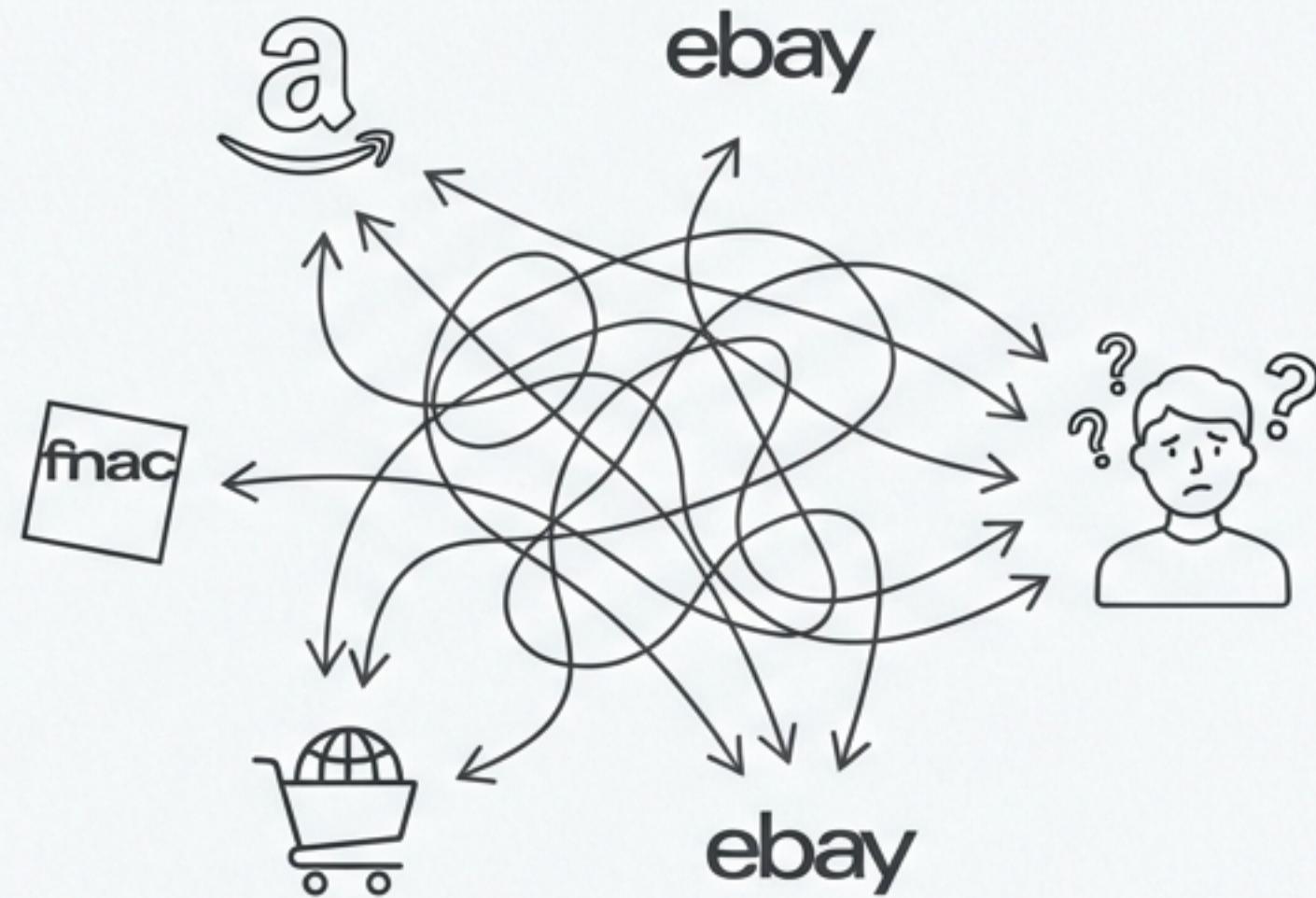


Plateforme E-Commerce Centralisée

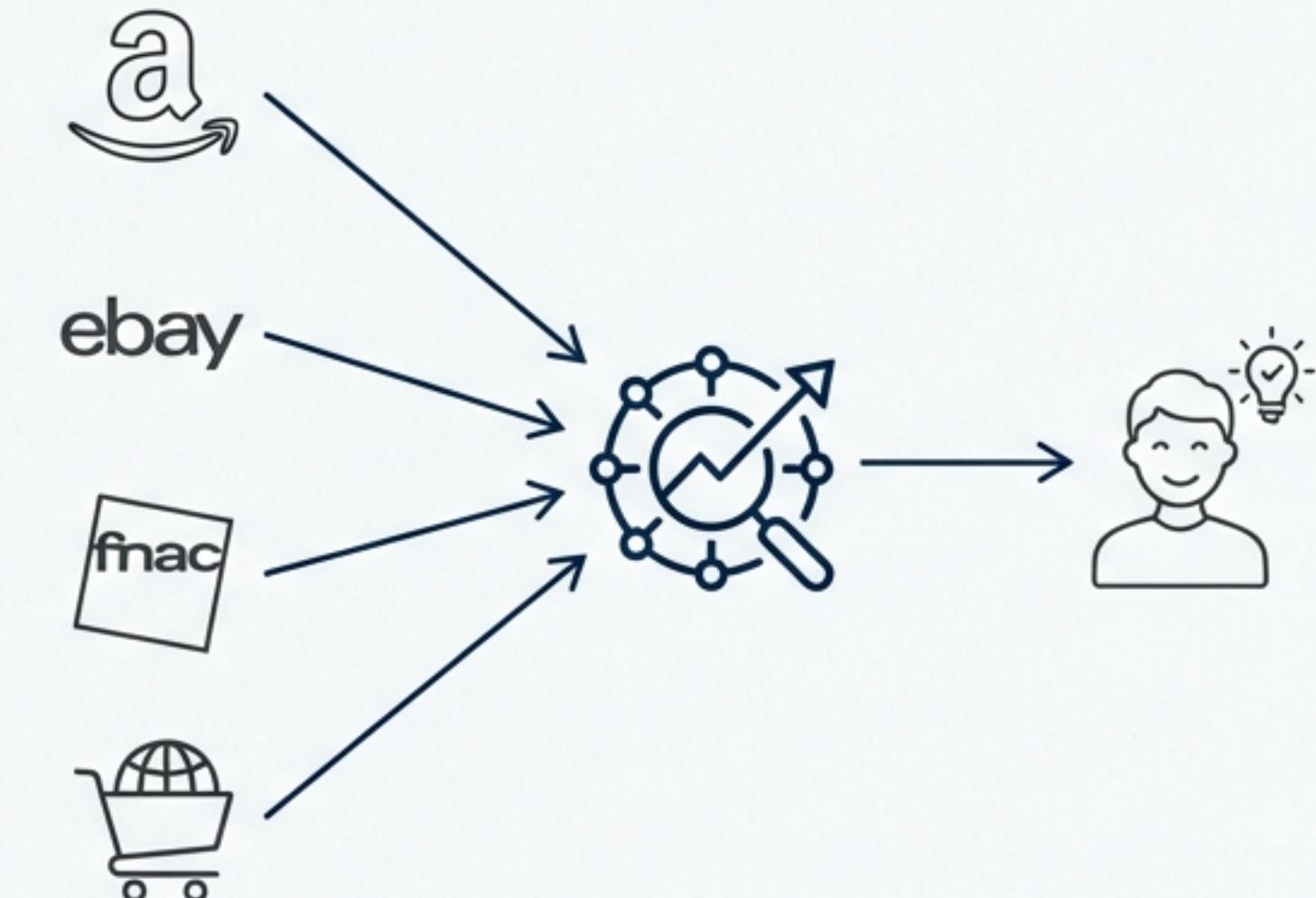
Une architecture microservices pour une expérience d'achat
unifiée et intelligente.

Projet Académique
Période : Octobre - Décembre 2025

Le Concept Clé : De la Fragmentation à l'Unification



Multiples sites, recherches répétées,
comparaison difficile.



Une seule recherche, des résultats agrégés,
une décision éclairée.

Notre plateforme permet aux utilisateurs de rechercher et comparer des produits de multiples sources (catalogue interne et sites externes) via une interface unique. La monétisation s'effectue par affiliation sur les produits externes.

Les Objectifs Stratégiques du Projet

Objectifs Fonctionnels



- Consulter un catalogue de produits interne complet.



- Agréger en temps réel des produits de sources multiples.



- Offrir une comparaison unifiée des prix et caractéristiques.



- Générer des revenus via des liens d'affiliation.

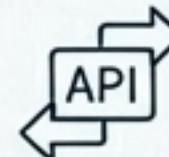


- Fournir des recommandations de produits via un service dédié.

Objectifs Techniques



- Mettre en œuvre une architecture microservices distribuée et résiliente.



- Intégrer une API externe (Amazon) pour l'agrégation de produits.



- Sécuriser les transactions via l'intégration de l'API PayPal.

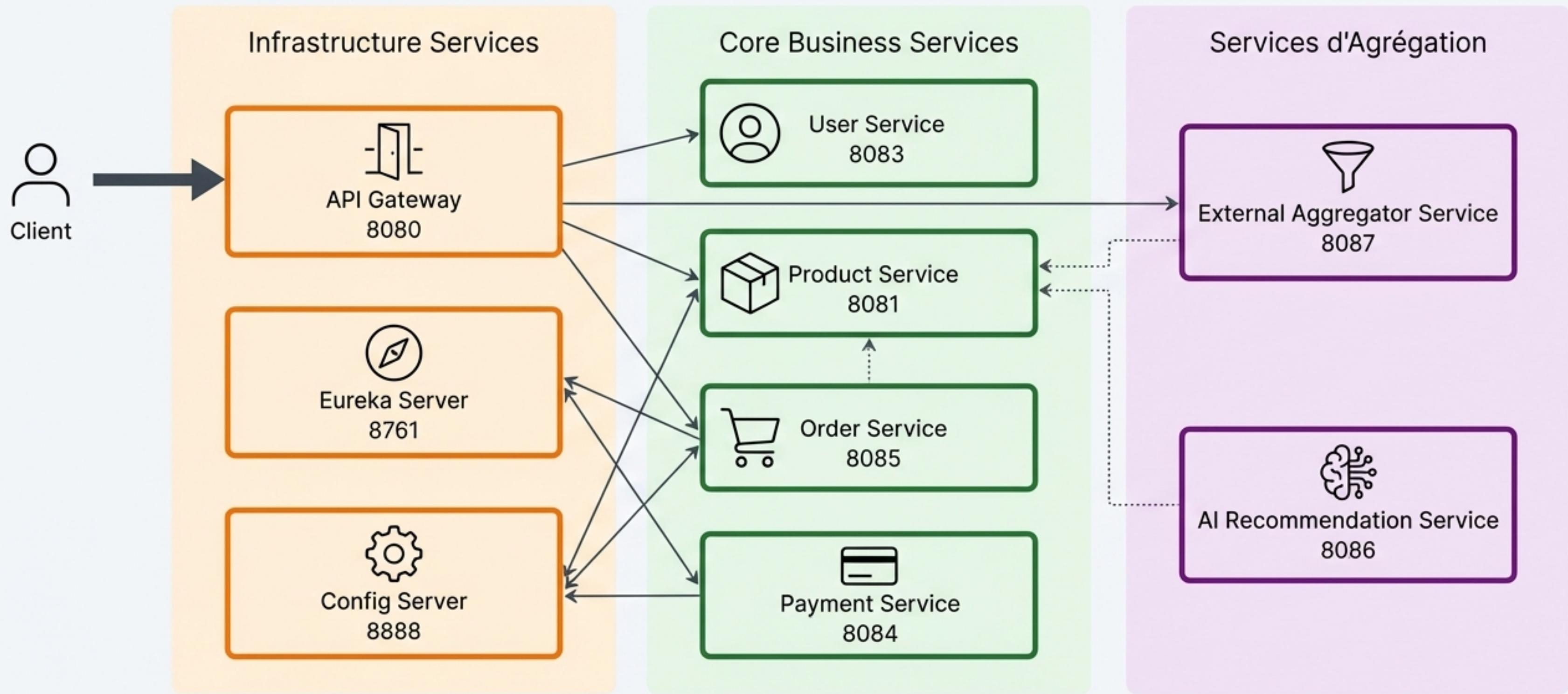


- Assurer la résilience et la gestion des erreurs avec un 'Circuit Breaker'.



- Centraliser la configuration et la découverte des services.

L'Architecture Globale en Microservices



Zoom : Les Piliers de l'Infrastructure et le Cœur Métier

Les Piliers de l'Infrastructure



API Gateway (`8080`)

Point d'entrée unique, routage des requêtes et load balancing.



Eureka Server (`8761`)

Registre central pour la découverte dynamique des services.



Config Server (`8888`)

Gestion centralisée des configurations pour tous les microservices.

Les Services du Cœur Métier



User Service (`8083`)

Gère les comptes utilisateurs (CLIENT, ADMIN) et l'authentification.



Product Service (`8081`)

Assure le CRUD et la gestion des stocks pour le catalogue interne.



Order Service (`8085`)

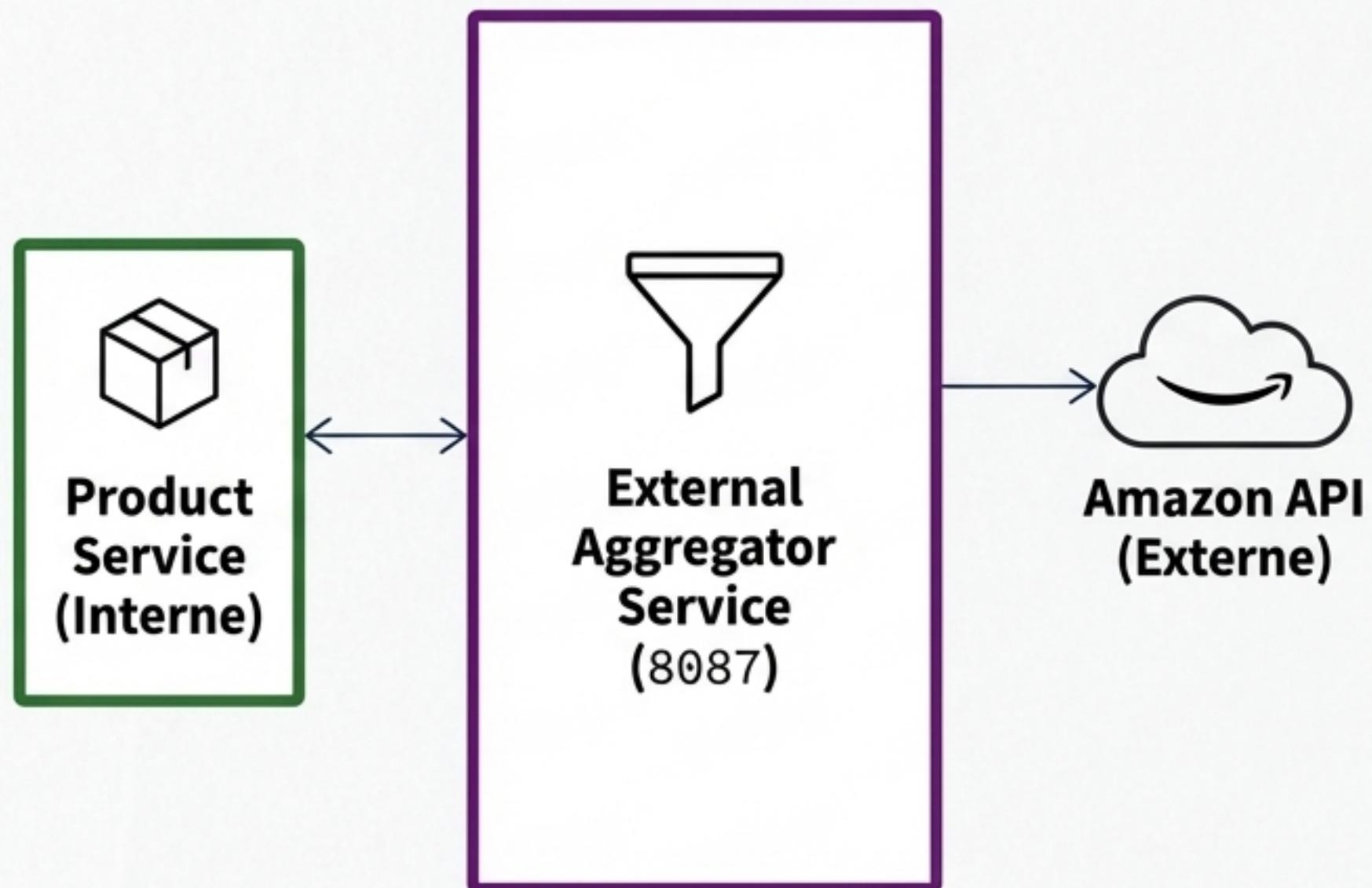
Orchestre la création et le suivi des commandes internes.



Payment Service (`8084`)

Intègre l'API PayPal pour traiter les paiements sécurisés.

Le Moteur de l'Innovation : Agrégation et Recommandation



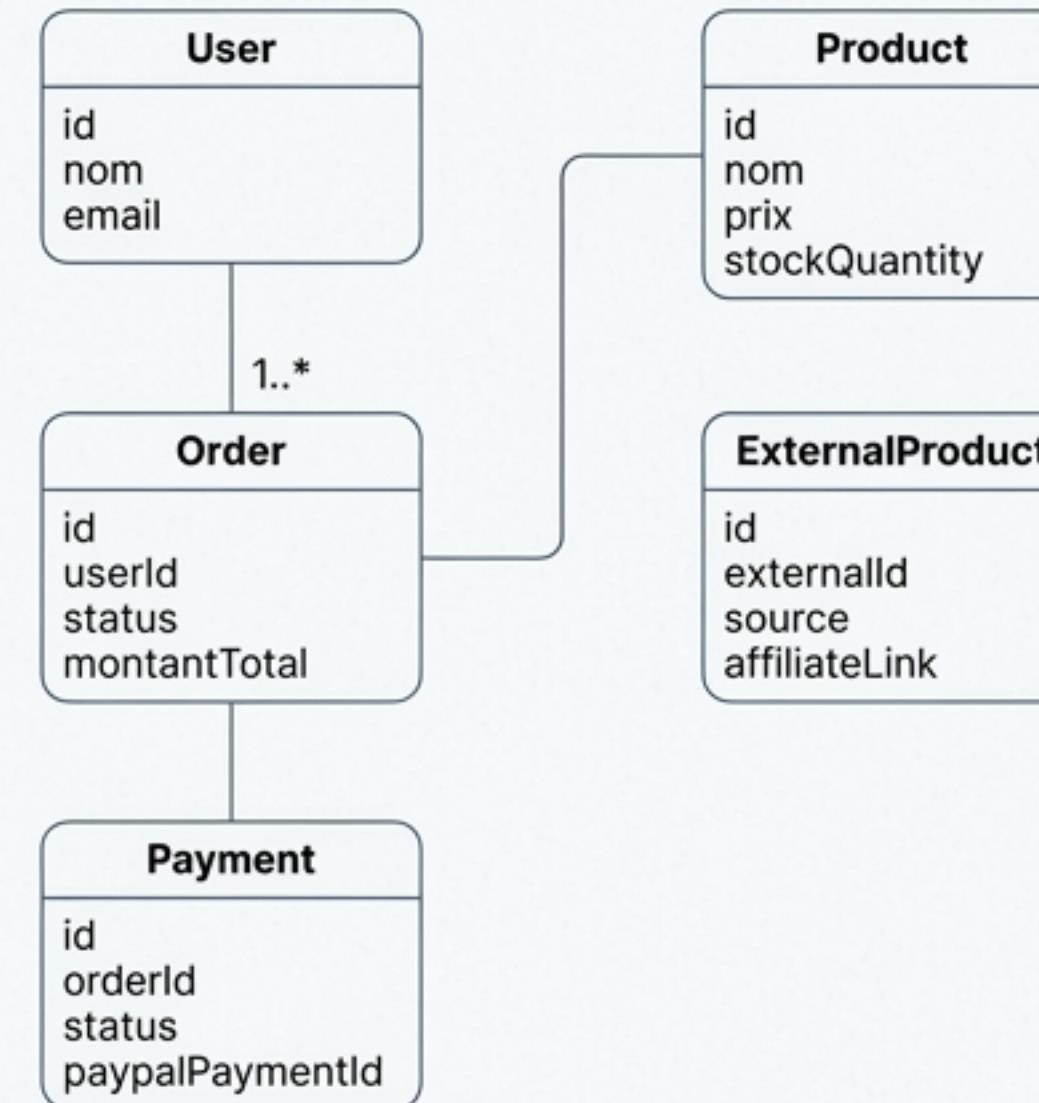
External Aggregator Service ('8087')

- Intègre l'API Amazon pour récupérer les produits externes en temps réel.
- Normalise les données de sources hétérogènes pour une présentation cohérente.
- Génère automatiquement les liens d'affiliation pour la monétisation.

AI Recommendation Service ('8086') - Version Simplifiée

- Analyse les requêtes utilisateurs via des mots-clés.
- Orchestre les appels au 'Product Service' et à l'External Aggregator'.
- Agrège et effectue un ranking simple des résultats pour les présenter à l'utilisateur.

Modèle de Données et Stratégie de Persistance



Principe fondamental

Une base de données H2 dédiée par microservice pour un découplage maximal.

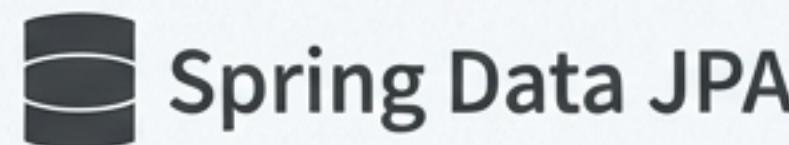
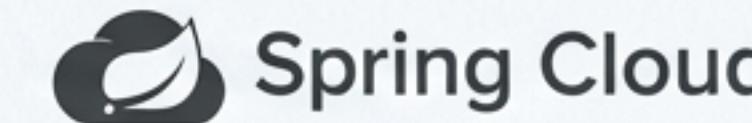
Bases de données déployées

`user_db`, `product_db`, `order_db`, `payment_db`, `aggregator_db`

L'Écosystème Technologique



Backend & Core



Communication & Résilience



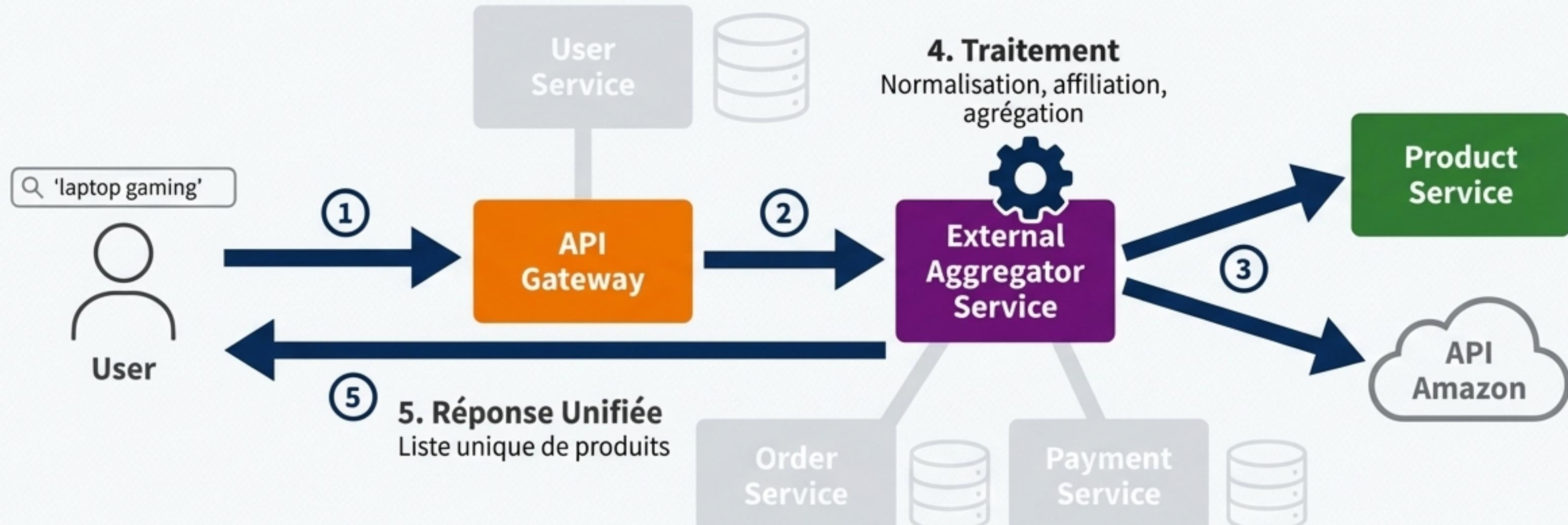
APIs Externes



Outils & Monitoring



Anatomie d'une Recherche : Le Parcours de la Requête 'laptop gaming'

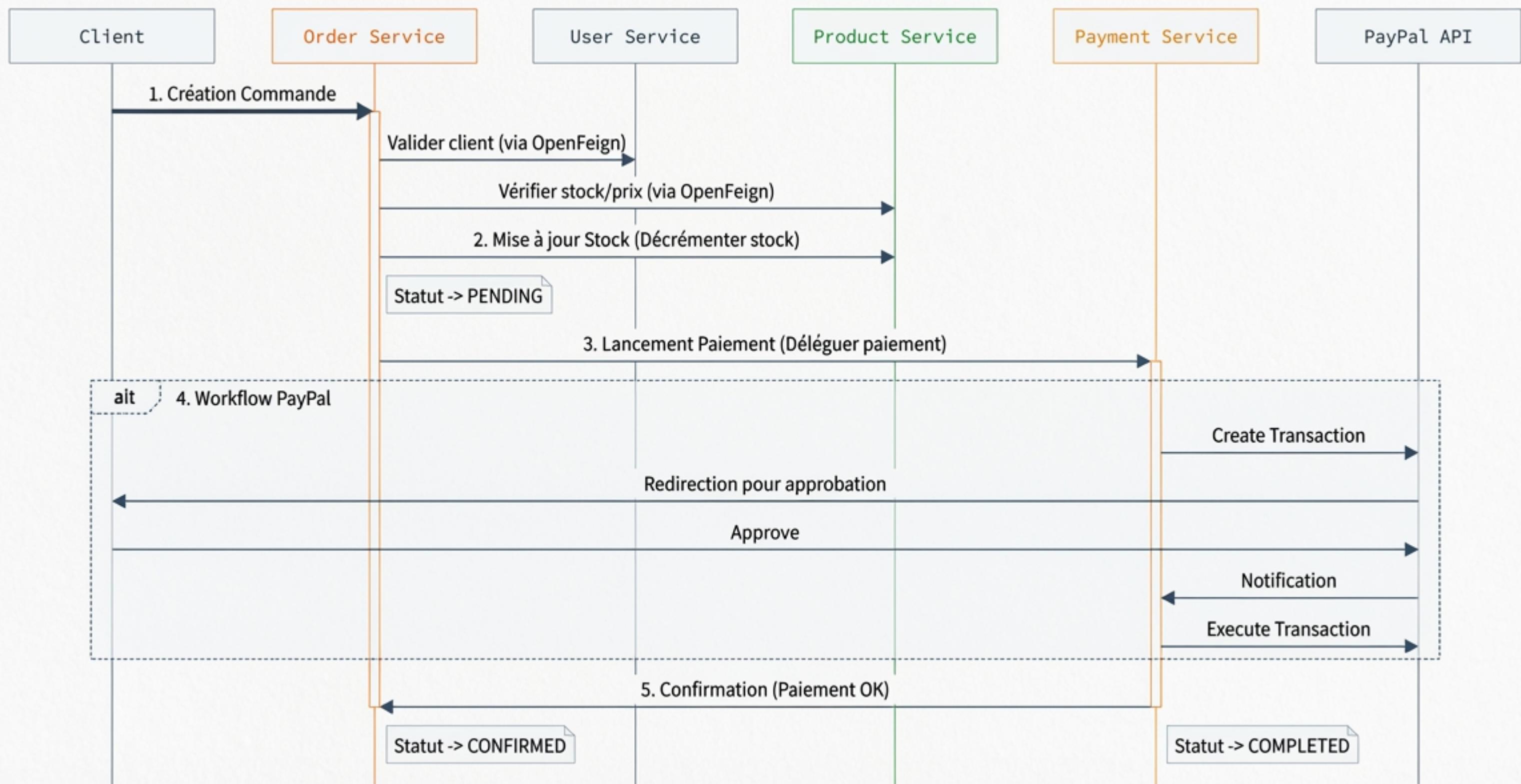


1. Requête Utilisateur
in Source Sans Pro

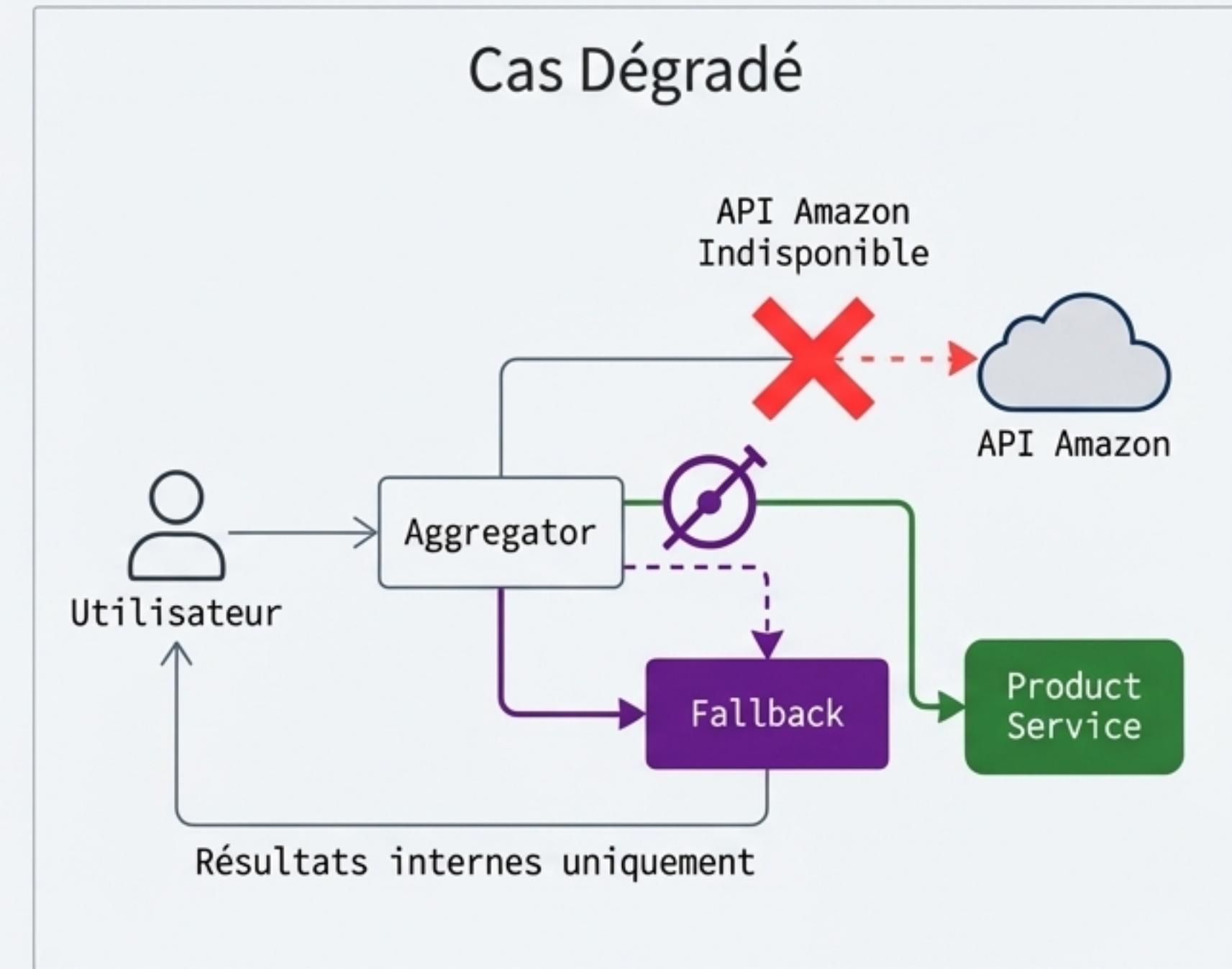
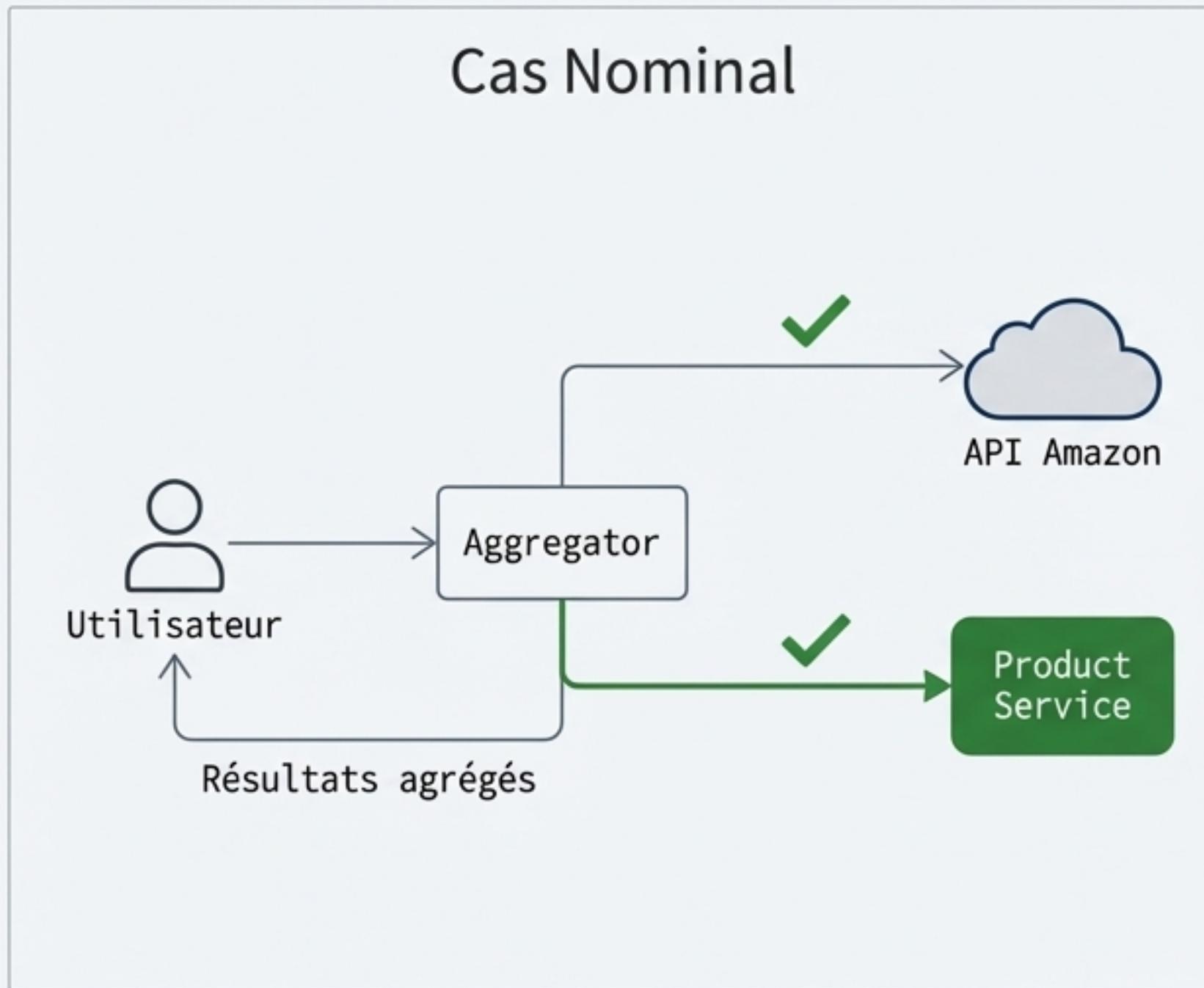
2. Routage
in #343A40

3. Appels Parallèles
in Source Sans Pro

Du Panier au Paiement : Un Achat Interne Sécurisé



La Résilience en Pratique : Le Scénario du Circuit Breaker



En cas d'échec de l'API externe, Resilience4j ouvre le circuit. Le service active un fallback et retourne uniquement les résultats internes, prévenant ainsi une défaillance en cascade du système.

Périmètre du Projet : Focus et Réalisme pour 8 Semaines

Inclus

- ✓ Architecture microservices complète (7 services fonctionnels).
- ✓ Catalogue interne avec gestion de stock.
- ✓ Agrégation d'UNE source externe (Amazon).
- ✓ Paiements sécurisés via l'intégration PayPal.
- ✓ Résilience assurée par des Circuit Breakers (Resilience4j).
- ✓ Communication inter-services via OpenFeign.
- ✓ Recommandations basées sur des mots-clés.

Exclu (Hors Scope)

- ✗ Intégration d'IA complexe (OpenAI GPT-4, Machine Learning).
- ✗ Agrégation de multiples sources externes.
- ✗ Cache avancé (Redis).
- ✗ Analyse de requêtes par la voix ou l'image.

Plan de Réalisation et Livrables Concrets



Livrables du Projet

	Code Source : Dépôt contenant les 7 microservices, scripts de démarrage.		Documentation : Cahier des charges, documentation technique et diagrammes UML.
	Tests : Tests unitaires et d'intégration validant les scénarios.		Collection Postman : Collection complète pour tester toutes les APIs REST.

Synthèse des Réalisations et Points de Validation

Principales Réalisations

- ✓ Architecture microservices de 7 services, démarrés et enregistrés sur Eureka.
- ✓ Agrégation fonctionnelle des produits d'une source externe (Amazon).
- ✓ Processus de paiement complet et sécurisé via PayPal Sandbox.
- ✓ Communication inter-services fluide et résiliente (OpenFeign & Resilience4j).
- ✓ Persistance des données découpée (bases H2 par service).

Critères de la Démonstration

1. Visualisation du dashboard Eureka avec les 7 services actifs.
2. Exécution d'une recherche affichant des résultats internes et externes.
3. Création d'une commande avec mise à jour du stock en temps réel.
4. Démonstration du flux de paiement PayPal.
5. Activation manuelle du Circuit Breaker pour montrer le fallback.

Une Fondation Robuste pour l'Avenir de l'E-commerce

Ce projet a permis de valider la mise en œuvre d'une architecture microservices complexe, distribuée et résiliente. La plateforme est non seulement fonctionnelle selon le périmètre défini, mais elle constitue surtout une base technique saine et évolutive, prête à intégrer de nouvelles sources de données et des fonctionnalités d'intelligence artificielle plus avancées.

Merci de votre attention.