

SAE 3.02 : Guide d'utilisation du serveur

1) Introduction

Bienvenue dans ce guide d'utilisation du serveur de tchat de la SAE3.02. A travers ce guide vous apprendrez à installer le serveur, comment l'utiliser correctement ainsi que son fonctionnement interne. Le développement de ce serveur de tchat en parallèle du client de tchat aura pris environ 20h. La difficulté principale rencontrée durant le développement a été la découverte des threads qui sont des fonctions pouvant tourner simultanément avec le script principal, le plus difficile étant de faire communiquer ces fonctions entre elles.

En espérant que ce guide vous soit utile, bonne lecture !

2) Installation

Base de données SQL

Sous Windows, téléchargez la version suivante de mysql installer et laissez-vous guider par l'installation. Pour ne pas avoir à modifier le serveur de tchat, lorsque mysql installer vous demande un identifiant et un mot de passe, entrez « root » comme identifiant et « toto » comme mot de passe. (Si vous choisissez vos propres identifiants, pensez à modifier leur valeur dans le fichier python du serveur de tchat à la ligne 10 : « sql_user » et « sql_mdp »).

Une fois tous les services mysql installés (notamment mysql community server et mysql client) le serveur sql sera lancé et vous pourrez vous y connecter en ligne de commande : `mysql -u root -p`

Récupérez le fichier sae302-dump.sql :

<https://github.com/BaptisteBC/R309/blob/master/SAE3.02/sae302-dump.sql>

Dans l'invite de commandes, tapez la commande suivante : `mysql.exe -u root -p < sae302-dump.sql`

Cela créera toute la base de données nécessaire au bon fonctionnement du serveur de tchat. A savoir qu'il existe un utilisateur « Admin » et son mot de passe « Admin ». Ce dernier ayant accès à tous les salons.

Serveur sous Windows

Commencez par installer la dernière version de Python sur votre machine.
<https://www.python.org/ftp/python/3.12.1/python-3.12.1-amd64.exe>

Sur la première fenêtre de l'installateur, cochez la case « Add python.exe to PATH », cela simplifiera la suite. Cliquez ensuite sur « Install Now » qui vous installera votre version de python ainsi que la commande `pip` dont nous aurons besoin plus tard.

Une fois l'installation de python complète, ouvrez l'invite de commandes (`Windows` > `cmd` > Invite de commandes). Afin de vérifier la présence de python, exécutez la commande `python.exe`

Vous devriez observer un environnement de travail similaire :

```
Python 3.12.1 (tags/v3.12.1:2305ca5, Dec 7 2023, 22:03:25) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> |
```

Cela confirme la bonne installation de python. Vous pouvez quitter cet environnement avec la combinaison de touches `Ctrl + Z` puis `Entrée`.

Vous devez ensuite installer la bibliothèque python mysql-connector qui permet la connexion à la base de données. Dans l'invite de commandes, tapez *pip install mysql-connector-python*

Une fois cette commande exécutée, vous devriez obtenir un résultat similaire à celui-ci :

```
Installing collected packages: mysql-connector-python
Successfully installed mysql-connector-python-8.2.0
```

Une fois cette installation correctement réalisée, téléchargez le script du serveur de tchat via le lien GitHub suivant : <https://github.com/BaptisteBC/R309/blob/master/SAE3.02/Serveur/serveur2.0.py>

Copiez le chemin d'accès au fichier serveur2.0.py et retournez dans l'invite de commandes pour taper la commande suivante : *python.exe votre\chemin\d'accès\serveur2.0.py*

Éléments modifiables

Vous pouvez modifier le script afin de le faire correspondre au mieux à vos configurations, notamment si vous avez déployé le serveur sql sur une autre machine. Les éléments modifiables dans le script sont l'adresse IP du serveur, le port, le maximum de client, l'identifiant au serveur sql, le mot de passe du serveur sql et l'adresse IP du serveur sql. Ces variables ont des valeurs par défaut qui fonctionnent si le serveur et la base de données sont sur la même machine.

Table des matières

SAE 3.02 : Guide d'utilisation du serveur	1
1) Introduction.....	1
2) Installation.....	1
Base de données SQL	1
Serveur sous Windows	1
Éléments modifiables	2
3) Lancement du serveur.....	4
1 ^{ère} étape : Authentification/Inscription	4
2 ^{ème} étape : Ecoute des clients	5
3 ^{ème} étape : Commandes administrateur	5
4) Schéma relationnel de la base de données.....	6

3) Lancement du serveur

Une fois le script lancé, deux lignes doivent s'afficher :

```
C:\Users\Elève>python.exe C:\Users\Elève\Desktop\R309\SAE3.02\Serveur\serveur2.0.py
Démarrage du serveur
Admin :
```

La première indique le lancement du script principal, à savoir la création de la connexion ouverte aux clients ainsi que les commandes administrateur à laquelle correspond la deuxième ligne « Admin : ».

A chaque connexion d'un client, le serveur engage un script d'écoute en parallèle de son script principal. Ce script d'écoute permet la réception des messages de chaque client, la vérification des permissions relatives aux salons, l'authentification et l'inscription et enfin la distribution de tous les messages aux clients connectés.

1^{ère} étape : Authentification/Inscription

Lorsque la connexion avec un client est effectuée, ce dernier reçoit le message « Client connecté ! » qui s'affiche sur sa fenêtre de dialogue avec le serveur. Le serveur de son côté lance un script d'authentification qui va attendre les informations envoyées par le client afin de les vérifier et de pouvoir réellement lancer le tchat.

Trois actions peuvent alors avoir lieu :

- **Authentification** : Lorsque le bouton « Authentification » est pressé du côté du client, le serveur reçoit un message « auth » qui lance la séquence d'authentification. Le serveur récupère l'identifiant et le mot de passe du client et commence par vérifier si l'identifiant existe dans la base de données. Si ce n'est pas le cas, il envoie un message au client pour l'inviter à réessayer avec un identifiant différent ou essayer de s'inscrire. Si l'identifiant existe dans la base de données, le serveur va vérifier que le mot de passe associé à l'identifiant soit le même que celui envoyé par le client. Si c'est le cas, le serveur accepte la connexion avec le client et lance son script d'écoute. A savoir : le serveur vérifie que le client ne soit pas banni (temporairement ou définitivement) et les scripts du tchat ne sont pas lancés si c'est le cas.
- **Inscription** : Lorsque le bouton « Inscription » est pressé du côté du client, le serveur reçoit un message « inscrire » qui lance la séquence d'inscription. Le serveur récupère l'identifiant et le mot de passe du client et commence par vérifier si l'identifiant existe dans la base de données. Si l'identifiant existe déjà, alors le serveur envoie un message au client pour l'inviter à s'identifier. Dans le cas contraire, le serveur inscrit le client avec son identifiant et son mot de passe dans la base de données. Il initialise aussi les permissions de salon (seul le « Général » étant autorisé par défaut). Une fois ce processus terminé, le serveur démarre le script d'écoute pour le bon fonctionnement du tchat.
- **Fin de connexion** : Cette action survient lorsque le client clique sur le bouton « Quitter », cela permet de signaler au serveur la fin de la connexion et donc qu'il n'a plus besoin de lancer le script d'écoute ni d'attendre de nouvelles informations d'authentification.

2^{ème} étape : Ecoute des clients

Une fois la connexion et l'authentification/inscription des clients complètes, le serveur lance son script d'écoute qui recevra tous les messages que le client lui enverra. Avant de retransmettre les messages le serveur vérifie que le client ait la permission d'écrire dans le salon. Si ce n'est pas le cas, le serveur envoie un message au client pour lui indiquer qu'il n'a pas la permission d'écrire dans le salon et un message s'affiche en ligne de commandes du serveur pour lui indiquer que le client lui demande la permission.

Si le client a la permission d'écrire dans le salon, ce dernier commence par enregistrer le message dans la table « journal » de la base de données, puis il redirige le message du client vers toutes les connexions en cours. Cela permet aux clients présents dans le même salon que les autres de voir tous les messages de ce salon.

De même pour l'authentification/inscription si le client clique sur le bouton « Quitter » alors le serveur reçoit un signal et arrête d'écouter les messages du client qui ferme sa connexion.

3^{ème} étape : Commandes administrateur

En parallèle des programmes précédents, le serveur lance un script de commandes administrateur symbolisé par la ligne « Admin : ». Quatre commandes sont actuellement disponibles :

- **kill** : envoie un signal à tous les clients qui sont forcés de fermer leur connexion. Le serveur termine ensuite tous ses scripts, ferme sa propre connexion et met fin proprement à son propre script.
- **ban utilisateur** : Cette commande permet de bannir définitivement l'utilisateur saisi en argument de la commande. Si l'identifiant est introuvable, le serveur vous le signale en ligne de commande. Sinon la valeur de la colonne « banned » est placée à True dans la table « login » de la base de données. Si le client est connecté au moment où il est banni, le serveur lui envoie le signal de fermer sa connexion. Si le client essaie de se reconnecter il sera informé de son bannissement dans la fenêtre de dialogue avec le serveur au moment de son authentification. (Le bannissement en fonction de l'adresse IP n'est pas fonctionnelle et donc elle n'est pas présente dans cette version du serveur)
- **unban utilisateur** : A l'inverse du ban, la commande unban permet de suspendre le bannissement d'un utilisateur. Le serveur place la valeur de la colonne « banned » à False, ce qui permettra au client de pouvoir se connecter à sa prochaine authentification.
- **kick utilisateur** : Cette commande est similaire à la commande ban mais le bannissement est temporaire (une heure). Au moment de l'authentification du client, le serveur vérifie si l'identifiant n'est pas banni. Si c'est le cas, il vérifie alors si le bannissement n'est pas temporaire. Si c'est le cas, le serveur effectue un calcul en ajoutant une heure à l'heure de bannissement et si le résultat est une heure déjà dépassée, alors le serveur place la valeur « Banned » à False et permet au client de s'identifier et d'accéder au tchat.

4) Schéma relationnel de la base de données

