ĐẠI HỌC QUỐC GIA TP.HCM TRƯỜNG ĐẠI HỌC BÁCH KHOA TP.HCM KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH



GVHD: Bùi Xuân Giang

BÁO CÁO BÀI TẬP LỚN MẠNG MÁY TÍNH

Sinh viên thực hiện: Phạm Văn Việt - 1713955

Nguyễn Đình Thịnh - 1713325

Nguyễn Tấn Tín - 1713511

Mục lục

| 1. Giới thiệu ứng dụng | 2 |
|--|---|
| 1.1 Login | |
| 1.2 Chat riêng tư | |
| 1.3 Truyền gửi file | |
| 1.4 Chat nhóm | |
| 1.5 Xem danh sách Client đang online | 2 |
| 2. Định nghĩa giao thức cho từng chức năng | |
| 3. Thiết kế ứng dụng | 3 |
| 3.1 Công nghệ sử dụng | 3 |
| 3.2 Các class chính | 3 |
| 4. Đánh giá kết quả hiện thực | 5 |
| 4.1 Kết quả đạt được | 5 |
| 4.2 Han chế cần phát triển sau này | 5 |

1. Giới thiệu ứng dụng

Úng dụng là phần mềm cho phép hai hay nhiều người dùng có thể giao tiếp với nhau với các tính năng chính: login, chat nhóm, tạo chat riêng tư, truyền gửi file trong lúc chat.

1.1 Login

Chức năng này cho phép người dùng đăng nhập vào ứng dụng khi biết IP và port của server.

1.2 Chat riêng tư

Chức năng này cho phép user tạo hội thoại riêng tư và gửi tin trực tiếp tới user khác. Một user có thể nhắn tin trực tiếp với nhiều user trong cùng một thời điểm.

1.3 Truyền gửi file

Chức năng này cho phép user gửi và nhận một số file với định dạng và kích thước được quy định trước bởi nhà phát triển trong chat riêng tư với user khác khi được cho phép.

1.4 Chat nhóm

Chức năng này cho phép những Client đã kết nối với Server chat với nhau không gồm tính năng gửi file.

1.5 Xem danh sách Client đang online

Hiển thi danh sách Client đã kết nối với Server

2. Định nghĩa giao thức cho từng chức năng

| Nội dung | Mục đích | Ghi chú |
|---|----------------------|----------------------------|
| messageBody | Gửi nội dung chat | messageBody là nội dung |
| | trong chat room | tin nhắn |
| clientName disconnects | Gửi thông báo client | clientName là tên client |
| <remove_client></remove_client> | đã ngắt kết nối | đã ngắt kết nối với Server |
| <start_chat_dialog></start_chat_dialog> | Yêu cầu tạo dialog | clientName1: tên client 1 |
| clientName1 to clientName2 | chat riêng tư giữa 2 | clientName2: tên client 2 |
| | client | |
| <send_direct_message></send_direct_message> | Gửi tin nhắn giữa 2 | clientName1: tên client 1 |
| clientName1 to clientName2 | client | clientName2: tên client 2 |
| msgBody | | msgBody: nội dung tin |
| | | nhắn |

| <pre><send_direct_file> clientName1 to clientName2 fileName fileSize</send_direct_file></pre> | Gửi file giữa 2 client | clientName1: tên client 1 clientName2: tên client 2 filename: tên file cần gửi fileSize: kích thước file cần gửi |
|---|----------------------------------|--|
| <stop_chat_dialog> clientName1 to clientName2</stop_chat_dialog> | Dừng chat riêng tư giữa 2 client | clientName1: tên client 1 clientName2: tên client 2 |

3. Thiết kế ứng dụng

3.1 Công nghệ sử dụng

- TCP Socket: Một kĩ thuật dùng để hỗ trợ lập trình các ứng dụng giao tiếp qua mạng. TCP Socket sử dụng Stream để thực hiện quá trình truyền dữ liệu của hai máy tính đã thiết lập kết nối.
- Java Swing: Là một phần của Java Foundation Classes (JFC) được sử dụng để tạo các ứng dụng Window-Based.

3.2 Các class chính

Class Diagram của ứng dụng:

<<Java Class>> **⊕** ChatDialog

client

- △ parentFrame: JFrame
- △ txtDisplayChat: JTextPane
- △ btnSend: JButton
- △ btnSendFile: JButton
- △ btnLike: JButton
- △ btnSad: JButton
- △ btnScared: JButton
- △ btnSmileBig: JButton
- △ btnSmileCry: JButton
- △ btnSmile: JButton
- △ btnCrving: JButton
- △ btnHeartEye: JButton
- △ txtMsg: JTextField
- △ w riter: DataOutputStream
- a socket: Socket
- oSSEND_DIRECT_MESSAGE: String
- oSSEND_DIRECT_FILE: String
- OSACTIVE_DIALOG: String
- oSDEACTIVE DIALOG: String
- ChatDialog(JFrame.Socket.String)
- updateChat_receive(String):void updateChat_send(String):void
- updateChat_notify(String):void
- updateSendFile notify(String):void
- updateChat send Symbol(String):void
- appendToPane(JTextPane,String):void
- activeDialog(boolean):void
- w indow Closed(Window Event):void
- w indow Activated(Window Event):void
- w indow Deactivated(Window Event):void w indow lconified(Window Event):void
- w indow Deiconified(Window Event):void
- w indow Opened(Window Event):void
- w indow Closing(Window Event):void
- actionPerformed(ActionEvent):void

<<Java Class>> ● FileReceive

file

- a saveDirDefault: String
- □ fileName: String
- n fileSize: long
- a socket: Socket

FileReceive(String,String,long,Socket) receive():void

<<.lava Class>>

- n fileName: String
- dos: DataOutputStream
- saveDirDefault: String

FileSend(String,String,DataOutputStream)

send():void

<<Java Class>> ⊕ TcpClient

client

- p side panel width; int
- bottom_panel_height: int
- a server_port: int
- a server_ip: String
- a clientName: String
- a oneTimeDialog: JDialog
- chatDialog: JDialog
- a receiver: Thread
- a sender: Thread
- a socket: Socket
- △ btnSend: JButton
- △ btnConnect: JButton
- △ btnLike: JButton
- △ btnSad: JButton
- △ btnScared: JButton
- △ btnSmileBig: JButton
- △ btnSmileCry: JButton
- △ btnSmile: JButton
- △ btnCrying: JButton
- △ btnHeartEye: JButton
- △ txtInput: JTextField
- △ txtDisplayChat: JTextPane
- △ txtName: JTextField
- △ txtIP: JTextField

-channels

04

- A txtPort: JTextField
- IstOnline: JList<String>
- model: DefaultListModel<String>
- a clickedAlready: boolean
- △ w riter: DataOutputStream
- △ reader: DataInputStream
- a clientDir: String
- △^Sthreads_running_flag: boolean
- SADD_CLIENT: String oSREMOVE CLIENT: String
- oSLIST_CLIENTS: String
- oSTART_CHAT_DIALOG: String
- oSSEND_DIRECT_MESSAGE: String
- oSSEND_DIRECT_FILE: String
- SACTIVE_DIALOG: String
- SDEACTIVE_DIALOG: String

TcpClient()

- createOneTimeDialog():void
- createUsersPanel():void
- o createMainPanel():void
- updateChat_receive(String,String):void
- updateChat_send(String):void
- updateChat_notify(String):void
- updateChat_send_Symbol(String):void
- appendToPane(JTextPane,String):void
- test(Object):void
- Smain(String[]):void
- actionPerformed(ActionEvent):void
- addAliasToList(String):void
- removeAliasFromList(String):void
- valueChanged(ListSelectionEvent):void
- mouseClicked(MouseEvent):void
- mouseEntered(MouseEvent):void mouseExited(MouseEvent):void
- mousePressed(MouseEvent):void
- mouseReleased(MouseEvent):void

<<Java Class>> ServerForm server

oSport: int

- n frmServerMangement: JFrame
- □ txtIP: JTextField
- txtPort: JTextField
- n IbiStatus: JLabel
- ^StxtMessage: TextArea
- △ btnStop: JButton
- △ btnStart: JButton
- Smain(String[]):void
- ServerForm()
- initialize():void
- m addEvents():void

~server 0..1

<<Java Class>> **⊕** TcpServer server

SuFPORT: int

- △SserverSocket: ServerSocket
- ^{∆S}hashMap: HashMap<Socket,String>
- △Socket: Socket
- △Sname: String
- △Scount: int
- n isStop: boolean
- a saveServerDirDefault: String
- oSADD_CLIENT: String
- oSREMOVE_CLIENT: String
- oSLIST CLIENTS: String
- oSSTART_CHAT_DIALOG: String
- SEND_DIRECT_MESSAGE: String oSSEND_DIRECT_FILE: String
- oSACTIVE_DIALOG: String
- oSDEACTIVE_DIALOG: String

*TcpServer(int) stopserver():void

<<Java Class>>

Server accepter

- serv er △ reader: DataInputStream
- △ out: DataOutputStream
- Server_accepter()
- orun():void remotePrint(Socket.String):void
- nameAlreadyPresent(String):boolean

Class cho server:

- ServerForm.java: sử dụng để thao tác với server như bật, tắt server, chọn nơi lưu trữ file khi Client gửi lên. Ngoài ra còn hiển thị các thông tin server như: IP, Port, các protocol mà client gửi lên, trạng thái của server.
- TcpServer.java: Đây là class để quản lí server, gồm đầy đủ thông tin, chức năng chính của server:
 - Tao môt SocketServer.
 - Quản lí danh sách các user đang online
 - Tạo một thread là Server_accepter để xử lý các request kết nối từ user. Có thể là yêu cầu kết nối với server, thoát ứng dụng, lấy danh sách user đang online. Nếu đăng nhập thành công, trả về cho client danh sách user đang online.

Class cho client:

- TcpClient.java: class gồm có GUI hiển thị giao diện login khi kết nối thành công, sẽ hiển thị giao diện chat room gồm có danh sách user khác đang online, bắt sự kiện gửi yêu cầu chat của user.
- ChatDialog.java: xử lí việc chat, gửi nhận File giữa hai user. Hiển thị nội dung chat giữa hai user.

Class khác:

- FileSend.java: xử lí yêu cầu gửi file từ Client lên Server và từ Server xuống Client.
- FileReceive.java: xử lí việc nhận file của Client và Server.

4. Đánh giá kết quả hiện thực

4.1 Kết quả đạt được

- Úng dụng được xây dựng dựa trên mô hình client-server cho việc quản lí việc trò chuyện giữa các user với nhau.
- Ứng dụng có các tính năng đơn giản như: chat private giữa hai user, một lúc đồng thời chat với nhiều user, chat room, gửi File trong quá trình chat.

4.2 Hạn chế cần phát triển sau này

- Mã nguồn còn chưa tối ưu cho ứng dụng.
- Úng dụng còn có thể thêm các tính năng như: video call, voice call, ...
- Giao diện ứng dụng còn đơn giản.