**CONFIGURATION DE L’UTILISATEUR**

> git config –global user.name ‘‘nom’’

> git config –global user.email ‘‘mon@email.com’’

nano monfichier.txt

git status : rouge pas suivi par git 🡪 git add monfichier.txt 🡪 git status (monfichier.txt sera vert donc suivi par git)

> git status

> git add monfichier.txt

> git status

> touch empty.txt //création fichier vide dont on ne spécifie que le nom

> git status

**AJOUT DE FICHIERS A GIT**

git add \*.txt //tous les fichiers .txt

git add \* //tout ⬄ git add .

git add –A //tout + les fichiers supprimés !

git add –A #git add .

**COMMIT (après avoir add nos fichiers) = ensemble des fichiers qui sont suivis par git**

> git commit –m « mon message » //-m pour message

> git status => plus de modification à prendre en compte = CLEAN

**HISTORIQUE DES COMMIT**

> git log

> git diff

**COLLABORATION DE CODE**

gitlab : settings -> members -> mettre sur

Sur git : cd nomDossierBinome

nano fichier\_que\_je\_veux\_modifier

git commit –m « message descriptif pour le binome »

git add fichier

git push origin suruneBranche

**PARTAGE DE CODE**

GITLAB   
new project

<https://gitlab.utc.fr/chengaur/formationGit>

Synchronisation code local au serveur

remote lien entre le dossier du serveur et ma machine : SSH ou **HTTPS**

origin

gitlab

variant

machine

Github

remote

Forges

1. Ajouter la REMOTE

> git remote add origin <https://gitlab.utc.fr/chengaur/formationGit>

2) Lier le PUSH et la REMOTE : push déplace le commit (ce qu’il y a en local) sur le serveur

> Git push origin master //notre dossier est push dans gitla

**BRANCHE**

> Git branch newBranch //crée une nouvelle branche

> git checkout newBranch //pour changer de branche

>

**CLONE**

Git clone <https://.....git> //on clone le dossier du binome sur notre machine