**PHẦN I. GAME CARO**

**Mục Lục**

[**1.** **Phát biểu bài toán:** 2](#_Toc497453119)

[**1.1.** **Giới thiệu:** 2](#_Toc497453120)

[**1.2.** **Chức năng:** 2](#_Toc497453121)

[**1.3.** **Một số kỹ thuật liên quan:** 2](#_Toc497453122)

[**2.** **Phân tích thiết kế hệ thống:** 2](#_Toc497453123)

[**2.1.** **Tác nhân :** 2](#_Toc497453124)

[**2.2.** **Biểu đồ ca sử dụng :** 3](#_Toc497453125)

[**2.3.** **Biểu đồ lớp :** 3](#_Toc497453126)

[**a.** **Chức năng đăng nhập, tạo phòng chơi :** 3](#_Toc497453127)

[**b.** **Chức năng quản lý danh sách người chơi, số điểm , số trận :** 4](#_Toc497453128)

[**2.4.** **Kịch bản:** 4](#_Toc497453129)

[**2.4.1.** **Kịch bản quản lý User:** 4](#_Toc497453130)

[**2.4.2.** **Kịch bản chơi game:** 5](#_Toc497453131)

[**2.5.** **Cơ sở dữ liệu :** 6](#_Toc497453132)

[**2.6.** **Giao thức trao đổi giữa Client và Server :** 7](#_Toc497453133)

[**3.** **Cài đặt:** 10](#_Toc497453134)

[**3.1.** **Các công cụ sử dụng:** 10](#_Toc497453135)

[**3.1.1.** **Môi trường cài đặt:** 10](#_Toc497453136)

[**3.1.2.** **Các công nghệ sử dụng:** 10](#_Toc497453137)

[**3.2.** **Các lớp và hàm xử lý chương trình:** 10](#_Toc497453138)

[**3.2.1.** **Phía Client:** 10](#_Toc497453139)

[**3.2.2.** **ServerCaro:** 10](#_Toc497453140)

[**3.3.** **Một số giao diện chính:** 11](#_Toc497453141)

[**4.** **Kết luận:** 15](#_Toc497453142)

[**5.** **Tài liệu tham khảo:** 15](#_Toc497453143)

1. **Phát biểu bài toán:**
   1. **Giới thiệu:**

Cờ caro là một trò chơi khá phổ biến và lâu đời, không chỉ vì luật chơi đơn giản mà còn không đòi hỏi nhiều về công cụ.Cờ caro là trò chơi đối kháng 2 người : gồm 2 người chơi, đối thủ sẽ tìm cách chiến thắng đối thủ kia trong số hữu hạn các nước.Trò chơi kết thúc khi có một người đạt đưược 5 con liền nhau trên cùng một hàng ngang, hàng dọc hoặc đường chéo. Nếu như kín bàn cơ mà chưa ai thắng thì trò chơi kết thúc và hoà nhau. Từ nhu cầu giải trí và tiện lợi hơn, Cờ caro đã được phát triển trên máy tính, điện thoại : thi đấu đối kháng online.Trên cùng mục đích muốn đem trò chơi đến với mọi người, nhóm đã chọn chương trình chơi cờ caro đối kháng online làm đề tài của mình.

* 1. **Chức năng:**

Chương trình chơi cờ caro gồm 2 phần: Server và Client.

Các chức năng chính của chương trình:

* Quản lý người dùng ( Đăng nhập, Đăng kí thành viên )
* Quản lý điểm, danh sách người chơi ( Sắp xếp thứ hạng theo điểm, số trận)
* Cho phép chơi game caro
* Chức năng Chat ( trò chuyện ) khi chơi game
* Hiển thị thông báo kết quả game, cập nhật vào cơ sở dữ liệu.

+) *Tại Server:*

* Quản lý các kết nối đăng nhập hợp lệ của client và liên kết các client với nhau để client có thể thách đấu trực tiếp với nhau.
* Thực hiện lưu lại tài khoản khi client đăng ký tài khoản mới.
* Thông báo cho người chơi kết quả của ván chơi.

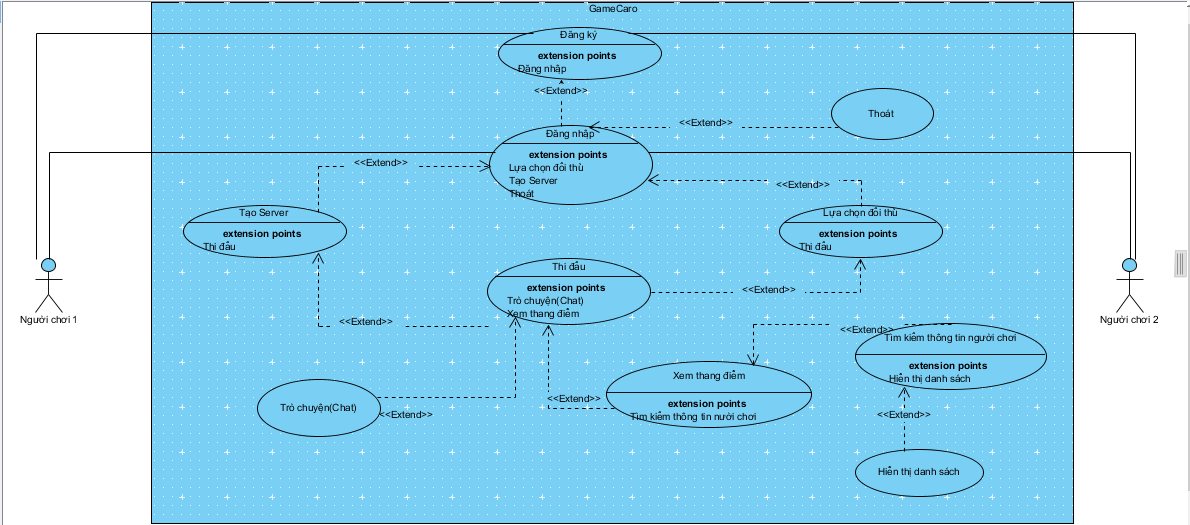
+) *Tại Client:*

* Kết nối vào server thông qua tên User và Password nếu đã có tài khoản.
* Nếu không có tài khoản thì phải tạo tài khoản với server.
* Thực hiện chơi game thông qua giao diện hiển thị trên màn hình.
* Có thể trò chuyện với đối thủ trong khi chơi game

* 1. **Một số kỹ thuật liên quan:**
* Sử dụng mô hình lập trình mạng Client/Server, giao thức TCP/IP
* Sử dụng Socket và kỹ thuật multiThread trong java
* Ngôn ngữ thực hiện : Java

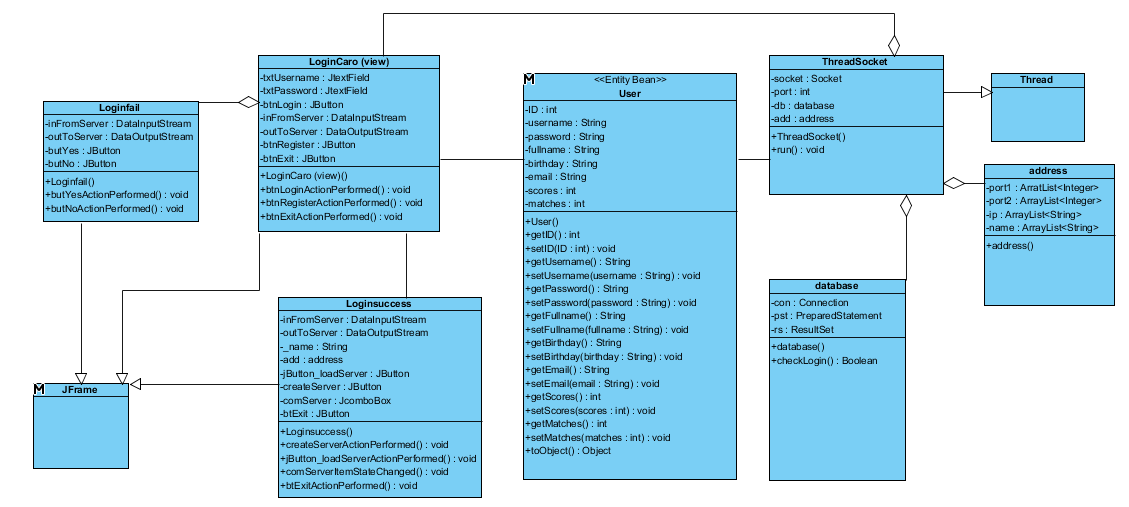
1. **Phân tích thiết kế hệ thống:**
   1. **Tác nhân :**

* Người chơi : là người sử dụng hệ thống có các thông tin được lưu trong cơ sở dữ liệu như : ID (Mã), username (Tên tài khoản), password (Mật khẩu), fullname (Tên đầy đủ ), birthday (ngày sinh), email, scores (điểm), matches (số trận)
  1. **Biểu đồ ca sử dụng :**

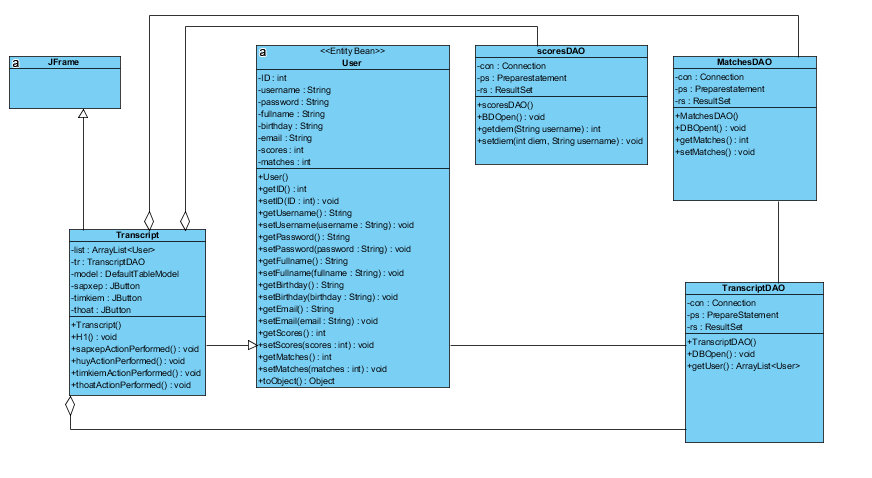


* 1. **Biểu đồ lớp :**

1. **Chức năng đăng nhập, tạo phòng chơi :**



1. **Chức năng quản lý danh sách người chơi, số điểm , số trận :**



* 1. **Kịch bản:**
     1. **Kịch bản quản lý User:**

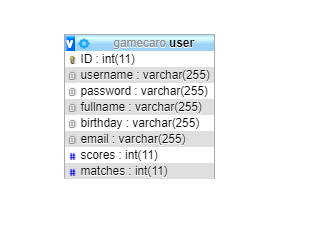
|  |  |  |  |
| --- | --- | --- | --- |
| STT | Phía Server | Phía Client | STT |
| 1 | Server chạy sẵn chờ Client kết nối vào |  | 1 |
| 2 |  | Client truy cập vào Server | 2 |
| 3 | Server nhận truy cập từ Client |  | 3 |
| 4 |  | Client nhập thông tin đăng nhập của người dùng và gửi lên server | 4 |
| 5 | Nhận thông báo từ Client gửi lên. Server sẽ truy vấn vào Cơ sở dữ liệu để xác thực Account đã tồn tại không và gửi lại kết quả? |  | 5 |
|  | 5.1. Nếu Account không tồn tại. Server gửi form Đăng kí cho người dùng | 5.1. Người chơi đăng kí tài khoản chơi ( bao gồm : username, password, full name, birthday, email ) và đăng nhập ( Bước 4 ) |  |
|  | 5.2. Nếu Account tồn tại. Server sẽ gửi form Đăng nhập thành công | 5.2. Người chơi lựa chọn trở thành server carođể đợi người chơi khác kết nối vào hoặc sẽ chọn những server caro đang chờ đợi để kết nối chơi game |  |
|  |  | 5.2.1. Nếu người chơi lựa chọn tạo server caro: Client sẽ gửi thông tin lên Server yêu cầu cung cấp port tạo ra server carochờ đợi một người khác kết nối vào game |  |
|  | 5.2.1. Nhận thông tin từ client cấp phát cho client 2 port ( 1 port để chơi game, 1 port dùng để chat) và gửi về cho client |  |  |
|  |  | 5.2.1. Nhận về 2 port từ Server và khởi chạy form chơi game sẵn sàng chờ đối thủ kết nối tới. |  |
|  |  | 5.2.2. Nếu người chơi lựa chọn chơi với những server caro đang chờ đợi kết nối game. Người chơi click vào Chọn danh sách server carođang chờ kết nối |  |
|  | 5.2.2. Nhận thông tin từ client gửi lên và gửi về client danh sách những người chơi đang chờ kết nối |  |  |
|  |  | 5.2.2. Client chọn chơi với 1 đối thủ trong danh sách người chơi. Đồng thời client sẽ gửi lên Server yêu cầu xóa người chơi đã được kết nối ra khỏi danh sách server caro đang chờ kết nối. |  |
|  | 5.2.2. Nhận thông tin của client gửi lên và xóa người chơi khổi danh sách chờ đợi kết nối |  |  |

* + 1. **Kịch bản chơi game:**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Phía Server caro | Phía Client caro | STT |
| 1 | Server caro được tạo ra với 2 port tương ứng  Khởi tạo 2 Thread với 2 port đợi chạy sẵn và chờ client game kết nối vào ( 1 port dùng để chơi game, 1 port dùng để chat ) |  | 1 |
| 2 |  | Nhận được IP và port tương ứng, kết nối vào sever caro đang chờ sẵn | 2 |
| 3 | Thiết lập kết nối với client caro |  | 3 |
| 4 | + Đánh caro tại vị trí (point) muốn đánh. Rồi gửi vị trí đó đến client caro  + Gõ thông tin muốn chat. Rồi gửi cho client caro khi trò chuyện  Kiểm tra vị trí vừa đánh xem có trùng vị trí cũ, có đủ 5 quân cơ, nếu đủ 5 quân cờ. Thông báo khi đã thằng. Sau khi thắng thiết lập lại bàn cờ chuẩn bị cho ván cờ mới |  | 4 |
| 5 |  | + Nhận vị trí quân cờ từ server caro. Repaint lại  + Nhận chuỗi trò chuyện từ server caro. Rồi show ra màn hình giao diện game  + Kiểm tra vị trí quân cờ của server caro. Thông báo thua, nếu kiểm tra thấy server caro thắng. Sau khi thua thì thiết lập lại bàn cờ, chuẩn bị cho ván cờ mới | 5 |
| 6 |  | + Đến lượt chơi của client caro, client caro đánh quân cờ tại vị trí (point) muốn đánh. Gửi cho server caro  + Gửi chuỗi trò chuyện khi chat cho server caro  + Kiểm tra, thông báo thắng khi client caro đánh thắng. Sau khi thăng thiết lập lại bàn cờ. Chuẩn bị cho ván cờ mới | 6 |
| 7 | + Nhận vị trí quân cờ từ client caro. Rồi repaint lại  + Nhận chuỗi trò chuyện tự client caro. Show ra màn hình  + Kiểm tra vị trí quân cờ của client caro. Thông báo thua nếu thầy client caro đánh thắng. Sau khi thua thiết lập lại bàn cờ chuẩn bị cho ván cờ mới |  |  |
|  |  |  |  |

* 1. **Cơ sở dữ liệu :**

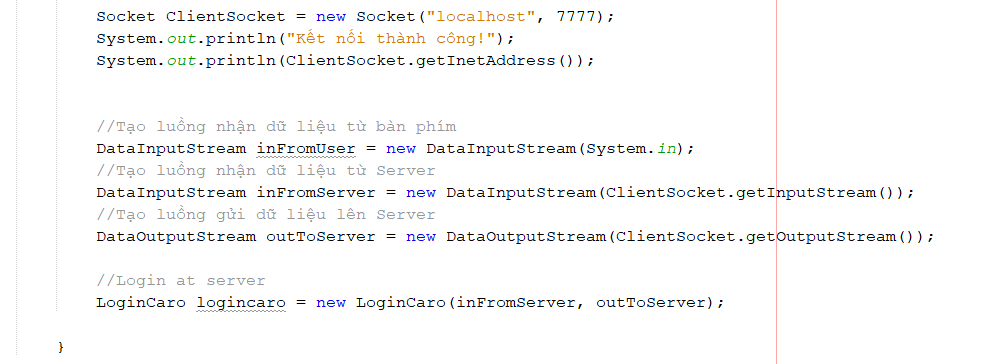
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Null** | **Ghi chú** |
| 1 | Id | Int(11) | N | Mã người dùng |
| 2 | Username | Varchar(255) | N | Tên tài khoản |
| 3 | Password | Varchar(255) | N | Mật khẩu |
| 4 | Fullname | Varchar(255) | N | Tên đầy đủ |
| 5 | Birthday | Varchar(255) | N | Ngày sinh |
| 6 | Email | Int(11) | N | email |
| 7 | Scores | Int(11) | N | Điểm |
| 8 | Matches | Int(11) | N | Số trận |



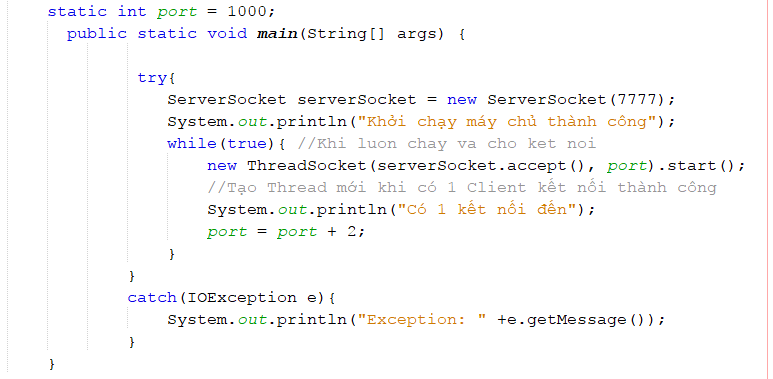
* 1. **Giao thức trao đổi giữa Client và Server :**

Để Client và Server có thể trao đổi dữ liệu với nhau, ta cần thiết lập 1 quy tắc chung, giữa 2 bên để đọc các đối tượng mà bên kia gửi, phân tích đối tượng đó để có các hành động phù hợp.

Trong chương trình game này, client và server trao đổi thông điệp sử dụng Socket thông qua giao thức TCP/IP và Thread tạo các luồng nhận gửi dữ liệu thông qua đối tượng DataInputStream / DataOutputStream.



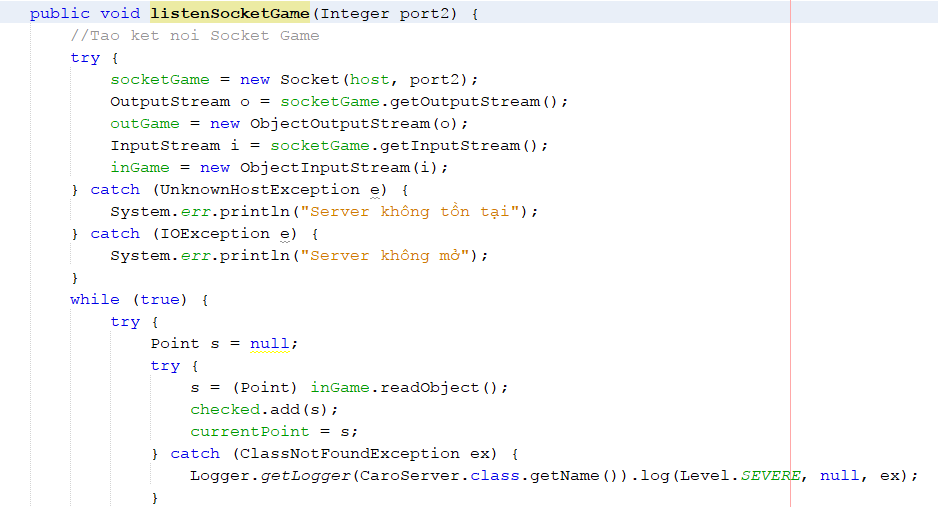
Server caro được tạo ra với 2 port tương ứng. Khởi tạo Thread với 2 port đợi chạy sẵn và chờ client game kết nối vào (1 port dùng để chơi game, 1 port dùng để chat)



Khởi tạo hàm ListenSocket dùng để trò chuyện (Chat). Để yêu cầu được kết nối với Server. Dữ liệu được gửi qua Server nếu đã kết nối đúng Socket.

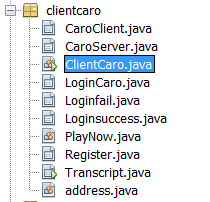


Khởi tạo hàm ListenSocket dùng để chơi Game. Dữ liệu luôn đổi chuyển đổi qua lại giữa Server và Client

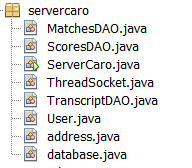


1. **Cài đặt:**
   1. **Các công cụ sử dụng:**
      1. **Môi trường cài đặt:**

* Netbeans IDE 8.2 ( Ngôn ngữ lập trình Java )
* Xampp
  + 1. **Các công nghệ sử dụng:**
* Socket TCP/IP
* Thread
* JDBC
  1. **Các lớp và hàm xử lý chương trình:**
     1. **Phía Client:**

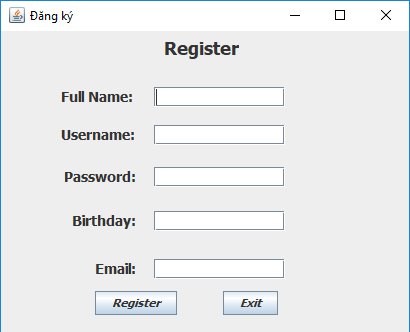


* ClientCaro.java: Hàm chính, chứa hảm main(), chịu trách nhiệm kết nối với Server
* LoginCaro.java: Tạo form đăng nhập, gửi Account lên server để và nhận giá trị trả về true hay false
* Loginfail: Form tạo thông báo Đăng nhập thất bại và gợi ý Đăng ký tài khoản
* Register.java: Tạo form đăng ký thông tin Account và gửi dữ liệu lên server, nhận giá trị trả về là true hay false.
* Loginsuccess.java: Tạo form cho phép người chơi tại server caro để chờ người chơi kết nối vào vào và cho phép người chơi lựa chọn những danh sách server đang chờ đợi kết nối để chơi game
* Transcript.java: Form hiển thị danh sách người chơi, thông tin người chơi, xếp hạng của game
* address.java: chứa thông tin IP và các port của server để client kết nối vào
* CaroClient.java: Form chứa giao diện game, kết nối với Caroserver để chơi game có hỗ trợ trò chuyện trong quá trình chơi
* CaroServer.java: Form chứa giao diện bàn cờ, chờ CaroClient kết nối đến để chơi game có hỗ trợ trò chuyện trong quá trình chơi.
  + 1. **ServerCaro:**

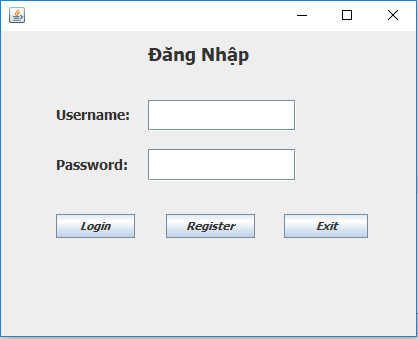


* ServerCaro.java: là hàm chính của chương trình, chứa hàm main(), có trách nhiệm lắng nghe kết nối và kết nối với client khi có yêu cầu.
* ThreadSocket: java: thực hiện tương tác với client, nhận thông tin từ client gửi lên và trả về kết quả phù hợp
* database.java: kết nối với cơ sở dữ liệu để truy xuất thông tin account và lưu thông tin user vào cơ sở dữ liệu
* address.java: lưu IP và port của máy làm server
  1. **Một số giao diện chính:**

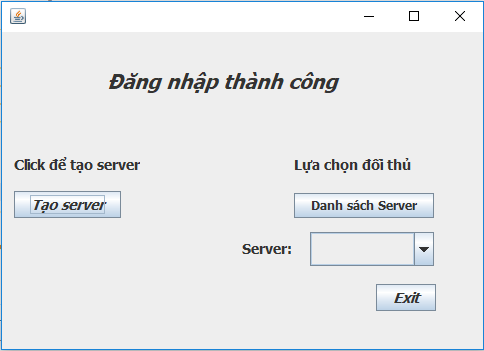
*+) Form đăng ký khi người chơi chưa có tài khoản :*



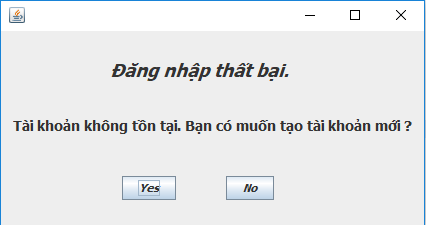
*+) Form đăng nhập khi người chơi đã có tài khoản :*



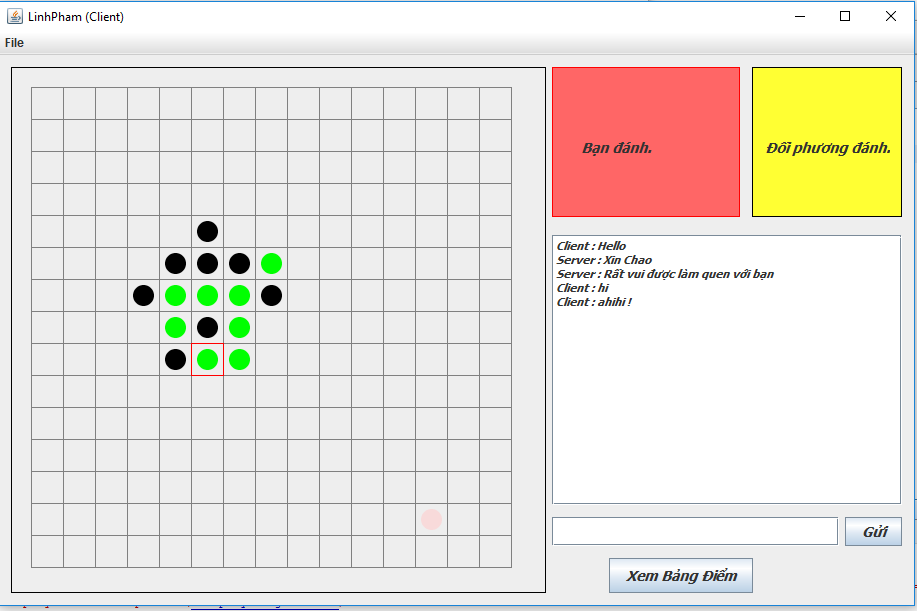
*+) Thông báo khi đăng nhập thành công, người chơi có thể tạo server, hoặc lựa chọn đối thủ cùng chơi*



*+) Thông báo khi đăng nhập thất bại :*



*+) Bàn cờ*



*+) Xem, tìm kiếm danh sách, bảng điểm của những người chơi khác :*

**Chức năng đang được hoàn thiện**

1. **Kết luận:**

Với những thuận lợi và khó khăn trong quá trình xây dựng chương trình, chương trình về cơ bản đã hoàn thành nhưng không tránh khỏi sai sót. Tuy nhiên, Chương trình đã hoàn thành được một số nội dung chính như sau :

* Quản lý người dùng ( Đăng nhập, Đăng kí thành viên )
* Quản lý điểm, danh sách người chơi ( Sắp xếp thứ hạng theo điểm, số trận), sắp xếp danh sách người chơi theo tên.
* Cho phép chơi game caro
* Chức năng Chat ( trò chuyện ) khi chơi game
* Hiển thị thông báo kết quả game, cập nhật vào cơ sở dữ liệu.

1. **Tài liệu tham khảo:**

Tài liệu tiếng Việt:

* Slide Lập trình mạng của thầy Đặng Ngọc Hùng

Các địa chỉ web:

* <http://www.wideskills.com/java-tutorial/java-threads-tutorial>
* Multi-threaded chat Application in Java (Server Side Programming) <http://www.geeksforgeeks.org/multi-threaded-chat-application-set-1/>
* Multi-threaded chat Application in Java (Client Side Programming) <http://www.geeksforgeeks.org/multi-threaded-chat-application-set-2/>

**PHẦN II. CHAT WITH MULTICAST**

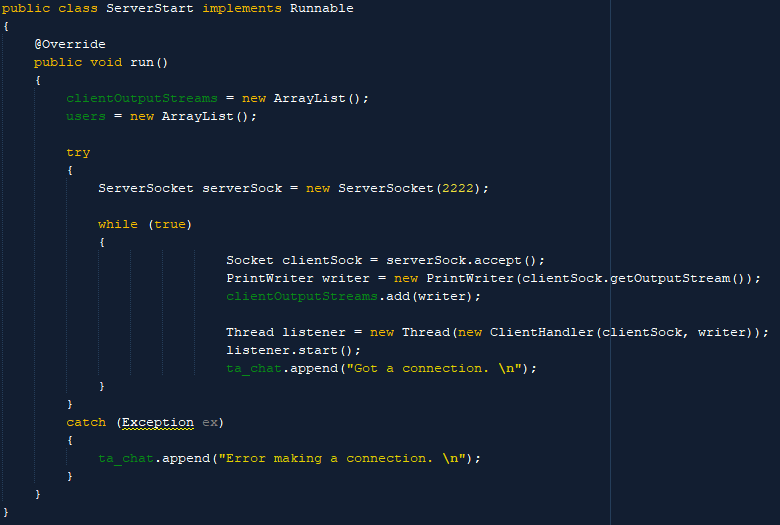
**1. Công cụ**

**- Netbean + JDK**

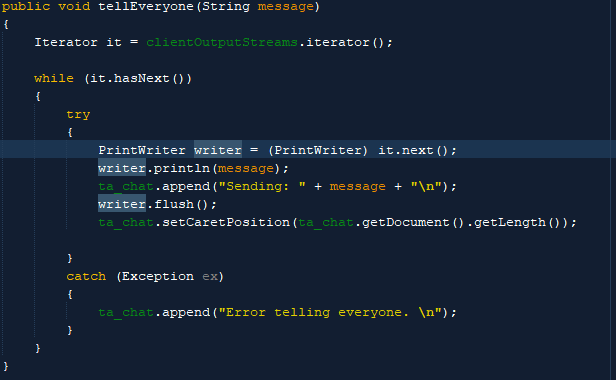
**2. Kiến thức**

**2.1. Tạo server**

**-** Server là một Thread để lắng nghe liên tục từ phía Client

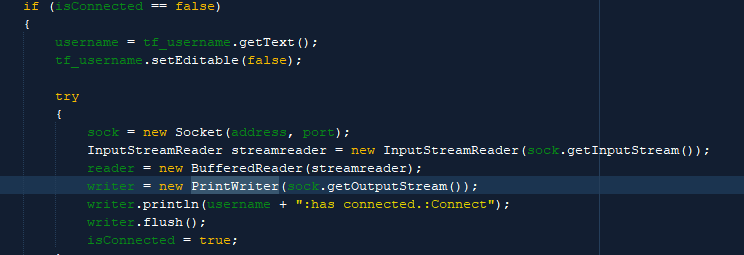


- Thông báo cho các Client đang chat khi có 1 Client mới kết nối vào phòng



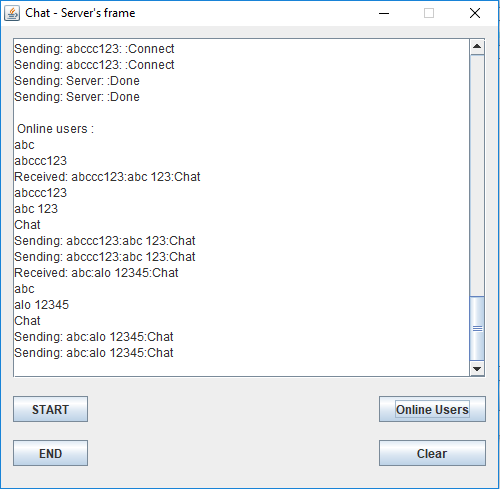
**2.2 Client**

**-** Tạo kết nối tới Server qua port



**3. Giao diện**

**- Server**



Chức năng:

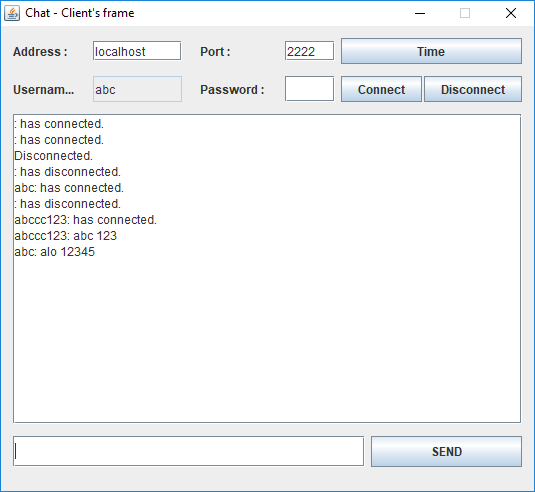
- Start Server

- End Server

- Hiện các users đang kết nối

- Xóa thông báo

**- Client**



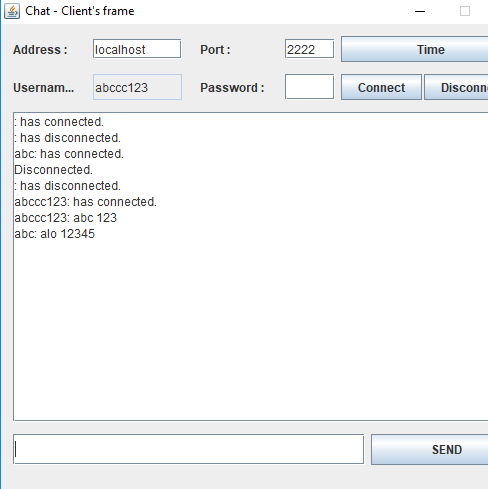
**Chức năng:**

- Nhập Address (default value: localhost)

- Port (default value: 2222)

- Username: Nhập tên (Bỏ trống để ẩn danh)

- Chat với các client khác



**4. Tài liệu tham khảo**

**- Slide “Multicast” của thầy Đặng Ngọc Hùng**

**Link tải source code:**

**- Google drive:** [**https://drive.google.com/drive/folders/1zUvLfMe8Dih7VxoCLvDFqFVBhZOxm\_cM?usp=sharing**](https://drive.google.com/drive/folders/1zUvLfMe8Dih7VxoCLvDFqFVBhZOxm_cM?usp=sharing)

**- Github**

**https://github.com/jnp2018/exam-lehongkhanh-nhom7-carogame**