

Programmation et Applications Interactives

Utilisation de GitLab pour partager les fichiers de votre projet

Git est un outil open source de contrôle de versions. Il permet de :

- conserver tout l'historique des modifications des codes sources et documents,
- travailler en local (mode déconnecté) : continuer à enregistrer des versions successives même sans réseau,
- travailler depuis plusieurs ordinateurs,
- travailler avec plusieurs personnes,
- se resynchroniser à tout moment de façon maîtrisée et cohérente,
- donc travailler depuis n'importe où,
- naviguer de façon claire et déterministe dans l'ensemble des versions,
- travailler directement depuis un IDE, par exemple VisualStudioCode.

L'objectif de ce TD est de mettre en place GitLab pour que vous puissiez partager les fichiers issus de votre travail lors des projets en groupe. Il vous servira tout d'abord pour le projet HTML, puis le projet Python et enfin tous les autres projets en groupe durant votre licence, quelle que soit la discipline. Nous allons procéder par étapes.

1. Activation des comptes gitlab du CREMI pour tous les membres du projet

Rendez vous sur <https://gitlab.emi.u-bordeaux.fr/> et connectez vous avec votre identifiant de l'ENT.

2. Création d'un projet sur le gitlab du CREMI par **UN membre du projet**

1. Connectez vous sur <https://gitlab.emi.u-bordeaux.fr/>.
2. Cliquez sur l'icône bleu « New project » en haut à droite.
3. Cliquez ensuite sur « Create blank project ».
4. Dans « Project name », choisissez un nom pour votre projet (attention, prenez un nom réellement en rapport avec le sujet choisi).
5. Pour « Visibility Level », choisissez « Private » (à moins que vous n'ayez envie que toute la promotion puisse accéder à votre travail).
6. Validez en cliquant sur le bouton bleu « Create project ».

Il faut maintenant ajouter les deux autres membres du groupe à ce projet. Pour cela :

1. Allez sur le menu en haut à gauche « Project information » et cliquez sur « Members ».
2. Cliquez sur le bouton bleu en haut à droite « Invite members ».
3. Dans la fenêtre qui s'ouvre :
 1. Entrez l'identifiant ENT ou le mail du membre. Quand vous cliquez sur ce champs, un menu déroulant vous propose des noms de membres,
 2. Choisissez le rôle Maintainer,
 3. Cliquez sur le bouton bleu « Invite ».

Vous pouvez voir le membre ajouté dans votre fenêtre et modifier son rôle si besoin, voire le supprimer en cas d'erreur.

3. Création du jeton d'accès / access token

Une fois cette étape faite, chaque membre doit paramétrer depuis son propre compte son accès au projet. Plusieurs méthodes sont possibles, clés SSH (SSH Keys), jetons d'accès (Access Tokens), ... Nous allons utiliser les « Access Tokens ». Pour ce faire :

1. Cliquez sur l'icône en haut à droite et choisissez dans le menu qui apparaît « Edit profile ».
2. Cliquez dans la barre latérale gauche sur AccessToken.
3. Choisissez un nom (p.ex. Vscodetoken, car nous allons l'utiliser pour l'éditeur VisualStudioCode), cochez toutes les cases, et cliquez sur « Create personal access token ».
4. **ATTENTION, TRÈS IMPORTANT** : cliquez sur l'œil à droite du champ « Your new personal access token » et copiez/collez le token dans un fichier texte nommé « mon_token.txt » (utiliser un éditeur de texte, **PAS** un traitement de texte) . Vous aurez besoin dans la prochaine étape, si vous ne le faites pas, il faudra recommencer la procédure de création.

4. Construction de l'URL pour cloner votre projet

Nous allons maintenant créer un lien web (URL) pour cloner le projet dans votre espace de travail (au CREMI ou sur votre ordinateur personnel) avec VisualStudioCode.

1. Retournez sur ma page d'accueil (icône U/CREMI en haut à gauche ou <https://gitlab.emi.u-bordeaux.fr/>), puis cliquez sur votre projet.
2. Cliquez sur le bouton bleu "Clone" à droite.
3. Copiez/Collez le lien du champs "Clone with HTTPS" dans le fichier « mon_token.txt » votre presse papier (icône « Copy URL »). Il est de type `https://gitlab.emi.u-bordeaux.fr/votrelogin/nom-de-votre-projet`.
4. A partir de ce lien et de votre token, construisez un lien incluant votre access token, sous la forme

`https://votrelogin:votretoken@gitlab.emi.u-bordeaux.fr/votrelogin/nom-du-projet.git` **si vous êtes le créateur ou la créatrice** du dépôt du projet sur gitLab.

ou

`https://votrelogin:votretoken@gitlab.emi.u-bordeaux.fr/login-du-créateur/nom-du-projet.git` si vous avez été invité(e).

5. Ce lien va nous servir dans VisualStudioCode à cloner votre projet.

5. Installation de Git

Si vous travaillez sur votre ordinateur personnel, vous devez maintenant installer et configurer Git.

1. Allez sur <https://git-scm.com/downloads>.
2. Choisissez votre système et téléchargez la dernière version de l'installateur de Git
 1. sous Windows, choisissez Standalone Installer, [64-bit Git for Windows Setup](#).
 2. sous macOS, choisissez l'installation avec [MacPorts](#) et tapez dans votre terminal la commande indiquée sans le \$: `sudo port install git` .
3. Installez Git.
 1. Modifier l'éditeur par défaut de "Vim" à "VS Code".
 2. Pour le choix de la console, mettre "Use Window's default console window"
4. Dans un terminal (linux ou mac) ou une invite de commande (windows), exécutez les deux instructions suivantes en mettant votre nom et email :
 1. `git config --global user.name "Prénom Nom"`
 2. `git config --global user.email prenom.nom@emi.u-bordeaux.fr`
 3. `git config --list` (permet de vérifier que tout s'est bien passé)

6. Clonage du dépôt GitLab

Vous pouvez maintenant cloner votre projet du GitLab du CREMI sur votre ordinateur ou votre espace de travail avec VisualStudioCode.

1. Ouvrez une nouvelle fenêtre VisualStudioCode.
2. Dans la barre de gauche, cliquez sur l'icône « Contrôle de code source ».
3. Cliquez sur le bouton bleu « Gérer l'approbation d'espace de travail », puis sur le bouton bleu « Faire confiance ».
4. Cliquez sur le bouton bleu « Cloner le dépôt » ou « Clone github repository » en anglais.
5. Gérez l'« approbation d'espace de travail ».
6. Dans la barre de commande qui vient d'apparaître en haut au milieu de l'écran, copiez votre URL au dessus du texte bleu « Cloner à partir de GitHub ».
7. Sélectionner le dossier sur votre ordinateur ou dans votre espace de travail où vous allez stocker votre projet. Pour rester cohérent avec les trois premiers TD web, je vous conseille de choisir le dossier web/projet.
8. Le clonage se met en route.
9. Une fois terminé, VisualStudioCode vous propose d'ouvrir le dépôt cloné.
10. Dites oui, cochez la case « Faire confiance à tous les auteurs du dossier parent projet » et cliquez sur le bouton bleu « Oui, je fais confiance aux auteurs ».
11. Découvrez les fichiers reçus. Il n'y en a normalement qu'un, nommé « README.md ».

7. Synchronisation de vos modifications

A partir de ce moment là, vous pouvez ajouter des fichiers à votre dossier, les modifier et une fois le travail terminé, les envoyer sur le projet GitLab du CREMI. On dit faire un Commit – push.

Pour tester ceci :

1. Créez un fichier txt qui porte votre nom ou votre login, dans lequel vous mettez votre nom, votre prénom et votre mail. Sauvegardez le fichier (**OBLIGATOIRE**).
2. Cliquez sur « Contrôle du code source ».
3. Dans la fenêtre « Contrôle du code source », pour chaque document dans la section : changement, appuyez sur le « + » pour indexer le changement. Cela évite que votre version du fichier n'écrase les modifications faites par les autres. Cela n'a pas d'importance pour le moment, mais cela en aura plus tard quand vous travaillerez en équipe sur les mêmes fichiers.
4. Dans l'encart au dessus du bouton bleu validation, tapez un message qui décrit les modifications faites (**OBLIGATOIRE**).
5. Cliquez sur le bouton bleu « Validation » vous pouvez aussi passer par le menu "...", sélectionnez Valider → Valider (commit → commit)).
6. Cliquez sur « Synchroniser les modifications ».
7. Une fois que tous les membres du groupe ont fait cette partie, faites un « pull » ("...", sélectionnez Tirer (pull)) pour récupérer les fichiers créés par les autres.
8. Ensuite, modifiez un fichier qui n'est pas le vôtre, et refaites un commit. Vous pouvez contrôler sur GitLab que la modification du fichier est bien prise en compte.
9. Faites à nouveau un « pull » pour vérifier les mises à jour en local.

VisualStudioCode vous permet de suivre vos modifications :

1. Modifiez à nouveau votre le fichier en ajoutant votre date de naissance et sauvegardez.
2. Vous pouvez voir les changements apportés au fichier en cliquant à gauche sur « Changement », puis en cliquant sur le nom du fichier.