

APPLICATION
PROGRAMMING
INTERFACE

API

Sommaire

- ▶ -Définition
- ▶ -Exemple
- ▶ -Utilités et inconvénients
- ▶ -API REST
- ▶ -API Gateway

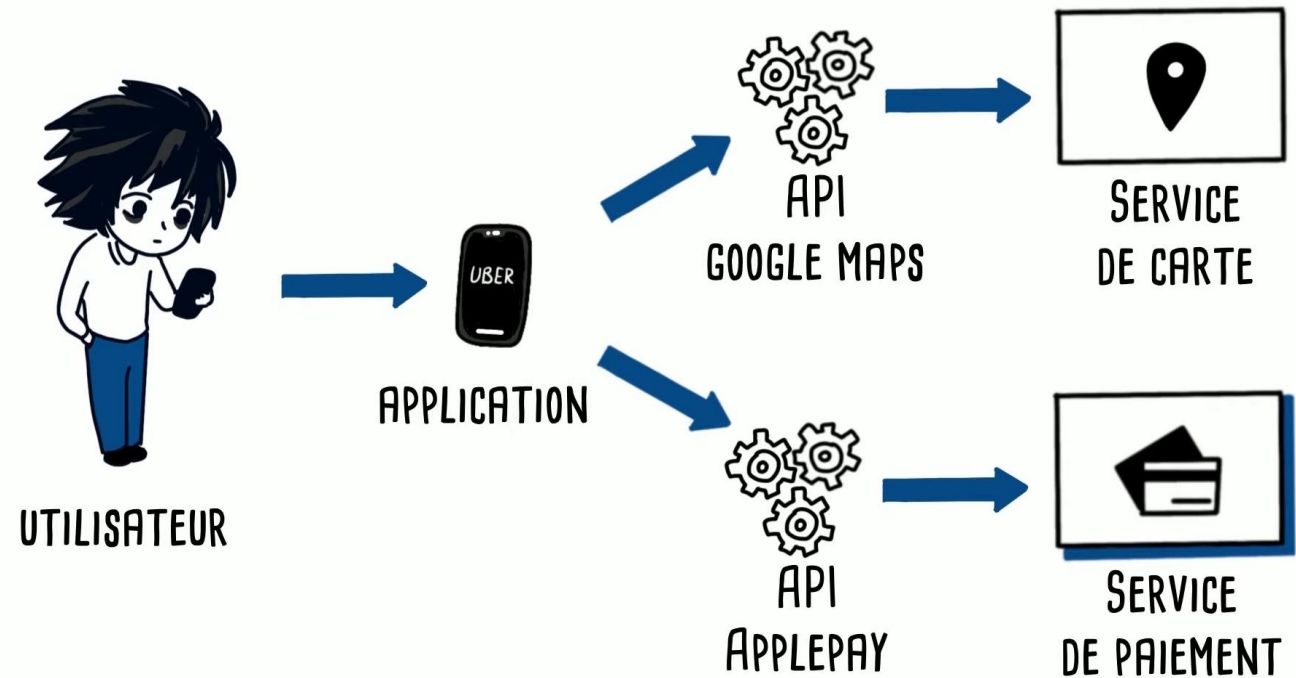
Définition

- ▶ Une API est une interface de programmation permettant d'accéder à un service fourni par un système tiers
- ▶ Le système tiers «expose » l'API
- ▶ Une API permet de faire dialoguer des applications, entre des applications consommatrices d'un service et une autre application qui est productrice de ce service
- ▶ Le programmeur ne connaît pas les détails du fonctionnement du logiciel tiers, il ne connaît que les contraintes d'accès défini par le système tier

Exemple

-L'itinéraire est calculé par l'API
google maps

-La course est payé en passant par
L'API ApplePay



Utilités et inconvénients

Utilités:

- ▶ Simplifie énormément le développement d'application en permettant l'accès à des services déjà développés par d'autres entreprises via leur API
- ▶ Economise donc du temps et de l'argent de développement

Inconvénients:

- ▶ Une documentation précise (données accessibles, conditions d'usage, les garanties de disponibilité et de performances...)
- ▶ Services payants (la plupart du temps)

API REST

- ▶ Une API respectant les standards du web
- ▶ REST est une architecture logicielle basé sur le HTTP
- ▶ Les échanges sont effectués à partir de requête HTTP
- ▶ Choisir l'architecture REST est une garantie d'une meilleure intégration aux communications du web

API Gateway

- ▶ API Gateway est un dispositif qui joue le rôle de passerelle pour permettre au client externes d'utiliser des Api internes
- ▶ Le client n'as pas à connaitre l'emplacement de toute les instance du service utilisé
- ▶ Orchestre les services (si une requête requiert des appels à plusieurs Api le Gateway s'en occupe)
- ▶ Transforme les requêtes pour que qu'elle soit valide envers les API aux niveau des données et des protocoles

API Gateway

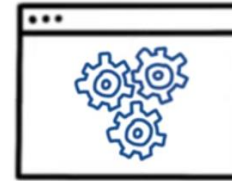
- ▶ Authentifie les requêtes et sécurise les accès aux API internes
- ▶ Permet le WAF, autorise ou bloque des requêtes en fonctions des règles de sécurité
- ▶ Surveille le fonctionnement des API backend
- ▶ Gère la monétisation des API

API Gateway

- ▶ Ne pas avoir à redévelopper la partie authentification pour chaque service
- ▶ Permet de gérer, publier et sécuriser facilement ses API
- ▶ Fournit une expérience Simple et optimale aux clients



MEILLEURE
PRODUCTIVITÉ



GESTION CENTRALISÉE
DES API



EXPÉRIENCE CLIENT
OPTIMISÉE