



Sujets > Intégration > Une API REST, qu'est-ce que c'est ?

Une API REST, qu'est-ce que c'est ?

Publié 21 février 2024 • 3 minutes (temps de lecture)

[Copier l'URL](#)

RACCOURCIS

Définition d'une API REST

Définition d'une API

L'architecture REST

Nos solutions

Définition d'une API REST

Une API REST est une interface de programmation d'application (API) qui respecte les principes de conception du style d'architecture REST, abréviation de « REpresentational State Transfer ». Ce style regroupe un ensemble de règles et de consignes pour la création d'API web.

Le manuel d'utilisation des API : 7 meilleures pratiques des équipes qui réussissent à exploiter les API

Définition d'une API

Une API est un ensemble de définitions et de protocoles qui facilite la création et l'intégration de logiciels d'applications. On compare parfois les API à un contrat entre un utilisateur

d'informations et un fournisseur d'informations, qui permet de définir le contenu demandé par le consommateur (l'appel) et le contenu demandé au producteur (la réponse). Par exemple, l'API d'un service de météo peut demander à l'utilisateur de fournir un code postal et au producteur de renvoyer une réponse en deux parties : la première concernant la température maximale et la seconde concernant la température minimale.

En d'autres termes, lorsqu'un utilisateur interagit avec un ordinateur ou un système pour obtenir des informations ou exécuter une fonction, l'API va l'aider à expliquer ce qu'il souhaite au système afin que ce dernier puisse comprendre et répondre à sa demande.

Les API servent de médiateur entre les utilisateurs ou les clients et les ressources ou services web qu'ils souhaitent obtenir. Ce sont aussi des outils qui permettent à une entreprise de partager des ressources et des informations de manière sécurisée et contrôlée, en déterminant qui peut accéder à quoi.

Autre avantage des API : il n'est pas nécessaire de connaître les détails de la mise en cache, c'est-à-dire comment les ressources sont récupérées ou d'où elles proviennent.

En savoir plus sur les API →

Livre numérique : Service Mesh ou gestion des API ? →

Ressources Red Hat

En savoir plus →

L'architecture REST

Le style d'architecture REST regroupe des contraintes de conception d'API. Il ne s'agit pas d'un protocole ni d'une norme. Lors du développement d'API, les principes REST peuvent être mis en œuvre de différentes manières.

Lorsqu'un client envoie une demande via une API RESTful, celle-ci transfère une représentation de l'état de la ressource au demandeur ou point de terminaison. Ces informations, ou représentations, sont fournies dans l'un des formats suivants via HTTP : JSON (JavaScript Object Notation), HTML, XML, Python, PHP ou texte brut. Le format de fichier le

plus communément utilisé est JSON, car, contrairement à ce que son nom peut laisser penser, il ne dépend pas d'un langage et peut être lu aussi bien par les humains que par les machines.

Autre élément à prendre en compte : les en-têtes et les paramètres utilisés dans les méthodes HTTP d'une API RESTful contiennent des informations d'identification essentielles relatives notamment aux métadonnées, à l'autorisation, à l'identifiant de ressource uniforme (URI), à la mise en cache et aux cookies. Il existe des en-têtes de demande et de réponse, qui contiennent chacun leurs propres informations de connexion HTTP et codes d'état.

Une API RESTful doit remplir les critères suivants :

- Architecture client-serveur constituée de clients, de serveurs et de ressources, qui gère des demandes via HTTP
- Communications stateless entre le client et le serveur, c'est-à-dire que les informations du client ne sont jamais stockées entre les requêtes GET qui doivent être traitées séparément et de manière totalement indépendante
- Données qui peuvent être mises en cache afin de rationaliser les interactions entre le client et le serveur
- Interface uniforme entre les composants qui permet de transférer les informations de manière standardisée, ce qui implique que :
 - Les ressources interrogées doivent être identifiables et séparées des représentations envoyées au client.
 - Les ressources doivent pouvoir être manipulées par le client au moyen de la représentation reçue, laquelle doit contenir suffisamment d'informations.
 - Les messages autodescriptifs renvoyés au client doivent contenir assez de détails pour décrire la manière dont il doit traiter les informations.
 - L'API doit inclure un hypertexte/hypermédia qui permet au client d'utiliser des hyperliens pour connaître toutes les autres actions disponibles après avoir accédé à une ressource.
- Système à couches, invisible pour le client, qui permet de hiérarchiser les différents types de serveurs (pour la sécurité, l'équilibrage de charge, etc.) impliqués dans la récupération des informations demandées
- Code à la demande (facultatif), permettant à l'API d'envoyer du code exécutable depuis le serveur vers le client (en cas de demande), afin d'étendre les fonctionnalités côté client

Même si elles doivent répondre à ces nombreux critères, les API REST sont considérées comme étant plus simples à utiliser qu'un protocole tel que SOAP (Simple Object Access Protocol), qui est soumis à des contraintes spécifiques (comme la messagerie XML, la sécurité intégrée et la conformité des transactions) le rendant plus lourd et moins rapide.

Puisque les principes REST peuvent être mis en œuvre en fonction des besoins, les API REST sont plus rapides et légères, et offrent davantage d'évolutivité. Elles sont donc idéales pour l'IoT (Internet des objets) et le développement d'applications mobiles.