

Projet Microservice M2 INTENSE 2022/2023

Technos utilisées

- java (1.8 / 11)
- spring-boot 1.x.x
- maven 3.x.x

Context / tools

- Actuator
- Circuit breaker : Hystrix (Netflix)
- Discovery Eureka
- Swagger 2

Versionning

- github
 - commiter / pusher chaque partie que vous finissez
 - mettez dans un README les urls que vous exposez dans les différentes parties

Projet

- créer un 1er microservice spring-boot “my-movie”:
 - créer un objet film qui a un titre, réalisateur, acteur principal et date de sortie.
 - créer un object acteur qui un nom, prénom date de naissance et un filmographie
 - ***l'acteur principal dans l'objet film est l'objet acteur et la filmographie d'un acteur vient de l'objet film***
 - récupérer les films et acteurs selon les filtre demandés :
 - acteur :
 - tous les acteurs
 - par nom
 - liste des acteurs pour un film donné (ça peut être un seul acteur)
 - film :
 - tous les film
 - par titre
 - liste des films selon l'année de sortie (ça peut être un seul film)
 - mettre en place un swagger2 sur ce microservice pour décrire les params pour les deux objets, les expositions des objets avec les différents filtres et les codes d'erreurs retournés.
 - ***mettre en place les opérations delete et update sur les objets et les décrire avec swagger aussi.***
 - Ce microservice doit être un Eureka client
 - Ce microservice doit exposer l'endpoint actuator “healthcheck”
- créer un 2ème microservice spring-boot “my-finder”:
 - ce microservice doit pouvoir accéder à tout ce qui est exposé par “my-movie” et l'exposer lui à son tour sur un endpoint particulier
 - si un ou plusieurs endpoint(s) du “my-movie” est / sont indisponible(s), mon microservice my-finder ne doit pas crasher mais m'afficher simplement un message d'indisponibilité. / ***ou proposer une alternative***
 - l'utilisation de la même méthode d'indisponibilité (fallback) est acceptable.
 - décrire les méthodes exposées par ce microservice avec swagger2, ainsi que les messages et code d'erreurs attendus.
 - mettre en place une méthode pour vérifier l'état de santé du microservice “my-movie” et renvoyer cet état
 - Ce microservice doit exposer l'endpoint actuator “prometheus
 - Ce microservice doit être un Eureka client
 - Ce microservice expose un dashboard d'Hystrix

- créer un 3ème microservice spring-boot “my-dashboard”:
 - ce microservice expose tout simplement un dashboard Eureka avec les microservices qui s’enregistrent auprès de lui
 - ce microservice est l’admin de Eureka
 - récupérer le dashboard hystrix du microservice “my-finder” et l’exposer par ce microservice :
 - exposer directement le hystrix.stream avec le mode graphique (les courbes succès, erreurs)