



Faculté: Sciences Informatiques

TD No 1: Systèmes

Nom et Prénom: SAINT-JEAN Marc-Evenort

Professeur: Ismael Saint Amour

Niveau: 3ème Année

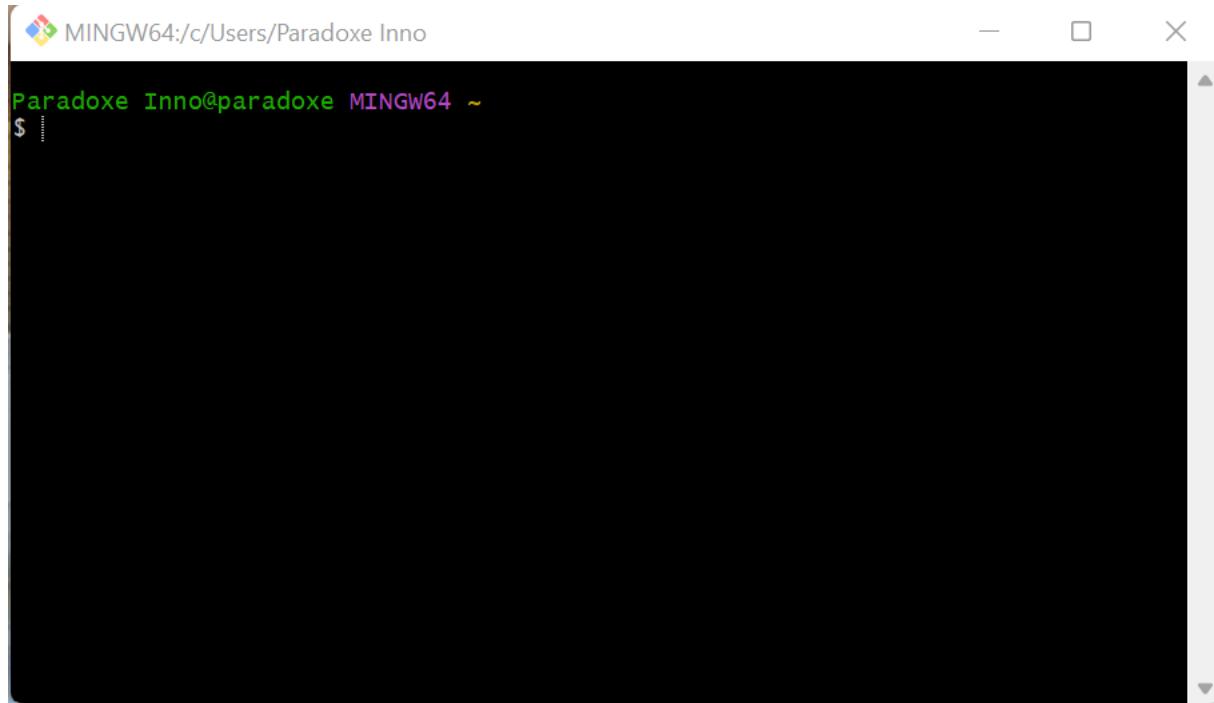
Date: Novembre 2025

Introduction:

Ce travail dirigé a pour objectifs de présenter et mettre en pratique les notions essentielles liées à l'utilisation de Git et GitHub.

- Comprendre les concepts fondamentaux de Git (Utiliser Git en local).
- Créer un compte GitHub et configurer une connexion SSH.
- Créer et cloner un dépôt distant.
- Organiser un projet dans un dépôt et héberger un rapport PDF.

A) Installation et Configuration de Git



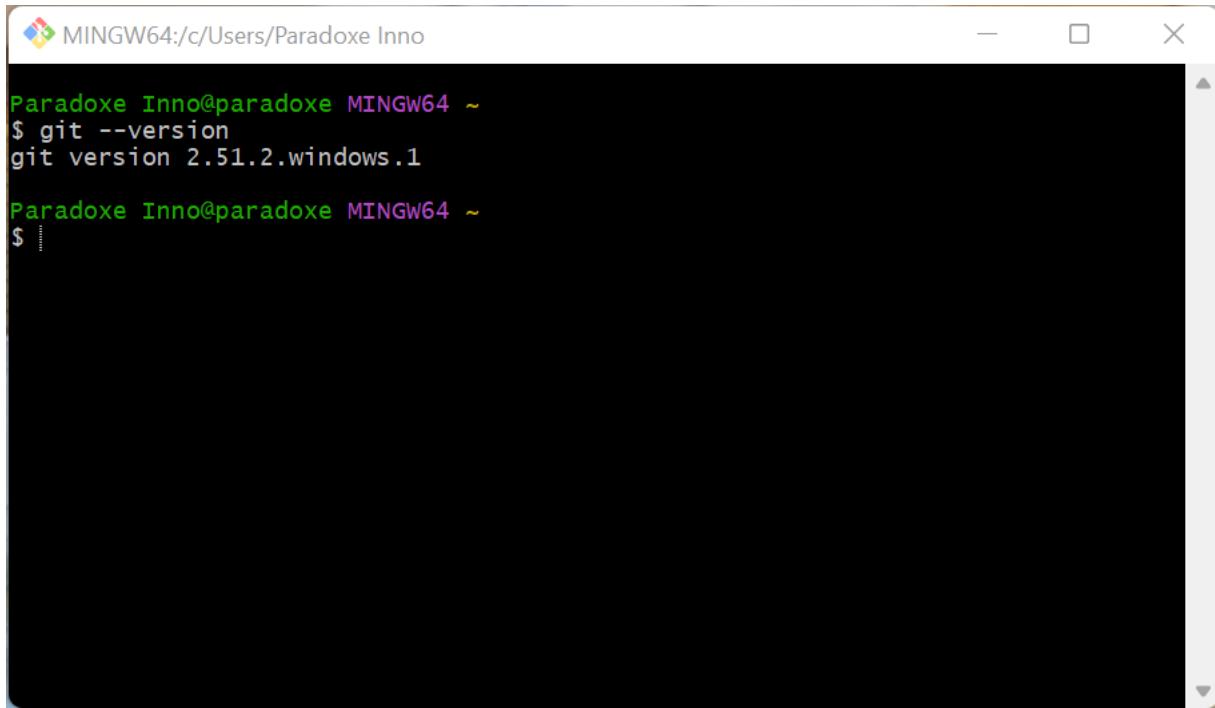
MINGW64:/c/Users/Paradoxe Inno

Paradoxe Inno@paradoxe MINGW64 ~

\$

- Après l'installation de Git sur mon ordinateur (windows), j'ai fait un double-clic sur Git bash (bascule) pour ouvrir le terminal.

B) Vérifier la version de Git



A screenshot of a terminal window titled "MINGW64:/c/Users/Paradoxe Inno". The window contains the following text:

```
Paradoxe Inno@paradoxe MINGW64 ~
$ git --version
git version 2.51.2.windows.1

Paradoxe Inno@paradoxe MINGW64 ~
$
```

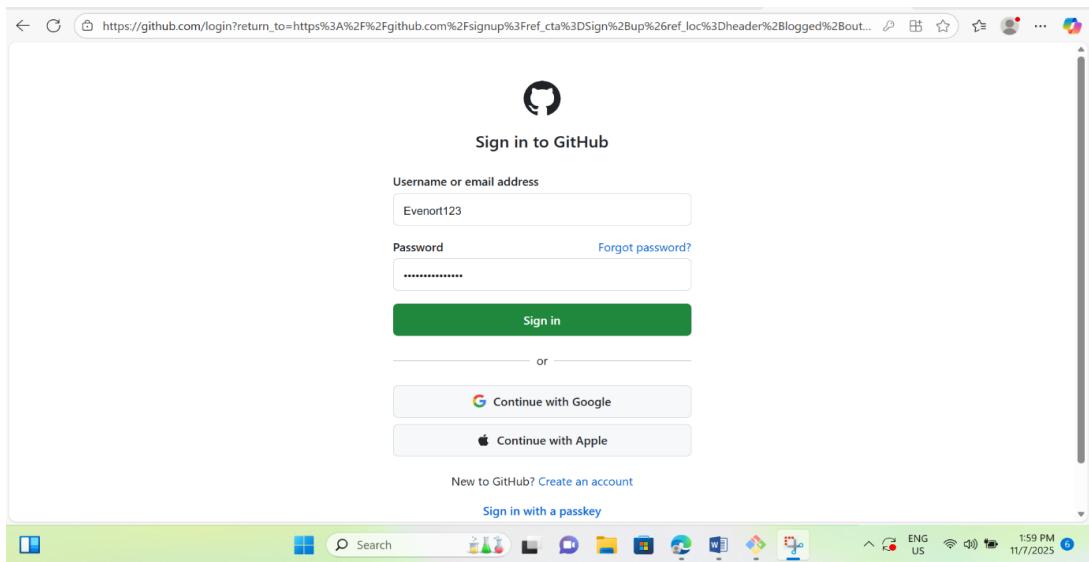
- La commande: **git --version** permet de vérifier la version de Git installée.

C)Création du compte Github

- a) j'ai enregistré mes informations personnelles

The screenshot shows the GitHub sign-up interface. At the top right, there's a link 'Already have an account? [Sign in →](#)'. Below it, the heading 'Sign up for GitHub' is displayed. There are two social sign-in options: 'Continue with Google' and 'Continue with Apple'. A horizontal line with the word 'or' follows. The next section is for 'Email*', with a placeholder 'Email'. The 'Password*' section follows, with a placeholder 'Password' and a note below stating: 'Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.' The 'Username*' section comes next, with a placeholder 'Username' and a note: 'Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.' Finally, there's a 'Your GitHub repository*' section.

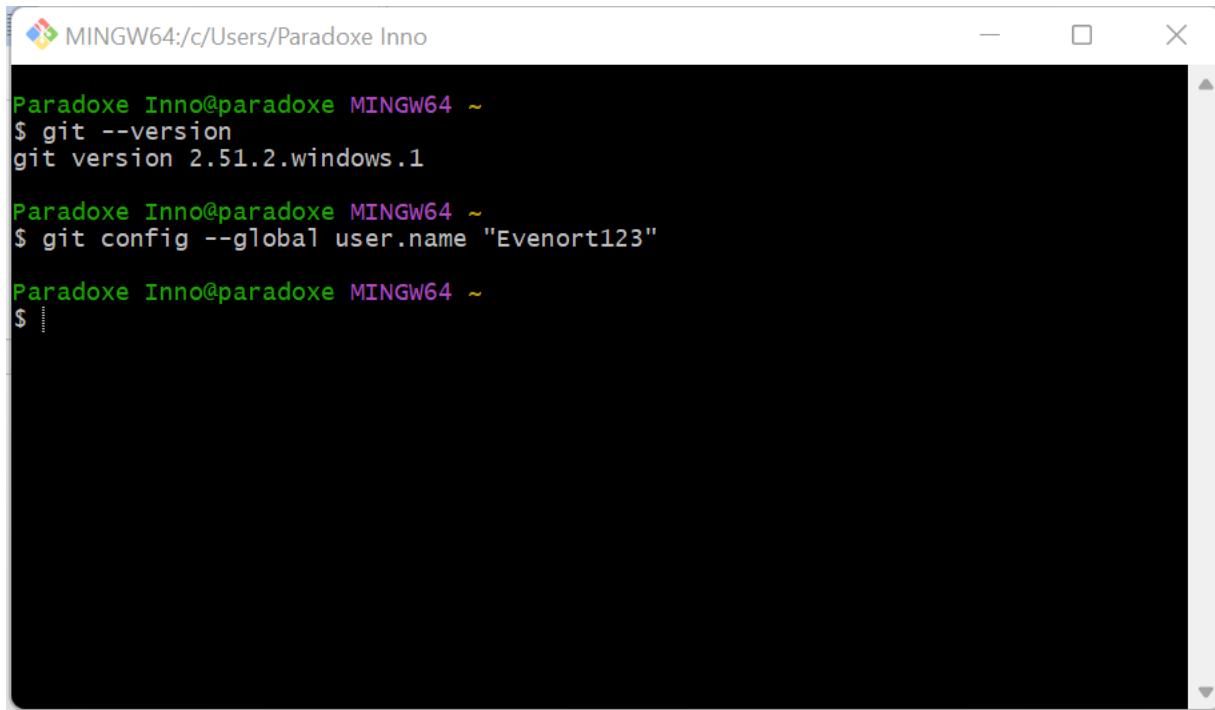
- b) Connexion à GitHub



- Je me suis connecté(e) à mon compte GitHub en saisissant mon nom d'utilisateur (ou mon adresse e-mail) et mon mot de passe sur la page de connexion.

Tout ces démarches nous permettent d'avoir un compte GitHub pleinement fonctionnel.

D) Configuration de Git



The screenshot shows a terminal window titled "MINGW64:/c/Users/Paradoxe Inno". The command line displays the following text:

```
Paradoxe Inno@paradoxe MINGW64 ~
$ git --version
git version 2.51.2.windows.1

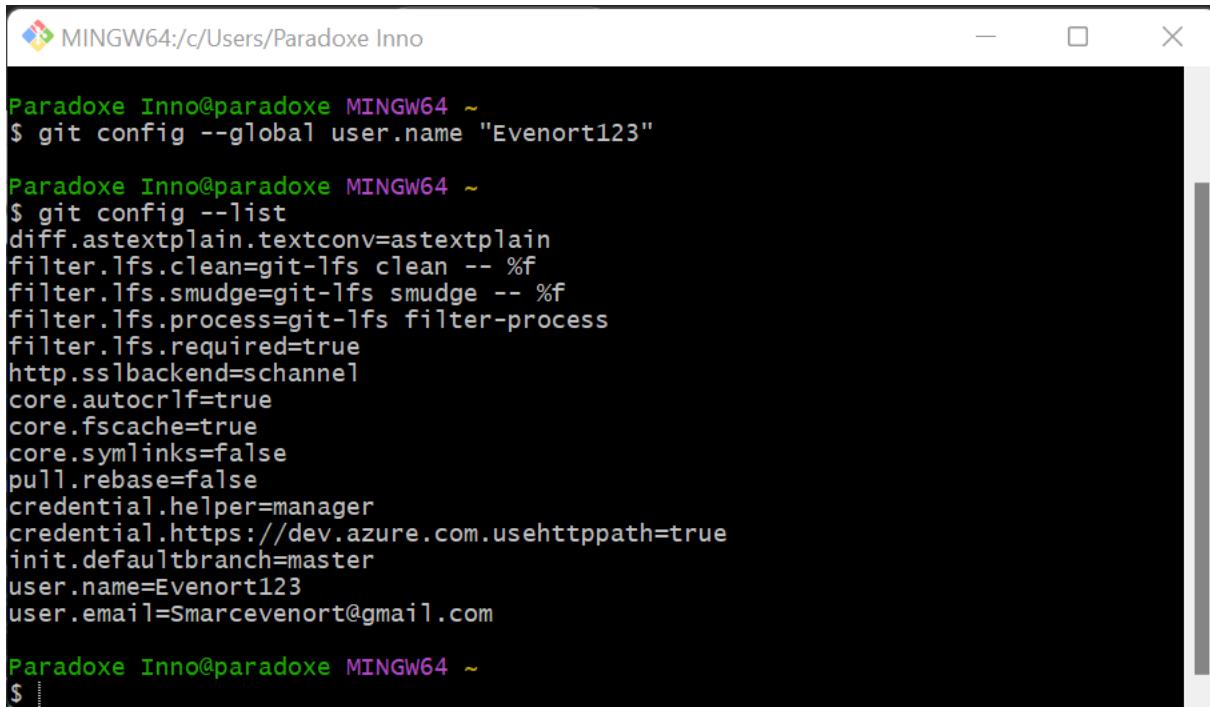
Paradoxe Inno@paradoxe MINGW64 ~
$ git config --global user.name "Evenort123"

Paradoxe Inno@paradoxe MINGW64 ~
$
```

- Dans cette étape, nous configurons Git avec notre nom d'utilisateur à l'aide de la commande: **git config --global user.name “Evenort123”**.

Cette image illustre la façon dont nous configurons Git dans le terminal.

E) Vérifier la Configuration



A screenshot of a terminal window titled "MINGW64:/c/Users/Paradoxe Inno". The window contains the following text:

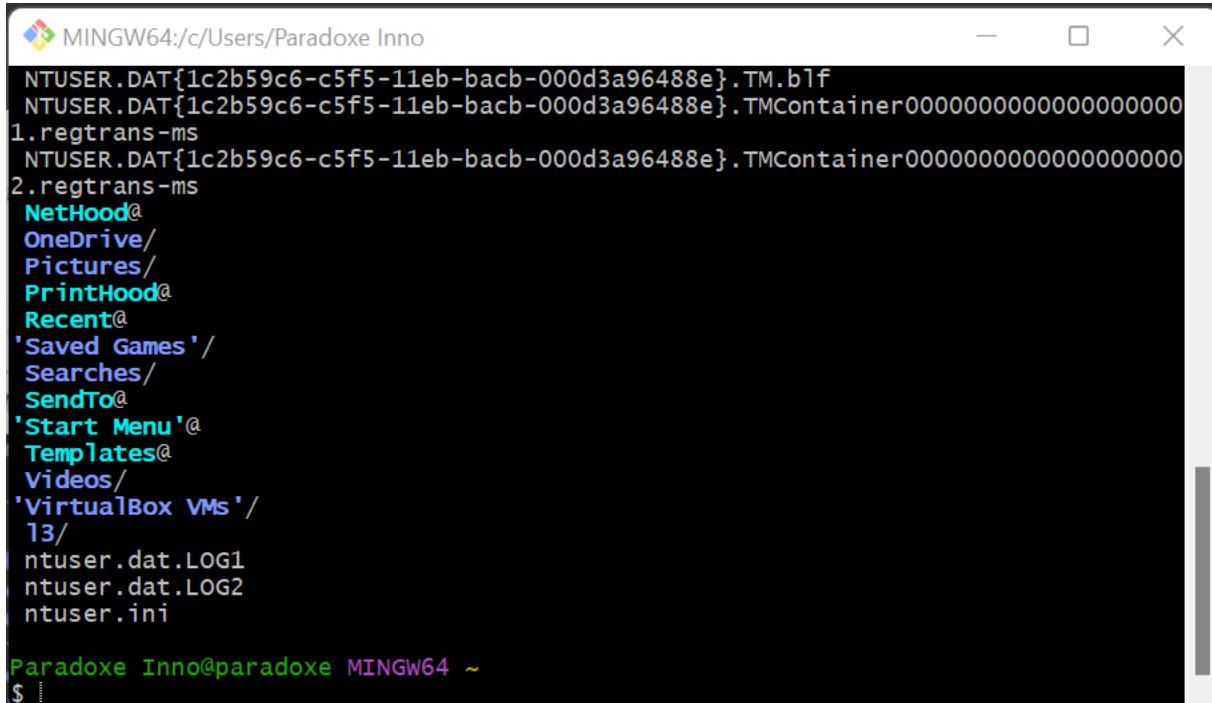
```
Paradoxe Inno@paradoxe MINGW64 ~
$ git config --global user.name "Evenort123"

Paradoxe Inno@paradoxe MINGW64 ~
$ git config --list
diff.astextplain.textconv=astextplain
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
http.sslbackend=schannel
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com/usehttppath=true
init.defaultbranch=master
user.name=Evenort123
user.email=Smarcevenort@gmail.com

Paradoxe Inno@paradoxe MINGW64 ~
$
```

- Pour vérifier la configuration de Git, on utilise la commande **git config --list**
Cette commande affiche les informations de configuration de l'application Git dans le terminal (Comme nom d'utilisateur et l'adresse e-mail).

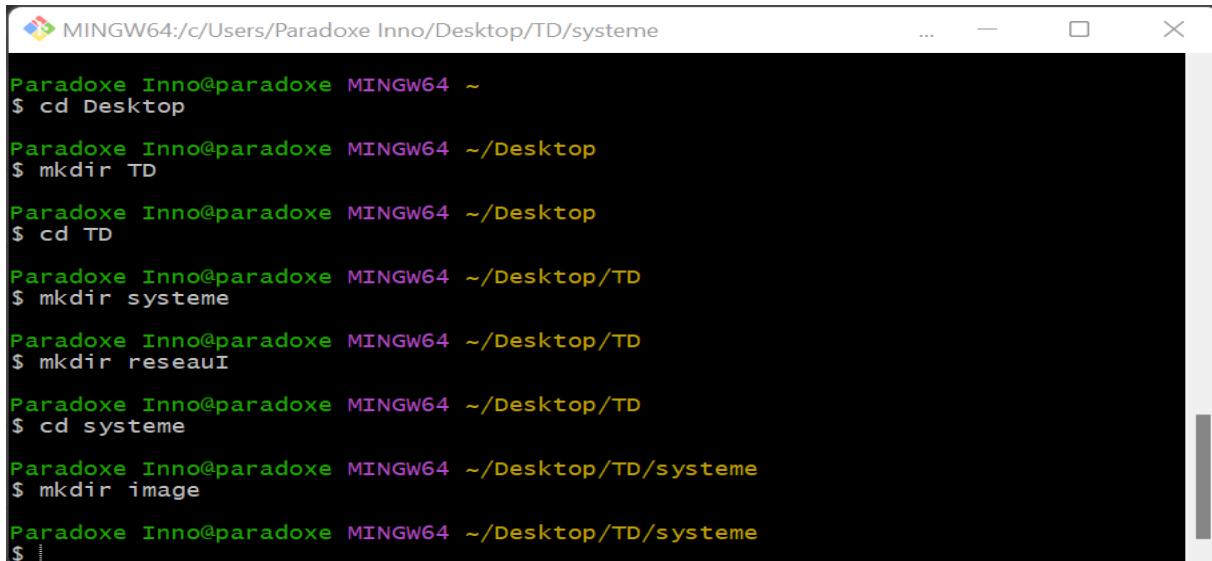
F) Créer un dossier de test



```
MINGW64:/c/Users/Paradoxe Inno
NTUSER.DAT{1c2b59c6-c5f5-11eb-bacb-000d3a96488e}.TM.blf
NTUSER.DAT{1c2b59c6-c5f5-11eb-bacb-000d3a96488e}.TMContainer00000000000000000000
1.regtrans-ms
NTUSER.DAT{1c2b59c6-c5f5-11eb-bacb-000d3a96488e}.TMContainer00000000000000000000
2.regtrans-ms
NetHood@
OneDrive/
Pictures/
PrintHood@
Recent@
'Saved Games'/
Searches/
SendTo@
'Start Menu'@
Templates@
Videos/
'VirtualBox VMs'/
13/
ntuser.dat.LOG1
ntuser.dat.LOG2
ntuser.ini

Paradoxe Inno@paradoxe MINGW64 ~
$
```

- Pour afficher le contenu du répertoire dans le terminal, j'ai utilisé la commande **ls**
- Créer des dossiers dans le terminal



```
MINGW64:/c/Users/Paradoxe Inno/Desktop/TD/systeme
...
Paradoxe Inno@paradoxe MINGW64 ~
$ cd Desktop

Paradoxe Inno@paradoxe MINGW64 ~/Desktop
$ mkdir TD

Paradoxe Inno@paradoxe MINGW64 ~/Desktop
$ cd TD

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD
$ mkdir systeme

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD
$ mkdir reseauI

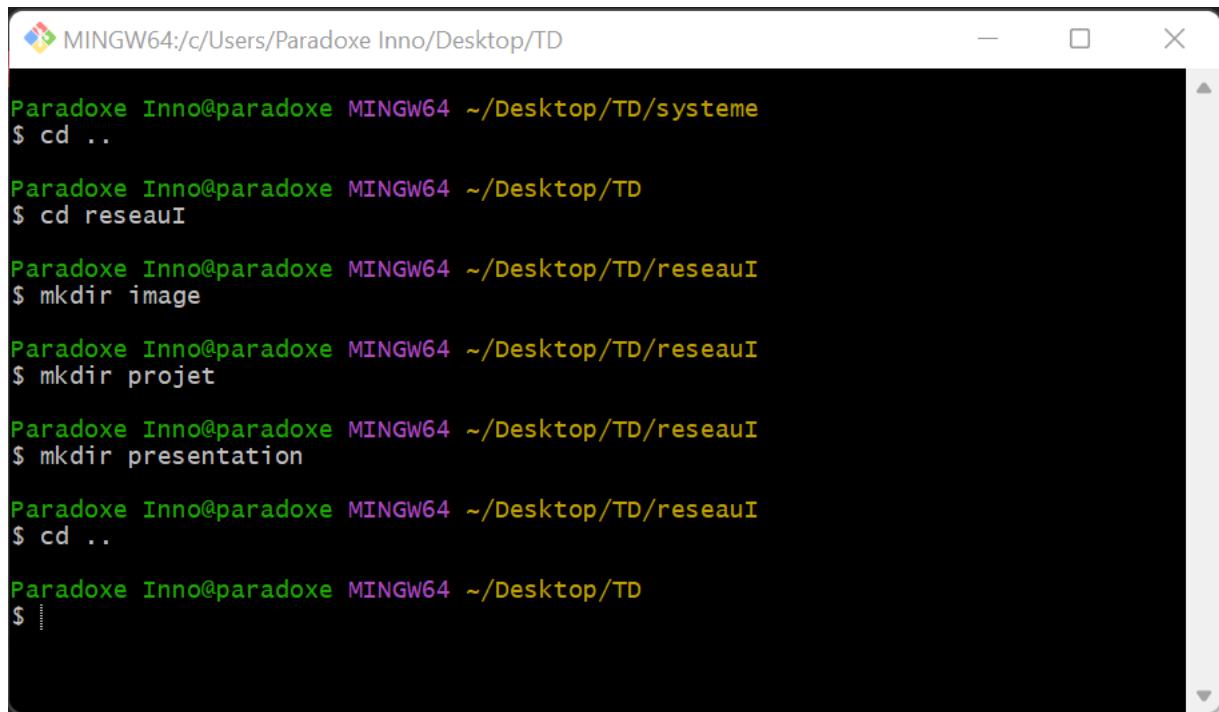
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD
$ cd systeme

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD/systeme
$ mkdir image

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD/systeme
$
```

Pour accéder au bureau de l'ordinateur, j'ai utilisé la commande **cd Desktop** et avec la commande **mkadir**, j'ai crée différents dossiers (TD,systeme,reseauI,image). Ces commandes permettent de créer plusieurs dossiers et sous-dossiers directement dans le terminal.

c) Utiliser la commande cd ..



The screenshot shows a terminal window titled "MINGW64:/c/Users/Paradoxe Inno/Desktop/TD". The terminal history is as follows:

```
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD/systeme
$ cd ..
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD
$ cd reseauI
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD/reseauI
$ mkdir image
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD/reseauI
$ mkdir projet
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD/reseauI
$ mkdir presentation
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD/reseauI
$ cd ..
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD
$
```

- Elle permet de revenir au dossier précédent, dans cette image la commande **cd ..** est utilisée pour naviguer du dossier systeme vers le dossier TD.

The terminal window shows the following session:

```

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD
$ git init
Initialized empty Git repository in C:/Users/Paradoxe Inno/Desktop/TD/.git/
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ echo "bonjour" > README.md

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    README.md

nothing added to commit but untracked files present (use "git add" to track)

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ git add .
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git
touches it

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   README.md

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ git commit -m "premier commit"
[master (root-commit) fbb54c9] premier commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ 

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ git log
commit fbb54c96f5f287706c7be9ee32873c533bfe76ee (HEAD -> master)
Author: Evenort123 <Smarcevenort@gmail.com>
Date:   Fri Nov 7 17:13:37 2025 -0500

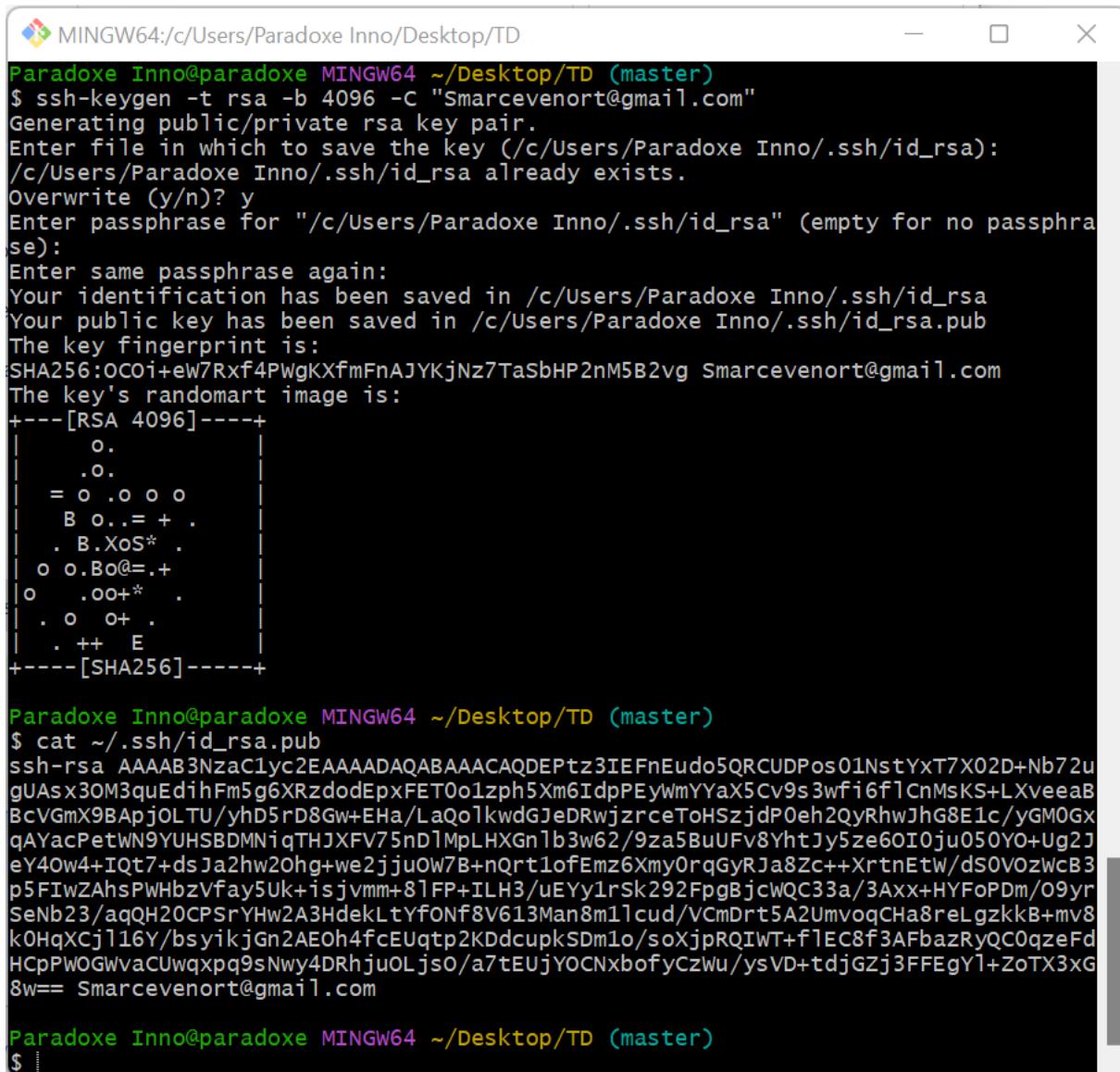
  premier commit

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ 

```

- Ces deux images nous indiquent la façon dont on peut initialiser Git dans un dossier.
- Elle montrent aussi comment créer différents fichiers et ajouter au suivi de Git.
- Enfin la commande git log permet de voir le commit Head (master), l'Auteur, la Date et vérifier que le travail est bien démarré.

G) Générer la clé SSH

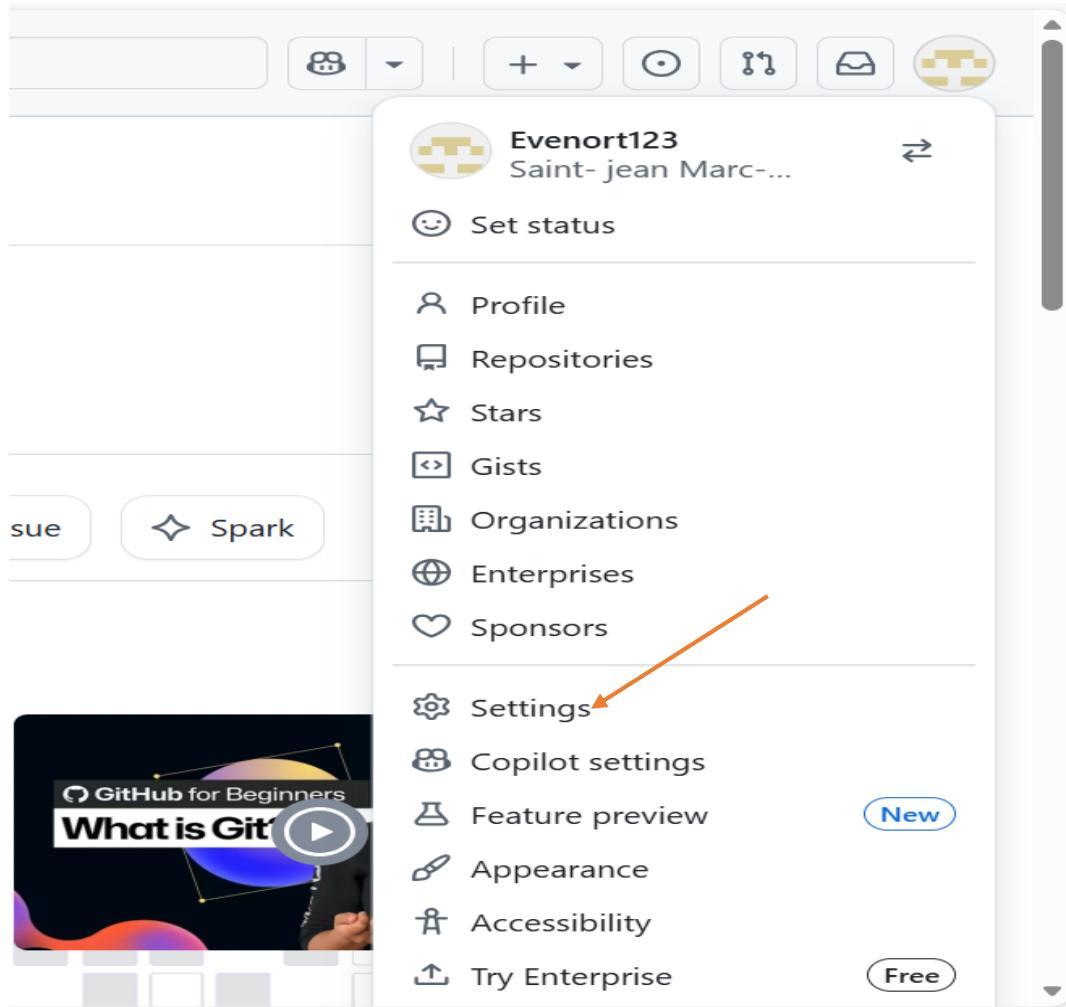


```
MINGW64:/c/Users/Paradoxe Inno/Desktop/TD
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ ssh-keygen -t rsa -b 4096 -C "Smarcevenort@gmail.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/Paradoxe Inno/.ssh/id_rsa):
/c/Users/Paradoxe Inno/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase for "/c/Users/Paradoxe Inno/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/Paradoxe Inno/.ssh/id_rsa
Your public key has been saved in /c/Users/Paradoxe Inno/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:OC0i+ew7Rxf4PWgKXfmFnAJYKjNz7TaSbHP2nM5B2vg Smarcevenort@gmail.com
The key's randomart image is:
+---[RSA 4096]---+
|   o.
|   .o.
|= o .o o o
| B o..= + .
| . B.XoS* .
| o o.B@=.+
| o .oo+=* .
| . o  o+ .
| . ++ E
+---[SHA256]---+
Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQDEPtz3IEFnEudo5QRUDPos01NstYxT7X02D+Nb7zu
gUAsx3OM3quEdihFm5g6XRzdodEpxFET0o1zph5xm6IdpPEywMYYax5Cv9s3wfi6f1CnMsKS+LXveeaB
BcVGmx9BApjOLTU/yhD5rD8Gw+EHa/LaQo1kwdGJeDRwjzrcet0HSzjd0eh2QyRhJhG8E1c/yGM0Gx
qAYAcPetWN9YUHSBDMNiqTHJXFV75nD1MpLHXGn1b3w62/9za5BuUFv8YhtJy5ze6OIOju050YO+Ug2J
eY40w4+IQt7+dsJa2hw20hg+we2jju0w7B+nQrt1ofEmz6Xmy0rqGyRJa8Zc++XrtnEtW/ds0VOzwcB3
p5FIwZAhsPWHbzVfay5Uk+i5jvmm+81FP+ILH3/uEYy1rSk292FpgBjcWQC33a/3Axx+HYFoPDm/09yr
SeNb23/aqQH20CPsrYHw2A3HdekLtYfONF8V613Man8m1lclud/VCmDrt5A2UmvoqCHA8reLgzkkB+mv8
k0HqXCj116Y/bsyikjGn2AE0h4fcEUqtp2KDdcupkSDm1o/soXjpRQIWt+f1EC8f3AFbazRyQC0qzeFd
HCpPWOGWvaCUwqxpq9sNwy4DRhjuOLjs0/a7tEUjYOCNxbofyCzwu/ysVD+tdjGZj3FFEgY1+zoTX3xG
8w== Smarcevenort@gmail.com

Paradoxe Inno@paradoxe MINGW64 ~/Desktop/TD (master)
$
```

- Cette image nous montre la clé [SHA256] et la clé publique qui permettent de créer une connexion SSH sécurisée entre l'ordinateur et un serveur distant. La commande **cat** **~/.ssh/id_rsa.pub** génère une paire de clés (une clé privée et une clé publique).

H) Accéder aux paramètres de GitHub



- En cliquant sur le profil de l'utilisateur, puis sur settings (paramètres), on accède au paramètres GitHub.dans cet endroit on trouve “**SSH et GPG keys**”, qui permet d’ajouter la clé SSH générée précédemement afin d’établir une connexion sécurisée entre Git et GitHub.

I) Accéder à la section SSH and GPG keys

The screenshot shows the GitHub user settings interface for a user named "Saint-jean Marc-Evenort". The top navigation bar includes a profile picture, the user's name, and the text "Your personal account". Below this, a horizontal menu bar contains links: "Public profile", "Account", "Appearance", "Accessibility" (which is highlighted with a blue bar), and "Notifications". A vertical sidebar on the left is titled "Access" and lists several options: "Billing and licensing", "Emails", "Password and authentication", "Sessions", "SSH and GPG keys" (which is also highlighted with a red arrow pointing to it), "Organizations", "Enterprises", and "Moderation".

- Dans cette image, on voit la zone fléchée indiquer la section “SSH and GPG keys”. En cliquant sur cette option, on accède à la page permettant d’ajouter la clé SSH générée précédemment dans le terminal. Elle permet aussi d’établir une connexion sécurisée entre Git et GitHub.

J) Ajouter la clé SSH sur GitHub

The screenshot shows the GitHub user interface for adding a new SSH key. On the left, there's a sidebar with navigation links like 'Public profile', 'Account', 'Appearance', 'Accessibility', 'Notifications', 'Access', 'Billing and licensing', 'Emails', 'Password and authentication', 'Sessions', 'SSH and GPG keys' (which is selected), 'Organizations', 'Enterprises', and 'Moderation'. The main area is titled 'Add new SSH Key' and contains fields for 'Title' (set to 'Mon Projet') and 'Key type' (set to 'Authentication Key'). A large text area labeled 'Key' contains a public RSA SSH key. At the bottom, there's a green 'Add SSH key' button.

- Dans cette image, j'ai crée une nouvelle clé SSH sous le nom “mon Projet”. J'ai ensuite collé la clé publique générée dans le champ prévu à cet effet, puis clique sur le bouton “**Add SSH key**” pour l'enregistrer.

Cette étape permet à GitHub de reconnaître automatiquement mon ordinateur grâce à la clé SSH, et ainsi d'établir une connexion sécurisée sans mot de pass.

K) Créer un dépôt GitHub

The screenshot shows the GitHub Settings interface for a user named "Saint-jean Marc-Evenort (Evenort123)". A success message at the top states "Okay, you have successfully deleted that key." On the left, a sidebar lists account settings like Public profile, Account, Appearance, Accessibility, Notifications, Access, Billing and licensing, Emails, Password and authentication, Sessions, and SSH and GPG keys. The main area is titled "SSH keys" and displays a list of associated SSH keys, including one for "Mon Projet". Below this is the "GPG keys" section with a "New GPG key" button. A context menu is open in the top right corner, with an arrow pointing to the "New repository" option.

Okay, you have successfully deleted that key.

Saint-jean Marc-Evenort (Evenort123)
Your personal account

SSH keys

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

Authentication keys

Mon Projet
SHA256:VOXw6GPA70LrjeLEwsbAC6gCyjAM9ohaSykv0DCsta
Added on Nov 7, 2025
Never used — Read/write

Check out our guide to [connecting to GitHub using SSH keys](#) or troubleshoot [common SSH problems](#).

GPG keys

New GPG key

New issue
New repository
Import repository
New codespace
New gist
New organization
New project

- Dans Cette image, on voit le cheminement à suivre pour créer un nouveau dépôt sur GitHub.

L) Création du dépôt et configuration

Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).
Required fields are marked with an asterisk (*).

1 General

Owner * Evenort123 / Repository name * "TD"

Your new repository will be created as -TD-.
The repository name can only contain ASCII letters, digits, and the characters ., -, and _.

Great repository names are short and memorable. How about [potential-octo-happiness](#)?

Description

0 / 350 characters

2 Configuration

Choose visibility * Public

Choose who can see and commit to this repository

Add README Off

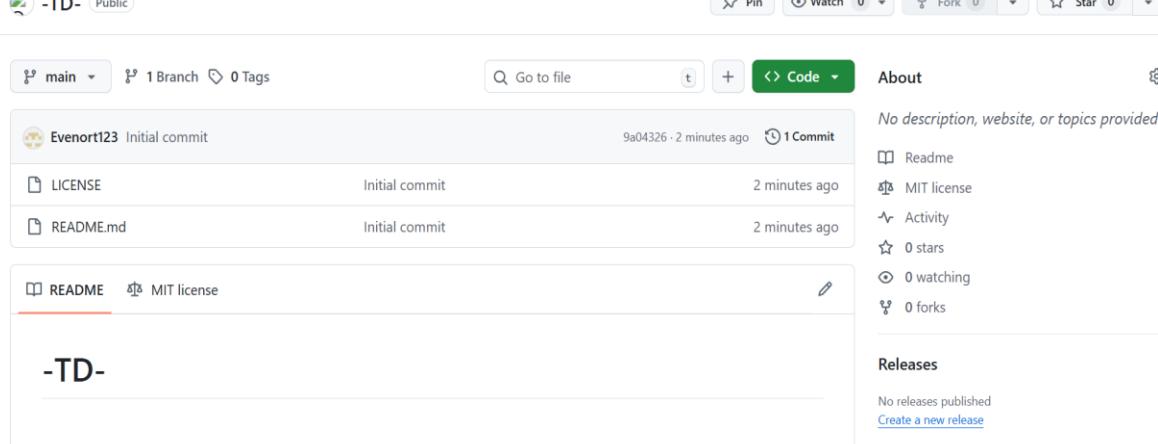
READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore No .gitignore

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

Add license MIT License

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



- TD- Public

main 1 Branch 0 Tags

Go to file + Code

Evenort123 Initial commit 9a04326 · 2 minutes ago 1 Commit

LICENSE Initial commit 2 minutes ago

README.md Initial commit 2 minutes ago

README MIT license

About

No description, website, or topics provided.

Readme

MIT license

Activity

0 stars

0 watching

0 forks

Releases

No releases published

Create a new release

- ces images nous montre le nom du dépôt (-TD-) ainsi la configuration utilisée lors de la création

8- Cloner un dépôt GitHub via SSH

The screenshot shows a GitHub repository page for a repository named "-TD-". The "Code" tab is selected. A modal window is open over the repository details, specifically the "Clone" section. The "SSH" tab is highlighted with a red arrow. Below it, the URL "git@github.com:Evenort123/-TD-.git" is displayed, with a blue arrow pointing to it. The modal also contains instructions: "Use a password-protected SSH key." and links to "Open with GitHub Desktop" and "Download ZIP". The repository details on the right include "About" (No description, website, or topics provided), "Readme", "MIT license", "Activity", "0 stars", "0 watching", "0 forks", and "Releases" (No releases published, Create a new release).

- Cette image montre comment copier le code de clonage SSH afin de l'utiliser dans le terminal pour cloner le dépôt sur l'ordinateur local.

The screenshot shows a terminal window titled "MINGW64:/c/Users/Paradoxe Inno/Desktop/TD". The user is in their home directory (~) and has navigated to the Desktop folder. They run the command "git clone git@github.com:Evenort123/-TD-.git". The output shows the cloning process: "Cloning into '-TD-'...", "remote: Enumerating objects: 4, done.", "remote: Counting objects: 100% (4/4), done.", "remote: Compressing objects: 100% (3/3), done.", "remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)", and "Receiving objects: 100% (4/4), done."

Ainsi, cette image montre clairement que le projet a été cloné avec succès sur la machine locale.

Conclusion

Ce TD m'a permis de comprendre et d'appliquer les bases de Git et GitHub.

Grâce à ce devoir j'ai appris comment à manipuler le terminal sans utiliser l'interface graphique de mon ordinateur telles que:

Initialiser un dépôt Git, Créer un dépôt GitHub et le cloner, j'ai découvert les différentes commandes, ainsi que la manière de cloner et d'héberger un document sur GitHub.

Ce devoir m'a donné une source de motivation pour ce cours de système. malgré les difficultés rencontrées au cours de ce travail, je veux travailler davantage afin de mieux comprendre le reste du cours.

Mêmes si j'ai rencontré quelques difficultés, je compte travailler encore plus, car dans la vie rien n'est facile.

Pour moi, la tâche est réussie et cette expérience m'a permis de progresser.

Dans ce travail dirigé, j'ai rencontré des difficultés énormes qui m'ont permis de mieux maîtriser le laboratoire.

Pour l'objectif je peux dire qu'il a été atteint.