

GITLAB CI/CD : GESTION DU CYCLE DE VIE DE VOS DÉVELOPPEMENTS LOGICIELS

Durée : 3 jours (21 heures)

CONNAISSANCES PREALABLES

- Connaissance des principes de développement logiciel.
- Pratique des systèmes Linux ou Windows Server.
- Connaissances de base en gestion de versions avec Git.
- Notions de développement applicatif ou d'administration système.
- Une première approche des concepts DevOps constitue un avantage..

PROFIL DES STAGIAIRES

- Développeurs.
- Ingénieurs logiciels.
- Ingénieurs DevOps.
- Administrateurs systèmes.
- Administrateurs plateformes CI/CD.
- Responsables industrialisation logicielle.
- Architectes logiciels.
- Chefs de projets techniques.
- Toute personne impliquée dans l'automatisation des processus de développement et de déploiement.

OBJECTIFS

À l'issue de la formation, les participants seront capables de :

- Comprendre l'architecture et les composants de GitLab.
- Utiliser GitLab pour gérer le cycle de vie des développements logiciels.
- Maîtriser les principes de l'intégration continue et du déploiement continu.
- Concevoir et maintenir des pipelines GitLab CI/CD.
- Automatiser les phases de build, test et déploiement.
- Gérer les runners GitLab.
- Intégrer les bonnes pratiques DevOps et DevSecOps.
- Superviser et optimiser les pipelines de livraison.
- Sécuriser les processus de développement automatisés.

CERTIFICATION PREPAREE

Aucune

METHODES PEDAGOGIQUES

- Présentations interactives.

- Démonstrations en environnement GitLab.
- Exercices pratiques guidés.
- Travaux dirigés.
- Études de cas.
- Ateliers de conception de pipelines.
- Retours d'expérience terrain.
- Support pédagogique remis aux participants.

FORMATEUR

- Consultant expert DevOps, GitLab et automatisation des chaînes CI/CD, disposant d'une expérience opérationnelle dans la mise en œuvre de plateformes GitLab, l'automatisation des déploiements et l'industrialisation des processus de développement logiciel.

METHODE D'EVALUATION DES ACQUIS

- Quiz de positionnement.
- Exercices pratiques tout au long de la formation.
- Réalisation de pipelines CI/CD.
- Études de cas.
- Évaluation finale des acquis.
- Débriefing collectif des travaux réalisés..

CONTENU DU COURS

Module 1 : Découvrir GitLab et le cycle de vie des développements logiciels (3h30)

Objectifs

- Comprendre le positionnement de GitLab dans l'écosystème DevOps.
- Découvrir les fonctionnalités de gestion du cycle de vie logiciel.
- Maîtriser les concepts fondamentaux de GitLab.

Contenu

Introduction à GitLab

- Présentation de GitLab.
- Les éditions GitLab.
- Architecture générale de la plateforme.
- Positionnement dans une démarche DevOps.

Le cycle de vie applicatif (ALM)

- Gestion du code source.
- Gestion des demandes d'évolution.
- Gestion des anomalies.
- Collaboration entre équipes.

Les composants GitLab

- Projets.
- Groupes.
- Dépôts Git.
- Branches.
- Merge Requests.
- Issues.

- Wikis.

Flux de développement collaboratif

- Git Flow.
- Feature Branch Workflow.
- Trunk Based Development.

Mise en pratique

Atelier :

Création d'un projet GitLab, gestion des branches et mise en œuvre d'un workflow collaboratif.

Module 2 : Maîtriser GitLab CI et les pipelines d'intégration continue (3h30)

Objectifs

- Comprendre les principes de l'intégration continue.
- Construire des pipelines GitLab CI.

Contenu

Introduction à l'intégration continue

- Principes de la CI.
- Automatisation des builds.
- Détection précoce des erreurs.

Les pipelines GitLab CI

- Architecture des pipelines.
- Stages.
- Jobs.
- Artifacts.
- Variables.

Le fichier .gitlab-ci.yml

- Structure.
- Syntaxe.
- Paramétrage.

Déclenchement des pipelines

- Commit.
- Merge Request.
- Planification.
- Déclenchement manuel.

Mise en pratique

Travaux dirigés :

Création d'un premier pipeline GitLab CI réalisant compilation et validation automatisées.

Module 3 : Automatiser les tests et garantir la qualité logicielle (3h)

Objectifs

- Intégrer les contrôles qualité dans les pipelines.
- Automatiser les tests applicatifs.

Contenu

Tests automatisés

- Tests unitaires.
- Tests d'intégration.
- Tests fonctionnels.

Intégration de la qualité logicielle

- Contrôle qualité continu.
- Analyse statique de code.
- Mesure de couverture.

Gestion des rapports

- Rapports de tests.
- Rapports de couverture.
- Publication des résultats.

Validation des Merge Requests

- Quality Gates.
- Validation automatique.

Mise en pratique

Atelier :

Ajout de tests automatisés et génération de rapports dans un pipeline GitLab.

Module 4 : Déploiement continu et gestion des environnements (3h30)

Objectifs

- Comprendre les mécanismes de déploiement continu.
- Automatiser les déploiements applicatifs.

Contenu

Principes du déploiement continu

- Continuous Delivery.
- Continuous Deployment.
- Différences et cas d'usage.

Gestion des environnements

- Développement.
- Qualification.
- Préproduction.
- Production.

Stratégies de déploiement

- Rolling Update.
- Blue/Green Deployment.
- Canary Release.

Déploiements automatisés

- Gestion des secrets.
- Variables protégées.
- Validation avant mise en production.

Mise en pratique

Étude de cas :

Conception d'une chaîne de déploiement multi-environnements.

Module 5 : GitLab Runners et optimisation des pipelines (2h30)

Objectifs

- Comprendre le rôle des runners.
- Optimiser les performances des pipelines.

Contenu

Présentation des GitLab Runners

- Fonctionnement.
- Architecture.
- Types de runners.

Installation et configuration

- Shared Runners.
- Group Runners.
- Project Runners.

Optimisation des pipelines

- Cache.
- Artifacts.
- Exécution parallèle.
- Pipelines conditionnels.

Supervision des exécutions

- Journalisation.
- Diagnostic des erreurs.

Mise en pratique

Travaux dirigés :

Configuration et optimisation d'un pipeline à l'aide des runners GitLab.

Module 6 : Sécurité et DevSecOps avec GitLab (2h30)

Objectifs

- Intégrer les contrôles de sécurité dans les pipelines.
- Découvrir les pratiques DevSecOps.

Contenu

Introduction au DevSecOps

- Sécurité intégrée au cycle de développement.
- Shift Left Security.

Fonctionnalités de sécurité GitLab

- Analyse de vulnérabilités.
- Détection de secrets.
- Dépendances logicielles.

Gestion des accès

- Authentification.
- Autorisations.
- Gestion des rôles.

Conformité et gouvernance

- Audit.
- Traçabilité.
- Bonnes pratiques.

Mise en pratique

Atelier :

Ajout de contrôles de sécurité automatisés dans un pipeline CI/CD.

Module 7 : Industrialiser une chaîne CI/CD complète avec GitLab (3h)

Objectifs

- Mettre en œuvre l'ensemble des concepts étudiés.
- Construire une chaîne de livraison complète.

Contenu

Conception d'une plateforme CI/CD

- Organisation des projets.
- Structuration des pipelines.
- Gestion des dépendances.

Bonnes pratiques d'industrialisation

- Réutilisation des modèles.
- Templates GitLab.
- Standardisation des processus.

Exploitation des pipelines à grande échelle

- Gestion multi-projets.
- Maintenance.
- Amélioration continue.

Retours d'expérience et recommandations

Mise en pratique

Cas fil rouge :

Construction d'une chaîne CI/CD complète intégrant :

- Gestion du code source.
- Compilation automatisée.
- Tests automatisés.
- Contrôles qualité.
- Analyse de sécurité.
- Déploiement automatisé.
- Validation finale.

Notre référent handicap se tient à votre disposition au [01.71.19.70.30](tel:01.71.19.70.30) ou par mail à <mailto:referent.handicap@edugroupe.com> pour recueillir vos éventuels besoins d'aménagements, afin de vous offrir la meilleure expérience possible.