

Pipeline CI/CD Jenkins

Jenkins

- Jenkins est un outil open source d'intégration continue (CI) et de livraison continue (CD) largement utilisé dans le domaine du développement logiciel.
- Il est principalement utilisé pour automatiser le processus de construction, de test et de déploiement des applications, ce qui permet aux équipes de développement de fournir des logiciels de haute qualité de manière plus rapide et plus fiable.

Jenkins : qu'est ce que c'est?

- Jenkins est un outil logiciel d'intégration continu.
- Il s'agit d'un logiciel open source, développé à l'aide du langage de programmation Java.
- Il permet de tester et de rapporter les changements effectués sur une large base de code en temps réel.
- En utilisant ce logiciel les développeurs peuvent détecter et résoudre les problèmes dans une base de code et rapidement. Ainsi les tests de nouveaux builds peuvent automatisés, ce qui permet d'intégrer plus facilement des changements à un projet de façon continue.
- L'objectif de Jenkins est en effet d'accélérer le développement de logiciels par le biais de l'automatisation. Jenkins permet l'intégration de toutes les étapes du cycle de développement.

Intégration Continue (CI)

- Intégration Continue (CI) : Jenkins permet aux développeurs de valider leur code en continu en automatisant les tâches suivantes à chaque commit de code :
 - Compilation du code source.
 - Exécution de tests automatisés, y compris des tests unitaires, des tests d'intégration et d'autres types de tests.
 - Analyse statique du code pour identifier les problèmes de qualité du code.
 - Génération d'artefacts de déploiement.

Livraison Continue (CD)

- Livraison Continue (CD) : Jenkins étend la CI en automatisant également le déploiement et la livraison des applications :
 - Automatisation du déploiement sur différents environnements, tels que des serveurs de pré-production ou de production.
 - Possibilité de mettre en œuvre des pratiques de déploiement progressif ou de basculement pour minimiser les interruptions de service.
 - Intégration avec des outils de gestion de configurations pour garantir la cohérence de l'infrastructure.

Les fonctionnalités de Jenkins

- **Personnalisation et extensibilité** : Jenkins est très personnalisable et extensible grâce à un écosystème de plugins. Il propose des centaines de plugins qui permettent d'intégrer Jenkins avec diverses technologies, outils et services, ce qui facilite l'adaptation de Jenkins à des besoins spécifiques.
- **Gestion des Builds et des Jobs** : Jenkins permet de gérer plusieurs projets, chacun étant associé à un ensemble de tâches (jobs) spécifiques. Les utilisateurs peuvent configurer ces jobs en utilisant une interface graphique ou en écrivant des scripts Jenkinsfile en tant que code.

Les fonctionnalités de Jenkins

- **Environnement Distribué** : Jenkins peut être configuré pour fonctionner en tant que maître-esclave, permettant ainsi la distribution des tâches de construction et de test sur plusieurs nœuds esclaves pour améliorer les performances et la parallélisation.
- **Gestion de la Sécurité** : Jenkins offre des fonctionnalités de gestion de la sécurité, telles que l'authentification des utilisateurs, la gestion des autorisations, et l'intégration avec des systèmes d'authentification externes.

Les fonctionnalités de Jenkins

- **Rapports et Notifications** : Jenkins génère des rapports détaillés sur l'état des builds, les résultats des tests et les erreurs. Il peut également envoyer des notifications par e-mail, Slack, ou d'autres canaux de communication en cas de succès ou d'échec des builds.
- **Gestion de Versions** : Jenkins peut intégrer des systèmes de gestion de versions tels que Git, Subversion, ou CVS pour récupérer automatiquement le code source.

Jenkinsfile

- Un Jenkinsfile est un fichier texte contenant des instructions au format Groovy ou Declarative Pipeline Syntax spécialement conçu pour décrire le processus de construction, de test, et de déploiement d'un projet logiciel dans Jenkins.
- Il s'agit d'une approche dite "pipeline as code" (pipeline en tant que code) qui permet de définir et de gérer le flux de travail d'intégration continue (CI) et de livraison continue (CD) en tant que code source.

Résumé

- Ordonnanceur / Scheduler
- Run de jobs
- Pipeline d'intégration continue : build – run – test
- Java / open source
- Utilisé en devops
- 1000 plugins
- GUI : port 8080
- CLI possible