

# Manuel du responsable de version

## Créer un dépôt Gitlab avec intégration continue

Cette documentation vise à organiser le développement d'un projet à plusieurs développeurs en utilisant un dépôt commun sur Gitlab, géré par le moteur de production Gradle et utilisant l'intégration continue. Le projet est configuré pour l'IDE IntelliJ IDEA que chaque développeur est encouragé à utiliser. Les tests logiciels sont réalisés en JUnit 4 avec la bibliothèque Mockito.

Un exemple de projet « `regate.zip` » intégrant tous ces outils est fourni sur la plateforme pédagogique comme base de travail.

### 1. Procédure de création d'un compte gitlab

1/ Créez un nouveau dépôt avec l'interface du Gitlab de l'école accessible à l'adresse <https://gitlab.ecole.ensicaen.fr>.

- Il est important de laisser le projet **vide** pour éviter les éventuels conflits de fusion.

2/ Dans le menu « *Members* », ajoutez les membres de votre équipe ainsi que votre encadrant de TP comme développeurs.

3/ Dans le menu « *settings::CI/CD* », ajoutez un *runner* pour réaliser l'intégration continue. Pour cela choisissez l'option « *Runner* » et sélectionnez « *Enable Shared Runner* ». Un runner agit comme une machine virtuelle Linux et permet d'exécuter des commandes Linux. Ces commandes sont rassemblées dans un fichier nommé « `.gitlab-ci.yml` » dont un exemple est donné dans le projet fournit en exemple.

4/ Dans le menu « *Settings::General::Badges* », ajoutez des badges d'information pour voir fonctionner l'intégration continue : la compilation et les tests. Pour cela, faites deux badges :

- ▶ Pour le premier badge, mettez littéralement (ie, sans changer un caractère) :

Link : `https://gitlab.ecole.ensicaen.fr/{project_path}`

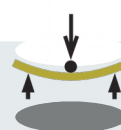
Badge image URL: `https://gitlab.ecole.ensicaen.fr/{project_path}/badges/master/pipeline.svg`

- ▶ Pour le second badge, mettez littéralement :

Link : `https://gitlab.ecole.ensicaen.fr/{project_path}`

Badge image URL : `https://gitlab.ecole.ensicaen.fr/{project_path}/badges/master/coverage.svg`

Si tout est correct, les badges (pipeline et coverage) doivent apparaître en haut de la page principale du projet :



CLOUARD Regis &gt; Vikazimut - Application &gt; Details

5/ Dans le menu « *Settings::CI/CD* », choisissez « *General pipelines* » et dans le champ « *Test coverage parsing* » mettez littéralement : `Total.*?([0-9]{1,3})%`. Ceci permet de récupérer la valeur de couverture dans le fichier retourné par l'outil Jacoco que nous utilisons pour le projet.

6/ Dans le menu « *General::Naming, tags, avatar* », vous pouvez aussi ajouter une petite description du projet ainsi qu'une image pour personnaliser votre projet.

## 2. Procédure de création de la première version du projet

Cette procédure est à faire sur le compte personnel du responsable de version. Elle utilise l'IDE IntelliJ IDEA :

1. Télécharger l'archive contenant l'exemple de projet « `template_regatta.zip` » à partir de la page du cours de la plateforme pédagogique.
2. Décompressez l'archive dans un dossier de travail.
3. Ouvrez le projet dans IntelliJ avec l'option « *Open or Import* ».
4. Lancez une exécution en utilisant la tâche *run* du menu *Gradle*. S'il le faut afficher la fenêtre Gradle par « *View::ToolWindow* ».
5. Lancez les tests à partir de la fenêtre Gradle à droite de l'interface et l'onglet « *Tasks::verification::check* ».
6. Dans le menu du projet, choisissez « *VCS::Enable Version Control Integration...* » et Git comme gestionnaire de version.
7. Dans le menu « *VCS::Git::remotes* », ajoutez le nom du dépôt distant :
  - **name** : origin
  - **URL** : (l'adresse est à récupérer sur le gitlab. Elle est du genre : `git@gitlab.ecole.ensicaen.fr:rclouard/essai.git`).
8. Modifiez le projet pour vous l'approprier. Pour cela, créez une nouvelle branche puis renommez le projet et les paquets sur une nouvelle branche git.
9. Poussez la branche sur la gitlab. Pour cela, faire « *VCS::Git::commit* » puis

« `VCS::Git::push` » (ou les deux raccourcis CTRL+K puis CTRL+SHIFT+K).

10. Sur Gitlab, créez une « merge-request » dans la branche master puis acceptez là.
11. Vérifiez que l'intégration continue s'exécute bien.

### 3. Procédure d'installation du dépôt des développeurs

Cette procédure est à faire par chaque développeur sur son compte personnel. Elle utilise l'IDE IntelliJ IDEA :

1. Fermer tous les projets dans IntelliJ et revenir à la fenêtre d'accueil.
2. Cliquer sur « *Checkout from Version Control* ».
3. Mettre l'URL du projet, quelque chose du genre :  
`git@gitlab.ecole.ensicaen.fr:rclouard/essai.git`.
4. Dans la fenêtre « *Event log* », choisissez « Import Gradle Project ».
5. Chaque développeur peut maintenant contribuer au projet en créant une branche par fonctionnalité puis en la poussant sur le gitlab.