CCI Nîmes

Présentation GitHub 

**Guide de survie**

Une image contenant logo, Graphique, Police, symbole

Description générée automatiquement

Réalisé et enseigné par

Jérôme Ferratier



Table des matières

Table des matières

[Table des matières 2](#_Toc147421142)

[1. GITHUB 3](#_Toc147421143)

[1.1. Création d’un nouveau compte 3](#_Toc147421144)

[1.2. Utilisation de SSH pour GitHub 3](#_Toc147421146)

[1.2.1. Pré-requis 3](#_Toc147421147)

[1.2.2. Création de la clé privé/publique SSH 3](#_Toc147421148)

[1.2.3. Insertion de la clé publique dans GitHub 3](#_Toc147421149)

[1.2.4. Configuration pour l'identification lors de commits 4](#_Toc147421150)

[1.3. Création d’un nouveau projet 4](#_Toc147421151)

[1.4. Partager un projet 4](#_Toc147421152)

1. GITHUB
   1. Création d’un nouveau compte

<https://github.com/>

Pour la création d’un nouveau compte :

- Attention utiliser une vrai adresse mail avec prenom.nom pour vous inscrire.

- Dans votre login renseigner bien votre prenom.nom.

* 1. Utilisation de SSH pour GitHub
     1. Pré-requis
* [Installer Git for Windows](https://git-scm.com/download/win)  
  Les paramètres d’install par défaut suffisent, mais vous pouvez tout de même choisir votre éditeur de texte préféré (Notepad++/Visual Studio Code/…).
  + 1. Création de la clé privé/publique SSH
* Ouvrir « Git Bash » (à partir du menu démarrer ou le lancer directement "C:\Program Files\Git\git-bash.exe").  
  ssh-keygen -t ed25519 -C "prenom.nom@cci.com"
* Je conseille de ne pas utiliser de mot de passe (appuyer sur Entrée dans rentrer de mot de passe crééra une clé sans mot de passe).
* Le générateur de clé va demander un chemin, appuyer sur Entrée (ça utilisera le chemin suggéré).
  + 1. Insertion de la clé publique dans GitHub
* Copier la clé publique dans votre presse-papier:  
  cat ~/.ssh/id\_ed25519.pub | clip
* Naviguer vers le [configurateur de communication avec clés SSH](https://github.com/settings/keys)
* Dans la textbox « Key », coller la clé publique. Normalement un titre sera automatiquement inséré. Terminez en appuyant sur « Add Key ».
  + 1. Configuration pour l'identification lors de commits
* Ouvrez une invite de commande (Win + R, cmd.exe ou ouvrir Git Bash), taper :  
  git config --global user.email "prenom.nom@cci.com"  
  git config --global user.name "Prenom Nom"
  1. Création d’un nouveau projet
* Dans Github faire :
  + Create a new repository
  + Choisir projet privé
  1. Partager un projet
* Dans GitHub faire :
  + Sélection de votre projet 🡪 Settings 🡪 Collaborators 🡪Add People
  1. Liste de commandes essentielles

| **Commande** | **Description** |
| --- | --- |
| git clone <url> | Clone un dépôt distant en local |
| git status | Affiche l’état des fichiers (modifiés, non suivis, etc.) |
| git add <fichier>  git add . | Ajoute un fichier à l’index pour le prochain commit  Ajoute tout le nouveau fichier à l’index pour le prochain commit |
| git commit -m "message" | Enregistre les modifications avec un message |
| git push | Envoie les commits locaux vers le dépôt distant |
| git pull | Récupère et fusionne les modifications du dépôt distant |
| git branch | Liste les branches locales |
| git checkout <branche> | Change de branche |
| git checkout -b <branche> | Crée une nouvelle branche et s’y place |
| git merge <branche> | Fusionne une branche dans la branche courante |
| git log | Affiche l’historique des commits |
| git diff | Montre les différences entre fichiers modifiés |
| git remote -v | Affiche les dépôts distants configurés |
| git stash | Met de côté les modifications en cours pour revenir à un état propre |
| git stash pop | Récupère les modifications mises de côté |
| git reset --hard | Annule toutes les modifications locales non commit |