

COMMENT INSTALLER ET UTILISER GIT ?

I – Installation

1.1) Télécharger Git à l'aide du lien suivant : <https://git-scm.com/downloads> en choisissant la version appropriée pour votre OS (Windows, Mac, Linux...).

2.2) Lancer l'installation du Git et attendre. Voilà vous avez Git sur votre ordinateur !

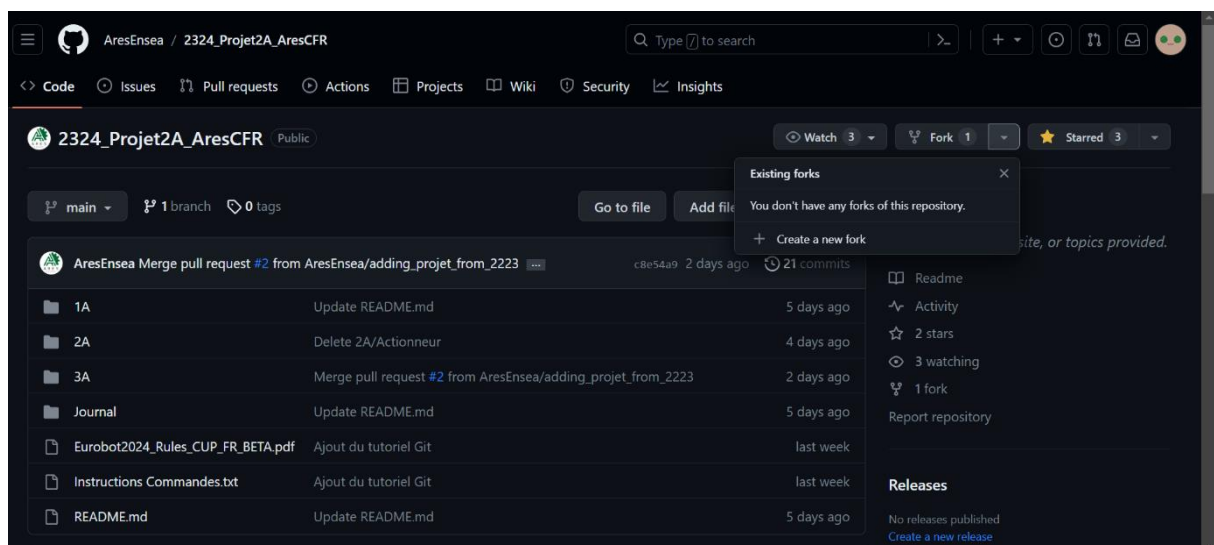
II – Récupérer le dossier du projet

Pour pouvoir mettre en commun tous les fichiers/dossiers que vous allez créer, il faut aller chercher le dossier de partage en ligne, sur Github.

2.1) Créer un compte Github

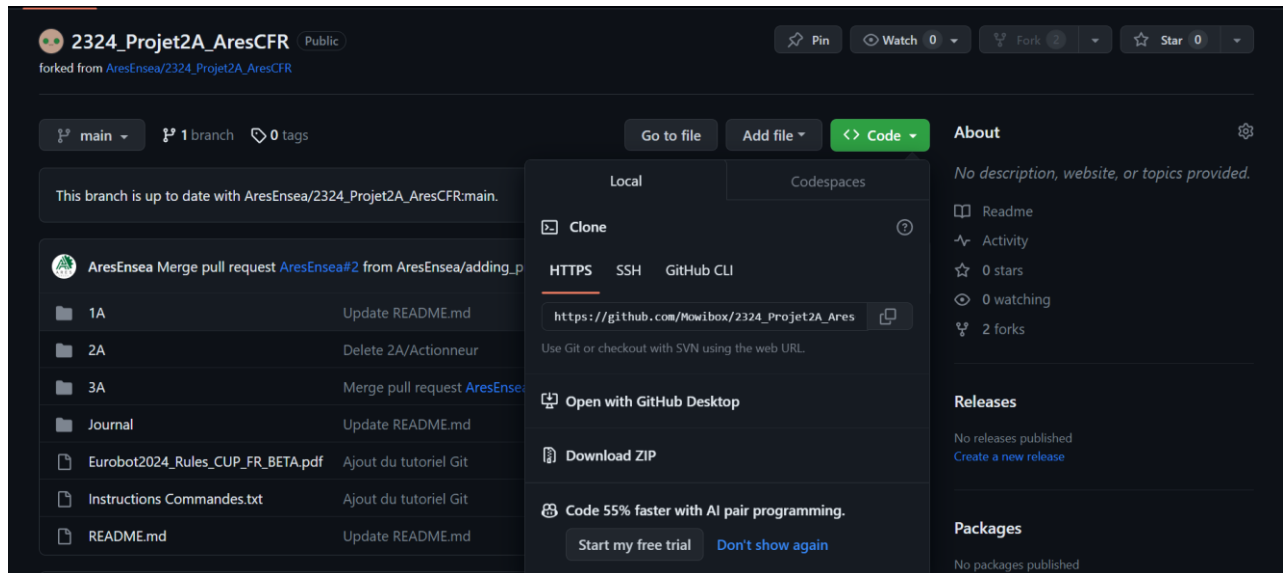
2.2) Se rendre sur la page : https://github.com/AresEnsea/2324_Projet2A_AresCFR

2.3) Pour récupérer le dossier, il faut cliquer sur la flèche à côté de Fork puis « Create a new Fork ».



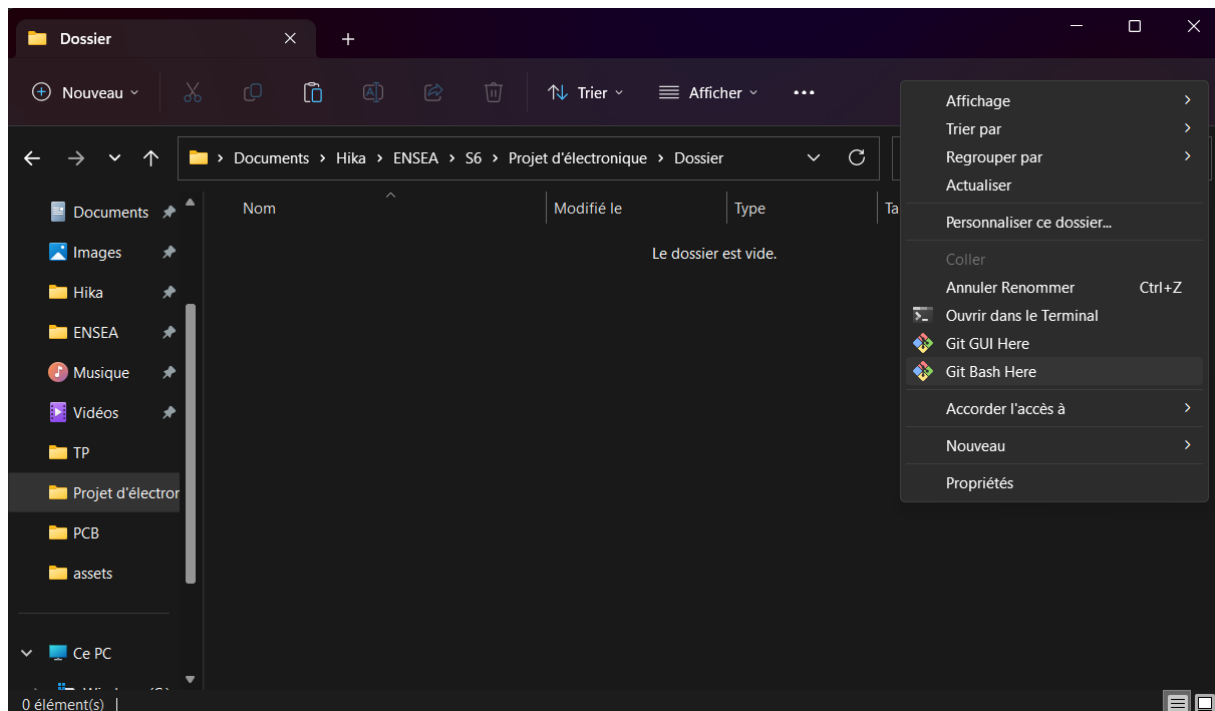
Une fois l'opération validée, vous serez redirigé vers une copie du dossier du projet que vous allez pouvoir manipuler localement sur votre ordinateur.

2.4) Pour récupérer votre dossier il faut cliquer sur le bouton **< > Code**. Un lien https apparaît alors. Copier le lien du repository :



2.5) Sur votre ordinateur, choisissez un emplacement où cloner le dossier du projet. Une fois l'emplacement décidé :

– Sur Windows : Mettez vous dans l'emplacement désiré puis avec un clic droit, accédez à « Git Bash Here » :



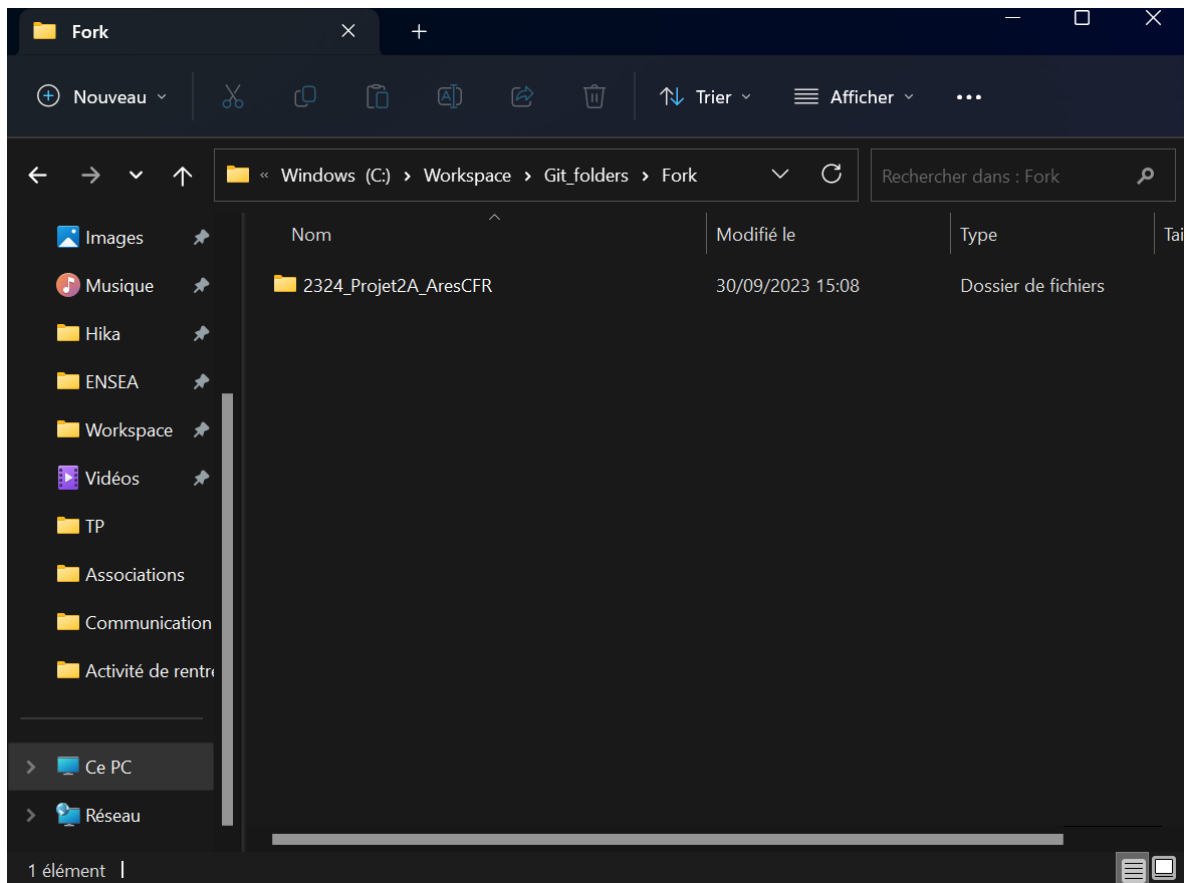
– Pour Mac et Linux, la console de Git est intégrée à la console du système. Vous pouvez alors ouvrir votre invite de commandes et vous déplacer à l'endroit voulu avec la commande « `cd` ».

2.5) Une fois la console Git lancée, vous n'avez plus qu'à exécuter la commande « `git clone` » suivie de l'adresse copiée précédemment :

```
Mowibox@DESKTOP-GLLIHSG MINGW64 /c/Workspace/Git_folders/Fork
$ git clone https://github.com/Mowibox/2324_Projet2A_AresCFR.git
Cloning into '2324_Projet2A_AresCFR'...
remote: Enumerating objects: 1996, done.
remote: Counting objects: 100% (1996/1996), done.
remote: Compressing objects: 100% (1511/1511), done.
remote: Total 1996 (delta 539), reused 1883 (delta 452), pack-reused 0
Receiving objects: 100% (1996/1996), 38.23 MiB | 18.97 MiB/s, done.
Resolving deltas: 100% (539/539), done.
Updating files: 100% (2099/2099), done.

Mowibox@DESKTOP-GLLIHSG MINGW64 /c/Workspace/Git_folders/Fork
```

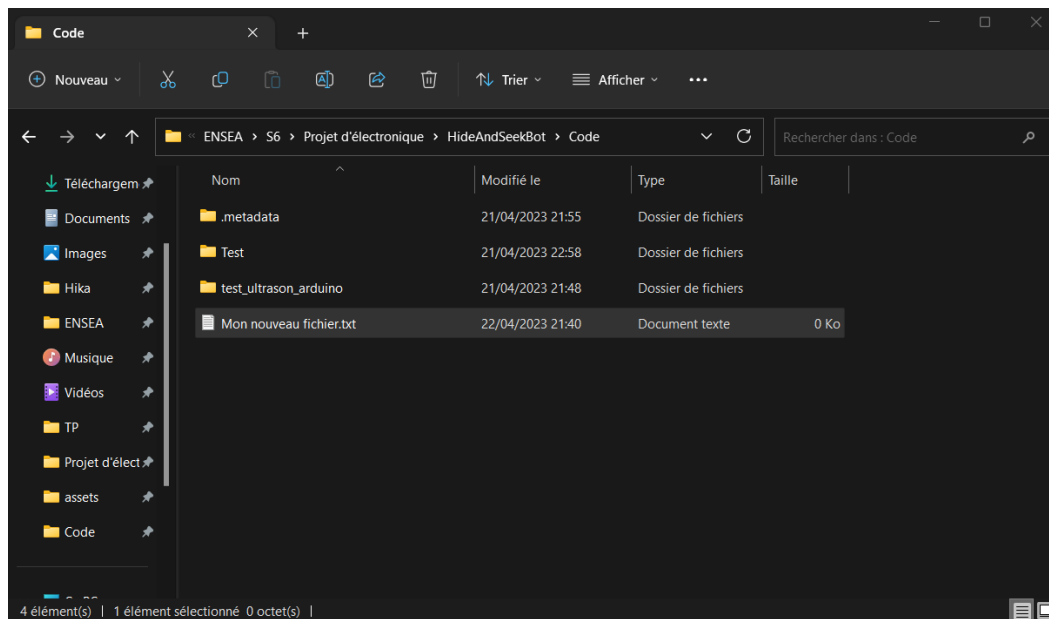
Et voilà ! Vous avez maintenant accès au dossier du projet sur votre ordinateur !



III – Apporter des modifications à votre projet

Dans le repository du projet vous devriez avoir accès au fichier « Instructions commandes.txt » qui récapitule en gros ce que je vais expliquer par la suite. Le contenu de ce fichier est également disponible dans la partie V.

3.1) Ajoutons de nouveaux fichiers à l'intérieur du dossier du projet. Créer le fichier que vous voulez, pour l'exemple, j'ai créé un fichier texte « Mon nouveau fichier.txt » :



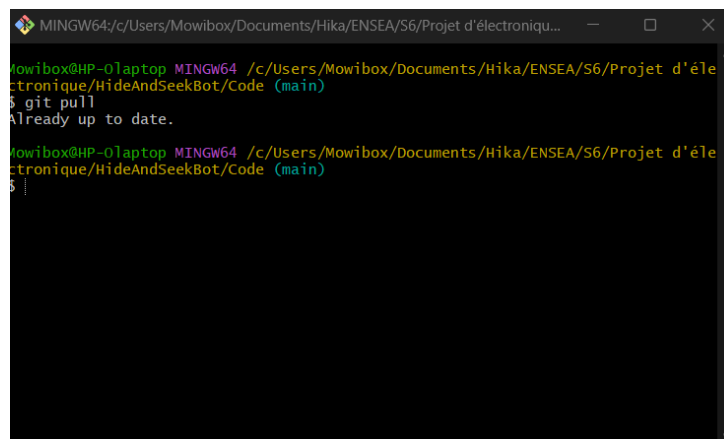
Puisque c'est dans votre ordinateur, seul vous y avez accès pour le moment, mais cela serait sympa d'en faire profiter tout le monde n'est-ce-pas ?

3.1) Ouvrir l'invite de commandes Git (Git Bash Here) au bon emplacement.

Avant d'envoyer nos propres modifications il faut vérifier que la version du repository sur notre pc est à jour car une autre personne du groupe aurait bien pu modifier le projet sur le serveur.

3.2) Taper la commande :

```
git pull
```



On peut lire 'Already up to date' : Apparemment, la version de mon pc était celle du serveur donc tout va bien dans le meilleur des mondes, nous allons pouvoir rajouter notre fichier.

3.3) Taper la commande :

```
git add *
```

Cette commande va initialiser l'ajout sur le serveur de tous les dossiers qui ont été modifiés.

3.4) Taper la commande :

```
git commit -m « Raison de la modification »
```

Lorsque l'on modifie des fichiers, cette commande permet d'afficher un message à côté de celui-ci, comme dans l'image de la sous-partie suivante où j'ai écrit : `git commit -m « C'était un test »`

3.5) Taper la commande :

```
git push
```

Et lorsque l'on va sur Github, une surprise nous attend :



Comme ce n'était qu'un test, nous allons maintenant enlever le fichier du serveur :

3.6) Refaire les étapes 3.1) et 3.2) puis taper à l'emplacement du fichier :

```
git rm « Nomdufichier.extension »
```

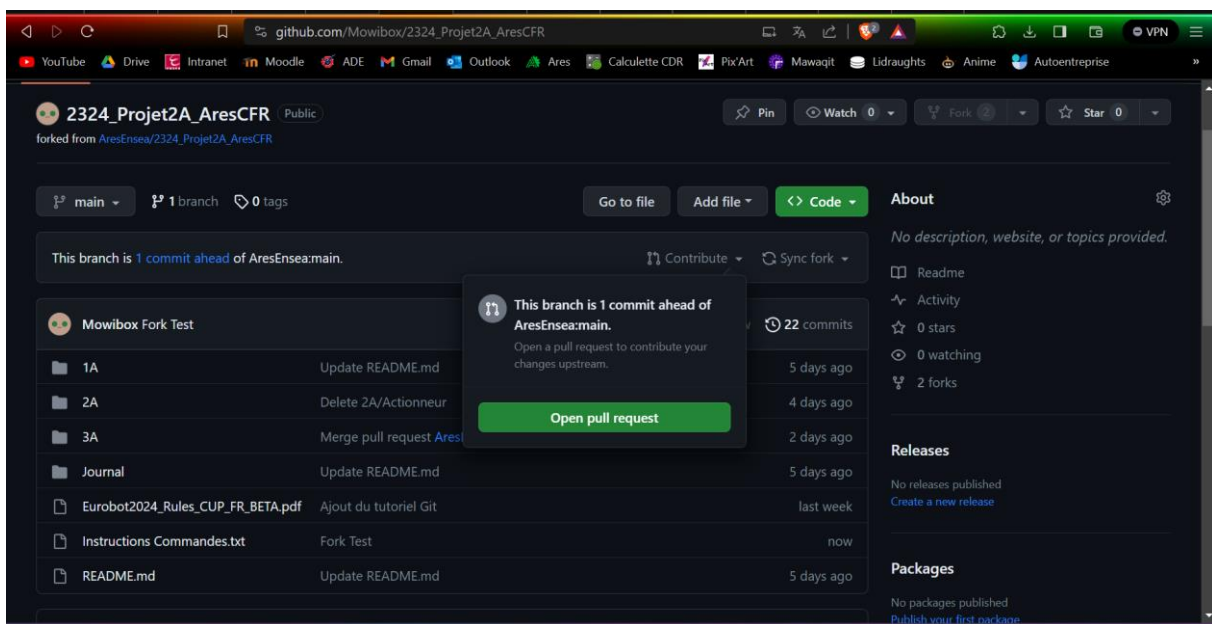
```
Mowibox@HP-0laptop MINGW64 /c/Users/Mowibox/Documents/Hika/ENSEA/S6/Projet d'éle  
ctronique/HideAndSeekBot/Code (main)  
$ git rm "Mon nouveau fichier.txt"  
rm 'Code/Mon nouveau fichier.txt'
```

3.7) Refaire les étapes 3.4) et 3.5). Voilà ! Plus de fichier dans le Github !

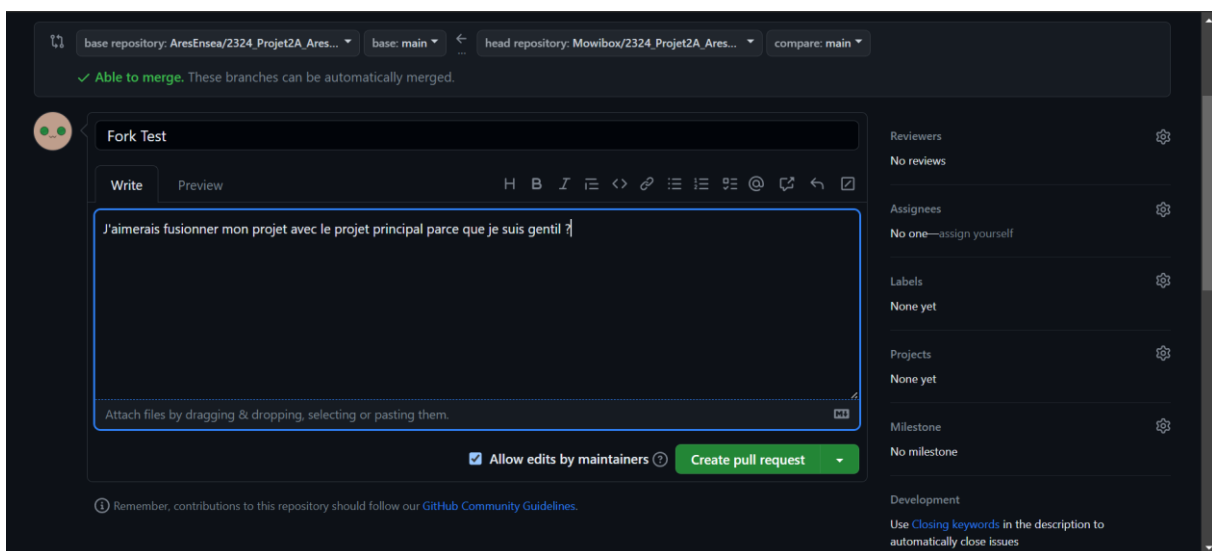
IV – Fusionner votre Fork avec le Main

Vous avez réalisé toutes les modifications qu'il faut à votre fork, c'est bien, mais le repository principal (le main) est resté identique pendant toutes vos manipulations (Le fork est une copie du main, cela permet d'éviter les accidents du style « oups, j'ai supprimé tous les dossiers par accident »). Il faut maintenant demander au chef de projet la fusion de vos modifications avec le repository principal.

4.1) Rendez-vous sur le github de votre projet fork et appuyez sur « Contribute » puis « Open pull request »



4.2) Écrivez un message qui explique les modifications que vous avez apporté au projet initial puis appuyez sur « Create pull request »



Un des modérateurs du projet principal examinera les modifications et une fois cela de fait, il peut accepter de « merge » votre fork avec le main, ce qui ajoute vos changements au projet principal.

V – En bref

ATTENTION : IL faut créer une fork puis la cloner et non cloner le main, sinon vous ne pourrez pas faire de modifications !

La première étape :

```
git pull (Met à jour les dossiers en fonction de ceux sur le serveur)
```

Pour ajouter des changements :

```
git add *  
git commit -m "Commentaire sur la modification apportée"  
git push
```

Pour supprimer des fichiers (par exemple un txt) :

```
git rm "Nom du fichier.txt"  
git commit -m "Commentaire sur la modification apportée"  
git push
```

D'autres commandes sont disponibles sur :

https://www.hostinger.fr/tutoriels/commandes-git#Git_config