

SAE 3.02 : Conception d'une architecture distribuée avec routage en oignon, Fichier d'installation.

Sommaire :

1. Guide d'installation de l'environnement :

- 1.1. Installation de VirtualBox (p. 2-7)
- 1.2. Installation des iso nécessaire et configuration des machine virtuelle (p. 7-21)
- 1.3. Configuration des différente VM et du pc physique (p. 21- 56)

2. Comment utiliser les différentes applications :

- 2.1. Utilisation du script python pour le master (p. 57- 60)
- 2.2. Utilisation du script python pour les routeurs (p. 60- 63)
- 2.3. Utilisation du script python pour les clients (p. 63- 66)

3. Dépannage (p. 66- 67)

1. Guide d'installation de l'environnement :

1.1. Installation de VirtualBox (passer à la suite si déjà fait (suite p. 7)) :

Pour commencer utiliser le lien fournit sur le fichier README à la partie 3 vous arriverez sur cette page, sur un navigateur internet comme firefox, chrome etc. :

The screenshot shows the 'Download VirtualBox' page. On the left, under 'VirtualBox Platform Packages', there is a list of platform packages: Windows hosts, macOS / Intel hosts, macOS / Apple Silicon hosts, Linux distributions, Solaris hosts, and Solaris 11 IPS hosts. The 'Windows hosts' option is circled in red with a red arrow pointing to it from the left side of the image. On the right, there is a detailed description of the 'VirtualBox Extension Pack' and its license terms. At the bottom right of the main content area, there are three buttons: 'PUEL License FAQ', 'PUEL License Text', and 'Accept and download'.

Vous devriez voir ça :

The screenshot shows the same 'Download VirtualBox' page as above. The 'Windows hosts' option is again circled in red with a red arrow. To the right, there is a sidebar titled 'Historique des téléchargements récents' (Recent Downloads) which lists several files: 'VirtualBox-7.2.4-170995-Win.exe', 'VirtualBox-7.2.4-170995-Win.exe' (repeated), 'VSCodeSetup-arm64-1.107.1.exe', 'VSCodeSetup-x64-1.107.1.exe', 'VSCodeUserSetup-arm64-1.107.1.exe', 'VSCodeUserSetup-x64-1.107.1.exe', and 'debian-12.12.0-smd64-netinst.iso'. At the bottom right of the main content area, there is a 'Historique complet des téléchargements' (Full download history) link.

Vous devrez trouver l'installateur de VirtualBox dans navigateur de fichier dans téléchargement :

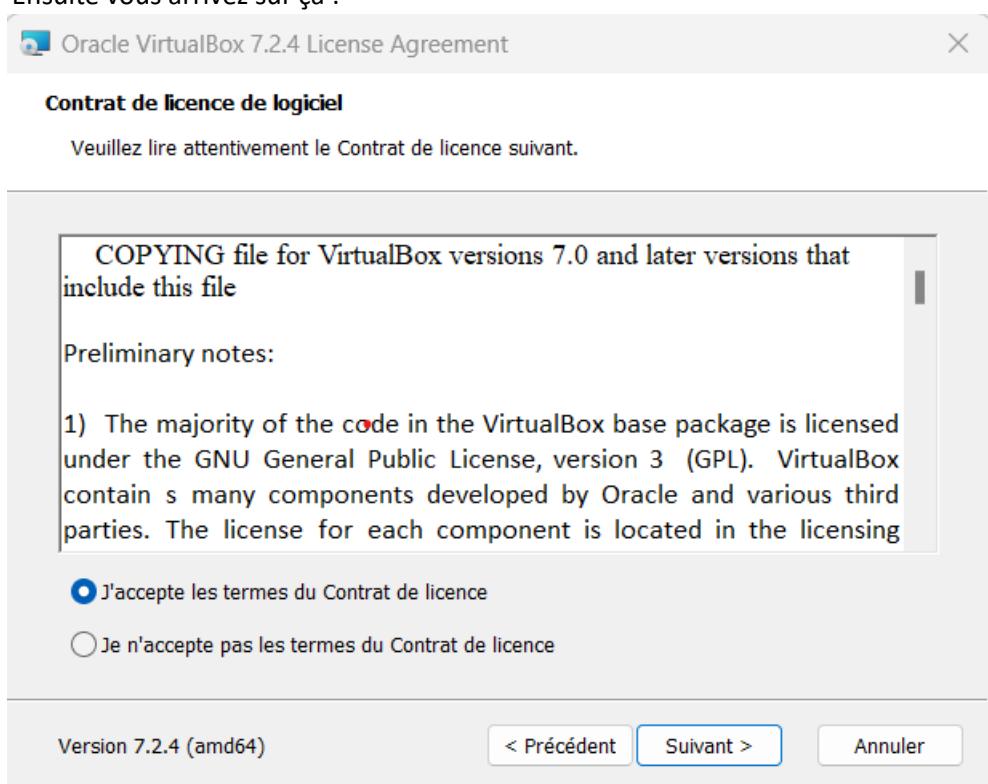


Cliquer dessus, vous aller arriver sur ça :



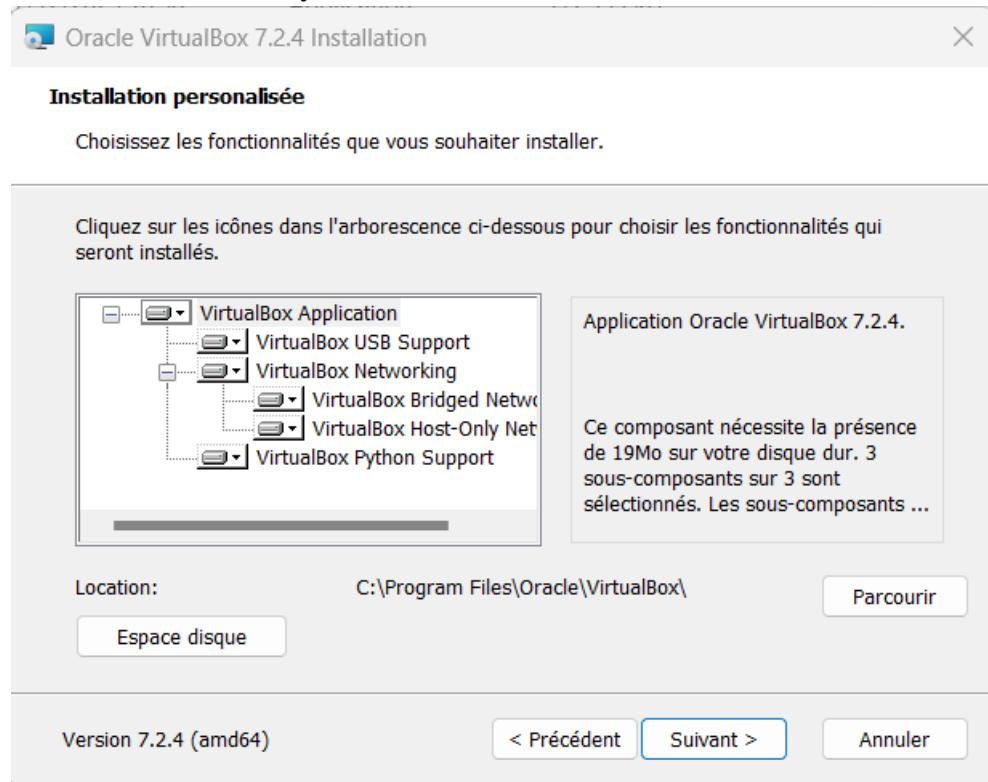
Cliquer sur suivant

Ensuite vous arrivez sur ça :



vous cocher « j'accepte les termes du Contrat de licence puis cliquer sur suivant

Ensute vous arriver sur ça :

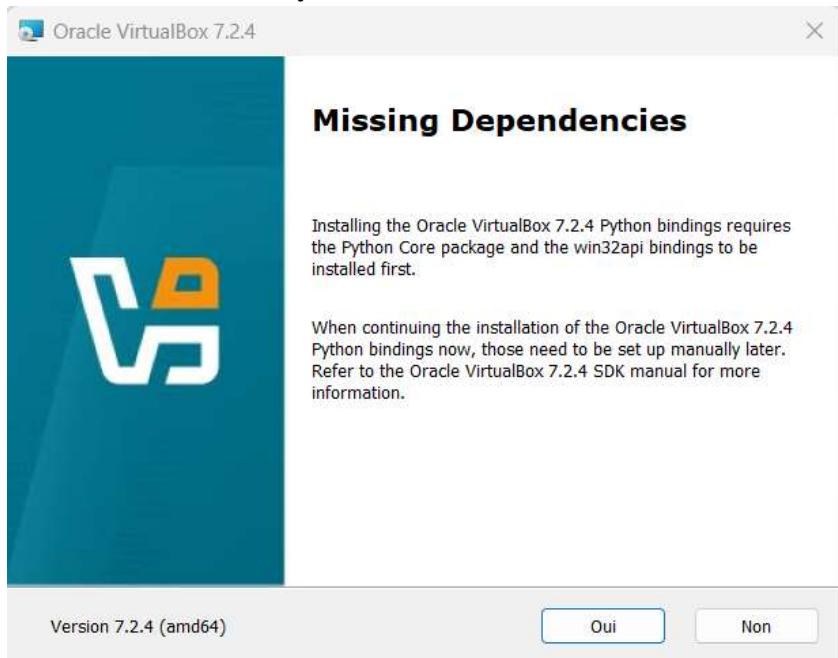


cliquer sur suivant vous arrivez sur ça :



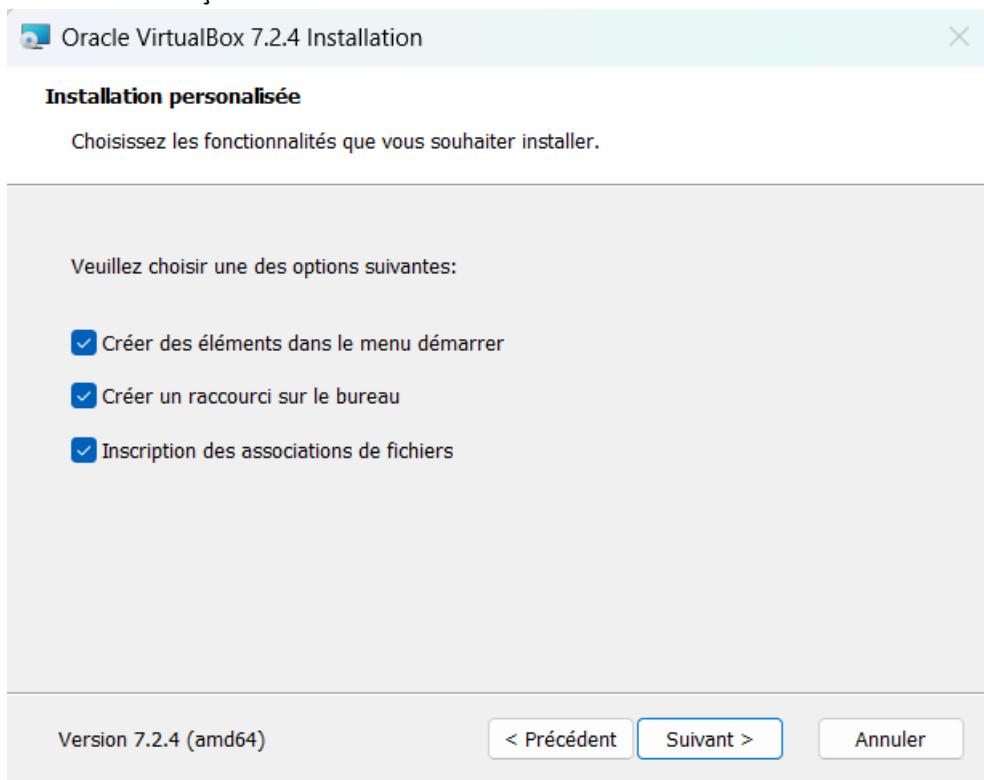
Cliquer sur oui.

vous arrivez ensuite sur ça :



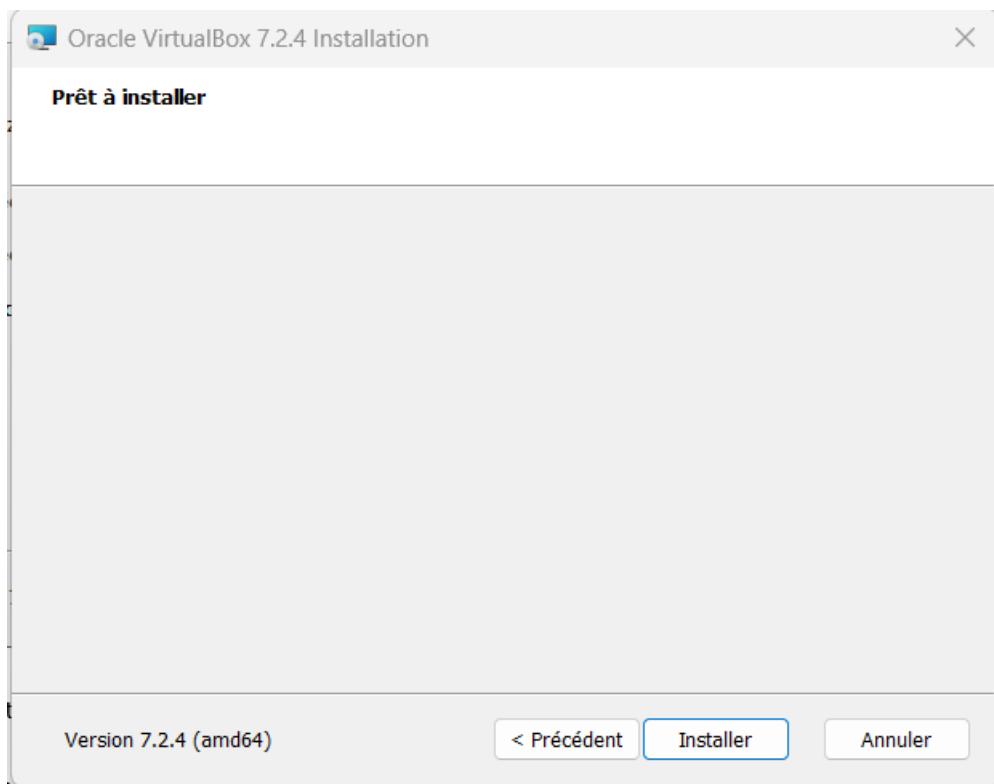
cliquer sur oui.

vous arrivez sur ça :



laisser tout cocher et cliquer sur suivant.

ensuite vous arriver sur ça :



Cliquer sur installer, laissez-le s'installer.

après avoir installé VirtualBox vous arriver à ça :



Laisser cocher et cliquer sur terminer et vous arriverez a la page d'accueil de VirtualBox :



Bravo vous avez installé VirtualBox maintenant nous allons nous occuper des machine virtuelle ou VM

1.2. Installation des iso nécessaire et configuration des machine virtuelle (passer a la suite si déjà fait (suite p. 21)) :

Maintenant nous allons crées le master, la VM qui gère les routeurs et le client B : pour commencer pour le master et la VM des routeurs nous allons utiliser le lien que j'ai fourni pour l'iso Debian 12 dans la partie 3 du README, vous arrivez dans cette page :

Rechercher... Rechercher

Actualités Tutorials Guides Téléchargements Windows Linux

Tendances Windows 11 : la mise à jour de décembre 2021 est disponible Fin de Windows 10 : que faire de son ancien PC ? Téléchargez le programme EDU de Windows 10 Windows 11 20H2 : comment installer la dernière mise à jour ?

Le Crabe Info | Téléchargements | Systèmes d'exploitation | Linux | Debian | Debian 12 (64 bits)

 **Debian 12** (64 bits)
50.9K vues · 28.6K · 4 commentaires

vous scroller jusqu'à arriver à ce niveau :



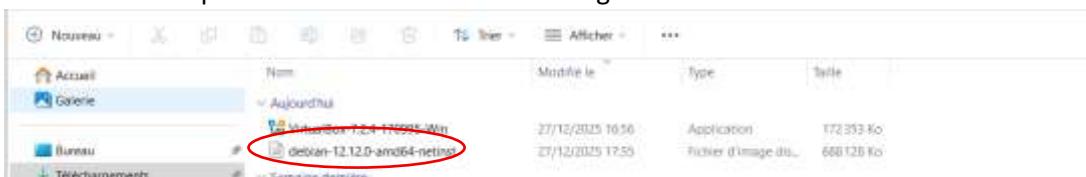
cliquer sur télécharger

vous devriez voir ça :



vous êtes en train d'installer l'iso Debian 12

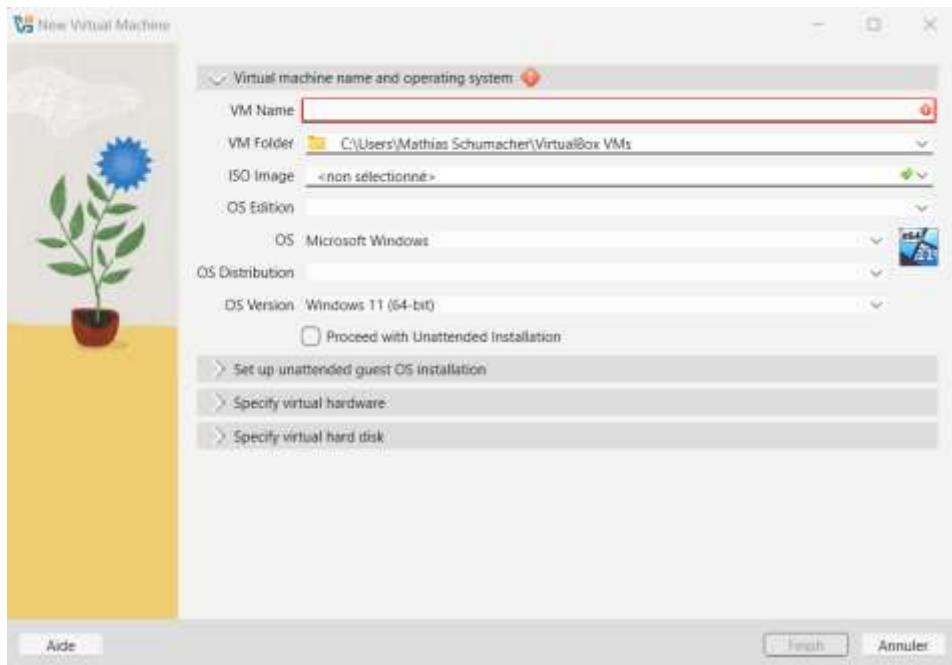
Vérifier dans l'explorateur de fichier dans téléchargement :



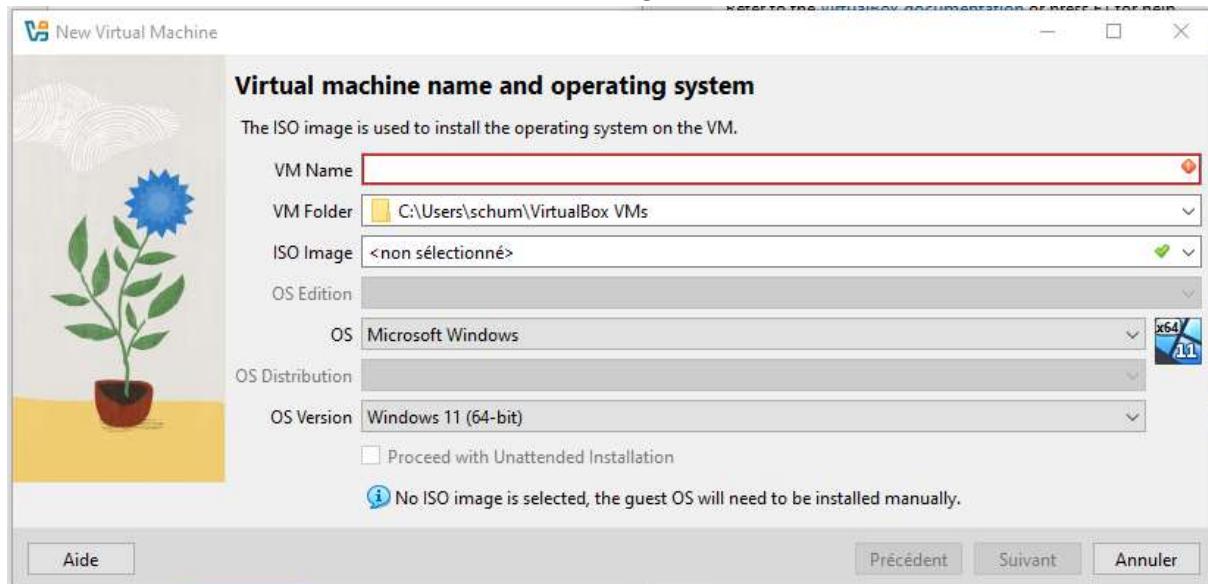
si vous voyez ce qui est entourée en rouge vous avez l'iso de Debian 12 téléchargé



Vous arriverez sur cette fenêtre :

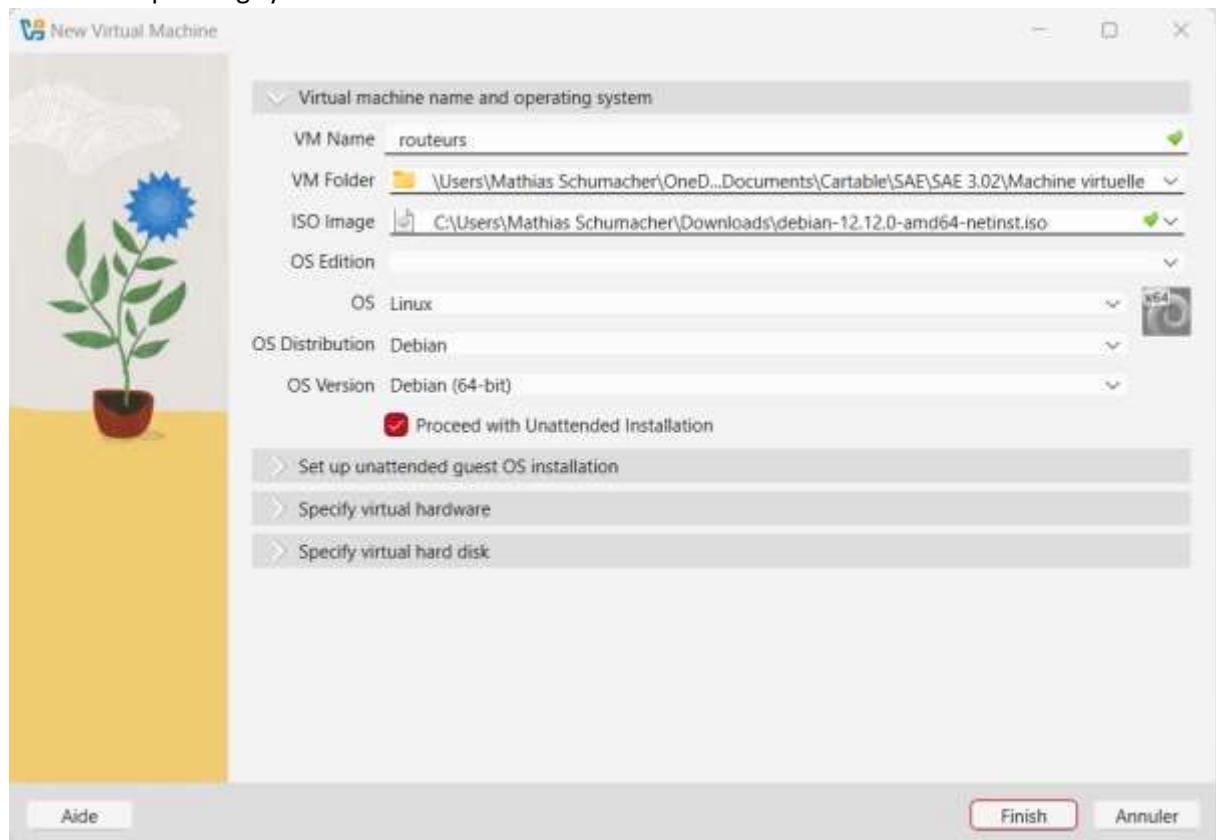


si vous êtes sous Windows 10, vous avez cette affichage :



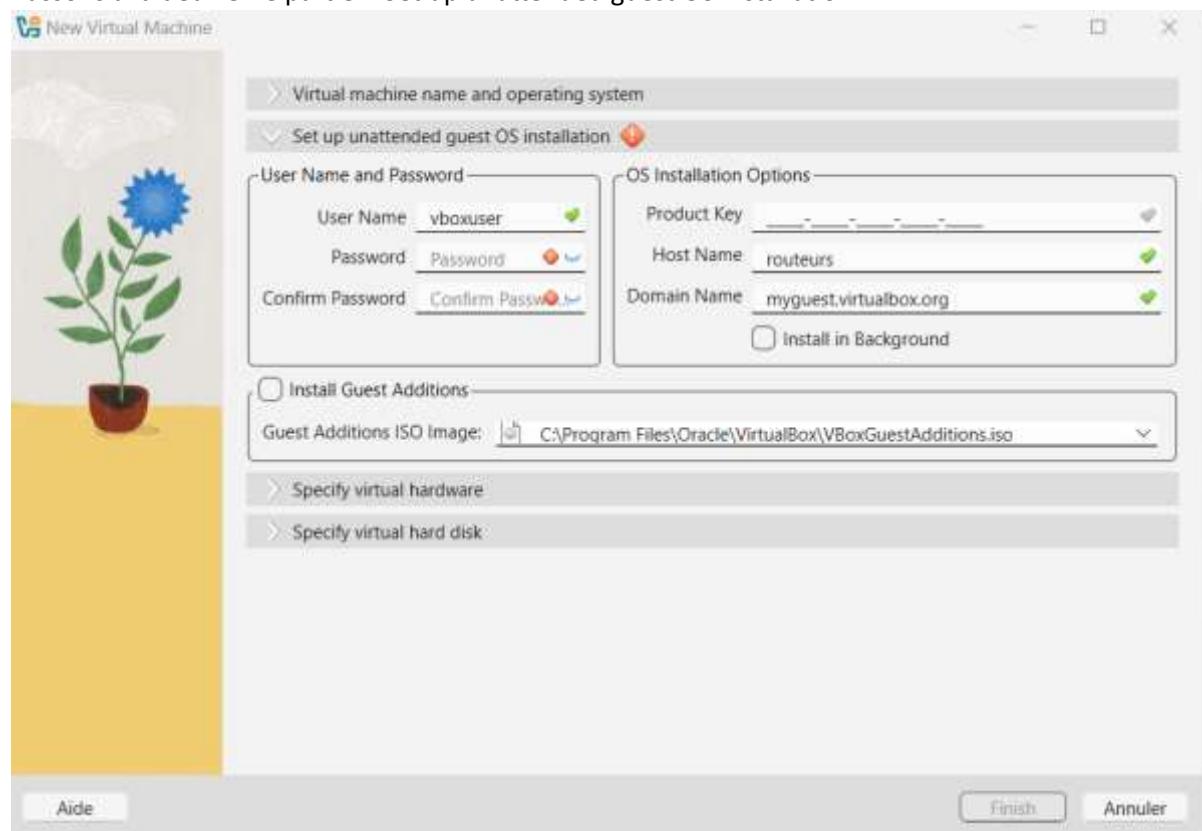
alors sa reste même manip mais pour passer a la suite vous appuierez sur « suivant ».

Maintenant nous allons passer à la configuration de la VM gérant les routeurs par « Virtual machine name and operating system »:

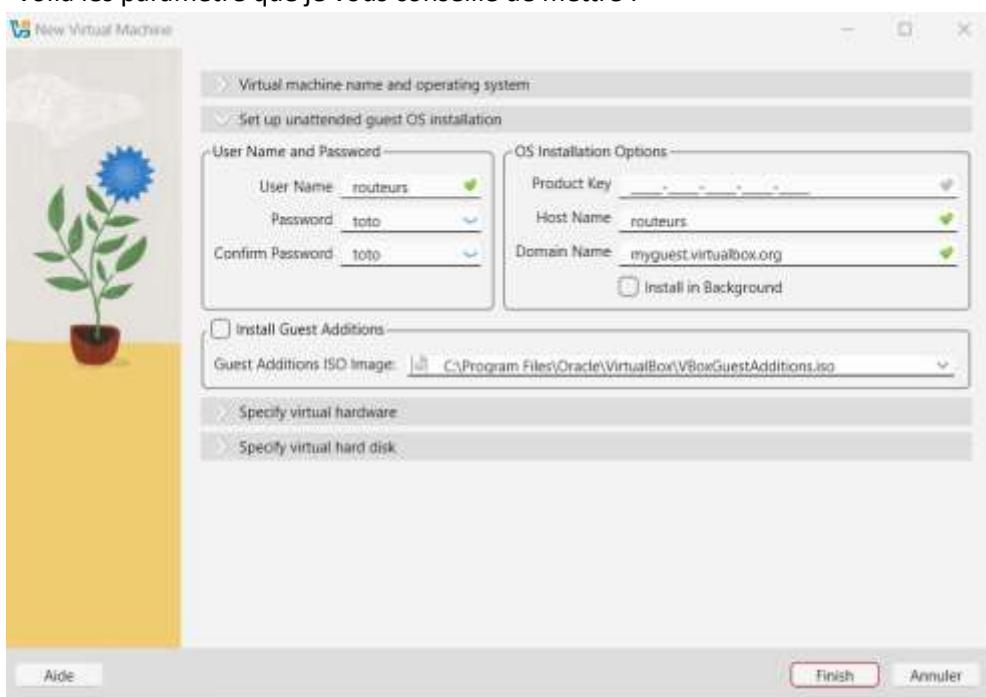


pour le VM name mettez « routeurs », VM folder vous pouvez laissez par défaut modifier pour un dossier particulier dans mon cas je l'ai stocker dans mon dossier « machine virtuelle » un dossier de mon projet SAE 3.02 pour l'iso vous sélectionner celui que vous avez installer dans téléchargement.

Passons à la deuxième partie « Set up unattended guest OS installation » :

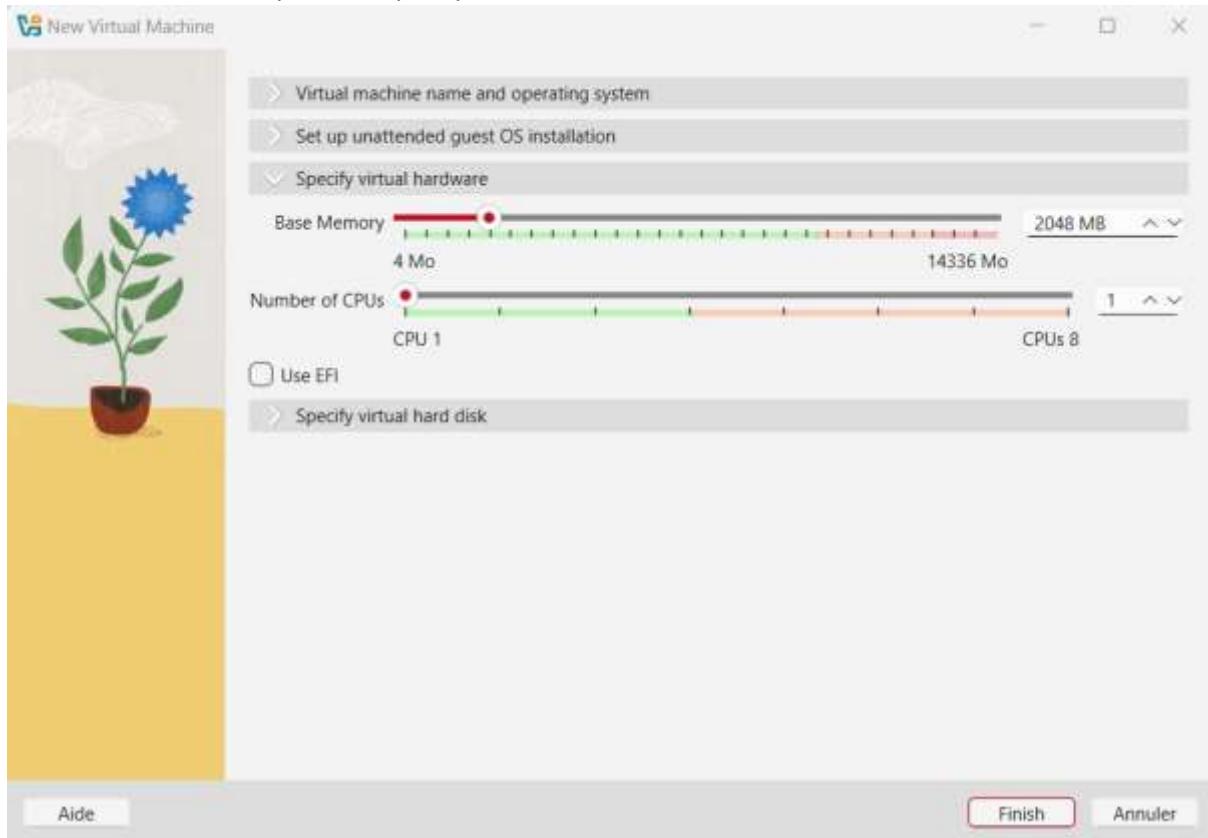


Voilà les paramètre que je vous conseille de mettre :



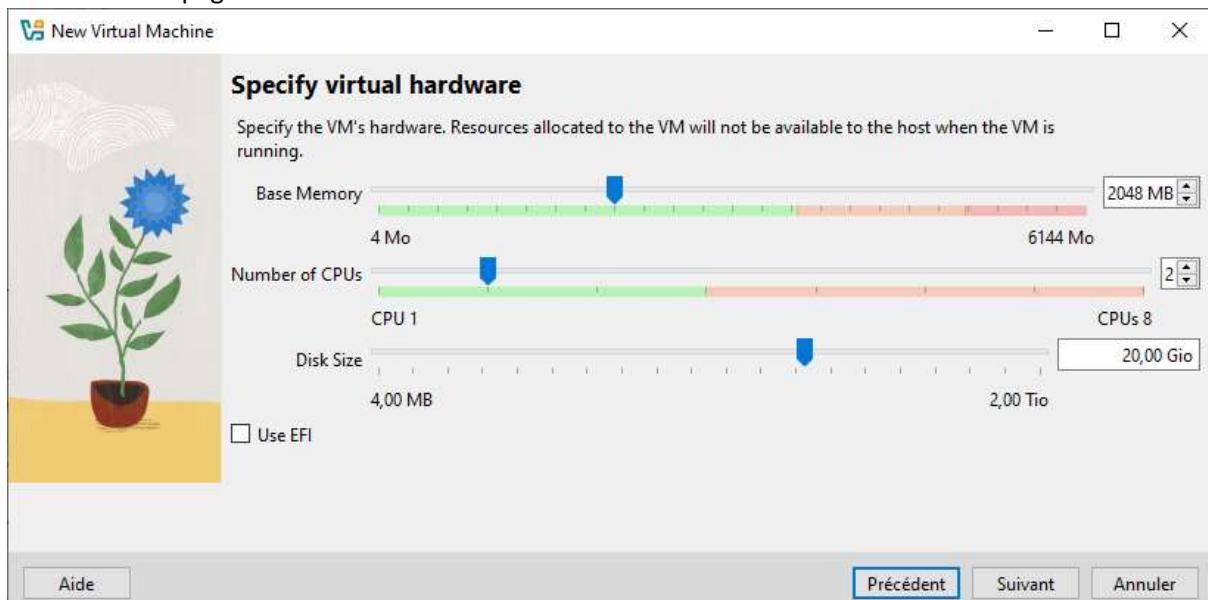
Pour User Name mettez « routeurs » avec comme mot de passe toto, le Product key vous le laissez vide et les autres paramètres de OS installation Option vous laisser comme montrer sur la capture

Passons à la troisième partie « Specify virtual hardware » :



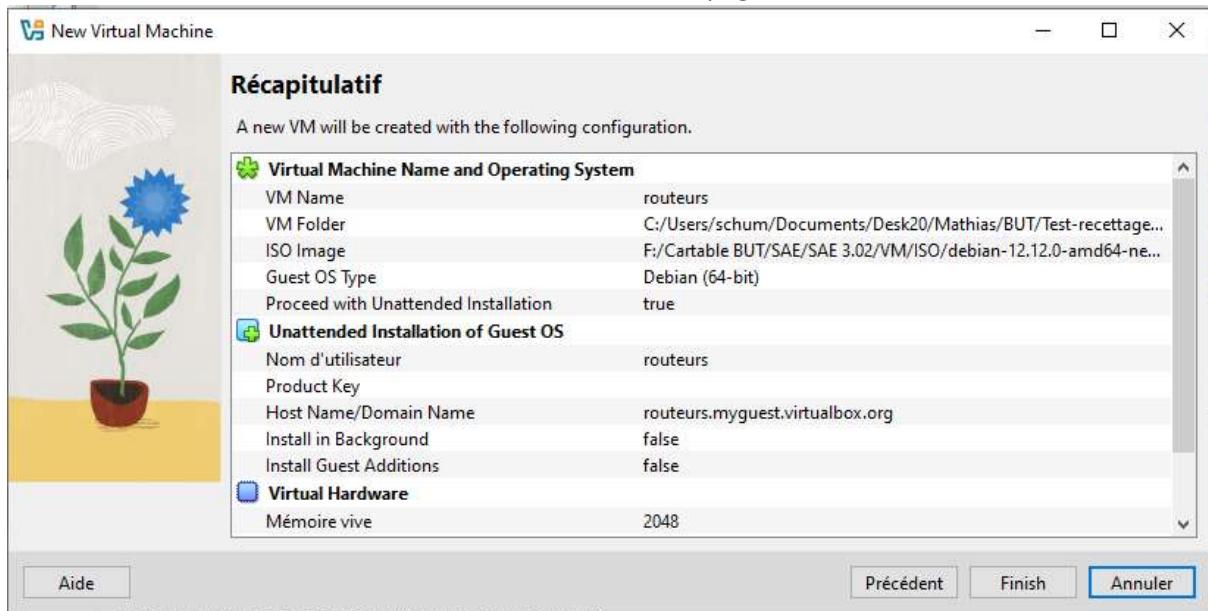
alors le seule paramètre a modifier est le « Number of CPUs » vous le mettez a 2

alors dans le cas de si vous êtes sur Windows 10, la mémoire pour votre disque dure virtuelle sera dans la même page :



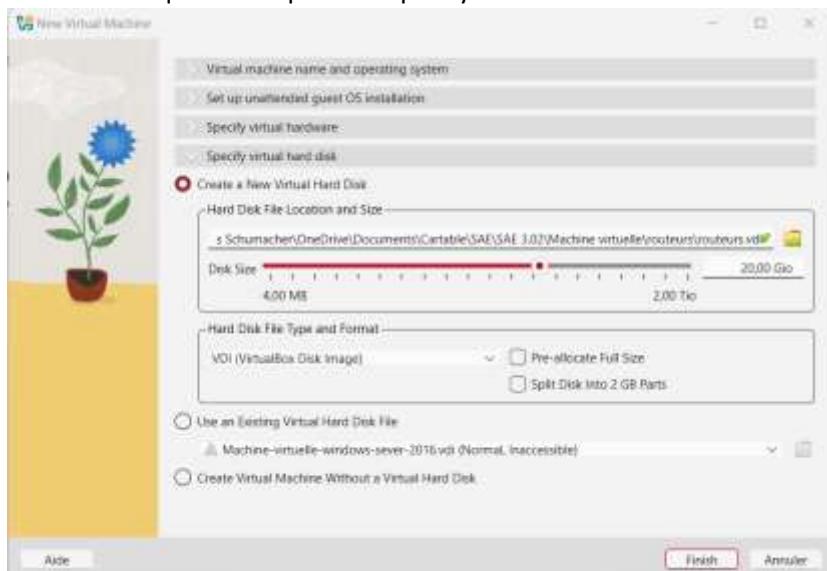
vous mettrez 40 Gio pour « Disk Size ».

Alors si vous êtes sur Windows 10, vous arriverez sur cette page :



vous cliquerez sur « finish » alors il faut savoir que si vous êtes sur Windows 10 pour les autres créations de VM vous aurez la même interface donc sa sera le même principe.

Passons à la quatrième partie « Specify virtual hard disk » :



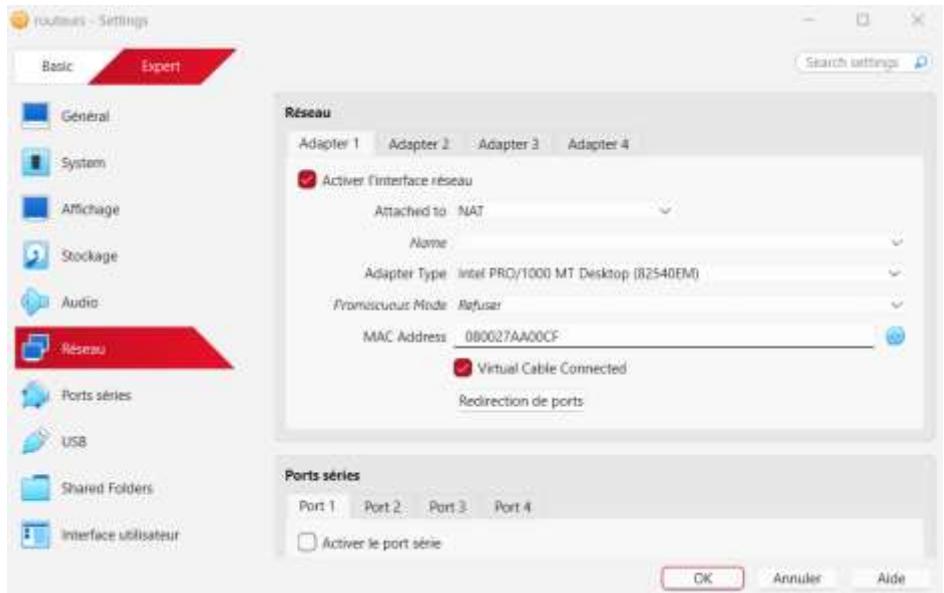
là vous mettez le disk Size à 40,00 Gio puis vous appuyer sur finish

après avoir appuyer sur finish sur l'onglet machine vous devriez voir :

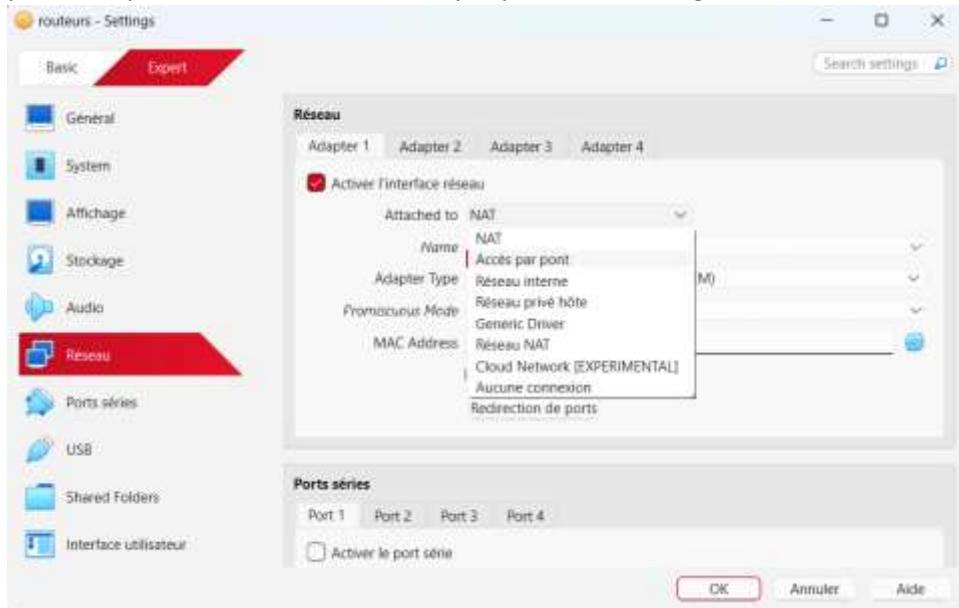


la vm routeurs

cliquer sur configuration, assurez-vous d'être en mode expert et cliquer sur réseau vous aller arriver sur cette page :

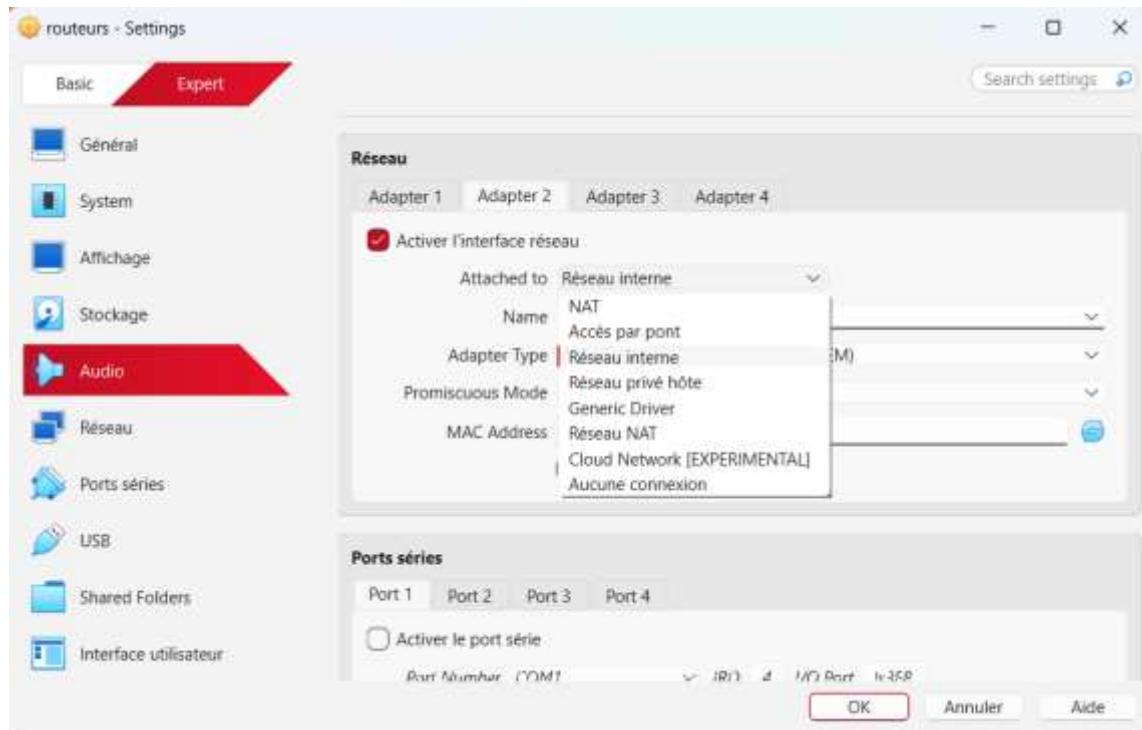


pour l'adapter 1 sélectionner « Accès par pont » ou « bridge » :



Ensuite cliquer sur adapter 2 :

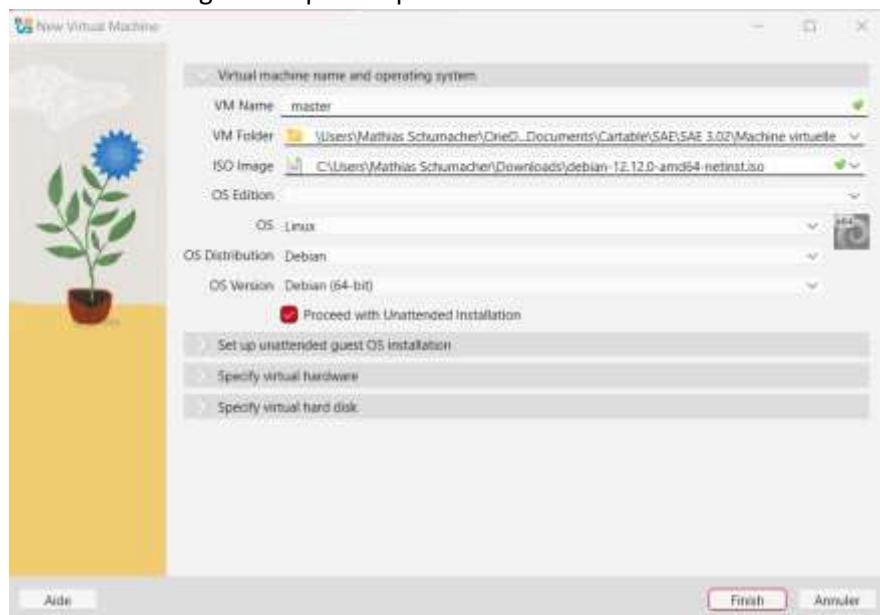
Et cocher activer l'interface réseaux et sélectionner « Réseaux Interne » :



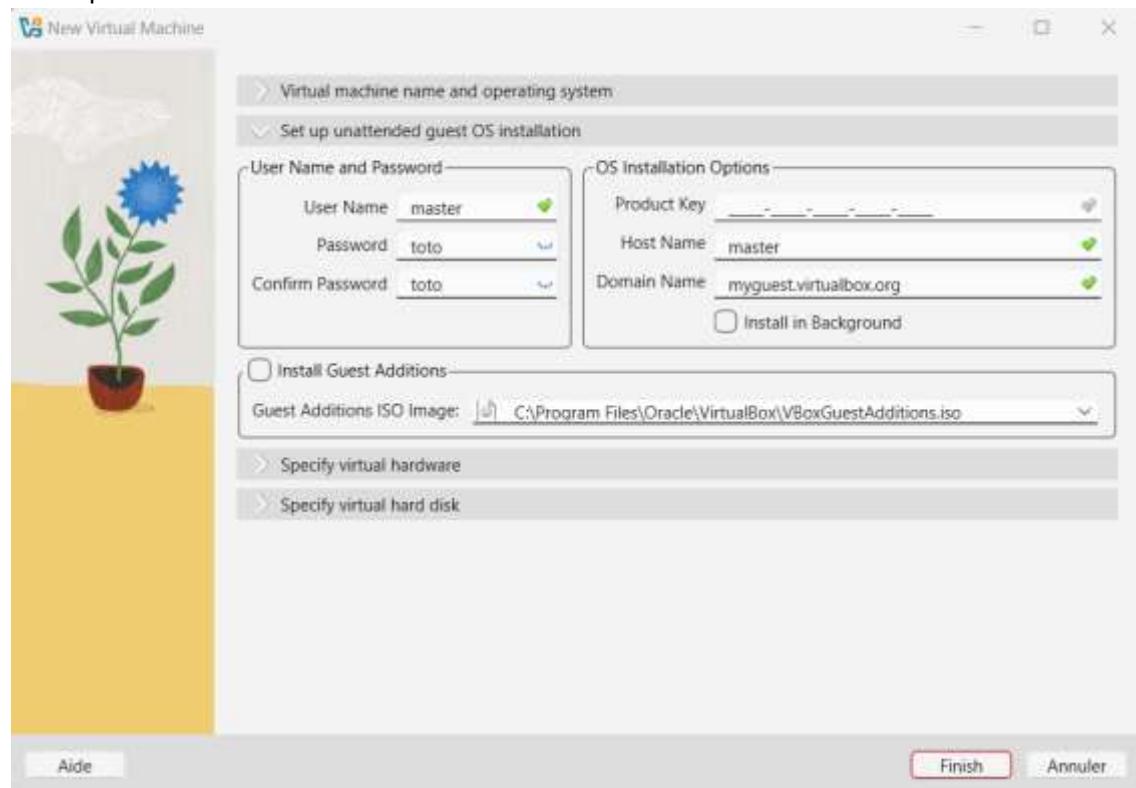
voilà vous avez configurer la VM routeur elle peut communiquer avec le pc physique et le reste du réseau interne.

Maintenant passons à la configuration du master :

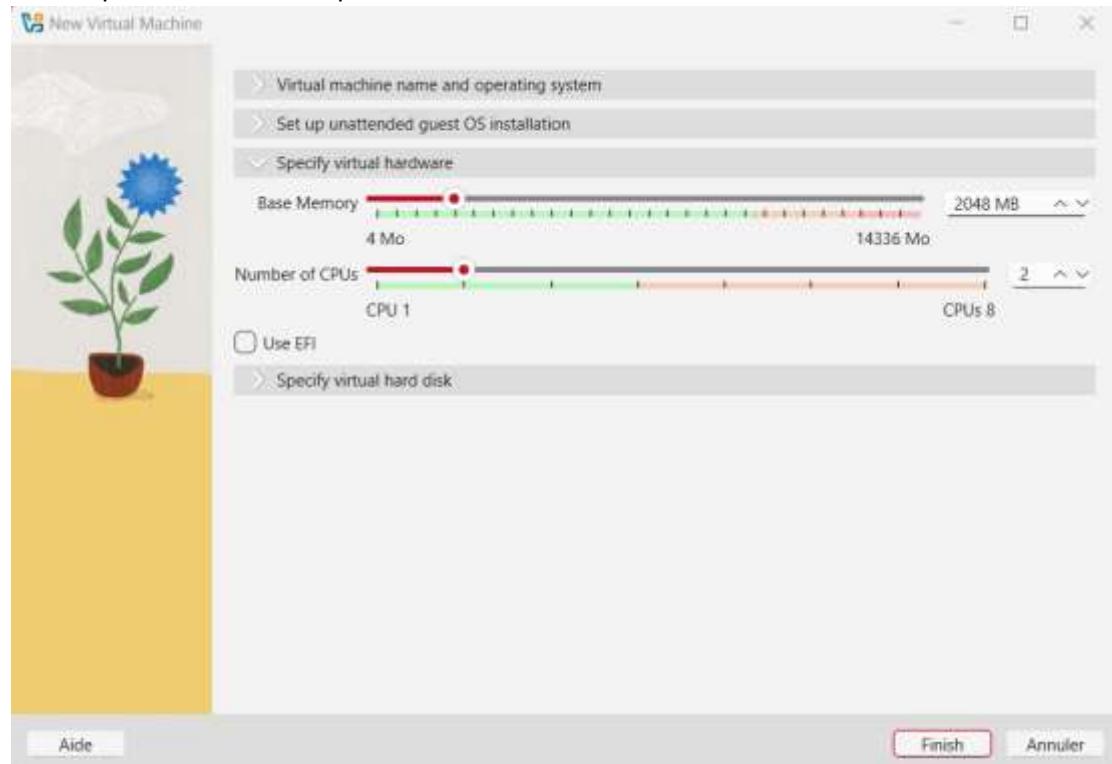
ces a peu près les mêmes paramètres pour le master dans la partie 1 vous lappelez master mais le reste de la configuration pour la partie 1 est la même :



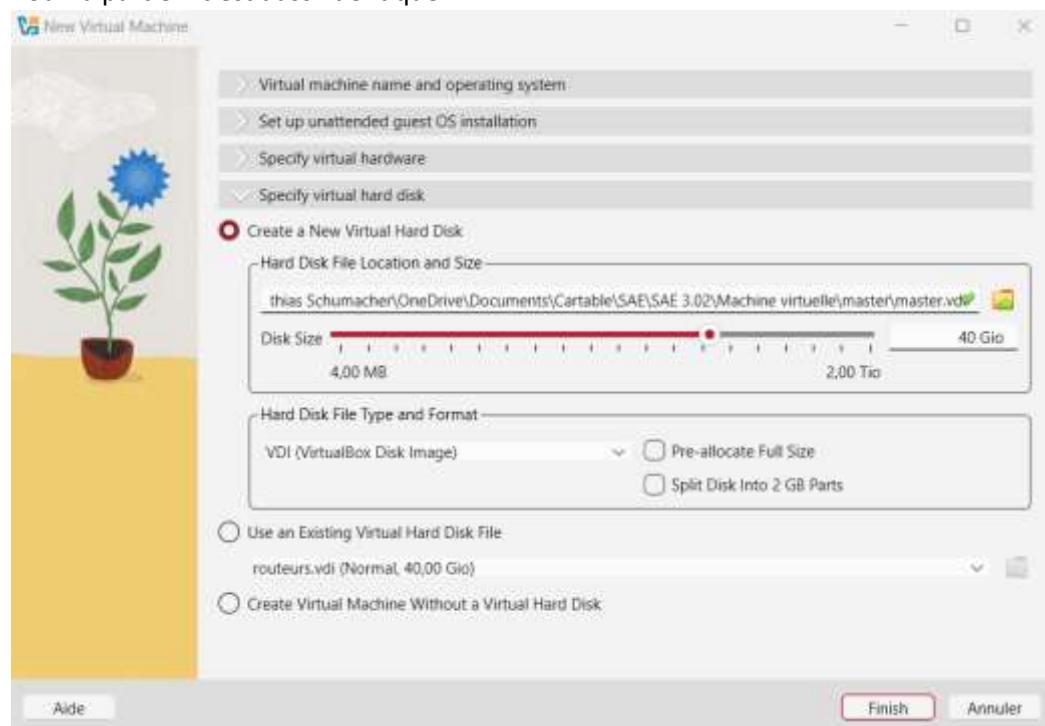
Pour la partie 2 c'est la même configuration mais vous mettez User name « master », le reste est identique :



Pour la partie 3 c'est identique :



Pour la partie 4 c'est aussi identique :



cliquer sur finish, devriez voir dans l'onglet machines ça :



pour les paramètre réseau le master sera en intnet mais n'aura que 1 interface :



pour l'installation de Debian 12 je vous renvoie a ce tuto pour installer Debian 12 en graphique :
<https://myitknowledge.fr/installation-de-debian-12-sur-virtualbox/>

Maintenant passons à l'installation et la configuration de la VM Windows qui fera office de client B, tout d'abords servez-vous du lien que je vous est fourni dans le README pour installer l'iso Windows 10 dans la partie 3 du README, vous devriez arriver à cette page :

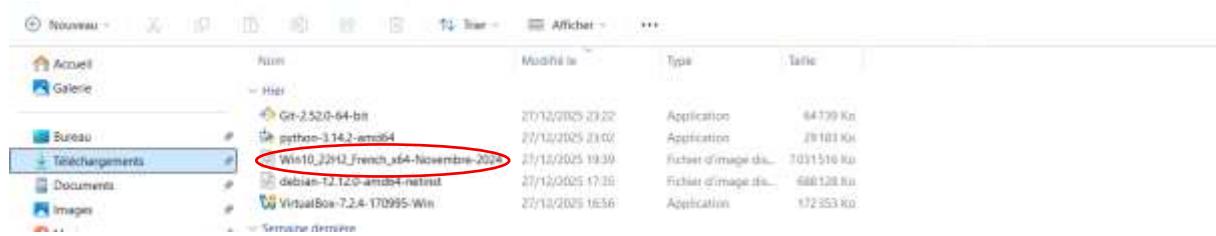


cliquer sur télécharger vous devriez voir ça :

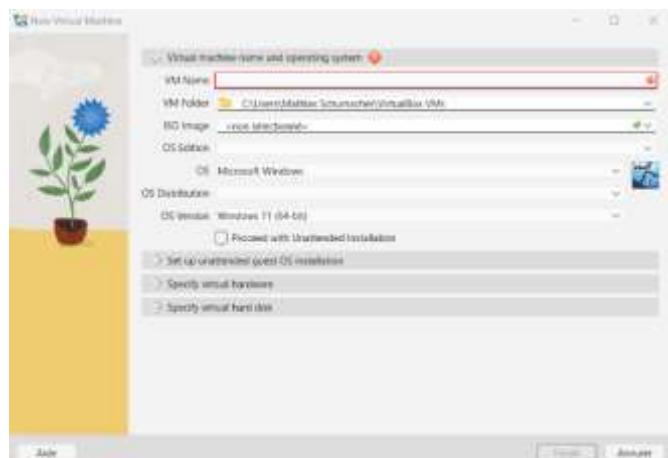


il est en train de s'installer.

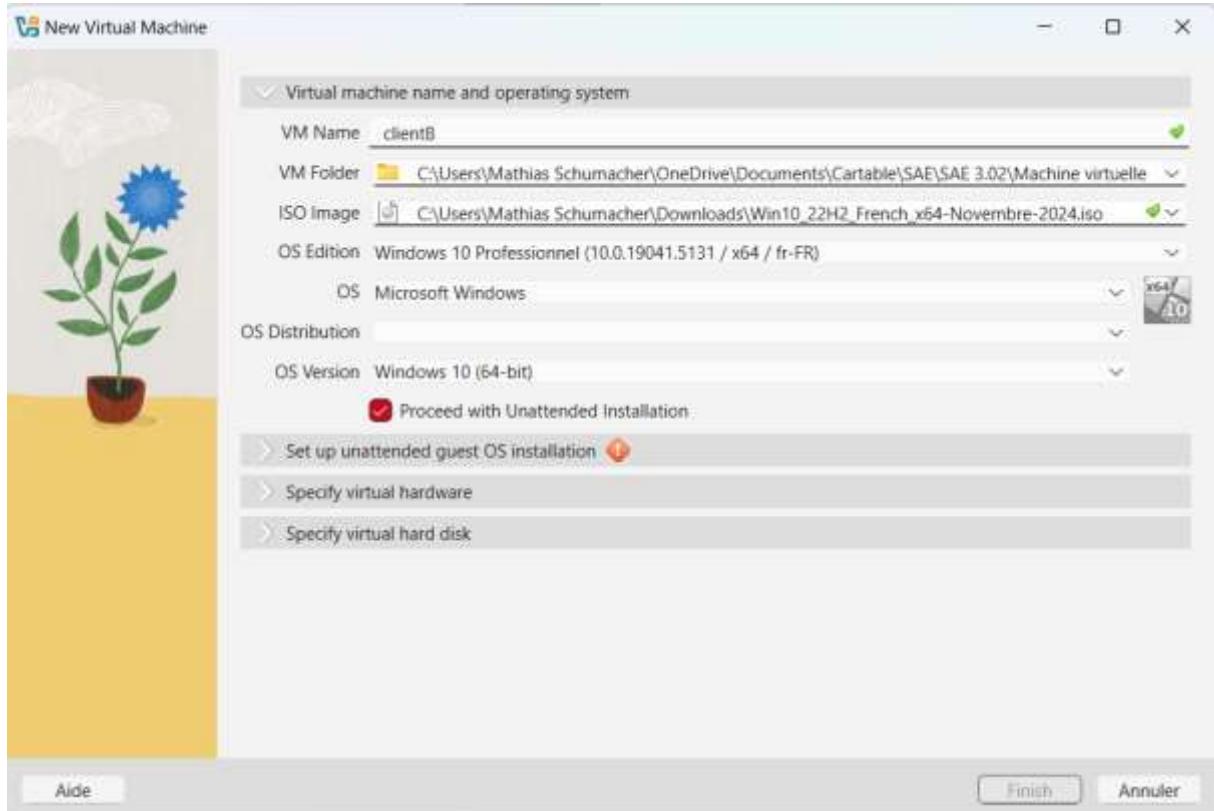
après que votre iso Windows 10 est installer, aller dans explorateur de fichier et dans téléchargement vous devriez voir ça :



maintenant nous allons passer à la configuration de la VM Windows 10 :

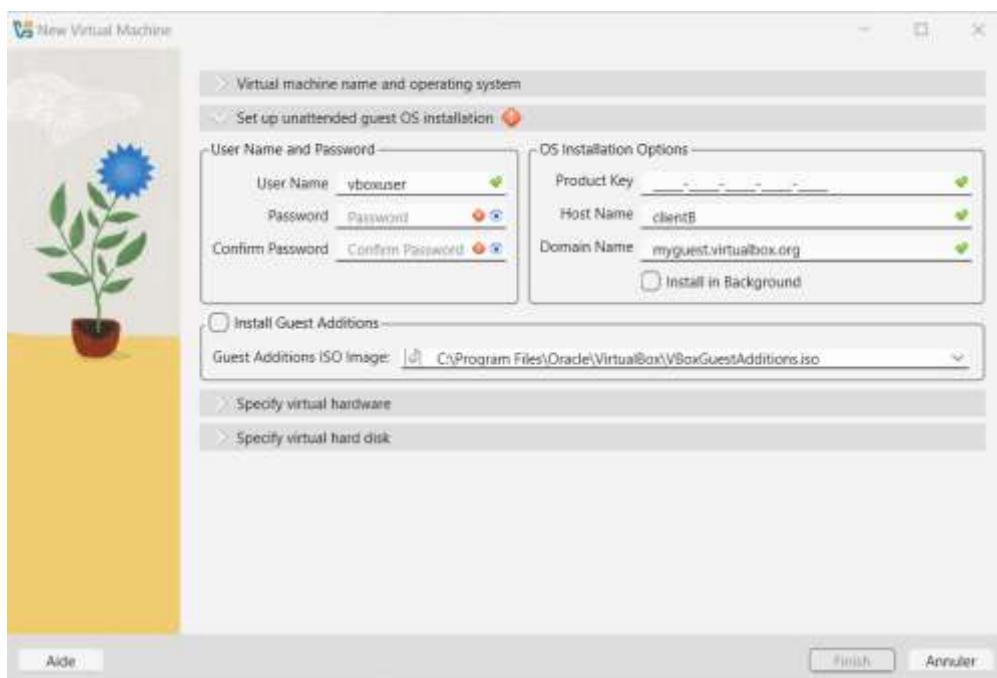


Voilà les config qu'il faut faire pour Virtual machine name and operating system :

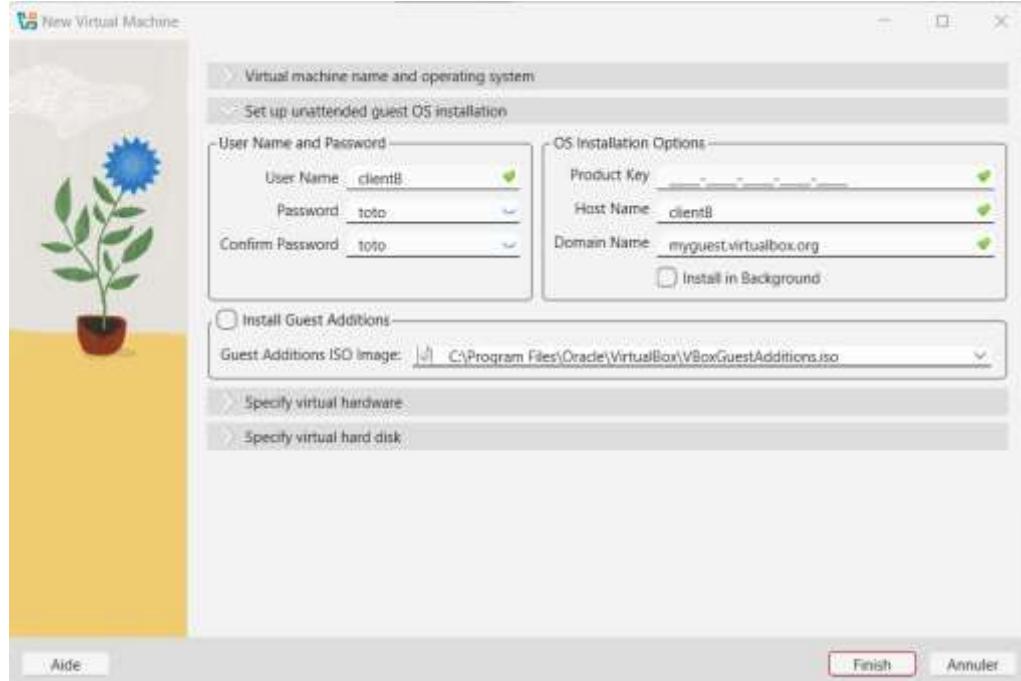


pour VM Name je vous conseille de mettre clientB, pour VM Folder vous pouvez laissez le chemin par défaut ou mettre un chemin que vous voulez pour retrouver facilement le dossier de votre VM, pour l'ISO image sélectionner l'iso que vous avez installer dans téléchargement, pour l'OS Edition choisissez Windows 10 professionnel car vous pourrez faire plus de chose avec une version professionnel de Windows 10.

Passons ensuite à Set up unattended guest OS installation :

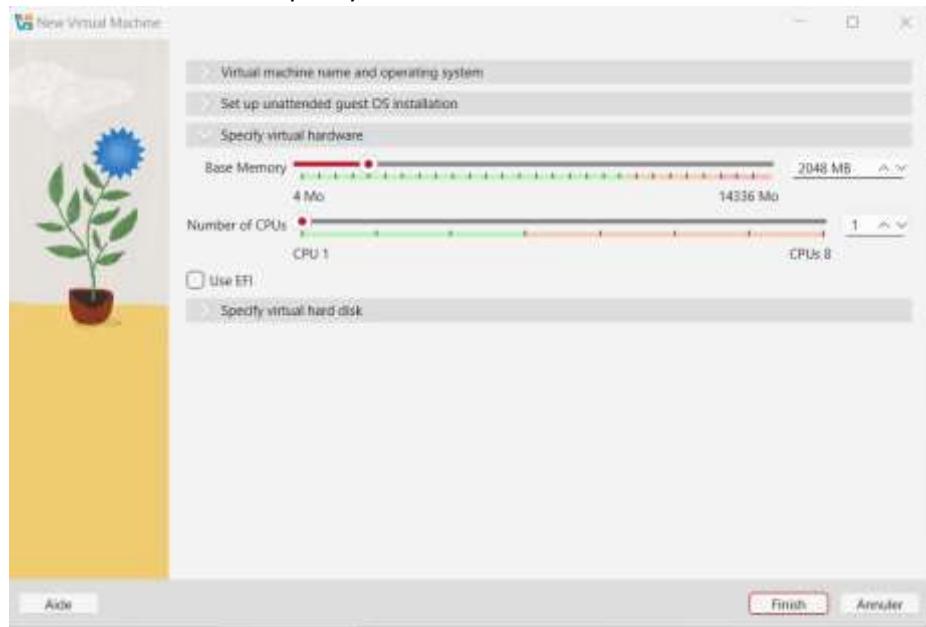


Voilà les paramètre a mettre pour cette partie :



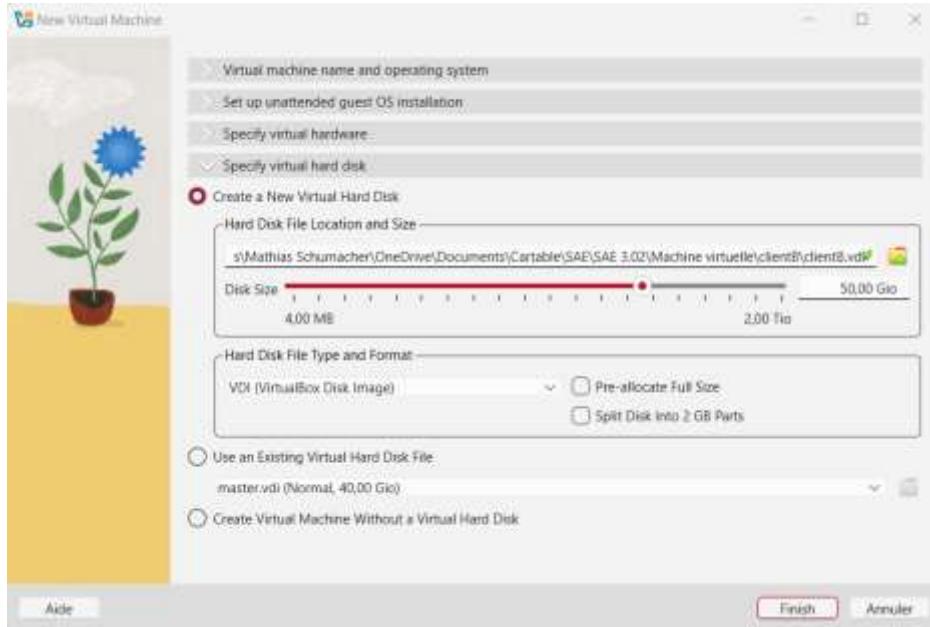
Pour User Name mettez clientB, pour le Password toto, le Product Key vous laissez vide et pour Host Name et Domain Name laissez ce qui est écrit

Passons maintenant à Specify virtual hardware :



Vous mettez en Base Memory 4096 MB et pour Number of CPUs mettez 3.

Pour finir dans Specify virtual hard disk :



Pour Disk Size vous mettez 64 Gio.

pour finir comme pour le master vous mettez le réseau en mode réseaux interne dans le réseau « intnet »

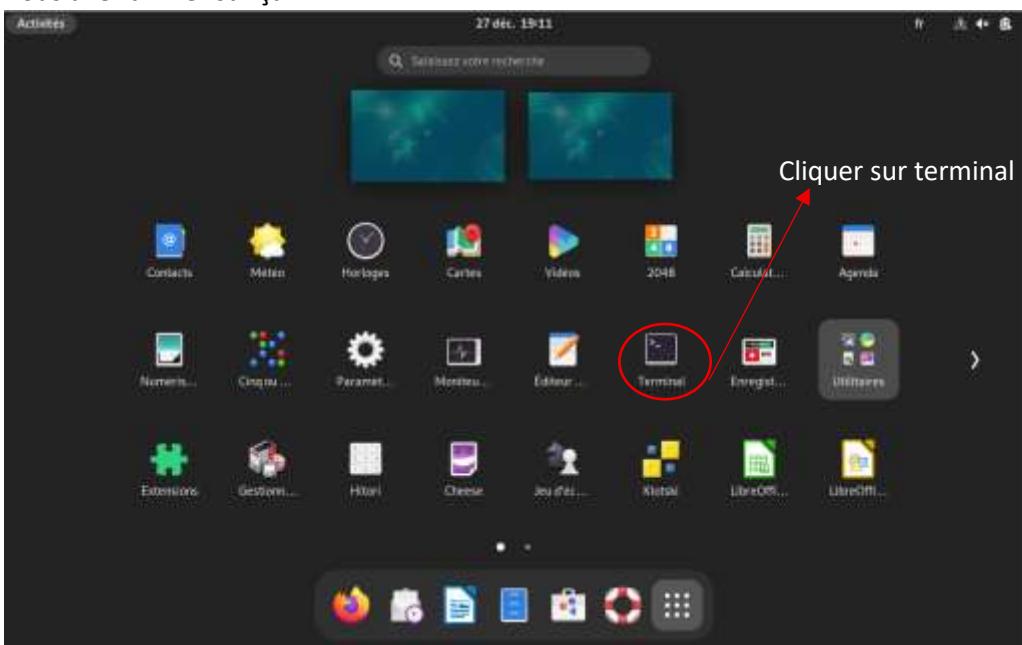
1.3. Configuration des différentes VM et du pc physique :

Configuration dans la VM routeurs :

pour commencer quand vous serez connectez à votre session aller dans « afficher application » :



Vous allez arriver sur ça :



Terminal va vous ouvrir cette page :

A terminal window is shown with the title "routeurs@vbox: ~". The command line prompt is "routeurs@vbox:~\$". The window has standard OS X-style controls at the top right.

pour savoir les adresse ip des différente interface faite la commande « ip addr » :

```
routeurs@vbox:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:a9:ff:2d brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.41/24 brd 192.168.1.255 scope global dynamic enp0s3
        valid_lft 85913sec preferred_lft 85913sec
    inet6 2a01:cb01:a0d:3600:a00:27ff:fea9:ff2d/64 scope global dynamic mngtmpad
        valid_lft 86359sec preferred_lft 559sec
    inet6 fe80::a00:27ff:fea9:ff2d/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:d8:57:4e brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.1/24 brd 10.0.0.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fed8:574e/64 scope link
```

garder en tête les adresses ip intnet pour la suite

ensuite on va devoir paramétrer les interface voici la commande « sudo nano /etc/network/interfaces » si sa vous affiche ça après avoir mis le mot de passe :

```
routeurs@vbox:~$ sudo /etc/network/interfaces
```

[sudo] Mot de passe de routeurs :

routeurs n'est pas dans le fichier sudoers.

alors faites su et mettez le mot de passe « normalement toto si vous avez suivi ma configuration », vous arriverez sur ça « en faisant la commande : nano /etc/network/interfaces sans le sudo car vous êtes admin » :



```
GNU nano 7.2          /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# Configuration de l'interface enp0s3 (WAN)
auto enp0s3
iface enp0s3 inet dhcp

# Configuration de l'interface enp0s8 (LAN)
auto enp0s8
iface enp0s8 inet static
    address 10.0.0.1
    netmask 255.255.255.0
    gateway 10.0.0.1

[ Lecture de 19 lignes ]
^G Aide      ^O Écrire      ^W Chercher      ^K Couper      ^T Exécuter      ^C Emplacement
^X Quitter   ^R Lire fich.  ^\ Remplacer   ^U Coller      ^J Justifier  ^/ Aller ligne
```

faite comme je le montre sur ma capture d'écran en adaptant en fonction de vos interface. Faites ensuite ctrl + X et vous sauvegarder.

pour valider vos paramètres faites la commande avec ou sans sudo en fonction de votre situation
« systemctl restart networking »

pour transformer votre VM debian en routeurs il faut faire la commande sans ou avec le sudo en fonction de votre situation « nano /etc/sysctl.conf » vous allez arriver sur ce fichier :



```
routeurs@vbox: ~
GNU nano 7.2          /etc/sysctl.conf

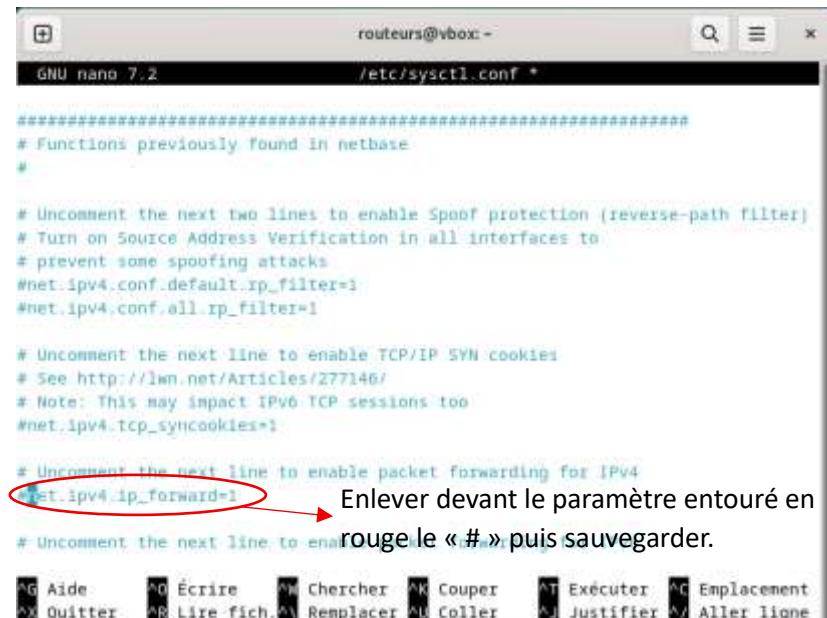
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#
#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
#
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

[ Lecture de 68 lignes ]
^G Aide      ^O Écrire    ^W Chercher  ^K Couper     ^T Exécuter  ^C Emplacement
^X Quitter   ^R Lire fich.  ^V Remplacer  ^U Coller     ^J Justifier ^/ Aller ligne
```

Scroller jusqu'à arriver à ce paramètre :



```
routeurs@vbox: ~
GNU nano 7.2          /etc/sysctl.conf *

#####
# Functions previously found in netbase
#
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1
Enlever devant le paramètre entouré en rouge le « # » puis sauvegarder.

# Uncomment the next line to enable packet forwarding for IPv4

[ Aide      ^O Écrire    ^W Chercher  ^K Couper     ^T Exécuter  ^C Emplacement
^X Quitter   ^R Lire fich.  ^V Remplacer  ^U Coller     ^J Justifier ^/ Aller ligne]
```

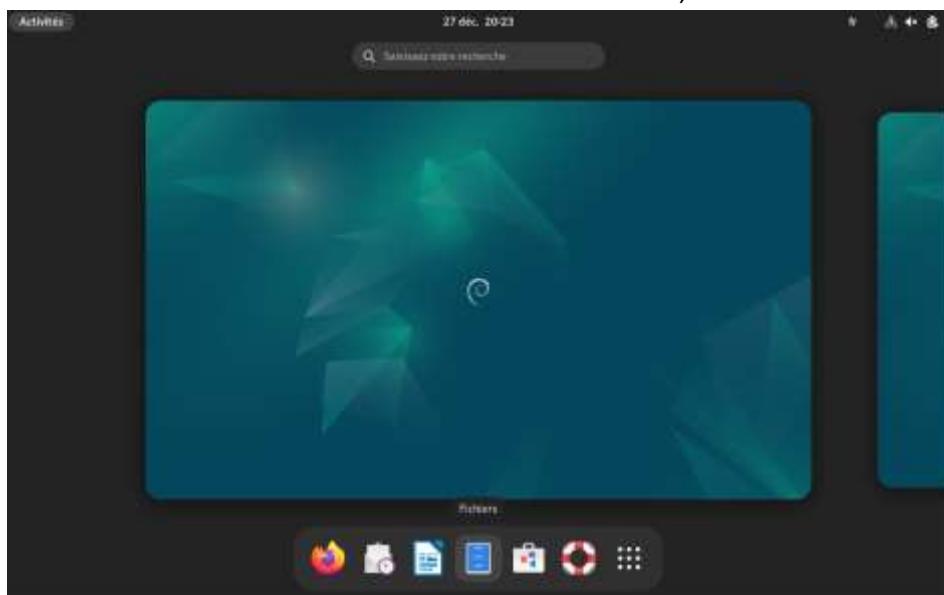
Pour redémarrer le réseau faire cette commande sans ou avec sudo en fonction de votre situation « sysctl -p ». pour vérifier si cela a été valider faire cette commande « cat /proc/sys/net/ipv4/ip_forward »

Maintenant nous allons installer iptables pour faire en sorte que les différentes VM en intnet relier au routeur il faut installer iptables, sur notre répertoire actuelle voilà ce qui est afficher :

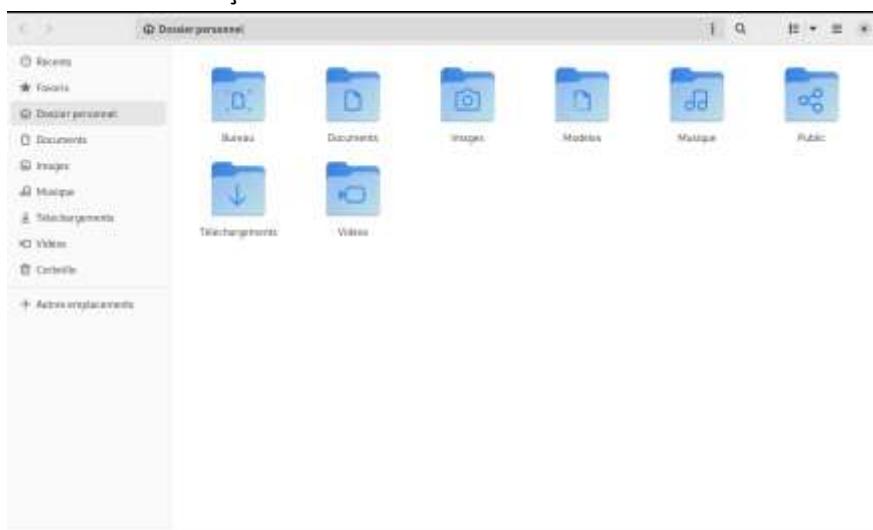
```
root@vbox:/home/routeurs# iptables  
bash: iptables : commande introuvable  
root@vbox:/home/routeurs# █
```

vous constaterez que malgré vous soyez admin vous n'arrivez pas à installer, ce très simple il faut faire la commande « su - » pour accéder au bon répertoire pour pouvoir installer iptables, avant d'installer iptables faire la commande « apt update » et une fois cela fait faire la commande « apt install iptables », une fois iptables installé faire la commande « iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE » adapter en fonction de votre interface dans mon cas c'est enp0s3 et pour rendre sa persistante il faut installer le paquet « apt install iptables-persistent » puis après avoir installé iptables-persistent faire la commande « netfilter-persistent save » pour rendre cette règle persistante.

ensuite nous allons voir la structure des dossier routeurs, tout d'abords il faut cliquer sur fichier :



Vous arriverez sur ça :



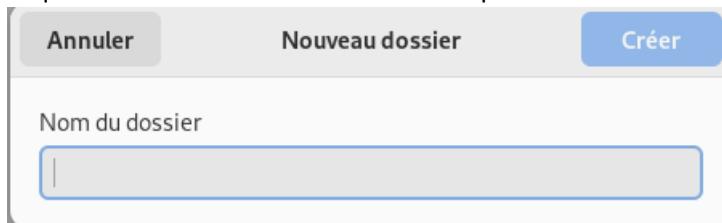
Cliquer sur document ou file en fonction de la langue vous devrez voir ça :



faire un clic droit :

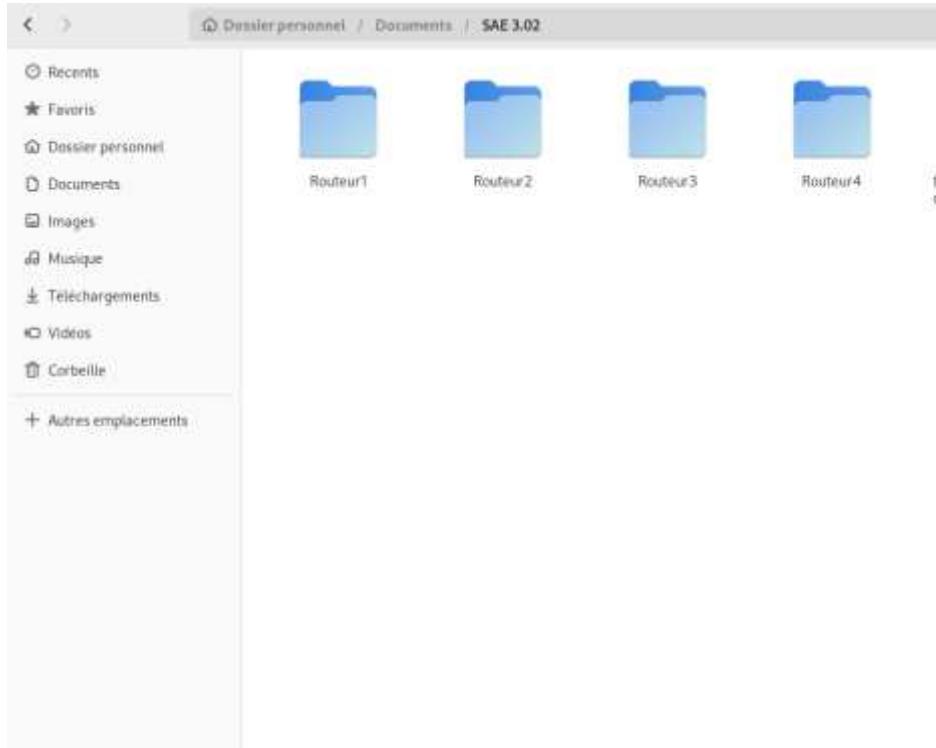


cliquer sur nouveau dossier ou new file pour créer un nouveau dossier, il faut ensuite écrire le nom :



vous appelez le dossier routeur1 puis routeur2 ainsi de suite.

dans ce dossier créer un dossier pour chaque routeur comme ci-dessous :



Ensute sur mon github aller dans le dossier script puis Routeur :

The screenshot shows a GitHub repository interface. The top bar shows the URL 'github.com/Warsoul68/SAE-3.02_Mathias_Schumacher_new/files/mainScript'. The repository name is 'SAE-3.02_Mathias_Schumacher_new'. The sidebar has tabs for 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. The 'Code' tab is selected. Under 'Files', there is a tree view with 'main' expanded, showing 'Document gestion de projet', 'Documentation', 'Script' (which is circled in red), 'Client', 'Master', 'Routeur' (which is also circled in red), 'Test fonctionnalité python', '.gitignore', 'BDD_pour_la_sae_3-02.sql', and 'README.md'. The main pane shows a list of commits under the 'Script' folder. One commit by 'Warsoul68' is highlighted: 'Modif dans le code du Master' (last week). Another commit by 'Warsoul68' is shown: 'Petite modif dans le code du chiffrement RSA' (last week). A third commit by 'Warsoul68' is shown: 'Modif dans le code du Master' (last week). A fourth commit by 'Warsoul68' is shown: 'correction de bug dans la classe routeur 2' (last week). Two arrows point to the 'Routeur' folder in the sidebar and the 'Routeur' folder in the commit list, with the text 'Cliquer sur Script' and 'Cliquer sur Routeur' respectively.

| Name | Last commit message | Last commit date |
|---------|----------------------------------------------|------------------|
| Master | Modif dans le code du Master | last week |
| Client | Petite modif dans le code du chiffrement RSA | last week |
| Routeur | correction de bug dans la classe routeur 2 | last week |

Vous arriverez sur cette page :

The screenshot shows a GitHub repository page for 'SAE-3.02_Mathias_Schumacher_new'. The 'Script/Routeur' folder is selected in the left sidebar. The main area displays a list of files: 'Routeur.py', 'chiffrement_RSA.py', and 'script_routeur.py'. A commit history for 'Routeur.py' is shown, with the most recent commit being 'correction de bug dans la classe Routeur 2' by 'Warsoul88' last week.

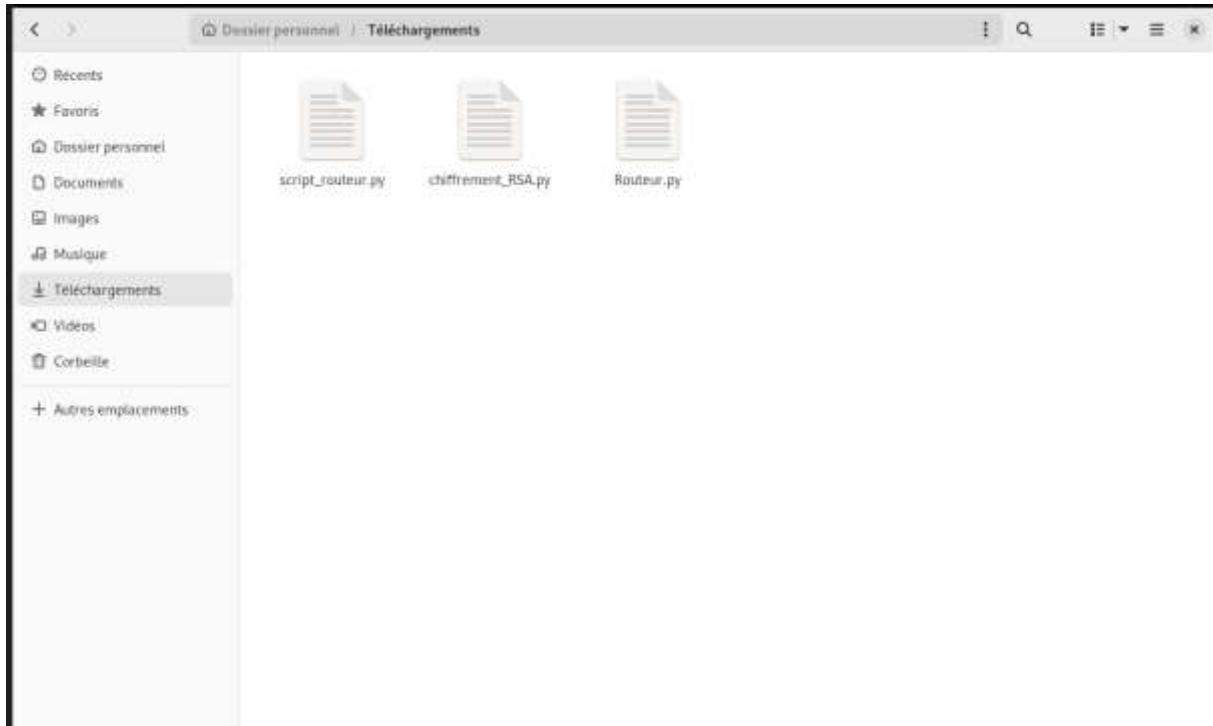
Cliquer par exemple sur Routeur, vous arriverez sur cette page :

The screenshot shows the GitHub code editor for the 'Routeur.py' file. The code is as follows:

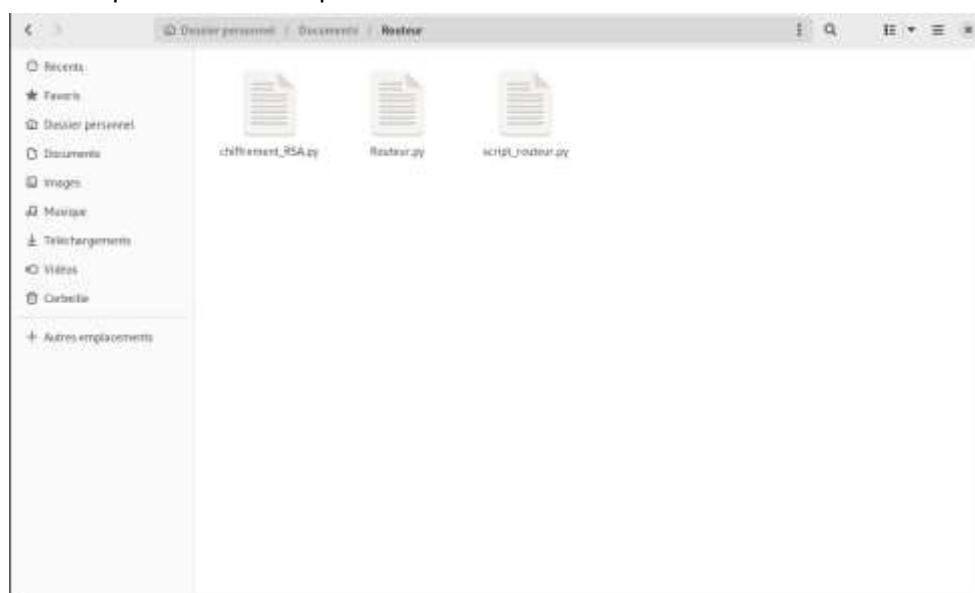
```
1 import socket
2 import threading
3 import sys
4 import datetime
5 import time
6 import random
7
8 # Importation de la classe de chiffrement
9 try:
10     from chiffrement_RSA import CryptoManager
11 except ImportError:
12     print("ERREUR CRITIQUE : le fichier chiffrement_RSA.py est introuvable !")
13     sys.exit(1)
14
15 def journalisation_log(quai, type_message, message):
16     maintenant = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
17     ligne_log = f"[{maintenant}] [{quai}] [{type_message}] {message}"
18     print(ligne_log)
19     nom_fichier = f"journal_{quai.lower()}.log"
20
21
```

A red arrow points to the download icon (a downward arrow) in the toolbar above the code editor, with the text 'Cliquer sur cette icône pour télécharger le script' (Click on this icon to download the script).

C'est le même principe pour chiffrement_RSA.py et script_routeur.py, tous ces scripts après l'installation devrait se trouver dans le dossier téléchargement ou download en fonction de la langue :



vous coupez coller ces scripts dans un dossier Routeurn comme ci-dessous :



Pour vérifier si vous avez python sur Debian 12 faire cette commande « python3 –version » si sa vous affiche ça, ça veut dire que python est installer :

```
routeurs@vbox:~$ python3 --version
```

```
Python 3.11.2
```

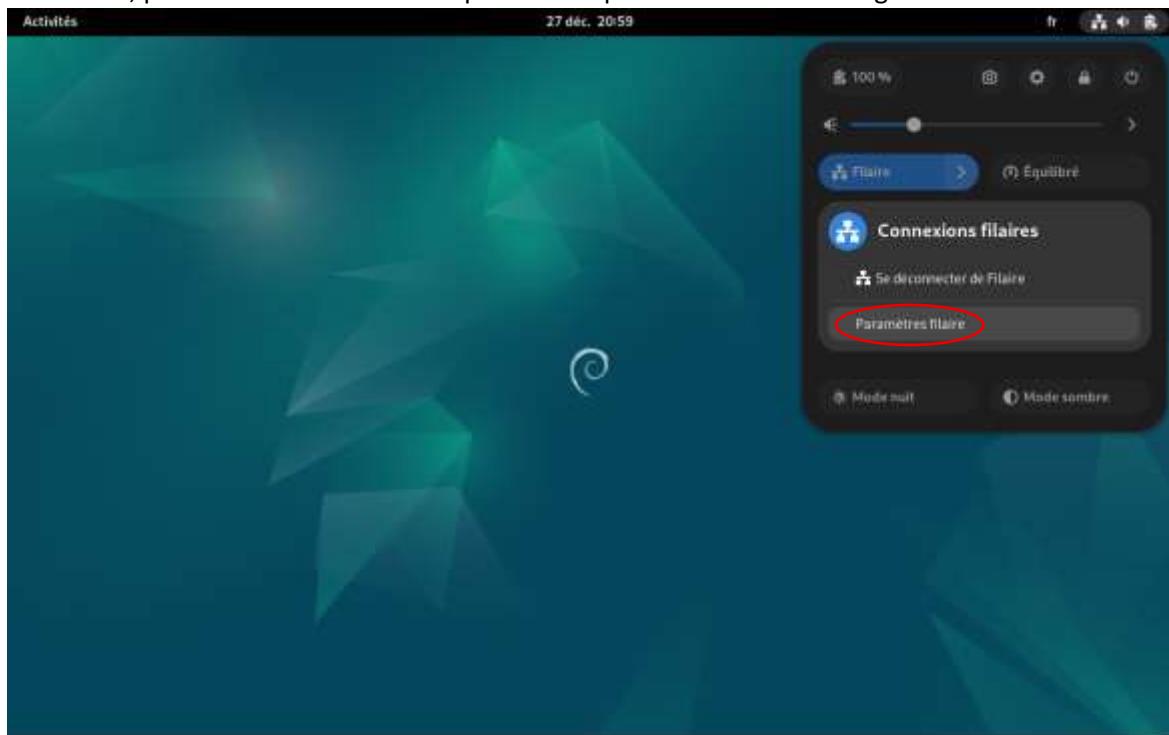
sinon faire cette commande pour l'installer avec ou sans sudo en fonction de votre situation « apt install python3 python3-pip -y »

Normalement si tout est bon vous pouvez passer à la configuration du master

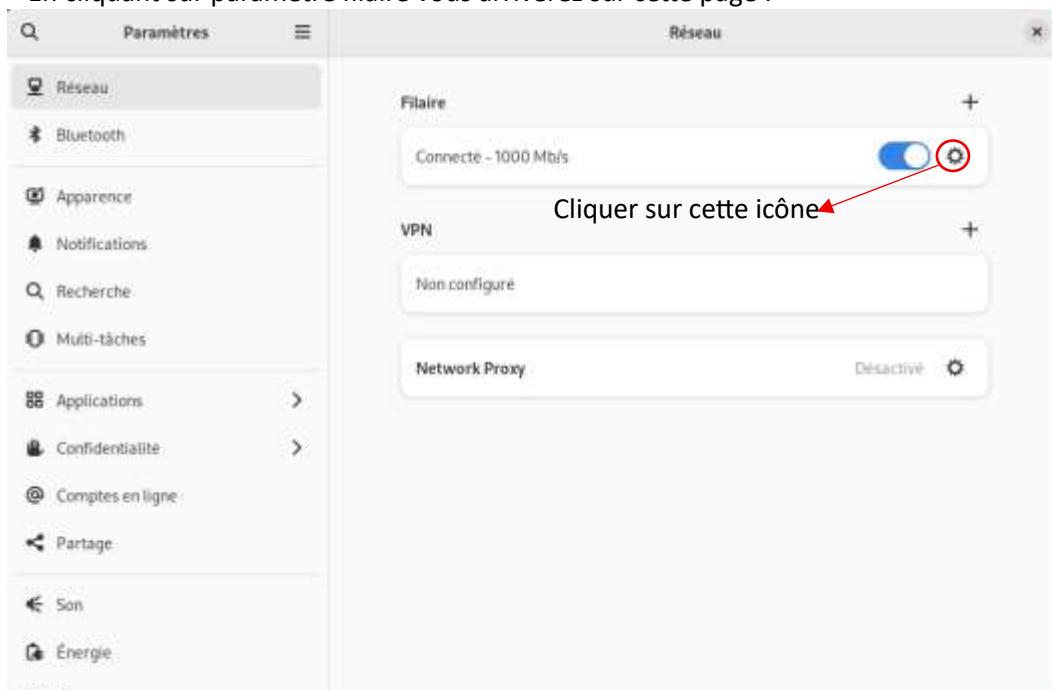
Configuration de la VM master :

Note : pour avoir la connexion internet sur votre master il faut que la VM routeurs reste allumer.

Pour commencer sur le master il faut mettre une adresse ip fixe sur le réseau intnet de la VM des routeurs, pour commencer il faut cliquer sur ce qui est entourée en rouge :



En cliquant sur paramètre filaire vous arriverez sur cette page :



En cliquant sur l'icône entouré en rouge vous arriverez sur cette page :

The screenshot shows a configuration window for a network connection profile. At the top, there are tabs: 'Annuler' (Cancel), 'Filaire' (Copper), and 'Appliquer' (Apply). Below these are five sub-tabs: 'Détails' (Details) which is selected, 'Identité' (Identity), 'IPv4', 'IPv6', and 'Sécurité' (Security). The main content area displays connection details: Vitesse de la connexion (1000 Mb/s), Adresse IPv4 (10.0.0.100), Adresse IPv6 (fe80::a00:27ff:fe0b:eb74), Adresse matérielle (08:00:27:0B:EB:74), and Route par défaut (10.0.0.1). There are three checked checkboxes: 'Connexion automatique' (Automatic connection), 'Rendre accessible aux autres utilisateurs' (Share this connection with other users), and 'Connexion avec quota : limite les données ou peut engendrer des frais' (Quota connection: limits data or may generate fees). A note below states: 'Les mises à jour logicielles et autres téléchargements importants ne seront pas démarrés automatiquement.' (Software updates and other important downloads will not be started automatically). At the bottom right is a red button labeled 'Supprimer le profil de la connexion' (Delete the connection profile).

Cliquer sur l'onglet IPv4 vous arriverez sur cette page :

The screenshot shows the 'IPv4' tab of the network connection configuration. At the top, there are tabs: 'Annuler' (Cancel), 'Filaire' (Copper), and 'Appliquer' (Apply). Below these are five sub-tabs: 'Détails' (Details), 'Identité' (Identity), 'IPv4' (selected), 'IPv6', and 'Sécurité'. The 'Méthode IPv4' section has four options: 'Automatique (DHCP)' (radio button unselected), 'Réseau local seulement' (radio button unselected), 'Manuel' (radio button selected), and 'Désactiver' (radio button unselected). The 'Adresses' section shows an IP configuration: Adresse (10.0.0.100), Masque de réseau (255.255.255.0), and Passerelle (10.0.0.1). The 'DNS' section has 'Automatique' (checkbox selected). The 'Routes' section shows a table with columns: 'Adresse', 'Masque de réseau', 'Passerelle', and 'Métrique'. The first row has entries: '10.0.0.100', '255.255.255.0', '10.0.0.1', and '1'. There is also a note: 'Séparer les adresses IP avec des virgules.' (Separate IP addresses with commas).

dans Méthode IPv4 cocher manuel et créer une adresse en mettant 10.0.0.100 en Adresse, 255.255.255.0 en Masque de réseau et 10.0.0.1 en Passerelle et appuyer sur appliquer de la

redémarrer votre master normalement si tout se passe bien vous serez connecter à internet.

maintenant nous allons faire des modification avec le terminal vous accéder de la même manière que pour la VM gérant les routeurs, si sudo ne marche pas en su pour faire les manip tout d'abords nous allons vérifier si mariadb est installer faire la commande « mariadb --version », si cela est afficher :

```
master@vbox:~$ mariadb --version
mariadb Ver 15.1 Distrib 10.11.14-MariaDB, for debian-linux-gnu (x86_64) using
EditLine wrapper
```

ça veut dire que il est installé sinon il faut faire la commande « apt update » pour mettre à jour puis la commande « apt install mariadb-server -y » après avoir installé mariadb il faut faire cette commande « mysql_secure_installation » :

```
root@vbox:/home/master# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!
```

```
In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.
```

Pour unix_socket, il faut mettre oui (y) :

```
Switch to unix_socket authentication [Y/n] Y
Enabled successfully!
Reloading privilege tables..
... Success!
```

Pour change the root password, il faut mettre oui (Y) :

```
Change the root password? [Y/n] Y
New password:
dans New password il faut mettre « toto ».
pour Re-enter new password, il faut mettre « toto »
si sa affiche ça c'est bon le mot de passe a été valider :
Password updated successfully!
Reloading privilege tables..
... Success!
```

Pour remove anonymous users, il faut mettre oui (Y) :

```
Remove anonymous users? [Y/n] Y
... Success!
```

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Pour Disallow root login remotely, il faut mettre oui (Y) :

```
Disallow root login remotely? [Y/n] Y
... Success!
```

Pour Remove test database and acces to it, il faut mettre oui (Y) :

```
Remove test database and access to it? [Y/n] Y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!
```

Pour Reload privilege tables now, il faut mettre oui (Y) :

```
Reload privilege tables now? [Y/n] Y
... Success!
```

Cleaning up...

```
All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.
```

```
Thanks for using MariaDB!
```

Après avoir fait tout ça, pour lancer mariadb faite cette commande « mariadb », sa devrait afficher ça sur votre terminal :

```
root@vbox:/home/master# mariadb
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.11.14-MariaDB-0+deb12u2 Debian 12
```

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
MariaDB [(none)]> █
```

pour commencer nous allons créer la base de données Routagedb avec cette commande « CREATE DATABASE Routagedb; », pour vérifier si elle a été créer faire cette commande « show databases » :

```
MariaDB [(none)]> show databases;
```

```
+-----+
| Database      |
+-----+
| Routagedb    |
| information_schema |
| mysql         |
| performance_schema |
| sys           |
+-----+
5 rows in set (0,003 sec)
```

ensuite pour accéder a la base de données faire cette commande « use Routagedb », si vous êtes connecter a la bonne base de données sa devrait afficher ça :

```
MariaDB [Routagedb]>
```

vous êtes sur la bonne base donnés, pour crée la table je vous conseille d'aller sur mon github :

The screenshot shows a GitHub repository page. At the top, it says "Warsoul68 / SAE-3.02_Mathias_Schumacher_new". Below the header, there are tabs for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The Code tab is selected. In the main area, there is a list of commits:

| Author | Commit Message | Date |
|-----------|-------------------------------------------|--------------|
| Warsoul68 | Mise à jour du planning de développement | 10 hours ago |
| Warsoul68 | Ajout de la documentation et du schéma | 5 days ago |
| Warsoul68 | Méth dans le code du Master | last week |
| Warsoul68 | Ajout du git pour le SAE | 2 weeks ago |
| Warsoul68 | ajout du gitignore | last week |
| Warsoul68 | mise à jour du code de la base de données | 2 weeks ago |
| Warsoul68 | Modification du readme | 8 hours ago |

On the right side of the page, there are sections for About, Releases, and Packages.

En cliquant sur BDD_pour_la_sae_3-02.sql, vous arriverez sur cette page :

The screenshot shows a GitHub file page for "BDD_pour_la_sae_3-02.sql". The file content is as follows:

```
1 CREATE TABLE TableRoulage(
2     id INT AUTO_INCREMENT PRIMARY KEY,
3     tp VARCHAR(50),
4     port INT,
5     cte TEXT
6 );
```

To the right of the file content, there is a text box with the following instructions:

Copier-coller le texte et coller
le terminal ou vous êtes
connectée à mariadb

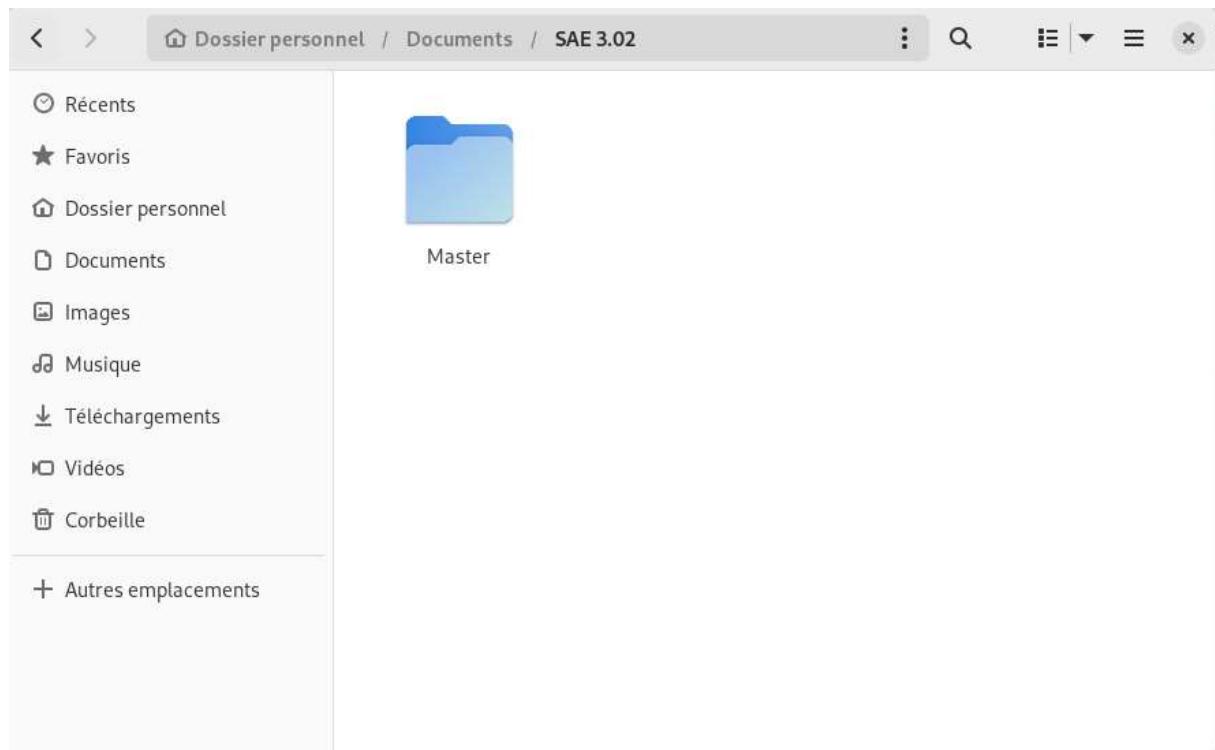
Pour vérifier si vous avez bien crée la table faite cette commande « show TABLES », si sa affiche ça ces

bon la table est créée :

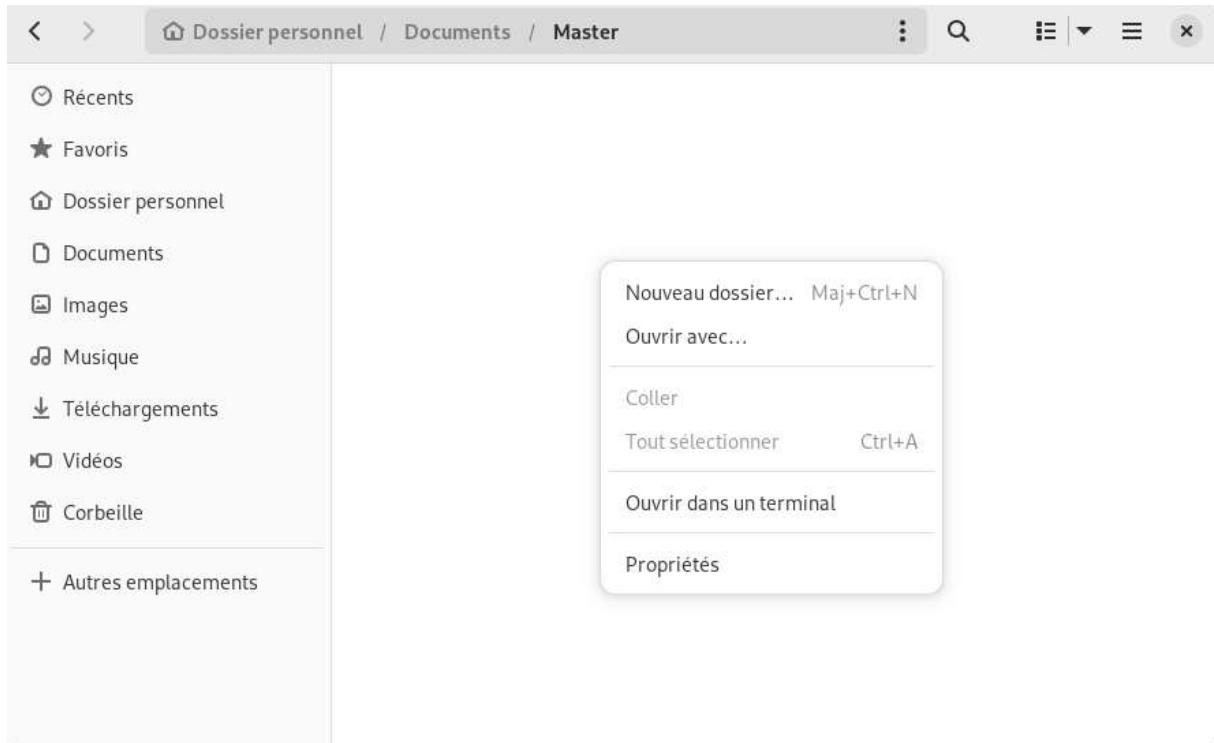
```
MariaDB [Routagedb]> show TABLES;
+-----+
| Tables_in_Routagedb |
+-----+
| TableRoutage        |
+-----+
1 row in set (0,002 sec)
```

Maintenant que la base de données et la table est créé nous allons passer à la partie la commande pour vérifier si python est installer et la commande pour l'installer sont les même que pour la VM qui gère les routeurs.

maintenant pour accéder au fichier c'est la même manip que pour la VM qui gère les routeurs : mais dans document vous devez créer un dossier master comme ci-dessous :



maintenant dans le master pour pouvoir installer mysql.connector et PyQt5, il faut créer un environnement virtuelle car l'une bibliothèque mysql.connector n'est pas valider par debian 12, pour commencer vous devez vous placer dans le master et faire clique droit :



vous devez cliquer ouvrir dans un terminal.

dans le terminal vous avez besoin de venv pour vérifier si il est installé fait cette commande « python3 -m venv –help », si ça affiche ça veut dire que il est installé :

```
root@vbox:/home/master/Documents/Master# python3 -m venv --help
usage: venv [-h] [--system-site-packages] [--symlinks | --copies] [--clear]
             [--upgrade] [--without-pip] [--prompt PROMPT] [--upgrade-deps]
             ENV_DIR [ENV_DIR ...]
```

Creates virtual Python environments in one or more target directories.

positional arguments:

```
  ENV_DIR           A directory to create the environment in.
```

sinon faire la commande « apt install python3-venv -y ».

quand venv est installé pour créer l'environnement virtuel il faut faire cette commande « python3 -m venv venv », pour se connecter à l'environnement virtuel il faut faire cette commande « source venv/bin/activate », si vous voyez venv devant votre répertoire ça vous y êtes cependant pour éviter les problèmes d'interface PyQt5 il faut créer le venv en n'étant pas root :

```
master@vbox:~/Documents/Master$ source venv/bin/activate
(venv) master@vbox:~/Documents/Master$
```

Après cela il faut installer la bibliothèque mysql.connector avec cette commande « pip install mysql-connector-python » et PyQt5 « pip install PyQt5 ».

après tout ce qu'il faut est installé, il faut maintenant récupérer sur mon git, les scripts pour le master sur cette page de mon git :

After clicking on the 'Script' folder, you will arrive at this page:

après avoir cliqué sur script vous arriverez sur cette page :

Ensuite vous devez télécharger les différents scripts et les coller dans le dossier Master que vous avez créé.

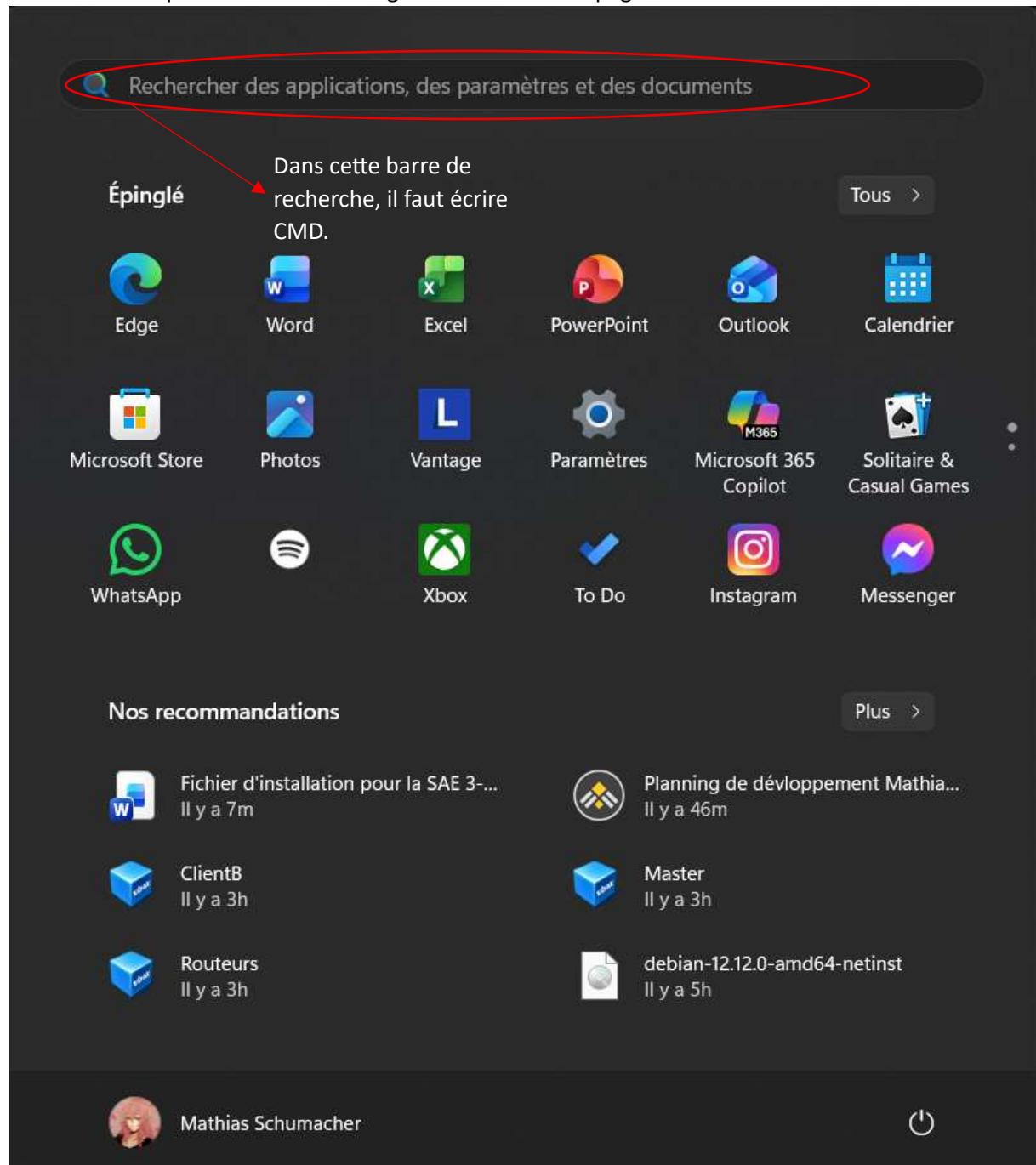
Configuration du pc physique sous Windows et de la VM Windows 10 :

pour commencer il faut installer python pour le pc physique et la VM Windows 10 pour accéder au terminal Windows :

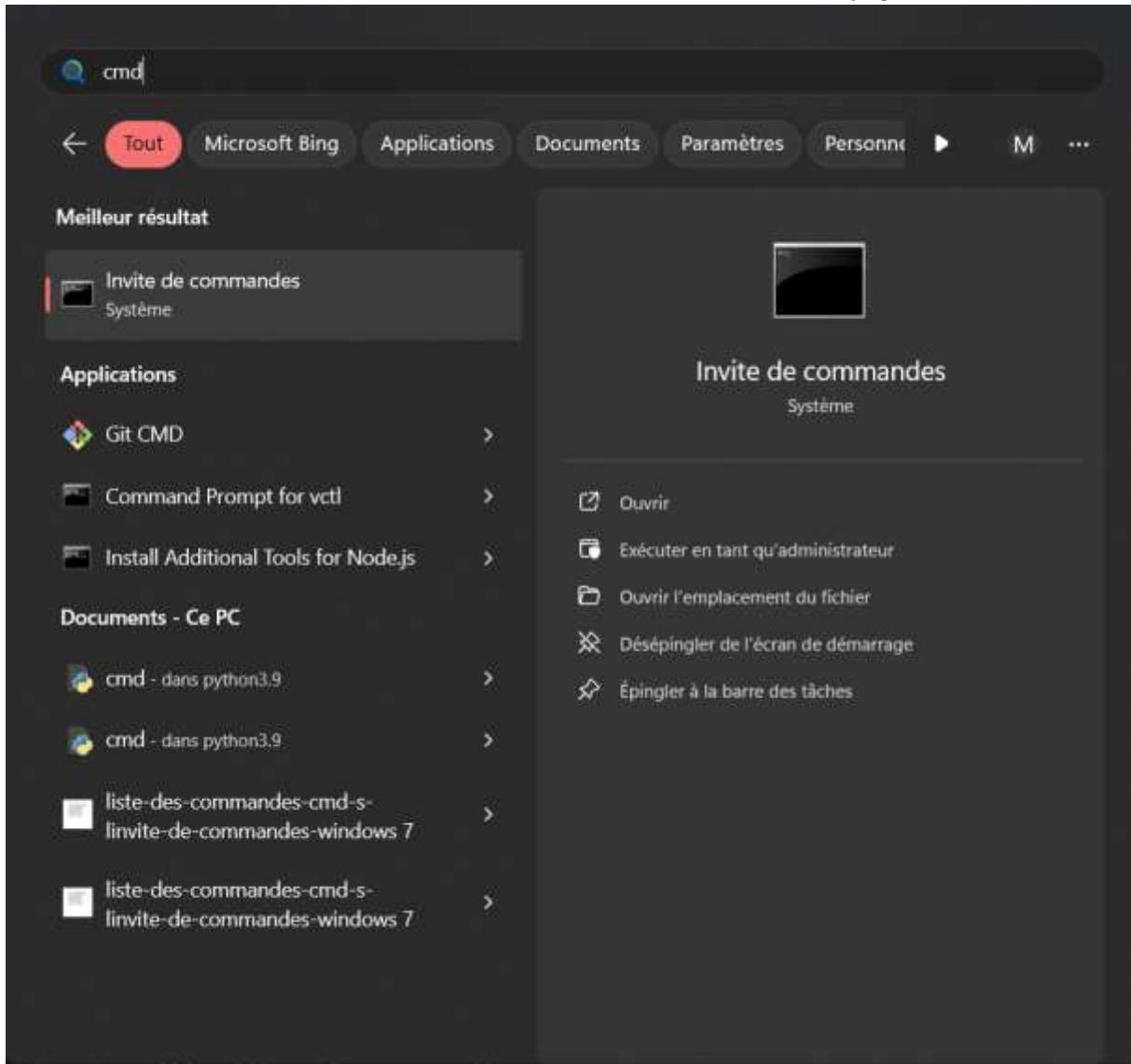


Il faut cliquer sur cette icône

Quand vous cliquer sur l'icône en rouge sa va ouvrir cette page :



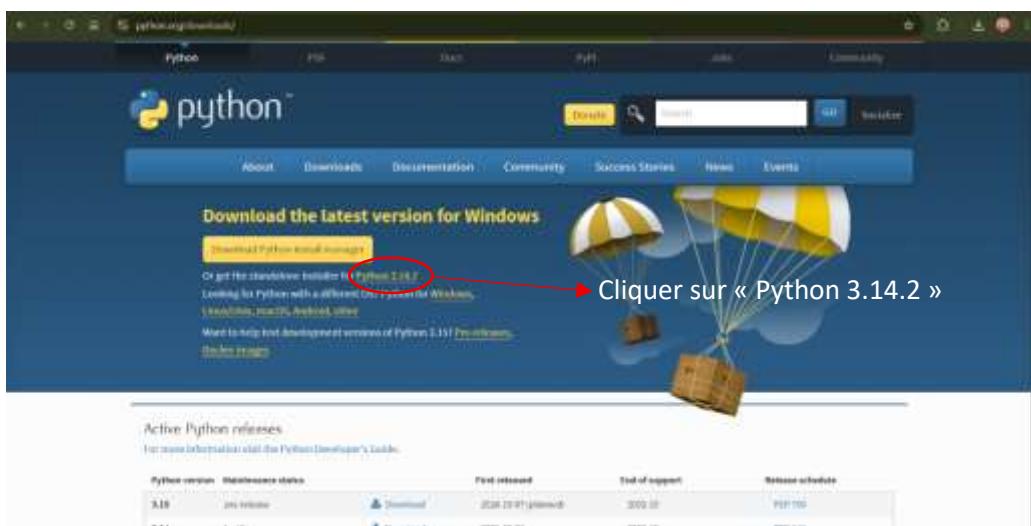
Quand vous aurez noté cmd sur la barre de recherche vous arriverez à cette page :



là il faut cliquer sur inviter de commande et faire la commande « py --version » pour voir s'il est installé, si sa affiche ça ces bon :

```
C:\Users\Mathias Schumacher>py --version
Python 3.12.1
```

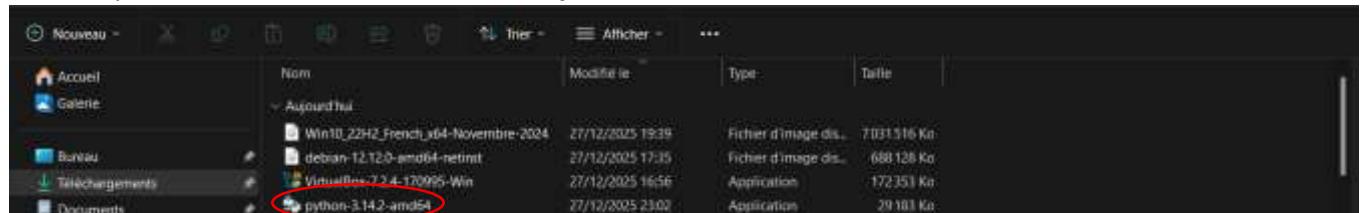
sinon il faut l'installer, je vous invite à vous servir du lien que je fournis pour installer python dans le README partie 3, vous arriver à cette page :



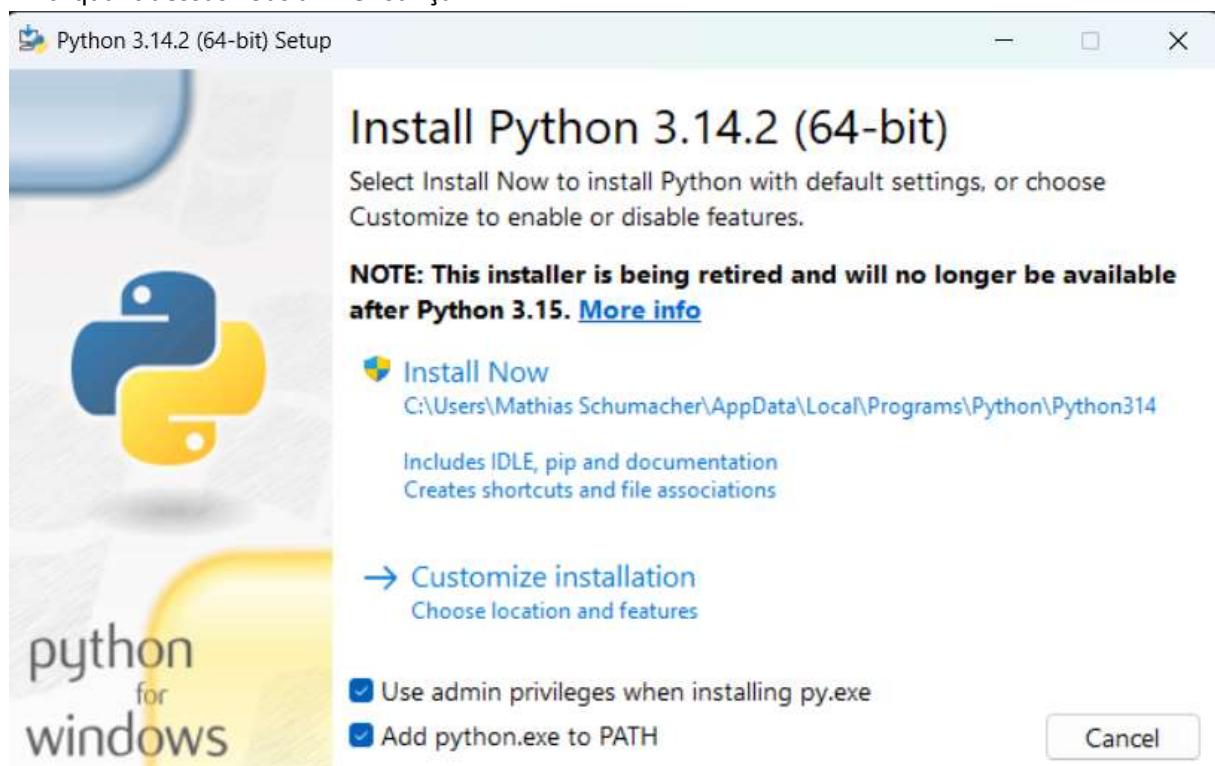
Quand vous serez entrain de l'installer vous devriez voir ça, s'installer :



Dans l'explorateur de fichier vous devriez voir ça :



En cliquant dessus vous arrivez sur ça :

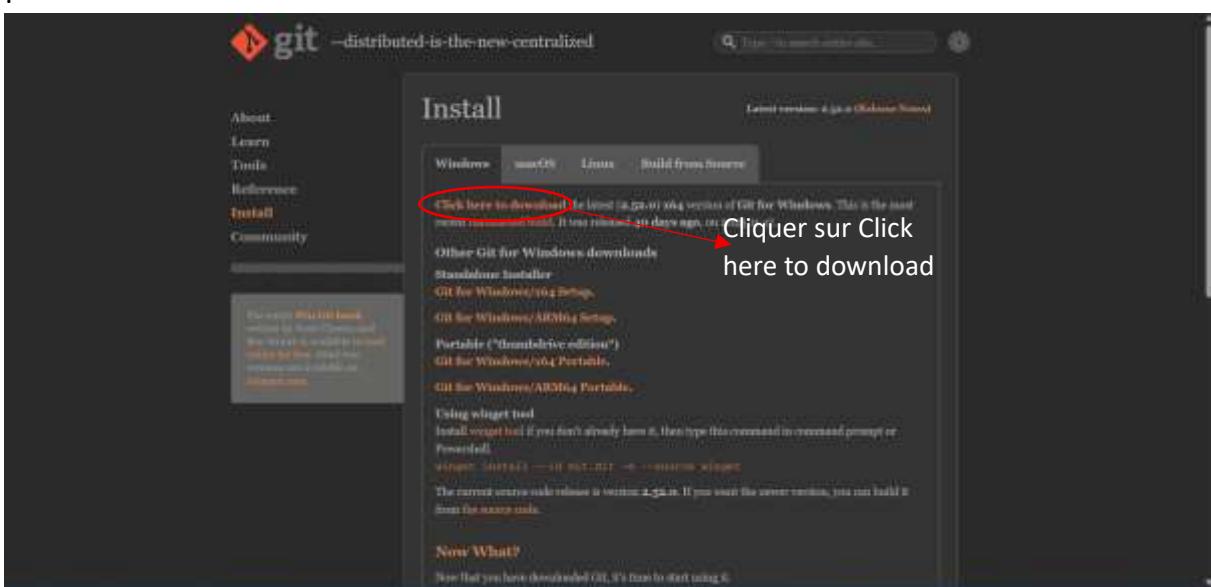


Je vous conseille de cocher « Use admin privileges when installing py.exe » et « Add ptyhon.exe to PATH », cliquer sur Install Now après il installera sans problème.

Maintenant passons a git comme pour python vous pouvez faire la commande « git --version », si sa vous affiche ça, ces bon git est installer :

```
C:\Users\Mathias Schumacher>git --version
git version 2.46.2.windows.1
```

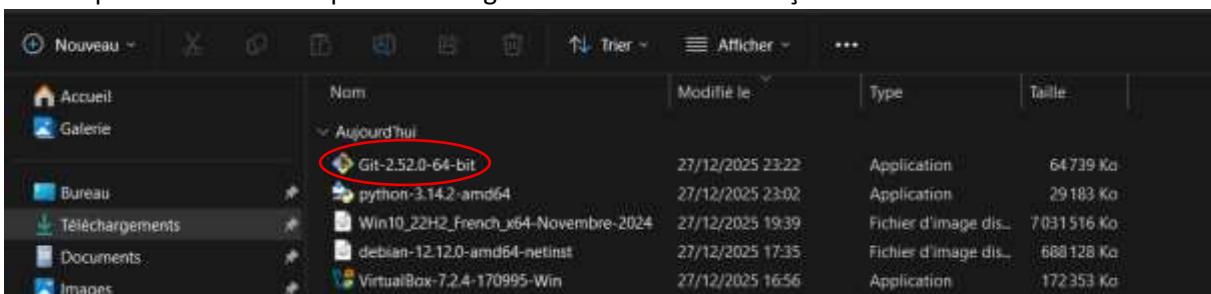
sinon utilise le lien que je fournis dans la partie 3 de mon README, vous devrez arriver sur cette page :



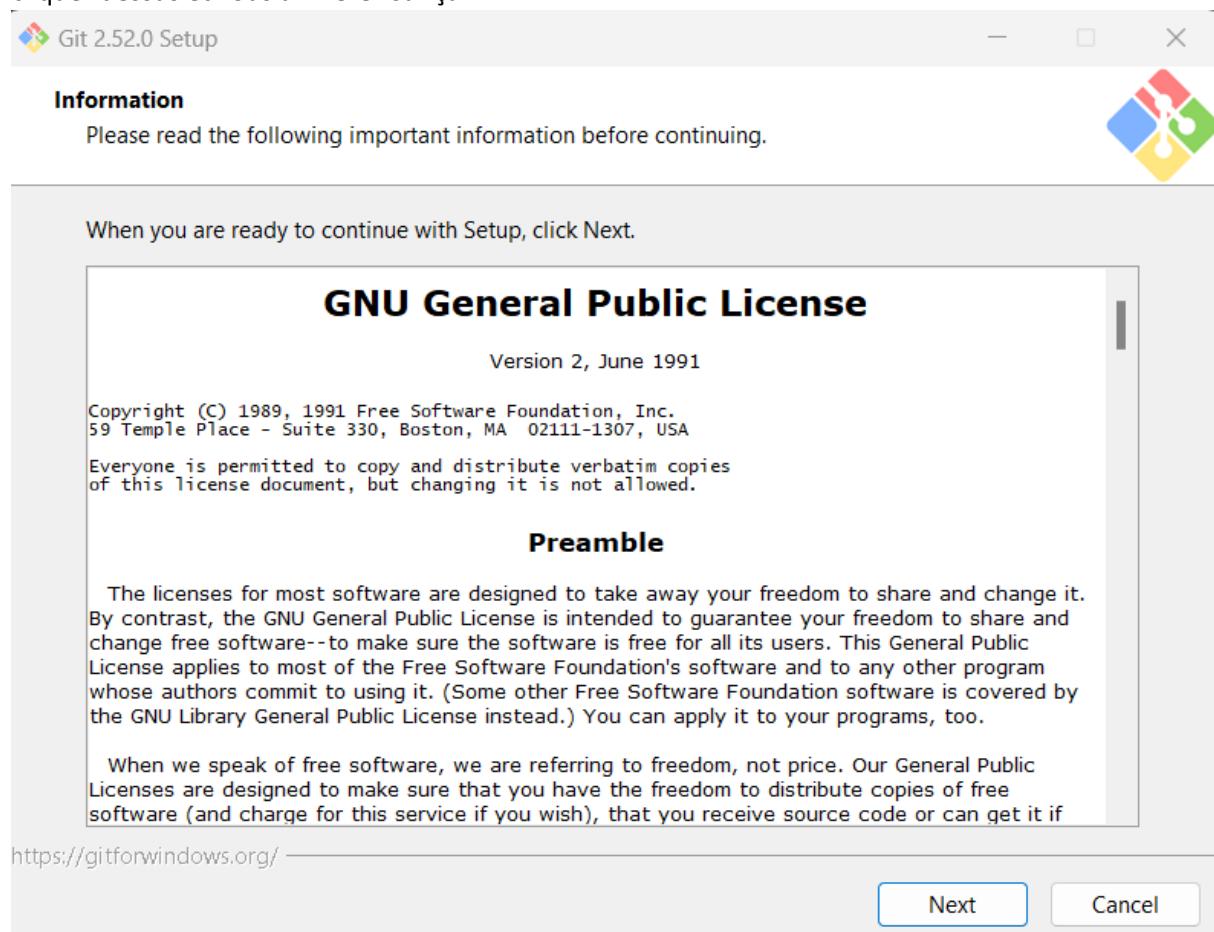
après avoir cliquer sur ça vous devriez voir ça dans vos téléchargements :



dans explorateur de fichier puis téléchargement vous devriez voir ça :

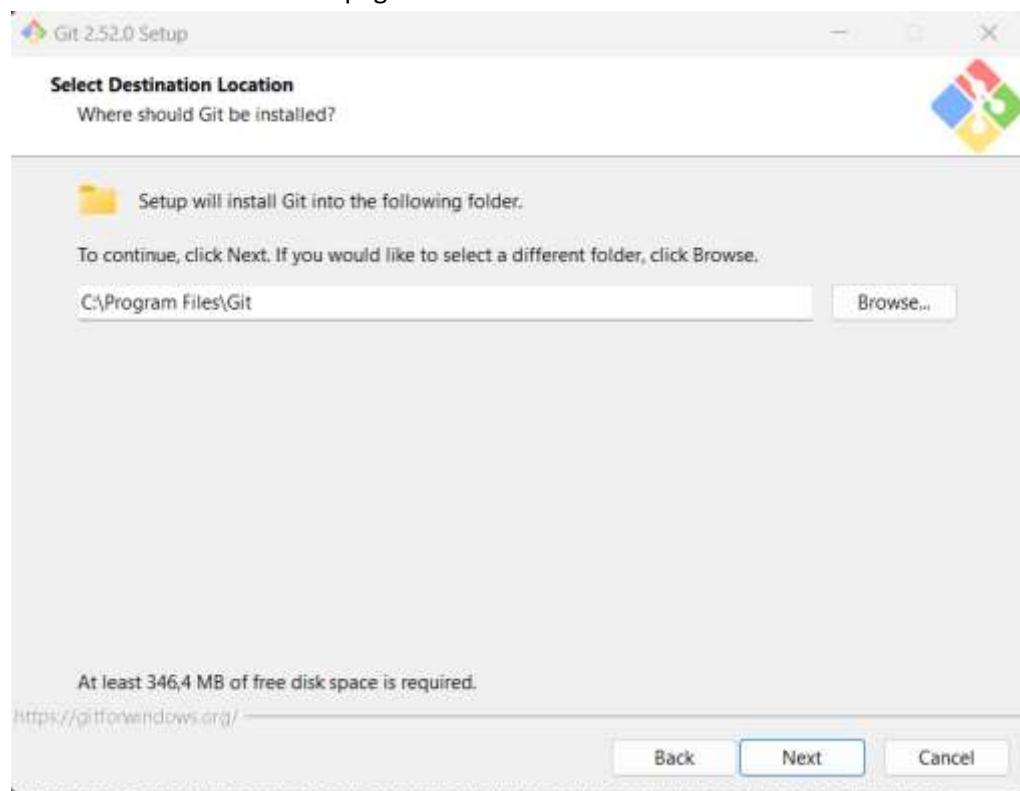


cliquer dessus et vous arriverez sur ça :



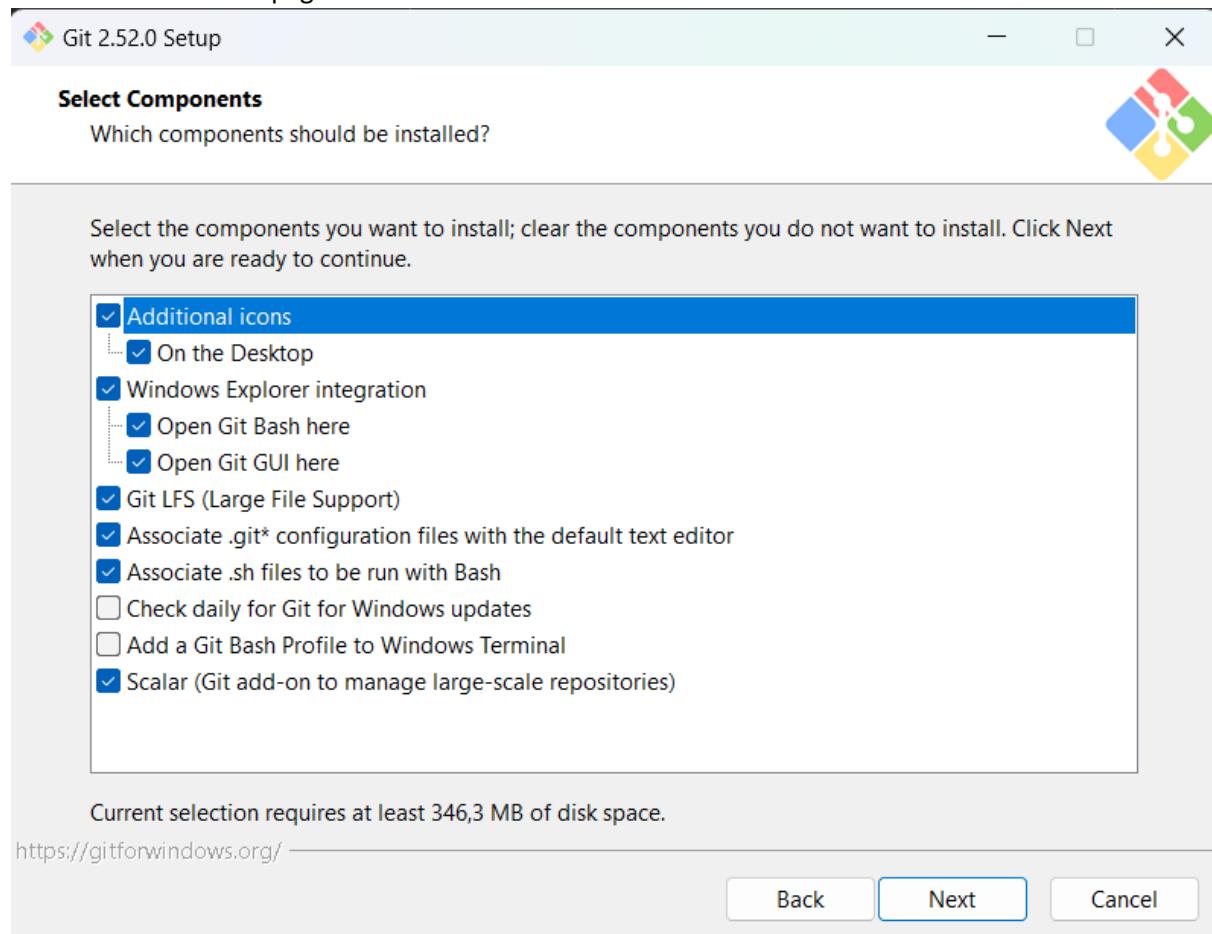
Cliquez sur Next.

Vous arrivez ensuite à cette page :



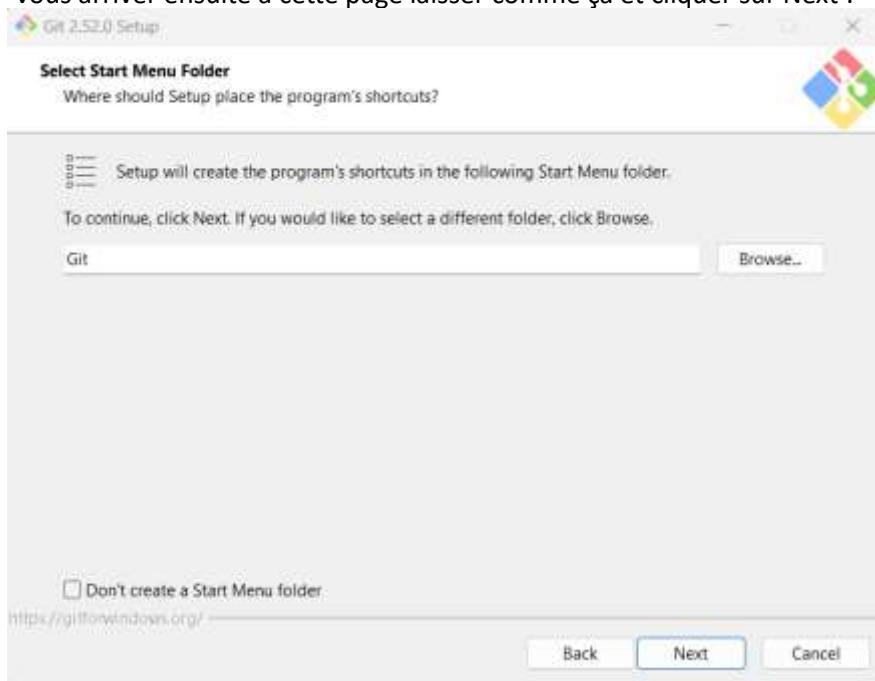
Cliquer sur Next.

vous arrivez sur cette page :

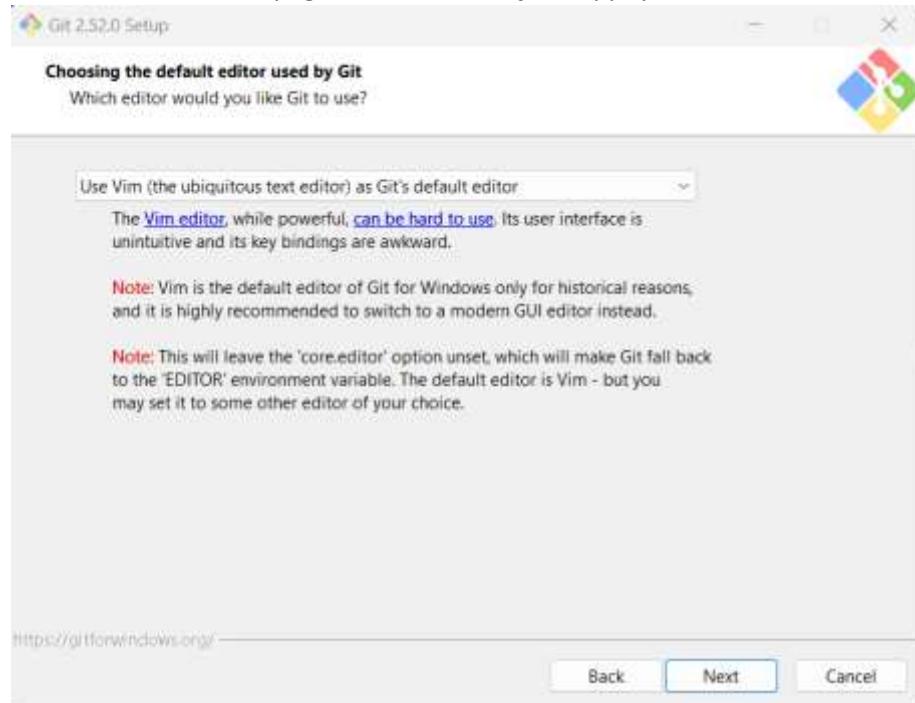


Cocher comme moi et appuyer sur Next.

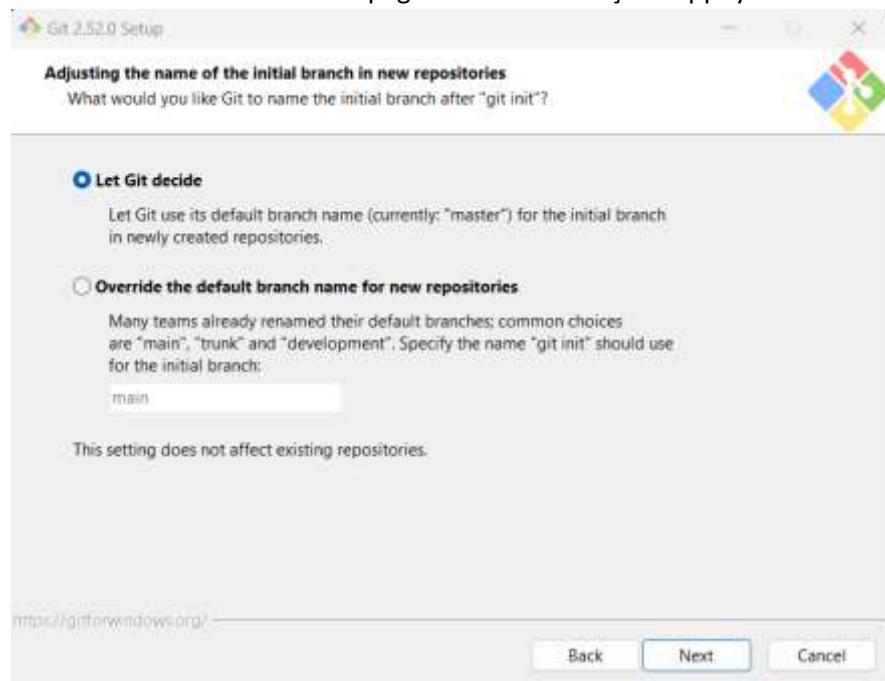
vous arriver ensuite a cette page laisser comme ça et cliquer sur Next :



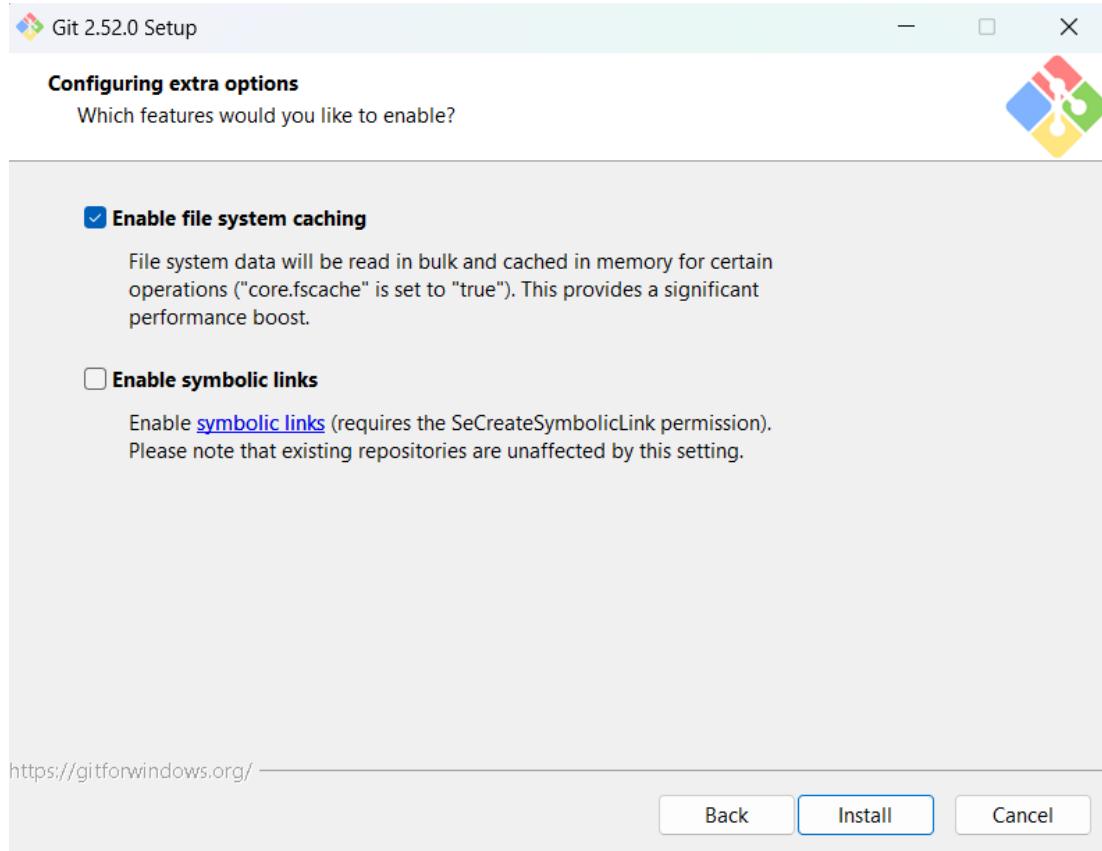
Vous arrivez sur cette page laissez comme ça et appuyer sur Next :



Vous arrivez ensuite sur cette page laissez comme ça et appuyer sur Next :



Pour la suite continuer à appuyer sur Next jusqu'à arriver à cette page :



puis cliquer sur Install.

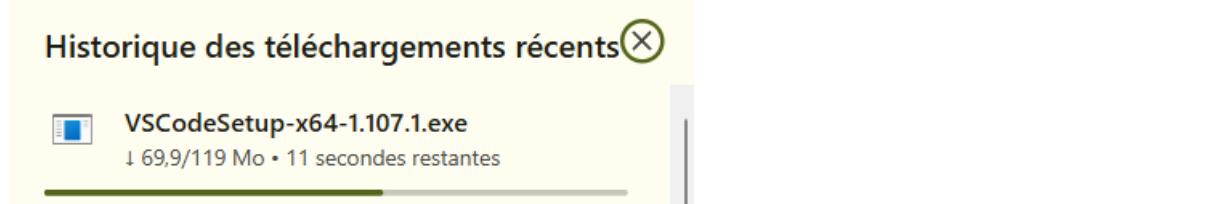
Maintenant nous allons faire dans un dossier que vous pouvez retrouver la commande « git clone https://github.com/Warsoul68/SAE-3.02_Mathias_Schumacher_new.git »
après cela vous avez entièrement récupérer mon projet GitHub.

Maintenant nous allons passer à l'installation de Visual Studio code, tout d'abords pour installer ce dernier, il faut utiliser le lien d'installation Visual Studio que je fournis dans mon README a la partie 3, en utilisant mon lien vous arriverez sur cette page :

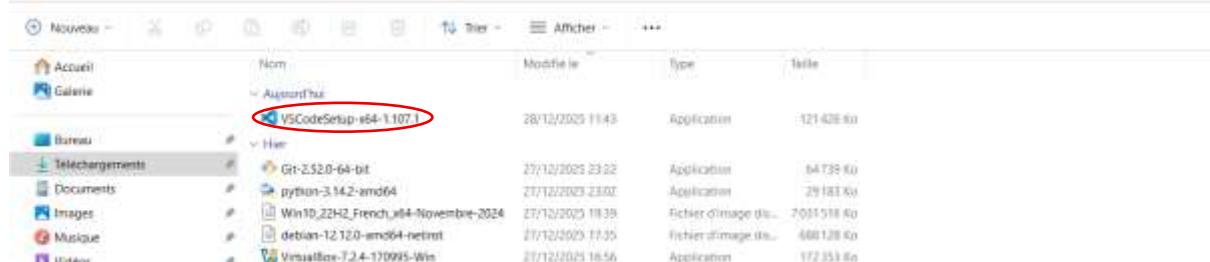
The screenshot shows the 'Download Visual Studio Code' page. It features icons for Windows, Linux, and Mac operating systems. Below each icon are download links for various package managers and formats (e.g., .deb, .rpm, .tar.gz, .zip). A red arrow points to the 'Windows x64' link under the Windows icon. At the bottom, there is a note about accepting terms and conditions.

Cliquer sur « x64 » la version System

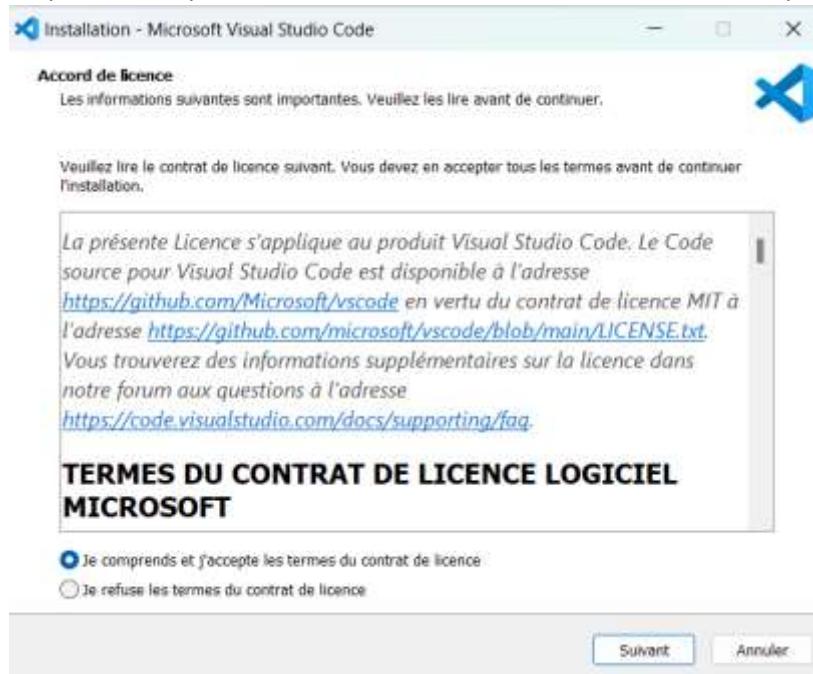
En cliquant dessus vous devriez voir ça sur vos téléchargements sur votre navigateur :



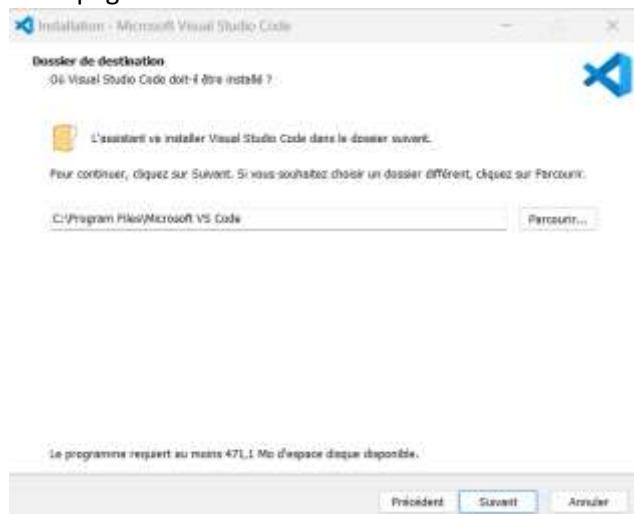
Dans explorateur de fichier, dans téléchargement vous devriez voir ça :



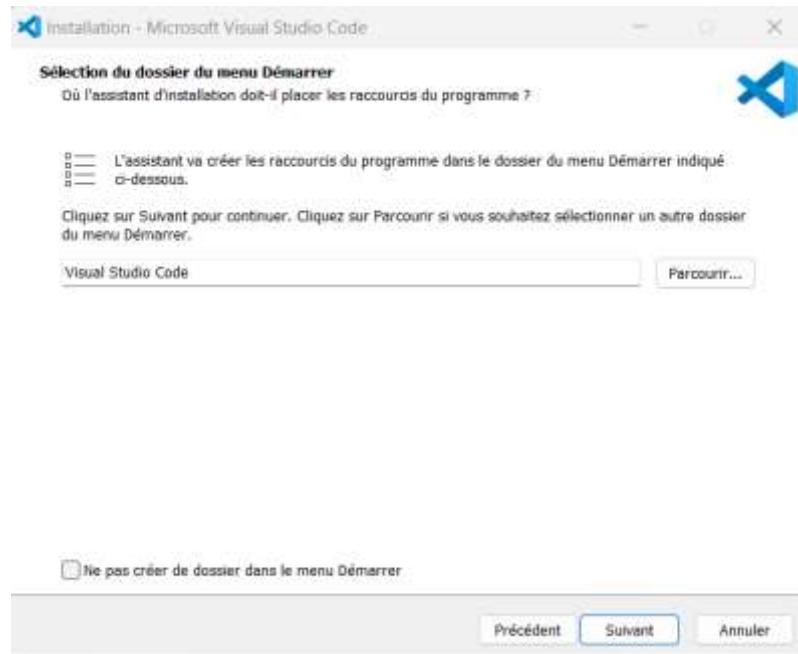
Cliquer dessus pour lancer l'installateur, vous aller arrivez sur cette page :



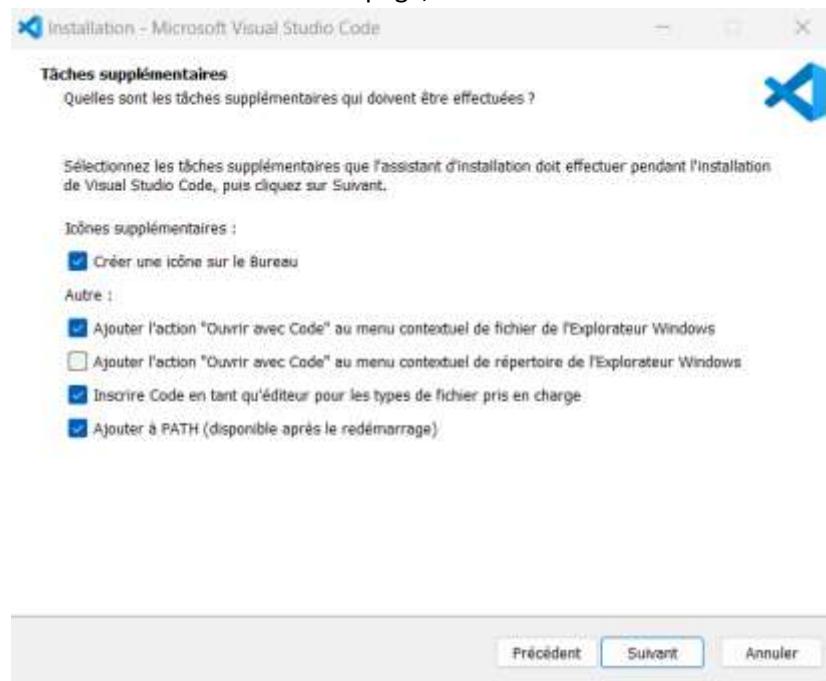
Cocher « Je comprends et j'accepte les termes du contrat de licence » puis suivant, vous arriverez sur cette page :



Laissez par défaut et cliquer sur suivant, vous arrivez sur cette page, vous laissez par défaut et cliquer sur suivant :



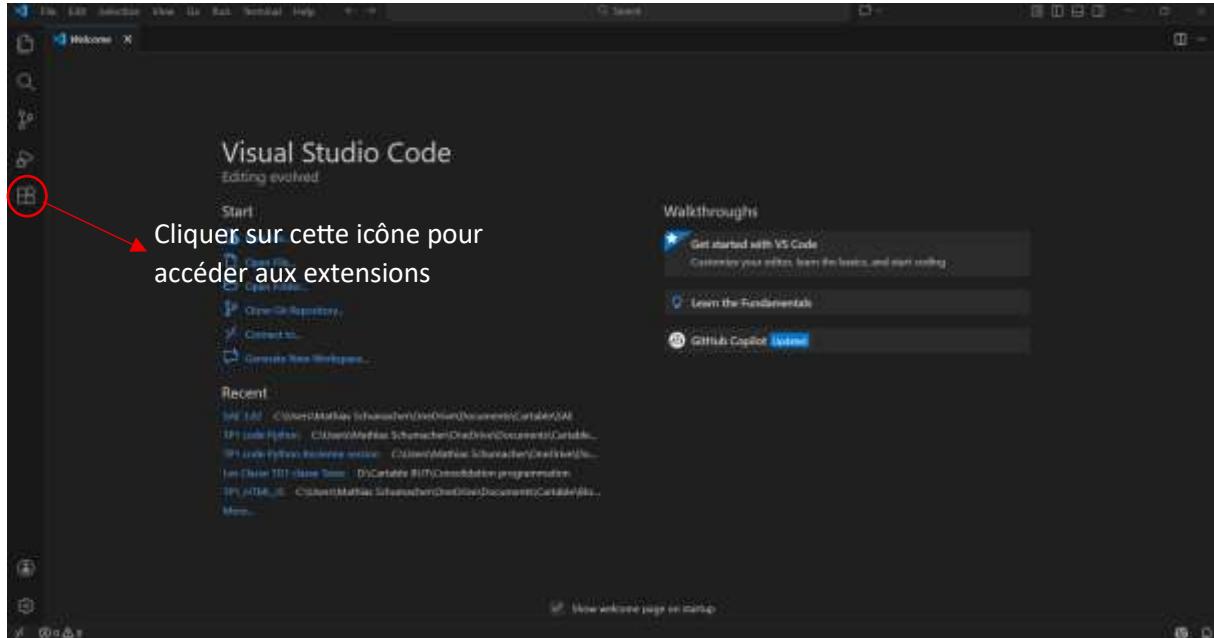
vous arrivez ensuite sur cette page, la-vous cocher comme moi et vous appuyer sur suivant :



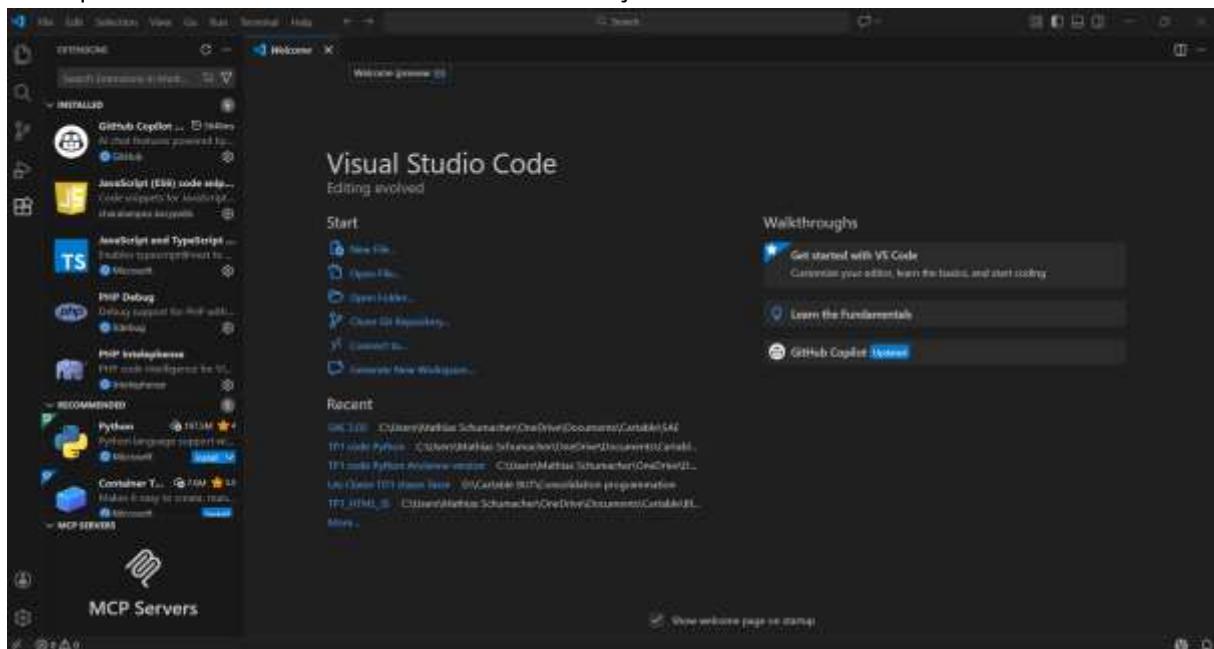
vous arrivez sur cette page, cliquer sur installer :



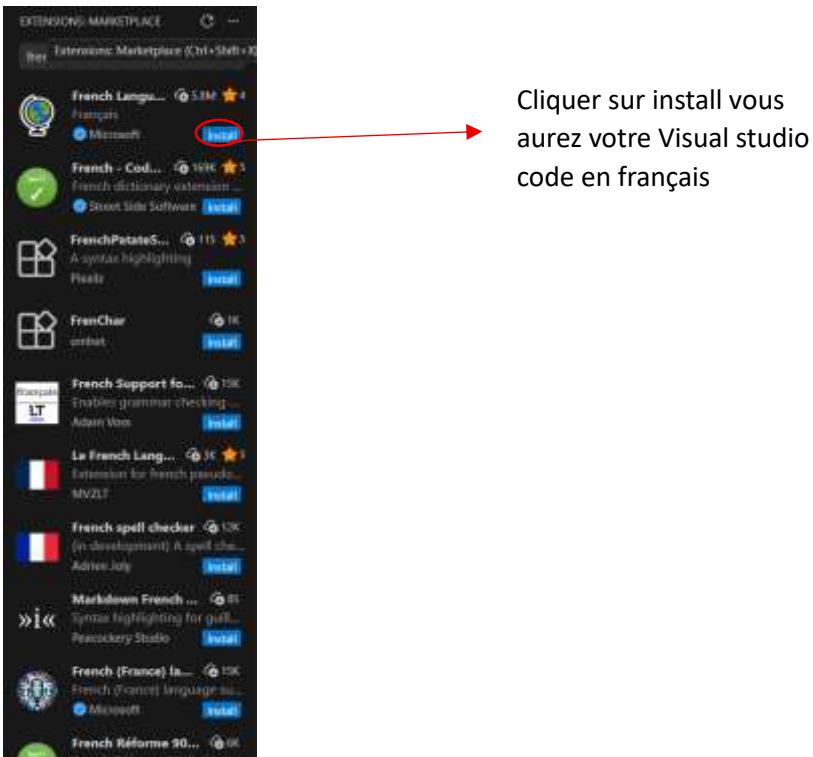
Il y a une dernière chose à voir quand vous arrivez sur Visual studio code à cette page :



en cliquant sur cette icône entourée vous allez voir ça :

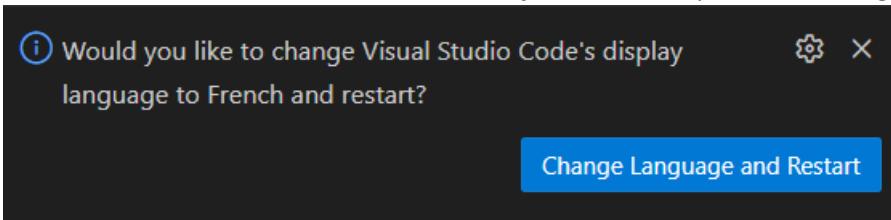


dans la barre de recherche écrivez « french », vous allez voir ça :

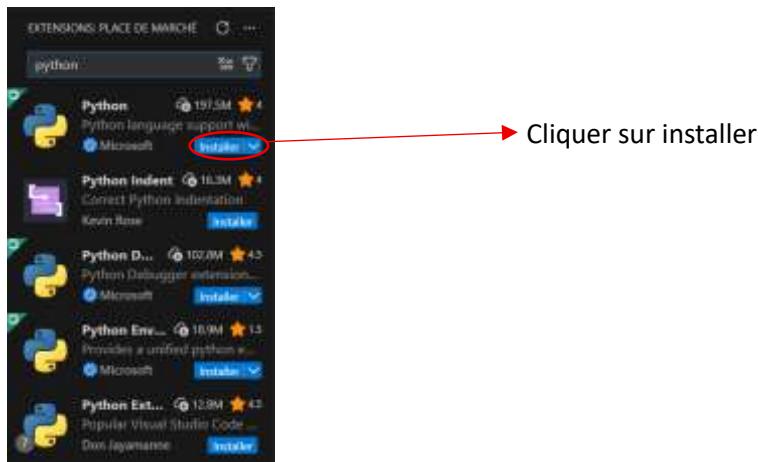


Cliquer sur install vous aurez votre Visual studio code en français

Quand l'extension sera installée il affichera ça, il faudra cliquer sur « Change Language and Restart » :



L'autre extension à installer est Python pour la trouver comme l'extension pour le français il faut marquer dans la barre de recherche « python » et vous devriez voir ça :



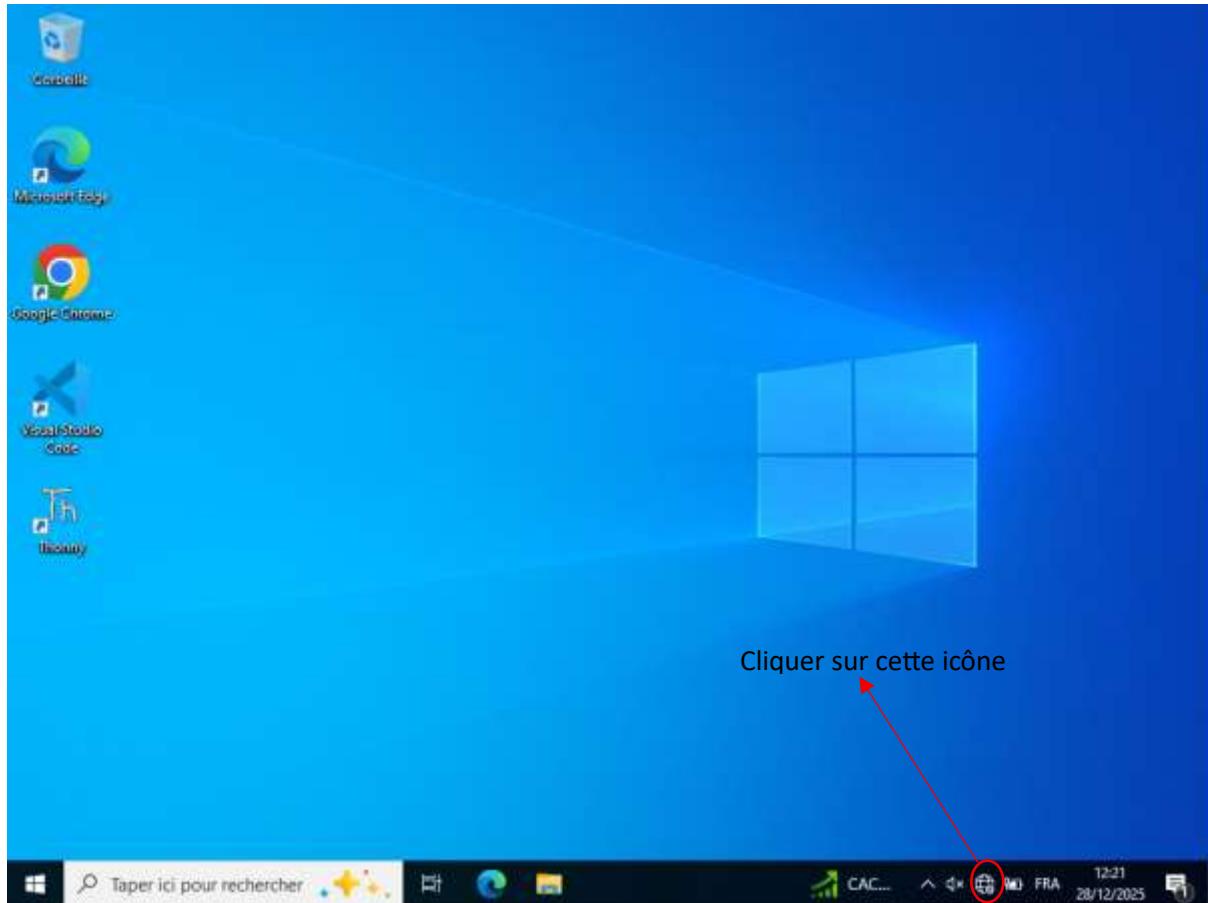
Maintenant vous avez toutes les extensions que vous avez besoin pour Visual studio code, ensuite pour installer PyQt5 pour le script client depuis le terminal de votre Visual studio code il faut faire la commande « pip install PyQt5 », je précise que toute ces installation sur votre pc physique (si vous

utilisez un Windows) sont les même pour votre VM Windows 10.

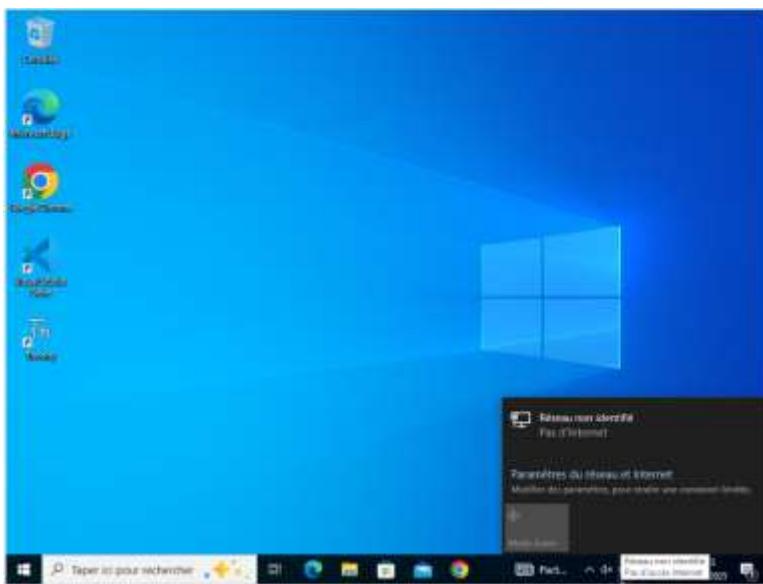
Maintenant nous allons voir le config propre à la VM Windows 10.

Note : pour que internet fonctionne sur votre VM Windows 10, il faut que la VM gérant le routeur soit allumer.

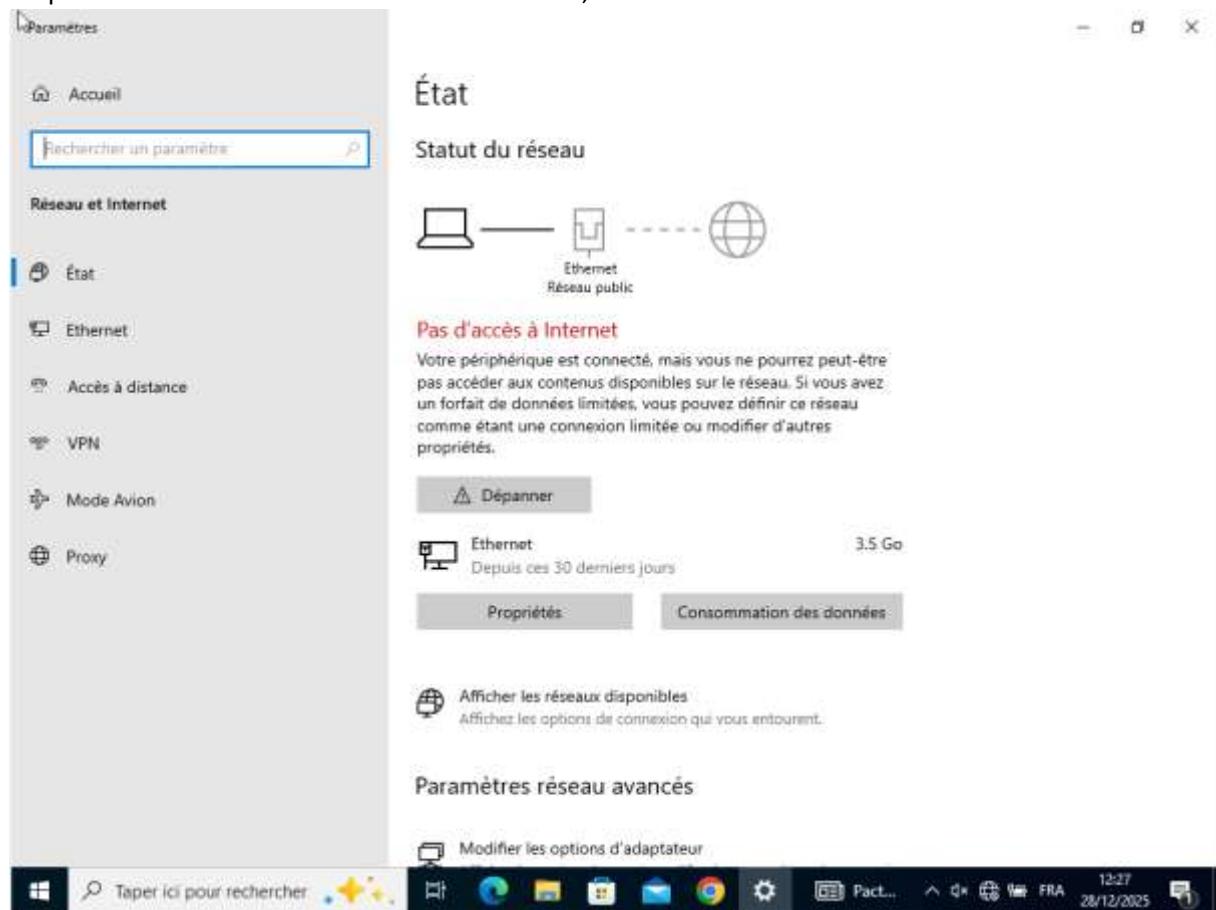
pour commencer, il va falloir attribuer à votre VM Windows 10 en intnet, une adresse ip fixe sur le réseau du routeur, quand vous serez sur votre bureau de votre VM Windows 10 :



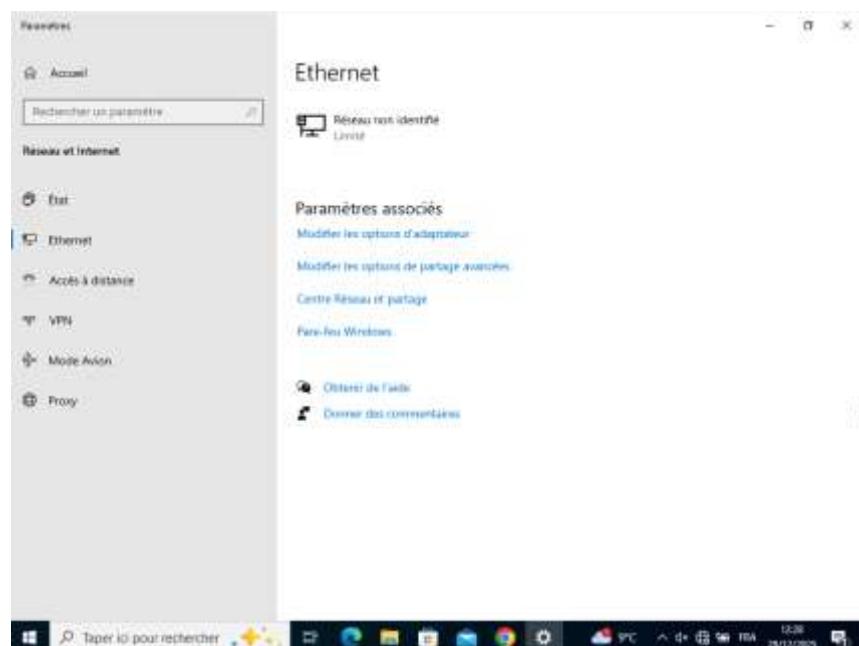
En cliquant sur l'icône entourée en rouge, ça va ouvrir cet onglet :



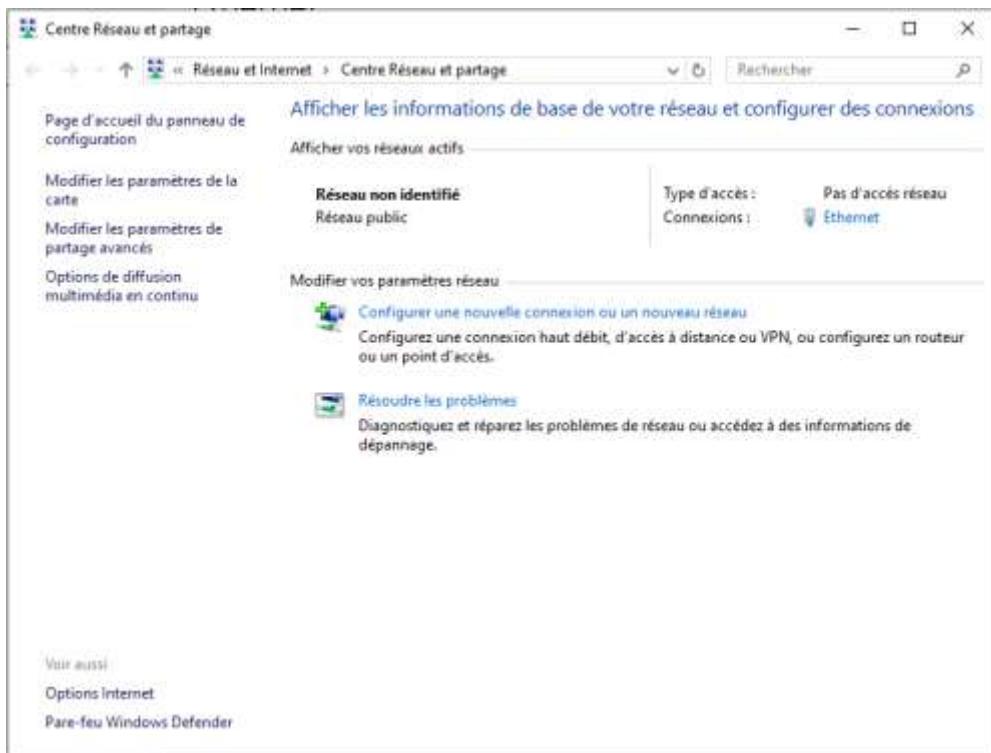
Cliquer sur « Paramètres du réseau et internet », sa va ouvrir cette fenêtre :



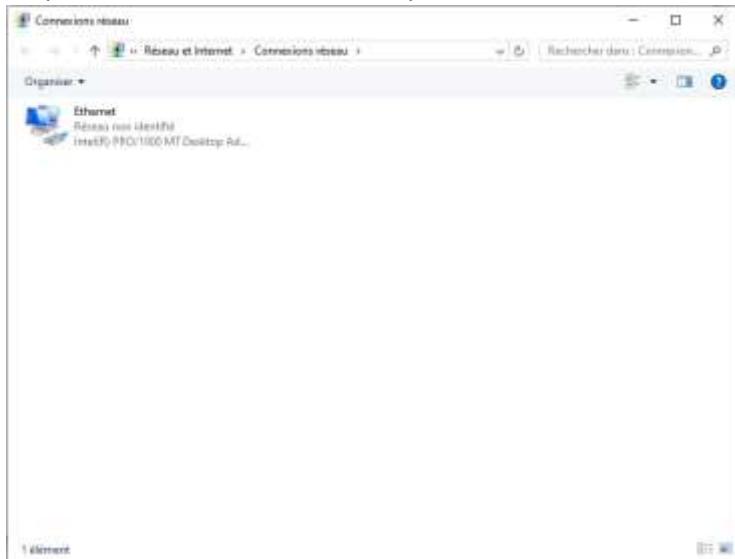
Cliquer ensuite sur « Ethernet », vous arriverez sur cette page :



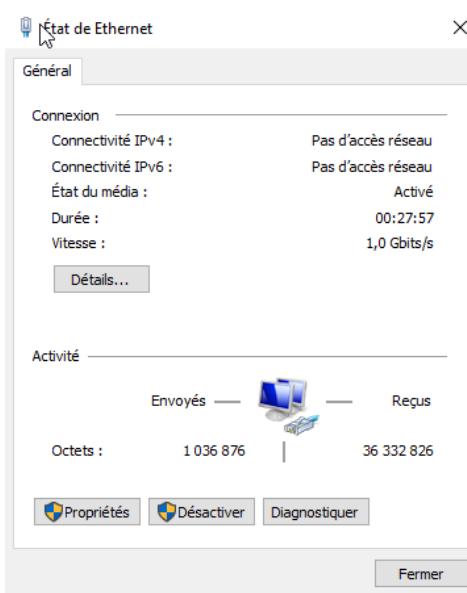
Cliquer ensuite sur « Centre Réseau et partage », ça va vous ouvrir cette page :



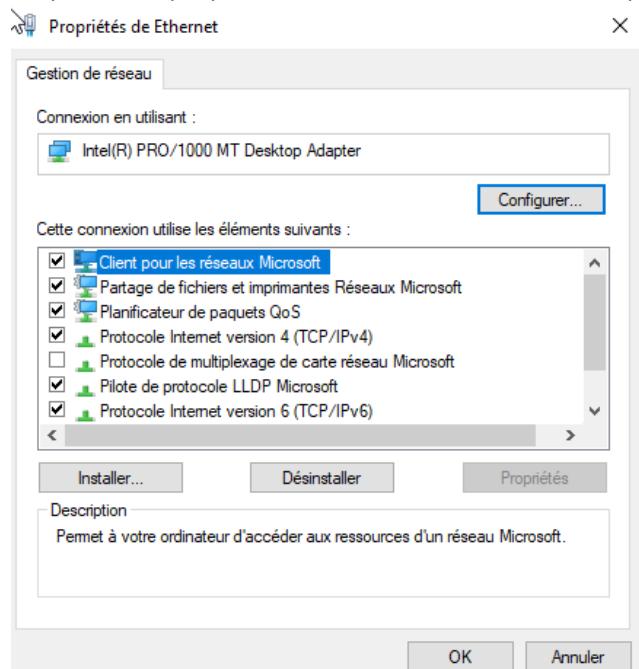
Cliquer ensuite sur « Modifier les paramètres de la carte », vous allez arriver sur cette page :



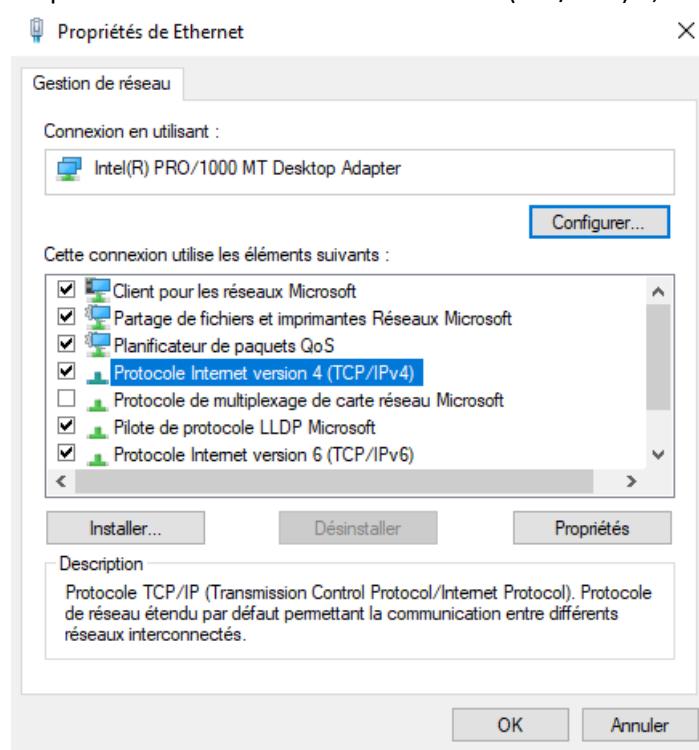
vous devez double cliquer sur « Ethernet » vous arriverez à cette page :



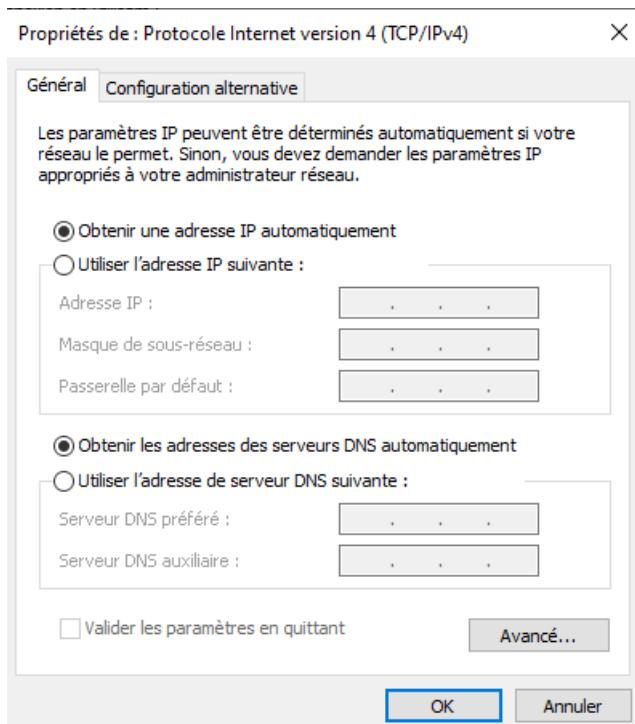
Cliquer sur « propriété » vous allez arriver sur cette page :



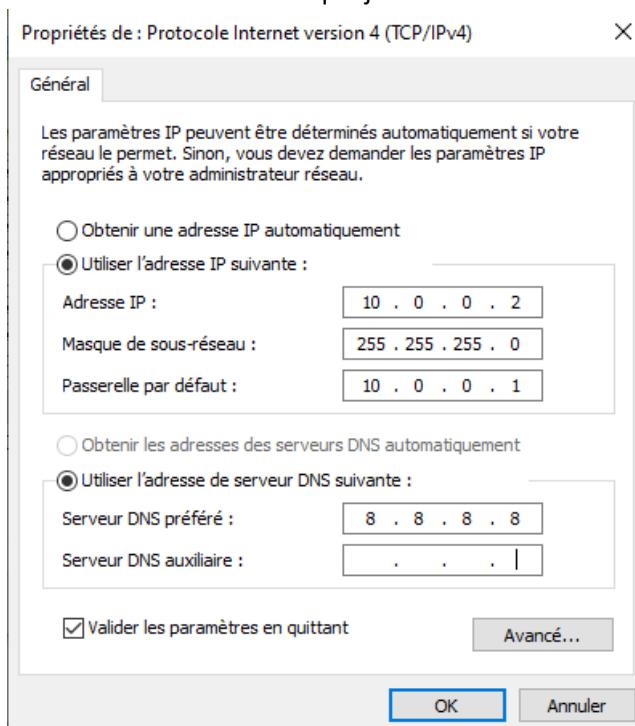
Cliquer sur « Protocole Internet version 4 (TCP/IPv4) », vous devriez voir ça :



Cliquez sur « Propriétés », vous allez arriver à cette page :



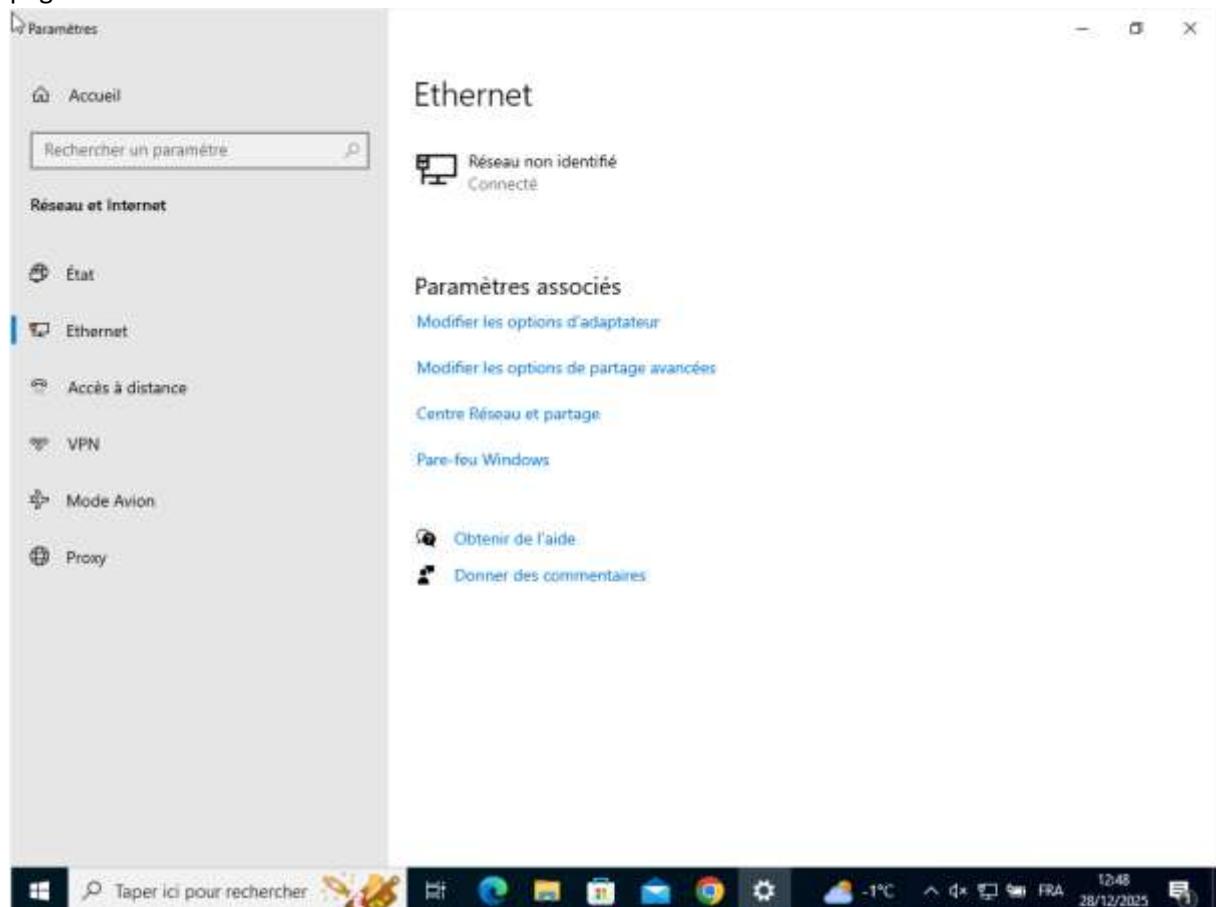
la cocher « Utiliser l'adresse IP suivante » et « Utiliser l'adresse de serveur DNS suivante », mettez-les information comme celle que j'ai mis dans les différent champs de saisie :



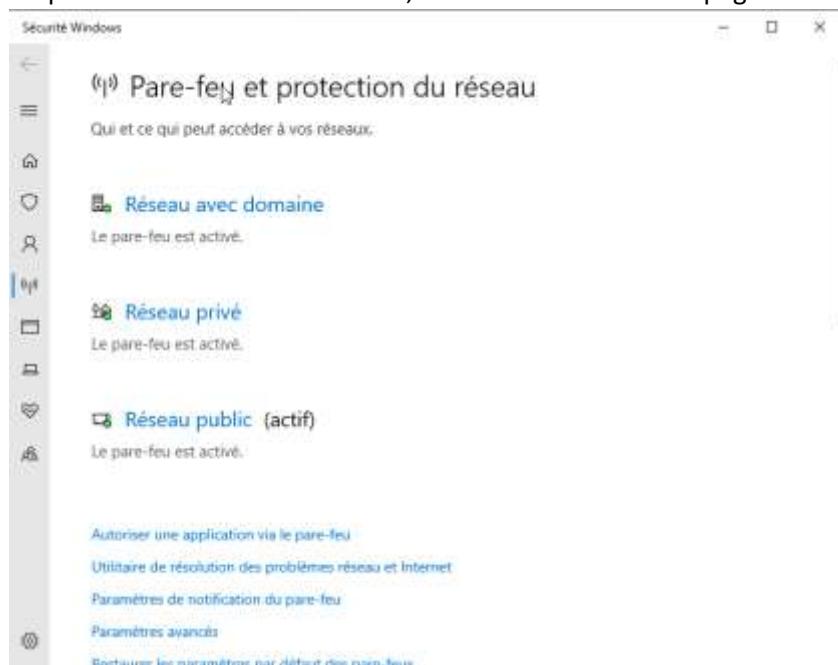
n'oublier pas de cocher « valider les paramètres en quittant, si l'icône en entourée en rouge cela veut dire que vous êtes connecter à internet :



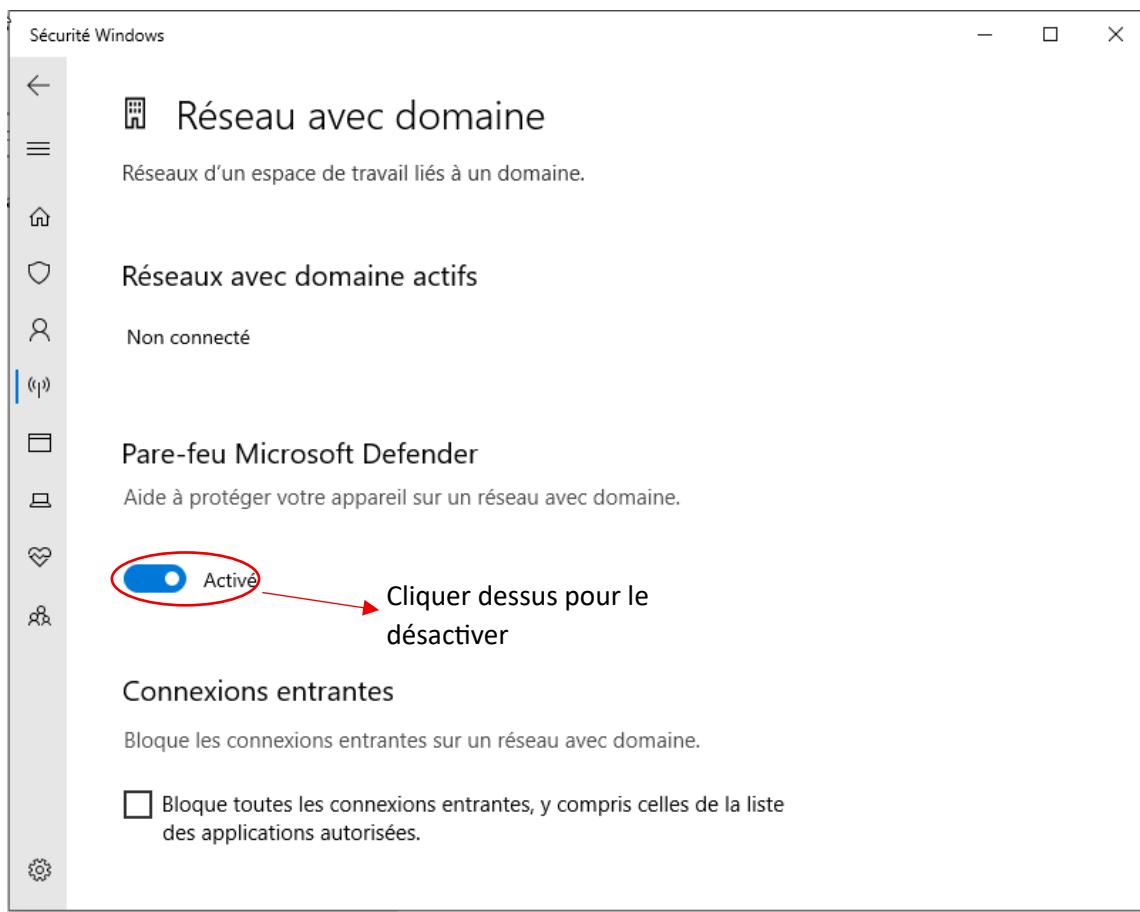
Maintenant il faut aussi désactiver le pare feu Windows defender pour corriger ça, il faut sur cette page :



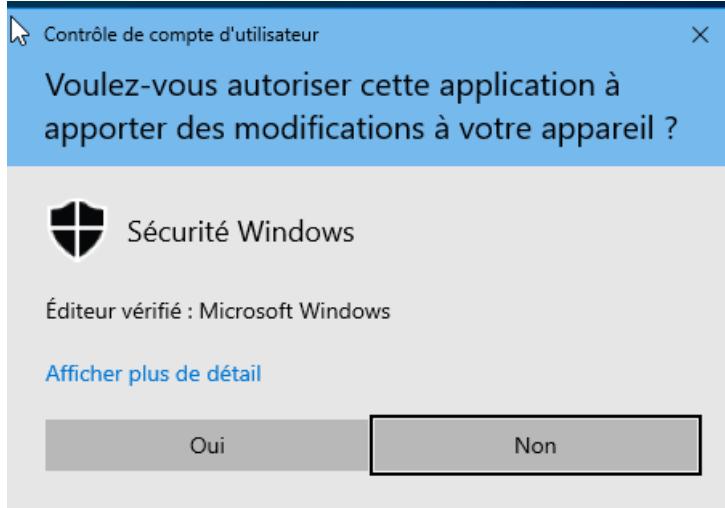
Cliquez sur « Pare-feu Windows », vous arriverez sur cette page :



Cliquez sur « Réseau avec domaine », vous allez arriver sur cette page :



après vous arrivez sur cette fenêtre, cliquer sur oui :



c'est la même chose pour les autres paramètres.

Vous avez terminé la configuration et la mise de l'environnement pour le projet, nous allons maintenant voir le mode d'emploi des différentes scripts python.

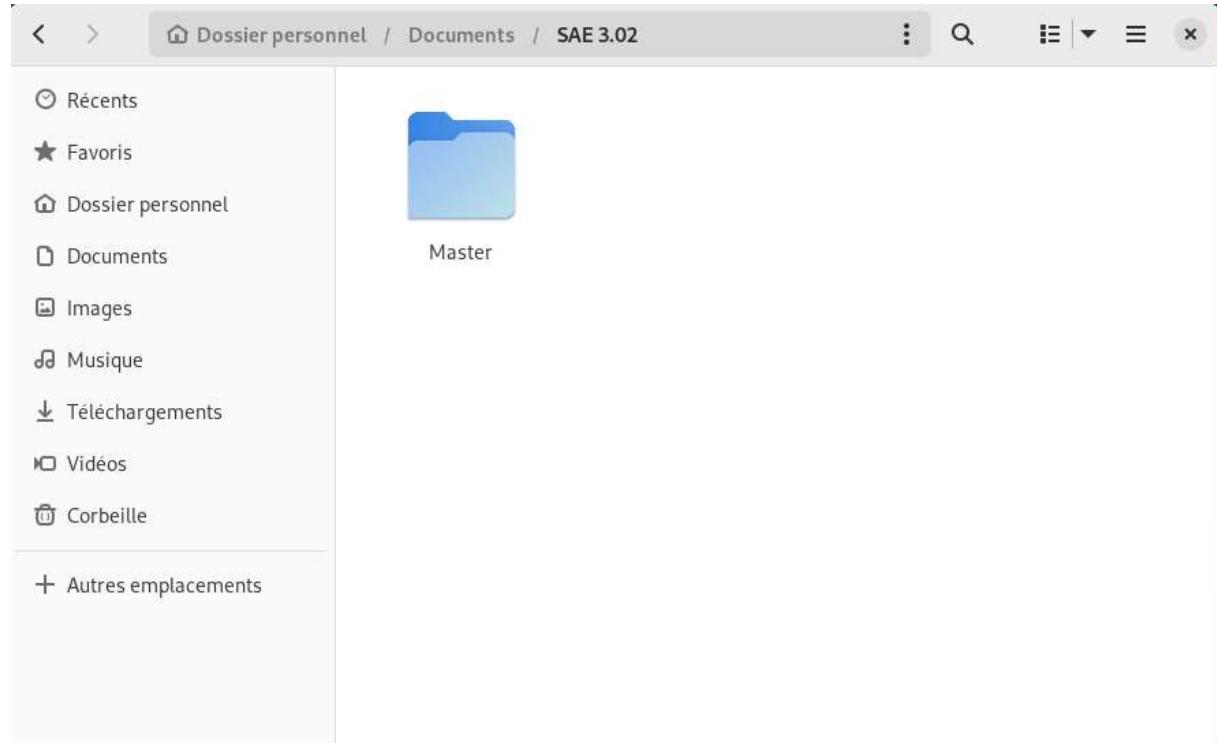
2. Comment utiliser les différentes applications :

Note : assurez vous que tout vos VM soit en marche.

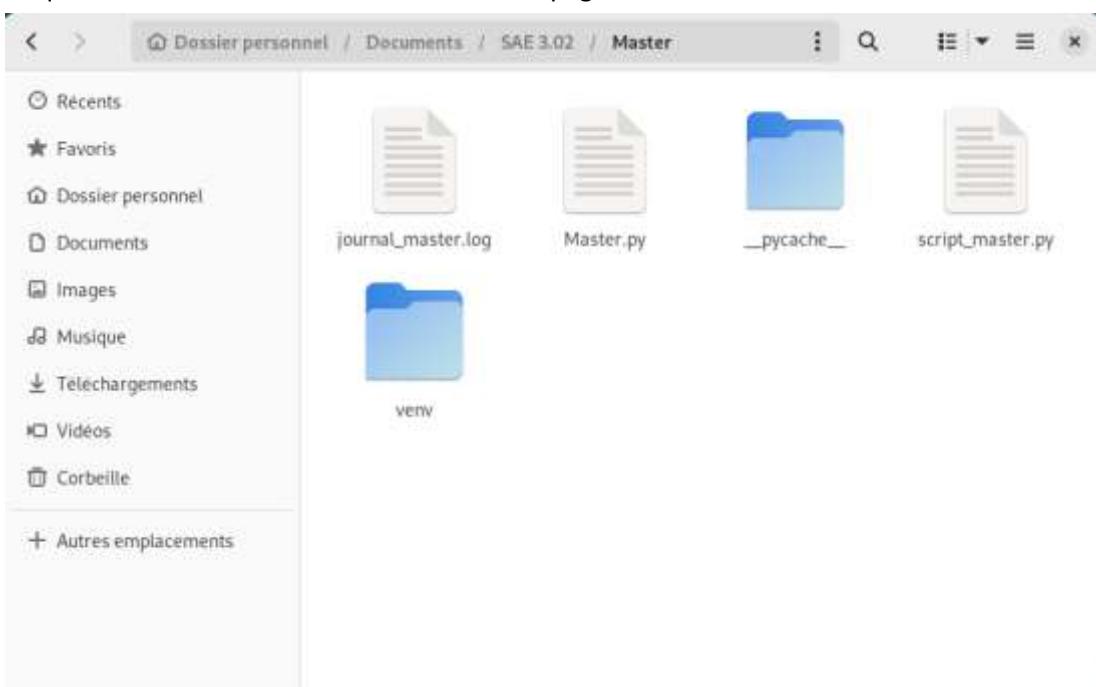
l'ordre de lancement des différentes applications est que vous devez déjà lancer le master, puis les différent et enfin les clients.

2.1. Utilisation du script python pour le master :

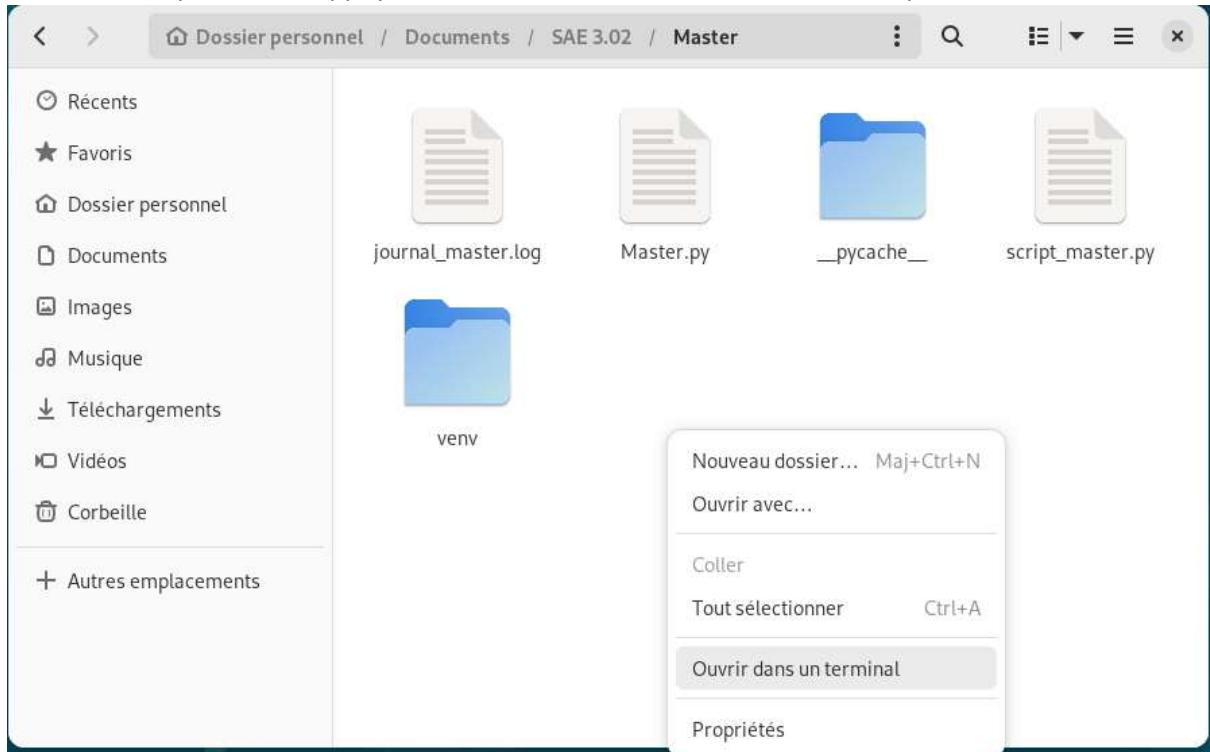
pour commencer sur votre Debian il faut aller dans fichier et dans document dans le dossier que vous avez crée pour le master, comme la capture ci-dessous :



cliquer sur « Master » et vous serez sur cette pages :



il faut faire clique droit et appuyer sur ouvrir dans un terminal, comme la capture ci-dessous :



ça va, vous ouvrir cette page :

A screenshot of a terminal window. The title bar says 'master@vbox: ~/Documents/SAE 3.02/Master'. The command line shows the prompt 'master@vbox: ~/Documents/SAE 3.02/Master\$'. The terminal is currently empty, awaiting input.

de la faire la commande « source venv/bin/activate », si vous voyez ça après avoir exécuter la commande, c'est bon :

```
(venv) master@vbox:~/Documents/SAE 3.02/Master$
```

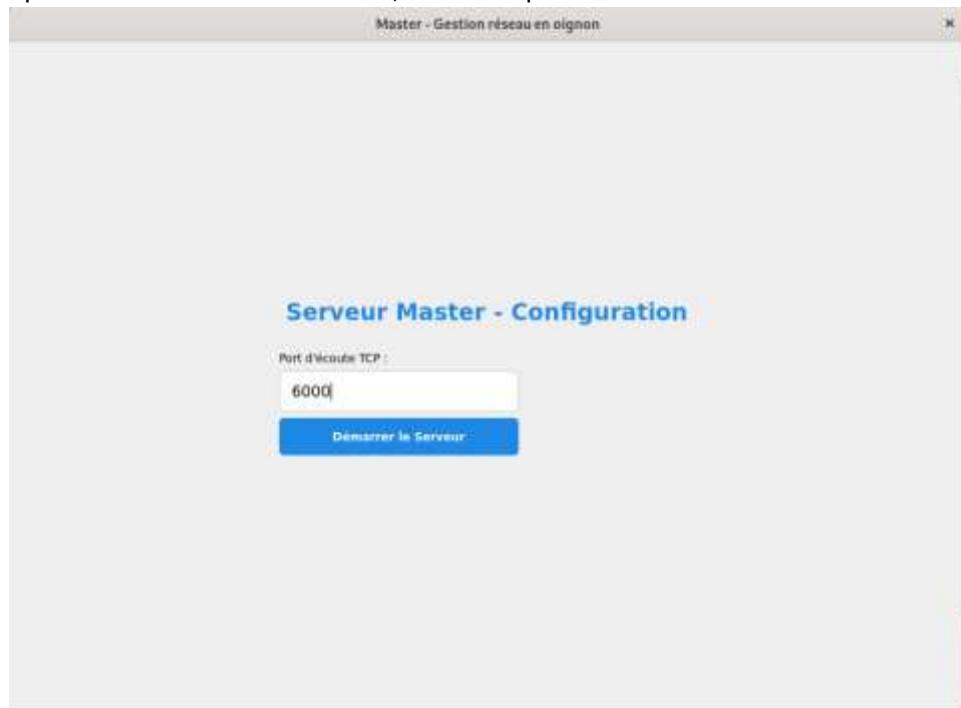
maintenant il faut faire cette commande « python3 script_master.py », pour lancer le script, si vous voyez cette interface, ça veut dire que vous avez lancer le script si vous avez cette comme montrer sur cette capture , faite cette commande « apt install libxcb-xinerama0 libqt5gui5 libxcb-icccm4 libxcb-image0 libxcb-keysyms1 libxcb-randr0 libxcb-render-util0 libxcb-shape0 libxcb-xfixes0 » :

```
qt.qpa.plugin: Could not load the Qt platform plugin "xcb" in "" even though it was found.
```

```
This application failed to start because no Qt platform plugin could be initialized. Reinstalling the application may fix this problem.
```

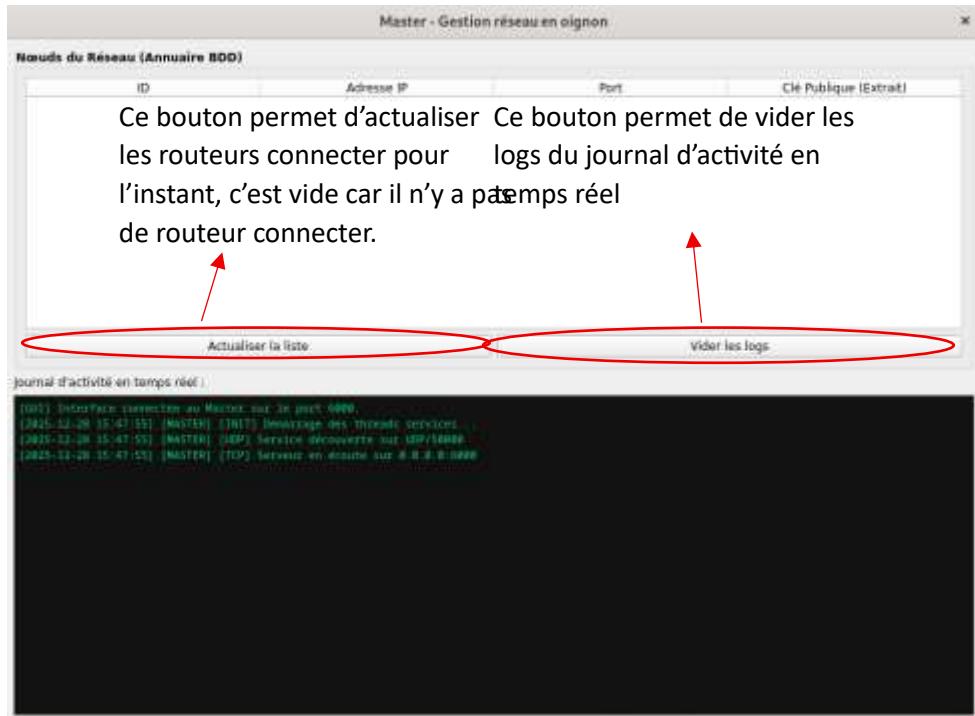
```
Available platform plugins are: eglfs, linuxfb, minimal, minimalegl, offscreen, vnc, wayland-egl, wayland, wayland-xcomposite-egl, wayland-xcomposite-glx, webgl, xcb.
```

après avoir fait cette commande, le master pourra se lancer.



dans la zone saisie saisissez le port TCP que vous voulez je vous conseille dans choisir un entre 6000 et 10000 pour éviter que vous soyiez sur un port déjà utiliser dans mon cas je vais mettre le port 6000.

si vous arrivez sur cette interface ces bon, maintenant je vais vous expliquer les différent bouton etc. sur cette capture d'écran :

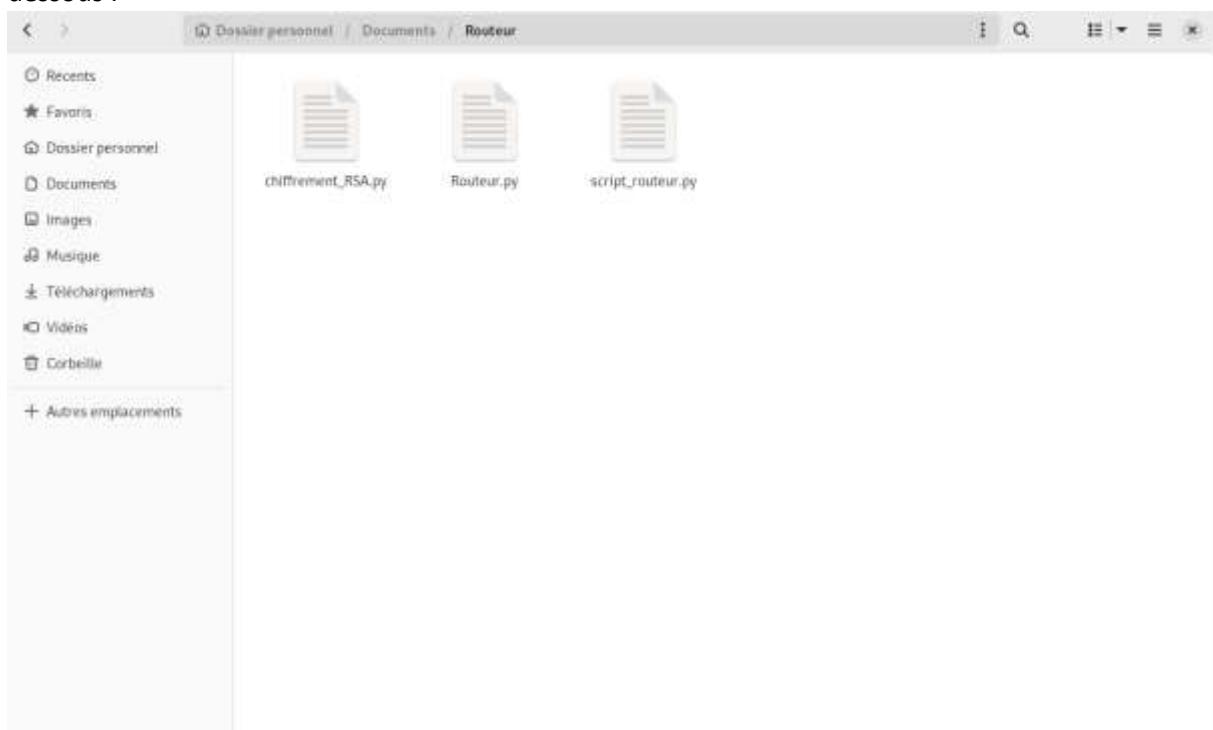


Le journal sur fond noir ressemblant à une console permet de voir les logs en temps réel pour voir si un routeur se connecte, les requête qu'envoie les routeurs ou les clients etc. ensuite la partie avec ID, Adresse IP etc. permet de voir les informations connecter quand un routeur se connecte.

vous savez comment utiliser le script python du master nous allons passer au script pour les routeurs.

2.1. Utilisation du script python pour les routeurs :

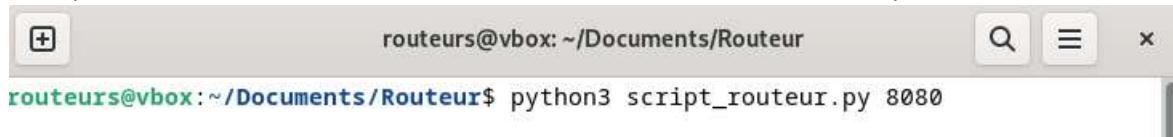
Pour commencer ces quasi la même manip que pour accéder et lancer le script des routeurs, si vous avez suivie mon guide d'installation, vous avez créé 4 dossiers pour chaque routeur qui se trouve dans document quand vous êtes dans un des dossier du routeur, comme la capture d'écran ci-dessous :



vous faites cliquer droite et cliquer sur ouvrir dans un terminal vous arriverez à cette page :

A screenshot of a terminal window titled "routeurs@vbox: ~/Documents/Routeur". The window is mostly empty, with only the title bar and a few icons visible.

Pour lancer le script vous faites cette commande « python3 script_routeur.py » et après vous indiquez le port (par exemple 8080), il faut savoir que si vous voulez que les routeurs soient différenciés par le master il faut mettre un port différent par routeur par exemple pour le routeur 1 vous mettez 8080 mais pour le routeur 2 vous mettez 8081 et ainsi de suite, comme sur la capture ci-dessous :

A screenshot of a terminal window titled "routeurs@vbox: ~/Documents/Routeur". The command "python3 script_routeur.py 8080" is typed into the terminal and is being executed.

Pour commencer il faut indiquer l'adresse ip du master, normalement si vous avez respecté la partie installation l'adresse ip de votre master devrait être 10.0.0.100 :

Entrez l'IP du Master (ex: 192.168.1.34) :

Il faut ensuite donner le port du master, il faut mettre ce que vous avez mis lors de la saisie du port pour le master dans le script du master, dans mon cas c'est 6000 :

Entrez le Port du Master (8080 par défaut) :

Si vous arrivez à cette interface sur le routeur cela veut dire que le routeur est connecté au master :

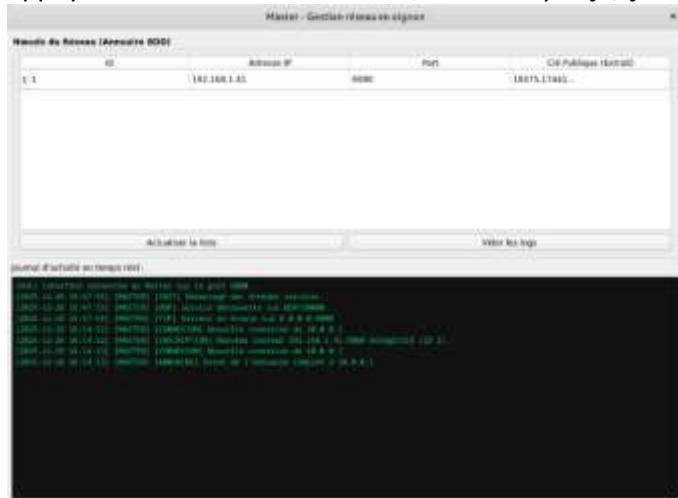
```
[2025-12-28 16:07:47] [ROUTEUR_8080] [INIT] Démarrage du Nœud...
[Crypto] Génération nouvelle paire RSA...
[2025-12-28 16:07:47] [ROUTEUR_8080] [SCRIPT] Initialisation du thread d'écoute
réseau...
Nœud Routeur hybride - port 8080
[2025-12-28 16:07:47] [ROUTEUR_8080] [ÉCOUTE] Prêt à relayer sur le port 8080
```

Menu principal

1. S'inscrire & Sync Annuaire (Auto)
2. Afficher l'Annuaire local
3. Envoyer un message (Client)
0. Quitter

Action >

maintenant nous allons voir les différente fonctionnalité, pour valider une action par exemple saisissez 1 pour « S'inscrire & Sync Annuaire (Auto) » vous permettant d'enregistrer votre routeur au master et de ce synchronisez avec les autre routeur si vous voyez ça sur l'interface du master en appuyant sur « actualiser la liste », si vous voyez ça, ça veut dire que le routeur est enregistrer :



En saisissant 2 vous pourrait voir l'annuaire local du master, en saisissant 2 vous pouvez envoyer un message a soit un client ou à autre routeur, en saisissant 0 vous quittez l'application.

Nous allons voir en détaille, l'action 3, en saisissant 3 sur Action >, ça vous affiche ça :

```
routeurs@vbox: ~/Documents/Routeur
[2025-12-28 16:07:47] [ROUTEUR_8080] [ECOUTE] Prêt à relayer sur le port 8080

Menu principal
1. S'inscrire & Sync Annuaire (Auto)
2. Afficher l'Annuaire local
3. Envoyer un message (Client)
0. Quitter

Action > 1
[2025-12-28 16:14:12] [ROUTEUR_8080] [MASTER] Inscription envoyée (IP: 192.168.1.41)
[2025-12-28 16:14:13] [ROUTEUR_8080] [ANNUAIRE] 1 nœuds synchronisés.

Menu principal
1. S'inscrire & Sync Annuaire (Auto)
2. Afficher l'Annuaire local
3. Envoyer un message (Client)
0. Quitter

Action > 3

Préparation de l'envoie
[2025-12-28 16:20:35] [ROUTEUR_8080] [ANNUAIRE] 1 nœuds synchronisés.
Port de la cible : █
```

pour le port de la cible vous mettez soit le port d'un autre routeur (exemple : 8081 si ces le port du routeur 2) ou le port d'un client (exemple : 8888) dans mon cas le client a le port 8888.

Ensuite on vous demande l'adresse ip du destinataire alors pour les routeurs il y a 2 possibilités soit l'adresse ip interne 10.0.0.1, si vous avez suivi mon guide d'installation pour l'adresse ip bridge d'un routeur (c'est la même pour tous les routeurs), ouvrir un terminal Debian 12 et faire la commande « ip addr », pour les routeurs si l'adresse ip bridge ne marche pas, utiliser l'adresse ip interne et vice et versa. Pour trouver l'adresse ip d'un client Windows, vous devez ouvrir un terminal Windows et faire la commande « ipconfig », pour la VM Windows 10, c'est 10.0.0.2 si vous avez suivie mon guide d'installation, si vous voulez trouver :

|IP de la cible :

Ensute il vous sera demander de saisir le message, saisissez le message :

Message à envoyer : █

Ensute il vous sera demander de saisir le nombre de saut, le nombre de saut est déterminer en fonction du nombre de routeur enregister dans la base de données du master, dans mon cas il n'en n'y a que 1 donc je vais saisir 1 :

|Nombre de relais (sauts) :

si vous voyez ça, sa veut dire que le message a été envoyé :

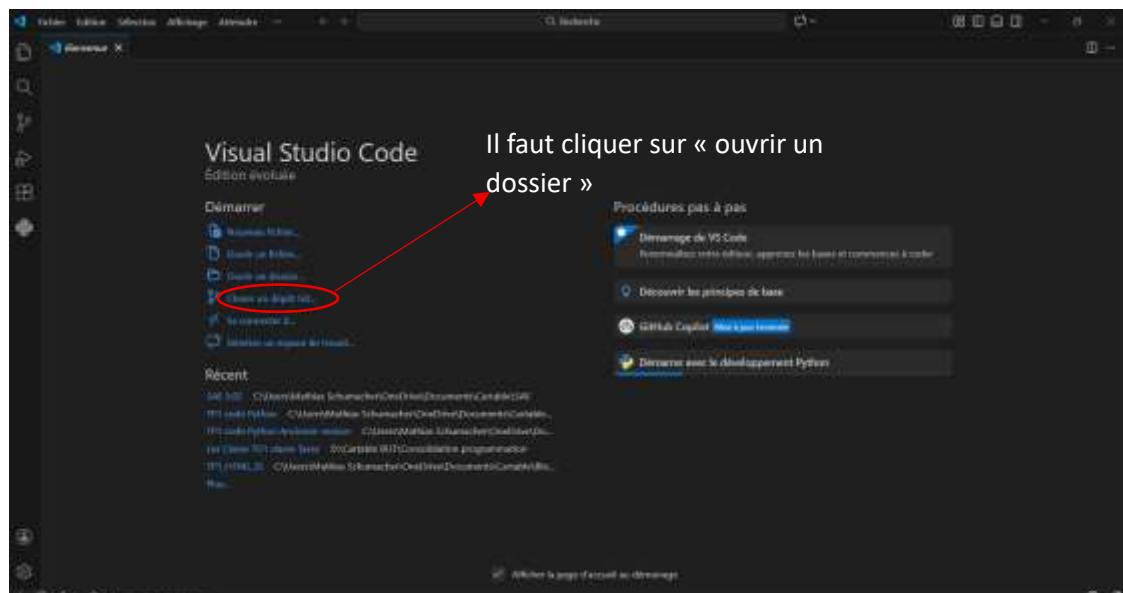
```
[+] Circuit généré (1 sauts) : ['1']
[→] Envoi du paquet chiffré vers le premier saut : 192.168.1.41:8080
[2025-12-28 16:37:49] [ROUTEUR_8080] [CRYPTO] Couche d'oignon retirée avec succès
[2025-12-28 16:37:49] [ROUTEUR_8080] [SORTIE] Livraison finale à 192.168.1.34:8838
```

et comme prévue dans le client j'ai bien reçu le message :

█ [2025-12-28 16:37:48] [CLIENT] [RECEPTION] Message reçu : test

2.3. Utilisation du script python pour les clients :

Pour finir nous allons voir comment utiliser le script client, la pour lancer le script ces un peu différent, dans votre Windows il faut lancer Visual studio code, vous aller arriver sur cette interface : il faut ensuite chercher l'endroit où vous avez mon git clone et ouvrir le dossier, vous aller voir ça :



il faut ensuite cliquer sur script_client.py vous devriez voir ça :

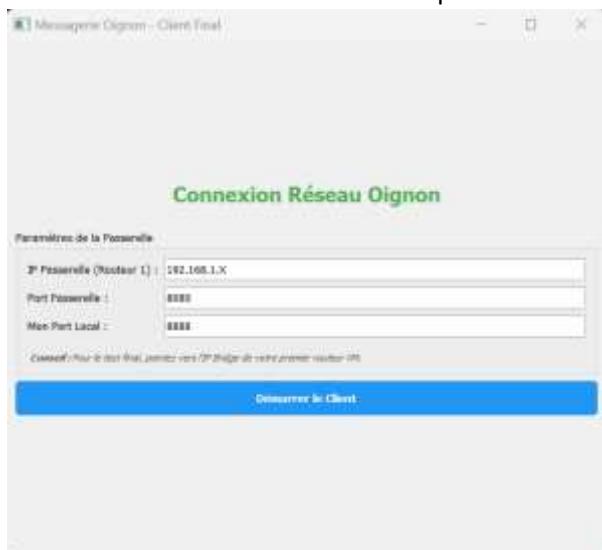
```
python script_client.py
  File "script_client.py", line 21
    Client_PageView(ipClient)
    ^
SyntaxError: invalid syntax
```

The screenshot shows a terminal window with the command `python script_client.py` entered. The output shows a SyntaxError at line 21, indicating an invalid syntax error due to a missing colon after the class definition.

ensuite sur le code faire clique droit, vous devriez voir ça :

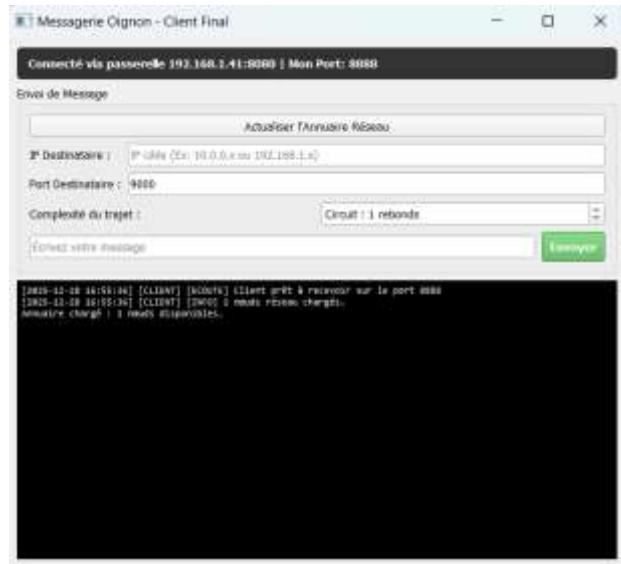


Ensuit vous allez sur exécuter Python puis cliquer sur « Exécuter le fichier python dans le terminal », vous devriez voir cette interface rapidement :



pour l'IP passerelle il y a 2 cas soit vous mettez l'adresse ip bridge du routeur si vous êtes sur le pc physique mais si vous êtes sur la VM Windows 10 , il faut mettre l'adresse ip interne du routeur 10.0.0.1 si vous avez bien suivie mon guide d'installation, pour le port passerelle vous mettez le port que vous avez donné a un de routeur dans mon cas sa sera 8080 pour « Mon Port Local » il faut mettre un port qui n'est pas celui de l'autre client, d'un des routeur ou celui du master dans mon cas sa sera 8888.

Si tout est bon vous arriverez sur cette interface :



Le bouton « Actualiser l'Annuaire Réseau » permet d'actualiser l'annuaire et récupérer l'annuaire des différent routeur, pour le champs de saisie « IP Destinataire : » vous saisissez l'adresse ip soit du client B sur la VM Windows 10 si vous envoyez un message depuis le pc physique pour envoyer un message a la VM Windows 10, il faut mettre l'adresse ip 10.0.0.2, pour envoyer un message depuis le pc physique a un routeur il faut saisir l'adresse bridge de la VM qui gère les routeur, si vous envoyez un message depuis la VM Windows 10, il faut pour envoyer a un message au pc physique saisir l'adresse ip du client A que vous avez obtenue avec la commande « ipconfig » sur votre pc physique, pour envoyer a un message a un routeur il faut saisir l'adresse ip interne des routeur 10.0.0.1 si vous bien suivie mon guide d'installation. Pour le port destinataire il faut soit saisir le port du client B sur la VM Windows 10 « exemple : 9999 », pour envoyer un message a un routeur il faut saisir un des port d'un des routeur « exemple : 8080 , pour le circuit ces en fonction du nombre de routeur enregistrer dans l'annuaire si il n'y que 1 un routeur on ne peut pas faire un chemin avec 2 routeurs, enfin vous écrivez un message et puis quand tout est bon appuyer sur le bouton « Envoyer », si vous voyez sa dans la console de l'interface client, sa veut dire que ces bon :

```
[2025-12-28 16:55:36] [CLIENT] [ECOUTE] Client prêt à recevoir sur le port 8888
[2025-12-28 16:55:36] [CLIENT] [INFO] 1 nœuds réseau chargés.
Annuaire chargé : 1 nœuds disponibles.
[2025-12-28 17:08:45] [CLIENT] [INFO] 1 nœuds réseau chargés.
[2025-12-28 17:08:45] [CLIENT] [ALERTE] Un seul nœud dispo : l'entrée sera la cible.
[2025-12-28 17:08:45] [CLIENT] [OIGNON] Circuit créé : ['1'] (Cible: 192.168.1.41:8080)
[2025-12-28 17:08:45] [CLIENT] [ROUTAGE] Connexion via LOCALE (Intnet) vers 192.168.1.41:8080
[2025-12-28 17:08:45] [CLIENT] [SUCCÈS] Paquet livré au premier saut : 1
```

et comme prévu le routeur a bien reçus le message :

```
Action > [2025-12-28 17:08:45] [ROUTEUR_8080] [CRYPTO] Couche d'oignon retirée avec succès.  
[2025-12-28 17:08:46] [ROUTEUR_8080] [ARRIVÉ] Message reçu pour moi : Test
```

voilà maintenant vous savez utiliser chaque script, nous allons maintenant voir le dépannage je vais vous donner les solutions pour corriger certain problème que j'ai eu durant les tests.

3. Dépannage :

1. Le PC physique ne parvient pas à joindre le réseau interne (10.0.0.0/24) :

Symptôme : Depuis votre PC physique (Client A), vous arrivez à "pinger" l'adresse Bridge du routeur, mais impossible de joindre le Master (10.0.0.100) ou le Client B (10.0.0.2).

Cause : Votre PC physique possède sa propre table de routage. Il ne sait pas par défaut que pour atteindre le réseau privé 10.0.0.0, il doit envoyer les paquets à l'adresse IP Bridge de votre VM Routeur.

Solution (Sous Windows) : Vous devez ajouter une route statique manuellement dans le terminal de votre machine physique.

Identifiez l'adresse IP Bridge de votre routeur sur Debian avec la commande ip addr (ex: 192.168.1.41).

Ouvrez une invite de commande (CMD) en tant qu'administrateur sur votre PC physique.

Tapez la commande suivante :

```
« route add 10.0.0.0 mask 255.255.255.0 <IP_BRIDGE_DU_ROUTEUR> »
```

(Remplacez <IP_BRIDGE_DU_ROUTEUR> par l'adresse notée à l'étape 1).

Vérification : Une fois la route ajoutée, tentez un ping 10.0.0.100. Si le Master répond, la communication entre votre PC physique et l'infrastructure virtuelle est opérationnelle.

2. Problèmes d'interfaces réseau sur Debian :

Symptôme : La commande ip addr ne montre pas les mêmes noms d'interfaces que dans ton guide (ex: eth0 au lieu de enp0s3).

Cause : Selon la version de VirtualBox ou le matériel, Debian nomme les cartes réseau différemment.

Solution : Il faut impérativement adapter le fichier /etc/network/interfaces avec les noms exacts affichés par ip addr. Adapter aussi la commande iptables pour pointer vers la bonne interface de sortie (WAN).

3. Problèmes liés à MariaDB et au Master :

Symptôme : Le script Master.py affiche une erreur de connexion à la base de données.

Causes possibles :

- Le mot de passe dans le code Python ne correspond pas à celui défini lors de la commande, `mysql_secure_installation` (tu as utilisé "toto").
- Le service MariaDB n'est pas lancé.

Solution :

- Vérifier le statut : `sudo systemctl status mariadb`.
- S'assurer que l'utilisateur root a bien les droits avec le mot de passe nativement.

4. Problèmes avec Python et les environnements virtuels :

Symptôme : Erreur `ModuleNotFoundError: No module named 'PyQt5'` ou `'mysql.connector'`.

Cause : Les bibliothèques ont été installées en dehors de l'environnement virtuel (`venv`) ou celui-ci n'est pas activé.

Solution :

- Toujours vérifier que (`venv`) est écrit devant le prompt du terminal.
- Si absent, relancer : `source venv/bin/activate`.

5. Problème de Forwarding (Le Client B n'a pas Internet) :

Symptôme : Le Client B peut pinguer le Routeur mais pas le Master ou Google.

Solution :

- Vérifier la valeur : `cat /proc/sys/net/ipv4/ip_forward`.
- Si elle affiche 0, relancer `sysctl -p` après avoir vérifié le fichier `/etc/sysctl.conf`.