

# Information sur l'utilisation de Gitlab Gestionnaire de sources Git

## Table des matières

### Accès à Gitlab

- Ajout d'utilisateurs CIP dans Gitlab
  - o Ajout autonome
    - Gitlab DI (Serveur - Projets bacc. ou pour les cours)
    - Gitlab dédié (Groupes de recherche ou projet particulier)
- Ajout de droits d'accès à un projet
- Ajout de droits pour la création de groupes et de projets
- Créer un Git sur Gitlab avec deux premiers fichiers **importants** (README.md et .gitignore)
- Création clé SSH pour pouvoir faire un clone d'un dépôt Git dans Gitlab
  - o Créer votre clé SSH via le serveur « drongo.dinf.ushesbrooke.ca »
  - o Copier votre clé publique (id\_rsa.pub) dans votre compte sur le serveur Gitlab
  - o Vérification du nouveau dépôt Git

## Accès à Gitlab

Vous pouvez accéder à Gitlab avec votre CIP via un navigateur web à l'adresse du serveur indiquée dans le courriel qui vous a été envoyé.

<https://dinf-nom-du-serveur.dinf.fsci.usherbrooke.ca/>

ex. : <https://depot.dinf.usherbrooke.ca/>

### **Ajout d'utilisateurs CIP dans Gitlab :**

Ajout autonome

**Gitlab DI** (Serveur - Git pour des projets bacc. ou pour des cours)

<https://depot.dinf.usherbrooke.ca/>

Tous les CIP des étudiants qui ont des cours au département d'informatique peuvent se connecter sur le Gitlab DI et se créer un compte d'une façon autonome. Ce compte aura accès à ce qui est public et ne pourra pas créer de projets ou de groupes par défaut.

Possibilités :

- 1) Être associé à un projet par le professeur responsable dans le cadre d'un cours ou de recherche
- 2) Être associé à un projet de bacc. par le responsable du projet
- 3) Faire la demande pour un projet personnel
- 4) Faire la demande dans le cadre d'un projet de bacc

**Gitlab Dédié** (Groupes de recherche ou projet particulier)

Tous les CIP qui sont dans le groupe de ressources utilisateurs du groupe de recherche peuvent se connecter sur le Gitlab du groupe de recherche et se créer un compte d'une façon autonome. Ce compte aura accès à ce qui est public et ne pourra pas créer de projets ou de groupes par défaut. Vous devrez ajouter les droits d'accès par la suite à ce compte pour qu'il puisse avoir accès à des projets et/ou la possibilité de créer un ou des projets ainsi que des groupes.

### **Ajout de droits d'accès à un projet :**

- Se connecter sur le serveur Gitlab avec un compte qui a les droits « Owner ou maintenir » sur un projet ou un groupe Gitlab
- Cliquer sur « Projects » dans la barre du haut puis choisir le projet pour lequel on veut ajouter des droits d'accès
- Cliquer sur « Settings » dans le menu de gauche puis sur « Members »  
(il faut que le CIP se soit déjà connecté sur Gitlab pour être visible dans la liste des usagers)
- Saisir le CIP dans le champs « Select members to invite » pour le chercher et l'ajouter dans la liste
- Une fois choisi, lui attribuer le rôle désiré et cliquer sur « add users to project »  
Voir doc. sur les rôles en cliquant sur le lien « Read more » sur « Read more about role permissions » sous « Choose a role permisison »

## Groupe pour les projets

Un projet est toujours associé à un groupe (ou sous-groupe) ou à un utilisateur (CIP) si vous avez le droit de créer un projet. La notion de groupe permet le regroupement d'un ou plusieurs projets sous un même nom et permet de gérer des permissions en fonctions de rôles associés aux utilisateurs. Elle évite aussi de vous retrouver avec des projets qui sont associés à votre nom d'utilisateur (CIP). Pour comprendre les droits d'accès aux projets et aux groupes, consulter :

[https://Nom\\_du\\_serveur\\_Gitlab.dinf.fsci.usherbrooke.ca/help/permissions/permissions](https://Nom_du_serveur_Gitlab.dinf.fsci.usherbrooke.ca/help/permissions/permissions)  
(Vous devez vous connecter avec votre CIP)

## Créer un Git sur Gitlab avec deux premiers fichiers importants (README.md et .gitignore)

(Information sur la structure d'un dépôt Git :  
[https://depot.dinf.fsci.usherbrooke.ca/help/workflow/gitlab\\_flow.md](https://depot.dinf.fsci.usherbrooke.ca/help/workflow/gitlab_flow.md)  
(vous devez vous connecter avec votre CIP)  
aussi  
<http://nvie.com/posts/a-successful-git-branching-model/>)

- Se connecter dans Gitlab
- Cliquer dans la barre du haut sur *Projects* --> *Yours projects*
  - choisir un projet existant

### **Fichiers importants** (README.md et .gitignore)

README.md

- Cliquer sur "README" pour ajouter un fichier de description du projet (README.md) dans le git
- Ce fichier contient de l'information sur le projet et le dépôt git
- Cliquer sur "Commit Changes"

.gitignore

- Cliquer sur "New file" puis dans template choisir ".gitignore".
- Choisir un modèle de fichiers d'exclusion dans le menu déroulant de droite nommé « Apply a .gitignore template »  
(un fichier contenant une liste de fichiers d'exclusions qui ne seront pas chargées dans le Git car ils sont inutiles, surchargent et alourdissent le dépôt vous pouvez par la suite le personnaliser pour vos besoins)
- cliquer sur "commit change" pour terminer.

## Création clé SSH pour pouvoir faire un clone d'un dépôt Git dans Gitlab :


**Créer votre clé SSH via le serveur Drongo (ou autre système utilisant openssh)**  
(ou consulter : <https://depot.dinf.usherbrooke.ca/help/ssh/README> )

Vous connecter sur le serveur Ubuntu (drongo.dinf.usherbrooke.ca) avec votre cip pour y générer les clés qui seront placées dans le répertoire .ssh. Par la suite vous pourrez transférer la clé privée sur le poste client (windows ou autre)

- Vous connecter sur le serveur : drongo.dinf.usherbrooke.ca en SSH
- Faire les commandes suivantes pour créer votre clé SSH :

- `ssh-keygen -b 4096 -t rsa -C "votre_courriel@usherbrooke.ca"`
  - o Il n'est pas nécessaire d'utiliser une passphrase car plus complexe à utiliser et si vous utilisez une passphrase vous devrez utiliser Pageant sur windows
- Deux fichiers seront générés dans le répertoire `.ssh`
  - `id_rsa`
  - `id_rsa.pub`
- `cd` (afin d'être bien dans votre répertoire de base "home dir")
- `cd .ssh`
- Faire afficher votre clé pour la copier dans le presse-papier
  - o `cat id_rsa.pub` (le début et la fin de la clé sont « `ssh-rsa.....votre_courriel` »)
  - o Sélectionner et copier (attention seulement prendre les caractères de la clé)

### **Copier votre clé publique (`id_rsa.pub`) dans votre compte sur le serveur Gitlab.**

- Vous connecter sur le serveur Gitlab avec votre CIP
- Cliquer le menu déroulant coin supérieur droit puis sur « Settings »
- Dans le menu de gauche Cliquer sur  « SSH Keys »
- Coller votre clé publique (`id_rsa.pub`) au complet (votre clé doit commencer par « `ssh-rsa...` » et se terminer par votre courriel au complet « `...courriel@usherbrooke.ca` » dans la partie identifiée « Key »
- Cliquer dans le carré « titre » et coller votre courriel dans le Titre
- Terminer en cliquant sur « Add key »
- Votre clé est disponible pour un être utilisée avec un logiciel Git client
- Tester votre clé à partir de Drongo :
  - o `ssh -T git@dinf-NOM_DU_SERVEUR.dinf.fsci.usherbrooke.ca`
  - o ex. : `ssh -T git@depot.dinf.fsci.usherbrooke.ca`
    - sortie qui confirme que la clé SSH fonctionne :
      - Welcome to GitLab, Votre CIP !
  - o Si on vous demande un mot de passe la clé ne fonctionne pas

### **Client Git :**

Transférer votre clé privée `id_rsa` sur votre poste de travail qui a le client Git.

**Git** : (client utilisé comme exemple : Smartgit)

La clé privée `id_rsa` générée par OpenSSH sur le serveur Drongo peut être utilisée directement par le client Git « Smartgit »

### **Vérification du nouveau dépôt Git : Faire un test de «Git clone» avec le client SmartGit :**

Smartgit (Windows ou Ubuntu)  
Lancer smartgit

Choisir « Clone existing repository »  
Ou via menu  
Repository → Clone  
Repository URL :

git@**Nom\_du\_serveur\_gitlab**.dinf.fsci.usherbrooke.ca:CIP\_ou\_groupe/**nouv-depot**.git  
(consulter l'URL pour SSH de votre Git dans Gitlab)

#### Next

#### Accepter la clé SSH du serveur

SSH Credentials :


Authentication Type : Private Key

Vous pointer sur votre clé id\_dsa transférée sur  
votre poste de travail

Cliquer sur Login

Choisir l'endroit où vous désirez créer le Git localement

#### Note pour voir les fichiers du dépôt GIT

Cliquer sur l'icône  à droite de l'icône  en haut à droite de l'écran de SmartGit

#### Ajouter un fichier dans le dépôt GIT :

- Créer un fichier dans le même répertoire où il y a «.git »
- Revenir dans l'application et faire un refresh dans la section « files »
  - o Cliquer sur le fichier :
    - Stage
    - Commit et Entrer un commentaire bidon
    - Commit & Push