

# Workflow conseillé pour travailler avec Git Github pour le projet 1

Ce workflow permet de simplifier le travail en collaboration en évitant de passer par des pull request.

## À l'initialisation du projet

- Un seul élève crée un repo Git et Github et ajoute la structure du projet
- Pousser les modifications initiales sur master
- Créer la branche develop en local et la pousser
- Dans github, aller dans settings / collaborator et ajouter les 2 ou 3 autres élèves de votre équipe (pour leur permettre de faire push quand besoin)
- Tout le monde clone le repo en local

## Au quotidien

- Depuis la branche principale (develop), créer une nouvelle branche avec le nom de la feature (par exemple, `git checkout -b feat/footer`)
- Travailler sur sa branche
- Lorsque vous avez terminé la feature, commit le résultat.
- Retourner sur la branche develop. Vous constaterez que les modifications réalisées sur la branche feat/footer ne sont plus visibles. C'est normal, elles sont restées sur la branche feat/footer.
- Récupérer les modifications réalisées entre temps sur le repo distant (`git pull origin develop`)
- Fusionner, depuis là où vous êtes (develop), votre branche de feature (`git merge feat/footer`)
- Régler les conflits s'il y en a.
- Commit.
- Push la nouvelle version de develop (`git push origin develop`)

## À la fin du projet (ou à chaque mise en production)

- En local, se déplacer sur develop.
- Mettre à jour la branche `git pull origin develop`
- Se déplacer sur la branche master
- Fusionner les modifications de develop vers master `git merge develop`
- Push master en distant. `git push origin master`