-> github.com

(limité en taille dans la version gratuite)

Créer un compte github
Créer un repository sur ce compte.
exemple *repos*

repository: répertoire pour sauvegarder une application par exemple.

En local:

git clone https://github.com/jam29/repos.git

clone un repo distant

git status

état

1ere fois, ajoute tous les fichiers dans la zone STAGE.

git add .
Crée une sauvegarde du répertoire en entier

git add -p
ajout des modifs avec vérification

git checkout -p
discard des modifs (genre console.log()…etc…)

git commit -m "*Ajout du titre* **"**
création d'une version dans la branche avec message explicatif

git branch
liste les branches locales

git branch *new_branch*
crée une branche *new_branch*

git checkout *branch_name*
se place dans la branche *branch_name*

Fusionner les branches : intégration de toutes les branches sur la même arborescence. Création d'un commit qui aura 2 parents
On se place sur la branche qui va recevoir les modifications d'une autre branche (en général master mais pas obligatoire)

git checkout master
git merge *the_branche* -m "merge avec the branch"

et on peut supprimer la branche qu'on a "mergé"

git branch -d *the_branch*
supprime la branche *the_branch*

ATTENTION !!! Il peut y avoir conflit sur des fichier, dans ce cas git interrompt le merge et insère des marqueurs dans les fichiers conflictuels. Il faudra intervenir manuellement sur ces fichiers et relancer le merge.

git diff
affiche les différences sur les fichiers conflictuels

git push

met à jour une branche distante à partir de la locale sur laquelle on est situé

demandera le nom du répertoire github ainsi que le mot de passe
pour éviter de le saisir à chaque push
git config --global credential.helper cache

git pull

met à jour une branche locale à partir d'une branche distante

git pull revient à effectuer les commandes:
git fetch
git merge

Ajouter un branche locale sur le serveur distant:

git commit -m '…'
git push --set-upstream origin laBrancheAAjouter

lister les branches distantes
git branch -r

récupérer un branche distante
git fetch origin brancheDistante

fusionner une branche distante sur la branche locale
git checkout master
git merge origin/aRemoteBranch

à l'opposé
git checkout -b myBranch origin/aBranch
git merge anotherLocalBranch

git stash

Sauvegarde provisoirement les modifications sans avoir besoin de faire un commit.

(*git stash* effectue un *git stash push* par défaut)

git stash pop

retour à l'état initial

Il y a possibilité de créer plusieurs git stash en les nommant.

exemple:
git stash save "WIP ajout d'un menu"
git stash save "WIP remplacement des callbacks imbriqués par des promises"

git log

Affiche tous les commits.

commit fcd84ab4a88a0cf4d0d2f2947bec31c92c79e19d
Author: jam openlab <jb.cavarec@orange.fr>
Date: Wed Apr 18 16:23:08 2018 +0200

stocks ok

commit ddd1af95bbf91bb94d8160e409d6406a31ca9467
Author: jam openlab <jb.cavarec@orange.fr>
Date: Mon Apr 16 15:24:37 2018 +0200

modif label"

commit 5e881ecaa3de458adce22a4f08e2134f4cfba85c
Author: jam openlab <jb.cavarec@orange.fr>
Date: Mon Apr 9 15:51:32 2018 +0200

saisie qtes stocks avancés avec declinaison en positif et negatif

commit ef45fe3d095b730b88c843f94f652f96089a0516
Author: jam openlab <jb.cavarec@orange.fr>
Date: Thu Apr 5 14:21:48 2018 +0200

entrepots en colonne

commit 0db782fae5feba9b7cde1a10f77de3fdb613eb5a
Author: jam openlab <jb.cavarec@orange.fr>
Date: Wed Apr 4 16:58:36 2018 +0200

saisie qte par entrepot

commit f8b15d43b1abbb8a9f5ff0595693ffc23fd13b9e
Author: jam openlab <jb.cavarec@orange.fr>
Date: Tue Apr 3 10:40:03 2018 +0200

depot par entrepot en mode select

commit 13cbbd27f78f6df5a95c5e76b4623ef2693336df
Author: jam openlab <jb.cavarec@orange.fr>
Date: Wed Mar 28 17:14:22 2018 +0200

affichage des entrepots pour 1 article avancé

git checkout *id_commit*

Se détache du commit courant pour aller sur le commit *id_commit*

Pour revenir à l'entête

git checkout *branche*