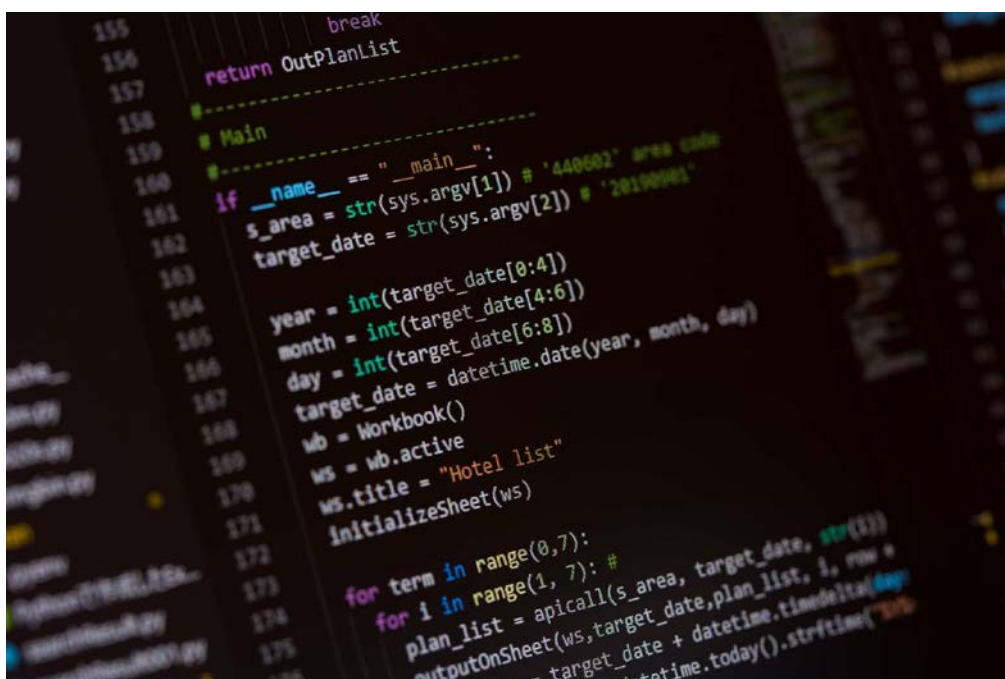


# Rapport de projet



## Participer à la vie de la communauté Open Source



Source : Google image

# Sommaire

|   |      |
|---|------|
| <b>Présentation du projet</b>             | p.3  |
| Mise en situation                         | p.3  |
| Objectifs                                 | p.3  |
| <b>Algorithme</b>                         | p.4  |
| Thème/Sujet du script/choix               | p.4  |
| Algorithme 1                              | p.5  |
| Algorithme 2                              | p.5  |
| Algorithme 3                              | p.6  |
| Algorithme 4                              | p.6  |
| <b>Ubuntu</b>                             | p.7  |
| <b>Python</b>                             | p.8  |
| Téléchargement et installation de Python  | p.8  |
| Téléchargement et installation de PyCharm | p.9  |
| Scripts                                   | p.10 |
| <b>Tests du script</b>                    | p.13 |
| <b>GitHub</b>                             | p.15 |
| Création du répertoire GitHub             | p.15 |
| Répertoire GitHub                         | p.17 |
| Licence Open Source GitHub                | p.17 |
| <b>Conclusion</b>                         | p.18 |
| <b>Annexe</b>                             | p.19 |
| <b>Lexique</b>                            | p.20 |

# Présentation du projet

## Mise en situation :

Le web repose en grande partie sur un gigantesque réseau d'entraide où chacun peut à la fois recevoir mais aussi contribuer à sa mesure.

Aujourd'hui, une des plateformes les plus populaires pour contribuer à la communauté Open Source se nomme GitHub.

GitHub propose toute l'infrastructure permettant de facilement mettre à disposition du code, de la documentation, de rechercher des projets et de pouvoir y contribuer.

Vous allez maintenant créer un script sous Python permettant d'automatiser des tâches d'administration et de partager votre code avec la communauté sur votre répertoire personnel GitHub.

## Objectifs :

- Identifier une tâche complexe ou un ensemble de tâches d'administration que vous aimeriez pouvoir automatiser.
- Une fois votre projet défini, créez un répertoire GitHub qui hébergera votre projet
- Il est maintenant temps de coder en Python votre outil d'automatisation, soit sous forme d'un script, soit sous forme d'un module Ansible
- Mettez votre code à disposition de la communauté
- Documentez à la fois le fonctionnement de votre code et la manière d'utiliser votre code sans entrer trop dans les détails de sa conception technique

# Algorithme

## Thème :

Script d'automatisation pour configurer une machine qui utilise Ubuntu en serveur.

## Sujet du script :

Configuration IP et installation des services DHCP, NTP et Samba.

## Choix :

J'ai choisi de partir dans cette direction car dans mon entreprise nous travaillons sur des serveurs et je me suis dit que ça pouvait être utile de travailler sur des scripts d'automatisation pour configurer des Serveurs sur Ubuntu.

Nous allons avoir un algorithme principal qui va regrouper ces quatre fonctions suivantes :

- 1- configurerip()
- 2- dhcp()
- 3- ntp()
- 4- samba()

Le programme est basé sur Ubuntu 18.04 avec un paramétrage basique.

### **Algorithmes 1 : configurerIP()**

**Rôle** : Configurer une IP fixe (Masque, DNS, Passerelle) du serveur.

#### **Début**

Ecrire '**Bienvenue dans la configuration IP du serveur Ubuntu.**'

Aller dans **/etc/network/interfaces**

Entrer '**le numéro de l'interface**'

Entrer '**le numéro de l'interface**'

Entrer '**l'adresse IP**'

Entrer '**le netmask**'

Entrer '**le gateway**'

Entrer '**le DNS**'

Ecrire '**Vous avez terminé la configuration IP du serveur Ubuntu.**'

#### **Fin**

### **Algorithmes 2 : dhcp()**

**Rôle** : Installation du service DHCP.

#### **Début**

Ecrire '**Bienvenue dans l'installation et la configuration du service DHCP.**'

Installer le service DHCP

Aller dans **/etc/dhcp/dhcpd.conf**

Entrer '**l'adresse IP et le netmask**'

Entrer '**l'adresse de début et l'adresse de fin**'

Entrer '**le DNS**'

Entrer '**la route par défaut**'

Entrer '**le broadcast**'

Entrer '**le DNS**'

Ecrire '**Vous avez terminé l'installation et la configuration du service DHCP.**'

#### **Fin**

### **Algorithmes 3 :** ntp()

**Rôle :** Installation du service NTP serveur.

#### **Début**

Ecrire '**Bienvenue dans l'installation du service NTP.**'

Installer le service NTP

#### **Fin**

### **Algorithmes 4 :** samba()

**Rôle :** Installation du service samba.

#### **Début**

Ecrire '**Bienvenue dans l'installation du service Samba.**'

Installer le service Samba

Aller dans '**/etc/samba/smb.conf**'

Entrer '**la configuration du fichier**'

Création d'un utilisateur et d'un groupe de partage

Préparation du dossier de partage

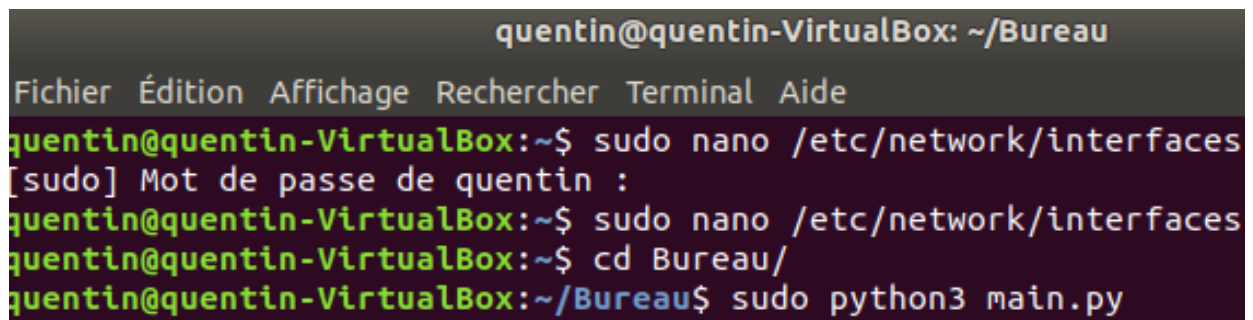
#### **Fin**

# Ubuntu

Pour ce script nous allons installer une machine Ubuntu 18.04 et faire un apt update. Nous allons nous servir de cette machine comme serveur.

Pour lancer le script sur Ubuntu il faut :

- mettre le script là où on le souhaite, pour ma part ça sera sur le bureau
- ouvrir l'invite de commande
- se déplacer vers l'endroit où se trouve le script « cd Bureau »
- taper « sudo python3 -nomDuScript-.py »



```
quentin@quentin-VirtualBox: ~/Bureau
Fichier Édition Affichage Rechercher Terminal Aide
quentin@quentin-VirtualBox:~$ sudo nano /etc/network/interfaces
[sudo] Mot de passe de quentin :
quentin@quentin-VirtualBox:~$ sudo nano /etc/network/interfaces
quentin@quentin-VirtualBox:~$ cd Bureau/
quentin@quentin-VirtualBox:~/Bureau$ sudo python3 main.py
```

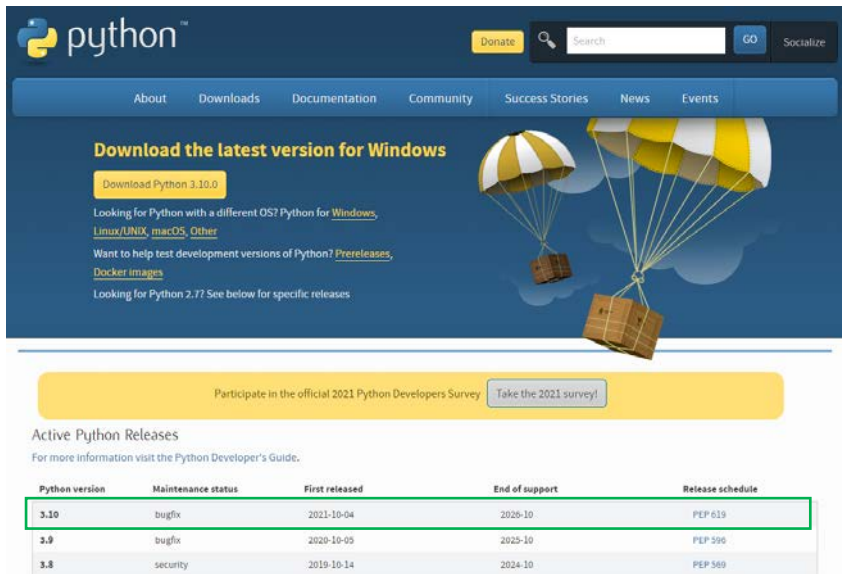
# Python

Aide 1 : <https://www.youtube.com/watch?v=oUJolR5bX6g>

Nous allons télécharger et installer python et de quoi développer sur python.

## Téléchargement et installation de python :

Il faut télécharger la dernière version de python et suivre son installation.



Maintenant pour vérifier si Python est bien installé sur l'ordinateur on peut taper la commande ci-dessous.

```
Microsoft Windows [version 10.0.19043.1288]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\Dotrakii>python --version
Python 3.10.0

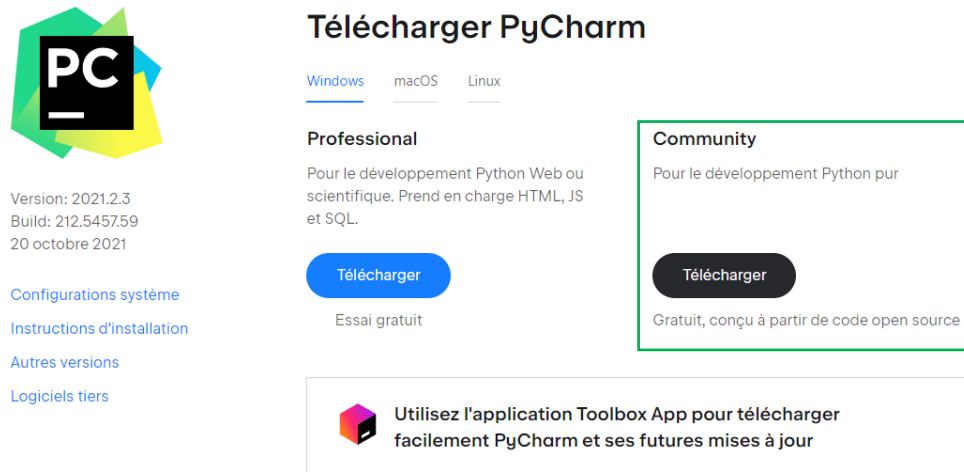
C:\Users\Dotrakii>
```



## Téléchargement et installation de PyCharm :

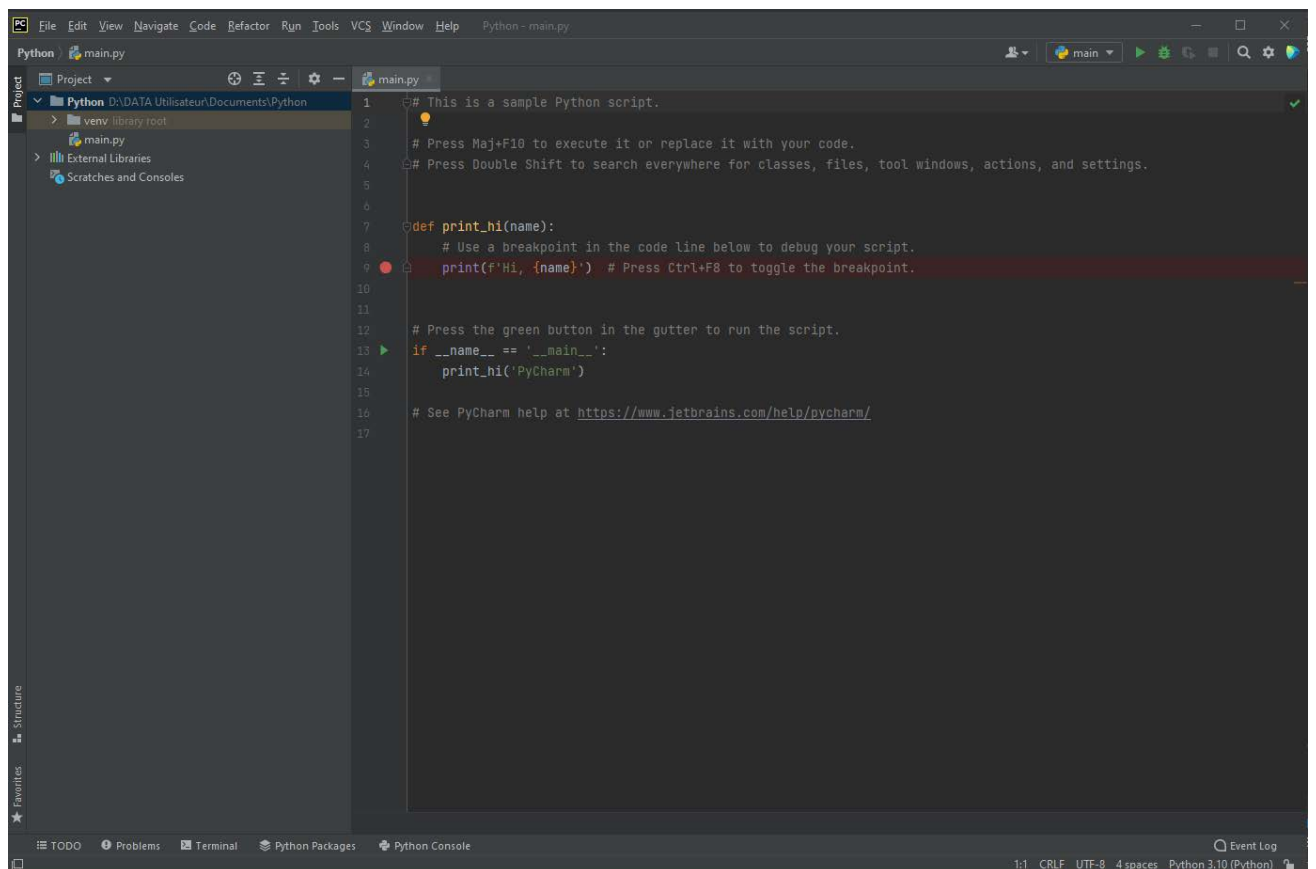
PyCharm est un logiciel qui permet de développer en langage python. J'ai choisi celui-là suite à une vidéo qui proposait deux choix et cette interface m'a plu.

Nous allons procéder comme l'installation de python, télécharger la dernière version et suivre l'installation de PyCharm.



The image shows the PyCharm download page. On the left is the PyCharm logo (a green and yellow hexagon with 'PC' in a black square) and version information: Version: 2021.2.3, Build: 212.5457.59, 20 octobre 2021. Below this are links for 'Configurations système', 'Instructions d'installation', 'Autres versions', and 'Logiciels tiers'. The main heading is 'Télécharger PyCharm' with tabs for 'Windows', 'macOS', and 'Linux'. Under 'Professional', it says 'Pour le développement Python Web ou scientifique. Prend en charge HTML, JS et SQL.' and has a blue 'Télécharger' button and 'Essai gratuit' text. Under 'Community', it says 'Pour le développement Python pur' and has a dark blue 'Télécharger' button and 'Gratuit, conçu à partir de code open source' text. At the bottom, a box says 'Utilisez l'application Toolbox App pour télécharger facilement PyCharm et ses futures mises à jour' with a Toolbox App icon.

Et voici le résultat quand on ouvre PyCharm :



## Scripts :

Voici la capture d'écran du programme avec les quatre fonctions ainsi que le programme principal.

### Fonction 1 :

```
def configurerip():
    print("Bienvenue dans la configuration IP du serveur Ubuntu.")

    numinterface = input("Entrez le numéro de l'interface réseau (ex : 3 pour enp0s3) : ")
    adresseip = input("Entrez l'adresse IP du serveur : ")
    netmask = input("Entrez le masque sous réseau : ")
    gateway = input("Entrez la passerelle : ")
    dns = input("Entrez l'adresse IP du DNS : ")

    configip = open("/etc/network/interfaces", "w")
    configip.write("auto enp0s" + numinterface + "\n")
    configip.write("iface enp0s" + numinterface + " inet static\n")
    configip.write("address " + adresseip + "\n")
    configip.write("netmask " + netmask + "\n")
    configip.write("gateway " + gateway + "\n")
    configip.write("dns-nameservers " + dns + "\n")
    print("Vous avez terminé la configuration IP du serveur Ubuntu.")

pass
```

### Fonction 2 :

```
def dhcp():
    print("Bienvenue dans l'installation et configuration du service DHCP.")
    os.system("apt-get install isc-dhcp-server")

    ip = input("Entrez l'adresse IP du réseau (ex : 192.168.*.0) : ")
    netmask = input("Entrez le masque sous réseau : ")
    range_debut = input("Entrez la première adresse IP : ")
    range_fin = input("Entrez la dernière adresse IP : ")
    dns = input("Entrez le dns : ")
    route = input("Entrez la route par défaut : ")
    broadcast = input("Entrez l'adresse de Broadcast : ")

    fichier = open("/etc/dhcp/dhcpd.conf", "w")
    fichier.write("# Configuration pour le réseau \n")
    fichier.write("\nsubnet " + ip + " netmask " + netmask + " {\n")
    fichier.write("\nrange " + range_debut + " " + range_fin + ";\n")
    fichier.write("\noption domain-name-servers " + dns + ";\n")
    fichier.write("\noption routers " + route + ";\n")
    fichier.write("\noption broadcast-address " + broadcast + ";\n")
    fichier.write("\ndefault-lease-time 3600;\n")
    fichier.write("\nmax-lease-time 7200;\n")
    fichier.write("\n}")

    oui = "o"
    non = "n"
    reponse = input("Est ce que le serveur DHCP fait aussi routeur ? (Oui : o Non : n) : \n")
    if reponse == oui:
        iv4 = input("Entrez l'interface réseau connecté au routeur (enp0s8) : ")
        fichier2 = open("/etc/default/isc-dhcp-server", "w")
        fichier2.write("\nINTERFACES=" + iv4 + "\n")
        fichier2.write("\nINTERFACESv6=" + "" + "\n")
    else:
        print("Non il ne fait pas routeur.")
    pass
    os.system("service isc-dhcp-server restart")
    print("Vous avez terminé l'installation et configuration du service DHCP.")

pass
```

### Fonction 3 :

```
def ntp():
    print("Bienvenue dans l'installation du service NTP.")
    os.system("apt-get install ntp")

pass
```

### Fonction 4 :

```
def samba():
    print("Bienvenue dans l'installation du service Samba.")
    os.system("apt-get install samba")

    fichier = open("/etc/samba/smb.conf", "a")
    fichier.write("[partage;\n")
    fichier.write("\n    comment = Partage de données;\n")
    fichier.write("\n    path = /srv/partage;\n")
    fichier.write("\n    guest ok = no;\n")
    fichier.write("\n    read only = no;\n")
    fichier.write("\n    browsable = yes;\n")
    fichier.write("\n    valid users = @partage;\n")

    # Création d'un utilisateur et d'un groupe de partage

    os.system("adduser quentin") # création d'un utilisateur
    os.system("smbpasswd -a quentin") # ajout de l'utilisateur
    os.system("groupadd partage") # création d'un groupe de partage
    os.system("gpasswd -a quentin partage") # ajout de l'utilisateur dans un groupe

    # Préparation du dossier partage

    os.system("mkdir /srv/partage")
    os.system("chgrp -R partage /srv/partage/")
    os.system("chmod -R g+rw /srv/partage/")

pass
```

### Programme principal :

```
#                               Programme : Installer&ConfigurerUbuntuSRV                               #

Oui = "o"
Non = "n"

reponse2 = input("Bonjour, voulez vous configurer Ubuntu ? (Oui : o Non : n) : \n")
if reponse2 == Oui:
    print("Nous allons commencer l'installation.")

    print("Première étape : Configuration de l'adressage IP du Serveur Ubuntu\n")
    configurerip()

    print("Deuxième étape : Installation du service DHCP")
    dhcp()

    print("troisième étape : Installation du service NTP")
    ntp()

    print("quatrième étape : Installation du service Samba")
    samba()

    print("Vous avez terminer la configuration de votre serveur Ubuntu.")

else:
    print("La configuration a été annulé.")
pass
```

### Notes importantes :

- faire attention aux chemins d'accès qu'il ne soit pas trop long
- faire attention à la réutilisation des variables
- faire attention à la structure du code (placement des fonctions, la présentation, les lignes de codes)
- faire attention aux chemins avec des espaces

# Tests du script

Voici les résultats des tests après le lancement du programme.

IP :

```
quentin@quentin-VirtualBox: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
GNU nano 2.9.3 /etc/network/interfaces  
  
auto enp0s8  
iface enp0s8 inet static  
address 192.168.0.2  
netmask 255.255.255.0  
gateway 192.168.0.250  
dns-nameservers 192.168.0.1
```

DHCP :

```
quentin@quentin-VirtualBox: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
GNU nano 2.9.3 /etc/dhcp/dhcpd.conf  
  
# Configuration pour le réseau  
  
subnet 192.168.0.0 netmask 255.255.255.0 {  
  
range 192.168.0.3 192.168.0.50;  
  
option domain-name-servers 192.168.0.1;  
  
option routers 192.168.0.2;  
  
option broadcast-address 192.168.0.255;  
  
default-lease-time 3600;  
max-lease-time 7200;  
}
```

```
quentin@quentin-VirtualBox: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
GNU nano 2.9.3 /etc/default/isc-dhcp-server  
  
INTERFACES="enp0s8"  
INTERFACESv6=""
```

```
quentin@quentin-VirtualBox:~$ sudo service isc-dhcp-server status  
● isc-dhcp-server.service - ISC DHCP IPv4 server  
   Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset:  
   Active: active (running) since Tue 2021-12-28 18:05:32 CET; 2s ago  
     Docs: man:dhcpd(8)  
    Main PID: 2053 (dhcpd)  
      Tasks: 1 (limit: 4667)  
    CGroup: /system.slice/isc-dhcp-server.service  
            └─2053 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -  
  
déc. 28 18:05:32 quentin-VirtualBox sh[2053]: Wrote 0 leases to leases file.  
déc. 28 18:05:32 quentin-VirtualBox dhcpd[2053]: PID file: /run/dhcp-server/dhcpd.pid  
déc. 28 18:05:32 quentin-VirtualBox dhcpd[2053]: Wrote 0 leases to leases file.  
déc. 28 18:05:32 quentin-VirtualBox dhcpd[2053]: Listening on LPF/enp0s8/08:00:27:01:eb  
déc. 28 18:05:32 quentin-VirtualBox sh[2053]: Listening on LPF/enp0s8/08:00:27:01:eb:e4  
déc. 28 18:05:32 quentin-VirtualBox sh[2053]: Sending on LPF/enp0s8/08:00:27:01:eb:e4  
déc. 28 18:05:32 quentin-VirtualBox sh[2053]: Sending on Socket/fallback/fallback-net  
déc. 28 18:05:32 quentin-VirtualBox dhcpd[2053]: Sending on LPF/enp0s8/08:00:27:01:eb  
déc. 28 18:05:32 quentin-VirtualBox dhcpd[2053]: Sending on Socket/fallback/fallback-  
déc. 28 18:05:32 quentin-VirtualBox dhcpd[2053]: Server starting service.  
lines 1-19/19 (END)
```

## NTP :

```
quentin@quentin-VirtualBox:~$ sudo service ntp status
● ntp.service - Network Time Service
   Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: enab
   Active: active (running) since Tue 2021-12-28 17:44:36 CET; 5min ago
     Docs: man:ntpd(8)
   Main PID: 3759 (ntpd)
    Tasks: 2 (limit: 4667)
   CGroup: /system.slice/ntp.service
           └─3759 /usr/sbin/ntpd -p /var/run/ntpd.pid -g -u 123:128

déc. 28 17:44:41 quentin-VirtualBox ntpd[3759]: Soliciting pool server 194.177.3
déc. 28 17:44:41 quentin-VirtualBox ntpd[3759]: Soliciting pool server 5.196.160
déc. 28 17:44:41 quentin-VirtualBox ntpd[3759]: Soliciting pool server 92.222.20
déc. 28 17:44:41 quentin-VirtualBox ntpd[3759]: Soliciting pool server 91.189.94
déc. 28 17:44:42 quentin-VirtualBox ntpd[3759]: Soliciting pool server 91.189.89
déc. 28 17:44:42 quentin-VirtualBox ntpd[3759]: Soliciting pool server 78.196.16
déc. 28 17:44:42 quentin-VirtualBox ntpd[3759]: Soliciting pool server 5.196.192
déc. 28 17:44:46 quentin-VirtualBox ntpd[3759]: receive: Unexpected origin times
déc. 28 17:44:46 quentin-VirtualBox ntpd[3759]: receive: Unexpected origin times
déc. 28 17:49:57 quentin-VirtualBox ntpd[3759]: kernel reports TIME_ERROR: 0x204
lines 1-19/19 (END)
```

## Samba :

```
quentin@quentin-VirtualBox: ~
Fichier Édition Affichage Rechercher Terminal Aide
GNU nano 2.9.3 /etc/samba/smb.conf

[partage];

    comment = Partage de données;

    path = /srv/partage;

    guest ok = no;

    read only = no;

    browsable = yes;

    valid users = @partage;
```

```
quentin@quentin-VirtualBox: /srv/partage
Fichier Édition Affichage Rechercher Terminal Aide
quentin@quentin-VirtualBox:~$ cd /srv/partage/
quentin@quentin-VirtualBox:/srv/partage$
```

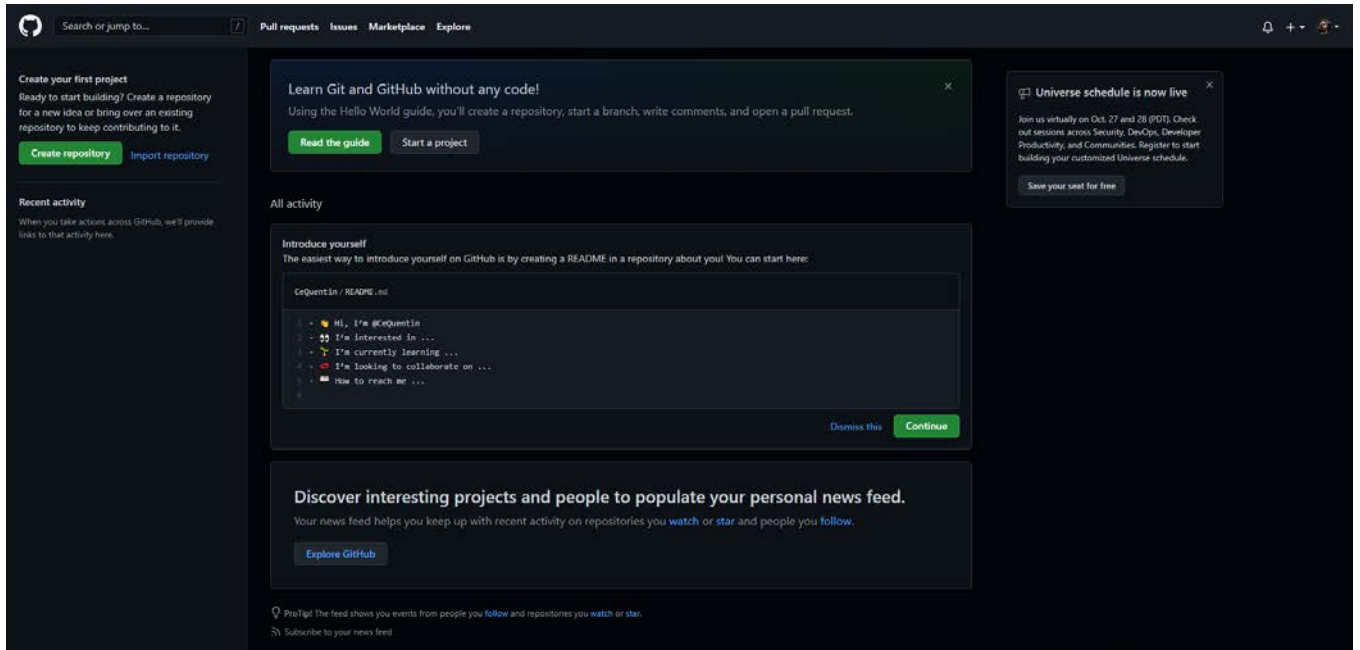
```
quentin@quentin-VirtualBox:~$ sudo systemctl status smbd
[sudo] Mot de passe de quentin :
● smbd.service - Samba SMB Daemon
   Loaded: loaded (/lib/systemd/system/smbd.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2021-12-28 17:45:09 CET; 4min 2s ago
     Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
   Main PID: 6124 (smbd)
   Status: "smbd: ready to serve connections..."
    Tasks: 4 (limit: 4667)
   CGroup: /system.slice/smbd.service
           └─6124 /usr/sbin/smbd --foreground --no-process-group
             └─6126 /usr/sbin/smbd --foreground --no-process-group
               └─6127 /usr/sbin/smbd --foreground --no-process-group
                 └─6130 /usr/sbin/smbd --foreground --no-process-group

déc. 28 17:45:09 quentin-VirtualBox systemd[1]: Starting Samba SMB Daemon...
déc. 28 17:45:09 quentin-VirtualBox systemd[1]: Started Samba SMB Daemon.
lines 1-17/17 (END)
```



# GitHub

## Création du compte GitHub :

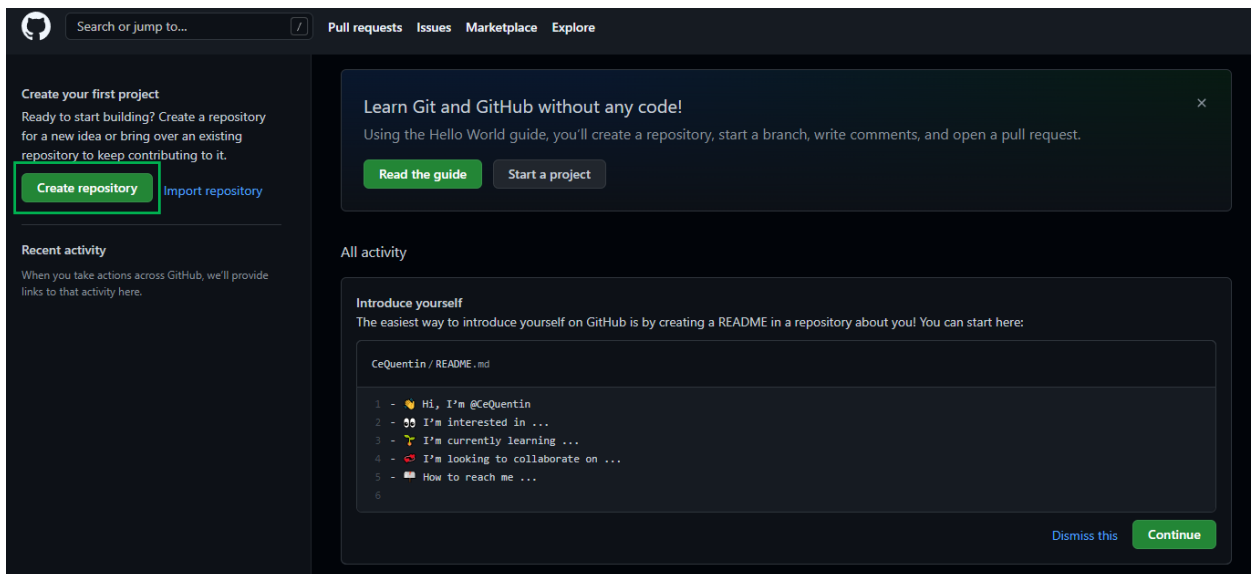


Voici mon compte GitHub avec le lien ci-dessous.

<https://github.com/CeQuentin/Script-de-configuration-Ubuntu-Serveur>

## Création du répertoire GitHub :

Pour créer un nouveau répertoire il faut cliquer en haut à gauche sur « **Create repository** ».



Ensuite il faut choisir le nom du répertoire ainsi que les options de ce dernier.

**Create a new repository**

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner **CeQuentin** / Repository name **Projet 6 : Configuration Windows** ✓

Great repository names are [Projet-6-Configuration-Windows-Serveur-2016](#). Your new repository will be created as **Projet-6-Configuration-Windows-Serveur-2016**.

Description (optional)

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

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

**Initialize this repository with:**  
Skip this step if you're importing an existing repository.

☒ **Add a README file**  
This is where you can write a long description for your project. [Learn more.](#)

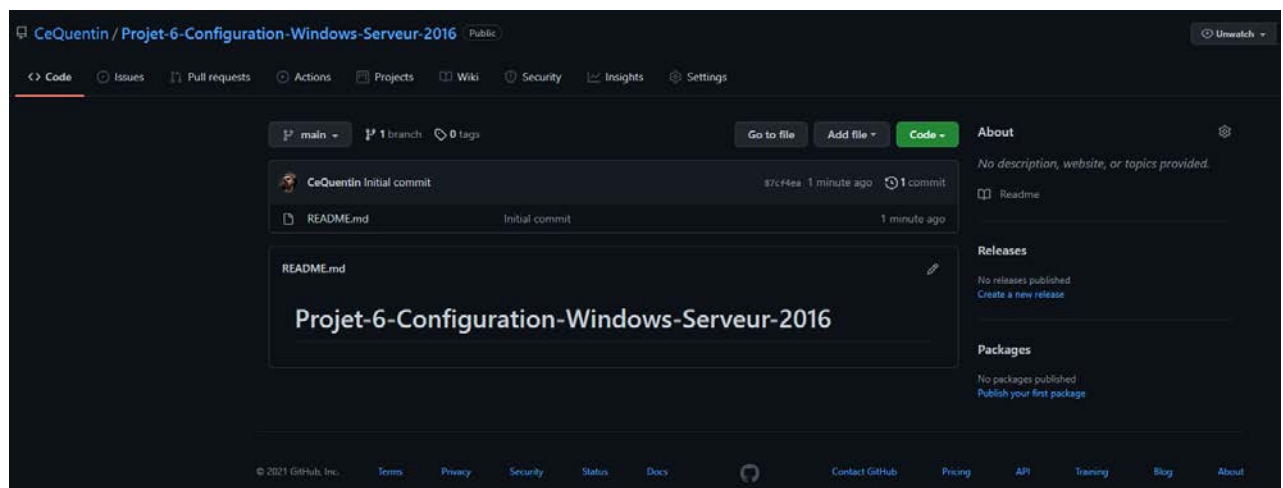
☐ **Add .gitignore**  
Choose which files not to track from a list of templates. [Learn more.](#)

☒ **Choose a license**  
A license tells others what they can and can't do with your code. [Learn more.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

**Create repository**

Et voilà le résultat, le répertoire a été créé.





## Répertoire GitHub :



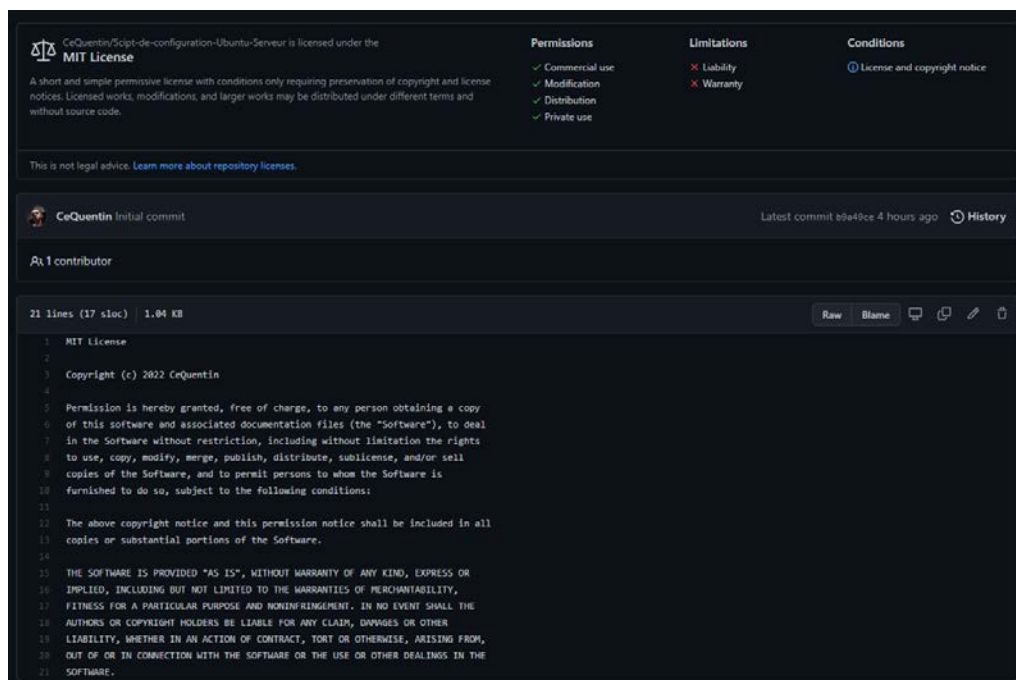
<https://github.com/CeQuentin/Script-de-configuration-Ubuntu-Serveur>

## Licence Open Source GitHub :

Il nous faut maintenant ajouter une licence open source sur notre projet.

Les licences Open Source permettent à d'autres personnes d'utiliser, de modifier et de distribuer librement le projet dans le référentiel. Nous ne sommes pas obligés de choisir une licence. Cependant, sans licence, nous conservons les droits du code et personne ne peut reproduire, distribuer ou créer des œuvres dérivées de ce code. Les trois licences open source les plus populaires sont **MIT**, **Apache 2.0** et **GPLv3**.

J'ai choisi la licence **MIT** car c'est l'une des plus utilisées et je souhaite donner l'accès à mon code à tout le monde.



# Conclusion

*Dans ce projet j'ai dû travailler sur la création d'un programme d'automatisation. Je suis parti sur un programme qui configure une adresse IP et installe les services DHCP, NTP et Samba sur Ubuntu. Enfin j'ai pu faire fonctionner ce programme en le testant sur une machine Ubuntu.*

# Annexe

[Home](#) / [Licenses](#)

## MIT License

Copy license text to clipboard

A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

| Permissions  | Conditions   | Limitations  |
|--|--|--|
| <ul style="list-style-type: none"><li>Commercial use</li><li>Distribution</li><li>Modification</li><li>Private use</li></ul> | <ul style="list-style-type: none"><li>License and copyright notice</li></ul> | <ul style="list-style-type: none"><li>Liability</li><li>Warranty</li></ul> |

MIT License

Copyright (c) [year] [fullname]

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Suggest this license

Make a pull request to suggest this license for a project that is **not licensed**. Please be polite: see if a license has already been suggested, try to suggest a license fitting for the project's **community**, and keep your communication with project maintainers friendly.

Enter GitHub repository URL

How to apply this license

Create a text file (typically named LICENSE or LICENSE.txt) in the root of your source code and copy the text of the license into the file. Replace [year] with the current year and [fullname] with the name (or names) of the copyright holders.

Optional steps

Add MIT to your project's package description, if applicable (e.g., [Node.js](#), [Ruby](#), and [Rust](#)). This will ensure the license is displayed in package directories.

[Source](#)

Who's using this license?

[Babel](#)  
[.NET Core](#)  
[Rails](#)

# Lexique

`#!/usr/bin/python 3` : localisation de Python

`# coding: utf-8` : codage de caractères informatiques permettant de coder l'ensemble des caractères internationaux.