



Présentation de GitHub

ARCHITECTURES LOGICIELLES ET QUALITE

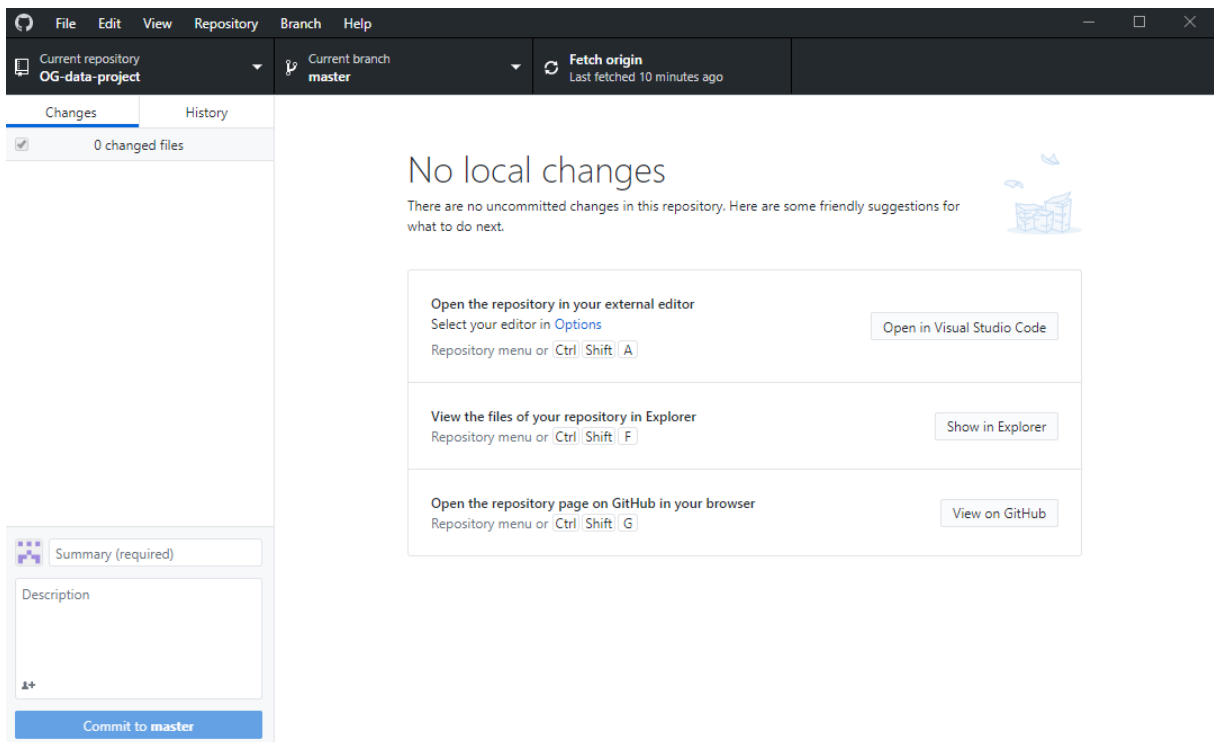
Jael Vavasseur

6 AVRIL 2020 | ISIMA – D.HILL

Présentation de GitHub

GitHub est une plateforme d'hébergement web et de gestion de développement logiciel utilisant le logiciel de versionning Git. Git est la technologie la plus utilisée pour faire du versionning et GitHub permet une utilisation facile et assez intuitive de Git permettant un travail en équipe facile et efficace.

En plus de sa plateforme en ligne où l'on peut gérer chacun de ses repositories, GitHub propose une application de bureau facilitant encore plus l'utilisation de la technologie Git en substituant la console de commande pour une interface graphique intuitive.



Installation

Pour utiliser la plateforme en ligne GitHub, il suffit d'avoir un compte GitHub et d'avoir soit une installation de GitHub desktop soit une simple installation Git. L'installation de Git se fait pour Windows par ce lien : <http://git-scm.com/download/win> ou par la commande suivante sur Linux :

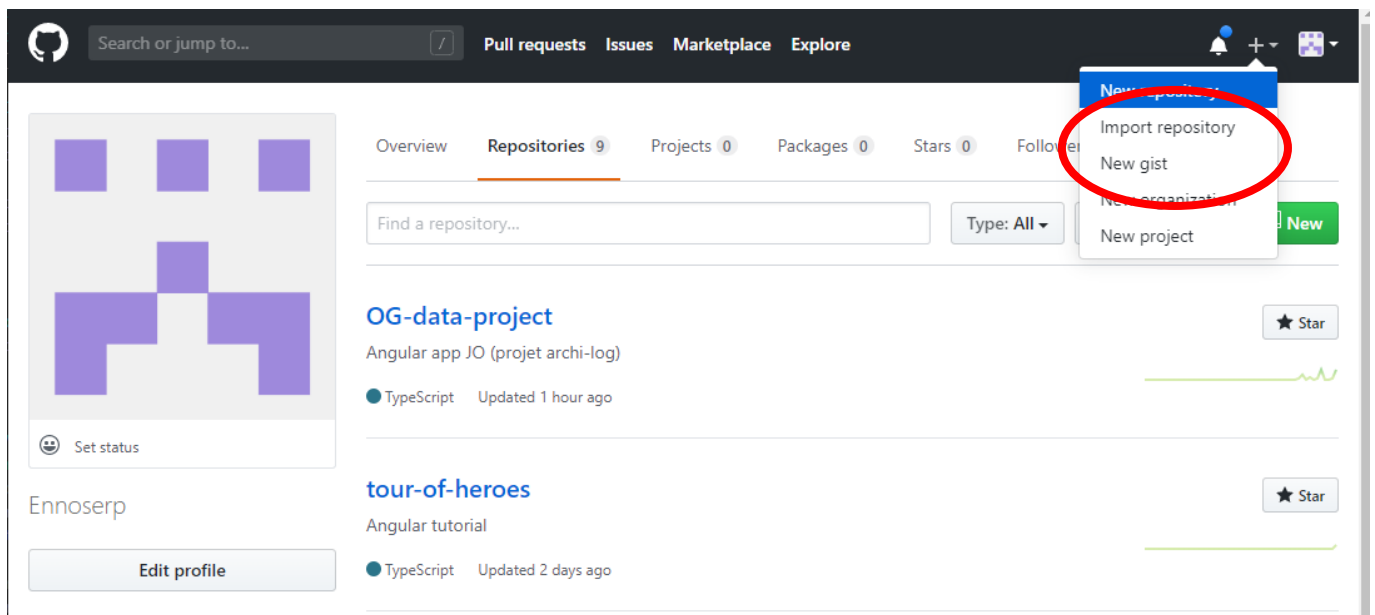
```
$ dnf install git-all
```

Une fois qu'on a installé Git ou GitHub desktop, on peut commencer à utiliser GitHub pour gérer et créer des nouveaux projets.

Utilisation

Pour créer un repository, il faut être connecté à la plateforme et cliquer sur « new repositories » comme indiqué ci-dessous. Cela correspond à la commande :

```
$ git init <nom-dossier>
```



On peut aussi partir d'un dossier existant avec le bouton « Import repository » qui correspond à la commande Git :

```
$ git clone /dossier
```

Une fois que le dossier est bien configuré et stocké sur la plateforme web on peut commencer à travailler seul, ou collaborer avec une équipe. Une fois le travail effectué, il faut sauvegarder ce travail. En ligne de commande il faut d'abord faire un git add :

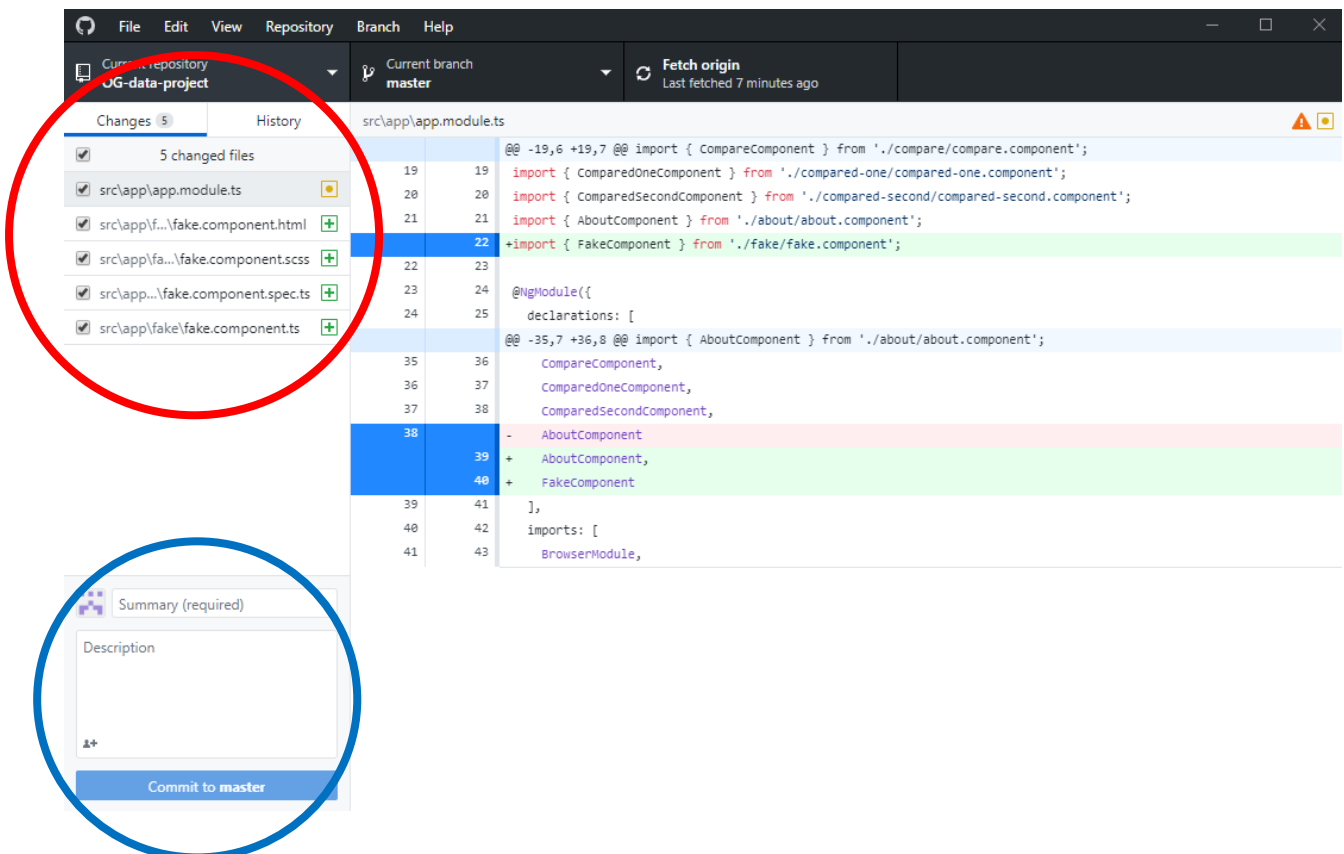
```
git add <nom_fichier>
```

Ensuite, on peut commit les changements en y ajoutant un message obligatoire décrivant les changements effectués.

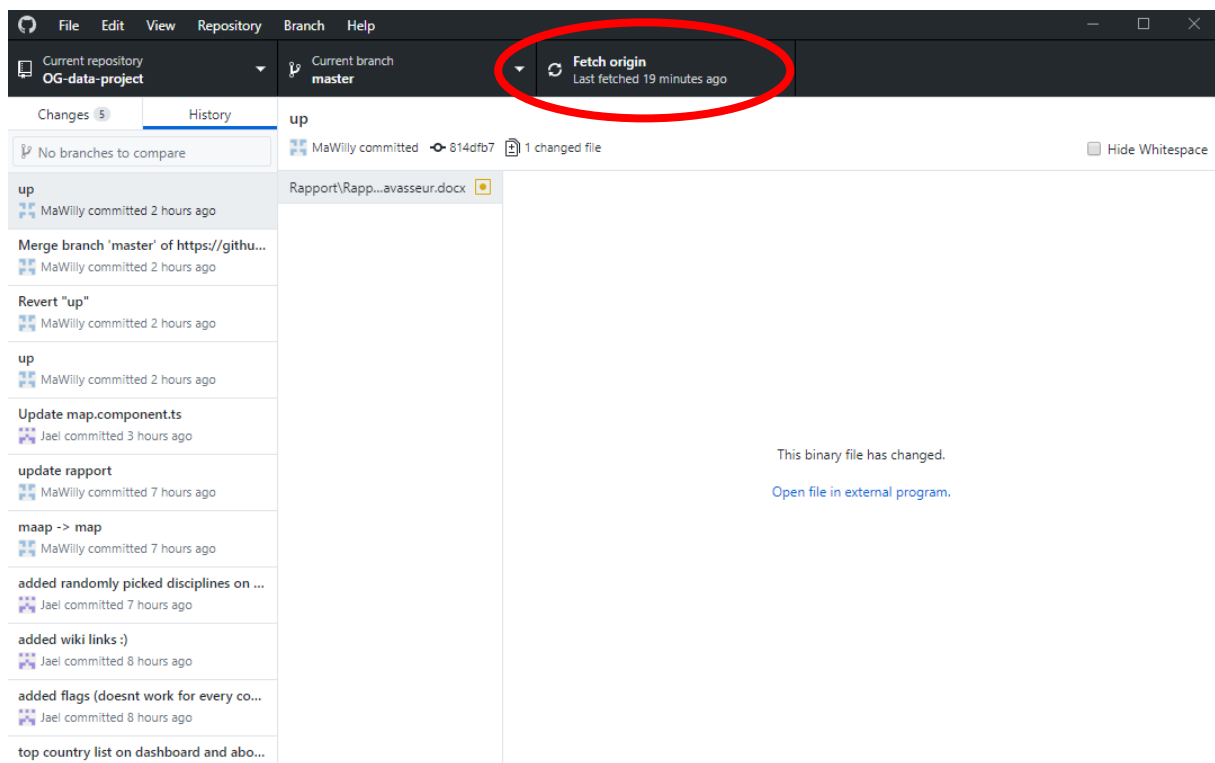
```
git commit -m "Message pour décrire le commit"
```

Les fichiers commit sont affectés au repository, mais ne sont pas encore envoyés au dépôt distant, et il suffit simplement de taper la commande *git push* pour ce faire.

En utilisant l'appli desktop GitHub, c'est encore plus simple : les fichiers modifiés ou ajoutés sont affichés sur la gauche et on peut voir en cliquant sur chaque fichier les modifications apportées à ce fichier au milieu de l'application.



On doit en bas à gauche de l'application (cercle bleu) mettre un commentaire requis pour le commit ainsi qu'une description optionnelle des modifications que l'on a apportées. Comme pour les lignes de commandes, on doit commit (on ne doit pas cependant add, c'est automatique) en appuyant ici sur le bouton commit to master (ou commit sur toute autre branche...). Le push se fait ensuite en cliquant sur le bouton entouré ci-dessous, qui ici ne propose pas de push puisqu'aucun commit n'a été fait depuis le dernier push.



On peut aussi avoir une vue sur toutes les commits effectués par tous les collaborateurs du projet dans l'onglet *History* ce qui est très pratique pour avoir une idée de l'évolution et de la situation d'un projet (notamment avec la date du dernier commit). On peut également facilement *revert* un commit grâce à cet historique, c'est-à-dire annuler toutes les modifications faites durant un commit et revenir à l'état du précédent commit.

Conclusion

GitHub est la plateforme la plus utilisée en ce moment pour stocker et gérer ses versions, et le logiciel Desktop proposé est parfait pour tous les développeurs néophytes qui n'ont pas ou peu de connaissances sur la technologie Git. Tout y est très facile d'utilisation et l'application est assez ergonomique et intuitive, ainsi on ne perd pas de temps inutilement à rechercher les commandes à utiliser et 2 clics suffisent pour faire un push.