

Mis à jour le 27/06/2025

S'inscrire

Formation Vert.X

3 jours (21 heures)

Présentation

Maîtrisez Vert.x pour concevoir des applications réactives, asynchrones et hautement performantes sur la JVM. Cette formation vous guide pas à pas dans le développement d'APIs, de microservices et de systèmes distribués, en exploitant toute la puissance de Vert.x et de son modèle événementiel non bloquant.

Vous apprendrez à créer et organiser vos services autour de Verticles, à structurer vos APIs REST avec Vert.x Web, à communiquer entre services via l'Event Bus, et à sécuriser vos échanges avec JWT, OAuth2 ou TLS. Vous découvrirez également comment gérer les accès aux bases de données SQL ou NoSQL, et structurer des traitements efficaces via le modèle réactif.

Vous serez formé à construire des architectures distribuées robustes avec service discovery, résilience, WebSockets et intégration Kafka, MQTT ou RabbitMQ, afin de répondre aux besoins temps réel ou de haute disponibilité.

Vous saurez tester, packager et déployer vos services Vert.x en production avec Maven, Docker et Kubernetes, tout en assurant leur observabilité via des outils de supervision compatibles Prometheus et Grafana.

Comme pour toutes nos formations, elle se déroulera sur ma toute dernière version de [Vert.X](#)

Objectifs

- Comprendre l'architecture réactive de Vert.x et son modèle événementiel non bloquant pour concevoir des applications
- Installer, configurer et structurer un projet Vert.x avec Maven ou Gradle, en maîtrisant les verticles, l'event loop et le cycle de vie des composants
- Développer des APIs REST robustes avec Vert.x Web, en intégrant le routage, la gestion des erreurs, la sécurisation des endpoints et les middlewares essentiels

- Interfacer une application Vert.x avec des bases de données SQL et NoSQL, gérer les transactions, et structurer les échanges de données avec un modèle réactif
- Construire une architecture distribuée à base de microservices Vert.x, en utilisant l'Event Bus, le service discovery, les WebSockets, Kafka ou MQTT
- Superviser, tester et déployer une application Vert.x en production, en intégrant Docker, Kubernetes, Prometheus et des stratégies de résilience comme les circuit breakers

Public visé

- Développeur Backend
- Développeurs DevOps
- Architectes logiciels

Pré-requis

- Connaissances en principe de la programmation asynchrone
- Maîtrise d'un langage de la JVM

Programme de la formation Vert.x

Introduction à Vert.x et à la programmation réactive

- Architecture réactive vs architecture impérative
- Performances, scalabilité, asynchronisme
- Cas d'usage concrets
- Event loop, event bus, back-pressure
- Threads, concurrents, non-bloquant
- Core, Web, Data, Auth, MQTT, Kafka, etc.
- Comparaison avec Spring WebFlux, Akka, Quarkus

Installation, structure et cycle de vie

- Via Maven / Gradle
- Utilisation de vertx CLI
- Organisation en modules
- Types de verticles : standard, worker, coroutine
- Méthodes start(), stop()
- Déploiement, mise à l'échelle, shutdown

Programmation événementielle avec les Verticles

- Création en Java ou Kotlin
- Utilisation de `Vertx.vertx().deployVerticle`
- EventBus local et distribué
- Adressage, publication/souscription, RPC via proxy
- Event loops, modèles threading
- Worker Verticles pour les tâches bloquantes

Développement d'APIs REST avec Vert.x Web

- Utilisation de Router, Route, RoutingContext
- Matching d'URL, query params, path params
- Headers, bodies, statuts
- Uploads, téléchargement de fichiers
- Fichiers HTML, CSS, JS
- Moteurs de templates
- Authentification, CORS, cookies, sessions
- Logger, BodyHandler, ErrorHandler

Accès aux données

- Connexion, pool de connexions
- Exécution de requêtes, mapping objets
- MongoDB Client : insertion, recherche, agrégation
- Redis Client : cache, pub/sub
- Rollback, propagation, failover
- Utilisation des futures / promises pour le contrôle de flux

Sécurité et authentification

- Utilisateurs et rôles
- Gestion des sessions, JWT, OAuth2
- Configuration TLS
- Bonnes pratiques de sécurité
- Intercepteurs de sécurité
- Permissions personnalisées

Architecture réactive et microservices

- Communication inter-service via EventBus
- Design loosely-coupled
- Consul, Kubernetes, service registry
- Vert.x Circuit Breaker
- Retry, timeout, fallback

Intégration avec d'autres systèmes

- Vert.x Kafka Client
- Production/consommation de messages
- Utilisation de vertx-mqtt-client
- Protocoles IoT
- Communication bidirectionnelle temps réel
- Gestion des connexions clients

Tests, déploiement et monitoring

- Unitaires avec JUnit + VertxUnit
- Tests d'intégration et mocks
- Fat JAR avec vertx-maven-plugin
- Dockerisation et orchestration
- Vert.x Dropwizard Metrics
- Intégration avec Prometheus / Grafana

Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

Positionnement à l'entrée en formation

Le positionnement à l'entrée en formation respecte les critères qualité Qualiopi. Dès son inscription définitive, l'apprenant reçoit un questionnaire d'auto-évaluation nous permettant d'apprécier son niveau estimé sur différents types de technologies, ses attentes et objectifs personnels quant à la formation à venir, dans les limites imposées par le format sélectionné. Ce questionnaire nous permet également d'anticiper certaines difficultés de connexion ou de sécurité interne en entreprise (intraentreprise ou classe virtuelle) qui pourraient être problématiques pour le suivi et le bon déroulement de la session de formation.

Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.