

Qualité de développement - Git avancé et IC

IUT lyon 1

Théo Rabut

Basé sur les cours de Matthieu Moy - Lyon1
et Thomas Ropars - Univ Grenobles-Alpes.

Git

- Rappels commandes et fonctionnalités de base
- Credentials
- Git status, git log

Gestion de tickets

Tickets ou Issues Gitlab

Vous avez la possibilité de créer un ticket sur la forge dans l'onglet "Issues". Ces tickets sont accessibles via le protocole Git. Ils ont un numéro "ID" qui sera appelé dans un message de commit
e.g. `git commit -m "Résolution bug affichage #1"`

Quand les utiliser ?

- Pour une fonctionnalité, un bug, un refactoring, des tests, etc.
- Sur certains projet, un commit doit obligatoirement être associé à un ticket.

Fork: rappel

- Duplication
- Modifications
- Soumission au dépôt original (pull request)
- Possibilité de garder ses modifications pour soi.

Merge-requests (aka pull-requests sur GitHub)

- Principe:
 - Un contributeur fait un "fork" du projet
 - Il fait un push dans une branche perso
 - Il demande à intégrer (merge) son code
- Intérêts:
 - Liste des branches à intégrer facile à voir
 - Discussion sur le code avant intégration
 - Possibilité de rejeter le mauvais code

Méthodo git avancé et agilité

- Branche pour chaque fonctionnalité, bug, ou modifications.
- Branche personnelle possible.
- Merge request pour envoyer sur la branche principale.
- Chaque commit est tagué vers une Issue Gitlab.

DevOps

Définition

Ensemble des techniques facilitant le passage du dev. à la prod.

Implications

- Communication
- Outils
- Détermine le fonctionnement d'une entreprise

Dans les faits

Dev Chargé du développement au moindre cout.

Ops Chargé de la mise en production.

DevOps: Automatisation

Continuous Integration

Une méthode de développement logiciel dans laquelle le logiciel est reconstruit et testé à chaque modification apportée par un programmeur.

Continuous Delivery

La livraison continue est une approche dans laquelle l'intégration continue associée à des techniques de déploiement automatiques assurent une mise en production rapide et fiable du logiciel.

Continuous Deployment

Le déploiement continu est une approche dans laquelle chaque modification apportée par un programmeur passe automatiquement toute la chaîne allant des tests à la mise en production. Il n'y a plus d'intervention humaine.

DevOps: Automatisation

Continuous Integration

Continuous Delivery

Continuous Deployment

Intégration Continue : l'idée

- Principe :
 - Intégrer le nouveau code au fur et à mesure qu'il est écrit
 - Garder une branche principale toujours fonctionnelle
- Outil indispensable : test automatisé à chaque push
- Abus de langage : « intégration continue » = « tests automatisés »

Intégration Continue - contexte

Historiquement réservé aux projets

- de grande taille
- impliquant de nombreuses personnes
- avec des itérations courtes

Technologies actuelles → accessible sur de petits projets → une fois qu'on y a goûté, on en fait partout !

Intégration Continue - principes

- Automatisation des phases du cycle de vie
 - Compilation, test, mise à disposition de binaires
- Institutions de bonnes pratiques
 - Commit réguliers
 - La branche par défaut compile
 - ...
- Surveillance
 - Tableaux de bord, etc

Continuous Integration: example with GitLab-CI

<https://gitlab.com/moy/gitlab-ci-demo>

- Configuration (`.gitlab-ci.yml`):

```
junit:
```

```
# docker image with Maven
```

```
image: maven:3.6.1-jdk-11
```

```
script:
```

```
- cd my-project/ && mvn test --batch-mode
```

- Use: work as usual ;-)
 - Tests launched at each `git push`
 - Pass/failed indicator for each merge-request

Serveurs d'IC

Permet d'exécuter régulièrement:

- Checkout
- Compilation
- Test
- Audit de code

Pour gitlab:

- peut servir de serveur d'IC;
- nécessite de gérer et de configurer des "runners" qui exécuteront les tâches

Alternatives: Jenkins, Travis-CI, GitHub actions, etc

SonarQube

Outil d'audit de code

- Analyse exécutée lors du cycle de vie
 - Via e.g. un goal maven
- Fournit des tableaux de bord:
 - Qualité du code
 - Couverture des tests unitaires