

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



MẠNG MÁY TÍNH

Bài Tập Lớn 1:

XÂY DỰNG ỨNG DỤNG CHAT DÙNG GIAO THỨC TCP/IP

Giáo viên hướng dẫn: Ths. Nguyễn Mạnh Thìn

Lớp: L08

DANH SÁCH THÀNH VIÊN

- | | | |
|----------------------|---|---------|
| 1. Đặng Văn Dũng | – | 1710853 |
| 2. Nguyễn Trung Tính | – | 1713521 |
| 3. Cao Đăng Dũng | – | 1710849 |

TP. HỒ CHÍ MINH - THÁNG 11/2019



MỤC LỤC

1. ĐỊNH NGHĨA VÀ MIÊU TẢ CÁC CHỨC NĂNG CỦA ỨNG DỤNG	2
1.1 Đăng kí tài khoản	2
1.2 Đăng nhập	2
1.3 Gửi và nhận tin nhắn	2
1.4 Gửi nhận file	2
1.5 Xem số người đang online trong hệ thống	2
1.6 Xem tên của người đang online.....	2
1.7 Hệ thống có sử dụng emoji để người dùng có thể thể hiện biểu cảm của mình	3
1.8 Hệ thống cho phép kết bạn với người khác	3
2. ĐỊNH NGHĨA CÁC GIAO THỨC SỬ DỤNG CHO TỪNG CHỨC NĂNG .	3
3. THIẾT KẾ CHI TIẾT ỨNG DỤNG	4
2.1 Công nghệ sử dụng.....	4
2.2 Kiến trúc ứng dụng	4
2.3 Class diagram	5
4. ĐÁNH GIÁ KẾT QUẢ	8
5. HƯỚNG DẪN SỬ DỤNG.....	8
6. MÃ NGUỒN	12

1. ĐỊNH NGHĨA VÀ MIÊU TẢ CÁC CHỨC NĂNG CỦA ỨNG DỤNG

1.1 Đăng kí tài khoản

- Ứng dụng cho phép người dùng có thể đăng kí tài khoản, thông tin đăng kí bao gồm tên người dùng và mật khẩu.
- Một người dùng có thể đăng kí cho mình nhiều tài khoản.

1.2 Đăng nhập

- Ứng dụng cho phép người dùng đăng nhập để dễ dàng quản lí và hiển thị thông tin. Khi người dùng đăng nhập vào hệ thống thì thông tin về user và password sẽ được gửi lên server để kiểm tra. Nếu xác minh đúng thì cho phép người dùng có thể chat với mọi người trong hệ thống. Nếu xác minh sai thì xuất lỗi ra để người dùng biết.

1.3 Gửi và nhận tin nhắn

- Sau khi đăng nhập thì người dùng có thể gửi và nhận tin nhắn với bất kì người nào trong hệ thống nếu người đó online. Khi click vào tên của người muốn nhắn tin thì ở chỗ người đó sẽ hiện ra 1 hộp thoại thông báo có người muốn nhắn tin với bạn. Nếu muốn nhắn tin thì người đó có thể đồng ý và nếu không thì ngược lại.
- Sau khi đã kết nối với nhau thì 2 người có thể chat qua lại với nhau.

1.4 Gửi nhận file

- Khi đang ở cửa sổ chat với một người khác trong hệ thống thì bạn có thể gửi file nếu sự đồng ý của người đó.

1.5 Xem số người đang online trong hệ thống

- Khi bạn là người quản lí server thì bạn có thể biết được hiện nay trong hệ thống đang có bao nhiêu người online.

1.6 Xem tên của người đang online

- Sau khi đăng nhập thì bạn có thể xem tên của những người đang online trong hệ thống. Vì thế nếu muốn chat với người nào thì bạn có thể click

vào chờ đợi sự đồng ý của người đó.

1.7 Hệ thống có sử dụng emoji để người dùng có thể thể hiện biểu cảm của mình

- Khi ở cửa sổ chat thì sẽ có vùng emoji để người dùng có thể sử dụng.

1.8 Hệ thống cho phép kết bạn với người khác

- Khi bạn muốn một người nào đó trong hệ thống làm bạn của mình thì có thể gửi lời mời kết bạn với người đó và sau khi đăng nhập thì những người là bạn của mình sẽ được hiển thị ở một vùng khác, giúp cho việc kết nối nhanh chóng và dễ dàng hơn.

2. ĐỊNH NGHĨA CÁC GIAO THỨC SỬ DỤNG CHO TỪNG CHỨC NĂNG

Nội dung protocol	Mục đích sử dụng	Ghi chú
<SESSION_REQ> ... </SESSION_REQ>	Yêu cầu khởi tạo 1 chat session từ phía client, ClientID là tên người dùng (tên tài khoản) được lưu trong database của server. NumPort là cổng mà Client có thể giao tiếp với Server.	Được sử dụng khi user đăng nhập vào hệ thống.
<SESSION_KEEP_ALIVE>... </SESSION_KEEP_ALIVE>	Kiểm tra xem hiện nay trong hệ thống có bao nhiêu user đang online	Được sử dụng để hiển thị danh sách những người đang hoạt động
<PEER_NAME> userID </PEER_NAME>	Dùng để hiển thị và kiểm soát tên user	Được sử dụng khi muốn hiển thị tên user
<PORT>NumPort</PORT>	Dùng để hiển thị và kiểm soát cổng giao tiếp giữa Client và Server hoặc Client và Client	Được sử dụng khi muốn hiển thị cổng kết nối giữa user và client
<STATUS>userID</STATUS>	Dùng để hiển thị tình trạng của Server	
<SESSION_DENY />	Từ chối khi người dùng nhập sai mật khẩu hoặc không có tài khoản khi login vào hệ thống	
<SESSION_ACCEPT></SESSION_ACCEPT>	Chấp nhận cho người dùng đăng nhập vào hệ thống khi nhập đúng mật khẩu của mình	
<CHAT_REQ>userID</CHAT_REQ>	Gửi yêu cầu chat đến một người khác đang online	userID là tên user của người yêu cầu



<IP>numIP</IP>	Dùng trong việc quản lí IP của user nào đó trong hệ thống	numIP là IP của user
<CHAT_DENY />	Từ chối chat session	
<CHAT_ACCEPT />	Chấp nhận chat session	
<CHAT_MSG>msg</CHAT_MSG>	Sử dụng khi gửi tin nhắn từ user này qua user kia	Msg là nội dung msg mà user muốn gửi
<PEER> peerData</PEER>	Xác định peer do Server trả về cho Client	peerData là peer của user gồm name, port, IP
<FILE_REQ>fileName</FILE_REQ>	Yêu cầu gửi file	FileName là tên file
<FILE_REQ_NOACK />	Từ chối yêu cầu gửi file	Sử dụng khi người nhận không muốn nhận file vì một lí do nào đó
<FILE_REQ_ACK></FILE_REQ_ACK>	Chấp nhận yêu cầu gửi file	
<FILE_DATA_BEGIN />	Bắt đầu gửi file sau khi nhận được sự đồng ý	
<FILE_DATA>fileData</FILE_DATA>	Nội dung file cần chuyển	
<FILE_DATA_END />	Kết thúc quá trình gửi file	
SERVER_ONLINE	Xác định trạng thái của User là online	
SERVER_OFFLINE	Xác định trạng thái của User là offline	
<REGISTER>username</REGISTER>	Gửi yêu cầu đăng kí lên Server	Username là tên tài khoản mà người dùng nhập vào.
<LOGIN_REQ>username,password</LOGIN_REQ>	Gửi yêu cầu Login lên server	Username và password là tên tài khoản và mật khẩu do người dùng nhập vào.
<LOGIN_ACCEPT>	Server phản hồi login thành công và cho người dùng vào hệ thống chat	
<LOGIN_REJECT>	Server phản hồi login không thành công	

3. THIẾT KẾ CHI TIẾT ỨNG DỤNG

2.1 Công nghệ sử dụng

- TCP Socket: Một kỹ thuật dùng để hỗ trợ lập trình các ứng dụng giao tiếp qua mạng.
- Java Swing: Sử dụng để tạo các ứng dụng ở dạng Form.

2.2 Kiến trúc ứng dụng

- Form: Ứng dụng gồm 5 form chính

Server form: Form quản lí server, hiển thị thông tin server, các request user gửi lên.

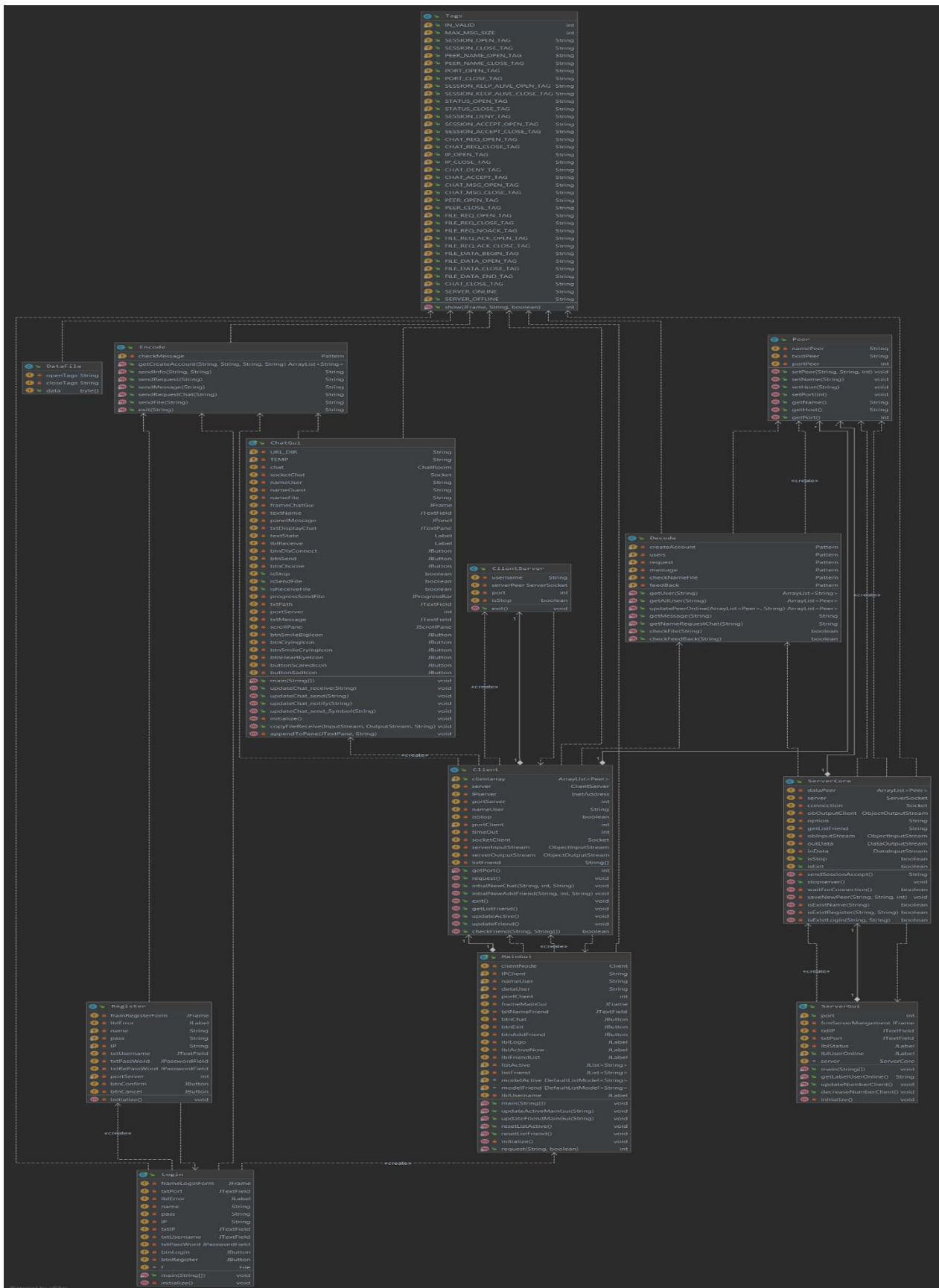
Login form: Form đăng nhập vào server. Để kết nối được, user phải cung cấp username chưa có ai đăng kí trước đó với server. Đồng thời trong Login form có thể cho người dùng chuyển qua Register Form để tạo tài khoản.

Register form: Là form cho người dùng khởi tạo tài khoản để đăng nhập vào hệ thống.

Main form: Ở đây thì hiển thị rõ 2 list. List đầu tiên là tên tất cả những người đang online trong hệ thống, list thứ hai thì hiển thị những người là bạn (và đang online trong hệ thống) của user đó.

Chat form: Khi hai user đã chấp nhận trò chuyện với nhau thì form này sẽ xuất hiện. Hiển thị thông tin cuộc trò chuyện giữa hai người.

2.3 Class diagram



- Class cho server

ServerGUI.java: sử dụng để thao tác với server như bật, tắt server. Ngoài ra còn hiển thị các thông tin server như: IP, Port, số lượng người kết nối, các protocol mà client gửi lên.

ServerCore.java: Đây là class để quản lí server, gồm đầy đủ thông tin, chức năng chính của server:

- Tạo một SocketServer.
- Quản lí danh sách các user đang online.
- Tạo một thread là WaitForConnect để xử lý các request kết nối từ user. Có thể là yêu cầu đăng kí, đăng nhập, thoát ứng dụng, lấy thông tin user đang online. Nếu đăng nhập thành công, trả về cho client danh sách user đang online, nếu không sẽ trả về một protocol từ chối đăng nhập.

- **Class cho client**

MainGUI.java: GUI class để hiển thị danh sách user khác đang online, bắt sự kiện gửi yêu cầu chat của user.

ChatGUI.java: Xử lý việc chat, gửi nhận File giữa hai user. Hiển thị nội dung chat giữa hai user.

ClientServer.java: Tạo và quản lí ServerSocket của mỗi user. Xử lý request chat gửi đến từ user khác.

Client.java: Quản lí thông tin hiện tại của user. Nó sẽ tạo một thread để liên tục gửi request tới server để cập nhật danh sách user. Ngoài ra nó còn xử lý request chat tới user khác.

Register.java: Quản lí thông tin về việc đăng kí tài khoản của user.

- **Các class khác**

DataFile.java: Tạo một đối tượng File để gửi đi.

Peer.java: Tạo một đối tượng Peer có các thuộc tính: IP, port, name.

Tags.java: Định nghĩa các protocol được sử dụng trong ứng dụng.

Encode.java: Định nghĩa các phương thức user dùng để gửi request lên server.

Dedode.java: Giải mã các phương thức để lấy thông tin user, port hay ip...

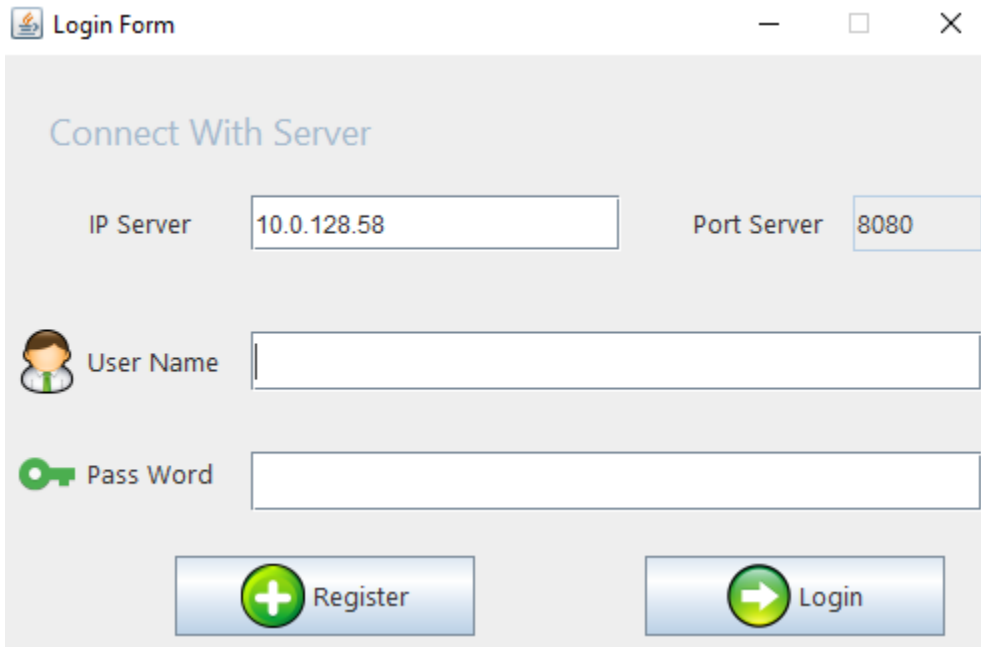
4. ĐÁNH GIÁ KẾT QUẢ

Ứng dụng được xây dựng dựa trên mô hình kết hợp giữa client-server cho việc quản lý các user và P2P cho việc trò chuyện giữa hai user với nhau.

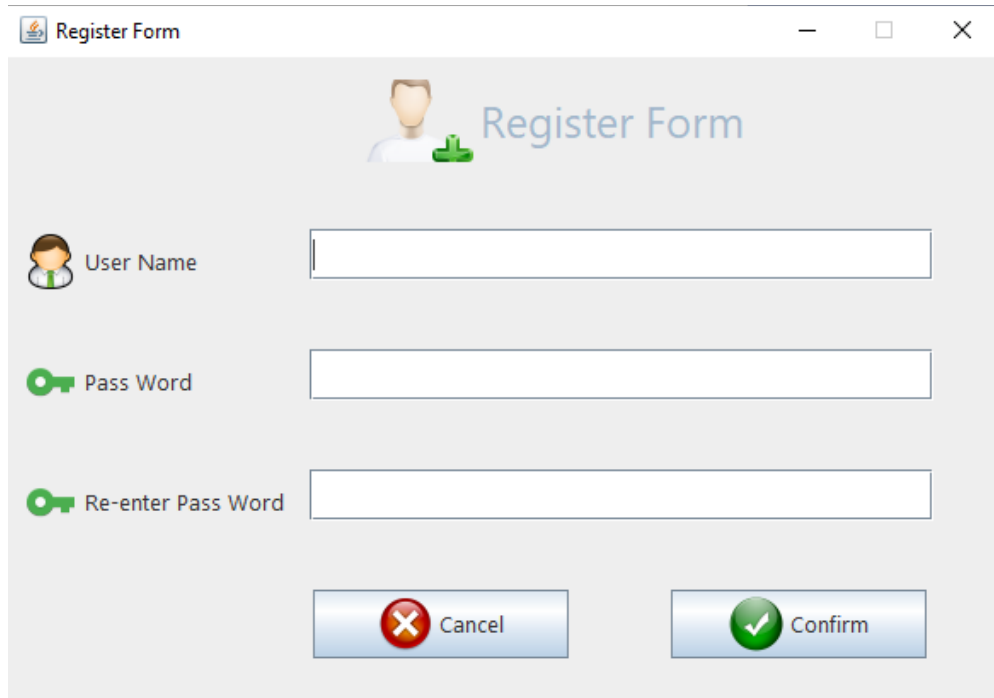
Ứng dụng có các tính năng như: chat giữa hai user, một lúc đồng thời chat với nhiều user, gửi File trong quá trình chat, kết bạn với user khác để việc kết nối được nhanh và dễ dàng hơn.

5. HƯỚNG DẪN SỬ DỤNG

- Đầu tiên thì người dùng cần có một tài khoản để đăng nhập vào hệ thống.

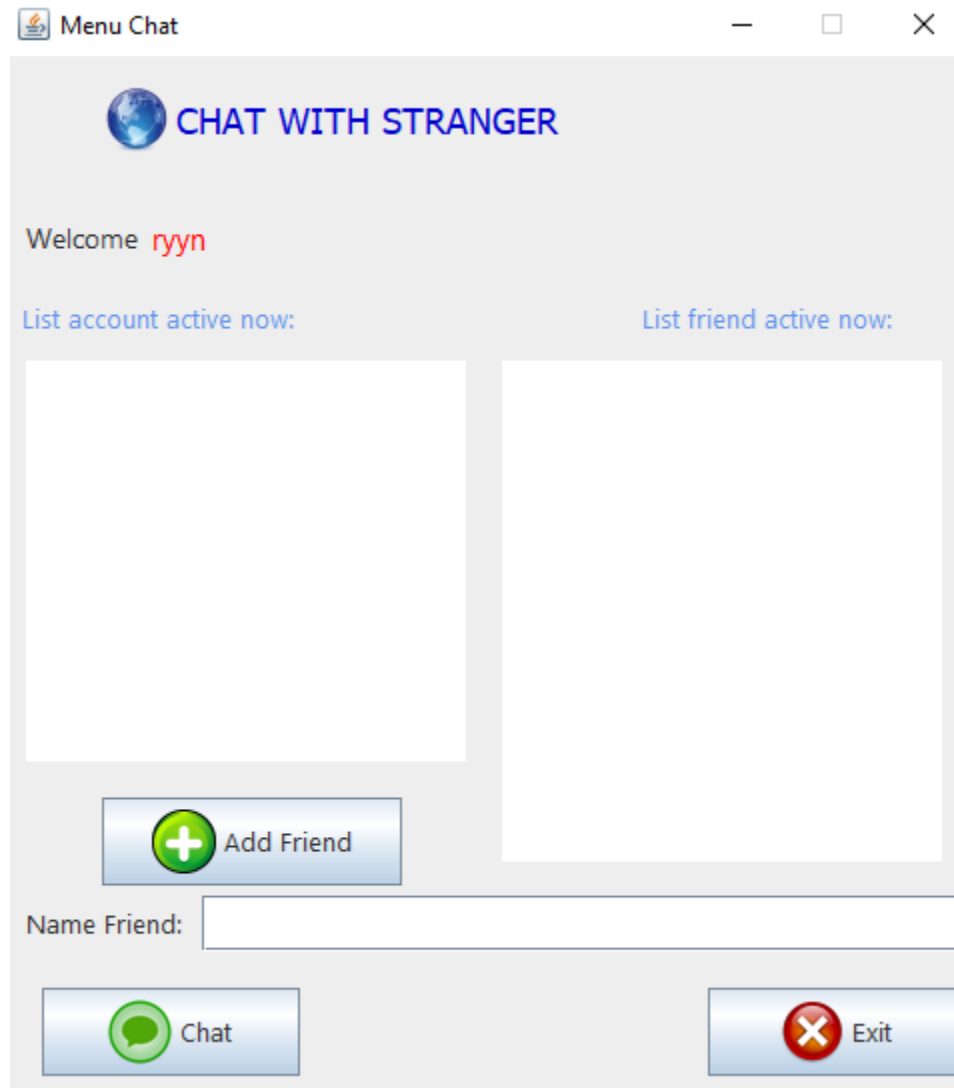


- Click vào Register để đăng kí tài khoản.
- Sau đó form đăng kí sẽ hiện ra.

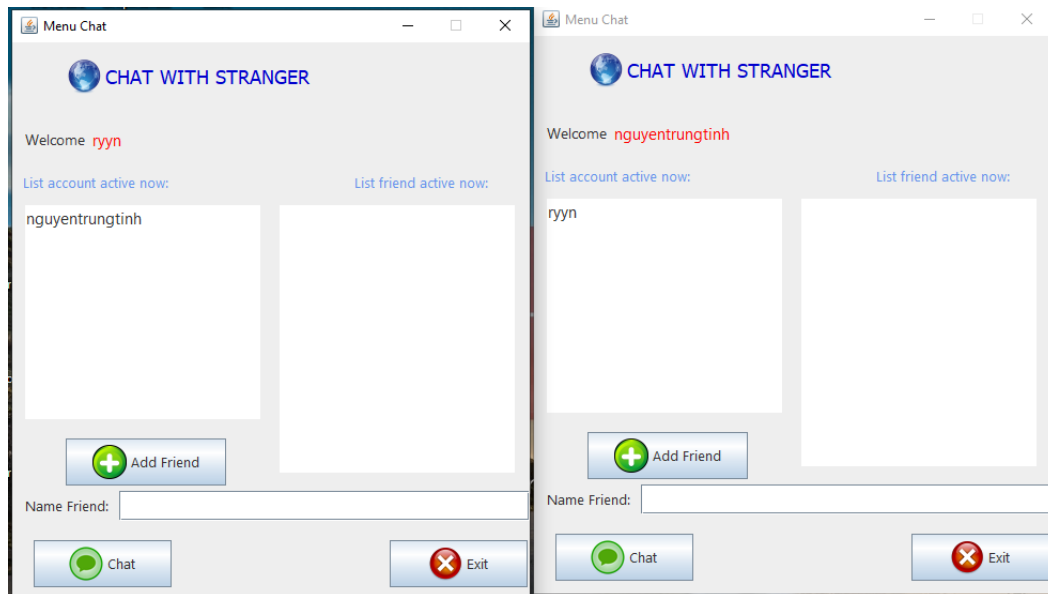


The image shows a Windows-style window titled "Register Form". Inside the window, at the top center, is a small icon of a person with a green plus sign next to it, followed by the text "Register Form". Below this, there are three input fields. The first is labeled "User Name" with a person icon to its left. The second is labeled "Pass Word" with a green key icon to its left. The third is labeled "Re-enter Pass Word" with a green key icon to its left. At the bottom of the form, there are two buttons: "Cancel" with a red circle containing a white 'X' icon, and "Confirm" with a green circle containing a white checkmark icon.

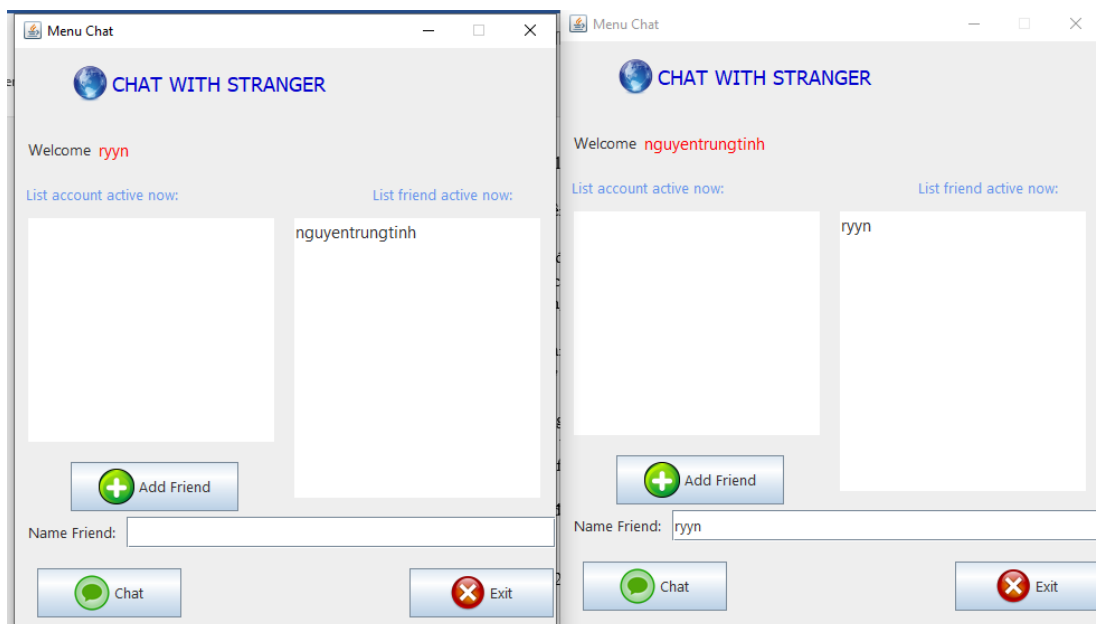
- Người dùng cần nhập đầy đủ vào 3 trường thông tin trên. Hai lần nhập mật khẩu phải trùng khớp với nhau. Nếu không hệ thống sẽ báo lỗi người dùng nhập sai.
- Sau khi đăng kí thì người dùng có thể đăng nhập vào hệ thống



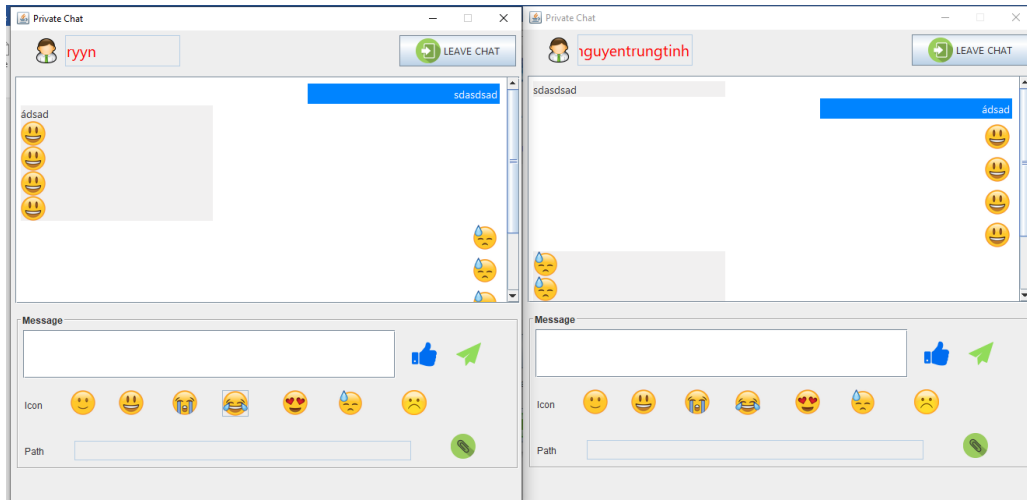
- Đây là giao diện chính của chương trình. Ở đây ta thấy 2 danh sách. Một là danh sách tất cả những người đang online trong hệ thống, 2 là danh sách tất cả những người là bạn của mình.
- Nếu muốn chat với người nào thì trước hết cần Click vào tên của người đó, sau đó ta click vào button chat ở phía dưới.



- Nếu muốn người nào đó trở thành bạn thì ta click vào tên người đó là chọn “Add friend”, sau đó chờ người đó chấp nhận thì 2 người sẽ là bạn của nhau.



- Giao diện khi 2 người là bạn của nhau.



- Giao diện chat giữa 2 người với nhau.

6. MÃ NGUỒN

Mã nguồn được lưu ở link sau: <https://github.com/trungtin961/CNass1.git>