

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



COMPUTER NETWORKING

ASSIGNMENT 1

PEER TO PEER CENTRALIZED

GVHD:	Bùi Xuân Giang	
SV:	Nguyễn Hữu Thắng	1713239
	Từ Ngọc Huy	1711567
	Võ Trung Thiên Tường	1710380

TP. HỒ CHÍ MINH, THÁNG 11/2019

Mục lục

1	Định nghĩa chức năng	2
1.1	Chức năng login	2
1.2	Chat riêng tư	2
1.3	Chat với nhiều người cùng lúc	2
1.4	Truyền gửi file	2
2	Định nghĩa giao thức	2
3	Thiết kế ứng dụng	2
3.1	Công nghệ sử dụng	2
3.2	Kiến trúc ứng dụng	3
3.2.1	Giao diện	3
3.2.2	Class Diagram	4
4	Hướng dẫn sử dụng	5
5	Đánh giá	6
5.1	Kết quả đạt được	6
5.2	Công việc tương lai	7

1 Định nghĩa chức năng

1.1 Chức năng login

Chức năng này cho phép người dùng đăng nhập vào ứng dụng khi biết IP và port của server và khi đăng nhập người dùng phải nhập username riêng biệt, không được trùng với những username đang online. Nếu người dùng nhập username đã tồn tại thì hệ thống sẽ hiển thị thông báo lỗi và bắt người dùng nhập lại username cho đến khi hợp lệ.

1.2 Chat riêng tư

Chức năng này cho phép user tạo hội thoại riêng tư và gửi tin nhắn trực tiếp tới user khác và những user khác sẽ không nhận được tin nhắn này.

1.3 Chat với nhiều người cùng lúc

Chức năng này cho phép user tạo nhiều hội thoại riêng tư cùng lúc để chat với nhiều user với nhau.

1.4 Truyền gửi file

Chức năng này cho phép user gửi và nhận một số file với khác nhau định dạng và kích thước tối đa vì file đã được cắt ra thành từng mảnh nhỏ và gửi đến user khác nên không xảy ra tràn buffer.

2 Định nghĩa giao thức

Giao thức được định nghĩa dùng thẻ XML format theo Bảng 1. Đây là mô hình lai giữa client-server và peer-to-peer(peer-to-peer centralized). Server giao tiếp với mỗi client để nhận thông tin từ client và client giao tiếp với server để biết được những client đang online và thông tin về client đó để tiến hành chat. Khi client biết được thông tin của destination client thì có thể tiến hành chat session qua peer-to-peer. Nếu server chết thì các client đã biết thông tin nhau có thể giao tiếp với nhau(giải quyết được single point of failure).

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.
- PyQt5: Lập trình giao diện của Python được sử dụng để tạo các ứng dụng Window-Based [1]. Hỗ trợ thao tác kéo thả sinh code giao diện. Giúp việc hiện thực giao diện dễ dàng hơn qua Qt Designer [3].
- Xmltodict : Một module của Python khiến cho việc làm việc với XML trở nên dễ dàng hơn [2].

Bảng 1: Định nghĩa giao thức

Nội dung	Mục đích	Ghi chú
<peer_name> username </peer_name>	Dùng để tạo client khi client đăng nhập vào hệ thống	Tên không được quá dài và phải đảm bảo duy nhất
<peer_port> port </peer_port>	Dùng để tạo port cho client	Port được random nên người dùng không cần nhập
<peer_host> host </peer_host>	Dùng để truyền IP của máy client tới server. IP này cũng được dùng để chat giữa các client	Host được tự động get
<peer_info> info </peer_info>	Dùng để gộp những thông tin của client để dễ dàng quản lý hơn	Chỉ giúp quản lý dễ dàng hơn
<alive>alive </alive>	Dùng để mỗi 5s server sẽ thông báo đến mỗi client. Nếu client không nhận được (chết) thì server sẽ xóa client đó ra khỏi danh sách đang online	
<empty_peer> empty_peer </empty_peer>	Dùng để sử dụng khi client là người đầu tiên online	
<session> user- name </session>	Dùng để thông tin của client tới destination client	Bắt đầu chat session
<content> content </content>	Để chứa nội dung chat	
<msg>msg</msg>	Dùng bọc giữa nội dung chat và username của destination client	Để dễ dàng quản lý message
<file_name> file_name </file_name>	Dùng để gửi tên file đến destination client. Khi nhận được tag này phía client sẽ đọc binary được gửi từ source client và ghi file xuống.	File có thể có kích thước và định dạng khác nhau

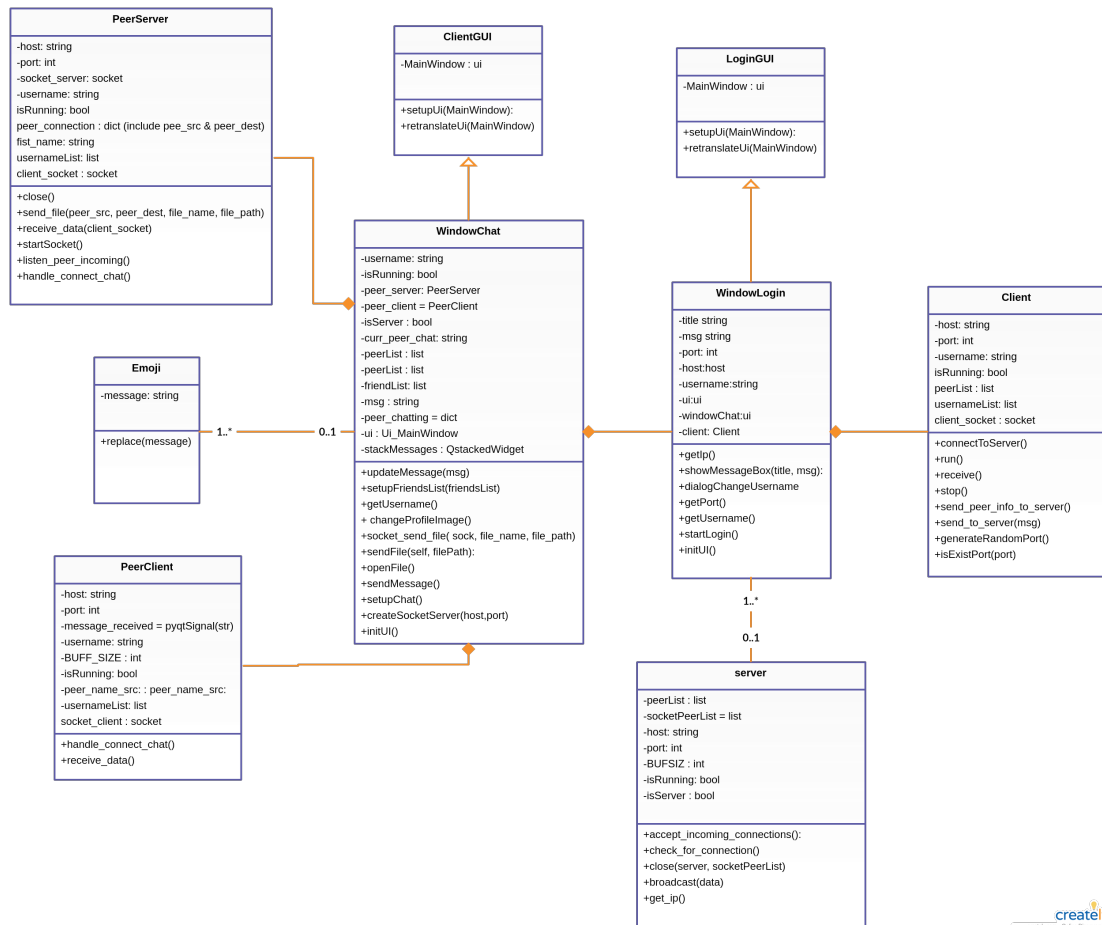
3.2 Kiến trúc ứng dụng

3.2.1 Giao diện

Ứng dụng gồm 2 giao diện chính:

- Giao diện Login: Đăng nhập vào server. Để kết nối được, user phải cung cấp username chưa có ai đăng kí trước đó với server và phải nhập đúng địa chỉ IP và port của server để đảm bảo kết nối tới server.
- Giao diện Chat: Khi client đăng nhập và kết nối thành công tới server thì giao diện chat sẽ hiện ra. Ở đây người dùng có thể chọn client để trò chuyện và share file cho nhau một cách dễ dàng.

Terminal server: Do sử dụng giao diện thì sẽ tốn khá nhiều tài nguyên, và không cần thiết nên server sử dụng terminal để tương tác.



Hình 1: Class Diagram

3.2.2 Class Diagram

Các class server:

- Server: Dùng để bắt kết nối client và quản lý thông tin của các client và điều phối nó tới các client online. Đồng thời mỗi 5s gửi message kiểm tra các client đang online.

Các class client:

- LoginGUI: GUI class hiển thị và nhập thông tin gửi tới server.
- ClientGUI: GUI class để hiển thị danh sách user khác đang online, bắt sự kiện gửi yêu cầu chat của user và hiển thị tin nhắn người dùng.
- WindowLogin: Dùng để xử lý sự kiện liên quan đến kết nối server.
- WindowChat: Dùng để xử lý sự kiện liên quan đến chat và gửi file.
- Client: Dùng để tạo socket và kết nối và nhận dữ liệu từ server.

- PeerServer : Tạo một đối tượng PeerServer có các thuộc tính host, port,name và thực hiện nhiệm vụ như một server trong peer-to-peer.
- PeerClient : Tạo một đối tượng PeerClient có các thuộc tính host,port,name và thực hiện nhiệm vụ như một client trong peer-to-peer.

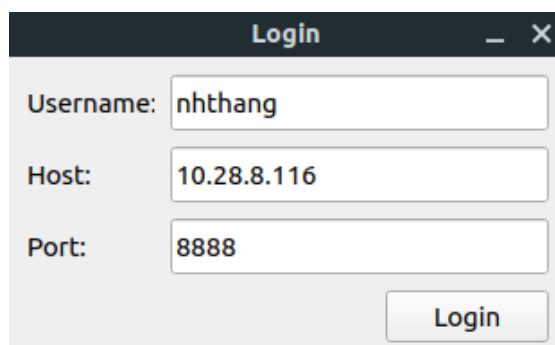
Các class khác:

- Tags: Định nghĩa các tag được sử dụng trong ứng dụng theo XML format.
- Emoji : Chuyển các emoji nhập từ bàn phím thành những emoticon khi chat
- Encode: Mã hóa thông tin thành các tag theo XML format.
- Decode: Giải mã XML format thành các thông tin tương ứng.

4 Hướng dẫn sử dụng

```
Server start with 10.28.8.116:8888  
Waiting for connection ...  
nhthang has connected.  
tuong has connected.  
huy has connected.
```

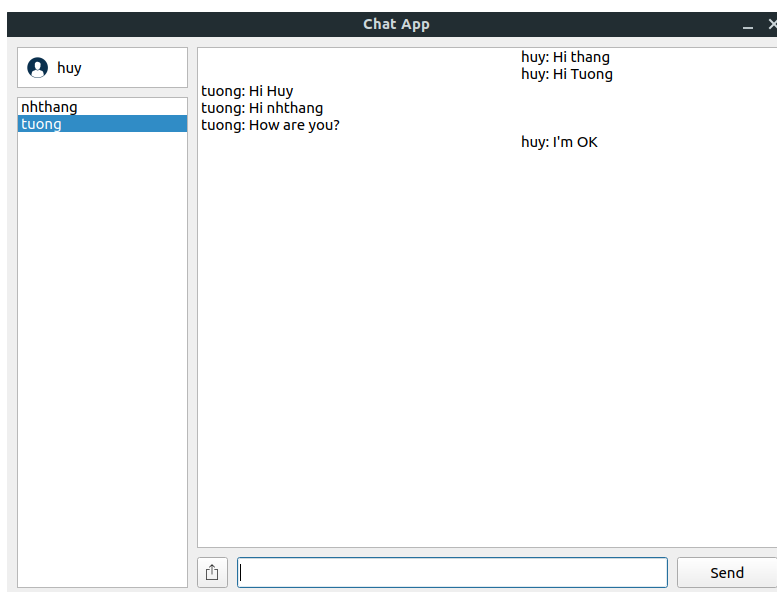
Hình 2: Server Terminal



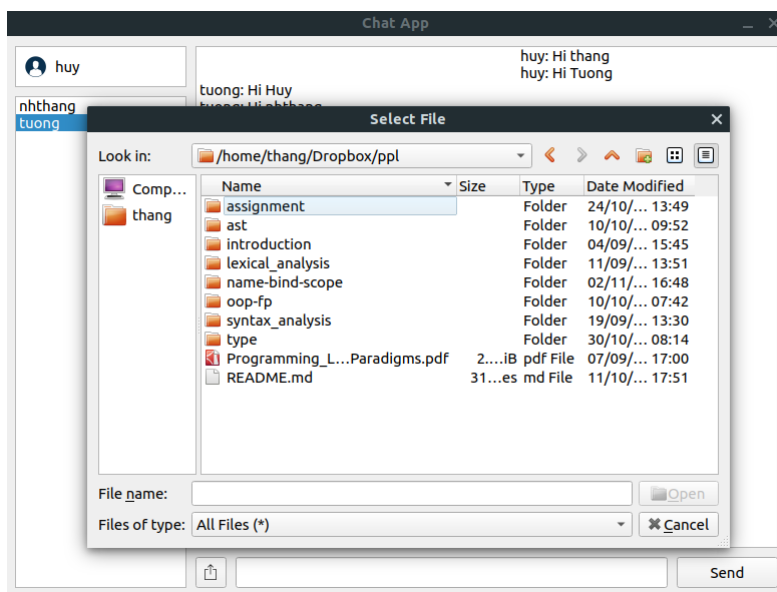
The image shows a 'Login' window with a dark title bar. It contains three input fields: 'Username:' with the value 'nhthang', 'Host:' with the value '10.28.8.116', and 'Port:' with the value '8888'. A 'Login' button is located at the bottom right of the form area.

Hình 3: Login GUI

- Ta chạy server.exe để khởi tạo server. Ứng dụng sẽ được chạy ngầm (hình 2)
- Ta nhập Username mà ta muốn hiển thị. Username không được trùng với cái peer trước đó và không được trống. Host và Port phải tương ứng với Host và Port của server. (Hình 3)
- Trước khi bắt đầu chat ta sẽ chọn đối tượng để chat trong danh sách bạn. Sau khi nhấp vào đối tượng thì ta có thể tiến hành nhắn tin ở khung chat phía dưới màn hình. Tin nhắn sẽ được hiện lên ở cửa sổ chat.(Hình 4)
- Để gửi file ta nhấp vào biểu tượng gửi file ở phía trái khung chat. Sau đó ta tiến hành nhập vào đường dẫn và chọn file đã tạo. Sau đó click 'open' (hình 5)



Hình 4: Chat GUI



Hình 5: Send file GUI

5 Đánh giá

5.1 Kết quả đạt được

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

- Ứng dụng có các tính năng đơn giản như: chat giữa hai user, một lúc đồng thời chat với nhiều user, gửi file trong quá trình chat.
- Đã giải quyết được vấn đề chia sẻ file lớn bằng cách cắt file ra thành nhiều phần nhỏ và gửi đi. File được chia sẻ có thể có nhiều định dạng khác nhau text, multimedia,... và kích thước khác nhau(đã thử nghiệm với file 4,6GB và nhận được sau 35s).
- Lấy được real interface của máy client cũng như server.
- Nắm được nhiều kỹ năng về lập trình Python cũng như lập trình đa luồng và giao diện. Ứng dụng trong quá trình chạy có thể lên tới 10 luồng.
- Hiểu được socket.send() và socket.sendfile() dùng để gửi file thì tốc độ của socket.sendfile() sẽ nhanh hơn [4].

5.2 Công việc tương lai

- Hoàn thành tách message cho từng user dùng StackedWidget.
- Tối ưu hiệu suất của ứng dụng.



Tài liệu

- [1] Qt Cross-platform <https://www.qt.io/>
- [2] xmldict - Python module that makes working with XML feel like you are working with JSON, as in this "spec" <https://github.com/martinblech/xmldict>
- [3] Qt Designer <https://doc.qt.io/qt-5/qtdesigner-manual.html>
- [4] File transfer over sockets without user-space memory in Python 3.5
<http://michaldul.com/python/sendfile/>