

Document d'installation SAE302:

Table des matières :

Document d'installation SAE302:	1
Prérequis :	1
Télécharger le code source :	2
Option 1 : Cloner le dépôt GitHub	2
Option 2 : Télécharger le fichier ZIP	2
Architecture du dossier :	3
Installer les dépendances :	3
Lancer le serveur :	4
Lancer le client :	4
Personnalisation et modifications :	5

Prérequis :

Avant de commencer, assurez-vous d'avoir les outils nécessaires pour installer, configurer, et exécuter correctement le projet. Voici la liste complète des éléments requis :

- **Système d'exploitation** : Windows, macOS ou Linux.
- **Python** : Version 3.9 ou supérieure. → <https://www.python.org/downloads/>
- **Compilateurs** :
 - **C/C++** : gcc et g++
 - Windows : installer MinGW (ne pas oublier d'ajouter MinGW au PATH) → <https://winlibs.com/>
 - Linux : via la commande « `sudo apt install build-essential` »
 - MacOS : via la commande « `xcode-select --install` »
 - **Java et javac** : JDK (Java Development Kit)
→ <https://www.oracle.com/java/technologies/javase-downloads.html>
- **Connexion Internet** : Pour télécharger les bibliothèques nécessaires et le code source.
- **Git** : Pour cloner le dépôt GitHub (sauf si vous téléchargez manuellement les fichiers). → <https://git-scm.com/downloads>

Assurez-vous que les exécutables sont accessibles depuis le terminal en executant ces commandes :

```
python3 --version
gcc --version
g++ --version
java --version
javac --version
```

Télécharger le code source :

Pour cette étape il existe deux manières de faire, soit cloner le dépôt GitHub à l'aide de git, soit télécharger manuellement le projet.

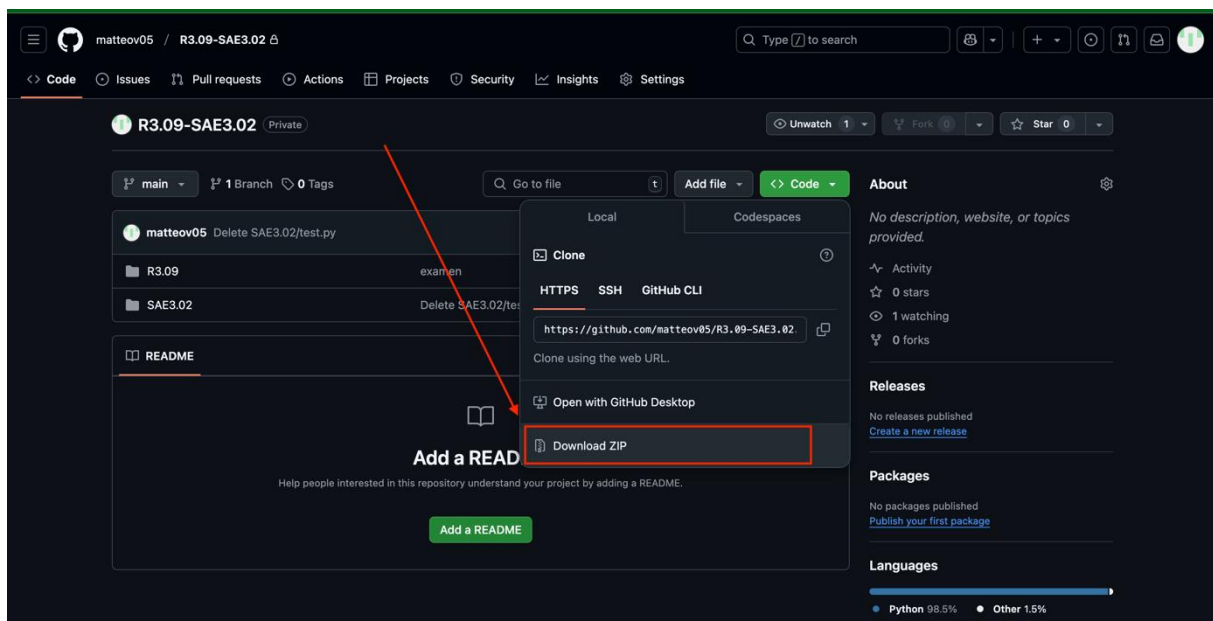
Option 1 : Cloner le dépôt GitHub

Si vous avez installé git, il y a encore une fois deux option soit allez dans le terminal ou dans l'explorateur de fichier windows ici nous ferons cela. Ensuite faites la commande ci-dessous dans la barre de l'explorateur de recherche puis enter :

```
git clone https://github.com/matteov05/R3.09-SAE3.02.git
```

Option 2 : Télécharger le fichier ZIP

- Accédez au dépôt GitHub dans un navigateur web via ce lien : <https://github.com/matteov05/R3.09-SAE3.02.git>
- Puis cliquez sur le bouton **Code** et sélectionnez **Download ZIP**.



- Ensuite vous trouverez dans les téléchargements votre archive zip ou il faudra extraire les fichiers à l'endroit souhaitez.

Téléchargements			
Aujourd'hui	Taille	Type	Date de l'ajout
R3.09-SAE3.02-main.zip	14 ko	Archive ZIP	aujourd'hui à 18:16

Architecture du dossier :

Peu importe l'option utilisé vous avez à présent le dossier R-3.09-SAE3.02.
A l'intérieur vous pourrez y retrouver deux dossiers celui qui va nous intéressez ici est le dossier SAE3.02 comme ci-dessous :

R3.09-SAE3.02-main			
Nom	Date de modification	Taille	Type
> R3.09	aujourd'hui à 17:55	--	Dossier
> SAE3.02	aujourd'hui à 17:55	--	Dossier

Une fois dedans vous retrouverez les deux codes python principaux celui du serveur et du client, mais aussi dans le dossier test les codes de test à envoyer au client lorsqu'il sera en marche.

SAE3.02			
Nom	Date de modification	Taille	Type
clientSAE302.py	aujourd'hui à 17:55	5 ko	Python
serveurSAE302.py	aujourd'hui à 17:55	8 ko	Python
> test	aujourd'hui à 17:55	--	Dossier

Installer les dépendances :

Le projet utilise un module python qui n'est pas natif, il doit donc être installé.

Tout d'abord nous allons vérifier si pip est installé :

Tapez la commande suivante :

`pip --version`

Si pip n'est pas installé, suivez ces étapes :

- **Windows** : Téléchargez et exécutez [get-pip.py](#).
- **Linux** : Installez avec la commande « `sudo apt install python3-pip` » .
- **macOS** : Téléchargez et exécutez [get-pip.py](#).

Puis une fois pip installé exécuter la commande ci-dessous pour installer le module qui permet la gestion de l'interface graphique : PyQt5.

```
pip install PyQt5
```

Lancer le serveur :

Pour lancer un ou plusieurs serveurs, il faut ouvrir une nouvelle fenêtre de terminal par serveur et lancer le fichier *serveurSAE302.py* avec comme argument le port que vous voulez lui donner. Par défaut, ce dernier est initialisé à 10000. Vous avez la possibilité de donner à un serveur plusieurs ports en arguments et on pourra donc se connecter avec le client sur plusieurs ports.

Assurez-vous bien à chaque fois que vous êtes dans le même dossier que le fichier présenté plus haut.

Comme vous pouvez le voir ci-dessous, je me suis déplacé où est situé le fichier ici sur le bureau à l'aide de « cd » puis j'ai lancé un serveur avec la commande « *python3 serveurSAE302.py* ». A noter que l'IP par défaut est toujours 0.0.0.0

```
● matteovargiu@Air-de-Matteo ~ % cd Desktop
○ matteovargiu@Air-de-Matteo Desktop % python3 serveurSAE302.py
  Serveur démarré sur 0.0.0.0:10000
  En attente de connexions...
```

Dans ce second exemple vous pouvez voir que j'ai lancé un serveur sur deux ports différents entré en argument.

```
○ matteovargiu@Air-de-Matteo Desktop % python3 serveurSAE302.py 55000 60000
  Serveur démarré sur 0.0.0.0:55000
  En attente de connexions...
  Serveur démarré sur 0.0.0.0:60000
  En attente de connexions...
```

A noter également que pour fermer un serveur il faut faire CTRL+C dans le terminal.

Lancer le client :

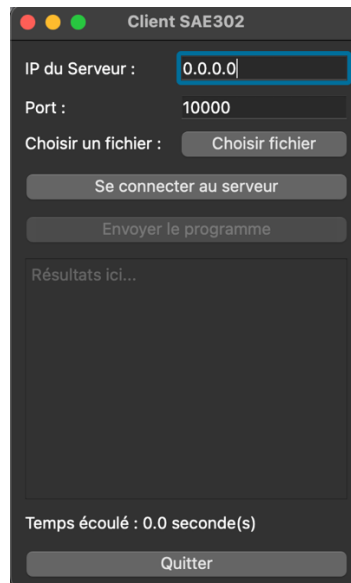
Pour lancer un client il faut faire comme le serveur lancer le fichier *clientSAE302.py* dans un nouveau terminal avec la commande « *python3 clientSAE302.py* ». Ici aucun argument à rajouter.

```
○ Air-de-Matteo:Desktop matteovargiu$ python3 clientSAE302.py
```

Vous aurez ensuite l'interface graphique du client qui s'affiche, ou vous pourrez modifier le port et l'IP du serveur si besoin.

En appuyant sur le bouton choisir un fichier vous avez la possibilité de choisir un fichier à envoyer au serveur une fois sélectionner appuyer sur « se connecter au serveur » puis vérifier si le serveur est occupé ou non.

Une fois connecté le bouton « envoyer le programme se débloquent » et vous aurez plus qu'à attendre le résultat.



Pour lancer un second client il suffit de recréer un terminal et de relancer le fichier.

Personnalisation et modifications :

Si vous souhaitez modifier le code, utilisez un éditeur de code tel que Visual Studio Code ou PyCharm. Et n'oubliez pas après modifications de relancer le serveur et le client pour appliquer les changements.

Vous avez à présent toutes les clés en main pour faire fonctionner ce projet ! N'hésitez pas à visionner ma vidéo de démonstration ou je montre les différentes fonctionnalités et comment s'en servir.

https://www.youtube.com/watch?v=KTV_QSFCRqc