

Document de conception

Rent & Stream Service

VO Lam Nhat Khuong
MENORET Valentin

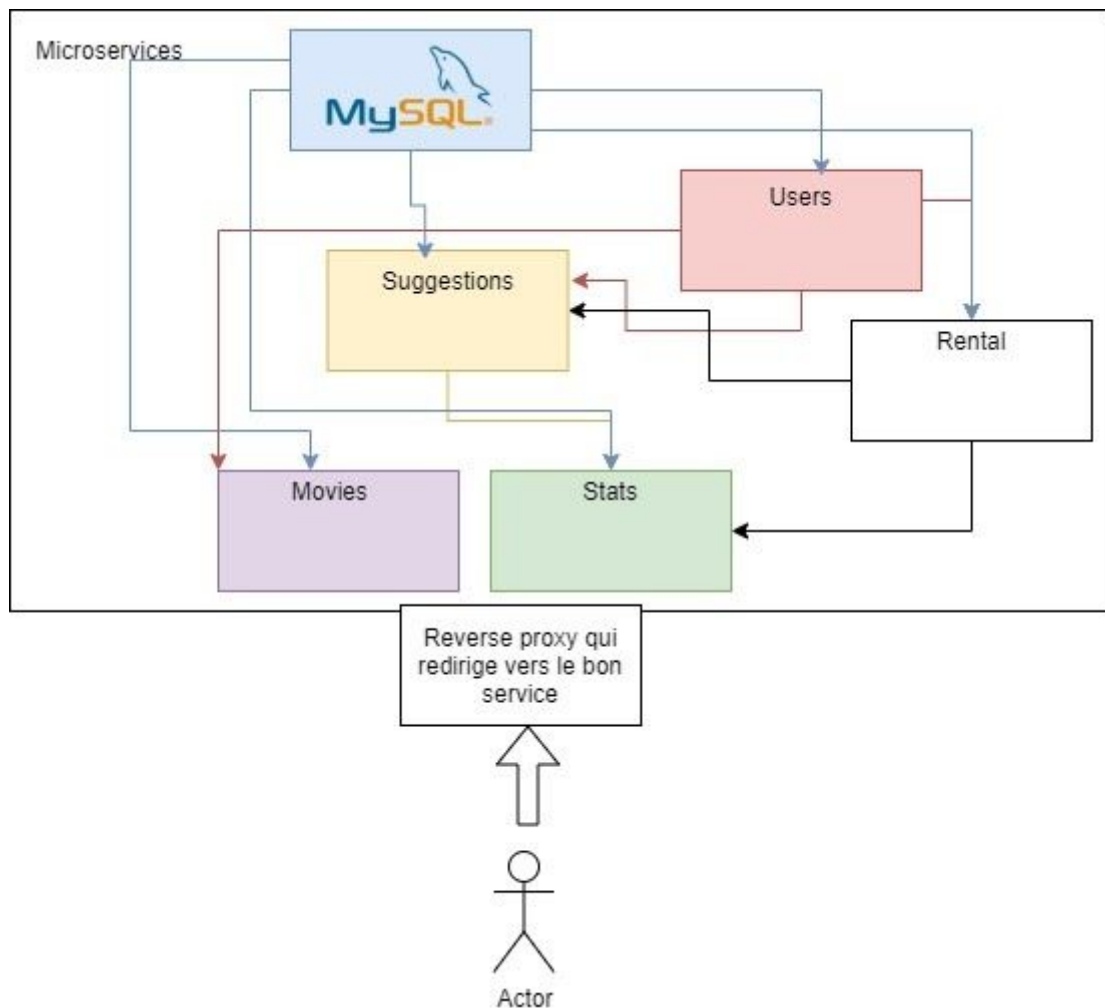


Schéma des composants de l'architecture

Comment les microservices communiquent-ils entre eux?

Les microservices communiquent d'une part via les appels HTTP au format JSON (qui passerait idéalement par le reverse proxy) mais aussi via la base données partagées si besoin (dans notre conception, nous partons du principe que chaque service a sa propre table bien à lui et n'a aucun accès aux autres tables de la base).

Description des microservices

Users

Table des utilisateurs. Stocke les informations principales de l'utilisateur tel que son nom et son adresse email. Elle est utilisée comme clé étrangère dans d'autres tables notamment Rental.

Suggestions

Suggestions faites aux utilisateurs à partir des données enregistrées sur les utilisateurs. Ce sont les recommandations qui sont faites à l'utilisateur à partir de ses données de visionnage.

Movies

Table des films. Ce service renvoie les différents films et permet de faire des requêtes sur ces derniers

Stats

Service sans base de données qui s'occupe de mettre à jour la table suggestion.

Rental

Service avec table de données qui gère la location de film aux utilisateurs. A ce but, il communique avec les deux services.

Documentation de l'API

GET	/users	Obtient la liste des utilisateurs
GET	/users/:id	Obtient les informations d'un utilisateur
PUT	/users/:id	Met à jour les informations d'un utilisateur
POST	/users/:id	Créer un nouvel utilisateur
DELETE	/users/:id	Supprime un utilisateur
POST	/users/token	Connexion de l'utilisateur
GET	/movies	Liste de tous les films
GET	/tags/:tag/movies	Liste de tous les films
DELETE	/movies/:id	Supprime un film
POST	/movies/:id	Créer un nouveau film

Définition du MVP

En plus de la base de donnée USERS, nous créons la base de données MOVIES pour y stocker les différents films :

```
mysql> CREATE DATABASE MOVIES;
```

```
CREATE TABLE `movie` (  
  `id` bigint(20) unsigned NOT NULL AUTO_INCREMENT,  
  `name` varchar(255) NOT NULL,  
  `realisator` varchar(255) NOT NULL,  
  `creation` DATE NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `id` (`id`)  
)
```