

**DocString pour l'application**  
**de Kevin Lang RT 211**

<b>1. Serveur.py</b>	<b>2</b>
<b>2. Client.py</b>	<b>2</b>
<b>3. Graphique.py</b>	<b>3</b>

Dans ce DocString pour la SAE 302 je vais expliquer toutes les fonctions que j'ai créé pour ce projet, et leurs utilités.

## 1. Serveur.py

Ce script met en place un serveur de chat. Il écoute les connexions entrantes, gère les messages de chat entre les clients, et intègre des fonctionnalités de gestion des utilisateurs et des canaux.

create\_database\_connection :

- Etablir une connexion avec la base de données

check\_access :

- Vérifie si un utilisateur a accès à un canal spécifique

add\_access :

- Ajoute l'autorisation d'accès à un canal pour un utilisateur dans la base de données

ServerWindow (Classe) :

- Interface graphique du serveur
  - initUI : Configure l'interface utilisateur du serveur
  - stop\_server : Arrête le serveur lorsqu'on clique sur le bouton

start\_server :

- Initialisation et démarrage du serveur de chat
  - broadcast : Envoie un message à tous les clients se trouvant dans le même canal
  - handle\_client : Gère la connexion individuelle de chaque client
  - disconnect\_client : Gère la déconnexion d'un client sur le serveur
  - receive : Accepte les nouvelles connexions clients sur le serveur

## 2. Client.py

Ce script définit la logique du côté client pour se connecter au serveur de chat et gérer la communication.

ChatClient(Classe) :

- Cette classe gère la connexion client et la communication avec le serveur de chat
  - receive : Écoute et gère les messages entrants du serveur
  - send : Envoie un message au serveur

### 3. Graphique.py

Ce script crée l'interface graphique du client pour le chat. Il permet à l'utilisateur de se connecter, de s'inscrire, de choisir un canal de chat et d'envoyer/recevoir des messages.

#### create\_database\_connection :

- Établit une connexion à une base de données MySQL

#### create\_user :

- Enregistre un nouvel utilisateur dans la base de données

#### check\_user :

- Vérifie si les informations de connexion fournies par un utilisateur sont correctes

#### appliquer\_feuille\_de\_style :

- Applique un thème de couleurs et d'affichage spécifique à l'application

#### ChannelSelectionWindow(Classe) :

- Interface pour la sélection d'un canal de chat
  - join\_channel : Ouvre la fenêtre de chat (ChatWindow) pour le canal sélectionné

#### ChatWindow(Classe) :

- Fenêtre de chat pour un canal spécifique
  - send\_message : Envoie le message saisi par l'utilisateur au serveur
  - display\_message : Affiche les messages reçus dans la fenêtre de chat
  - change\_channel : Permet à l'utilisateur de changer de canal

#### RegisterWindow(Classe) :

- Interface pour l'inscription des utilisateurs
  - register\_account : Enregistre un nouvel utilisateur avec les informations saisies

#### LoginWindow(Classe) :

- Interface pour la connexion des utilisateurs
  - check\_credentials : Vérifie les identifiants et ouvre la fenêtre de sélection de canal (ChannelSelectionWindow) en cas de succès
  - open\_register\_window : Ouvre la fenêtre d'inscription (RegisterWindow)

#### ChatClient(Classe) :

- Gère la connexion réseau et la communication avec le serveur de chat
  - receive : Écoute en continu les messages entrants du serveur
  - send : Envoie un message au serveur