

Bâtir une Architecture Microservice avec Quarkus

Bâtir une architecture microservices avec Quarkus

Attention : cette formation est une introduction théorique et pratique aux microservices et à Quarkus. Elle n'est pas destinée aux développeurs pratiquant déjà activement Quarkus et qui souhaiteraient approfondir leurs connaissances.

Cette formation a pour objectif de démystifier les architectures microservices. Elle allie présentations et séances de développement d'une application complète avec Quarkus, en mettant l'accent sur le découpage en microservices, et les pratiques de déploiement.

Après une présentation de la terminologie liées aux architectures microservices, vous devrez ensuite découper une application monolithique en plusieurs microservices. Viendront ensuite les réponses aux questions « comment distribuer ces microservices ? », « comment découvrir les APIs sur le réseau ? », « comment gérer les pannes réseaux ? », « comment faire face à la montée en charge ? ».

Enfin, vous terminerez cette formation en résumant les avantages et inconvénients des microservices afin de savoir si vos futurs projets pourront en tirer profits.

Détails

- Code : AE-QUA
- Durée : 3 jours (21 heures)

Public

- Architectes
- Consultants informatiques
- Développeurs
- Ingénieurs
- Professionnels de l'IT

Pré-requis

- Connaissance du protocole HTTP et des concepts REST
- Développer en Java
- Posséder une connaissance des architectures Web
- Avoir des notions de déploiement Docker

Objectifs

- Comprendre les avantages et les inconvénients d'une architecture microservices
- Découper un monolithe en plusieurs microservices
- Apprendre à développer et intégrer différents microservices avec Quarkus
- Prendre en compte les différentes pannes que peut subir une architecture microservices
- Déployer et tester une application microservices

Programme

La formation se compose de 40% de théorie et 60% de travaux pratiques

Principe des architecture microservices

- Agilité et monolithes
- Agilité et microservices
- Définitions
- Avantages et inconvénients

Patterns, vocabulaire et concepts liés aux microservices

- Organisation
- Stockage des données
- Interface utilisateurs
- Services et services distribués
- Sécurité
- Scalabilité et disponibilité
- Monitoring
- Déploiement et environnements

Microprofile

- Config
- Fault Tolerance
- Health Check
- Metrics
- Open API
- Rest Client
- JWT Authentication
- Open Tracing API

Quarkus

- HotSpot et GraalVM
- Développement et hot reload avec Quarkus
- Configuration et profiles
- Déploiement
- Extensions

Services REST

- Développer et exposer des microservices REST
- Documenter les services avec Open API

- Consommer les services avec Swagger Codegen
- Gérer les CORS
- Afficher des données sur une SPA Angular + TypeScript + Bootstrap

Monitorer les microservices

- Health checks
- Métriques
- Prometheus

Gestion des pannes et Scale des microservices (théorie)

- Circuit Breaker
- Load balancer
- Scaler horizontalement

Enjeux des architectures microservices

- Enjeux business
- Enjeux techniques
- Enjeux de production
- Avez-vous besoin d'une architecture microservices ?

Outils et frameworks utilisés :

- Java SE, Maven, Docker, Microprofile, Quarkus, REST, JAX-RS, JSON-B, Postgres, Junit

Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés – Cas pratiques – Synthèse
- **Validation** :Exercices de validation – Attestation de stages