



# Projet Minichat

Section client-serveur

Hathoute Hamza

**Groupe GH**

Département Sciences du Numérique - Première année  
2020-2021

# Contenu

<b>1</b>	<b>Choix effectués</b>	<b>3</b>
1.1	Serveur . . . . .	3
1.2	Client . . . . .	3
<b>2</b>	<b>Debug</b>	<b>3</b>

# 1 Choix effectués

## 1.1 Serveur

Le serveur dispose d'une liste *fd\_to\_id* qui, à chaque descripteur de fichier client vers serveur, l'associe à l'identifiant du client. Ceci optimise la recherche du nom du client puisque dans la boucle du serveur, le message est associé au descripteur C2S.

Un autre choix était d'utiliser une variable globale *curcount*, celle si contient l'identifiant du dernier participant. Lorsque la liste des participants est pleine (c'est à dire *curcount* == *MAXPARTICIPANTS*), l'ajout d'un nouveau participant provoque la réorganisation de la liste des participants en décalant les participants vers le bas de la liste...

J'ai choisi d'ouvrir les FIFO de lecture (*ecoute*, *c2s*) en mode **O\_RDONLY** puisque sur certains systèmes, *select* retourne directement si le descripteur est ouvert en lecture mais plus de processus l'ouvre en écriture ([StackOverflow](#)).

## 1.2 Client

Pour le client, pas de choix particuliers...

# 2 Debug

Le code contient une fonction *dprint(const char \*format, ...)* qui est utilisée pour le debug; enlever le commentaire de *define DEBUG* affichera des messages qui sont plus ou moins utiles.