

# AIDE pour récupérer le projet PPE3 NOTAGAME sous GitHub

## Forker un Repo(sitory) ou dépôt

Un *fork* est une copie d'un dépôt. Forker un dépôt vous permet d'expérimenter librement des modifications sans toucher au projet original.

Plus communément, les forks sont utilisés soit pour proposer des modifications sur le projet de quelqu'un d'autre **ou pour utiliser le projet de quelqu'un d'autre comme point de départ.**

**Utiliser le projet de quelqu'un d'autre comme point de départ : ici le projet existant PPE3NOTAGAME**

## Créer un compte GitHub pour le dépôt global du projet

Un seul compte GitHub suffit pour centraliser tout le code source du PPE3. Git et Github appellent cela un repository ou "repo" pour faire court, un répertoire numérique ou un espace de stockage où vous pouvez accéder à votre projet, ses fichiers, et toutes les versions de ses fichiers que Git sauvegarde.

Premièrement, vous devrez vous enregistrer pour disposer d'un compte sur GitHub.com. C'est aussi simple que de s'enregistrer sur n'importe quel autre réseau social. Conservez l'e-mail que vous avez choisi à portée de main ; nous en aurons besoin de nouveau.


Vous pourriez vous arrêter là et github fonctionnerait bien. Mais si vous voulez travailler sur votre projet sur votre ordinateur local, vous devez avoir installé Git. En fait, Github ne fonctionnera pas sur votre ordinateur local si vous n'installez pas Git.

Retournons sur GitHub.com et cliquez sur la petite icône de texte à côté de votre nom d'utilisateur. Ou allez vers la nouvelle page repository si toutes les icônes sont les mêmes. Donnez à votre dépôt un nom court et mémorisable. Allez-y et rendez-le public, pourquoi cacher votre tentative d'apprendre Github !.

Ne vous inquiétez pas de cliquer sur le bouton radio à côté intitulé "Initialize this repository with a README." Un fichier Readme est généralement un fichier texte qui explique sommairement le projet. Mais nous pouvons produire localement notre propre fichier **Readme** pour l'entraînement.

### Create a new repository

A repository contains all the files for your project, including the revision history.

Owner:  ChristopheDucamp / Repository name:

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

Description (optional):

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

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

☐ **Initialize this repository with a README**  
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None**

**Create repository**

Cliquez sur le bouton vert "Create Repository" et c'est fait. Vous avez maintenant un espace en ligne pour votre projet restant à faire vivre.

## Forkez un repository

Forker un repository est un processus très simple en deux étapes. Nous avons créé un repository pour vous entraîner.


1. Sur GitHub, naviguez vers le repository PPE3NOTAGAME
2. Dans le coin en haut et à droite de la page, cliquez sur **Fork**

Voilà, c'est fait ! Vous disposez désormais d'un *fork* du repository original **fderobien/PPE3NOTAGAME repository**

## Maintenez votre fork synchronisé

### Étape 1 : Installer et Paramétrer Git

Si ce n'est déjà fait, vous devez d'abord installer et régler Git. N'oubliez pas non plus de régler l'authentification vers GitHub à partir de Git.

1. Ouvrir le Terminal Git BASH sur votre ordinateur local 
2. Créer un répertoire qui va récupérer votre projet PPE3 (mkdir nomduDossier)
3. Aller dans ce répertoire (cd nomduDossier)


4. Faire un git init pour configurer ce dossier en tant que dossier GIT

## Étape 2 : Créer un clone local de votre fork

À ce stade, vous avez un fork du repository PPE3NOTAGAME, mais vous n'avez pas les fichiers dans ce repository **sur votre ordinateur**. Créons localement **un clone** de votre fork sur votre ordinateur.

1. Sur VOTRE GitHub, naviguez vers votre fork du repository PPE3NOTAGAME.
2. Dans la barre latérale de droite de votre page du repository du fork, cliquez sur l'icône fléchée pour copiez l'URL de clone pour votre fork.



3. Sur votre ordinateur, Ouvrez le Terminal **GIT BASH** 
4. Tapez **git clone**, et collez l'URL que vous avez copiée à l'étape 2. Cela ressemblera à cela, avec votre nom d'utilisateur GitHub au lieu de **YOUR-USERNAME**: **git clone https://github.com/YOUR-USERNAME/PPE3NOTAGAME**
5. Pressez la touche **Retour**. Votre clone local sera créé.

```
git clone https://github.com/YOUR-USERNAME/PPE3NOTAGAME
```

```
Cloning into `PPE3NOTAGAME`...
```

```
remote: Counting objects: 10, done.
```

```
remote: Compressing objects: 100% (8/8), done.
```

```
remote: Total 10 (delta 1), reused 10 (delta 1)
```

```
Unpacking objects: 100% (10/10), done.
```

Vous avez désormais une copie locale de votre fork du repository PPE3NOTAGAME!

## Étape 3 : Travailler en local puis faire un COMMIT sur GitHub (origin)

Vous avez forké avec succès un dépôt, mais vous avez maintenant plein d'autres choses à faire :

### Faire des modifications en local et les COMMITEZ

Vous travaillez sur vos fichiers en local et vous faites des commits au fur et à mesure de votre avancement. On va le TESTER

- Prendre un fichier existant et faites une modification mineure.

- Git status : ce fichier apparaîtrait en rouge car il a été modifié
- Git add NomDuFichier
- Git commit -m « modif du fichier pour tester » : vous venez de faire « une photo » de ce fichier en local

## Plus sur les commits

Pensez à un *commit* comme un instantané de votre projet - code, fichiers, tout - à un instant t. Après votre premier commit, git ne sauvegardera que les fichiers qui ont été modifiés, d'où le gain d'espace.

**Attention :** git fera de son mieux pour compresser vos fichiers, mais les gros fichiers et les binaires peuvent amener un dépôt à devenir obèse et peu maniable. Essayez d'éviter de committer des choses comme des fichiers compressés (zips, rars, jars), du code compilé (fichiers object, bibliothèques, exécutables), sauvegardes de database et fichiers média (flv, psd, musique, films)

## Pousser des commits : PUSH

Une fois que vous avez produit un ou quelques commits sur votre dossier local et que vous voulez les pousser vers votre dépôt REMOTE distant (ici GitHub), vous faites ça de la même façon que vous le feriez avec un dépôt normal :

```
git push origin master
```

```
# Pousse Les commits vers votre dépôt remote (distant) stocké sur GitHub
```

Sur GitHub vous pouvez tester votre PUSH en ouvrant le fichier. Votre modification est bien présente...

## Récupérer des commits : PULL

Si un autre membre du projet a fait lui aussi un *commit* très important pour la suite, vous devez récupérer ces fichiers dans votre dépôt local.

Aussi il vous sera nécessaire de faire un GIT PULL. Ainsi vous fusionnerez vos fichiers avec les fichiers modifiés par vos collègues (Attention aux conflits).

## Fêtons ça !

Bravo ! Vous êtes prêt à développer la V2 du projet PPE3 NOTAGAME