

**Rapport d’activités générales des taches effectués.**

**Réalisé par : abdelhaq sali**

**Table des matières**

[*Introduction :* 2](#_Toc26347487)

[*Scénario #1* 3](#_Toc26347488)

[1 : First Step : Immersion 3](#_Toc26347489)

[2 : Second Step : La découverte 4](#_Toc26347490)

[3 : third step : Historique 5](#_Toc26347491)

[*Scénario2* 6](#_Toc26347492)

[1 : First Step : Tagging 6](#_Toc26347493)

[2 : Second Step : Stashing and Saving work in Progress 6](#_Toc26347494)

[*Scénario 3* 8](#_Toc26347495)

[Second Step : Branching and merging sur GITHUB 8](#_Toc26347496)

[Third step : compare pull Requests 10](#_Toc26347497)

[Conclution : 11](#_Toc26347498)

# ***Introduction :***

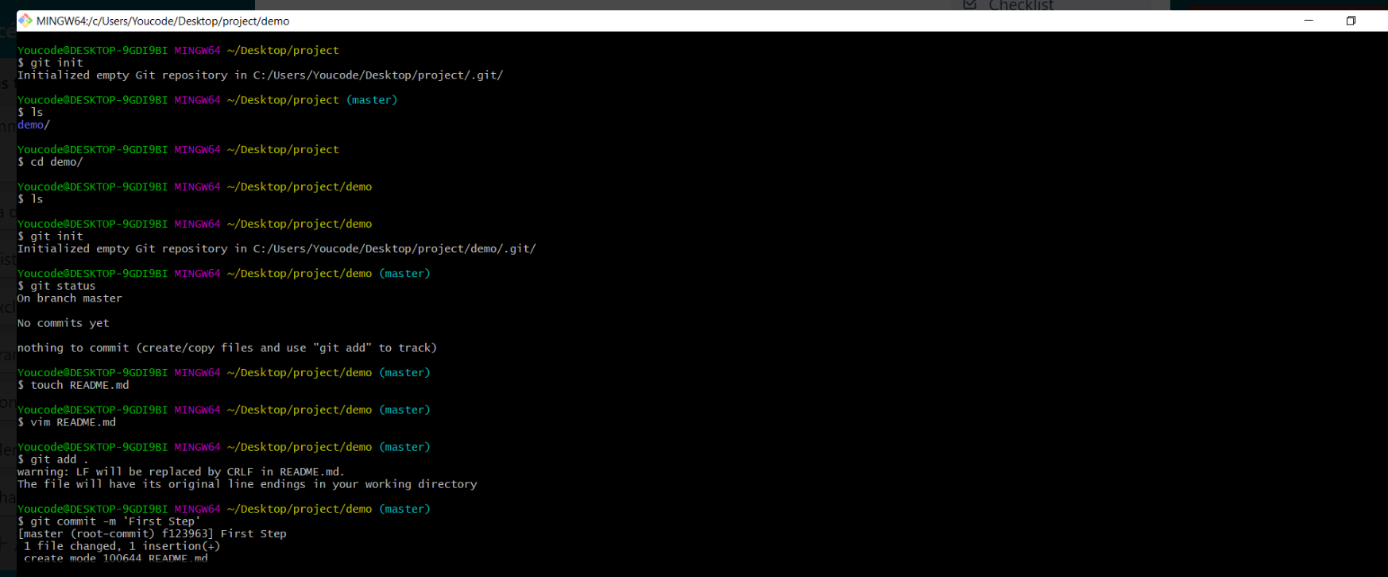
**La gestion des workflows sous GIT/GITHUB sera la base de notre apprentissage en terme de la gestion de projet agile.**

**Un Scrum Master maintient la motivation de l’équipe au plus près des objectifs stratégiques. Il/Elle est le garant de l’amélioration continue des pratiques professionnelles. Il/Elle favorise une communication conviviale et un management visuel en gestion de projet agile.**

**En faisant preuve de transparence et d’inspections bienveillantes, il/elle est capable d'analyser rapidement le niveau de maturité des demandes de l’équipe et leur impact sur la gestion quotidienne du projet.**

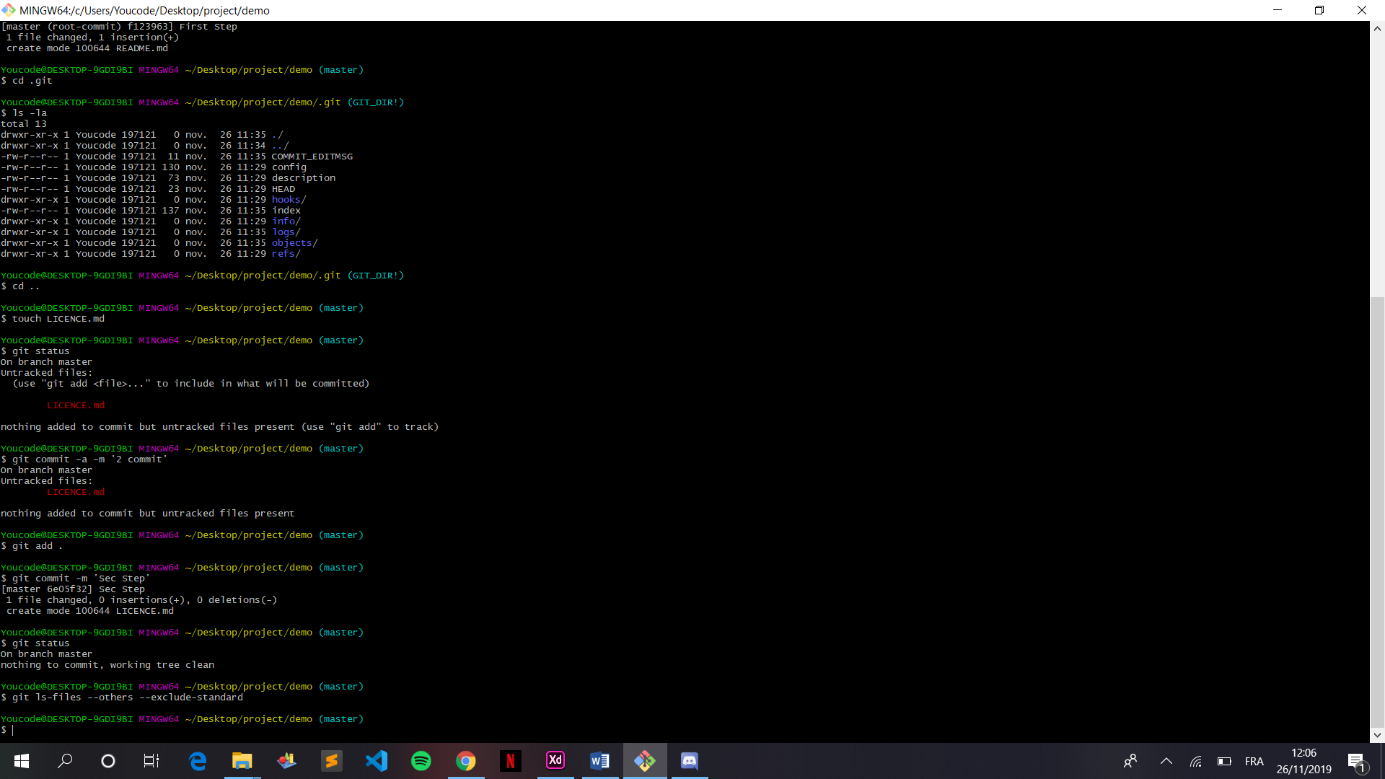
# ***Scénario #1***

### 1 : First Step : Immersion

Git : est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds.

git : (c’est un dossier caché situé à la racine du projet). Il contient l’historique des modifications des fichiers et la configuration de Git pour ce projet. git1.PNG

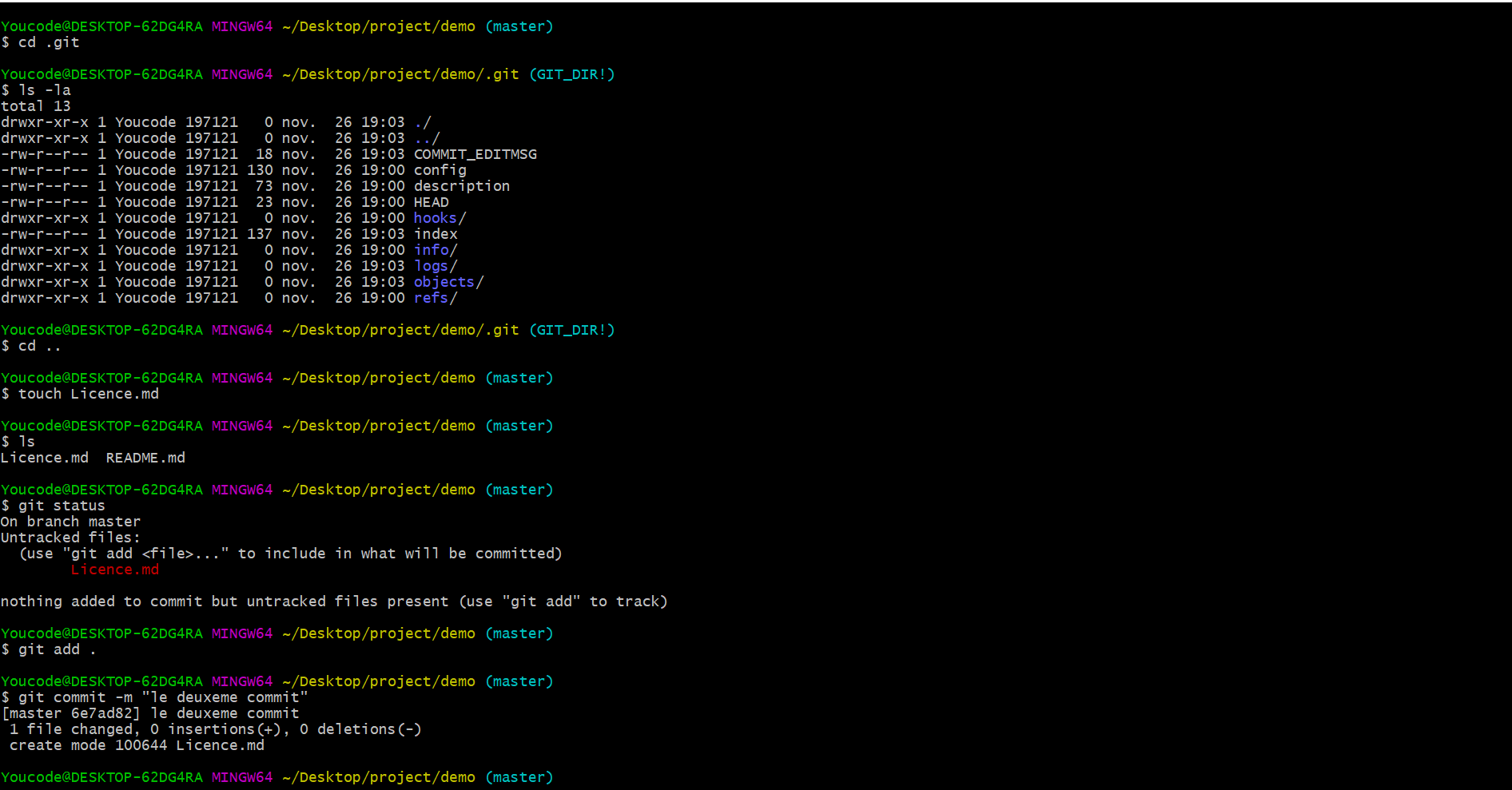
### 2 : Second Step : La découverte



HEAD : est une référence au dernier commit de la branche de sortie actuelle. Vous pouvez considérer la tête comme la "branche actuelle".

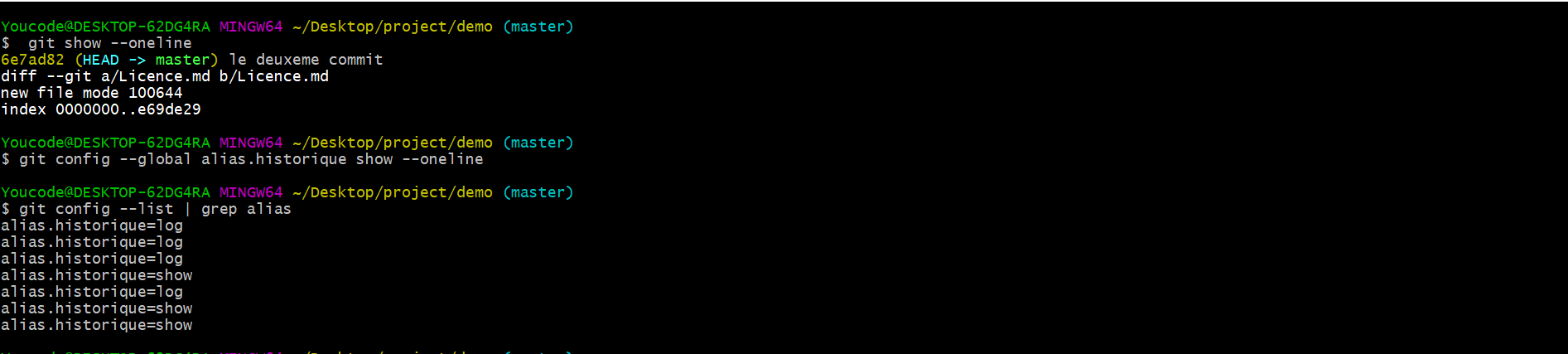
LOGS : L'outil Git Log vous permet d'afficher des informations sur les validations précédentes effectuées dans un projet.

BRANCHES : Une branche dans Git est simplement un pointeur mobile léger sur l'un de ces commits. Le nom de branche par défaut dans Git est master



### **3 : third step : Historique**

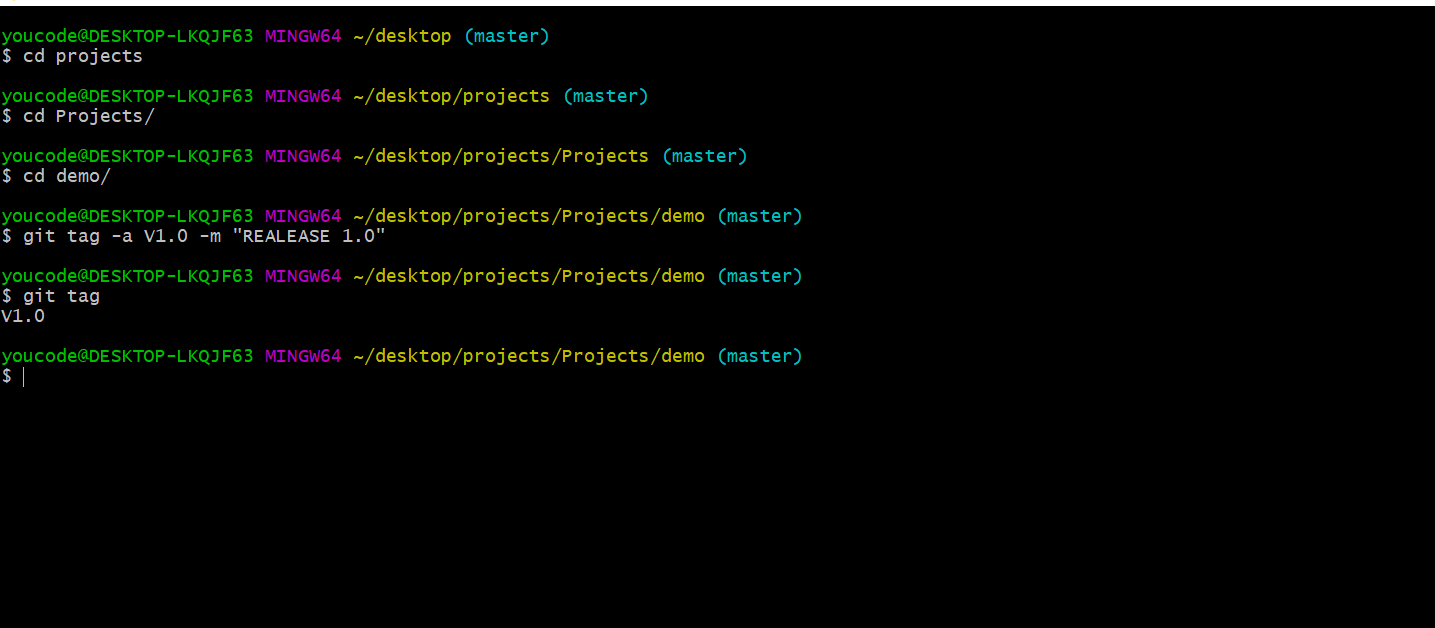
Pour commencer cette étape, nous avons tout d’abord Affichez le dernier commit,toute en ajoutant l’option d’affichage de la hiérarchie de la branche, avec les commit et leur branche aussi. Nous avons fait ça avec la commande suivant :



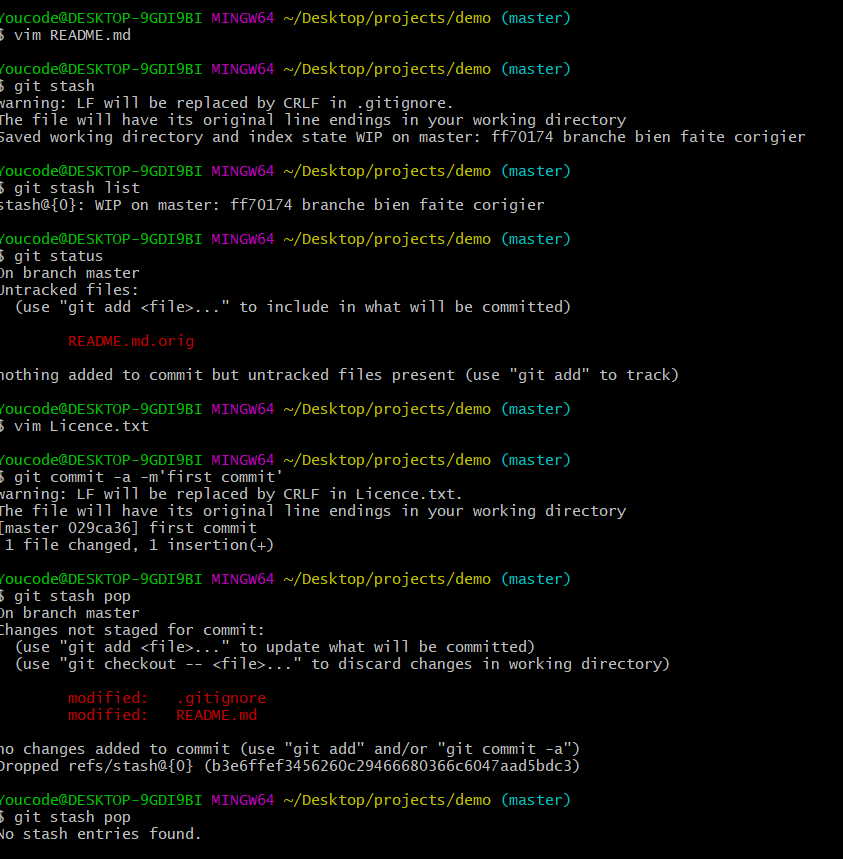


# ***Scénario2***

### 1 : First Step : Tagging



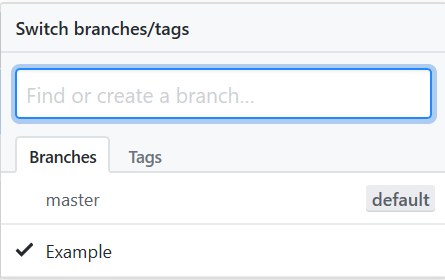
### 2 : Second Step : Stashing and Saving work in Progress

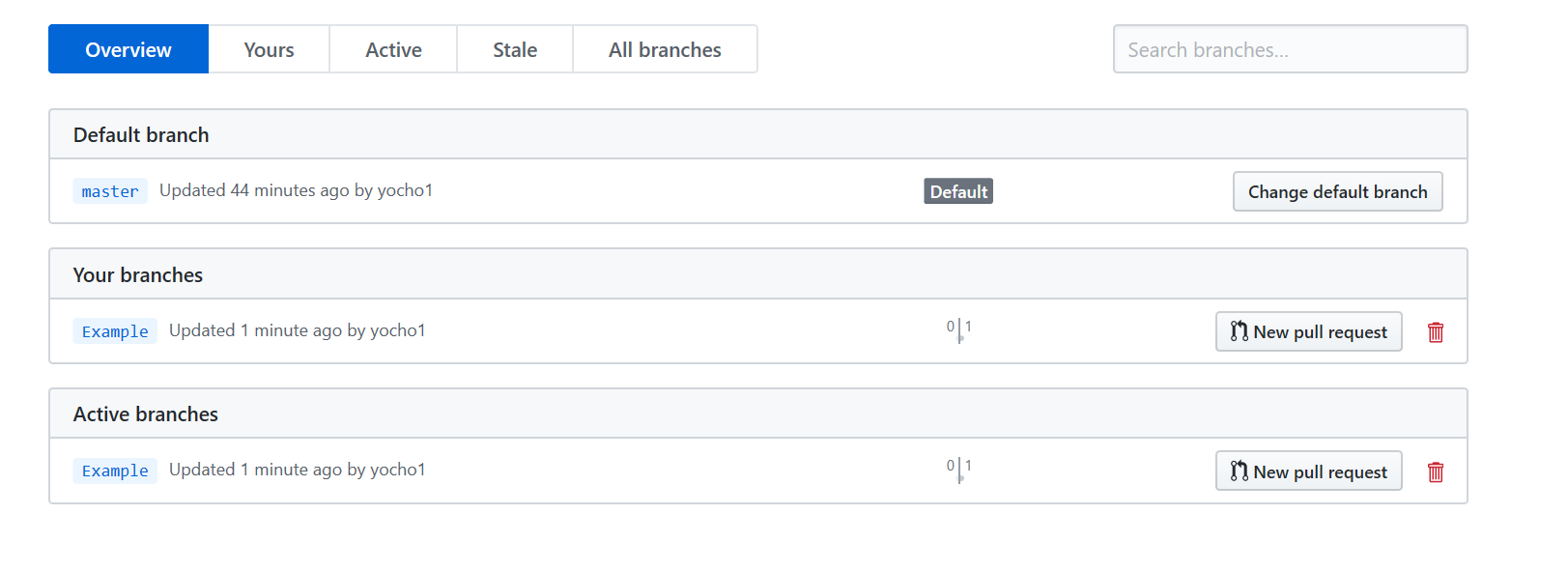


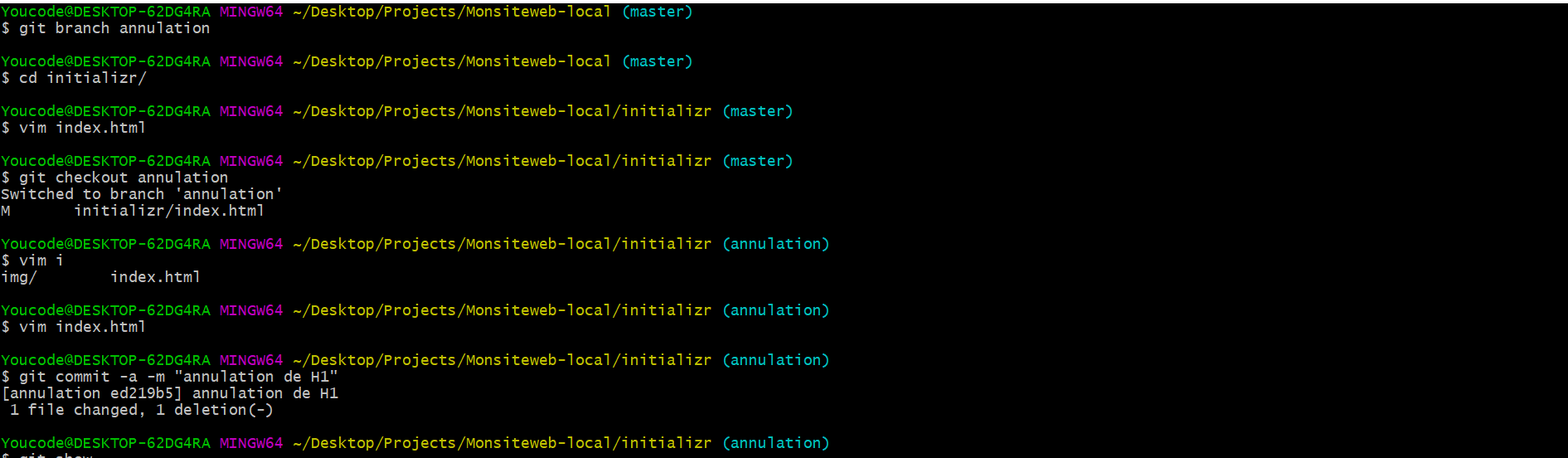
git stash: temporairement les modifications que vous avez apportées à votre copie de travail afin que vous puissiez travailler sur autre chose, puis revenir et les réappliquer plus tard

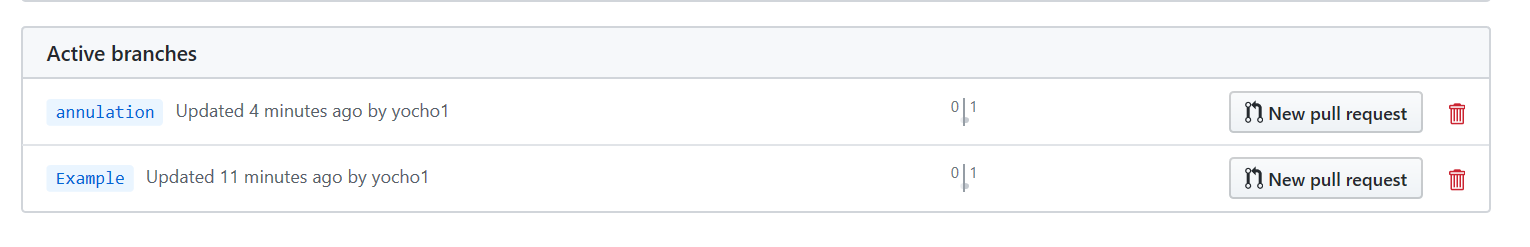
# ***Scénario 3***

### **Second Step : Branching and merging sur GITHUB**







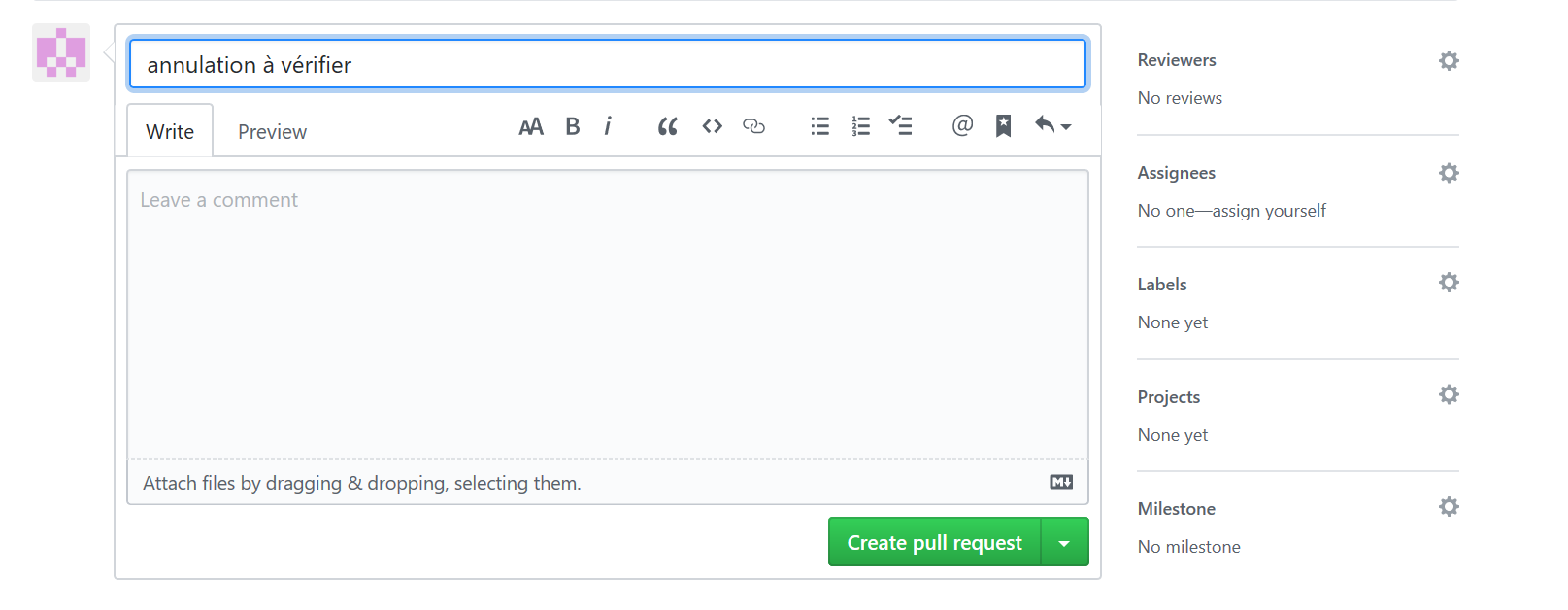


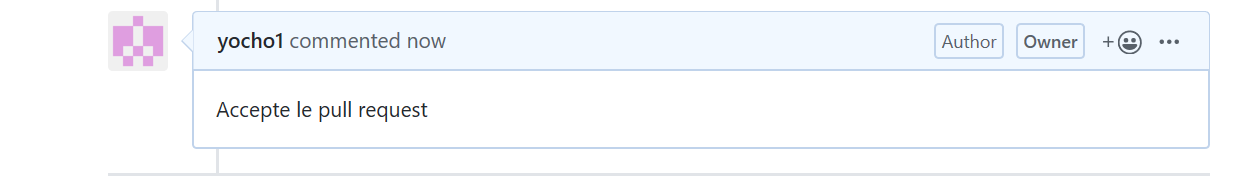


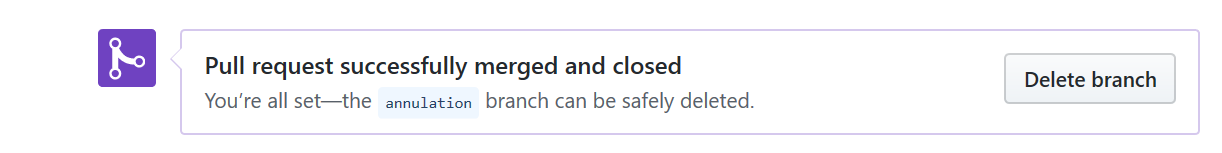
### Third step : compare pull Requests

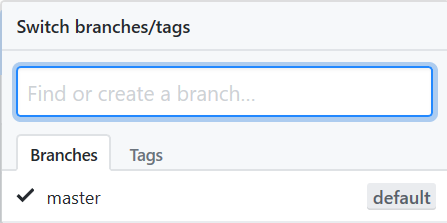
https://media.discordapp.net/attachments/643752973562019842/651514751922470922/Annotation_2019-11-28_004550.png?width=1443&height=72











# **Concluion :**

**En conclusion, Git fournit un moyen de garder une trace des versions antérieures des logiciels et des papiers, facilitant la collaboration entre différents auteurs et assurant la sauvegarde de votre logiciel. Cela s’est avéré très utile pour la communauté des logiciels libres et les milieux universitaires. Git a aussi plusieurs autres commandes qui peuvent être utiles pour effectuer des actions moins standard telles que remove données sensibles, nettoyer l'historique pour économiser de l'espace, etc.**