

Formation Microservices, Concevoir, déployer et tester

Présentation

Cette formation de trois jours initie les développeurs à l'architecture micro-services : découpe raisonnée d'un domaine métier, création de services Spring Boot autonomes, sécurisation JWT, communication synchrone ou événementielle, observabilité, conteneurisation et déploiement. Chaque demi-journée se conclut par un atelier pratique : cartographie du domaine, service autonome sécurisé, puis mise en production et supervision, que le formateur adapte pour ancrer immédiatement les notions apprises. Un module « IA » montre comment un LLM peut générer un test de contrat ou un brouillon de manifeste de déploiement, tout en soulignant les bonnes pratiques de relecture et de conformité RGPD.

Durée : 21,00 heures (3 jours)

Tarif INTRA : [Nous consulter](#)

Objectifs de la formation

A l'issue de la formation, le stagiaire sera capable de concevoir, développer et déployer des microservices en utilisant les bonnes pratiques et les technologies adaptées.

- Comprendre l'architecture des microservices et identifier ses avantages
- Analyser différents frameworks et identifier le plus approprié au déploiement d'un microservice
- Distinguer les différents conteneurs (Docker, Azure) et leurs avantages pour les microservices
- Sécuriser la communication entre microservices
- Développer un microservice

Prérequis

Connaissances du langage Java

Public

Développeurs, architectes



Programme de la formation

Jour 1 – Comprendre et concevoir une architecture micro-services

- Distinguer micro-services et architecture monolithique, identifier gains et pièges
- Définir les frontières d'un service à partir du métier
- Rédiger un contrat d'API versionné conforme aux standards REST/gRPC

Contenu Jour 1

- Architecture distribuée expliquée : scalabilité indépendante, tolérance aux pannes, exigences d'automatisation.
- Méthodes de découpe métier : Domain-Driven Design, Bounded Context ; critères de taille, couplage, fréquence de changement.
- Contrats d'interface : pourquoi publier un contrat (OpenAPI, gRPC proto) avant d'écrire le code ; version et compatibilité.
- Introduction à la documentation automatisée
- Mise en situation : analyse critique d'une architecture existante

Atelier fil rouge Jour 1

- Choisir un domaine métier familier (ex. gestion d'événements, suivi de stocks, e-commerce).
- Identifier 3 à 4 micro-services logiques en appliquant les principes de découpe (responsabilité unique, faible couplage).
- Pour l'un des services, rédiger une première version de contrat d'API : principales opérations, structures de données échangées, versions...
- Générer un test de contrat

Jour 2 – Construire, sécuriser et rendre robustes les microservices

- Développer un service Spring Boot autonome et persistant.
- Sécuriser les échanges et gérer les défaillances réseau.
- Écrire les tests unitaires, d'intégration et de contrat

Contenu Jour 2

- Service Spring Boot autonome : démarrage rapide, configuration externalisée, profils dev/test/prod, tests unitaires de contrôleur.
- Persistance indépendante : une base ou un schéma par service, transactions locales ; événements d'intégration (Kafka, RabbitMQ) pour la communication et la cohérence inter-services.
- Sécurité et résilience : API Gateway, JWT pour l'authentification centrale, time-outs, retries et circuit breakers (Resilience4j) pour éviter les cascades de panne.
- Tests : unitaires, d'intégration, contrat

Atelier fil rouge Jour 2

- À partir du découpage précédent (ou d'un scénario fourni), développer un micro-service capable de créer, lire, mettre à jour et supprimer ses propres données.
- Mettre en place une couche de persistance dédiée, isolée des autres services.
- Ajouter un mécanisme d'authentification / autorisation et prévoir un dispositif de résilience (par ex. interruption contrôlée lorsqu'un appel externe échoue).

Jour 3 – Observabilité, déploiement continu et aide par l'IA

- Surveiller l'état des services et diagnostiquer les incidents
- Sécuriser les échanges et gérer les défaillances réseau.
- Conteneuriser un service et livrer vers un cluster
- Découvrir comment un LLM peut accélérer les tâches répétitives Devops ou qualité

Contenu Jour 3

- Observabilité « trois piliers » : logs structurés, métriques Prometheus, traces OpenTelemetry, visualisation Grafana pour remonter un parcours requête.
- Docker & Kubernetes simplifiés : image multi-stage, manifestes K8s/Helm, alternatives managées (AWS ECS, Azure Container Apps).
- Pipeline CI/CD : exemple GitHub Actions (build puis tests puis push image puis déploiement staging).
- IA pour la plateforme : exemple de prompt pour générer un test de contrat Pact ou un manifeste K8s, discussion sur biais et réglementation (RGPD) et prompt engineering encadré.

Atelier fil rouge Jour 3

- Emballer le service du Jour 2 dans un format exécutable et portable.
- Rédiger un descriptif de déploiement permettant de lancer ce service dans un environnement orchestré (cluster ou plate-forme cloud).
- Activer la collecte de journaux, métriques et traces, suivre la requête de bout en bout et détecter une anomalie simple à corriger manuellement.

Organisation**Formateur**

Les formateurs de Docaposte Institute sont des experts de leur domaine, disposant d'une expérience terrain qu'ils enrichissent continuellement. Leurs connaissances techniques et pédagogiques sont rigoureusement validées en amont par nos référents internes. Riches de leur expérience sur le sujet, ils sauront accompagner vos collaborateurs dans leur montée en compétences.

Moyens pédagogiques et techniques

- Alternance d'apports théoriques et de mises en pratique contextualisées.
- Progression pédagogique structurée, de la compréhension à la mise en œuvre opérationnelle.
- Pédagogie immersive favorisant une mise en application immédiate des compétences.
- Formation interactive et participative s'appuyant sur des ateliers pratiques, études de cas et échanges collectifs (+ 60 % du temps de formation.)
- Séquences pédagogiques rythmées et évaluations formatives régulières pour renforcer l'ancrage des apprentissages.
- Supports de cours, ressources documentaires et références mis à disposition par le formateur

Dispositif de suivi de l'exécution et de l'évaluation des résultats de la formation

En amont de la formation :

- Recueil des besoins des apprenants afin de disposer des informations essentielles au bon déroulé de la formation (profil, niveau, attentes particulières...).
- Auto-positionnement des apprenants afin de mesurer le niveau de départ.

Tout au long de la formation :

- Évaluation continue des acquis avec des questions orales, des exercices, des QCM, des cas pratiques ou mises en situation...

A la fin de la formation :

- Auto-positionnement des apprenants afin de mesurer l'acquisition des compétences.
- Evaluation par le formateur des compétences acquises par les apprenants.
- Questionnaire de satisfaction à chaud afin de recueillir la satisfaction des apprenants à l'issue de la formation.
- Questionnaire de satisfaction à froid afin d'évaluer les apports ancrés de la formation et leurs mises en application au quotidien.

Accessibilité

Nos formations peuvent être adaptées à certaines conditions de handicap. Nous contacter pour toute information et demande spécifique.