

## Maitriser la programmation Spring Boot avancée

Date et durée
Code formation : SPRING2-CF-FR Durée : 5 jours Nombre d'heures : 35 heures
Description
<p>Spring Boot est un <b>framework Java open-source</b> qui offre une solution simple pour créer des applications autonomes au niveau production. Il repose sur le très populaire Spring Framework et vise à simplifier la configuration et le déploiement des applications Spring. Spring Boot suit le principe de la « <b>convention plutôt que la configuration</b> », ce qui permet aux développeurs de se concentrer sur l'écriture de la logique d'entreprise plutôt que de s'occuper de l'infrastructure et du code passe-partout.</p> <p>Spring Boot favorise le <b>développement d'architectures microservices</b> en offrant des fonctionnalités telles que la découverte de services, la configuration distribuée et la surveillance par le biais d'intégrations avec d'autres projets Spring tels que Spring Cloud. Il prend également en charge diverses <b>technologies d'accès aux données</b>, notamment JDBC, JPA et les bases de données NoSQL.</p> <p>Cette formation avancée sur Spring s'adresse aux développeurs ayant une connaissance préalable du <b>Framework Spring</b> et de Spring Boot. Le programme couvre de manière synthétique les apports de la dernière version du Framework et permet de s'initier à la sécurité des applications web. Vous apprendrez ainsi à <b>mettre en place une infrastructure à clé publique</b> et à faire de la programmation orientée aspect.</p>
Objectifs
<p>Au cours de cette <b>formation avancée Spring Boot</b>, vous atteindrez les objectifs pédagogiques suivants :</p> <ul style="list-style-type: none"><li>• appliquer les bonnes pratiques de la programmation orientée aspect ;</li><li>• sécuriser des applications avec Spring Security ;</li><li>• implémenter une infrastructure à clé publique ;</li><li>• déployer une architecture de microservices pour des applications et des outils techniques ;</li><li>• tester une application Spring Boot.</li></ul>
Points forts
Une formation avancée sur Spring Boot, un formateur en génie logiciel et des travaux pratiques en groupe (projet Fil rouge).
Modalités d'évaluation
Travaux Pratiques
Pré-requis
Suivre la <b>formation avancée Spring Boot</b> nécessite le prérequis suivant :

- connaître les fondamentaux du langage Spring Boot ou avoir complété la formation suivante (recommandée) :

Les formations ci-dessous sont recommandées.

Maitriser la programmation Spring Boot

Public

#### **Cette formation s'adresse aux publics suivants :**

- tout développeurs qui souhaitent se perfectionner sur le framework Spring Boot ;
- toute autre personne amenée à gérer des projets de développement logiciel.

Cette formation s'adresse aux profils suivants

Développeur

Architecte logiciel / Applicatif

Chef de projet / Responsable de projet

Programme

#### **Validation des données**

- Présentation du framework Hibernate Validator.
- Les annotations avec un exemple de base.
- Les annotations avancées.
- Le gestionnaire d'exception.

#### **Utilisation des tests unitaires et des tests d'intégration**

- Le concept de la pyramide des tests.
- L'approche du développement piloté par les tests.
- Présentation du framework JUnit.
- Les annotations JUnit.jupiter.
- Les assertions avec AssertJ.
- La vérification de la couverture des tests.
- Les annotations avancées de JUnit.
- Le principe FIRST.
- La simulation de composants avec le framework Mockito.
- L'auto-complétion avec Mockito.
- Le test de la couche de données.
- La structure de test des contrôleurs (MockMvc).
- La mise en œuvre des tests d'intégration.

#### **Généralités en matière de sécurité**

- Les extraits du code pénal.
- Les mesures de sécurité.
- La différence entre l'authentification et l'autorisation.
- Le domaine ou Realm.
- Le projet OWASP (Open Web Application Security Project).

#### **Sécurisation de Spring**

- L'authentification de base.
- L'API Postman et l'authentification.
- La classe de configuration.
- La configuration de Spring Security pour l'utilisation d'une base de données.
- La configuration de l'authentification de base HTTP.
- Le traitement des requêtes non authentifiées.
- L'autorisation des vulnérabilités de sécurité.
- La terminologie.

## **Communications cryptées**

- La différence entre HTTP et HTTPS.
- Les protocoles TLS et HTTPS.
- Le cryptographie à clé asymétrique.
- L'utilisation du protocole HTTPS.
- La fonction de hachage cryptographique.
- La signature numérique.
- Le certificat numérique et le certificat auto-signé.
- Les autorités de certification et les autorités racines.
- La mise en place d'une infrastructure à clé publique (PKI).
- La configuration de Spring pour le chiffrement.

## **Connaissances des vulnérabilités identifiées**

- L'injection de code SQL.
- L'attaque de l'homme du milieu (MITM).
- Le contrôle de l'intégrité des données.
- La politique de la même origine.
- Le partage des ressources inter-origines (CORS).
- La falsification des requêtes inter-sites (Cross-Site Request Forgery).

## **Contribution à la programmation orientée aspects (AOP) de Spring**

- Les limites de la programmation orientée objet.
- Les principaux concepts et la terminologie (aspect, advice, point de jonction et pointcut).
- Les différents types de conseils.
- La programmation Spring AOP par rapport à la programmation par AspectJ.
- L'implémentation d'un advice avec @Around, @Before et @After.
- Les bonnes pratiques en AOP (expressions pointcut, proxies et mise en cache).

## **Introduction à Spring Cloud**

- L'architecture distribuée et les microservices.
- Les modèles d'architecture microservices.
- Tour d'horizon de Spring Cloud.
- L'externalisation des configurations et la gestion de leurs versions avec Spring Cloud Config.
- L'enregistrement et la localisation dynamique des services avec Spring Cloud Netflix.
- L'implémentation de l'équilibrage de charge avec Spring Cloud LoadBalancer.
- La création d'une API Gateway avec ZUUL et la sécurisation de son accès avec Spring Security.
- La dernière ligne supprimée.