

IUT COLMAR

# Installation Serveur SAE302

Sous Windows

Timothée GASSER RT212  
20/12/2023

## Table des matières

1. Introduction.....	1
2. Serveur MySQL .....	1
1. Téléchargement serveur MYSQL.....	1
2. Import de la BDD .....	3
3. Programme Python serveur .....	4
1. Import des paquets .....	4
2. Exécution du programme .....	5
4. Programme Python client.....	5
1. Import des paquets .....	5
2. Exécution du programme .....	5

### 1. Introduction

Ce document vous permettra d'installer le serveur de messagerie de A à Z sur un système d'exploitation Windows. Pour les distributions Linux, le principe reste sensiblement le même. Avant de commencer, il est nécessaire d'avoir comme prérequis un environnement de développement adapté à Python avec un interpréteur. Pour ma part, j'utilise PyCharm.

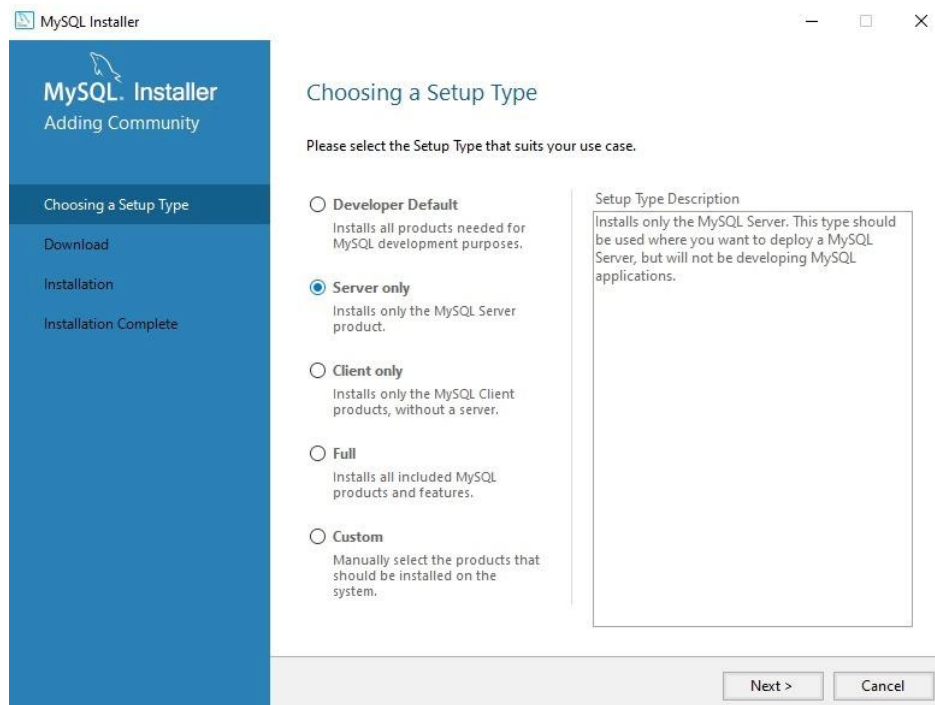
Il faudra également que vous récupériez le projet sur GitHub à cette [adresse](#). Il est conseillé de lire le fichier README pour bien comprendre l'architecture des fichiers et d'autres points annexes.

Dans ce document, je vous présente un serveur MySQL, qui est la solution la plus simple et légère à héberger. Si vous souhaitez approfondir l'administration et avoir une meilleure visibilité sur l'évolution des tables, notamment les logs et l'historique des conversations, je vous conseille d'utiliser phpMyAdmin.

### 2. Serveur MySQL

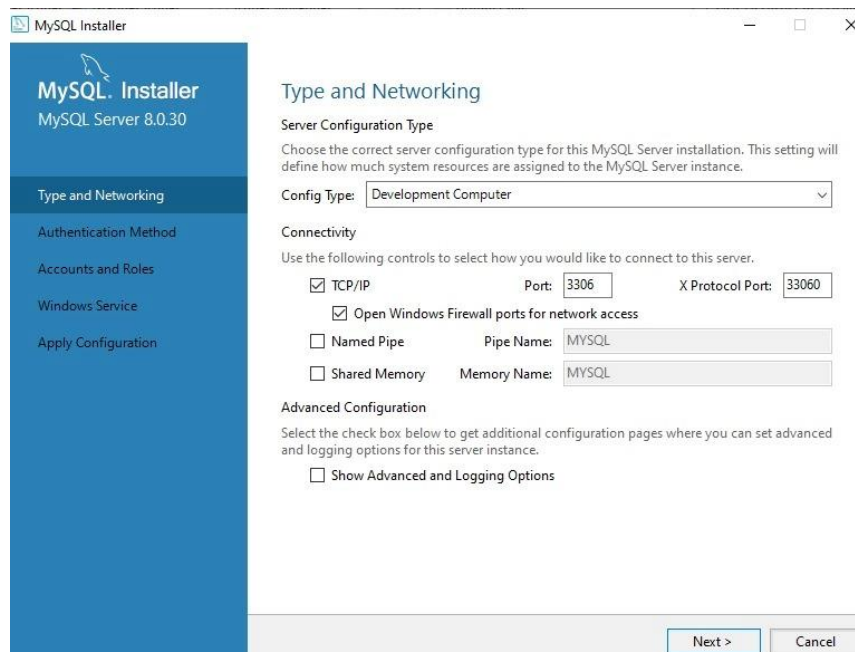
#### 1. Téléchargement serveur MYSQL

Vous pouvez directement télécharger le programme d'installation via le lien suivant [ici](#). Une fois exécuté, vous devez sélectionner "Serveur seulement", puis cliquer sur le bouton Suivant. Ensuite, cliquez sur Exécuter.

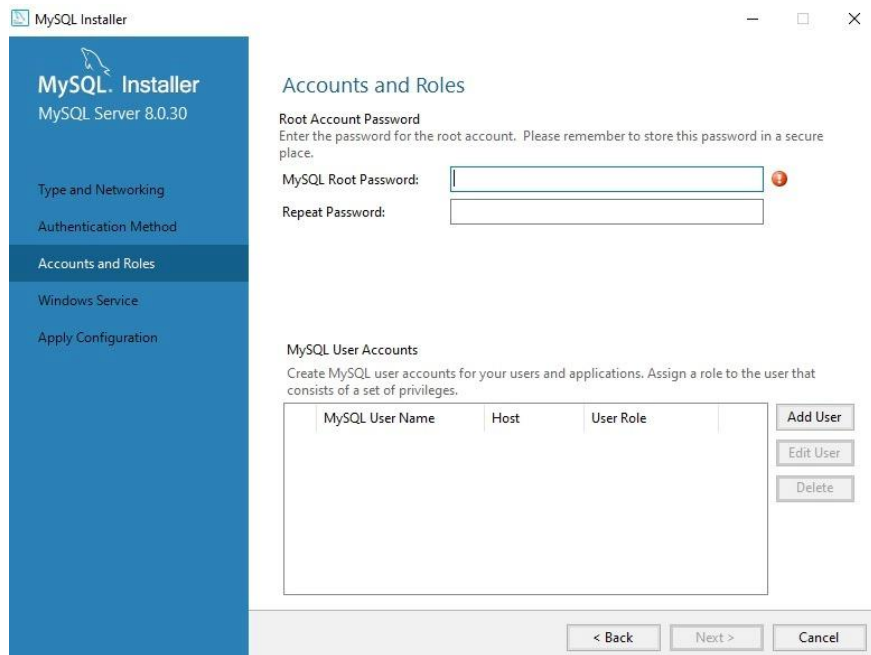


À partir de MySQL, l'installation va se lancer, cela peut prendre quelques minutes. Une fois terminée, appuyez sur Suivant.

Vous arriverez sur une page de configuration du serveur où vous pourrez notamment choisir de modifier le port associé à MySQL. Appuyez sur Suivant pour passer à la suite.



Sélectionnez " Use Strong Password" puis cliquez sur Suivant. Vous arriverez sur cette page :



Ici, vous devez définir le mot de passe pour l'utilisateur "root" qui aura tous les droits sur votre serveur MySQL. Choisissez un mot de passe fort et assurez-vous de l'enregistrer dans votre gestionnaire de mots de passe pour ne pas l'oublier. Ensuite, appuyez sur Suivant, puis à nouveau sur Suivant et enfin sur Exécuter, Finish. L'Installation est terminée !!!

## 2. Import de la BDD

Pour importer la base de données, assurez-vous d'avoir téléchargé le dépôt GitHub. Une fois extrait, repérez le chemin depuis la racine du fichier SAE302.sql. Il se trouve dans sar302/SAE/Serveur/SAE302.sql. Vous pouvez également en profiter pour modifier le nom d'utilisateur et le mot de passe utilisés par le programme. Pour cela, éditez le fichier SAE302.sql et adaptez la ligne :

```
CREATE USER 'toto'@'%' IDENTIFIED BY 'toto';
```

Une fois cela réalisé, lancez le « mysql command line client », connectez-vous avec le mot de passe donné précédemment.

Ensuite, exécutez la commande « SOURCE C:/Votre\_chemin/SAE302.sql ; ».

En exécutant cette commande, vous créez la base de données « SAE302 », un utilisateur pour le programme (avec des privilèges réduits pour des raisons de sécurité). Cet import crée également toutes les tables puis insère des données dans deux d'entre elles :

Salon pour créer le salon général (accessible à tous les utilisateurs), et le salon admin (soumis à validation), ainsi que le salon salle\_de\_pause (accessible sur demande).

Et la table utilisateur, pour créer un utilisateur administrateur, par défaut le login est 'root' et le mot de passe est 'admin'.

Voilà la BDD est installée !

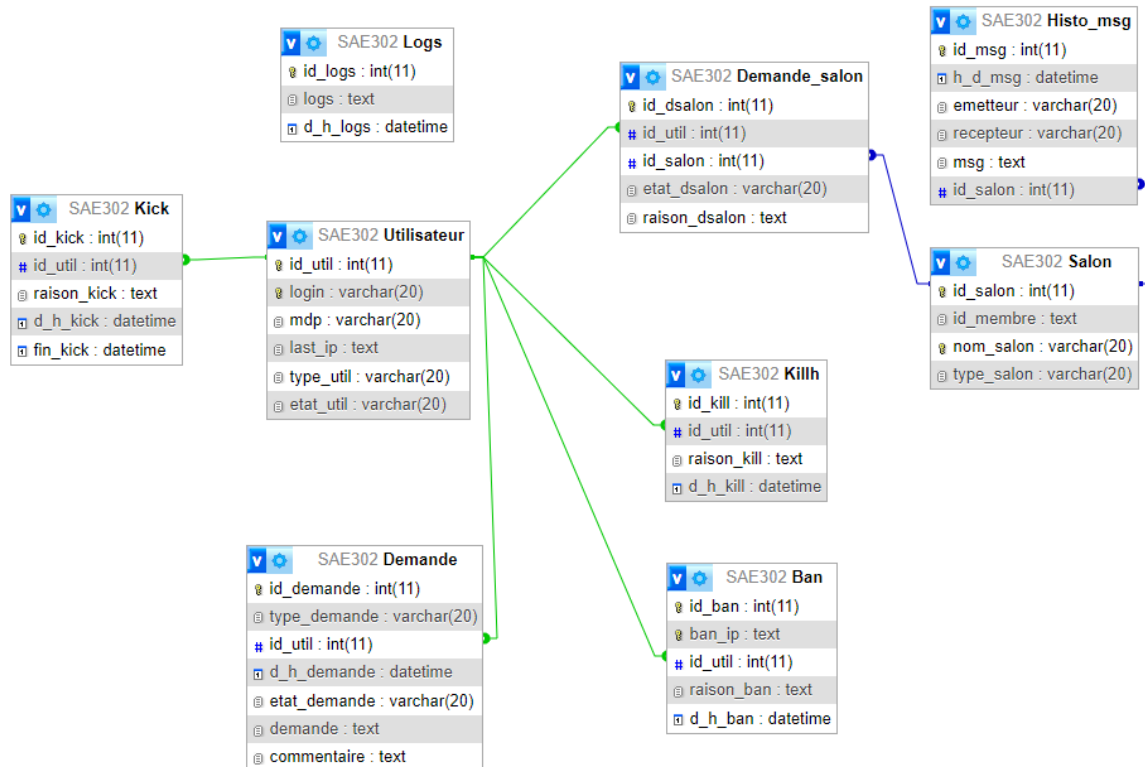


Figure 1 UML SAE302

### 3. Programme Python serveur

#### 1. Import des paquets

Pour que le programme fonctionne, il a besoin de quelques paquets. L'utilisation d'un environnement virtuel (venv) est recommandée :

```
pip3 install socket threading mysql-connector-python datetime
```

Attention à ne pas oublier de rajouter PyQt6 si vous prévoyez d'utiliser le client sur la même machine.

## 2. Exécution du programme

Avant d'exécuter le programme, pensez à modifier l'adresse IP du serveur MySQL, le port et les identifiants s'ils ont été modifiés. Cette modification s'effectue au niveau du fichier `administration.py` :

```
def connect_to_db():
    """Établit une connexion à la base de données..."""
    try:
        db_connection = mysql.connector.connect(
            host="***.***.***.***", #ip serveur mysql
            port="***", #port serveur mysql
            user='toto',
            password='toto',
            database='SAE302'
        )
```

Une fois les modifications apportées, vous pouvez lancer le programme `SRV.py` avec votre interpréteur Python. Une fois lancé, il vous demandera le port d'écoute ; entrez un port non utilisé et appuyez sur "Entrée". Votre serveur est désormais fonctionnel et prêt à accueillir des clients.

## 4. Programme Python client

### 1. Import des paquets

Pour que le programme fonctionne, il a besoin de quelques paquets. L'utilisation d'un environnement virtuel (venv) est recommandée :

```
pip3 install socket threading PyQt6
```

### 2. Exécution du programme

Pour lancer le client, utilisez votre interpréteur Python pour exécuter soit le client avec interface graphique : `SAE302/SAE/client/Client.py`

Ou alors le client sans interface : `SAE302/SAE/client/Client_CLI.py`

L'avantage du CLI est qu'il permet d'exploiter pleinement les capacités du serveur. L'interface graphique manque de fonctionnalités. Si vous vous connectez en tant qu'administrateur à l'interface graphique, vous serez directement redirigé vers l'interface en ligne de commande.