



**POLYTECHNIQUE
MONTREAL**

**LE GÉNIE
EN PREMIÈRE CLASSE**

Réseaux Informatiques
INF3405

Laboratoire 2

Présenté par:
Ibrahima Séga Sangaré (1788085)
et
Khalil Benani (1566707)

Soumis à:
Bilal Itani

22 février 2018

Introduction

Dans ce deuxième laboratoire du cours INF3405, l'objectif était d'avoir une introduction à l'utilisation des sockets ainsi qu'aux threads via une application de filtre d'images. L'application a été développée dans le langage Java en respectant des contraintes spécifiques. Le but de l'application était de pouvoir établir une connexion serveur client à l'aide d'une adresse IP, un port ainsi qu'un identifiant et un mot de passe. Lorsque le client parvient à se connecter, il peut alors envoyer une image au côté serveur qui applique le filtre Sobel sur l'image et la lui renvoie en lui donnant les informations sur l'emplacement du nouveau fichier. Le client peut se déconnecter et se reconnecter en gardant le même identifiant ou créer un autre compte. Les comptes doivent être sauvegardés par le serveur à cet effet.

1.Présentation

La solution que nous avons utilisée se basait sur le concept de threads et de sockets. D'abord, le serveur doit être lancé et on doit vérifier qu'une adresse IP correcte est entrée. Cela a été implémenté avec des patterns qui restreignent le format d'adresse IP que l'utilisateur peut entrer. Ensuite, le numéro de port a été limité entre les valeurs 5000 et 5050 comme il a été précisé dans l'énoncé. Dès que ces conditions sont remplies, un socket serveur est créé et associé à un thread de traitement d'image qui est exécuté à chaque fois qu'un client se connecte. Du côté client, les étapes de validation de l'adresse IP et du port sont identiques au serveur. La validation de l'identifiant et du mot de passe se fait en envoyant les données entrées par l'utilisateur au serveur. Le serveur crée un nouvel utilisateur s'il n'existait pas déjà dans la base de données. Si l'identifiant existe déjà, l'utilisateur doit avoir entré le bon mot de passe pour se connecter ou il lui sera demandé d'entrer à nouveau ses informations.

Lorsque le client se connecte, un menu s'affiche pour qu'il entre le nom d'une image qui se trouve dans le répertoire actuel ou bien qu'il quitte l'application. Lorsque le nom d'une image est entré, on vérifie que l'image existe bel et bien, sinon un message d'erreur est retourné. Puis le nom de la nouvelle image est demandé. Le contenu de l'image est transformé en tableau de bytes et envoyé au serveur étant donné que ce sont essentiellement les bytes qui sont modifiés par le filtre. Des messages s'affichent en temps réel pour indiquer le client des traitements. Le serveur reçoit les bytes que le client envoie lui envoie et y applique le filtre en indiquant quel client a fait la demande avec la date, l'identifiant du client, son adresse IP, et le nom de l'image reçue. Le client reçoit alors l'image et est avertie de l'emplacement du nouveau fichier. Le client peut continuer à envoyer des images ou se déconnecter.

2. Difficultés rencontrées

Étant donné que le projet et beaucoup d'autres cours sont donnés au moyen du langage c++, alors il était un peu difficile de debugger java, même si une grande documentation est disponible sur internet.

La plus grande difficulté était la manipulation des threads moyennement le langage java puisqu'on savait quoi faire mais on arrivait pas vraiment à l'implémenter de la bonne manière.

3. Critiques et améliorations

Somme toute, les requis qui sont demandés sont relativement clairs. Toutefois, il faudrait peut-être préciser de manière plus explicite ce qui est demandé par rapport à la gestion des mots de passe.

4. Conclusion

Nous avons réellement apprécié ce court projet car en dehors de fait de nous avoir permis de mettre en pratique certaines notions de cours, nous avons trouvé cela intéressant de développer une application concrète en réseautique. Sans ce projet, le cours aurait manqué cruellement de côté pratique et je trouve primordial le fait d'avoir un aperçu de ce à quoi peut ressembler le développement en réseautique. D'autre part, ce projet nous a permis d'apprendre des connaissances complémentaires à celles du cours et nous a donc obligé à sortir un petit peu de notre zone de confort pour comprendre le fonctionnement des sockets, d'un serveur et d'un client.