

# Réalisez une application mobile de recommandation de contenu

**A** Microsoft Azure



*Projet 9*  
*Parcours Ingénieur IA*  
**OPENCLASSROOMS**

# Sommaire



**CONTEXTE**



**DESCRIPTION DU PROJET**



**SYSTÈME de RECOMMANDATION**



**VERSIONING GIT**



**SERVERLESS - AZURE FUNCTION**



**ARCHITECTURE**



## Contexte

- **My Content** est une start-up
- **Test** une solution de recommandation d'articles
- **Réalise** un MVP -> Application Android

## Objectifs

1. Modèle de recommandation
2. API serverLess « Azure function »
3. Architecture technique future

# DESCRIPTION PROJET

## 1 - Cahier des charges

### *Données d'entraînement :*

- « My Content » ne possède pas de données
- Jeu de données -> **News Portal User Interactions by Globo.com**

### *Fonctionnalités :*

- L'utilisateur de l'application, doit recevoir une sélection de cinq articles
- Prise en compte de l'ajout de nouveaux utilisateurs et de nouveaux articles

### *Environnement de travail :*

- Le projet devra utiliser une infrastructure serverLess sur le cloud Microsoft **Azure**
- Les scripts développés seront stockés dans un dossier GitHub

# DESCRIPTION PROJET

## 2 - Environnement de travail

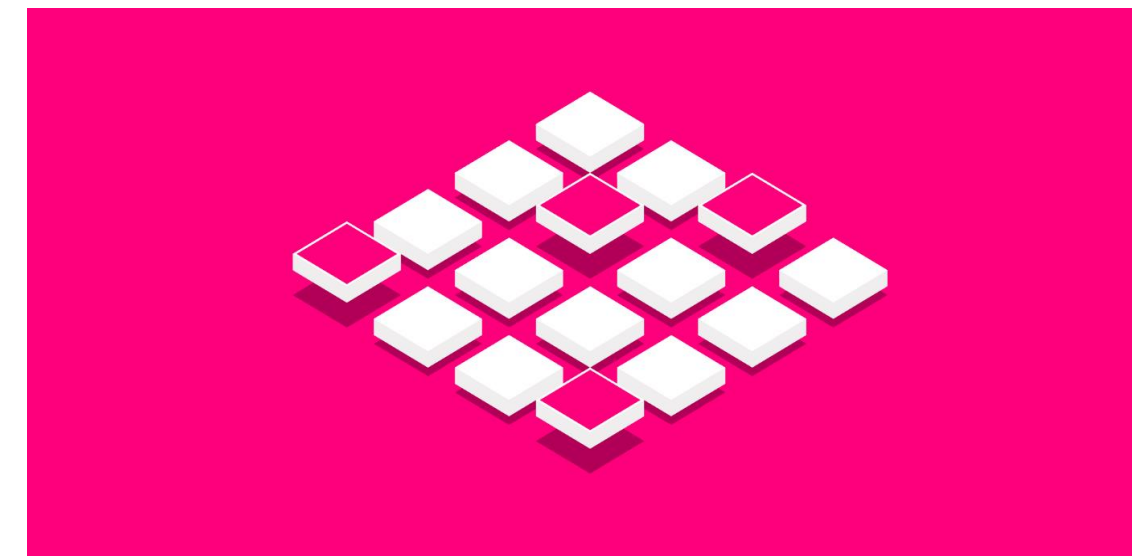
 Microsoft Azure

android  
studio 

  
python™



GitHub

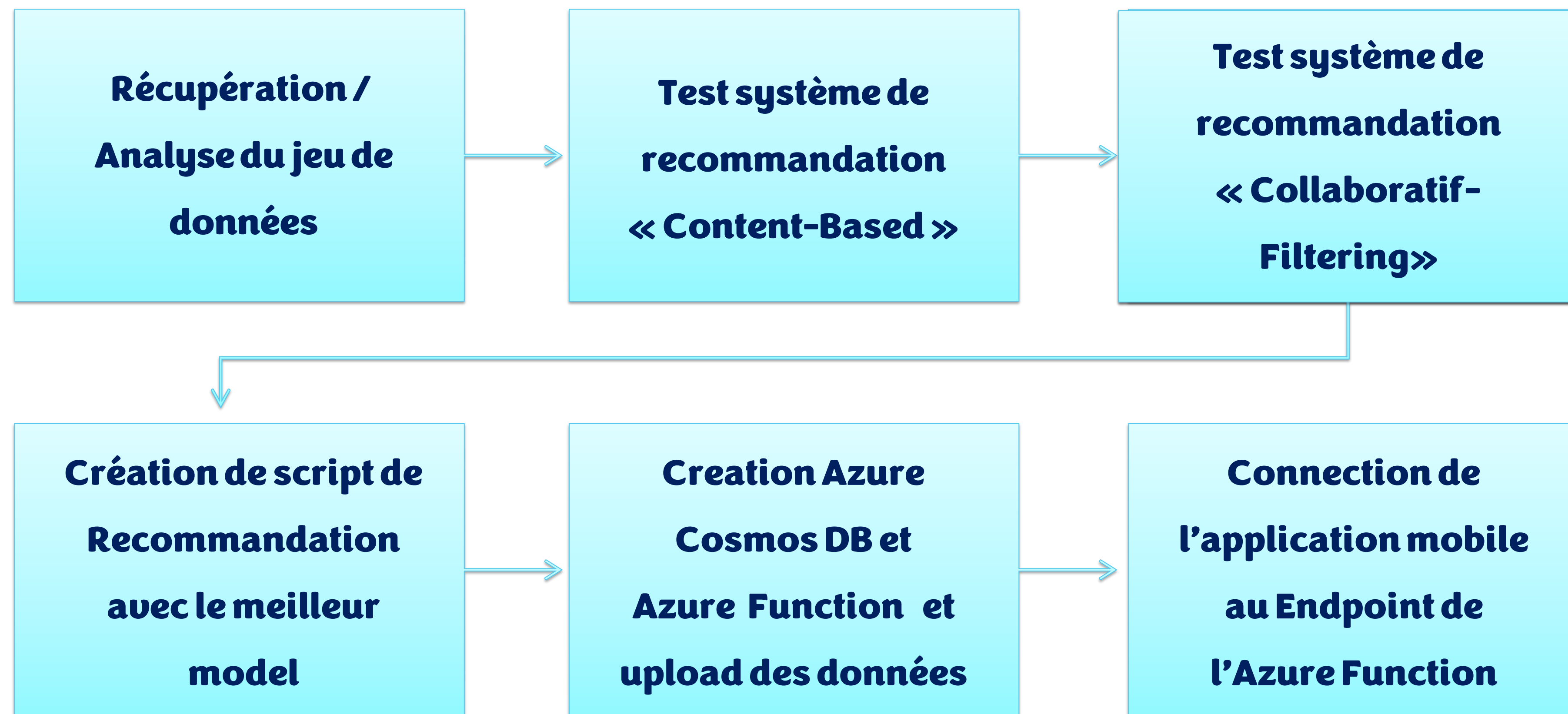


Implicit



# DESCRIPTION PROJET

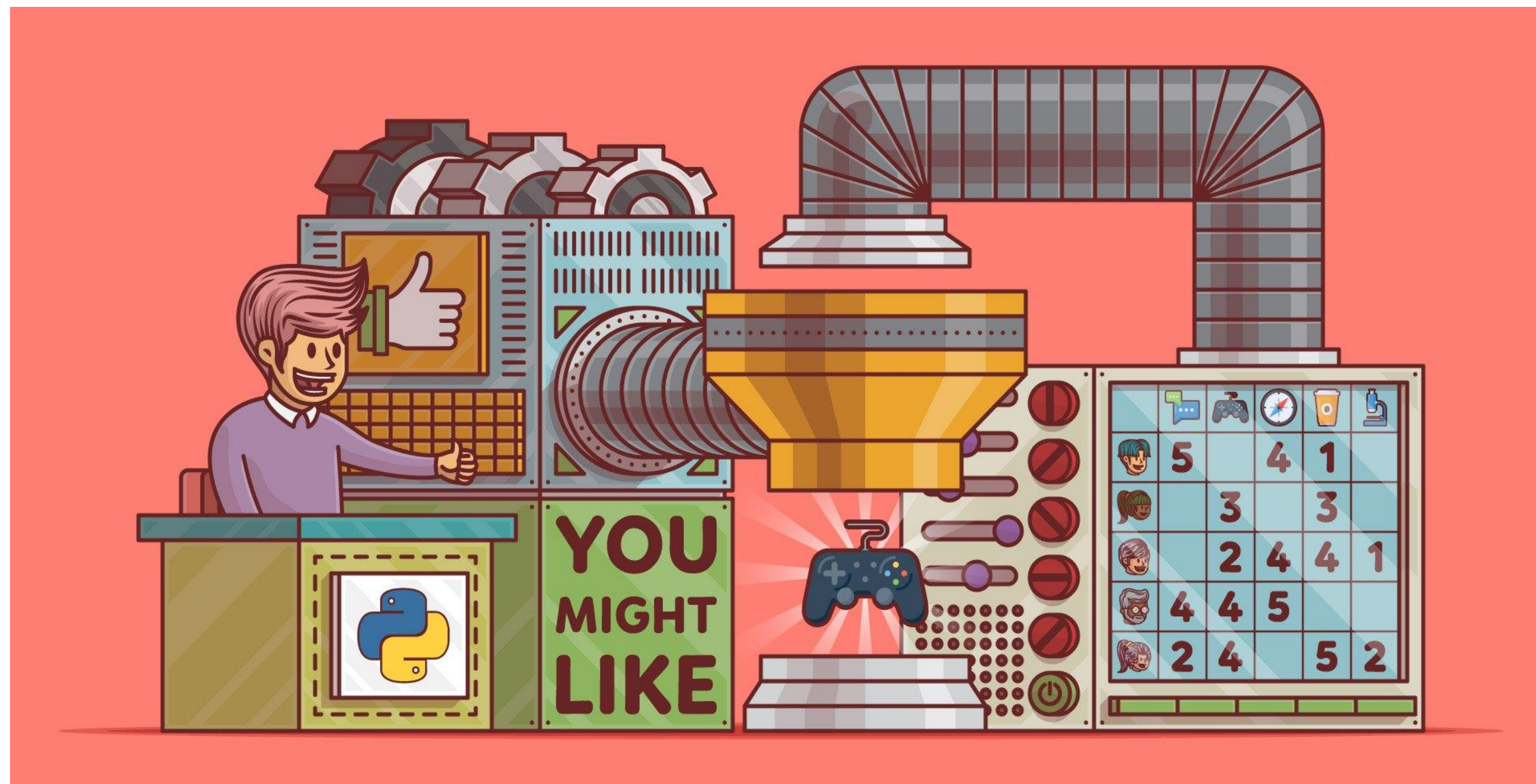
## 3 – Démarche méthodologique



# SYSTÈME DE RECOMMANDATION

*Qu'est ce qu'un système de recommandation*

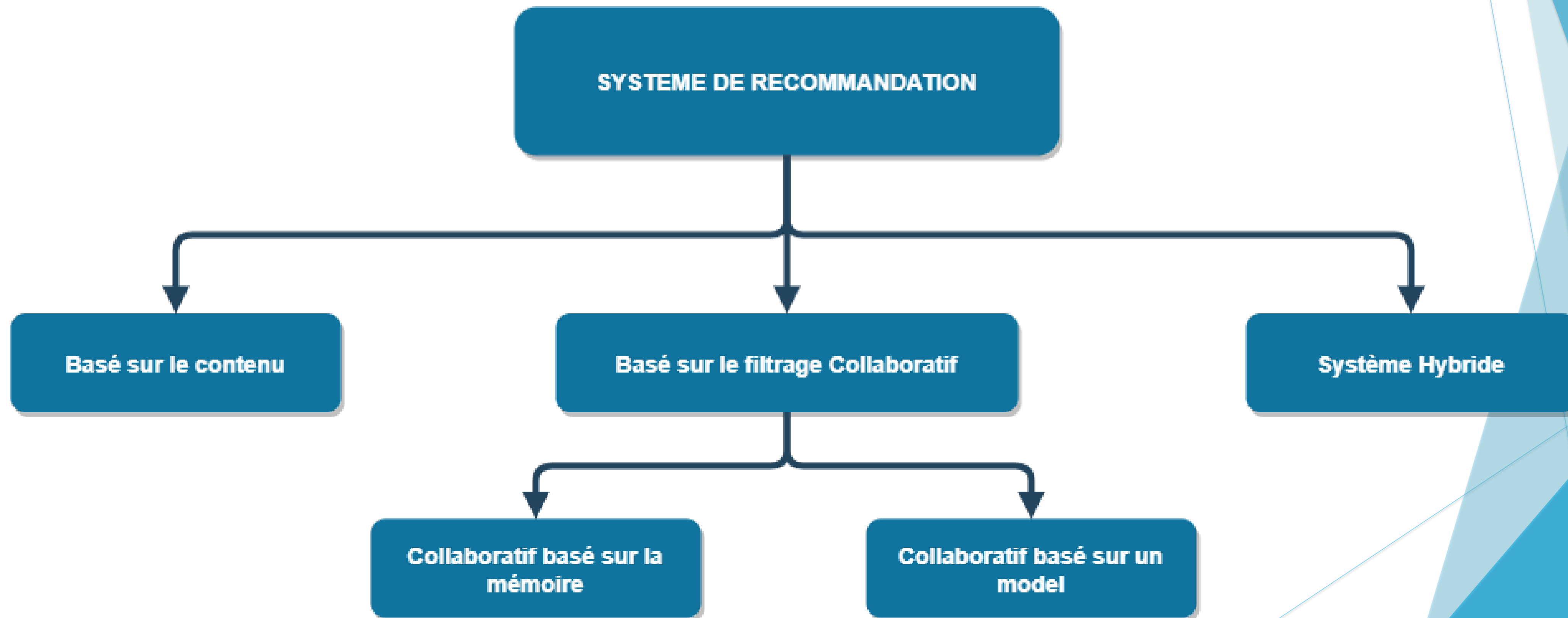
- Algorithmes visant à suggérer des éléments pertinents aux utilisateurs





# SYSTÈME DE RECOMMANDATION

*Les différents types de systèmes de recommandation*

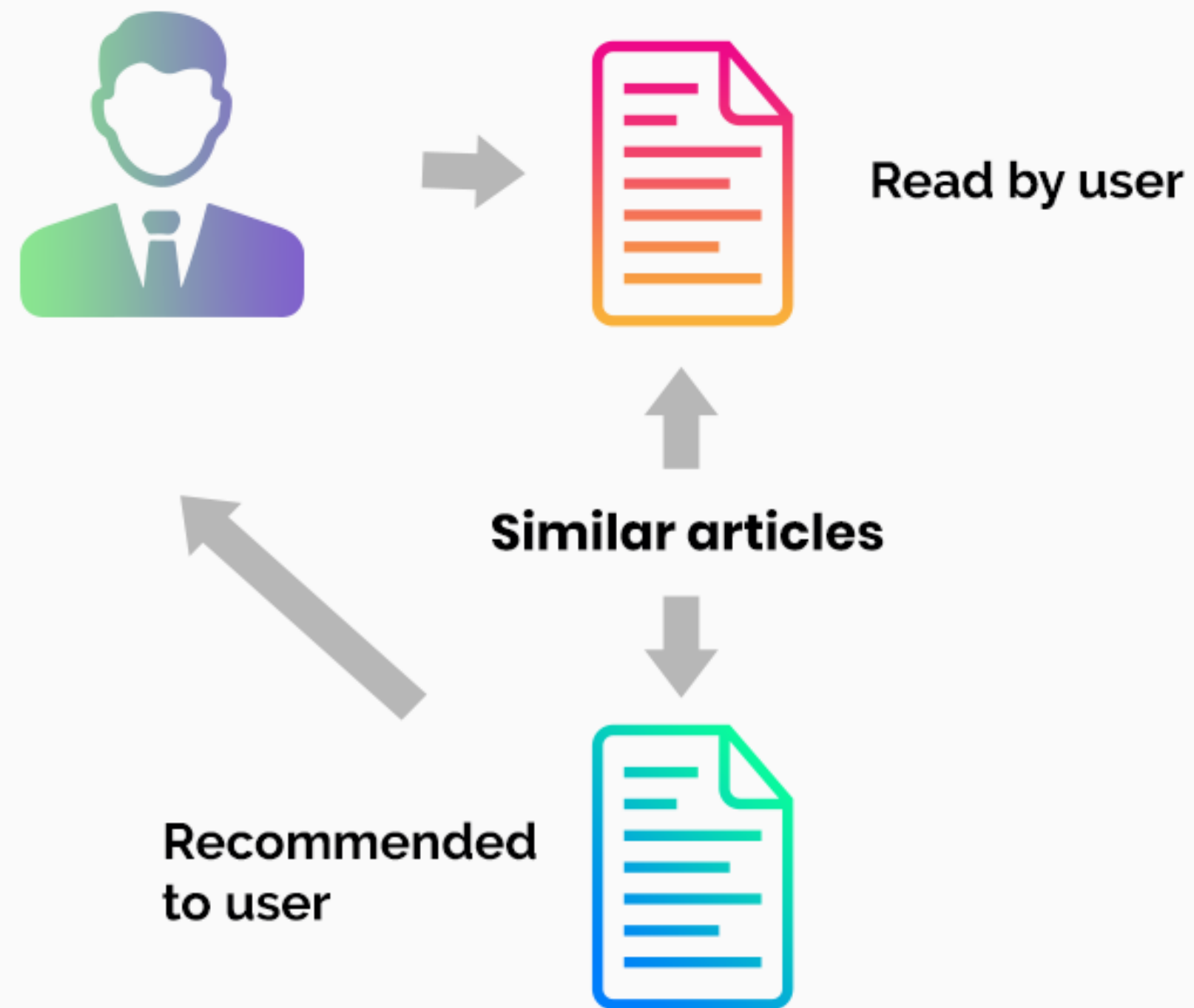




# SYSTÈME DE RECOMMANDATION

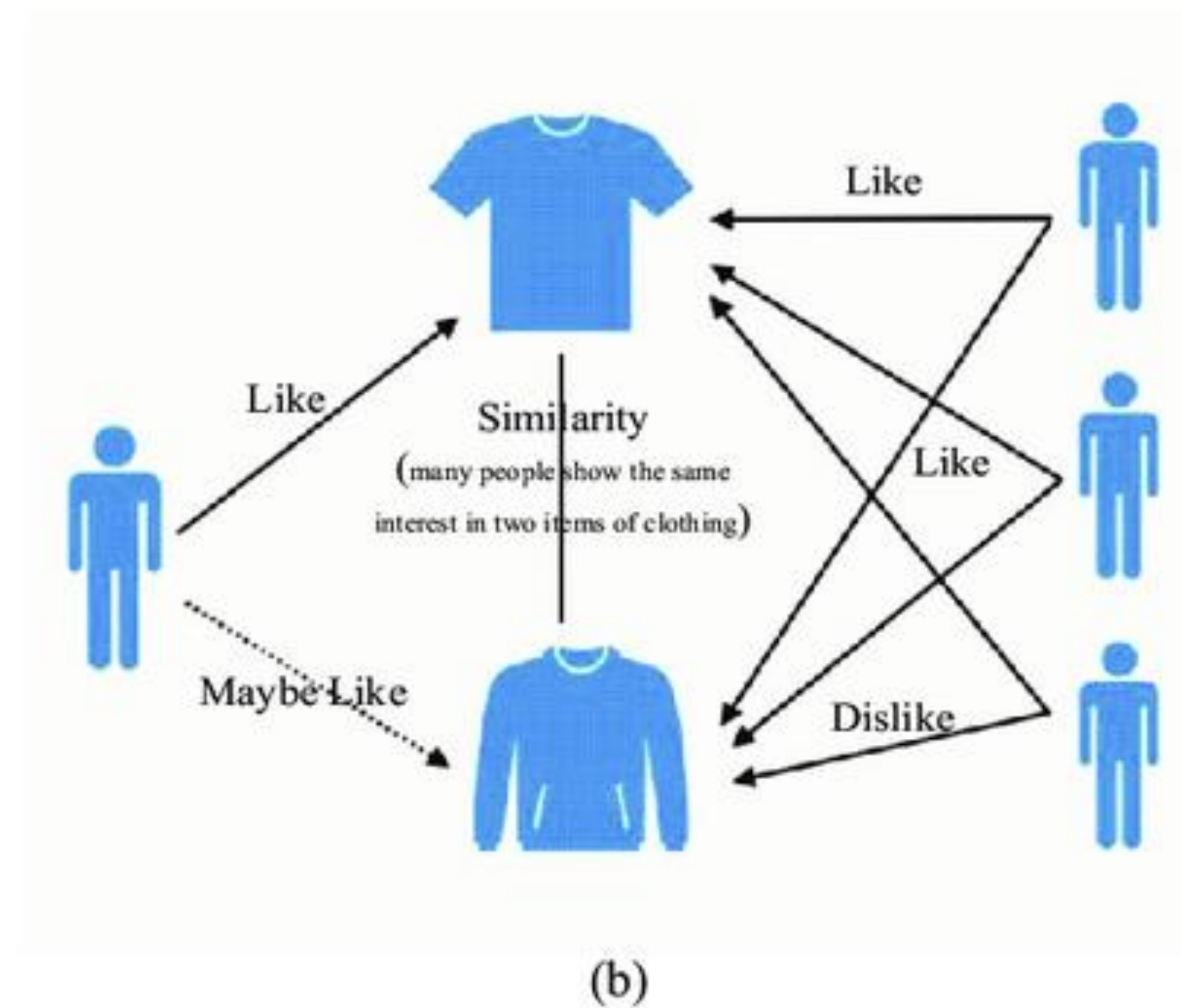
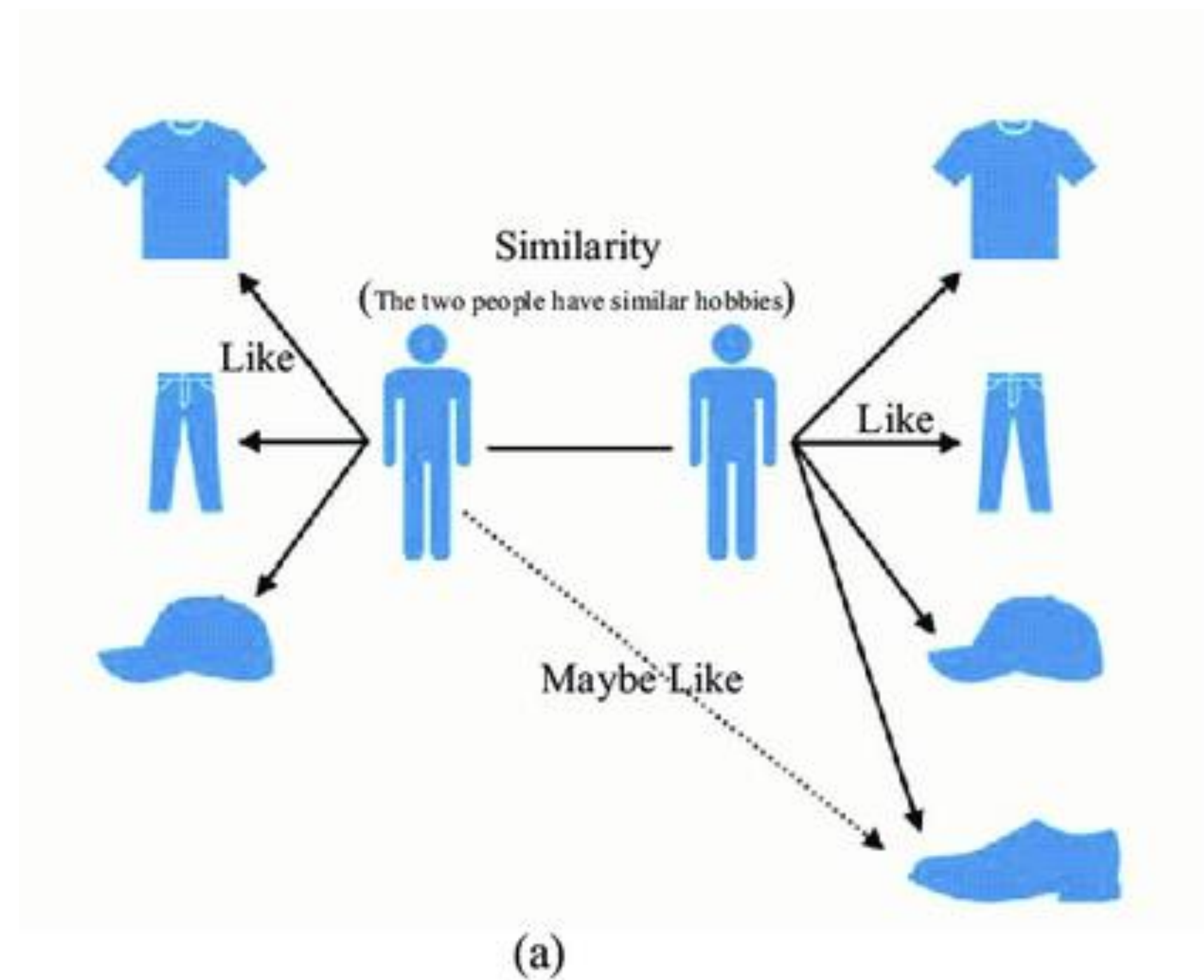
*Le système basé sur les données*

## Content-based filtering



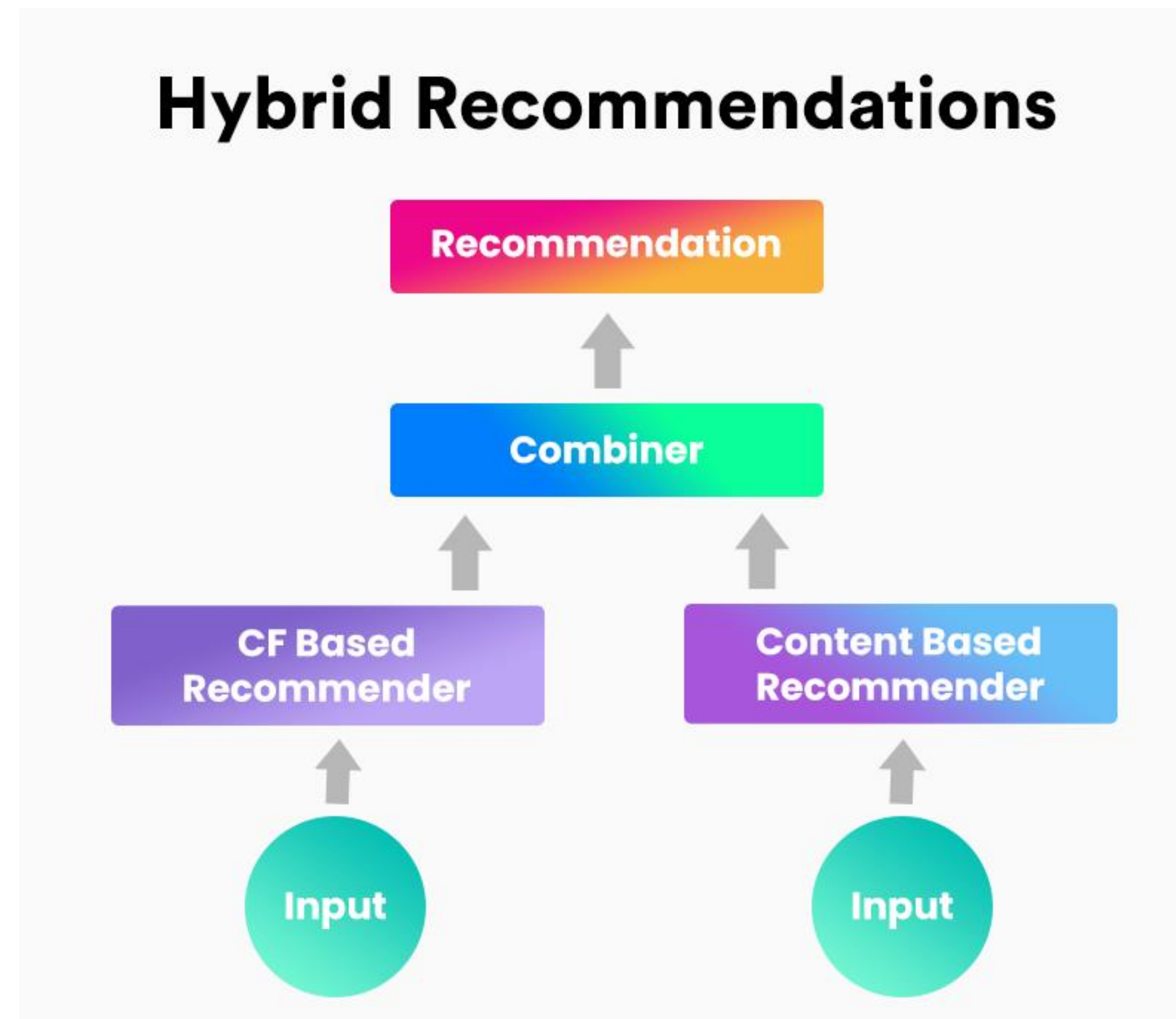
# SYSTÈME DE RECOMMANDATION

*Le système basé sur la méthode de filtrage collaboratif*



# SYSTÈME DE RECOMMANDATION

*Le système de recommandation hybride*





# SYSTÈME DE RECOMMANDATION

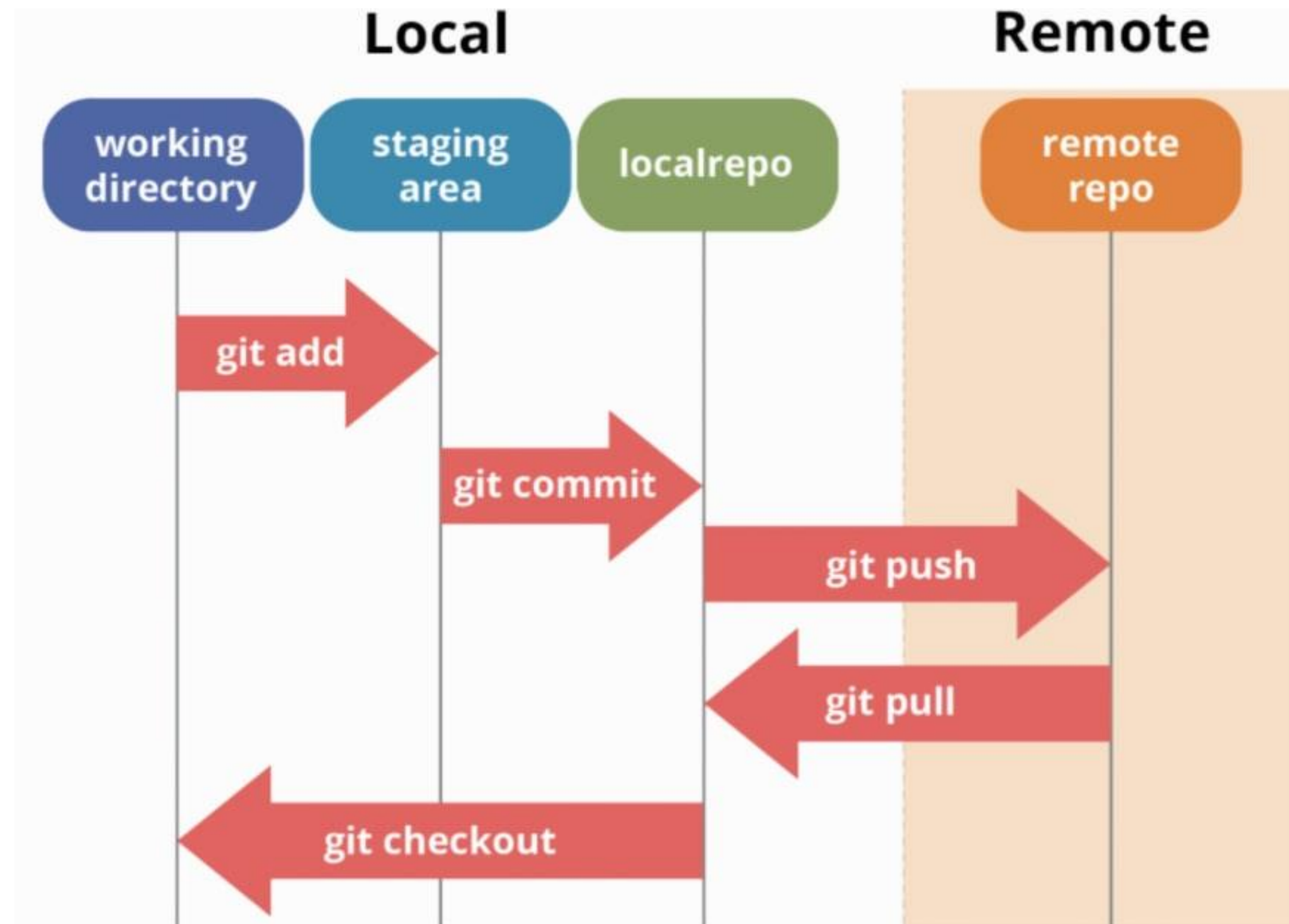
## AVANTAGES ET INCONVÉNIENTS

	CONTENT-BASED	COLLABORATIF FILTERING
AVANTAGES	<i>Peut recevoir des recommandations pertinentes dès les premières interactions</i>	<i>Pas besoin d'informations sur les articles et utilisateurs</i>
INCONVÉNIENTS	<i>Nécessite une grande quantité d'article et d'information sur les articles</i>	<i>Recommandation peu pertinentes/impossible si pas suffisamment d'interactions utilisateurs-articles</i>



# VERSIONING GIT

*Systeme de versioning*



# SERVERLESS

## *Qu'est ce que l'informatique serverless*

- Un modèle de développement cloud-native
- Le fournisseur met à l'échelle et gère automatiquement l'infrastructure
- Elimine la nécessité de gérer l'infrastructure.

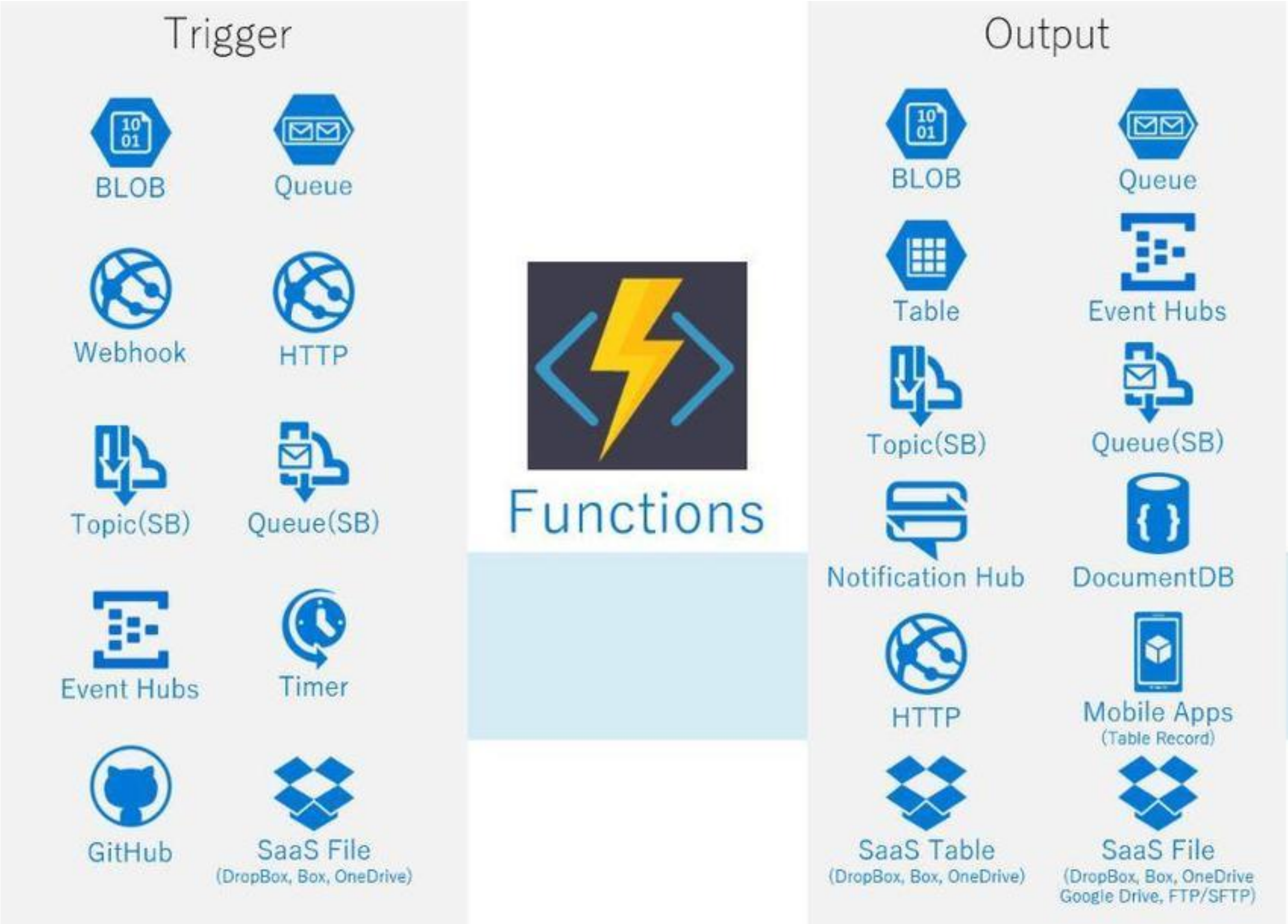
## *Les Avantages*

- Aucune gestion de l'infrastructure
- Time-to-Market plus rapide
- Scalabilité dynamique

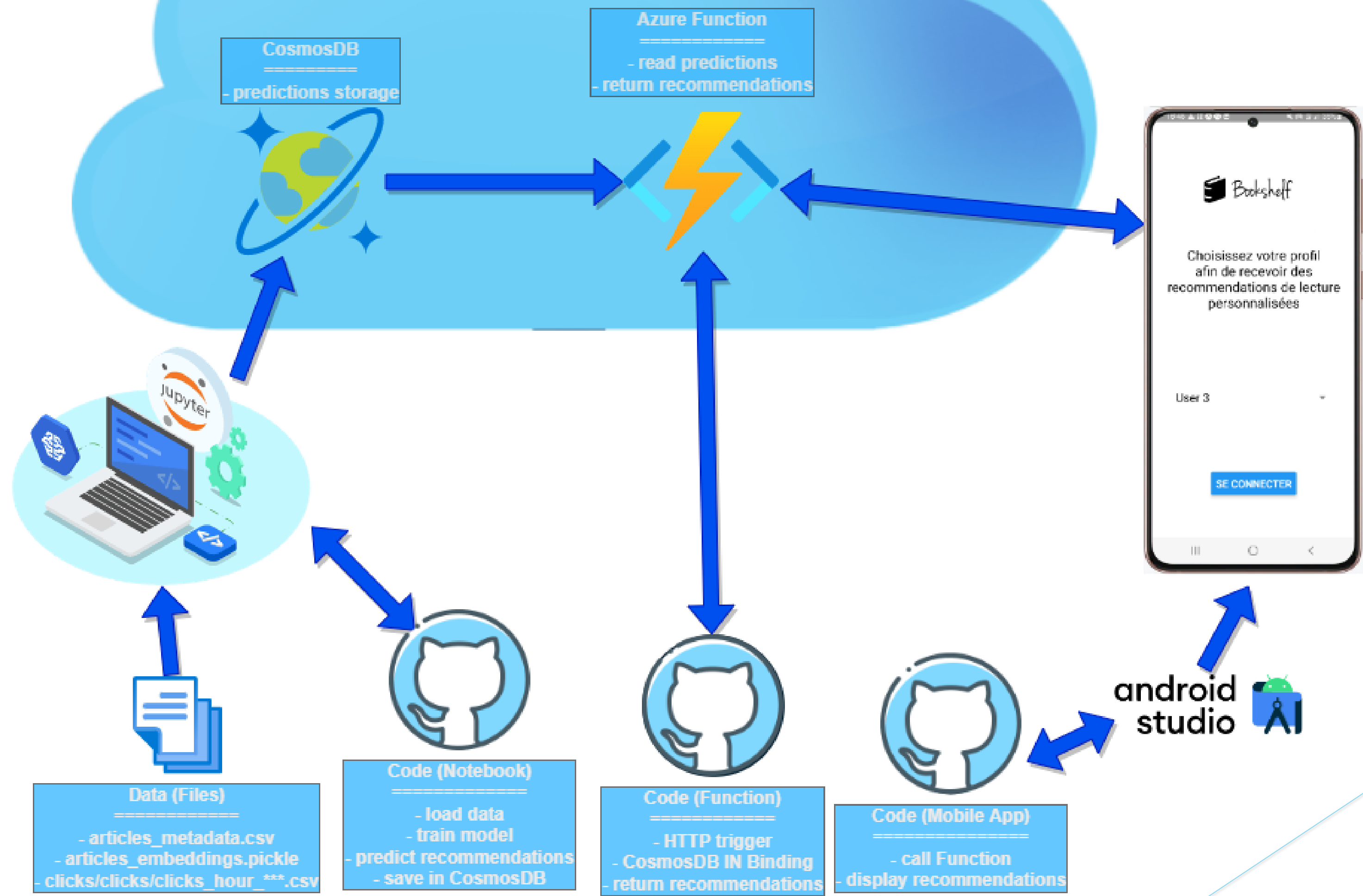




# SERVERLESS – AZURE FUNCTION



# ARCHITECTURE MVP

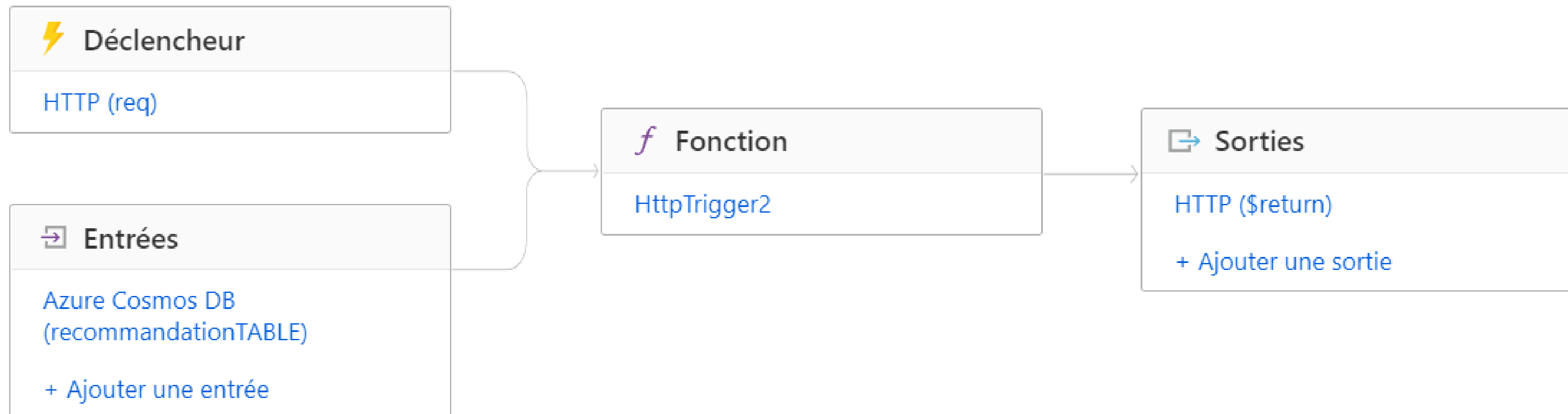




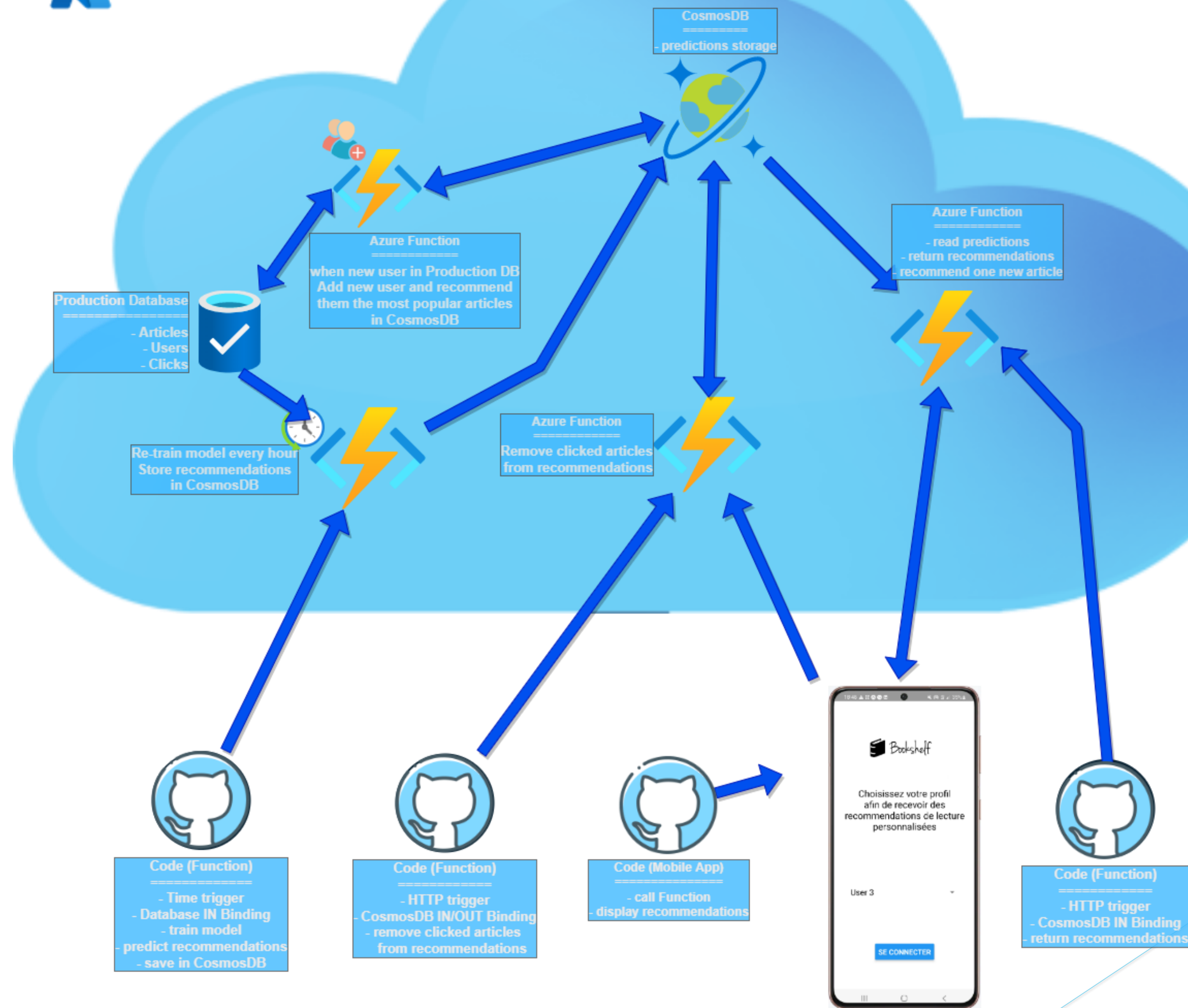
# SERVERLESS – AZURE FUNCTION



## Schéma d'Intégration – HttpTrigger + CosmosDB input Binding



# ARCHITECTURE CIBLE



# **PROCHAINES ETAPES**

**OPTIMISER LES HYPERPARAMETRES**

**TESTER UN MODEL HYBRIDE TYPE LIGHTFM**

**IMPLEMENTER L'ARCHITECTURE CIBLE**

**ÉVALUER EN CONDITIONS RÉELLES – TEST A/B**



# DEMONSTRATION

