**GASHIMWE Aimée-Emelyne** Le 9/11/2023

**IRADUKUNDA Aimable**

**IRADUKUNDA Elvis**

**IRAKOZE Blaise**

**ISHIMWE Graciella**

**ITERITEKA Odon**

**MANIRAKIZA Belard**

FSI/TIC: BACIII

UB : CAMPUS KIRIRI

**Rapport du Projet de Programmation des services Réseaux**

Nous avons créé une plateforme de chat temps réel qui permet aux utilisateurs connectés de pouvoir communiquer entre eux via un réseau. Cette plateforme pour la créer nous avons utilisé les sockets TCP permettant d’obtenir une communication bidirectionnelle entre le client et le serveur, en nous servant du langage de programmation next.js.

Notre serveur est capable de gérer plusieurs requêtes à la fois entre autres des connexions simultanées des clients, il est capable de recevoir des messages des clients et de les diffuser à tous les utilisateurs connectés, il affiche les messages envoyés par les clients sur sa console avec les informations telles que l’heure, le nom d’utilisateur du client, etc.

L’utilisateur client se connecte au serveur en spécifiant l’adresse IP et le port du serveur.

Le client a un nom d’utilisateur unique qui est affiché avec ses messages, il est capable d’envoyer des messages au serveur pour être diffusés aux autres utilisateurs connectés et il reçoit les messages envoyés par les autres utilisateurs sur sa console.

Les fonctionnalités implémentées dans notre plateforme chat sont les suivantes :

-Authentification : Un utilisateur doit avant tout s’authentifier avec un nom d’utilisateur et un mot de passe s’il est déjà inscrit et dans le cas contraire il doit s’inscrire puis s’authentifier afin de pouvoir se connecter.

-Messagerie Privée : Le client peut envoyer des messages privés à un autre utilisateur sans que les autres ne le sachent.

-Messagerie broadcast : Un client peut envoyer un message qui va être reçu par tous les utilisateurs connectés.

-Création de groupes : Un utilisateur peut créer un groupe et ajouter ses membres, il est le seul à pouvoir ajouter des membres.

-Suppression des messages : un utilisateur peut supprimer un message qu’il a envoyé.

-Une notification de connexion et de déconnexion s’il y a un utilisateur qui se connecte où se déconnecte

-Déconnexion : Un utilisateur peut se déconnecter

Au cours de la réalisation de notre projet, nous avons rencontré différentes difficultés telles que :

* Au commencement, nous avons commencé à développer notre application en python mais cela nous semblait un peu difficile.
* Parmi les membres du groupe, il y avait certains qui n’etaient pas fort en JavaScript avec son Framework de NextJs et nous avons été obligé de commencer à zero afin qu’ils puissent comprendre les concepts de base et de connaitre comment ils peuvent développer l’application client-serveur en NextJs Framework de JavaScript.

Au cours de la réalisation de notre projet, nous avons développé nos compétences en programmation des logiciels clients-Serveur qui utilisent des services réseau pour que le client puisse communiquer avec le serveur. Et nous tenons à remercier notre professeur pour ce projet car cela nous a permis non seulement développer nos compétences mais aussi de voir le niveau de chaque membre du groupe et de pouvoir mieux s’entraider pour élever le niveau de chacun.