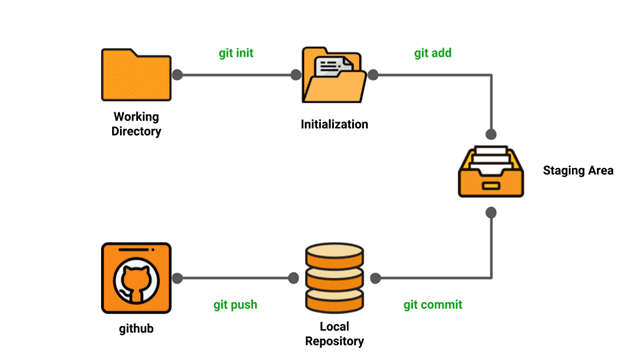
# Git & Github

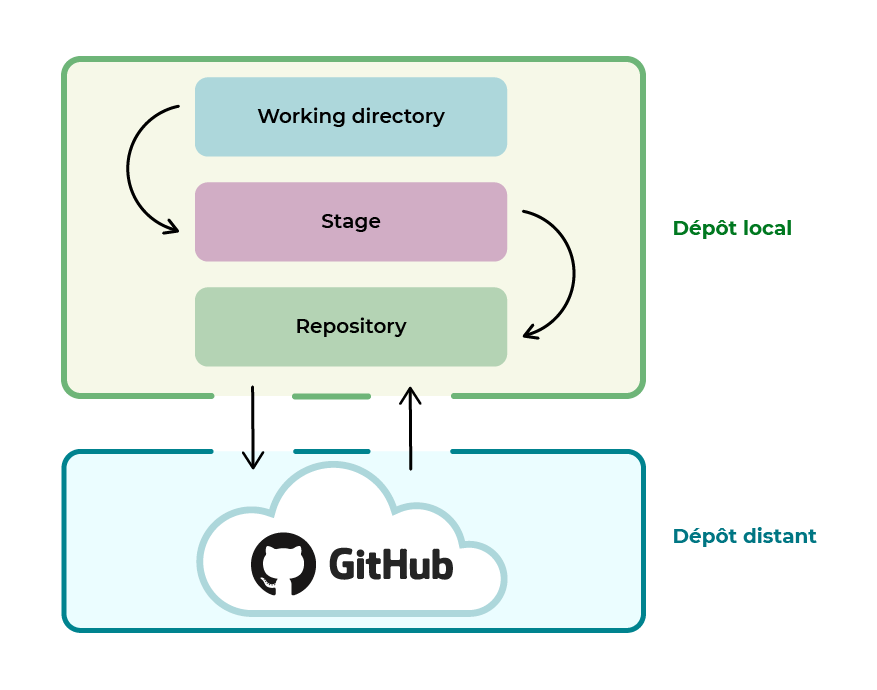
## Rôle de chacun

Git et Github sont différents en tous points, Git est une application qui s’utilise en ligne de commande, il est installé localement à la machine et est maintenu par Linux, il se concentre sur le contrôle des versions et le partage de codes.  
Il existe aussi Git desktop, que nous utilisons. C’est exactement pareil que Git mais cela utilise une interface graphique.

Github quant à lui est un dépôt distant. Il possède une interface graphique accessible et hébergé sur le web, il est maintenu par Microsoft, son objectif est de centraliser les codes sources et de les héberger, Github est aussi un service d’hébergement pour des répertoires Git.

Git





# Notre organisation

## Git

Dans notre projet, nous avons créé 6 branches : une par étudiants, un pour notre professeur chargé du suivi du projet et une branche principale.

Une image contenant diagramme, capture d’écran

Description générée automatiquementCommit : Le commit est utilisé pour ajouter du contenu sur une branche.

Merge : Il agit comme une fusion de branches, il a connaissance de ce que contient chaque branche et ne dupliquera donc pas deux fichiers identiques, il n’y aura qu’un seul fichier.

## Git Desktop

Nous utilisons Git desktop, c’est identique à Git, il possède juste un affichage graphique. Nous envoyons nos modifications sur nos propres branches. Les branches sont hébergées sur un répertoire commun et privé sur Github. L’utilisation d’un dépôt privé nous assure une sécurité quant aux fichiers publiés et seul les membres du répertoire y ont accès contrairement à un répertoire public qui est visible par tous.

## Git Visual Studio

Pour ceux utilisant Visual Studio, qui est un IDE (Environnement de développement intégré) ils ont la possibilité d’accéder directement à Git via un affichage graphique différent de Git desktop.

Une image contenant texte, logiciel, Logiciel multimédia, Police

Description générée automatiquementUne image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

Ici, nous avons accès à Par exemple ci-dessus, se trouve la branche de l’étudiant « Enzo », chaque branche des on y voit le détail de ses modifications ainsi que leur date.  
étudiants.

A signifie : éléments ajoutés

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

M signifie : éléments modifiés

Une image contenant texte, capture d’écran, Logiciel multimédia, logiciel

Description générée automatiquementDans cet exemple,   
à gauche se trouve l’ancien programme.

À droite le nouveau programme avec les indications le concernant.

Une image contenant capture d’écran, carré, Rectangle, motif

Description générée automatiquement

Mise en avant des modifications précises

Lignes ajoutées comparé à l’ancien programme

Lignes modifiées par rapport à l’ancien programme

Espace dédié aux éléments ajoutés afin d’avoir un visuel des nouvelles modifications

Éléments supprimés ou modifiés

## Github

Github est simple d’utilisation et très intuitif, voici un exemple ci-dessous.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Nous pouvons accéder aux différentes branches très simplement en faisant dérouler la liste des branches comme ci-dessous.

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

## Notre dépôt

[zozowolf/CAMS (github.com)](https://github.com/zozowolf/CAMS)