

# Procédure d'installation SAE302

## Table des matières

|   |   |
|---|---|
| Prérequis .....                                   | 2 |
| Récupération des fichiers source .....            | 3 |
| Via le navigateur de fichier .....                | 3 |
| Via interface graphique .....                     | 4 |
| Différents éléments du projet .....               | 5 |
| Installation des dépendances Python .....         | 6 |
| Lancer un serveur .....                           | 7 |
| Lancer un client .....                            | 8 |
| Plusieurs clients et serveurs simultanément ..... | 9 |

# Prérequis

Afin de procéder à la mise en place des différents éléments du projet et leur bon fonctionnement, il est nécessaire que les conditions suivantes soient validées :

- Effectuer l'installation sur une machine Windows 10 ou Windows 11
- Avoir les outils suivants d'installés sur le système d'exploitation :
  - Git : <https://git-scm.com/downloads/win>
  - Python 3.12\* : <https://www.python.org/downloads/release/python-3120/>
  - Java\* : <https://www.java.com/fr/download/manual.jsp>
  - Outils de compilation Gcc et G++\* : <https://winlibs.com/> (pour le tutoriel d'installation complet de : <https://www.youtube.com/watch?v=y-i96kqT53A>)
- Avoir accès à internet pour accéder au dossier GitHub contenant le projet

\* A ajouter dans les variables d'environnement système

# Récupération des fichiers source

Il y a deux façons de récupérer le projet, soit avec git via un terminal ou le navigateur de fichier, soit directement via GitHub. Dans les deux cas vous finirez avec un fichier «`.gitignore`» qui sert à ne pas sauvegarder certains contenus dans le répertoire GitHub et le dossier suivant comportant les codes de la ressource R3.09 ainsi que de la SAE3.02 :

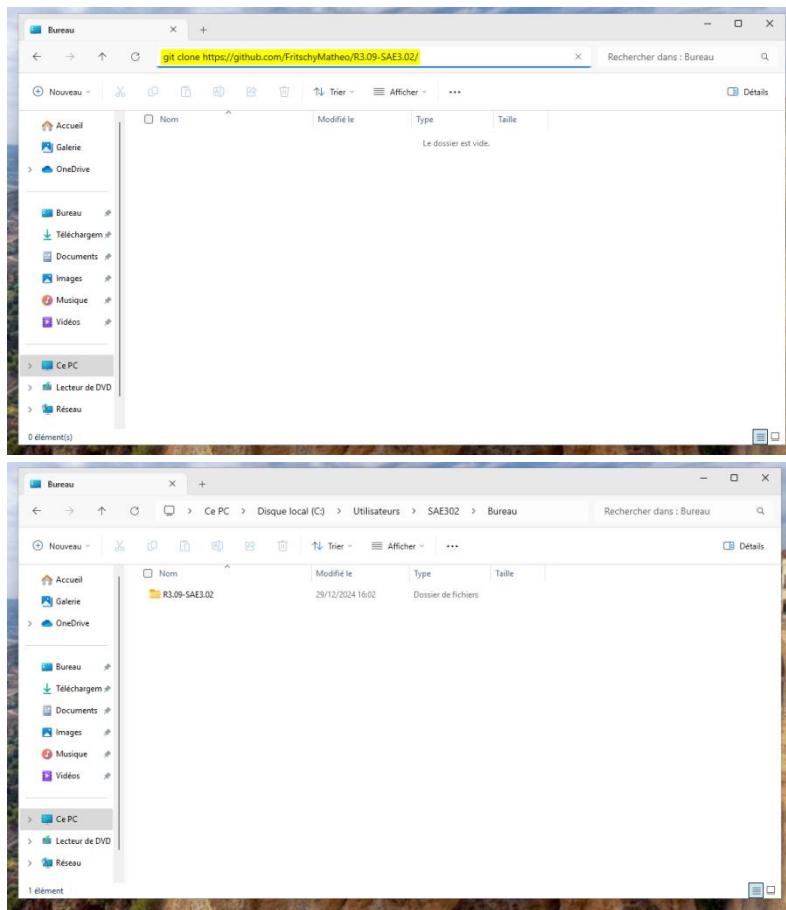
|            |                  |                     |
|------------|------------------|---------------------|
| R3.09      | 29/12/2024 16:18 | Dossier de fichiers |
| SAE3.02    | 29/12/2024 16:18 | Dossier de fichiers |
| .gitignore | 29/12/2024 16:18 | txtfile 4 Ko        |

## Via le navigateur de fichier

Rendez vous à l'endroit où vous voulez télécharger le code et faites la commande suivante à la place du chemin du dossier ou bien dans un terminal à l'endroit voulu :

```
git clone https://github.com/FritschyMatheo/R3.09-SAE3.02/
```

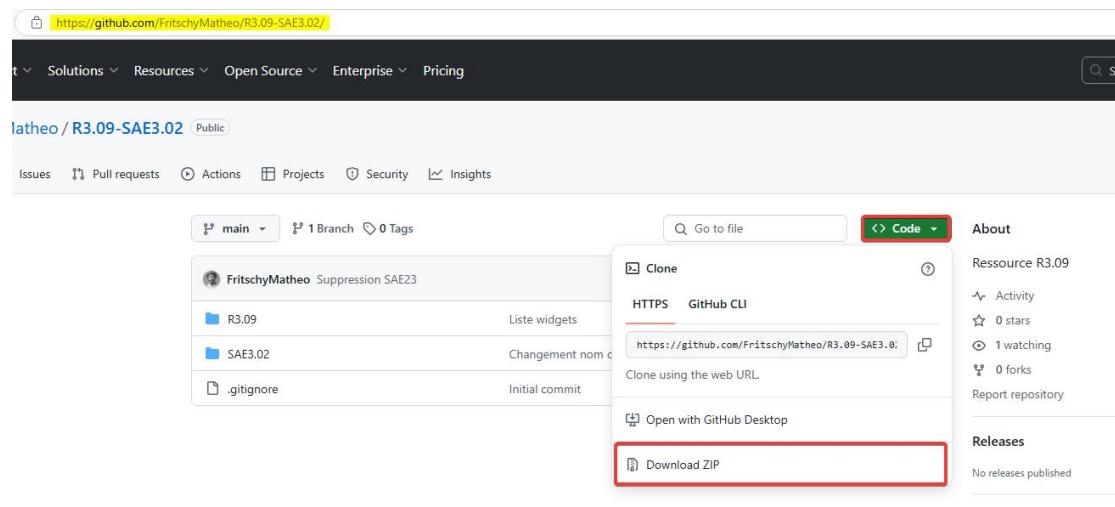
Puis appuyer sur Entrée.



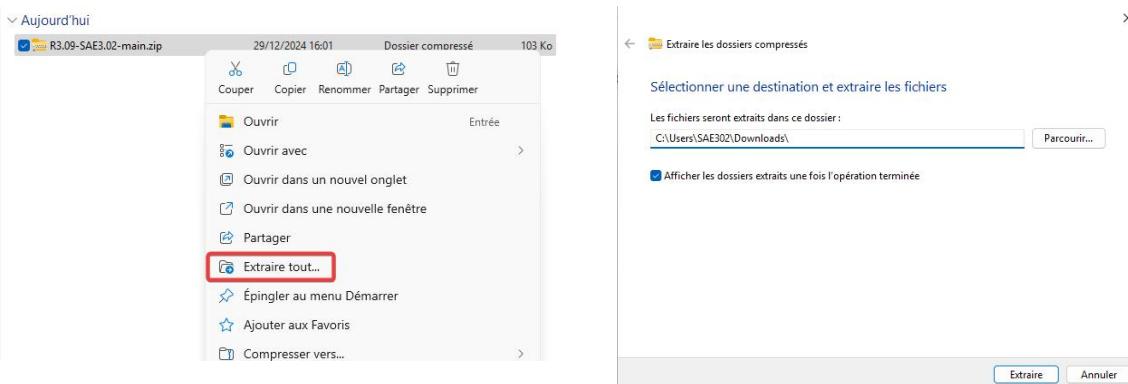
## Via interface graphique

Rendez vous directement sur le dossier GitHub via le lien :

<https://github.com/FritschyMatheo/R3.09-SAE3.02> et téléchargez le depuis le menu déroulant «Code» > «Download ZIP».



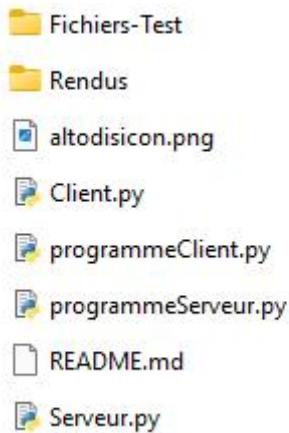
Le dossier se trouve alors dans les téléchargements et il faut maintenant l'extraire où vous voulez travailler avec comme ici dans les téléchargements par exemple :



|                        |                  |                     |        |
|------------------------|------------------|---------------------|--------|
| R3.09-SAE3.02-main.zip | 29/12/2024 16:01 | Dossier compressé   | 103 Ko |
| R3.09-SAE3.02-main     | 29/12/2024 16:24 | Dossier de fichiers |        |

# Différents éléments du projet

Dans les deux cas, ce qui nous intéresse pour utiliser les serveurs et les clients se trouve dans le dossier «**SAE3.02**». Ce dossier contient donc plusieurs fichiers et un dossier :



- **Fichiers-Test** : ce sont les fichiers de code de différents langages qui permettent de tester le bon fonctionnement du projet.
- **Rendus** : contient tous les rendus du projet (procédure d'installation, document de réponse)
- **Altodisicon.png** : logo de l'entreprise dans laquelle je suis en alternance qui est utilisé comme icon des fenêtres pour les clients.
- **Client.py** : fichier contenant le code la classe Client.
- **programmeClient.py** : fichier contenant le code principal des clients.
- **programmeServeur.py** : fichier contenant le code principal des serveurs.
- **README.md** : fichier de présentation de la SAE et des différents fichiers.
- **Serveur.py** : fichier contenant le code la classe Serveur.

# Installation des dépendances Python

Le projet utilise certains modules Python qui ne sont pas natifs et doivent donc être installés. Pour ce faire, il vous suffit d'utiliser les commandes suivantes\* dans un terminal :

```
python.exe -m pip install --upgrade pip
```

Pour mettre à jour l'outil d'installation pip.

```
pip install PyQt6
```

Pour installer le module qui permet la gestion d'interface graphique du code : **PyQt6**.

*\*Si cela ne fonctionne pas avec «python.exe», il se peut que vous deviez utiliser «py.exe» ou encore «python3.exe» selon la version de Python que vous utilisez. Cela dit, pour le bon fonctionnement du projet et principalement pour la librairie PyQt6, je recommande grandement **python 3.12**.*

## Lancer un serveur

Pour chaque serveur que vous voulez lancer, il faut ouvrir une nouvelle fenêtre de terminal et lancer le fichier «programmeServeur.py» avec comme argument le port que vous voulez lui donner. Par défaut ce dernier est initialisé à 50000 s'il n'y a pas d'argument. Le port du serveur doit être compris entre 49152 et 65535.

Ici, j'utilise PowerShell sous Windows 11 mais cela fonctionne également avec Cmd et sur Windows 10 et même principe sous Linux. Cependant, il faut s'assurer d'être dans le même dossier que les fichiers présentés dans la partie différents éléments du projet.

Avec la commande : *python.exe .\programmeServeur.py 55000*, vous lancez donc un serveur avec l'adresse ip local 127.0.0.1 et avec le port 55000.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

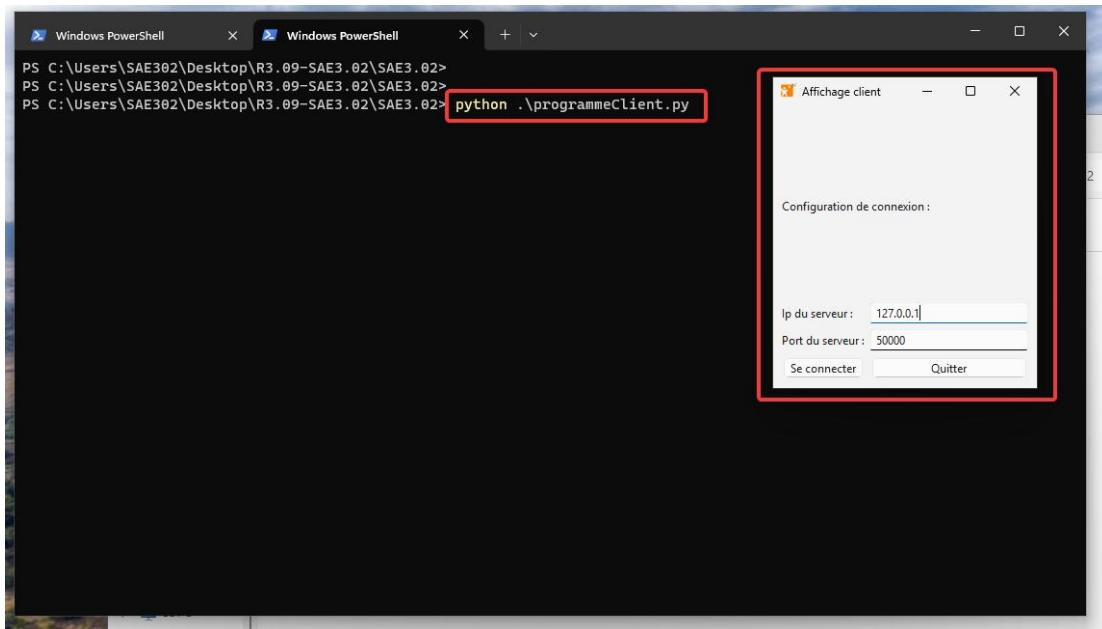
Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https://aka.ms/PSWindows

PS C:\Users\SAE302\Desktop\R3.09-SAE3.02\SAE3.02> python .\programmeServeur.py 55000
Le serveur Serveur a comme ip : 127.0.0.1 et utilise le port : 55000
En attente de connexion...
```

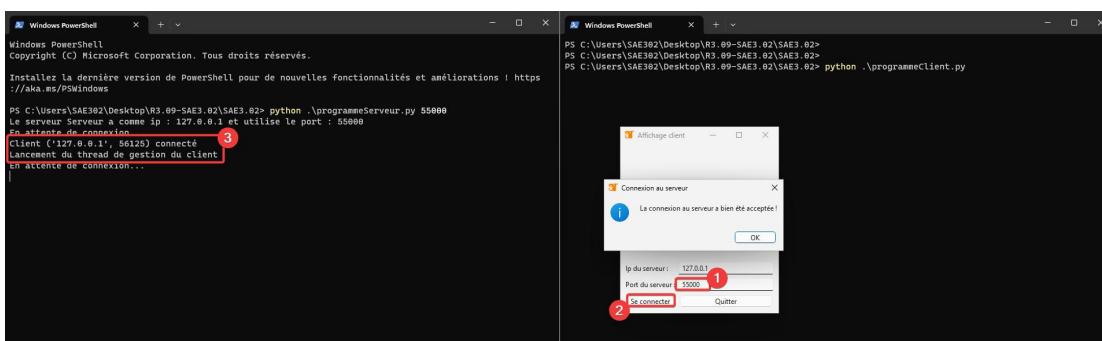
## Lancer un client

Pour lancer un client, tout comme le serveur dans un terminal, il suffit de faire :  
 python.exe programmeClient.py

Il n'y a pas d'argument à entrer ici et un client se lance avec le port de base et l'adresse ip 127.0.0.1 du serveur :



Dans le cas présent, le serveur a le port 55000 donc il faut le changer sur le client pour se connecter :



## Plusieurs clients et serveurs simultanément

Il est tout aussi simple de lancer plusieurs serveurs et clients qu'un seul. Il suffit d'ouvrir un terminal pour chaque entité et changer le port de chaque serveur. Voilà ce que cela donne pour deux clients et deux serveurs :

