2025/12/10 10:02 1/5 Bases Github

Bases Github

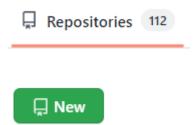
Github est une plateforme Web collaborative reposant sur Git, logiciel de gestion de versions.

Créer un repository

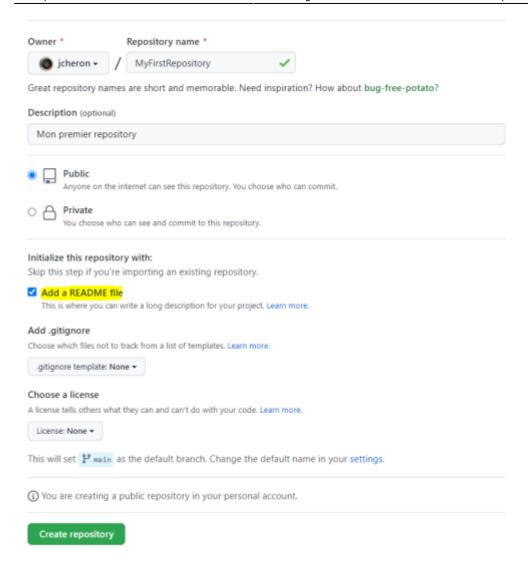
Un repository est un projet hébergé sur la plateforme, permettant de travailler à plusieurs.

La solution la plus simple, surtout pour les débutants, est de créer le repository sur github :

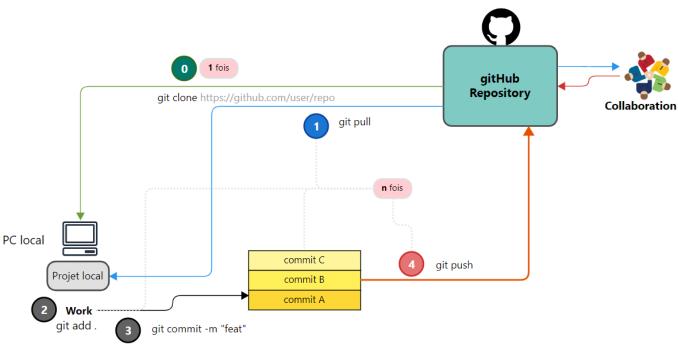
- 1. Créer au besoin un compte github si vous n'en avez pas ;
- 2. Allez dans Repositories, puis cliquez sur New:



Cocher la case Add a README file, pour que le système de fichiers du repository existe (ce qui va permettre de le cloner ensuite).



Principe



?900

2025/12/10 10:02 3/5 Bases Github

Pré-requis

Paramètres Git locaux

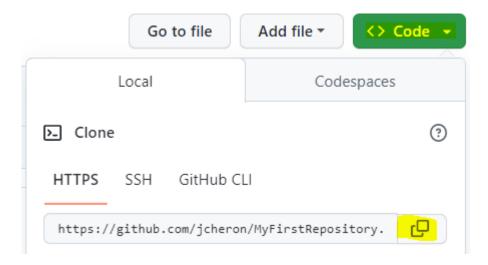
Git ayant besoin de vous identifier, ajouter vos identifiants à la configuration locale de git

Travailler en local

0 - Clone

Pour travailler en local sur le projet, il faut déjà en créer une copie en local : un clone :

Copier l'adresse du repository sur Github, avec le bouton Code :



git clone https://github.com/username/repositoryName.git
cd repositoryName

Ignorer des modifications

Créer le fichier **.gitignore** (il n'y a pas de commande git pour ça)
Il va permettre d'ajouter les fichiers ou les dossiers à ne pas "tracker", en respectant le principe :

• une ligne par exclusion :

```
/conf
main.conf
/lib
/cache
```

1 - Pull

Si vous avez déjà le repository en local, mais qu'il a été modifié à distance (par vous ou un collaborateur), il faut le mettre à jour et intégrer les dernières modifications avant de travailler (sinon, vous risquez les conflits futurs) :

```
git pull
```

2 - Work

Vous travaillez en local sur votre projet...

3 - Commit

- Vous avez travaillé et modifié des fichiers (ajout/modification/suppression)
- Vous pouvez créer un commit, qui représente votre progression (1 commit = 1 unité de code)

git status

Pour ajouter un fichier à tracker (sera ajouté aux prochains commits) :

```
git add filename
```

Pour ajouter tous les fichiers à tracker :

```
git add .
```

Refaire un **git status** pour vérifier :

```
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
   (use "git restore --staged <file>..." to unstage)
        modified: pom.xml
        modified: src/main/kotlin/edu/spring/btp/BtpApplication.kt
```

Commiter les modifications :

```
git commit -m "feat: ajout de [NomFonctionnalite]"
```

Il est possible de continuer ce cycle plusieurs fois :

- Mise à jour / repo distant (pull)
- Travail/modifications
- commit
- ...

4 - push

Il s'agit maintenant d'envoyer les commits locaux non pushés sur **github** :

git push

Pour aller plus loin: Concepts Git

From:

http://slamwiki2.kobject.net/ - SlamWiki 2.1

Permanent link:

http://slamwiki2.kobject.net/cours/git-start

Last update: 2025/12/10 02:22

