**GIT**

**Pourquoi** : Garder trace des modifications de code, enregistrées dans un commit et intéressant de pouvoir revenir sur l’historique des commits en cas de bug

2 types de logiciels de gestion de version : centralisé (serveur central contrôlant le code) et distribué (accès au code pour toutes les machines, ex : git, avantages : moins de risque de perte car plusieurs machines et possible de développer sans accès à un serveur central inutile)

**Installation :** Installer et configurer login et adresse mail puis sélectionner éditeur texte (Vim)

**Utilisation :** créer un repository : dossier de travail (créer dossier et activer git sous ce dossier par la commande **git init** qui créé un index des fichiers à gérer.

**Commandes :**

* Ajout fichier : **git add nom\_fichier** ou **git add .** (tous les fichiers dans le repository)
* Commit : **git commit –m « message »**
* Historique : **git log** pour avoir l’historique répertorié avec SHA, auteur, date, description
* Mise à jour : **git commit –a –m « message »**
* Navigation dans l’historique : **git checkout SHACommit** ou **master** (commit le + récent)
* Annulation : autre commit pour annuler les actions du précédent **git revert SHADuCommit,** modification message **git commit –amend –m « nouveau message »,** annulation de tous les changements non commités : **git reset --hard‌**

**Remotes :**

Sauvegarde du code commité sur une autre machine (en interne ou externe (via Github donc tout le monde y aura accès et plus facile de travailler encore))

**GitHub :**

Communiquer sur problèmes de code, partager code, proposer modifications code, récupérer code via pull requests, récupérer code via autre repository.

Récupération d’un repository : **git clone lienFourniParGitHub**

Création repository : aller sur Github et faire **Create new repository** (option **initialise with a readme :** clone de repository si pas repository créé sur machine locale, liste des versions affichée

Push / synchronisation :

* modifications locales avec code sur Github : dans le repo local faire un commit et un push : **git push origin master** (envoyer dans remote origin (Github) en ligne de la branche master contenant le code local courant), possible de voir l’historique graphiquement sur Github
* modifications de remote Github vers local : **git pull origin master** dans l’autre sens

Exercice