Architecture microservice LAO

Réaliser une architecture microservices en Java

Les microservices structurent une application comme un ensemble de services faiblement couplés. Elles nécessitent une grande maturité dans les pratiques de développement, de déploiement et de maintienen en conditions opérationnelles. Vous mettrez en œuvre, les technologies essentielles pour bâtir une telle architecture.

OBJECTIFS PEDAGOGIQUES

Définir une architecture basée sur les microservices

Développer et consommer des services REST

Mettre en place des microservices à l'aide d'un framework Java

Déployer des microservices dans des conteneurs DockerAdministrer et surveiller des microservices

1) Introduction

2) Les architectures microservices

3) Communications dans une architecturemicroservices

4) Développer un microservice en Java

5) Exécuter un micrcoservice à l'aide deconteneurs

6) Déploiement d'une architecture microservices

7) Administration et Monitoring

Travaux pratiques

1) Introduction

- Regard sur l'évolution logicielle et organisationnelle.

- Avantages et inconvénients des applications monolithiques.

- Méthodologies Agiles et pratiques DevOps.

- Processus d'intégration continue et de déploiement continu.

2) Les architectures microservices

- Définition d'une architecture microservices.

- Caractéristiques des microservices.

- L'émergence des architectures microservices.

- Les principaux acteurs.

- Quelques exemples des géants du Web.

3) Communications dans une architecture microservices

- Choix du style de collaboration : REST request/reply versus Publish

-Subscribe Messaging.

- Principes de base et rappels HTTP.

- Mettre en œuvre des services REST avec Java.

- Le principe HATEOAS.

- Documenter un service REST avec Swagger.

- Tester un service REST.

Travaux pratiques

Définition d'une API Rest avec Swagger et implémentation des services avec Jersey. Tester un service REST avec SoapUI.

4) Développer un microservice en Java- Présentation de frameworks Java "tout en un".

- Comparaison entre SpringBoot et Dropwizard.

- Développer une application avec Dropwizard ou SpringBoot.

- Packager une application production-ready.

Travaux pratiques

Développer une application avec Dropwizard ou SpringBoot.

5) Exécuter un micrcoservice à l'aide de conteneurs

- Présentation de Docker.

- Fonctionnement des conteneurs avec Docker.

- Concevoir des images personnalisées.

- Configurer et démarrer des conteneurs.

Travaux pratiques

Création d'un conteneur personnalisé avec Docker.

6) Déploiement d'une architecture microservices

- Techniques d'optimisation de déploiement

- Mettre en œuvre une application multi-conteneurs avec Docker Compose.

- Solutions d'infrastructure pour les microservices (Kubernetes, Mesos, Swarm).

- Solutions Cloud (Cloud Foundry, Heroku).

Travaux pratiques

Mise en place d'une architecture multi-conteneurs avec Docker Compose.

7) Administration et Monitoring

- Centralisation des logs.

- Le monitoring de microservices.

- Métriques et Health Check.

Travaux pratiques

Mise en place de métriques d'une application pour les visualiser.