

Laboratoires de développement WEB I

GitLab de l'ÉSI
Utilisation avec Brackets

Résumé

Les projets que vous allez développer en WEB I devront utiliser le GITLAB de l'école comme vous avez appris à le faire dans l'UE DEV2. Dans ce document, nous faisons un petit rappel des concepts liés à GIT et nous vous montrons en pratique comment l'utiliser avec BRACKETS.

1 Un petit rappel

Pour commencer, refaisons bien la différence entre GIT et GITLAB.

- **Git** est un système de gestion de version. Il fonctionne en mode commande. Certaines applications (dont BRACKETS via une extension) proposent une interface graphique pour GIT.
- **Gitlab-ési** est un serveur géré par l'école sur lequel nous allons pouvoir sauvegarder notre projet GIT. Il est disponible à l'adresse : <https://git.esi-bru.be>.

Lors de l'utilisation de GIT, le développeur passe par plusieurs phases que nous rappelons ici.

- **init** : Commencer à faire suivre un projet par le système GIT.
- **clone** : Prendre une copie d'un projet GIT sur le serveur pour le mettre sur la machine locale où il n'existait pas encore.
- **pull** : Mettre à jour une copie locale du projet à partir d'une version plus récente sur le serveur.
- **commit** : Acter en local des changements apportés au projet. Un commentaire accompagne le *commit* pour décrire ces changements. Rappelez-vous que sauvegarder un fichier ne le commit pas automatiquement. Il s'agit d'explicitement faire une photographie de tout votre projet à un moment donné.
- **push** : Lorsque des *commit* ont été réalisés, le *push* répercute les changements sur le serveur. Rappelez-vous que le push n'envoie sur le serveur que ce qui a été commité.

L'extension de BRACKETS que nous allons utiliser va nous permettre de réaliser les opérations ci-dessus via des boutons ou en tapant les commandes correspondantes dans un terminal dédié.



2 Installer Git dans Brackets


Pour pouvoir utiliser facilement GIT avec BRACKETS il suffit d'installer la bonne extension :

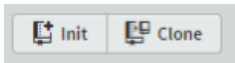
1. Lancez BRACKETS ;
2. Lancez le gestionnaire d'extensions : **file/Extension Manager...** ;
3. Demandez à trier les extensions par le nombre de downloads ;
4. L'extension **Brackets Git** apparait dans les premières ;



5. Cliquez sur son bouton d'installation (l'extension se télécharge) ;
6. Une fenêtre de configuration vous est proposée, cliquez directement sur **Save** ;
7. Il vous est proposé de redémarrer BRACKETS, acceptez.

3 Lancer Git

Dès lors, l'icône  apparait à droite. Elle permet d'ouvrir une fenêtre offrant des boutons pour les différentes étapes du développement GIT.

Lorsque le projet n'est pas encore suivi, on a : 

Si le projet est déjà suivi, on a :

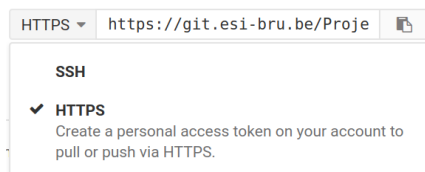


L'icône  permet de taper les commandes directement dans un terminal dédié.

4 Création du projet sur gitlab

Dans le cadre de cette école, les projets sont créés sur le serveur par le professeur.

1. Demandez à votre professeur de vous créer un projet sur le serveur.
2. Vous recevrez alors un mail vous menant à ce projet sur le gitlab de l'ÉSI.
3. Repérez où trouver l'adresse de référence de ce projet. Elle est de la forme **https://git.esi-bru.be/[...].git**



Vous en aurez besoin lorsque vous allez *cloner* le projet. Attention, il faudra prendre la version **HTTPS** et pas **SSH**.

5 Utilisation dans le cadre du projet

Dans le cadre des projets de WEB I, voici comment procéder, étape par étape.

1. Lorsque vous commencer à travailler sur une machine, il y a 3 cas :
 - (a) Si le projet existe déjà sur la machine (parce que vous y avez déjà travaillé), vous allez faire un *pull* (cf. 5.2).
 - (b) Si vous travaillez pour la première fois sur cette machine mais que le projet a déjà été créé sur le serveur, vous allez le cloner (cf. 5.1).
 - (c) Si vous voulez déjà commencer à travailler avant que le projet ne soit créé sur le serveur, vous allez faire un *init* plutôt (cf. 5.3 page suivante).
2. Apporter des modifications au code et commiter les changements (cf. 5.4 page suivante).
3. De temps en temps et certainement lorsque vous quittez votre machine, pusher votre projet (cf. 5.5 page suivante).

5.1 Cloner le projet sur une machine

Votre projet est créé sur le serveur. Voyons comment en faire une copie sur une machine de l'école ou chez vous.

Remarque : Si vous rencontrez un message d'erreur à ce stade, nous vous proposons une solution dans la section 6 page 5.

1. Lancez BRACKETS et placez-vous dans un dossier **vide** (File/Open Folder...).
2. Cliquez sur l'icône de l'extension GIT afin d'ouvrir la fenêtre.
3. Cliquez sur le bouton **Clone**.
4. Dans la boîte de dialogue **Clone repository**, entrez
 - L'URL du projet notée au point 3 de la section 4 page précédente ;
 - Votre *login* sur gitlab (à priori votre numéro d'étudiant) ;
 - Votre *mot de passe* sur gitlab (vous en avez défini un en DEV2) ;

Votre projet est à présent disponible sur la machine et vous pouvez y travailler comme indiqué à la section 5.4 page suivante.

5.2 Continuer un projet

Si vous revenez sur une machine qui possède déjà une (vienne) version du projet, vous pouvez la mettre à jour grâce à un *pull*.

- Ouvrez votre projet dans BRACKETS.
- Cliquez sur l'icône *pull* 
- Entrez votre login et mot de passe et cliquez sur OK.

Votre projet est à présent à jour et vous pouvez y travailler comme indiqué à la section 5.4 page suivante.

5.3 Initier un projet en local

Si vous voulez déjà commencer un projet alors qu'il n'est pas encore disponible sur le serveur, vous devez passer par un *init*.

Remarque : Si vous rencontrez un message d'erreur à ce stade, nous vous proposons une solution dans la section [6 page suivante](#).

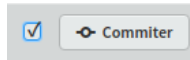
1. Lancez BRACKETS et placez-vous dans un dossier **vide** (**File/Open Folder...**).
2. Cliquez sur l'icône de l'extension GIT afin d'ouvrir la fenêtre.
3. Cliquez sur le bouton **Init**.

5.4 Commiter un projet

Votre projet existe sur la machine. et vous pouvez y travailler. Lorsque vous avez terminé une étape, il est important de la commiter (c-à-d la sauver en local).

1. Assurez-vous que tous les fichiers soient sauvés (**File/Save All...**).
2. Cliquez sur la case à cocher à gauche du bouton **Commit** afin de sélectionner

tous les fichiers et cliquez sur le bouton **Commit**.




3. Dans la fenêtre qui apparaît, introduisez un message décrivant ce que vous venez de faire dans votre projet et cliquez sur **OK**.

De temps en temps et certainement lorsque vous quittez votre machine, sauvez votre projet sur le serveur (cf. [5.5](#)).

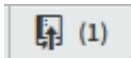
5.5 Sauver un projet sur le serveur

Les *commits* se font en local. Avant de quitter votre PC, il est important de sauver le tout sur le serveur.

Si vous n'êtes jamais passé par la phase *clone* sur cette machine (parce que vous avez fait un *init*), il faut d'abord lier votre projet au serveur. Pour cela,

1. Cliquez sur le bouton  et choisissez **Créer un nouveau distant...**
2. À la première question, entrez **origin**.
3. À la seconde question, entrez l'URL du projet notée au point [3](#) de la section [4 page 2](#).

Ensuite :

1. Cliquez sur le bouton de push . Remarquez qu'il indique le nombre de *commits* qui ne sont pas encore sauvés sur le serveur.
2. Entrez votre login et mot de passe et cliquez sur **OK**.

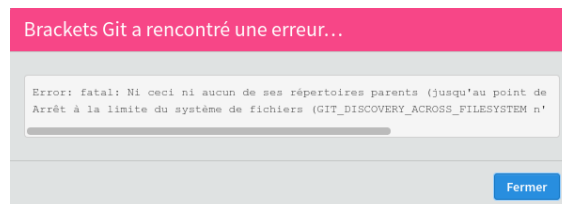
3. Par sécurité, vous pouvez vous connecter au serveur pour constater que le(s) commit(s) a (ont) bien été envoyé(s).

Attention ! Avant de quitter une machine veuillez à bien tout pusher sur le serveur sinon il vous manquera des choses quand vous voudrez le continuer sur une **autre** machine.


6 Bug dans l'extension

Lors de nos tests, nous avons remarqué un bug avec cette extension. Lorsqu'elle est installée, elle génère une erreur si on travaille avec un projet non suivi par git... ce qui est très embêtant lorsqu'on commence un nouveau projet.


Si vous rencontrez ce message d'erreur



nous vous proposons de le régler en utilisant le mode commande de git.

1. Ouvrez un terminal via l'icône  de l'extension.
2. Pour cloner le projet
 - (a) Entrez la commande : `git clone https://git.esi-bru.be/[...].git` ;
 - (b) Donnez votre login et mot de passe ;
 - (c) Entrez `exit` pour fermer la console.
3. Pour faire un init à la place, entrez la commande : `git init`.
4. Il vous faudra probablement ré-ouvrir le projet pour qu'il en tienne compte et vous pouvez alors commencer à travailler sur votre projet comme indiqué à la section [5.4 page précédente](#).

7 Le mode commande

Si vous préférez effectuer les opérations en mode commande via le terminal dédié () voici un résumé des commandes les plus fréquentes.

- `git clone https://git.esi-bru.be/[...].git` : pour cloner un projet ;
- `git init` : pour initier un projet local ;
- `git pull` : pour mettre à jour un projet existant ;
- `git commit -am "message"` : pour commiter toutes les modifications ;
- `git push` : pour sauver sur le serveur les derniers commits ;