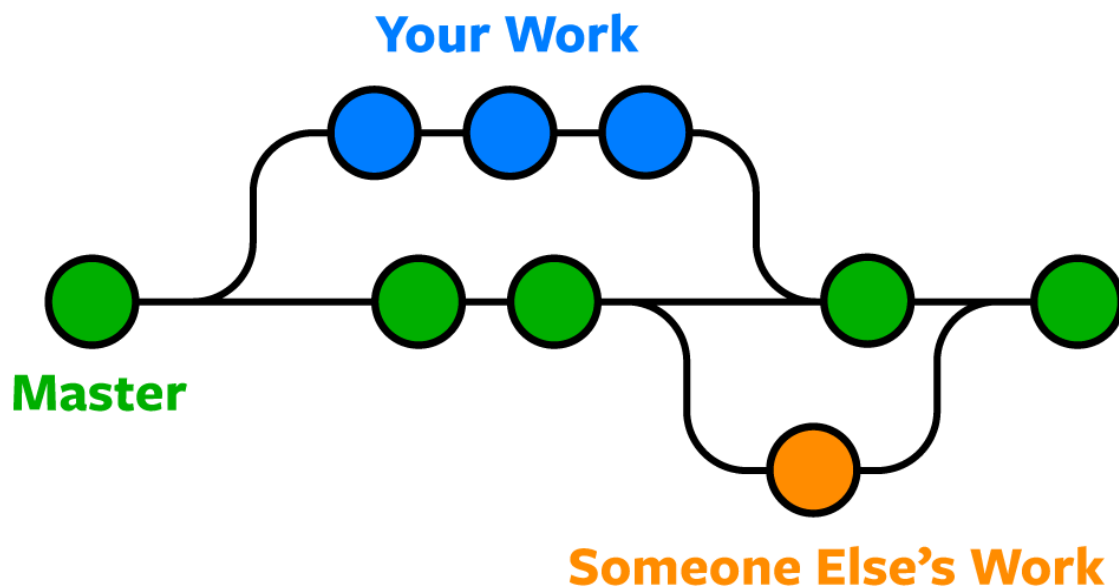


Comment créer et changer de branche

Par Rafael Da Silva Mesquita



Qu'est ce que GitHub ?

GitHub est une plateforme de développement collaborative basée sur Git, un système de gestion de version décentralisé. C'est un endroit où les développeurs peuvent héberger, collaborer et contribuer à des projets logiciels.

Sur GitHub, les utilisateurs peuvent créer des dépôts (repositories), pour leurs projets, suivre les modifications apportées à leur code source, signaler des problèmes, proposer des modifications (pull request), et collaborer avec d'autres développeurs. Il est largement utilisé dans le monde du développement logiciel, mais aussi pour d'autres types de projets collaboratifs.

Prérequis pour changer de branche :

Pour déposer leur code sur GitHub, les utilisateurs doivent satisfaire à certains prérequis de base.

1. Les utilisateurs doivent s'assurer que leur projet est un référentiel Git. Si ce n'est pas encore le cas, il peuvent initialiser un référentiel à l'aide de la commande suivante :

```
git init
```

2. Il faut s'assurer de travailler dans le bon répertoire où le projet est situé. En utilisant la commande suivante, les utilisateurs peuvent vérifier s'ils sont bien dans un répertoire Git :

```
git rev-parse --is-inside-work-tree
```

Si cette commande renvoie "true", alors le projet est bien dans un répertoire.

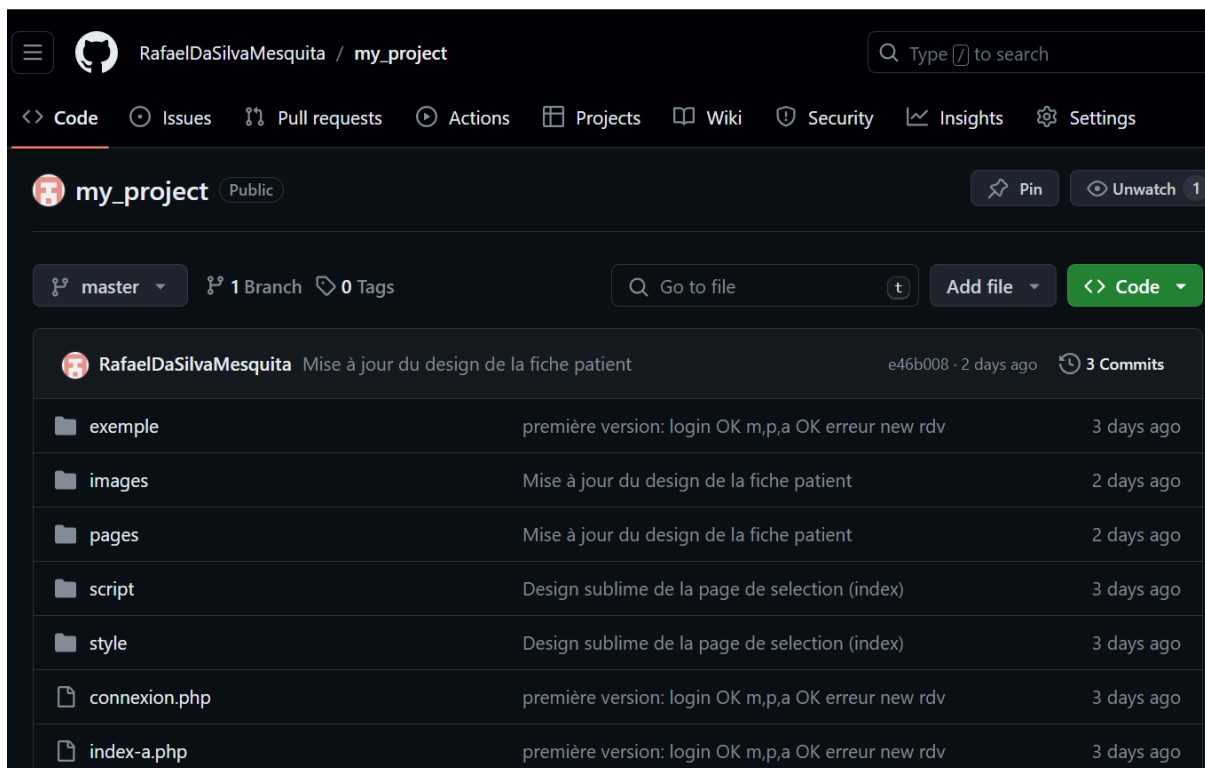
3. Avant de créer une nouvelle branche, il est nécessaire de s'assurer qu'il n'y est aucun changement non comité dans la branche actuelle. Les utilisateurs ont la possibilité de vérifier l'état de leur répertoire de travail en écrivant la commande "git status".
4. Pour finir, il est essentiel de vérifier qu'il n'y est pas de commit en cours, par exemple, si l'utilisateur est en plein milieu d'un processus de commit. Et aussi s'assurer de n'avoir aucun processus de fusion (merge) en cours.

Dans le cadre de la création d'une branche, il est possible de créer une branche directement sur GitHub (<https://github.com/>), sans passer forcément par un terminal.

Création de la branche sur GitHub :

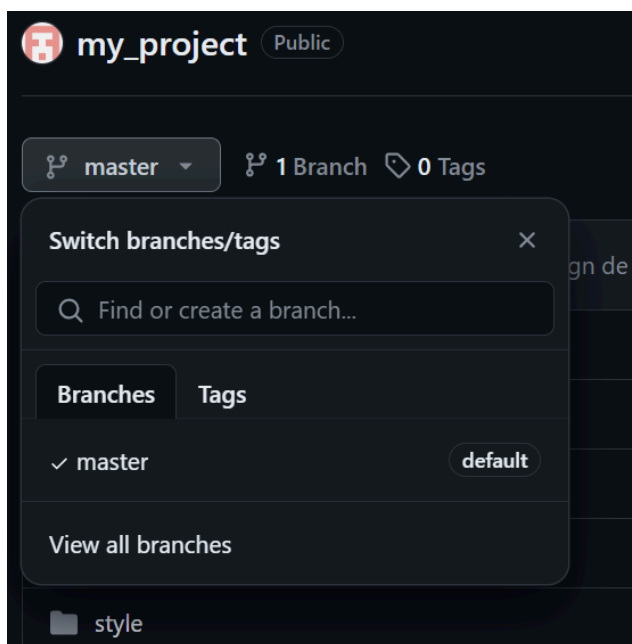
Étape 1 : Accéder au dépôt

Ouvrir le dépôt sur GitHub dans votre navigateur, puis allez dans l'onglet "code" :



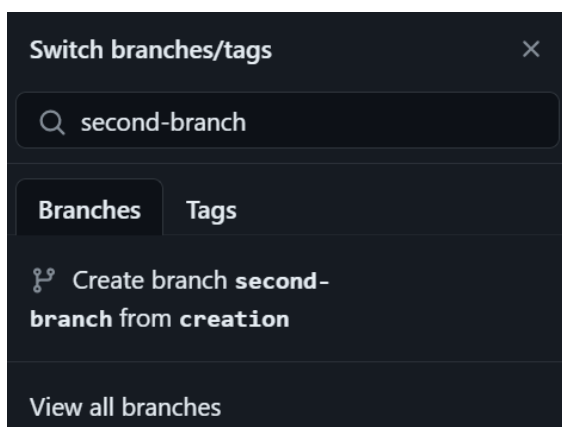
Étape 2 : Sélection de la branche actuelle

A côté de la branche principale (généralement nommée "main" ou bien "master"), il y a un menu déroulant. Cliquez dessus pour voir la liste des branches existantes.



Étape 3 : Créer la nouvelle branche

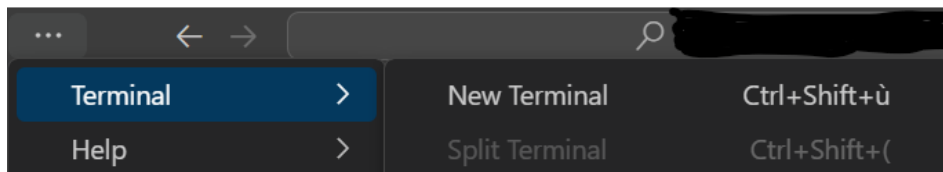
Dans le champ de texte, saisissez le nom de la nouvelle branche que vous voulez créer. Et appuyer sur votre touche “entrer” ou bien cliquer sur le bouton “Create branch” pour créer une nouvelle branche.



Changement de branche localement (dans votre espace de travail) :

Étape 1 : Ouvrir votre terminal

Ouvrez votre terminal dans Visual Studio Code en cliquant sur les 3 points dans la barre d'outils, sur "terminal" puis "nouveau terminal", ou bien utilisez le terminal de votre choix.



Étape 2 : Vérification de la branche actuelle

Utilisez la commande suivante pour vérifier les branches actuelles de votre répertoire local :

```
git branch
```

Étape 3 : Changer de branche

Exécutez la commande suivante pour changer de branche (remplacez "nom_de_la_branche" par le nom de la branche que vous souhaitez rejoindre).

```
git checkout nom_de_la_branche
```

ou bien en saisissant la commande “git switch” :

```
git switch nom_de_la_branche
```

Remarque : à partir de la version Git 2.23, il est recommandé d'utiliser “git switch” et “git restore” plutôt que “git checkout” pour les opérations de changement de branche et de restauration des modifications.

Changer de branche et créer une nouvelle branche en une seule commande :

Entrez la commande suivante pour changer de branche et en créer une nouvelle en une seule étape :

```
git checkout -b nom_de_la_nouvelle_branche
```

ou avec “git switch” :

```
git switch -c nom_de_la_nouvelle_branche
```

Ces étapes devraient vous permettre de créer une nouvelle branche sur GitHub et de changer de branche localement dans votre espace de travail. N'oubliez pas de pousser vos changements vers GitHub après avoir effectué des modifications dans votre nouvelle branche avec “git push origin nom_de_la_nouvelle_branche”.

Pour plus d'informations, ceci est susceptible de vous intéresser :

<https://docs.github.com/fr/get-started>

