

CS-GY 9053 I2 – Introduction to Java - Fall 2021

Final Project: Chat software-zz3523

Description

This program allows multiple users to create an account in the group and chat privately with each other.

How to use

Server:

- Download the entire file and open the config.properties.
 - url is where the database located, to create the database you could use e.g. Navicat to import the database by excuting the .sql file. [create database](#)
 - username and pwd is the username and password for your MySQL useraccount, and you may need to change it to your own MySQL user.
 - mykey is used to encode the passwords of our users
- launch serverConn.java in the server folder [create database](#)
- Error may occur when connecting to the database. If that happens, you could try to reconfigure the MySQL authentication method to Legacy authentication method (retain MySQL 5.x compatibility)

Client:

- Download the entire file and open the configConnection, and change HOST to your server's IP address.
- launch loginInterface.java in the clientView folder [create database](#)

Login

You could login with some default username and password already stored in the database. E.g.\ Username: Frank, Password frank;\ Username: Lucy, Password lucy;\ ... [create database](#)

Registration

Click the create an account button, then you could create an account with your own username and password. [create database](#)

Chat

Double click the friend name label to open a chat. * If both of clients open the chat, they can send messages to each other by clicking the send button.(Enter does not work here.) [create database](#) * If A is trying to send a message to B that is offline, the server will notify A that your target B is offline. [create database](#) * If both of clients are online, but B has not opened that chat window, B's friendname will turn into blue to notify her that she has a new message from A. [create database](#)

Exit

Close the client before shutting down the server

Implementation

Classes

User

This class formats the information of users including id, username, password, online status

LeaveMessage

This class formats the information when the receiver is not able to receive the message from sender.

client

userConn

Methods to send messages to the server. And receive the messages from server return the result to user interface. Methods includes login, register and get friend list.

clientView

chat

Chat Interface. Create threadChatClient to receive message and shut down the thread when closing the window. Send message to server when send button is pressed.

friendlist

Friend List Interface. Based on userConn. Double Click to launch a chat with other. When closing the window, tell the server that the user is now offline.

loginInterface

Login Interface. Based on userConn.

register

Registration Interface. Based on userConn.

server

serverConn

Based on the message type sent by the user the server will operate the database using methods from serverDataManage and send back messages to users.

serverDataManage

Manage the user data.

serverMessageDataManage

Manage the unread message data.

utility

configConnection

Set the server IP address and port number

connectionUtil

Tools to establish the connection between server and clients

dataBaseUtil

Tools to connect to the database

Message

Formats the message sent between server and clients

PropertiesUtil

Referenced from Internet to get values from config.properties

statusType

The message types sent between server and clients to distinguish which action to execute.

threadChatClient

If receive a chat message, then append it to JTextArea. If receive a target offline message, append the content in message from server to JTextArea.

threadChatServer

Thread lauched when user open a chat window (message type CHAT_CONN). Then depends on whether the target user is online(server has a socket with target username) or open the chat window(server has also a thread in the inverse direction) or target is offline(server does not have anything), the server will notify the target or send the message to notify the user that the target is offline.

threadRefreshFriendlist

Depending on the received message, the client side thread will refresh the friendlist when user is getting online, offline, registration, or leaving a message.