

Pas à pas GitHub :

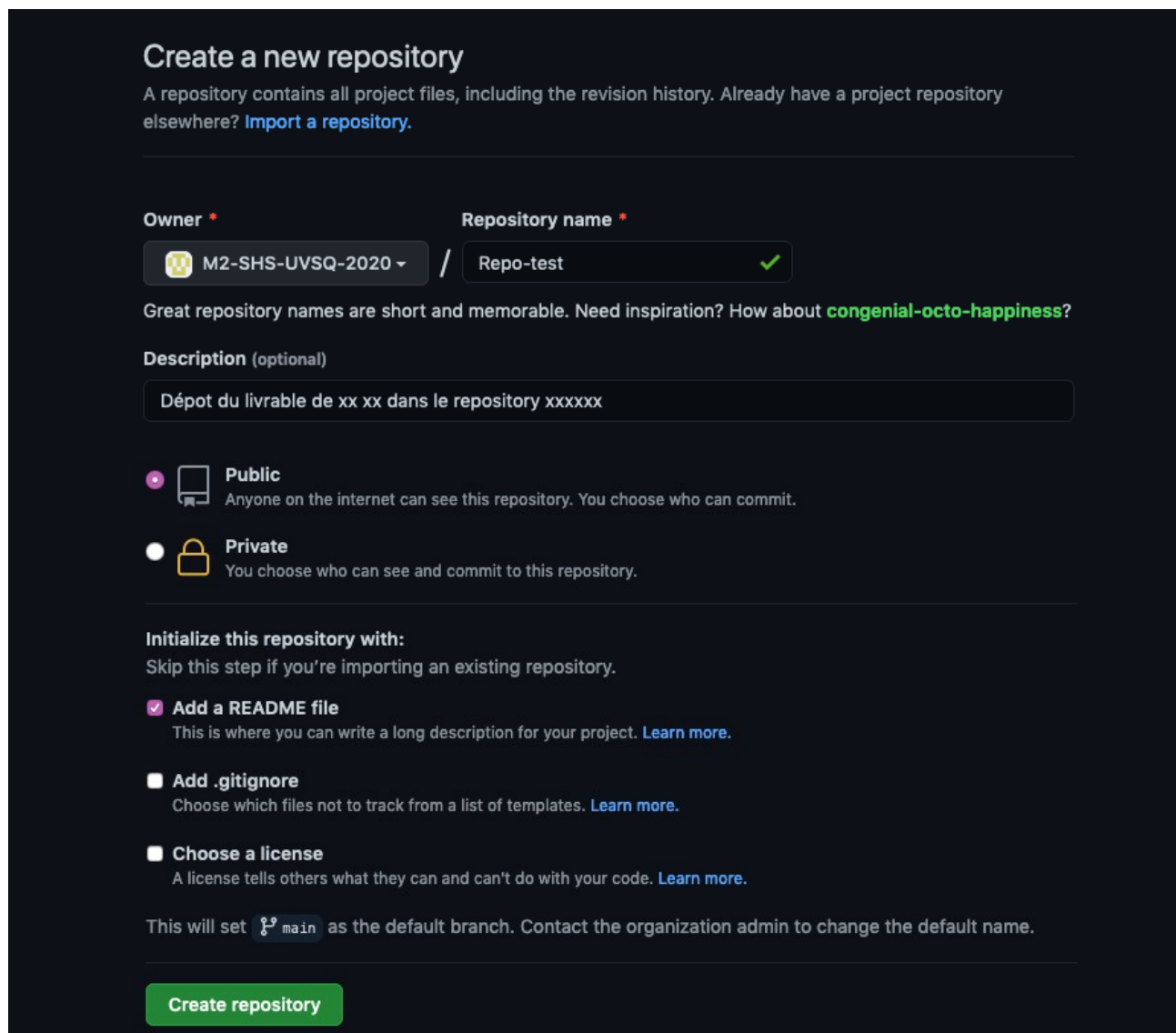
La première chose à faire est de vous créer un *repository* dans l'organisation :

Nous vous conseillons un nom de *repository* parlant, exemple : « votre sujet + votre nom, prénom »

Lorsque vous êtes du M2-SHS-UVSQ-2020 → « *New* »

Vous choisissez un *repository name*, y insérez une description, choisissez de le mettre en public, et ne vous occupez pas de la licence.



Un *README file* est LA bonne pratique, si vous vous souvenez bien → c'est la notice d'utilisation de votre *repository*. Vous pourrez y mettre une description plus grande de ce qu'il contient, les buts, les moyens et les objectifs de ce dernier.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Owner * **Repository name ***


 M2-SHS-UVSQ-2020 / Repo-test 

Great repository names are short and memorable. Need inspiration? How about [congenial-octo-happiness?](#)

Description (optional)

Dépot du livrable de xx xx dans le repository xxxxxx

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.


☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more.](#)

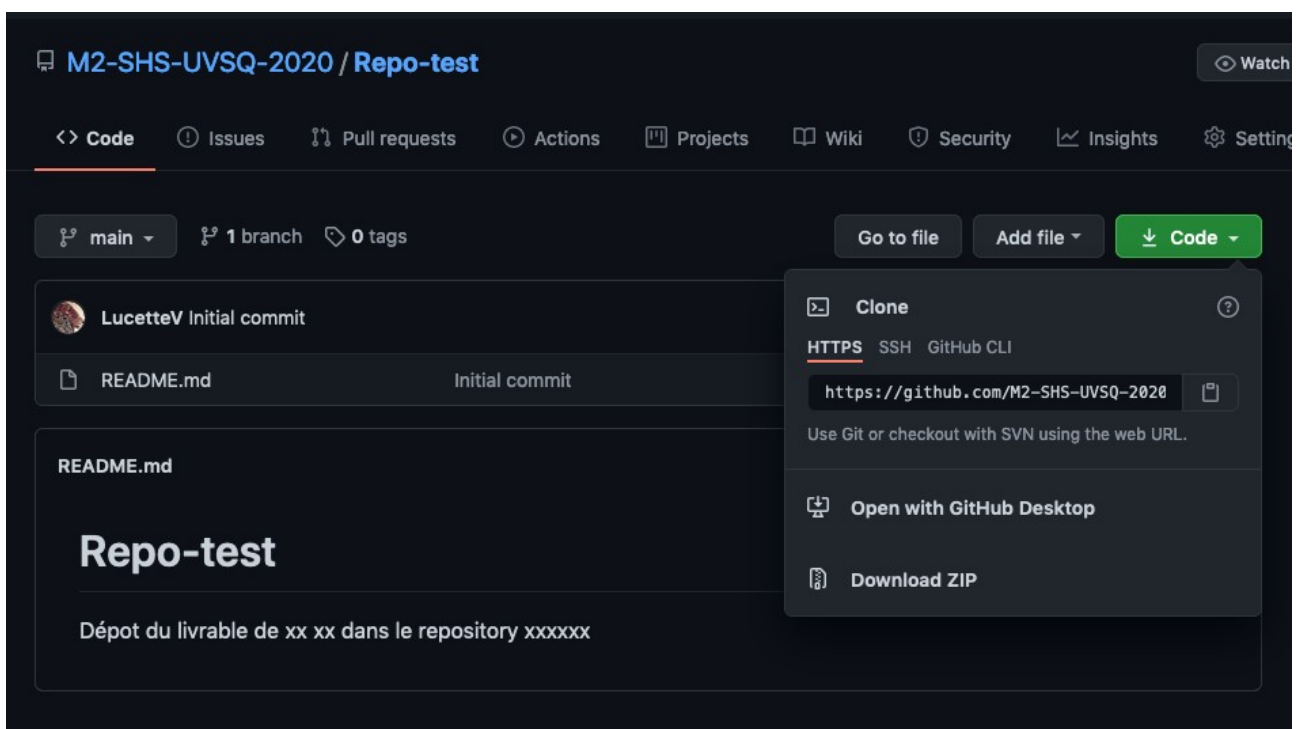
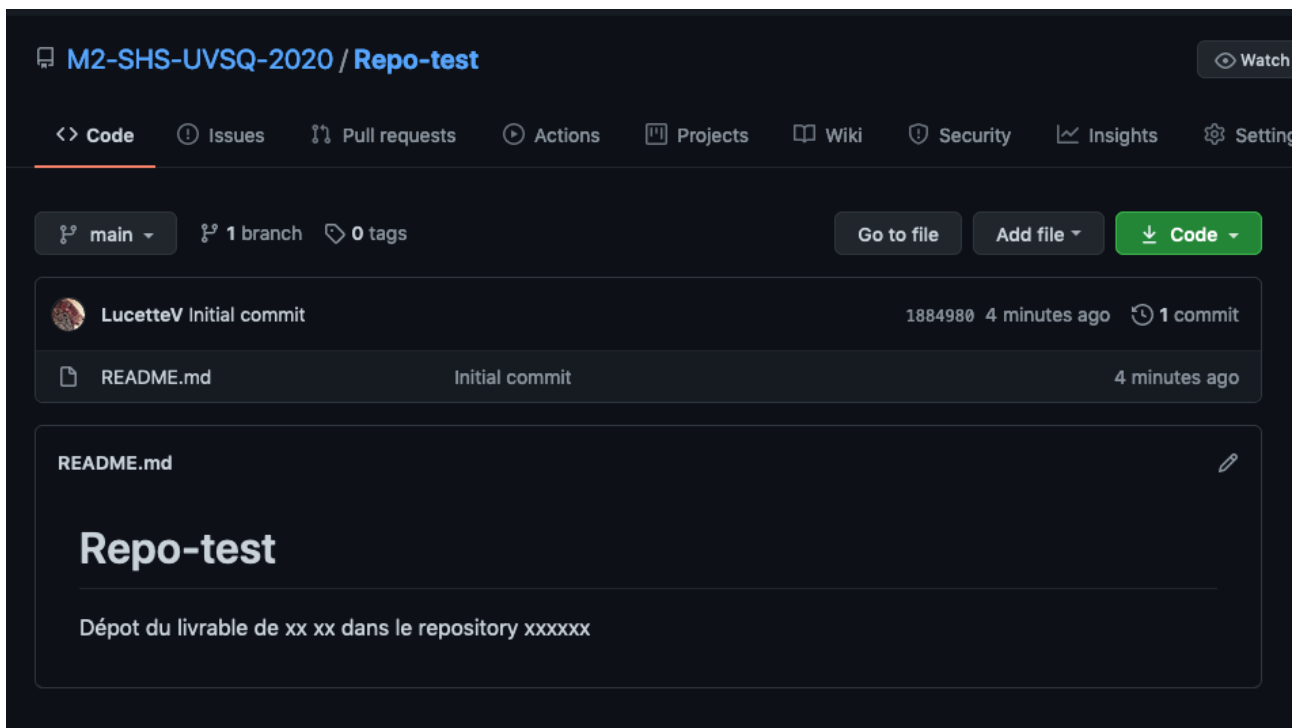
This will set  **main** as the default branch. Contact the organization admin to change the default name.

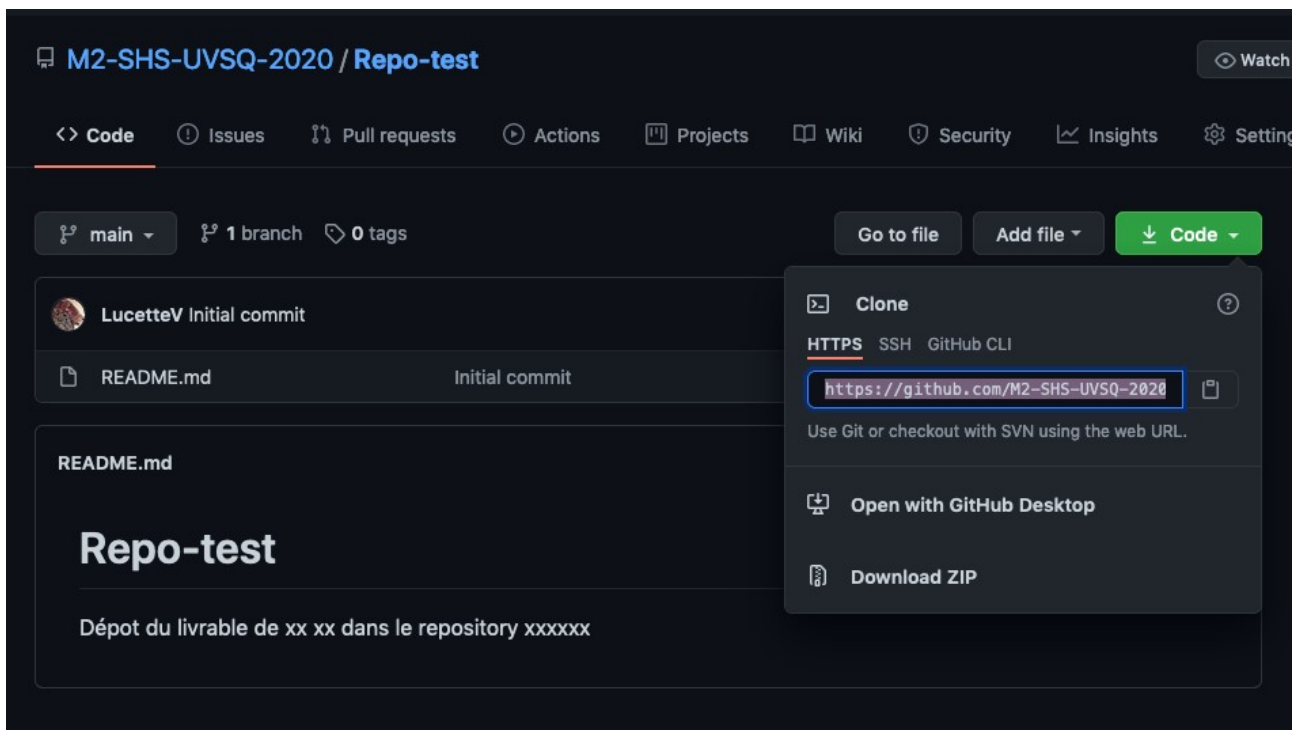
Create repository

Maintenant, nous allons cloner le *repository* dans un dossier créer sur l'ordinateur :

Vous revenez sur la page de M2-SHS-UVSQ-2020 et vous voyez votre nouveau repository → maintenant vous allez cliquer sur **Code**.

Git clone → copier/coller l'adresse `https://` (pour ce•lles•ux qui n'ont pas réussi à dompter le terminal : *Open with GitHub Desktop* [il faut avoir au préalable installer l'application sur votre ordinateur])





Ensuite vous ouvrez le terminal (sous Mac) ou Invite de commande (sous Windows) et tapez les commandes suivantes :

```
Repo-test — -bash — 132x27
Last login: Sun Dec 20 09:50:20 on ttys001
[macbook-pro-de-lucie:~ lucievieillon$ cd Desktop
[macbook-pro-de-lucie:Desktop lucievieillon$ mk dir "UVSQ-livrables"
-bash: mk: command not found
[macbook-pro-de-lucie:Desktop lucievieillon$ mkdir 'UVSQ-livrables'
[macbook-pro-de-lucie:Desktop lucievieillon$ cd UVSQ-livrables
[macbook-pro-de-lucie:UVSQ-livrables lucievieillon$ git init
Initialized empty Git repository in /Users/lucievieillon/Desktop/UVSQ-livrables/.git/
[macbook-pro-de-lucie:UVSQ-livrables lucievieillon$ git add https://github.com/M2-SHS-UVSQ-2020/Repo-test.git
[fatal: pathspec 'https://github.com/M2-SHS-UVSQ-2020/Repo-test.git' did not match any files
[macbook-pro-de-lucie:UVSQ-livrables lucievieillon$ git clone https://github.com/M2-SHS-UVSQ-2020/Repo-test.git
Cloning into 'Repo-test'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
[macbook-pro-de-lucie:UVSQ-livrables lucievieillon$ ls
Repo-test
[macbook-pro-de-lucie:UVSQ-livrables lucievieillon$ cd Repo-test
[macbook-pro-de-lucie:Repo-test lucievieillon$
```

Ensuite :

Vous faites glissez sur votre ordinateur votre livrable vers le *repository* distant que vous avez créé (ma commande cible le Desktop, mais si vous l’avez mis ailleurs vous devez cibler soit Documents ou autres, et n’oubliez pas que le nom de la cible dépend de la manière dont l’ordinateur l’appelle!) Vous retournez dans le terminal et une fois que vous vous êtes bien assurés d’être dans votre repository :

```
git status
```

[vous devriez avoir un message expliquant d’un fichier a été ajouté mais non ‘commiter’]

```
git add 'le-nom-que-vous-avez-donné-à-votre-livrable'
```

[ou n'importe quel message que vous vous voulez tant que c'est limpide !! & n'oubliez pas les coma et non des guillemets]

```
git status
```

vous devriez avoir un message type :

```
macbook-pro-de-lucie:Repo-test lucievieillon$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    deleted:    ~/.lock.pas-a-pas-github.odt#
    modified:   pas-a-pas-github.pdf
```

```
git commit -m 'ajout du livrable'
```

```
git status
```

```
macbook-pro-de-lucie:Repo-test lucievieillon$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

```
git push
```

```
macbook-pro-de-lucie:Repo-test lucievieillon$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 12.24 KiB | 2.04 MiB/s, done.
Total 3 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/M2-SHS-UVSQ-2020/Repo-test.git
  930eeec..d640862  main -> main
```

```
git status
```

[un dernier pour vérifier que tout est ok]

```
macbook-pro-de-lucie:Repo-test lucievieillon$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
macbook-pro-de-lucie:Repo-test lucievieillon$
```