

# Petit Tutoriel Github

## A propos de Github

Git est un système de gestion de versions décentralisé. Github est une des possibilités d'hébergement de votre dépôt local en ligne. Git réalise des instantanés des répertoires et fichiers contenus dans votre dépôt local. Les différents collaborateurs d'un projet ont sur leur ordinateur la version complète du contenu du dépôt en ligne et peuvent réaliser leurs modifications avant de l'envoyer sur le serveur github où ils possèdent un compte. Dans notre cas notre cas, le projet est sur : [https ://github.com/tuxtrip/](https://github.com/tuxtrip/)

## Configuration de git

Utilisateurs Windows : Télécharger git et l'installer

1. Créer un compte sur github et envoyer votre nom d'utilisateur par mail.
2. lancer le shell de l'application git que vous avez téléchargé et installé sur votre ordinateur
3. Configuration des paramètres globaux de votre dépôt local Git  
Tapez :
  - **git config - -global user.name "votre nom d'utilisateur github"**
  - **git config - -global user.email 'votre adresse email'**(Ne pas oublier les guillemets. Il n'y a pas d'espace entre les deux tirets, ils sont collés)
4. Créez votre répertoire local et télécharger le dépôt github sur votre ordinateur en tapant la commande :  
**git clone https ://github.com/tuxtrip/plateformetouristique**  
Le dépôt github du projet est disponible sur votre ordinateur.

## Quelques précisions à propos du dépôt plateforme touristique

Il possède 3 branches :

- La branche master qui contient les informations générales sur le projet
- La branche appli-web, le répertoire de travail sur l'application Web

- La branche appli-mobile, le répertoire de travail sur l'application mobile

Pour vous positionner dans le bon sous projet il faut taper la commande :

**git checkout master** ou **git checkout appli-web** ou **git checkout appli-mobile** selon votre choix.

Une fois positionné dans le bon sous projet vous pouvez commencer à travailler.

## Généralités sur l'usage de git

Git possède 3 niveaux :

- Le niveau des fichiers non indexés
- Celui des fichiers Indexés
- Celui des fichiers validés et prêts à l'envoi c'est à dire indexé et « commité ».

Les fichiers et répertoires présents dans votre espace de travail après clonage du dépôt « plateformetouristique » sont des fichiers indexés. Ils consistent l'image (snapshot) du dépôt initial. Si vous modifiez ces fichiers il faudra les ajouter à l'index avant de les commiter et de les uploader sur [github.com/tuxtrip/](https://github.com/tuxtrip/).

Pour cela après modification d'un fichier quelconque montravail.txt par exemple, il faut taper :

**git add montravail.txt**

afin de réindexer le fichier sachant que dès qu'il a été modifié il ne fait plus partie de l'index (seule la version initiale de ce fichier est conservée dans l'image du dépôt et peut être restaurée en cas de besoin).

Vous pouvez vérifier le statut du dépôt git et l'état des fichiers avec la commande :

**git status**

Une fois le fichier réindexé, Vous pouvez le commiter avec la commande :

**git commit -m "Voici la modification faite sur mon le fichier montravail.txt"**

Mettre à jour le dépôt online [github.com/tuxtrip/plateformetouristique](https://github.com/tuxtrip/plateformetouristique) en faisant un :

**git pull**

La commande permet de récupérer une version récente du dépôt si jamais quelqu'un a entre-temps envoyé des données.

Ensuite taper :

**git push - -set-upstream origin nomdelabranche**

Le nom de la branche étant celui du sous projet (master, appli-web ou appli-mobile sur lequel vous avez opéré vos modifications).

Git vous demande d'introduire le login et le mot de passe de votre propre compte github. (Ne pas utiliser le login et le mot de passe tuxtrip)

Si vous voulez voir les dernières modifications faites sur le dépôt online vous pouvez utiliser la commande

**git log - -decorate**

avec ses différentes options.

Pour mieux visualiser les modifications vous pouvez lire la commande **git diff** (voir la doc).

Si vous voulez créer votre propre branche vous pouvez utiliser la commande :

**git branch nomdevotrebranche**

Pour effacer une branche :

**git push origin - -delete nomdelabranche**

Le reste de la procédure est identique à ce qui a été décrit précédemment.

## Ajouter un répertoire

Pour ajouter un répertoire il faut créer un fichier dans ce répertoire sinon ça peut ne pas marcher.

Exemple : Ajout du répertoire MonNouveauRepertoire/

Vous créez dans ce répertoire le fichier test.txt par exemple ou vous y mettez le fichier de votre choix. et vous tapez :

**git add MonNouveauRepertoire/test.txt**

si ce répertoire contient plusieurs fichiers :

**git add MonNouveauRepertoire/\***

Ensuite faire le commit, le pull et le push comme indiqué précédemment.

## Effacer un fichier ou un répertoire

**git rm nomdufichieraeffacer**

Pour un répertoire

**git rm -r nomdurepertoireaeffacer**

La commande git rm efface le fichier et la désindexe en même temps. Si vous avez effacé le fichier avec la souris, il faudra indexer cet effacement en retapant :

**git rm nomdufichieraeffacer**

ou

**git rm -r nomdurepertoireaeffacer**

Faire le commit, le pull et le push comme indiqué précédemment.

Pour aller plus loin, La documentation git est présente dans le dépôt master.

**Merci à vous, Bon travail**