**Note Git:**

**L’idéale serait que nous lisions tous le cours git Openclassroom:** <https://openclassrooms.com/courses/gerer-son-code-avec-git-et-github/creer-son-premier-repository>

**Mais sinon:**

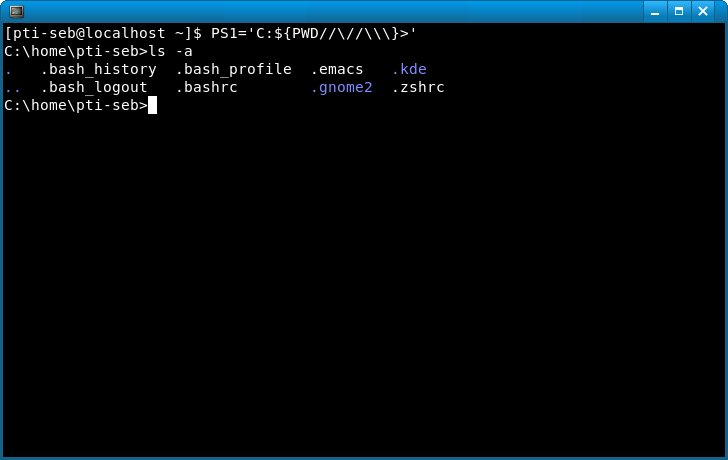
**Git :** Est un logiciel de gestion de versions qui permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectuées dessus.

**GitHub:** Est un service web d'hébergement en ligne et de gestion de développement de logiciels.

Dans le cadre de notre projet ces deux outils combiné vont nous permettre d’héberger notre application sur La plateforme en ligne github et de le versionner en utilisant git dans l’optique de travailler en groupe sur notre projet tout en gardant l’historique de l’évolution des différents fichiers de notre application nous pourrons donc en cas de problème récupérer une version précédente de celle-ci.:

Git s’utilisera en ligne de commandes (cli) donc dans une console comme l'invite de commande windows.

exemple:



Les pc d’aston étant sous windows le plus simple serait de télécharger et d’installer une la console git bash puis de l’ouvrir via le moteur de recherche window.

ll sera donc nécessaire de télécharger et d’installer git: <https://git-for-windows.github.io/> pour avoir accès la console git bash et de d’ouvrire via le moteur de recherche windows **la console Git bash.**

**Les lignes de commandes (cli)**: Dans une console l'utilisateur communique directement avec la machine par l'intermédiaire de commandes que celle -ci comprend.

les commandes principale pour naviguer dans l'arborescence de fichier d’un poste de travail (l'équivalent du DOM en JavaScript) dans une console linux sont:

* **ls**: list tous les fichier du répertoir courant.
* **pwd** : afficher le chemin(path) relatif au répertoire actuel.
* **cd**: permet de se déplacer dans un répertoire ou un fichier de répertoire courant exemple : cd mondemonrepertoire. (cd sans paramètres permet de revenir à la racine de votre arborescence et cd .. permet de se déplacer ver le répertoir parent ).
* **mv**: permet de déplacer un fichier ou de renommer un fichier.
* **rm**: permet de supprimer un fichier.(à utiliser avec précaution).

**Ces commandes sont utilisable avec des options pour plus d'information voir documentation ou tapez le nom de la commande puis help dans la console**.

**Les commandes git:** pour utiliser git nous utiliserons des commandes comme les précédentes.

voici les commandes principales à connaître pour utiliser git.

**git init** : transforme le répertoir courant en dépôt git.

**git clone**: sert à récupérer un dépôt github héberger ailleurs que sur votre machine local.

**git remote add** : ajoute un dépôt git à la liste des dépôts courant disponible dans le répertoire git dans lequel vous vous trouvez (consultable avec la cmd git remote -v ) sur lequel vous pouvez récupérer (fetch) et envoyer (push) du contenue.

**git push**: Permet d’envoyer les fichier modifier qui ont été add et commit dans le répertoir distant mentionner en option.

**git remote rename**: permet de modifier le nom d’un répertoir.

**git add:** permet d’envoyer des fichier dans l’air d’envoi (l’endroit ou ce situe les fichiers modifiés avant d'être envoyer sur le dépôt distant )

**git stash:** Permet de mettre de côté un fichier modifier que je ne veux pas pas push

**git stash pop:** permet de récupérer les modifications mise en mémoire avec la commande git stash.

**git commit:** permet d'ajouter un commentaire aux prochains push qui contiendra tous les fichier modifier.

**git pull:** permet de récupérer une version à jour du master ( permet d’éviter certains conflits lorsqu’elle est effectuée avant de travailler sur une nouvelle branche.)

**git branch:** affiche la branch dans laquelle on se trouve.

**git status**: Nous permet de connaître tous les fichiers modifiés dans le répertoir git courant.

**git checkout -b** + nom de la branch : crée une nouvelle branche.

**Sources:**

Doc open classroom git/github : <https://openclassrooms.com/courses/gerer-son-code-avec-git-et-github/creer-son-premier-repository>

https://www.google.fr/

le mode invite de commande:

http://www.courstechinfo.be/OS/CmdDos.html