Chat System

Manuel d’utilisation administrateur

Pour récupérer le projet il faut le git clone à ce lien <https://github.com/emileseb/projet_coo.git>

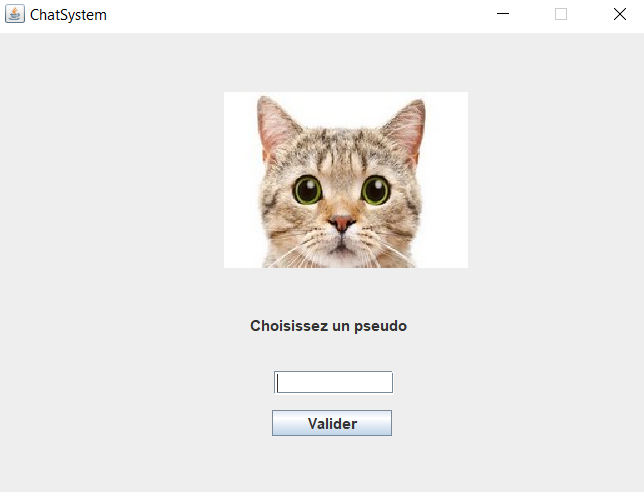
Avant de le lancer, il faut préciser sur quelle interface réseau l’application communiquera, pour un réseau local d’une entreprise ce sera l’interface sur laquelle les ordinateurs sont connectés au réseau local. L’interface par défaut est eth0. Soit il faut connecter les ordinateurs au réseau local sur eth0 soit il est possible de modifier l’interface en changeant la ligne 20 de la classe Utilisateur du package utilisateur.

Pour lancer l’application :

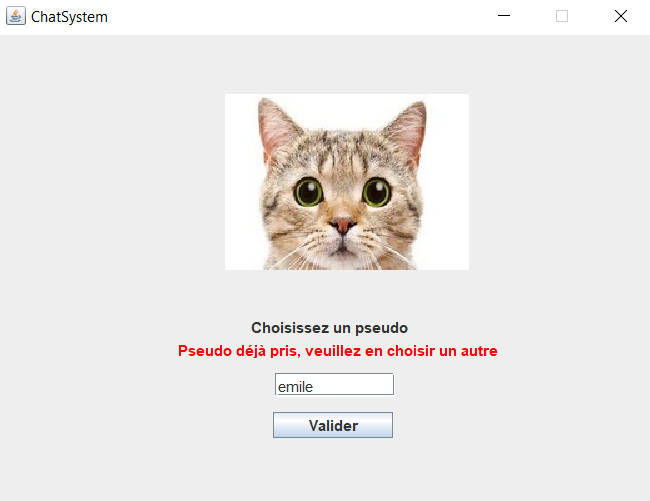
* Sous Windows il faut double cliquer sur l’exécutable nommé ChatSystem
* Sous Linux il faut exécuter la commande java -jar ChatSystem.jar
* Ou bien dans un IDE Run la classe Application.

Manuel d’utilisation utilisateur

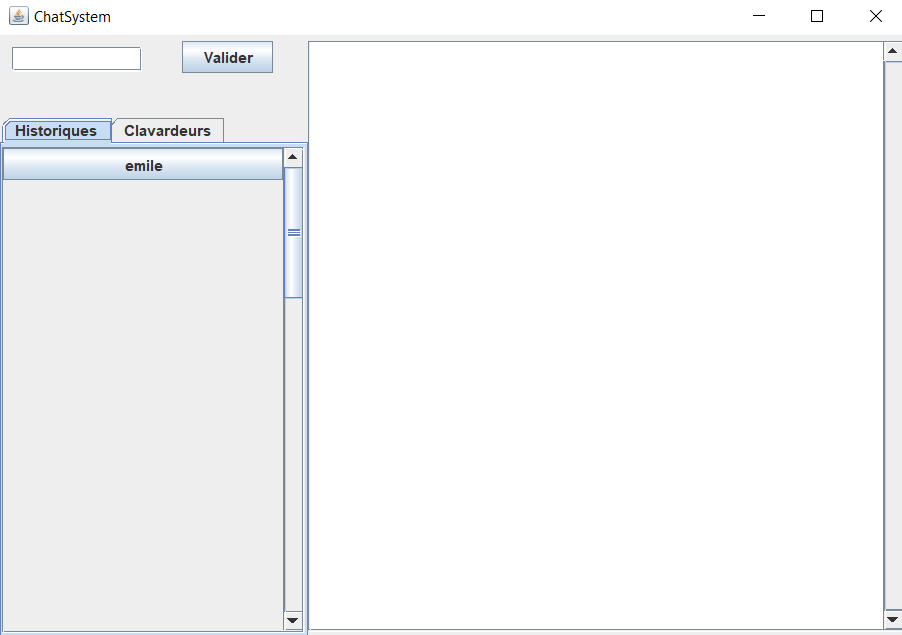
Au lancement de l’application une première fenêtre va s’ouvrir où il faudra choisir un pseudo.



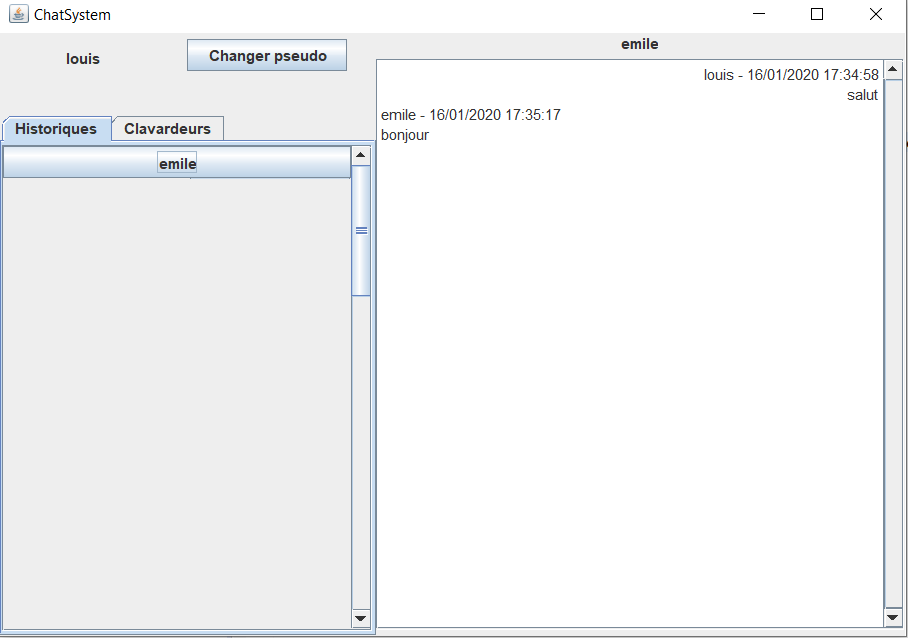
Si le pseudo est déjà pris par un autre utilisateur alors il y aura un message d’erreur indiquant à l’utilisateur de changer de pseudo.

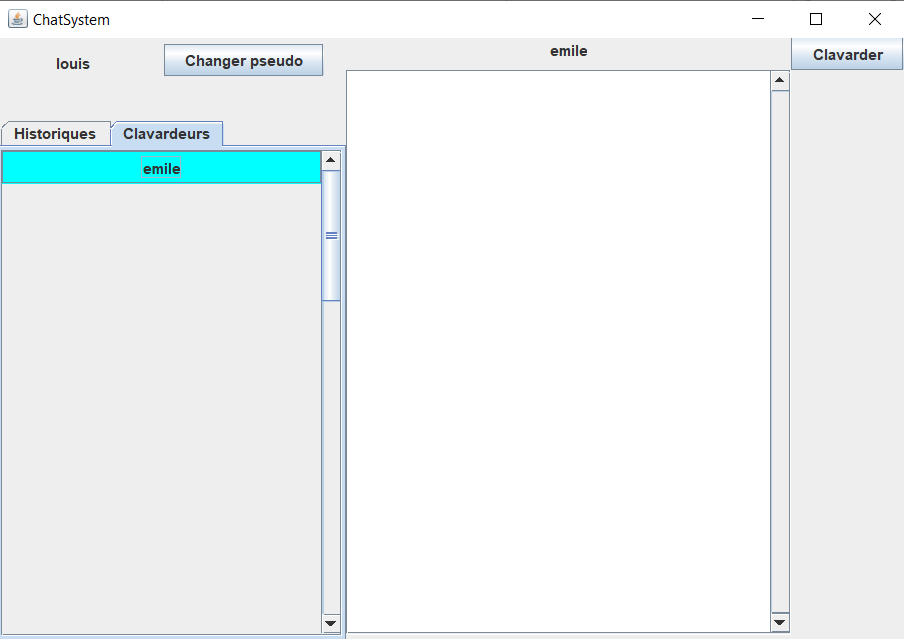


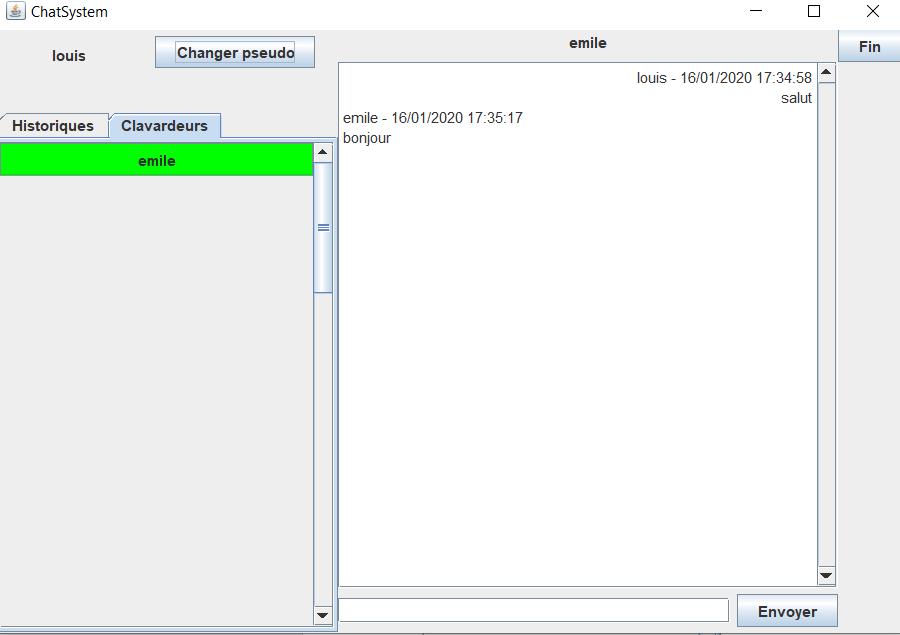
Une fois rentré une autre fenêtre s’ouvrira offrant plusieurs options.

Il sera possible de changer de pseudo à tout moment en haut à gauche de la fenêtre. Il suffit de cliquer sur le bouton « Changer pseudo » puis un bouton « Valider » apparait. Si le pseudo est déjà pris alors il y aura un message pour l’indiquer, comme sur la fenêtre d’accueil.

Il y aura deux onglets, le premier est pour pouvoir consulter les historiques des conversations à tout moment et ce même si l’utilisateur n’est pas connecté.



Le deuxième est pour voir les utilisateurs connectés sur le système. Les utilisateurs actifs seront visibles en bleu et les utilisateurs avec qui le client a des conversations en cours seront en vert.

Pour lancer une conversation il suffit de cliquer sur un utilisateur actif et de cliquer sur le bouton Clavarder. Pour fermer cette conversation il suffit de cliquer sur le bouton Fin. 

En fermant l’application toutes les conversations se ferment ce qui permet de ne pas à avoir à cliquer sur le bouton Fin.

Il se peut que par moment que l’utilisateur reçoive des notifications de changements de pseudos dans ce cas la liste change avec les pseudos corrects. De même si un utilisateur se déconnecte alors il n’apparaîtra plus dans la liste.

(Il se peut qu’il y ait des problèmes d’affichage, même si le changement de pseudo ou la déconnexion a été reçu par moment cela ne prendra pas effet sur l’interface graphique, il suffit de cliquer sur un onglet pour que cela prenne effet, c’est un problème qu’il reste à corriger, du certainement à l’utilisation d’onglets)

Nous utilisons TCP afin de garantir la fiabilité des messages envoyés.

Procédures de tests

Tests effectués avec deux utilisateurs connectés sur le réseau.

* Demande de clavardage de l’utilisateur 1 puis fermeture de la connexion par l’utilisateur 1 ou 2 fonctionne. Quand l’un des utilisateur ferme et rouvre son agent, l’historique est conservé.
* Changement de pseudo pendant une conversation fonctionne, un pseudo déjà pris ne peut être repris et un message est affiché, les sessions de clavardages ne se ferment pas.
* Fermeture de l’application de l’utilisateur 1 avec des conversations ouvertes (unilatérale), l’utilisateur 2 reçoit bien la fermeture de connexion puis la déconnexion de l’utilisateur 1
* Ouvertures et fermetures d’agents pendant que l’agent « Test » est ouvert. On remarque bien qu’après rafraichissement (clic sur n’importe quel onglet) la liste se met à jour.
* Vérification de l’exigence des ressources grâce au gestionnaire de tâches.