

RAPPORT PROGRAMMATION SYSTÈME



VADIER Simon

SHEIKH ARSHAD Muhammad Saad

RAPPORT

1. Contexte

Notre projet a pour but de mettre en place une messagerie instantanée entre des clients connectés sur un serveur. Ces clients, une fois connectés, doivent en premier lieu enregistrer un pseudonyme et seront affectés à un ID. Cet ID permet aux différents clients, par le biais d'un menu, de s'ajouter en ami et ensuite de pouvoir démarrer un salon de discussion privé. Ce salon peut-être quitté en entrant "fin".

2. Organisation des programmes

Le répertoire comprend tous les modules et librairies utiles pour la compilation. Ils s'appuient sur ceux utilisés lors du cours de PSE à l'exception des fichiers `ligne.h`, `ligne.c` et `dataspect.h` qui ont été modifiées. Dans le dossier "Projet" on retrouve le `makefile`, `serveur.c` et `client.c` qui ont eux aussi comme base le cours de PSE.

Les fonctionnalités des fonctions de `serveur.c` :

Affiche d'abord un message pour que la personne se connecte. Ecriture dans une base la liste des personnes en ligne

- accepte les connexions
- création d'un pool de thread attendant des connexions
- Ajout d'un ami :
 - Ecriture de la liste des personnes connectées (Lecture des logion de tous les connectés)
 - Ecriture dans une base l'ajout d'ami
- Démarrer une conversation :
 - Affichage de la liste des amis connectés

Les fonctionnalités de `client.c` :

- Affiche un menu :
 - Ajouter un ami
 - Démarrer une conversation
 - Quitter
- Permet l'envoi et la réception d'un message pour un client.

3. Difficultés

Une des grandes difficultés se trouve dans la partie messagerie. En effet, on avait pour but de créer un salon de discussion privé entre deux clients, de ce fait il faut utiliser des fonctions non-bloquantes pour la lecture des messages qu'un client souhaite envoyer pour ne pas empêcher la réception d'un quelconque message.

Les messages du premier client s'envoient sur le serveur et celui-ci les renvoie sur le second client mais ce dernier ne les reçoit pas. Le programme s'arrête sur un core-dumped.