



Suivi projet MAMI

1- Sur GitHub, créez un rep et ajoutez-y les membres de votre équipe.

Création d'un nouveau repo et ajout des différents collaborateurs


Create a new repository

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

Repository template
Start your repository with a template repository's contents.


No template ▾


Owner * **Repository name ***

 maxousql ▾ /

Great repository names are short and memorable. Need inspiration? How about **furry-lamp**?

Description (optional)

☒  **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.](#)

.gitignore template: None ▾

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

...or create a new repository on the command line

```
echo "# ProjetMAMI" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin git@github.com:maxousql/ProjetMAMI.git
git push -u origin main
```

```
maxla@LAPTOP-B32AL1PL MINGW64 ~
$ cd Desktop/

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop
$ cd ESGI/B2/Versioning-GIT/
Cours/          GitHub Cours/      Projet final MAMI/

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop
$ cd ESGI/B2/Versioning-GIT/Projet\ final\ MAMI/

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop/ESGI/B2/Versioning-GIT/Projet final MAMI
$ echo "# ProjetMAMI" >> README.md

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop/ESGI/B2/Versioning-GIT/Projet final MAMI
$ git init
Initialized empty Git repository in C:/Users/maxla/Desktop/ESGI/B2/Versioning-GI
T/Projet final MAMI/.git/

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop/ESGI/B2/Versioning-GIT/Projet final MAMI
(master)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

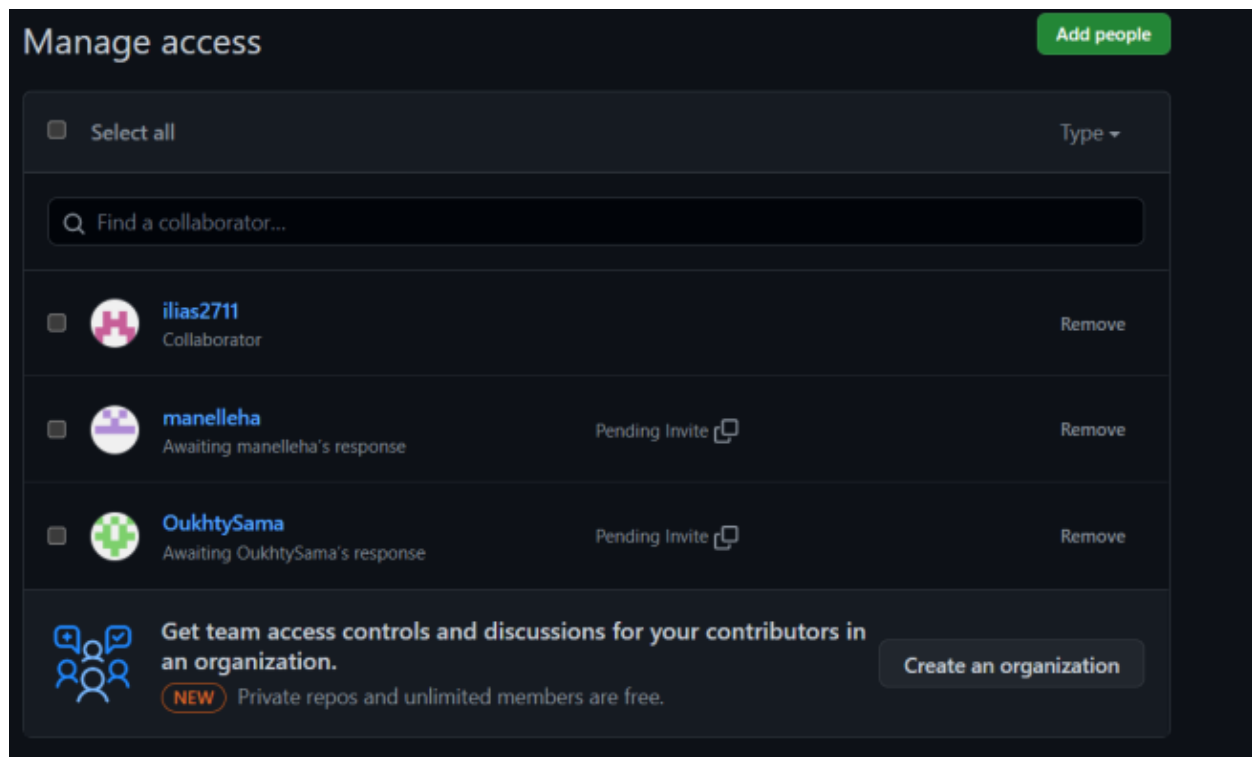
maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop/ESGI/B2/Versioning-GIT/Projet final MAMI (master)
$ git commit -m "first commit"
[master (root-commit) 01cccfa] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop/ESGI/B2/Versioning-GIT/Projet final MAMI (master)
$ git branch -M main

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop/ESGI/B2/Versioning-GIT/Projet final MAMI (main)
$ git remote add origin git@github.com:maxousql/ProjetMAMI.git

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop/ESGI/B2/Versioning-GIT/Projet final MAMI (main)
$ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 237 bytes | 237.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:maxousql/ProjetMAMI.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

maxla@LAPTOP-B32AL1PL MINGW64 ~/Desktop/ESGI/B2/Versioning-GIT/Projet final MAMI (main)
$
```



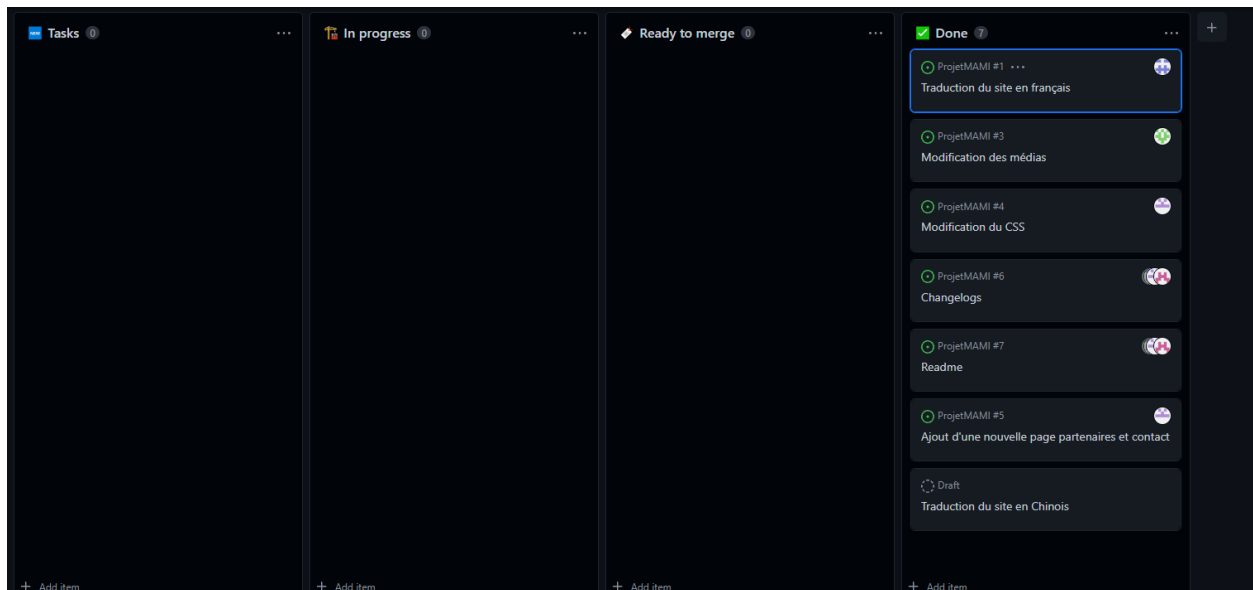
2- Récupérez le code d'un site web statique contenant plusieurs pages.

Téléchargement du template 34-wow.

Récupération du projet par les collaborateurs avec git clone + l'url du projet

3- Décrivez le workflow vu en cours et adaptez-le au travail suivant & effectuer sur votre projet :

- Traduisez tout le site : Ilias
- Modifiez le CSS global : Manelle
- Ajoutez/Modifiez le contenu de type media : Allan
- Ajoutez des nouvelles pages : Maxime



Création des branches

```
git branch
```

```
git checkout -b nouvelle_branche_locale nom_du_dépôt_distant/nouvelle_branche
```

→ créer une branche locale basée sur le contenu de la branche distante.

et push

```
git add .
```

```
git commit -m "feat: add new branch"
```

```
git pull
```

```
git push -u origin <name branch>
```

Bonus :

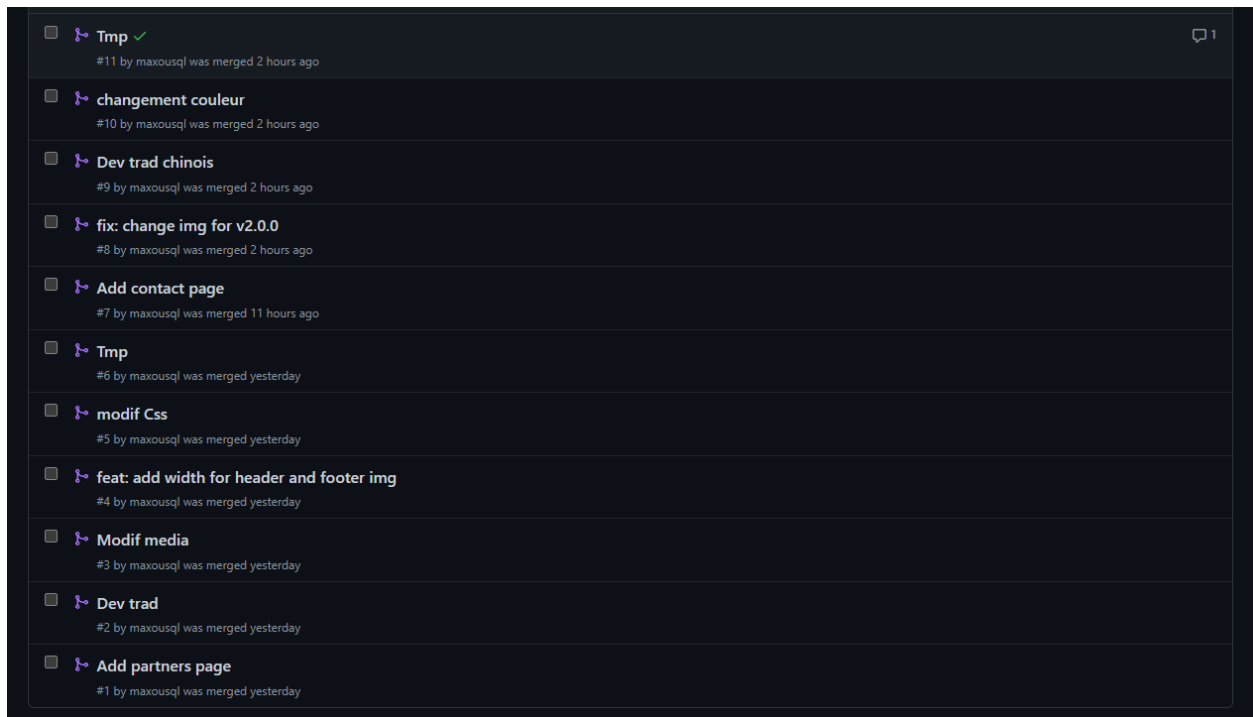
Formatez vos messages de commits, à l'aide des articles donnés.



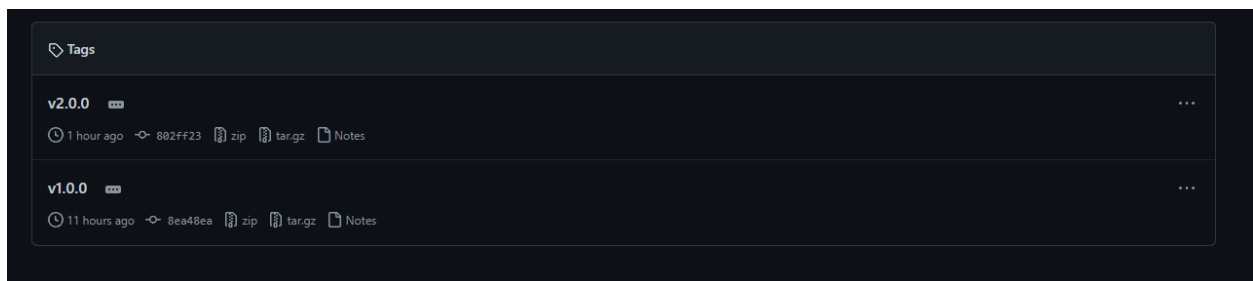
4- Une fois le travail termine et selon le workflow décrit précédemment mettez en commun le code et versionnez votre projet. A partir des messages de commits, créez un fichier changelog.md.

Pour le versioning : recherchez la commande `git tag` . Décrivez ce que représente un tag.

Nous avons rencontré des problèmes lors du premier merge, c'est pourquoi nous avons décider d'utiliser une branch temporaire (tmp) afin de pouvoir effectuer nos différents merge.



Une fois la branch temporaire fonctionnel et à jour, nous l'avons merge sur notre branch main.



git tag : Les tags sont généralement utilisés pour capturer un point de l'historique utilisé pour une version marquée.

Les tags légers et les tags annotés diffèrent dans la quantité de métadonnées associées qui sont stockées. Les tags annotés sont stockés comme des objets complets dans la base de données Git.

Nous avons donc utilisés uniquement des tags annotés. Pour cela nous avons effectué la commande suivante : `git tag -a v1.4 -m "v1.0.0"`

5- Répétez la démarche précédente et versionnez à nouveau.

Nous avons donc apporté de nouvelle modification à notre site web en partant de la v1.0.0 afin de pouvoir créer une version 2.0.0.

6- Déploiement (Bonus) :

Nous avons créer un compte sur netlify.com via GitHub.

Suite à cela nous avons générer le déploiement automatique du site web vers netlify.com en fonction de notre branch main.

Le site en donc lié à notre branch main, donc à chaque fois que nous allons merge de nouvelle version sur cette branch cela sera automatiquement actualisé sur l'hebergeur.

Problème rencontré :

Nous avons créer un premier dépôt mais lors du premier merge nous avons eu un conflit important que nous n'avons pas réussi à résoudre. Cela a complètement corrompu les fichiers locaux de Manelle ainsi que Git. C'est pourquoi nous avons décider de recréer un dépôt et de merge cette fois-ci en utilisant une branch temporaire.

Grâce à cette méthode nous n'avons rencontrés aucun autres problèmes.