

## GIT HUB

Contrôle de version : Le contrôle de version, également appelé contrôle de source, désigne la pratique consistant à suivre et à gérer les changements apportés au code d'un logiciel. Les systèmes de contrôle de version sont des outils logiciels qui permettent aux équipes de développement de gérer les changements apportés au code source au fil du temps.

Décentralisation assure une garde d'information grâce à leur clients même si un serveur tombe ou plusieurs clients.

L'utilisation d'un drive de type « ONE DRIVE » n'est pas une valeur sûre !

Markdown :

## RIPO

Git enregistre à chaque version sans supprimer pour retrouver des versions précédentes.

Utilisé souvent « git status » pour savoir sur quelle branche nous sommes et pour en ajouter.

« Git commit -m '' » (M pour message)

Utilisé souvent « commit » pour faire un snapshot est bien faire un msg très explicite .

Git n'a pas besoin d'internet sauf pour partager avec ses collègues.

Checkout est pour récupérer le projet qui a été fait.

« remote »

« origin » c'est un alias

« git push -u origin »

« rm -rf git/\* » supprime la totalité du .git

« git commit -a » ca fait ADD plus commit (Déconseillé)

« Git rm —cached text.txt »

« Git log » liste tout les commit

Head est un pointeur sur le « commit » le plus récent sur le code qu'on travail

« Git mv » sert a changer de nom de dossier

La création d'une branche est une copie d'un projet a partir d'un commit sans modifié le projet de base .

Une branche est pointeur vers un dossier.

«git checkout 'nom de la branche' » pour changer de branche

« git marge »

« git branch -d » pour supprimer une branche

« git pull » pour récupérer le travail fait