

UNIVERSITÉ DE TECHNOLOGIE D'HAÏTI (UNITECH)

Faculté des Sciences de Génie et d'Architecture



Cours de Cybersécurité

TD No. 1- Systèmes

Nom & Prénom : Martchello Gaëthan Aimé

Niveau : II

Date : 16/01/2026

-Plan-

- Description des résultats de la tâche** **page 3**
- Résultats de l'exécution des commandes** **page 3**
- Conclusions** **page 7**

1- Description des résultats de la tâche

Objectif du TD

Ce TD avait pour objectif de maîtriser les commandes Git fondamentales pour la gestion de version et le travail collaboratif via GitHub. Les principaux apprentissages visés étaient :

- Initialiser et gérer des dépôts Git locaux
- Effectuer des opérations CRUD (Create, Read, Update, Delete) avec Git
- Synchroniser le travail local avec un dépôt distant sur GitHub
- Gérer la structure des projets avec plusieurs dossiers et fichiers

Démarche suivie

La démarche a été divisée en deux phases principales :

Phase 1 : Gestion du dossier "systeme"

Bash, cd TD, ls, cd systeme, git status, git add ., git commit -m "Ajout rapport1", git branch -M main, git push -u origin main

Phase 2: Gestion du dossier "image"

Bash, cd .., cd image, git status, git add ., git commit -m "Ajout les images td1", git branch -M main, git push -u origin main

Cette approche séquentielle a permis de traiter chaque composant du projet indépendamment tout en maintenant une organisation claire.

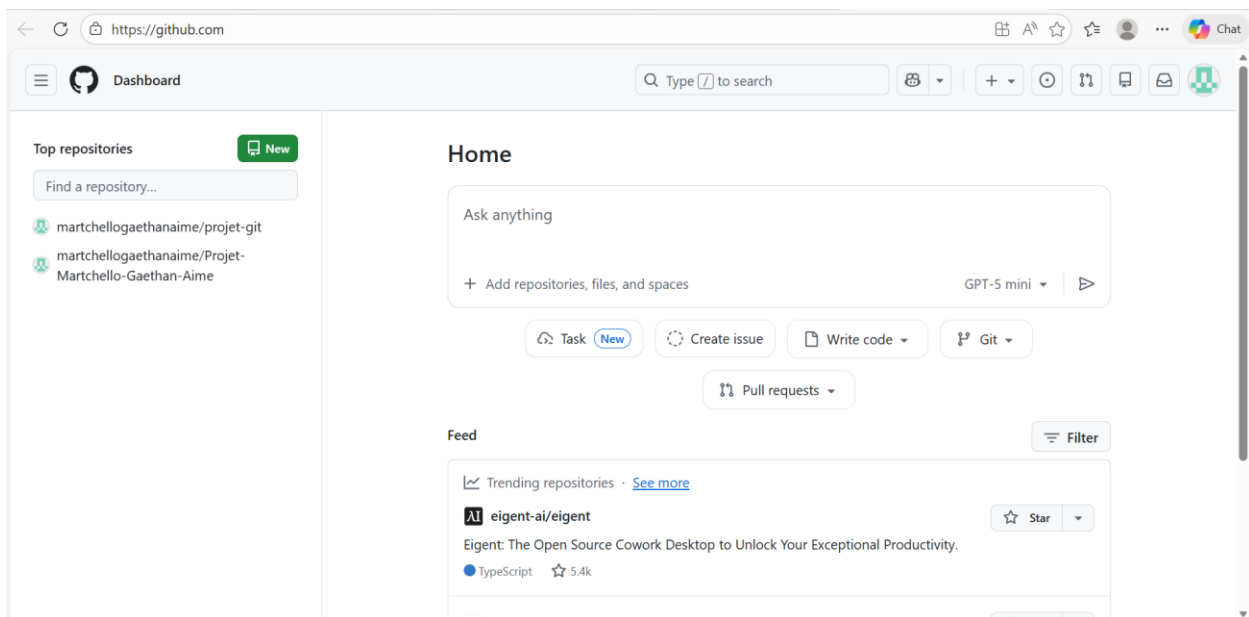
2- Résultats de l'exécution des commandes

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\WINDOWS\system32> git --version
git version 2.47.3.windows.1
PS C:\WINDOWS\system32>
```

1 Installation et vérification Version Git



2 Interface GitHub après inscription

```
PS C:\WINDOWS\system32> git config --global user.name "martchelloaethanaime"
PS C:\WINDOWS\system32> git config --global user.email "martchelloaethanaime@gmail.com"
PS C:\WINDOWS\system32>
```



3 Synchronisation GitHub et PowerShell

```

PS C:\WINDOWS\system32> git config --list
core.symlinks=false
core.autocrlf=true
core.fscache=true
color.interactive=true
color.ui=auto
pack.packsizelimit=2g
help.format=html
diff.astextplain.textconv=astextplain
rebase.autosquash=true
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
credential.helper=manager
http.sslbackend=schannel
http.schannelusesslinfo=false
credential.https://dev.azure.com.usehttppath=true
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
filter.lfs.clean=git-lfs clean -- %f
user.name=martchellogaethanaime
user.email=martchellogaethanaime@gmail.com
core.pager=cat
PS C:\WINDOWS\system32>

```

4 Vérification de la synchronisation avec git config --list

Name	Date modified
 .git	1/15/2026 10:54 AM
 README	1/15/2026 10:53 AM

5 Intégration du fichier masqué après la commande git init

```

PS C:\Users\HP\Desktop\projet-git2026> echo "Hello Git 2026!" > README.md
PS C:\Users\HP\Desktop\projet-git2026> git add README.md
PS C:\Users\HP\Desktop\projet-git2026> git commit -m "Mon TD Intra"
[main (root-commit) 48d9409] Mon TD Intra
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
PS C:\Users\HP\Desktop\projet-git2026> git status
On branch main
nothing to commit, working tree clean
PS C:\Users\HP\Desktop\projet-git2026> |

```

6 Ajout et commit des fichiers

```

PS C:\Users\HP\Desktop\projet-git2026> git log
commit 48d9409b8711bbc515506760359afa0d0f3d6ed0 (HEAD -> main)
Author: martchellogaethanaime <martchellogaethanaime@gmail.com>
Date: Thu Jan 15 10:53:33 2026 -0500

    Mon TD Intra
PS C:\Users\HP\Desktop\projet-git2026> |

```


7 Résultat de la commande git log

```

PS C:\Users\HP> C:\Windows\System32\OpenSSH\ssh-keygen.exe -t ed25519 -C "martchellogaethanaime@gmail.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (C:\Users\HP/.ssh/id_ed25519):
C:\Users\HP/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\HP/.ssh/id_ed25519
Your public key has been saved in C:\Users\HP/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:HnPIF/C5rXHrfXJTMNsKI+kiHScBRbmyEmPiZYLAnpU martchellogaethanaime@gmail.com
The key's randomart image is:
+--[ED25519 256]--+
| . . . o+ . |
|.. E . . o . |
|o.o . . + |
|.oo * . . o + o |
| . * o oS.=.o = |
| . . .o=++o.. o |
| . . =..o . |
| . o . . .o.o |
| . . . .+. |
+-----[SHA256]-----+
PS C:\Users\HP>

```

8 Générer une nouvelle clé SSH avec ssh-keygen



martchellogaethanaime

projet-intra-cybersecurite

projet-intra-cybersecurite is available.

Great repository names are short and memorable. How about [bug-free-fishstick](#)?

Description

Resume du projet

16 / 350 characters

2 Configuration

Choose visibility *

Choose who can see and commit to this repository

Public

Add README

READMEs can be used as longer descriptions. [About READMEs](#)

On

Add .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

No .gitignore

Add license

Licenses explain how others can use your code. [About licenses](#)

No license

Create repository

9 Création d'un new repository

6

```

PS C:\Users\HP\Desktop\projet-intra-cybersecurite> git commit -m "Ajout module"
[main 9b6dd66] Ajout module
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Module.txt
PS C:\Users\HP\Desktop\projet-intra-cybersecurite> git branch -M main
PS C:\Users\HP\Desktop\projet-intra-cybersecurite> git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 326 bytes | 163.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:martchellogaethanaime/projet-intra-cybersecurite.git
 096e3d2..9b6dd66  main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\HP\Desktop\projet-intra-cybersecurite> cd ..
PS C:\Users\HP\Desktop> git clone git@github.com:martchellogaethanaime/TD.git
Cloning into 'TD'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
Receiving objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
PS C:\Users\HP\Desktop> |

```

10 Ajout de module, commit et utilisation de git clone et de git push

```

PS C:\Users\HP\Desktop\TD\ReseauI> ls

Directory: C:\Users\HP\Desktop\TD\ReseauI

Mode                LastWriteTime         Length Name
----                -
d-----          1/15/2026   1:45 PM             image
d-----          1/15/2026   1:45 PM      presentation
d-----          1/15/2026   1:45 PM             projet

PS C:\Users\HP\Desktop\TD\ReseauI> |

```

11 Vérification de la création des sous-répertoires

3- Conclusions

Acquisitions principales

Maîtrise du workflow Git basique : add → commit → push

Compréhension des états Git : untracked, staged, committed

Gestion des branches : Utilisation de git branch -M main pour nommer la branche principale

Synchronisation distante : Configuration et utilisation de origin pour GitHub

Organisation de projet : Gestion séparée de différents composants (documents, images)

Difficultés rencontrées et solutions

- Configuration SSH pour GitHub : Échec de l'authentification SSH malgré une clé correctement générée. Format de clé invalide lors du copier-coller dans GitHub (espaces parasites). J'ai dû faire des recherches pour trouver la commande Get-Content pour pouvoir afficher la bonne clé SSH dans Powershell. J'ai appris l'importance du format exact dans les configurations cryptographiques
- Première connexion SSH : Acceptation du fingerprint de GitHub via yes (y)
- Dossier .git dans un sous-dossier : J'ai appris à faire la Suppression du .git interne pour maintenir un dépôt unique
- Push sans connexion internet : Distinction entre opérations locales (add, commit) et distantes (push, pull)

Objectifs atteints :

- Création et gestion de dépôts Git
- Synchronisation avec GitHub
- Gestion de multiples fichiers et dossiers
- Résolution des problèmes d'authentification

Compétences validées :

- Installation et configuration de Git
- Utilisation des commandes fondamentales
- Gestion des erreurs courantes
- Organisation de projet avec Git