



# DEVOPS

## OBJECTIFS DE LA FORMATION

À l'issue de la formation, le stagiaire sera capable de :

- Comprendre et mettre en œuvre la méthode agile Scrum
- Construire un pipeline d'intégration et de déploiement continu
- Construire et déployer des applications conteneurisées
- Déployer des applications Kubernetes



### Public

Développeurs, Administrateur système, Ingénieur et analyste de production, Architecte, Chef de projet



### Prérequis

Maîtriser l'environnement Linux et les commandes de base du Shell (navigation, manipulation de fichiers, permissions).

Connaître les concepts fondamentaux de gestion de code source (Git) et des pipelines de déploiement.

Avoir des notions de conteneurisation et d'orchestration est un plus, mais non obligatoire.



### Durée

21h (3 jours)



### Évaluation

Contrôle continu (quiz, exercices, oral)

Évaluation de fin de formation



### Tarif

À partir de 1790€ HT

# CONTENU DE LA FORMATION

## MODULE 1 : INTRODUCTION AU DEVOPS

- Qu'est-ce que le DevOps, quel en est l'intérêt ?
- Les composantes du DevOps (planification, développement agile, intégration continue)
- Partage des responsabilités (environnement hors production/production)

*Exercice : Quiz*

## MODULE 2 : LES MÉTHODES AGILES

- Pourquoi développer en agile ?
- Organisation agile
  - Les intervenants
  - Focus sur le Product Owner et le Scrum master
  - Les différentes méthodes agiles
  - Les rituels
  - Les outils

*Exercices : Quiz + exercices pratiques*

## MODULE 3 : LA GESTION EN CONFIGURATION DU CODE SOURCE

- Pourquoi la gestion en configuration du code source ?
  - Le travail en mode collaboratif
  - Le travail en mode collaboratif
  - Les solutions du marché (git, ...)
- Focus sur Git
  - Accéder à Git et naviguer dans le code source
  - La notion de repository et de branches
  - Identifier les metadatas (commentaires, modifications, ...)
  - Les commandes importantes (pull, commit, push, ...)

- Pourquoi la gestion en configuration du code source ?
  - Le travail en mode collaboratif
  - La conservation des sources
  - Les solutions du marché (git, ...)
- Focus sur Git
  - Accéder à Git et naviguer dans le code source
  - La notion de repository et de branches
  - Identifier les metadatas (commentaires, modifications, ...)
  - Les commandes importantes (pull, commit, push, ...)

*Exercices pratiques*

## **MODULE 4 : LES CONTENEURS**

- Pourquoi packager une application sous forme de containers ?
- Les composants associés aux containers
  - Les différents moteurs de container (Docker, podman, ...)
  - Les orchestrateurs de containers (Kubernetes, Openshift, GKE, ...)
  - Le Dockerfile et les images Docker
  - Les Registries Docker
  - Comment construire une image Docker
  - Docker vs les orchestrateurs (que choisir ?)

*Exercices pratiques*

## **MODULE 5 : L'INTÉGRATION CONTINUE**

- Pourquoi l'intégration continue ?
- Les composantes de l'intégration continue
  - La plateforme d'intégration continue (Jenkins, GitLab, Bamboo, ...)
  - Les outils de build (Maven, Npm, ...)
  - La gestion des dépendances (Nexus, Artifactory)
  - Les tests (tests unitaires, tests de navigation, ...)
  - La qualité du code (SonarQube)

- Focus sur Jenkins
  - Comment se connecter
  - Naviguer dans Jenkins
  - Comment créer un job Maven et produire une image Docker
  - Construction du package XL Deploy en prévision du déploiement
  - Les pipelines (Niveau 2/3)

*Exercices pratiques*

## MODULE 6 : LE DÉPLOIEMENT CONTINU

- Le déploiement continu
  - Qu'est-ce que c'est ?
  - Responsabilités
  - Déploiement automatique
  - Externalisation de l'environnement
  - Déploiement d'un même package sur tous les environnements
  - Fonctionnalité de gestion des variables en fonction de l'environnement
- Déploiement sur Kubernetes
  - Architecture Kubernetes
  - Les principaux objets Kubernetes
  - Présentation de l'interface et du client mode commande (kubectl)
- Architecture XL Deploy
  - Qu'est-ce qu'XL Deploy ?
  - Notion d'application, d'environnement, de dictionnaire et d'infrastructure
  - Comment se connecter à XL Deploy
  - Naviguer dans l'interface
  - Configurer le déploiement d'une application
  - Déploiement du package en production sur Kubernetes
  - Utilisation des TAGS (Niveau 2)
  - Les orchestrateurs (Niveau 3)

*Exercices pratiques*

## MODULE 7 : LE MONITORING / LA SUPERVISION

- Pourquoi monitorer/superviser une application ?
- La collecte des données
  - S'assurer du bon fonctionnement de l'application
  - Identifier les anomalies sur une application
- Monitorer une application
  - Mettre en place la supervision
  - Identifier les améliorations possibles
  - Identifier les erreurs

*Exercices pratiques*

## MODULE 8 : L'AMÉLIORATION CONTINUE

- Pourquoi l'amélioration continue ?
- Comment opérer l'amélioration continue ?
  - La collecte des données
  - La collecte des logs applicatives
  - La consolidation des données et leur analyse
  - Les outils de consolidation de données (ELK, Splunk, ...)
- Focus sur la stack ELK
  - Comment se connecter
  - Naviguer dans l'interface
  - Accéder aux données et les analyser

*Exercices pratiques*

# INFORMATIONS PRATIQUES ET PEDAGOGIQUES

## Méthodes pédagogiques

- Une formation dispensée en présentiel ou à distance
- Une formation en direct avec le formateur
- Une formation 50% théorie / 50% pratique
- Une formation interactive : échanges, démonstrations, exercices, quiz, cas pratiques
- Un aide-mémoire remis en fin de formation

## Moyens techniques

- Matériel : ordinateur connecté à internet
- Poste virtuel sur un cloud (mis à disposition par Necform)

## Formateur

Notre formateur, ingénieur DevOps/Cloud, cumule plus de 20 ans d'expérience sur différentes technologies en rapport avec les métiers des infrastructures informatiques. Son expertise a été validée par nos équipes pédagogiques, tant sur ses compétences techniques que sur ses aptitudes pédagogiques.

## Modalités d'évaluation

- Prise de contact et test de vérification des prérequis en amont de la formation afin d'adapter le contenu du programme de formation si besoin
- Test d'entrée en formation pour évaluer les connaissances et compétences de l'apprenant
- Contrôle continu : quiz, exercices, évaluations orales, cas pratiques
- Evaluation finale : test écrit

La validation de la formation nécessite l'obtention d'un score minimum de 12/20 au test/projet final.

### **Modalités d'accompagnement et de suivi du stagiaire**

- Accompagnement et suivi par le formateur-expert et par le responsable pédagogique tout au long de la formation
- Echanges par mail et téléphone possible
- En cas d'absentéisme ou de décrochage, Necform prend contact immédiatement avec l'apprenant pour comprendre ses motivations et l'inviter à mener sa formation à son terme.
- Remise d'une attestation de formation en fin de parcours.

### **Accès personnes en situation de handicap**

La formation est accessible aux personnes en situation de handicap. Contacter notre référent rapidement pour prendre les mesures nécessaires : [contact@necform.fr](mailto:contact@necform.fr)

### **Modalités d'inscription**

- Date limite d'inscription :
  - 1 jour avant la formation pour un financement personnel (dans la limite des places disponibles)
  - 14 jours avant la formation (pour les financements externes)
- Aucune inscription possible après le début de la formation.