**GIT ET GITHUB**

**Apprenez les Concepts de bases du gestionnaire**

**de version le plus populaire du monde   
  
par , *MAX AURIOL BILEY***



# Introduction La **gestion de versions** (en [anglais](https://fr.wikipedia.org/wiki/Anglais) : *version control* ou *revision control*) consiste à gérer l'ensemble des versions d'un ou plusieurs fichiers (généralement [en texte](https://fr.wikipedia.org/wiki/Fichier_texte)). Essentiellement utilisée dans le domaine de la [création de logiciels](https://fr.wikipedia.org/wiki/Programmation_informatique), elle concerne surtout la gestion des [codes source](https://fr.wikipedia.org/wiki/Codes_source).

Cette activité étant fastidieuse et relativement complexe, un appui logiciel est presque indispensable. À cet effet, il existe différents [logiciels de gestion de versions](https://fr.wikipedia.org/wiki/Logiciel_de_gestion_de_versions) qui, bien qu'ayant des concepts communs, apportent chacun leur propre vocabulaire et leurs propres usages.   
  
C’est de ce contexte, que nous aborderons le gestionnaire de version le plus polaire du monde Git et son interface graphique GitHub, afin que vous vous familiarisiez facilement avec le concept.

Initiation à Git et Github   
  
Bien que similaire de par leurs noms git et gihub ont deux mission différentes.   
L’un est un logiciel de versionning , c’est à dire qu’il permet d’avoir plusieurs version d’un même fichier ce qui permet au dévéloppeur de disposer exactement de la copie du fichier dont ils sont besoins facilitant ainsi la collaboration et le suivis des projets logiciel . L’autre , est un Hébergeur qui stocke l’ensemble des dépotes Git du monde et seras la parcelle principal avec la quelle vous suivrez l’ensemble de vos projet .   
  
1. L’ interface Github

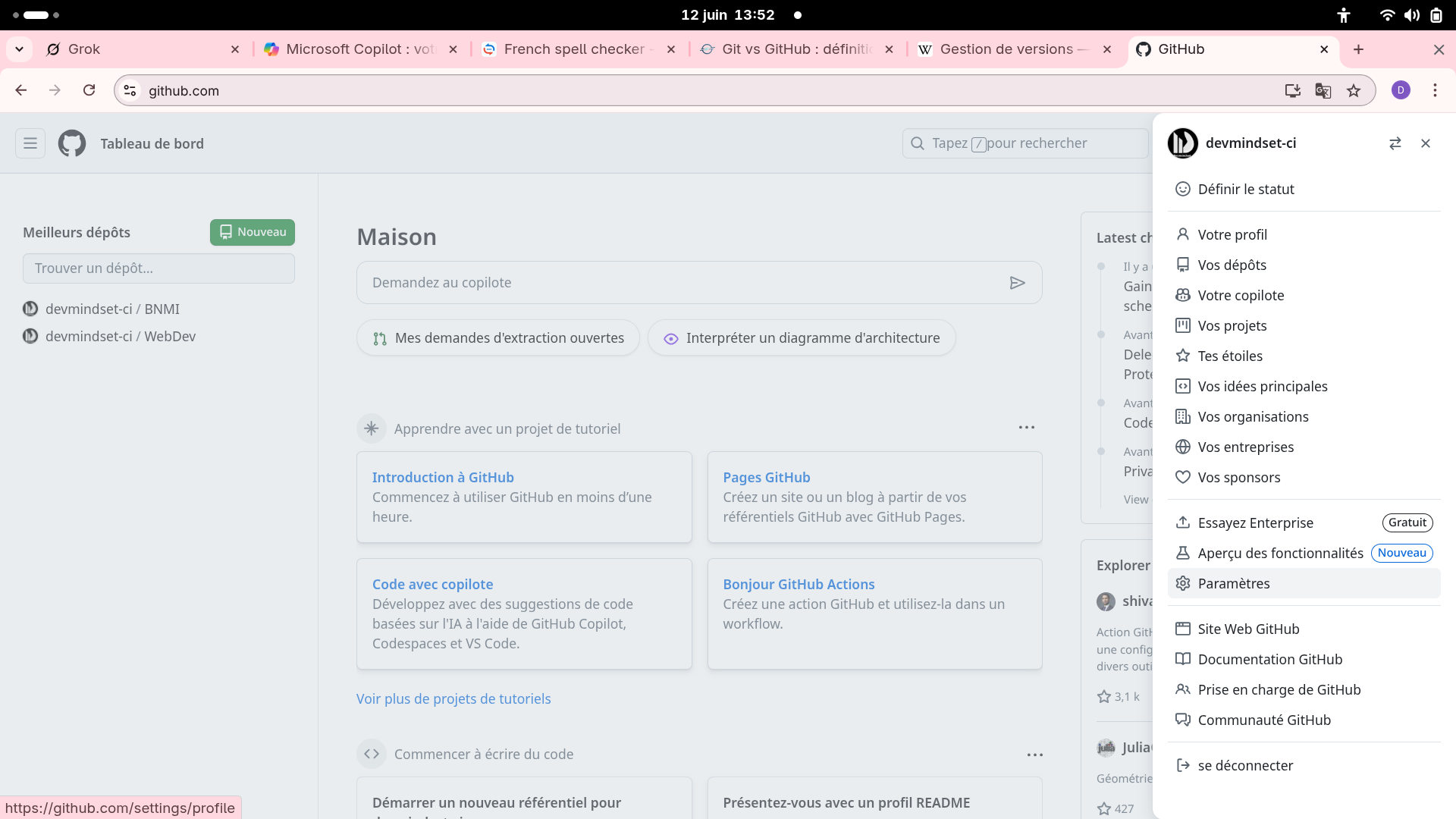
Visitez le lien ci-dessous pour vous familiarisez avec l’interface GITHUB   
  
>>>> <https://github.com/>  
  
  
2. Configuration de Base   
  
 2.1. Crée un compte Github

Commençons déjà par créer un Compte sur Github  
  
>>> [Cliquez-ici](https://github.com/signup?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home)  
  
NB : Si vous possédez un e-mail du fournisseurs de messagerie **gmail** utilisez le pour créer un compte rapidement   
  
***Téléchargez Git Bash***  
  
Pour windows   
*>>>* [*Cliquez-ici*](https://git-scm.com/downloads/win)

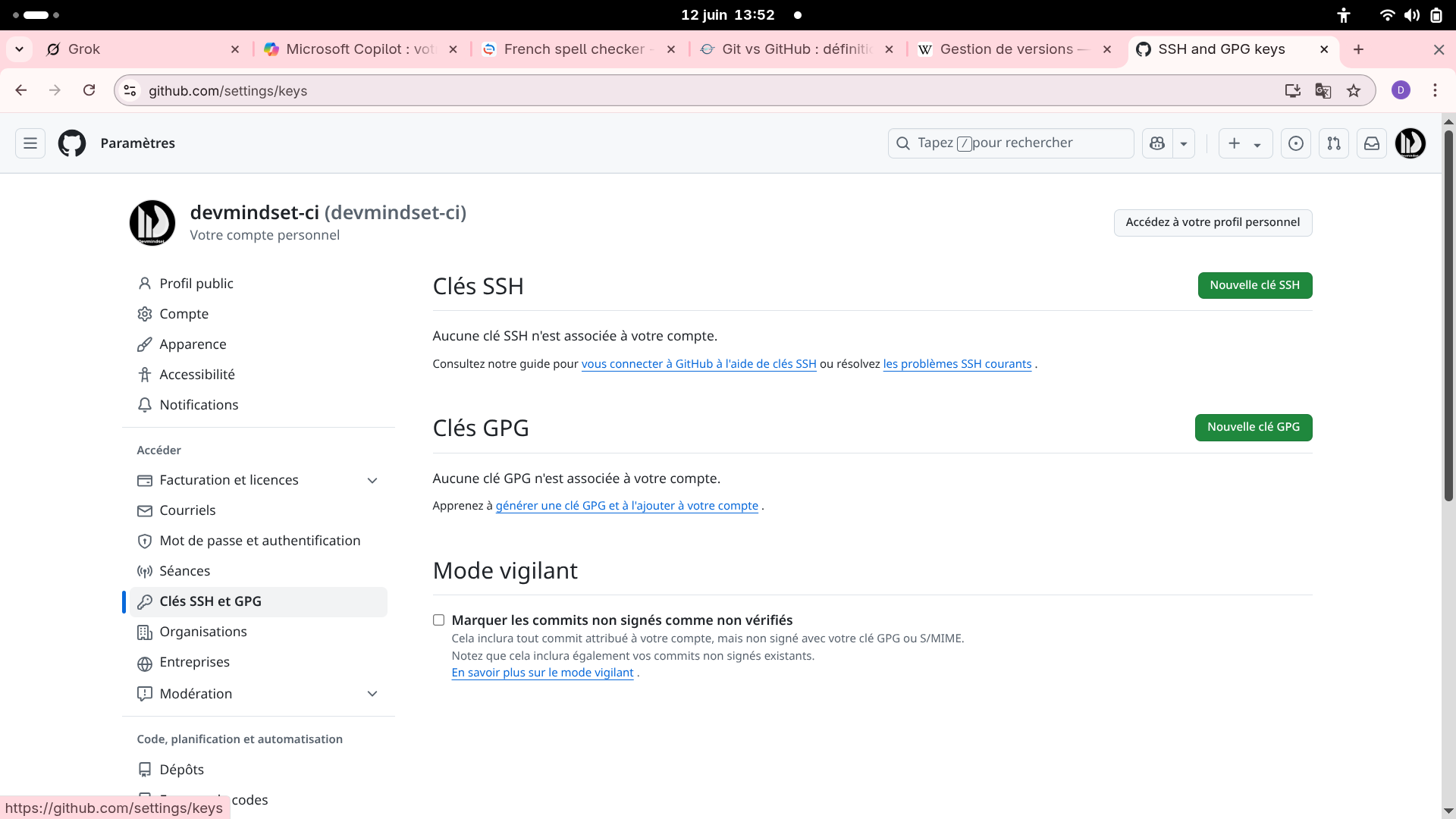
Pour linux   
*>>>* [*Cliquez-ici*](https://git-scm.com/downloads/linux)   
  
*NB : pour certaine distributions Linux , les commandes d’installation varie d’une version à l’autre soigné donc très spécifique sur la façon dont vous cherchez les commandes d’installation pour git .*   
   
 **2.2. Configuration de compte et connexion via SSH**   
  
Ouvrez votre terminal Git Bash et entré succinctement les commande suivants :   
  
**>>> git config --global user.name "Votre Nom"**  
  
**>>> git config --global user.email "**[**votre.email@github.com**](mailto:votre.email@example.com)**"**

Vérifiez la config avec cette commande   
  
**>>> git config –list  
  
Configuration permanente avec SSH**Générer une clé ssh  
 **>>> ssh-keygen -t rsa -b 4096 -C "**[**votre.email@github.com**](mailto:votre.email@github.com)**"**Ajouté la clé SSH à l’agent SSH **>>> eval "$(ssh-agent -s)"**

**ssh-add ~/.ssh/id\_rsa**  
  
copiez la clé publique   
>>> cat ~/.ssh/id\_rsa.pub  
  
**>>> Retourner sur votre compte github et accédez au paramètre entouré en rouge**



**>>> Une fois dans les paramètre cliquez sur Clés SSH et GPG . Puis nouvelle clé SSH .  
>>> Coller ensuite le fichier cat ~/.ssh/id\_rsa.pub dans la zone de texte .**

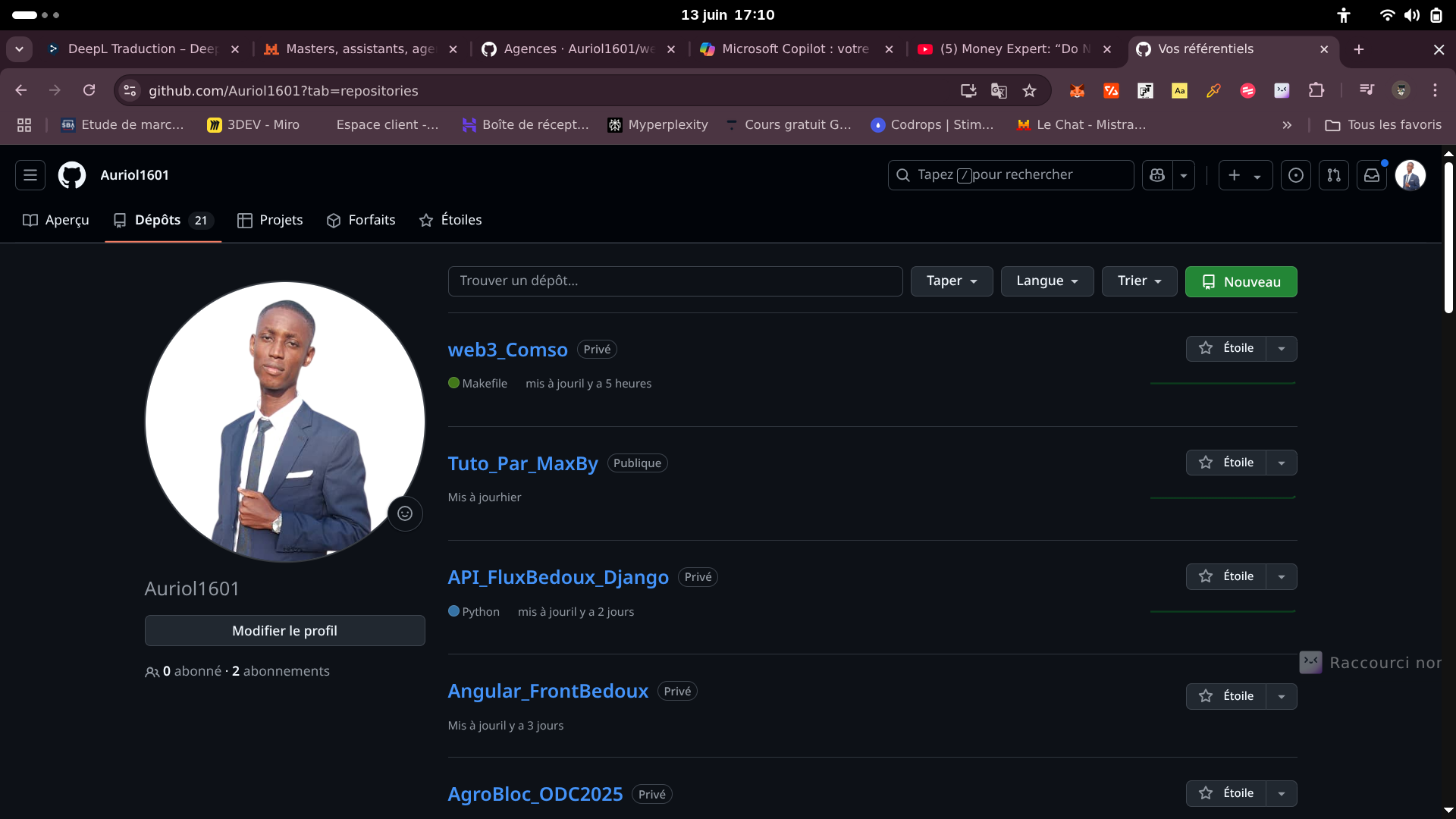


*Maintenant que vous avez un compte , passons maintenant au vive du sujet .*  
  
3. Les Commandes de base

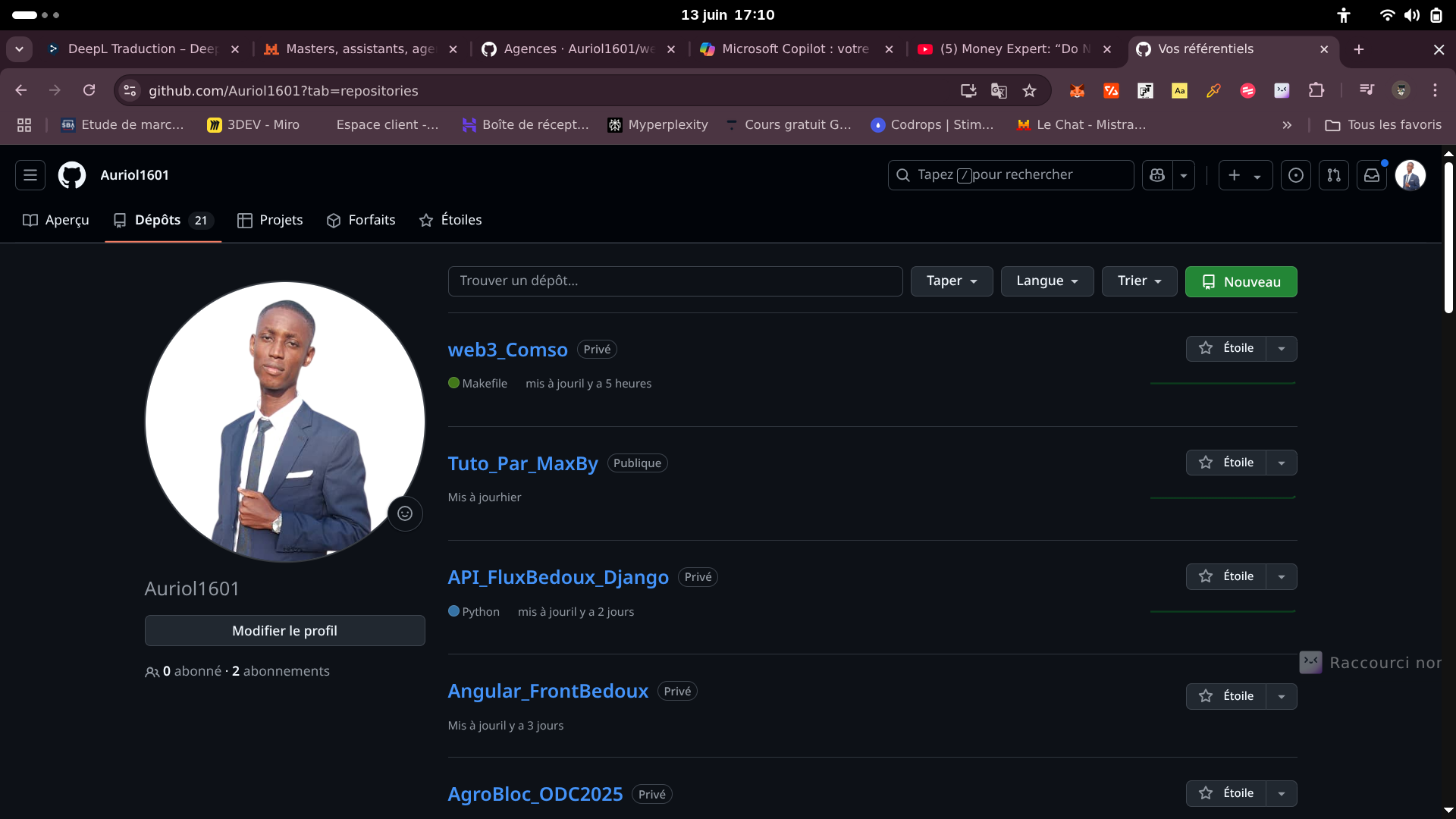
3.1. Le Credo ACP **( add commit push** )   
  
Le crédot *ACP[[1]](#footnote-2)* est un ensemble de commande git avec le quel vous devriez rapidement vous familiariser pour envoyé votre projet sur la plateforme hébergement ***GITHUB .*** *Pour ce faire nous allons créer un projet fictive et tester tous cela Rendez-vous sur Github.*

*Crée un dépôt sur github et cloner votre dépôt sur votre machine en local*

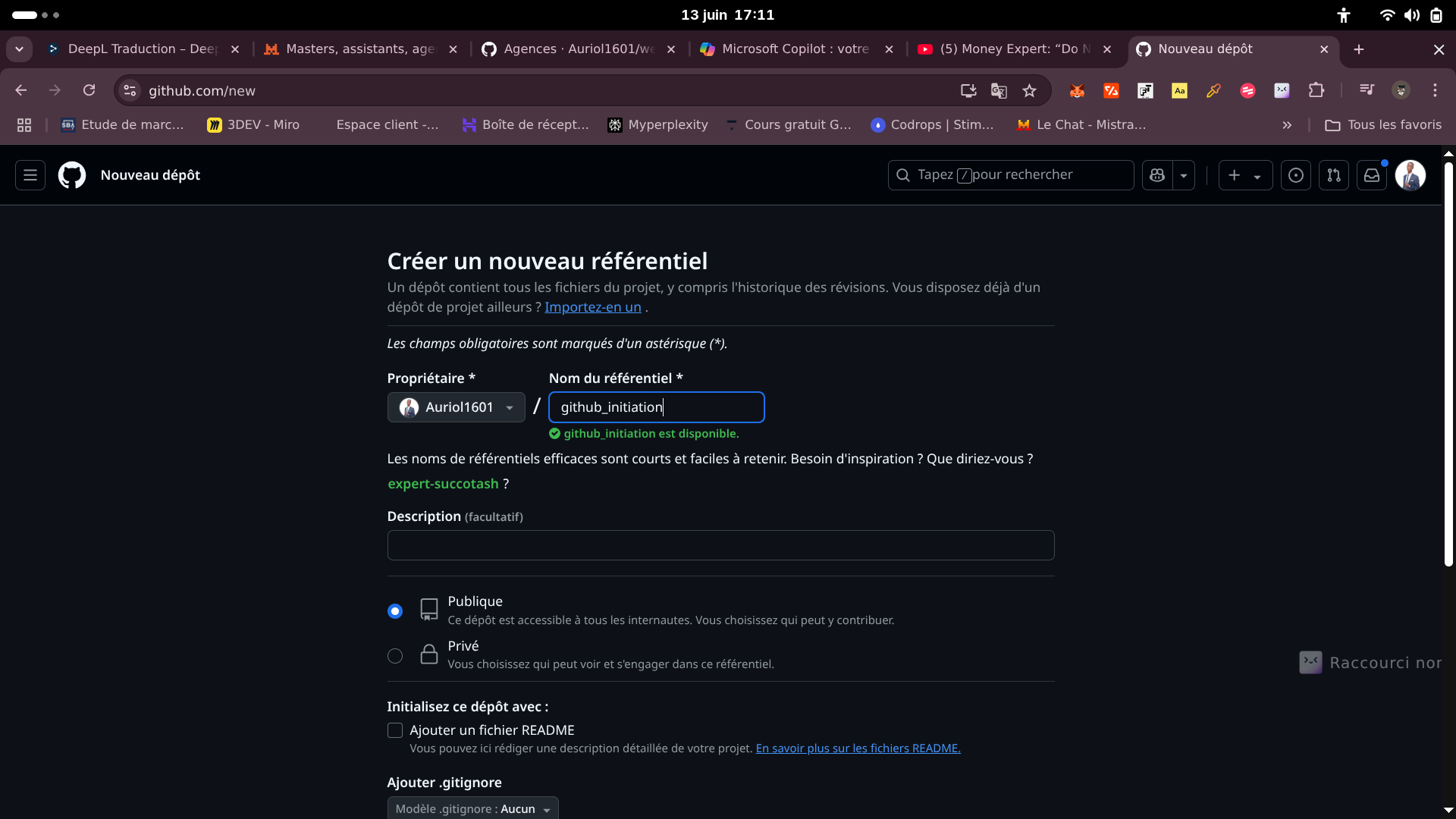
*1. Rendez-vous sur votre espace github   
2. Accédez au menu Encadrez en rouge sur l’image ci-dessous.*

**

*3.Cliquez sur Nouveau Encadrez en vert*

**

***Crée un dépôt depuis votre machine local et pousser le vers votre dépôt distant***

******

***NB :*** *définissez notre projet sur* ***Privé*** *pour que seulement vous accède au projet* ***github\_initiation*** *après ça création .*

3.2. Les commandes de base   
 revenir à un commit   
4. Gestion de Branche   
 4.1. Commande de Base utiliser en equipe   
5. Git Avancée   
  
6.Le Glossaire

Commit   
repository   
PullRequest   
Issues  
REAMDME   
   
Référence et Bibliographie

Gestion de version : https://fr.wikipedia.org/wiki/Gestion\_de\_versions  
 Alain Demenet Git Vs github : <http://jedha.co/blog/git-et-github-definitions-differences-utilite>  
  
---   
echo "# Tuto\_By\_MaxBy" >> README.md

git init

git add README.md

git commit -m "premier commit"

git branch -M main

git remote add origin git@github.com:Auriol1601/Tuto\_By\_MaxBy.git

git push -u origin main

1. Add Commit Push [↑](#footnote-ref-2)