

# Documentation d'installation et d'utilisation

En 3 parties.

Akif Aslankilic  
Frédéric Drouhin  
2024/2025



## — INTRODUCTION —

L'objectif de ce document et de vous aidez a lancer les programmes '*client.py*' et '*serveur.py*' sans aucun problème. Je vais vous guider pas à pas jusqu'à la connexion entre le client et le serveur.

Par la suite, vous allez pouvoir consulter un autre fichier nommé '*Rapport Finale*' où on vous expliquera les instructions pour déployer et tester le système.

Je vous mets ci-dessous les documents annexes qui peuvent vous servir :

Télécharger Python : [Installer Python](#)

Télécharger C : [Montre-moi](#)

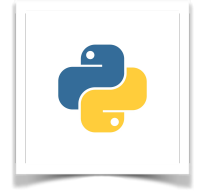
Problème avec les terminaux d'Apple ? : [SuperUser](#)

Lien du GITHUB pour installer les fichiers : [GitHub](#)

Bonne lecture.



## 1. Installation du Client et du Serveur



### 1.1 Installation de Python

Pour commencer, il nous faut tout d'abord, installé Python sur notre machine, à l'aide de ces différentes méthodes :

- En l'installant sur le [site de officiel de Python](https://www.python.org/downloads/) et cochez l'option « **Add Python to PATH** » pour le mettre dans la variable d'environnement automatiquement :  
<https://www.python.org/downloads/>  
Après cela, vous pouvez créer un fichier 'test.py' par exemple et le **lancer depuis le terminal** comme ceci : `python3 test.py`
- On peut aussi téléchargez et installez un **IDE** comme **PyCharm** que j'utilise et que je recommande lorsque je fais du Python. Vous pouvez aussi faire ça sur **Visual Studio Code**. Lors du lancement de votre projet, et une fois que votre fichier .py est en place , l'IDE vous proposera automatiquement l'installation de Python. Si ce n'est pas le cas, recherchez le et installez le dans l'onglet 'Extension'

### 1.2 Installation des librairies



L'installation de **PyQt5** se fait directement depuis la ligne de commande (sur Win.) :

```
akif@MacBook-Pro-de-Akif SAE3.02 % pip install Pyqt5
```

ou: **brew install pyqt5** si vous êtes **MAC**.

Il faut aussi importer les librairies '**socket**' et '**Threading**' qui sont des librairies déjà incluses lors de l'installation de Python.

Pensez aussi à installer ceci pour le fonctionnement du programme C :

```
pip install gcc - ou - brew install gcc
```



### 1.3 Mise en place des fichiers

Maintenant qu'on a bien téléchargé nos librairies, il faut maintenant trouver un endroit où les mettre.

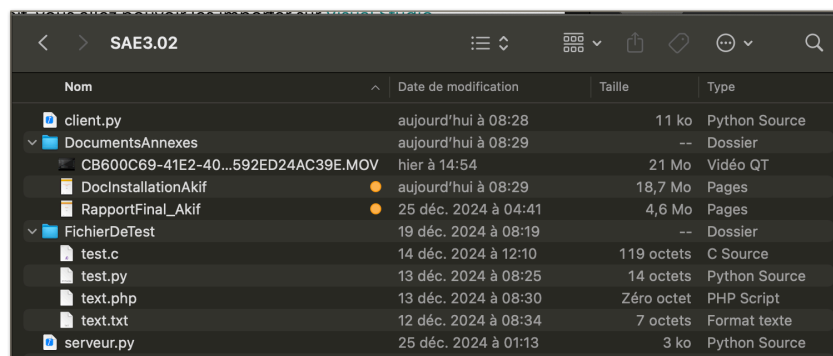
Veuillez télécharger le dossier depuis ce lien Github : [GitHub](#)

Cliquez sur 'Code' puis 'Download Zip'.

Ensuite dézipper le dossier depuis l'explorateur de fichier.

Une fois ceci fait, vous allez pouvoir les importer sur [Visual Studio Code](#) OU exécuter ces fichiers depuis le terminal.

**Maintenant que vous avez tout installé, veuillez mettre les différents fichiers comme indiqué ci-dessous :**



Nom	Date de modification	Taille	Type
client.py	aujourd'hui à 08:28	11 ko	Python Source
DocumentsAnnexes	aujourd'hui à 08:29	--	Dossier
CB600C69-41E2-40...592ED24AC39E.MOV	hier à 14:54	21 Mo	Vidéo QT
DocInstallationAkif	aujourd'hui à 08:29	18,7 Mo	Pages
RapportFinal_Akif	25 déc. 2024 à 04:41	4,6 Mo	Pages
FichierDeTest	19 déc. 2024 à 08:19	--	Dossier
test.c	14 déc. 2024 à 12:10	119 octets	C Source
test.py	13 déc. 2024 à 08:25	14 octets	Python Source
text.php	13 déc. 2024 à 08:30	Zéro octet	PHP Script
text.txt	12 déc. 2024 à 08:34	7 octets	Format texte
serveur.py	25 déc. 2024 à 01:13	3 ko	Python Source



## 2. Démarrage du serveur

Le démarrage du serveur est assez trivial. Il suffit de lancer une commande directement dans le terminal.

La commande est la suivante :

```
python3 serveur.py <HOST> <PORT>
```

**HOST** : Ici on mettra l'**adresse IP**, ou on souhaite que notre serveur se connecte. Souvent, on applique `0.0.0.0` pour pouvoir accepter l'ensemble des connexions.

Personnellement, j'ai opté pour l'adresse `127.0.0.1` qui est une interface virtuelle créée par le système d'exploitation. Mais cela n'a d'importance dans le lancement du programme si on le spécifie comme il se le doit.

**PORT** : Ici, on mettra le **port**, sachant qu'il y en a 65536, certains sont utilisés comme le port 21,22,80, et j'en passe.. Cependant, cela va tout de même jouer sur la connexion entre le serveur et le client.

Exemple : `python3 serveur.py 127.0.0.1 4200`

Le serveur est maintenant en attente de **connexions entrantes**. Vous verrez un message indiquant que le serveur est en écoute. Il faut aussi vérifier que le port est bien disponible.

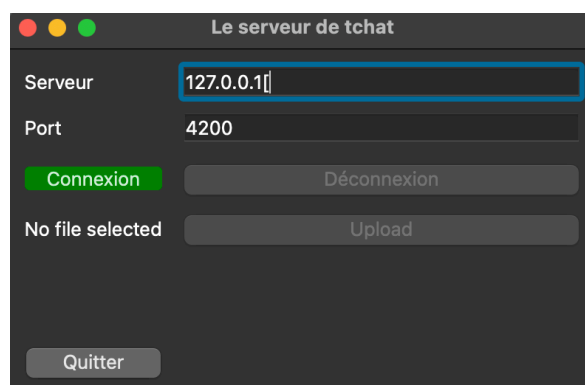
## 3. Démarrage du Client

À présent, nous allons voir comment démarrer le client depuis le terminal. Le lancement du client est plus simple que celle du serveur. Veuillez a bien lancé le serveur avant de vouloir lancer le client. Sur un autre terminal, exécuter la commande est la suivante :

```
python3 client.py
```



Constatez le résultat :



Comme vous pouvez le constater, l'interface graphique s'est bien lancé. À présent, il faut remplir les champs qui nous sont demandés

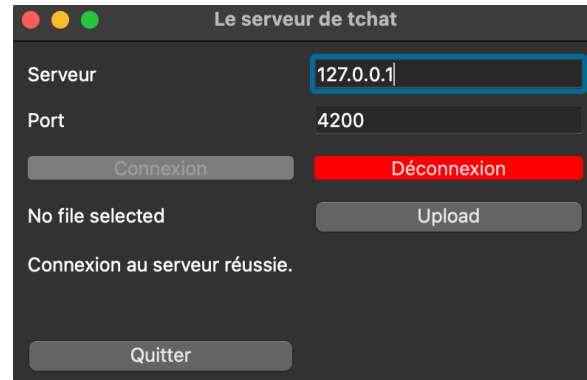
Bon, sur l'image que vous voyez ci-dessus, les champs sont déjà pré-rempli avant même qu'on se connecte au serveur.

Bien sur, on peut les modifier à notre souhait.

Un bouton « **Connexion** » fait alors son apparition pour pouvoir se connecter au serveur.

Les boutons « **Déconnexion** » et « Upload » sont grisés car indisponible, à moins que vous souhaitez vous déconnecter avant même d'atterrir sur un serveur, ou envoyez des données dans le vide.

Maintenant que les champs sont remplis, connectons-nous au serveur :



Maintenant, c'est le bouton « Connexion » qui devient grisé. Cependant, les boutons grisés avant sont maintenant disponibles à l'utilisation. La connexion au serveur est **établie** et est affichée dans l'espace dédié à l'affichage sur l'interface graphique.

#### – Connexion au serveur réussie. –

Si l'on revient au serveur, il nous a envoyé quelques petites informations pour nous prévenir la présence d'un client sur son serveur.

```
Aucun client connecté. En attente d'une connexion...
Une connexion est en cours...
Le client d'IP ('127.0.0.1', 61075) s'est connecté
Test 1
Test 2
La connexion a prit 0.0005 seconde(s) à s'établir avec le client.
```

Pour commencer, le serveur attend une connexion :

#### – Aucun client connecté. En attente d'une connexion... –

Ensuite, le serveur nous indique qu'une connexion est en cours :

**– Une connexion est en cours... –**

Enfin, nous pouvons constater que le client est bien connecté et qu'il a utilisé son port 61075 pour se connecter :

**– Le client d'IP ('127.0.0.1', 61075) s'est connecté –**

Les 'tests' 1 et 2 sont juste des éléments qui m'ont aidé à programmer la ligne suivante :

**– La connexion a prit 0.0005 seconde(s) à s'établir avec le client. –**

Cette ligne indique le temps qu'a prit le client à se connecter au serveur.

**C'est tout ce que vous devez savoir sur la connexion entre le serveur et le client.**

À présent, lisez le 'Rapport final' pour comprendre comment fonctionne le programme, dans lequel je vais vous expliquer l'architecture globale, le choix des technologies, les instructions pour déployer et tester le système, les défis rencontrés, les solutions pour les parer et les améliorations possibles.

