



LOG430 Architecture logicielle Itération #1

Objectifs

- Configuration de votre environnement
- Déployer l'infrastructure du laboratoire
- Déployer des mécaniques de télémétrie
- Documenter la mise en place de l'infrastructure de télémétrie
- Documenter ce que la télémétrie vous révèle sur l'architecture du système
- Documenter la responsabilité détaillée de chaque composant du système
- Documenter les modules de chaque composant du système
- Proposer des solutions pour améliorer l'architecture du système
- Configurer Zipkins ou Jaeger

Configuration de votre environnement

- voir la Wiki sur Github

Démarrer l'application

- Remplir microservices\DockerCompose.env
- Utiliser dockstation (ou docker cli)
- Faire un lien vers microservices\DockerCompose\docker-compose.yml
- Build (global, pas sur un seul conteneur)
- Start

Déployer des mécaniques de télémétrie (open telemetry)

- Ajouter le conteneur Zipkin ou Jaeger
- Recueillir les données de télémétrie dans vos microservices STM, RouteTimeProvider et TripComparator.
- Envoyer ces données au conteneur Zipkin ou Jaeger.

Documenter ce que la télémétrie et l'analyse du code vous révèlent sur l'architecture du système

- Analyser votre architecture en termes de
 - ☐ disponibilité
 - ☐ performance

Perturbation de l'infrastructure

En sachant que la variable docker NanoCpus à 1000000000 est l'équivalent de 1 cœur de processeur, réaliser vos analyses télémétriques en fonction de cette variable à

- ☐ 0.01 cœur
- ☐ 0.1 cœur
- ☐ 1 cœur
- ☐ 2 cœurs
- ☐ 8 cœurs

Documenter la responsabilité détaillée de chaque composant du système

- ☐ on veut comprendre le rôle de chaque composant du système, en incluant les composants de télémétrie ajoutés
- ☐ réaliser une documentation (vue de type module) sur le microservice STM
- ☐ réaliser une documentation (vue de type composant et connecteur)
- ☐ réaliser une documentation (vue de type allocation)

Proposer des solutions pour améliorer l'architecture du système

En fonctions de vos analyses et de la description sommaire des itérations à venir documentées dans le document de spécifications-itérations, proposer des solutions pour améliorer l'architecture du système en termes de:

- ☐ disponibilité
- ☐ performance

Explication des tactiques de modifiabilité présentes dans la STM

Lors du laboratoire 3, vous utiliserez les tactiques de modifiabilité déjà présentes dans le microservice STM pour implémenter des changements et ainsi résister aux attaques. Expliquer quelles tactiques de modifiabilité sont présentes dans le microservice STM.