**QUY TRÌNH XỬ LÝ CỦA NHÓM MÌNH**

Các tên lớp đặc biệt sẽ có màu riêng để phân biệt. Các lớp cách nhau một dòng.

Các mã thông điệp sẽ có màu xám.

Các thắc mắc và lỗi sẽ có màu đỏ.

Tên phương thức được in nghiêng.

**Project Hearts\_Server**

Lớp **Client**

Có id, tên, Socket, tổng điểm. Liệu có thiếu thông tin?

BuffWriter để truyền nhận thông điệp.

Lớp **ListenClientThread** kế thừa từ lớp Thread của hệ thống

Có **ListenThread**, BuffReader, IdClient, Running kiểm tra đang chạy.

*Phương thức run:*

Xử lý các thông điệp bắt đầu là 1 (code 1). Báo cho **ListenThread** xử lý.

Xử lý khi 1 client ngưng chơi đột ngột.

ListenT.GameT khác null thì stop?

Lớp **ListenThread** kế thừa từ lớp Thread của hệ thống

Có biến Running kiểm tra đang chạy hay không, Socket của Server (lớp của hệ thống), số lượng client, danh sách các Client, danh sách các **ListenClientThread** và một biến **GameThread**.

*Phương thức run:*

Kiểm tra đã đủ người chơi hay chưa.

Lớp **GameThread** kế thừa từ lớp Thread của hệ thống

Có **ListenThread** và mảng chuỗi MessageClient.

*Phương thức run:*

Gửi yêu cầu username và kiểm tra trùng tên.

Gửi yêu cầu bắt đầu chơi (code 1).

Tạo LuotDi, ban đầu là 0, có giá trị từ 0 đến 3.

Tạo DiemVanHienTai. Sao không để client lưu thông tin này?

Chia bài cho 4 client. Kiểm tra client có 2 chuồn và ghi nhớ lại. Gửi danh sách bài cho client (code 3).

Xem có trao đổi bài không (code 4). Kiểm tra lại người giữ 2 chuồn sau khi đổi bài. Chưa hiểu gán LuotDi? Gửi bài lại cho client. Vì sao chỉ có 3 client?

Thông báo tới lượt đi (code 5). Không đồng nhất với cấu trúc thông điệp.

Nhận lá bài từ client và gửi cho các client còn lại (code 6).

Kiểm tra chất bài và shoot the moon – có quyền đi cơ (code 7).

Xử lý khi kết thúc lượt chơi. Tên bị mâu thuẫn giữa Thuận và Tiến. Gửi tổng điểm cho 4 client (code 2).

Kiểm tra xem có trường hợp client có 13 lá cơ và đầm bích để thay đổi điểm.

Gửi kết quả ván chơi (code 10). Code 2 và code 10 liệu có giống nhau nếu code 2 gửi 4 lần?

Kết thúc ván chơi (code 8). Comment bị sai: kết thúc ván chơi, không phải kết thúc trò chơi.

*Phương thức XepBai:*

Tạo danh sách lá bài. Tên bị mâu thuẫn giữa Thuận và Tiến.

Xáo bài (cách random bài đã đề cập trong QuyTrinhXuLy.docx).

Lớp **Hearts\_ServerApp**: tự phát sinh.

Lớp **Hearts\_ServerAboutBox**: tự phát sinh. Cần sửa lại giống như của Tiến.

Lớp **Hearts\_Server**: tự phát sinh.

Cho phép nhập port để lắng nghe, button Bắt đầu, button đóng.

Có biến PortLangNghe, ServerSocket, **ListenThread**

*Phương thức jTextPortLangNgheKeyPressed:*

Chưa hiểu 2 dòng lệnh phẩn xử lý bên trong có ý nghĩa gì?

*Phương thức jButtonBatDauMouseClicked:*

Lấy PortLangNghe từ textbox. Khởi tạo ServerSocket từ port này. Không cho nhập port nữa. Khi cho nhấn button này (Bắt đầu) nữa. Khởi tạo **ListenThread** dựa trên ServerSocket vừa mới tạo.

**Project Hearts\_Client**

Lớp **ListenThread** kế thừa từ lớp Thread của hệ thống

Có Socket (client socket), BufferedReader, **ConnectJFrame**, **LoginJFrame**, **GameJFrame**, biến boolean Running.

*Phương thức run:*

Tạo chuỗi MessageServer, chuỗi NoiDung, biến số nguyên Co.

Lấy nội dung từ BufferedReader gán vào MessageServer.

Lấy ra mã thông điệp gán vào Co.

Code 1: chưa đủ người chơi, hiện thông báo.

Code 1: đã đủ người chơi, đóng client socket và BufferedReader. Hiện lên JFrame ConnectF để kết nối.

Code 1: yêu cầu username. Hiện JFrame LoginF để đăng nhập.

Code 1: trùng tên, hiện thông báo. Hiện lên JFrame ConnectF để kết nối.

Code 1: bắt đầu chơi. Ẩn JFrame LoginF. Khởi tạo JFrame GameF, gán thông tin cho biến NoiDung. Kiểm tra trùng tên.

Code 2-11: hiện ra thông báo ở JFrame GameF.

Lớp **ConnectJFrame**

Có 2 textbox để nhập IP và port và một button Kết nối.

Có Socket (client socket), **ListenThread**, BufferedWriter và chuỗi MyName.

*Phương thức sendMessage:*

Thêm ký tự xuống dòng vào chuỗi. Sau đó dùng BufferedWriter để ghi chuỗi.

*Phương thức jButtonKetNoiMouseClicked với tham số là chuỗi thông điệp:*

Tạo client socket dựa vào IP và Port.

Tạo InputStream, BufferedReader, OutputStream

Tạo **ListenThread** với tham số là ConnectJFrame hiện tại, client socket và BufferedReader.

Lớp **GameJFrame**

Lớp này để test?

Lớp **LoginJFrame**

Có biến **ConnectJFrame** (ConnectF).

*Phương thức jButtonDangKyMouseClicked:*

Kiểm tra tên đăng nhập nếu rỗng thì thông báo. Ngược lại ConnectF sẽ gửi thông điệp báo lại tên đăng nhập với mã thông điệp là 1, đồng thời gán thuộc tính MyName là tên đăng nhập.

Lớp Hearts\_ClientApp: tự phát sinh.

Lớp Hearts\_ClientAboutBox: tự phát sinh. Cần sửa lại giống như của Tiến.