

Documentation du Projet de messagerie local

Sommaire

1. Introduction
2. Utilisation
 - Serveur
 - Client
3. Partie Technique
 - Serveur
 - Configuration
 - Gestion des Clients
 - Communication
 - Client
 - Connexion au Serveur
 - Communication
4. Améliorations Possibles
 - Chiffrement des Messages
 - Création/Connexion avec des Comptes
 - Gestion d'Amis avec Chat Privé

1. Introduction

Le projet de chat implémente un serveur de chat et un client basé sur une architecture client-serveur. Il utilise GTK pour l'interface utilisateur graphique. Ce document fournit une documentation détaillée pour l'utilisation et la compréhension du projet.

2. Utilisation

2.1 Serveur

Le serveur est responsable de la gestion des connexions clients et de la diffusion des messages entre eux.

-Compilation du Serveur:

```
Last login: Thu Jan 25 18:37:03 on ttys002
ravin@Air-de-Ravin projetS1 % gcc -o server server.c `pkg-config --cflags --libs gtk+-3.0` -I/opt/homebrew/opt/openssl@3/include -L/opt/homebrew/opt/openssl@3/lib -lssl -lcrypto
```

```
gcc -o server server.c `pkg-config --cflags --libs gtk+-3.0`
-I/opt/homebrew/opt/openssl@3/include -L/opt/homebrew/opt/openssl@3/lib -lssl -lcrypto
```

-Exécution du Serveur:

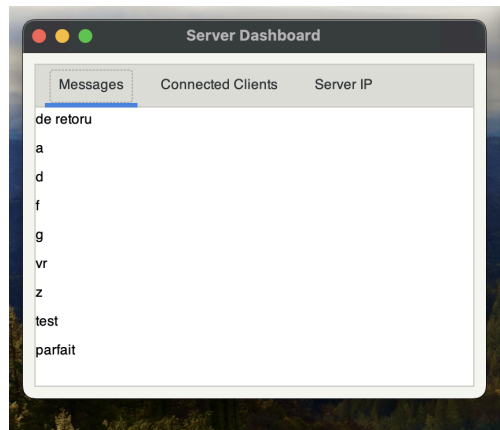
si on ne précise pas le fichier de config par défaut c'est le fichier config.ini qui sera selectionner

```
ravin@Air-de-Ravin projetS1 % ./server
Server IP: 192.168.1.26
Port: 12345
Messages file: messages.txt

(server:75009): Gtk-CRITICAL **: 18:39:02.435: gtk_box_pack: assertion 'GTK_IS_WIDGET (child)' failed
```

./server [config_file]

- Fenêtre GTK:



2.2 Client

Le client permet à l'utilisateur d'envoyer et de recevoir des messages du serveur.

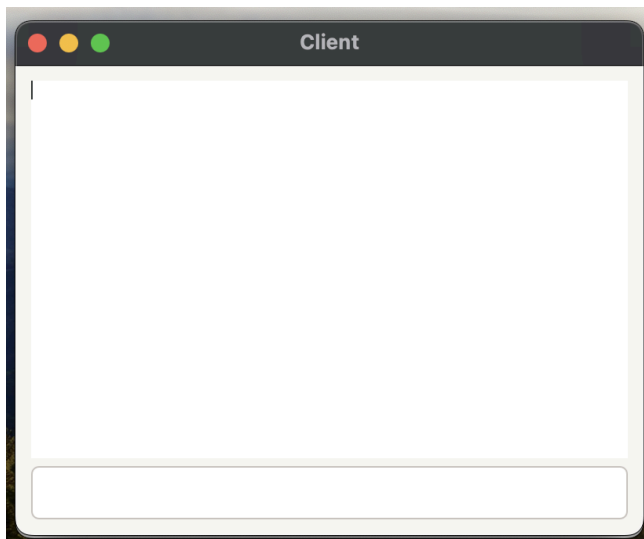
- Compilation du Client:

```
gcc -o client1 client1.c `pkg-config --cflags --libs gtk+-3.0`
```

- Exécution du Client:

```
ravin@Air-de-Ravin projetS1 % ./client1
```

Ouverture de la fenêtre gtk:



3. Partie Technique

3.1 Serveur

3.1.1 Configuration

Le serveur lit la configuration depuis un fichier `config.ini`, spécifiant l'adresse IP, le port, et le fichier de messages.

3.1.2 Gestion des Clients

- Connexion des Clients:

Le serveur accepte les connexions des clients et les associe à des threads distincts.

- Déconnexion des Clients:

En cas de déconnexion d'un client, le serveur met à jour la liste des clients connectés.

3.1.3 Communication

- Réception des Messages:

Le serveur reçoit les messages des clients et les diffuse à tous les autres clients.

- Enregistrement des Messages:

Les messages sont enregistrés dans un fichier `messages.txt` pour une persistance.

3.2 Client

3.2.1 Connexion au Serveur

- Initialisation:

Le client initialise une socket et se connecte au serveur.

- Interface Utilisateur:

Le client utilise GTK pour créer une interface graphique avec une zone de messages et un champ de saisie.

3.2.2 Communication

- Envoi des Messages:

Le client envoie les messages au serveur, qui les redistribue à tous les clients connectés.

- Réception des Messages:

Un thread séparé gère la réception des messages en continu.

4. Améliorations Possibles

4.1 Chiffrement des Messages

Ajouter un chiffrement des messages pour assurer la confidentialité des communications entre le serveur et les clients.

4.2 Création/Connexion avec des Comptes

Permettre aux utilisateurs de créer des comptes et de se connecter avec des identifiants pour une expérience plus personnalisée.

4.3 Gestion d'Amis avec Chat Privé

Intégrer une fonctionnalité de gestion d'amis avec la possibilité d'initier des discussions privées entre utilisateurs.