DAWAN

Plan de la formation

www.dawan.fr

Formation Architecture logicielle Avancée pour développeurs Java

Durée :	5 jours (35 heures)
Tarifs inter-entreprise :	3 775,00 € HT (standard) 3 020,00 € HT (remisé)
Public :	Développeurs Java expérimentés, Architectes
Pré-requis :	Expertise en développement Java Spring - Maîtrise de la programmation orientée objet et des concepts SOLID
Objectifs :	Mettre en œuvre des architectures microservices, orientée évènements - Apprendre les différents anti-patterns d'architecture
Modalités pédagogiques, techniques et d'encadrement :	• Formation synchrone en présentiel et distanciel. • Méthodologie basée sur l'Active Learning : 75 % de pratique minimum. • Un PC par participant en présentiel, possibilité de mettre à disposition en bureau à distance un PC et l'environnement adéquat. • Un formateur expert.
Modalités d'évaluation :	• Définition des besoins et attentes des apprenants en amont de la formation. • Auto-positionnement à l'entrée et la sortie de la formation. • Suivi continu par les formateurs durant les ateliers pratiques. • Évaluation à chaud de l'adéquation au besoin professionnel des apprenants le dernier jour de formation.
Sanction :	Attestation de fin de formation mentionnant le résultat des acquis
Référence :	ARC101593-F
Note de satisfaction des participants:	4,92 / 5

Contacts :	commercial@dawan.fr - 09 72 37 73 73
Modalités d'accès :	Possibilité de faire un devis en ligne (www.dawan.fr, moncompteformation.gouv.fr, maformation.fr, etc.) ou en appelant au standard.
Délais d'accès :	Variable selon le type de financement.
Accessibilité :	Si vous êtes en situation de handicap, nous sommes en mesure de vous accueillir, n'hésitez pas à nous contacter à referenthandicap@dawan.fr, nous étudierons ensemble vos besoins

Se rappeler des fondamentaux

Architecture logicielle : définition, documentation, vues architecturales
Démarche de production d'un dossier d'architecture technique
Panorama de composants logiciels et contraintes d'architecture
Styles d'architecture et organisation des couches d'une application

Concevoir une application basée sur les microservices

Décomposition fonctionnelle et Domain Driven Design
Exigences de conception des microservices : réactivité, compatibilité rétroactive, communication flexible, idempotency, fonctionnement efficace
La méthodologie à 12 facteurs
Structure d'un microservice, organisation des couches
Utilisation des évènements au sein d'un microservice

Atelier : Implémenter un micro-service avec Spring Boot, Web et Data JPA

Utiliser un message broker

Middleware Orienté Messages : intérêt, modes, protocoles
Architecture microservices orientée évènements
Panorama de messages brokers et comparaison

Atelier : Implémentation d'une architecture microservices orientée messages (producer/consumer) avec Spring AMQP ou JMS

Implémenter des WebSockets

WebSocket : le standard, intérêt et contextes d'utilisation
Schéma de communication des WebSockets
APIs disponibles

Atelier : Implémentation de web sockets avec Spring WebSocket et utilisation du protocole STOMP

Gérer efficacement des logs

Démarche globale de gestion des traces : support, type, quoi stocker
Interfaçage d'un microservice avec une solution de logs : ELK ou Graylog
Réflexion d'architecture : log direct, log via message broker, log direct vers fichier puis agrégation via un outil

Atelier : Configuration d'un microservice pour tracer en direct vers Elasticsearch, ou vers un message broker puis consommation via Logstash

Anti-patterns d'architecture logicielle

Anti-patterns : définition, détection
Panorama d'anti-patterns :
- Autogenerated Stovepipe
- Stovepipe Enterprise
- Jumble
- Stovepipe System
- Cover Your Assets
- Vendor Lock-In
- Wolf Ticket
- Architecture by Implication
- Warm Bodies
- Design by Committee
- Swiss Army Knife
- Reinvent the Wheel
- The Grand Old Duke of York