sobre gatos

URL: <https://alexwohlbruck.github.io/cat-facts/>

Dog CEO's Dog API – Perros

URL: <https://dog.ceo/dog-api/>

IGDB (Internet Game Database) API – Videojuegos

URL: <https://www.igdb.com/api>

PokeAPI – Pokémon

URL: <https://pokeapi.co/>

OpenWeatherMap: API para obtener información sobre el clima en tiempo real

URL: <https://openweathermap.org/api>





Conclusiones



Integración

Con ellas, podemos **conectar** aplicaciones desarrolladas en diferentes plataformas o lenguajes de programación, lo que mejora la **interoperabilidad** y facilita la creación de **soluciones** complejas.

Modularidad

Las APIs permiten **agregar** nuevas **funciones** de manera modular, lo que facilita la **escalabilidad** y **evolución** de las aplicaciones sin necesidad de rediseñarlas completamente.

Acceso

Gracias a las APIs, podemos **integrar** fácilmente **funcionalidades** avanzadas (como pagos, análisis de datos, servicios meteorológicos) sin tener que construirlas por nosotros mismos, ahorrando tiempo y recursos.

Seguridad

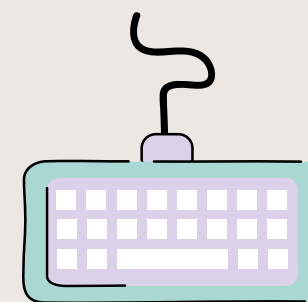
Las APIs manejan **datos valiosos**, por lo que es esencial implementar mecanismos de **autenticación** y **cifrado**, como **OAuth** y **HTTPS**, para proteger tanto a los usuarios como a los sistemas involucrados.

Agilizan el desarrollo

Al usar **APIs** ya existentes, los **desarrolladores** pueden concentrarse en lo que hace única a su aplicación, en lugar de reinventar funcionalidades comunes, lo que permite un desarrollo más **rápido** y **eficiente**.

Utilización adecuada

Al integrar **servicios externos** de calidad, las APIs pueden **enriquecer** la experiencia del usuario, ofreciendo **funcionalidades** avanzadas y personalizadas que de otro modo serían difíciles de implementar.



Gracias



Por Juan Duran

“Coding, Gaming and Leveling Up”