

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



MẠNG MÁY TÍNH (CO3093)

ASSIGNMENT 1

DEVELOP FILE-SHARING APPLICATION

GVHD: Lê Bảo Khánh
SV: Lê Thanh Toàn Đạt - 2115386
Võ Mạnh Hào - 1913238
Lê Minh Hoàng - 1913424
Nguyễn Quốc Huy - 2113519

Thành phố Hồ Chí Minh, Tháng 10 2023



Mục lục

1	Nhiệm vụ, vai trò các thành viên	2
2	Giới thiệu và cơ sở lý thuyết	2
2.1	Giới thiệu đề tài	2
2.2	Giới thiệu về Client-Server và Peer-to-Peer	3
2.2.1	Client-Server	3
2.2.2	Peer-to-Peer	4
3	Mô tả chức năng	5
3.1	Tính năng cơ bản của server và client	5
3.2	Activity Diagram	6
3.3	Sequence Diagram	6
3.4	Các hàm chức năng	6
3.4.1	Server	6
3.4.2	Client	7
3.5	Luồng thực thi cho từng hàm	7
3.5.1	Server	7
3.5.2	Client	8
4	Manual	9
4.1	Server Side	9
4.2	Client Side	12
5	Source code	18
6	Tài liệu tham khảo	18



1 Nhiệm vụ, vai trò các thành viên

STT	Họ và tên	MSSV	Công việc	Mức độ hoàn thành
1	Lê Thanh Toàn Đạt	2115386	-Xây dựng giao diện Client -Hiện thực chức năng đăng ký, đăng nhập và các chức năng bên phía server.	100%
2	Võ Mạnh Hào	1913238	-Hiện thực các tính năng bên phía Client -Viết báo cáo -Soạn slide trình chiếu	100%
3	Lê Minh Hoàng	1913424	-Xây dựng và hiện thực các tính bên phía Client -Viết báo cáo	100%
4	Nguyễn Quốc Huy	2113519	-Xây dựng CSDL và giao diện Server -Hiện thực các tính năng bên phía server	100%

2 Giới thiệu và cơ sở lý thuyết

2.1 Giới thiệu đề tài

Trong hai thập kỉ trở lại đây, với sự phát triển không ngừng của của kỹ thuật, công nghệ, bắt nguồn từ xuất hiện lần đầu của hệ điều hành "thân thiện cho người dùng đầu tiên" Windows 95 đã mang sự tiện lợi cho việc ai cũng có thể sử dụng máy tính để thao tác, đến sự ra đời của các mạng xã hội, các công ty công nghệ lớn như Google, Facebook, Youtube đến sự ra đời của các chiếc điện thoại cảm ứng đầu tiên vào năm 2007 cho đến nay đời sống của con người luôn gắn liền với các thiết bị điện tử. Từ các thanh thiếu niên, thích thú đăng ảnh, các video ngắn lên mạng xã hội, hay các cô chú nhân viên văn phòng sử dụng các công cụ trên máy tính/điện thoại để làm việc đến các cụ già, người lớn tuổi sử dụng mạng xã hội để liên hệ cho con cháu,... Để giải trí, làm việc và học tập hiệu quả con người có nhu chia sẻ các file cho nhau để công bố các nghiên cứu mình tìm được cho người khác, chia sẻ những kinh nghiệm hữu ích của mình đến với mọi người, để mọi người có thể cùng hợp tác với mình để thực hiện điều gì đó. Có thể nói là nhu cầu chia sẻ, gửi dữ liệu từ một người lên internet cho một hoặc nhiều người khác cùng nhận tăng cao, nhận ra điều này các tập đoàn công nghệ lớn liên tục cho ra đời các công cụ có khả năng share file hoặc transfer file như google drive, dropbox, onedrive, github,...

Trong bài tập lớn môn Mạng máy tính lần này, để sinh viên làm quen với mạng internet cũng như cách truyền tập tin cơ bản từ client này đến client khác, thì sinh viên phải tạo một phần mềm truyền file Peer-to-Peer bằng giao thức TCP/IP, với Server chỉ làm nhiệm vụ lưu trữ tên các file mà một client đang share và địa chỉ IP của client đó nhằm cung cấp cho một client khác nếu client khác đó muốn tải file của client này về máy, còn mọi hoạt động kết nối và truyền file chỉ xảy ra ở 2 client với nhau.

2.2 Giới thiệu về Client-Server và Peer-to-Peer

2.2.1 Client-Server

Client: Với vai trò là máy khách, chúng sẽ không cung cấp tài nguyên đến các máy tính khác mà chỉ sử dụng tài nguyên được cung cấp từ máy chủ. Một client trong mô hình này có thể là một server cho mô hình khác, tùy thuộc vào nhu cầu sử dụng của người dùng.

Server: Là máy tính có khả năng cung cấp tài nguyên và các dịch vụ đến các máy khách khác trong hệ thống mạng. Server đóng vai trò hỗ trợ cho các hoạt động trên máy khách client diễn ra hiệu quả hơn.

Mô hình Client Server là mô hình mạng máy tính trong đó các máy tính con được đóng vai trò như một máy khách, chúng làm nhiệm vụ gửi yêu cầu đến các máy chủ. Để máy chủ xử lý yêu cầu và trả kết quả về cho máy khách đó.

Để máy khách và máy chủ có thể giao tiếp được với nhau thì giữa chúng phải có một chuẩn nhất định, và chuẩn đó được gọi là giao thức. Một số giao thức được sử dụng phổ biến hiện nay như: HTTPS, TCP/IP, FTP,...

Nếu máy khách muốn lấy được thông tin từ máy chủ, chúng phải tuân theo một giao thức mà máy chủ đó đưa ra. Nếu yêu cầu đó được chấp nhận thì máy chủ sẽ thu thập thông tin và trả về kết quả cho máy khách yêu cầu. Bởi vì Server - máy chủ luôn luôn trong trạng thái sẵn sàng để nhận request từ client nên chỉ cần client gửi yêu cầu tín hiệu và chấp nhận yêu cầu đó thì server sẽ trả kết quả về phía client trong thời gian ngắn nhất.

Ưu điểm:

- **Tập trung:** Theo như mô hình này thì tất cả mọi thông tin cần thiết đều sẽ được đặt ở một vị trí duy nhất. Đây là một ưu điểm vô cùng hữu ích được những người quản trị viên mạng yêu thích bởi vì họ có thể toàn quyền quản lý cũng như điều hành mọi việc. Tính năng này giúp cho mọi sự cố trong mạng đều sẽ được giải quyết ở cùng một nơi thống nhất. Đồng thời, việc cập nhật cơ sở tài nguyên, dữ liệu cũng sẽ dễ dàng hơn rất nhiều.
- **Bảo mật:** Trong mạng Client Server, tất cả các dữ liệu đều sẽ được bảo vệ một cách tối đa nhờ vào hệ thống kiến trúc tập trung của mạng. Thông qua đó, nó sẽ giúp người dùng kiểm soát truy cập để chỉ có những ai được cấp quyền truy cập thì mới được thực hiện các thao tác cần thiết.
- **Khả năng mở rộng:** Mô hình mạng kết nối Client Server có khả năng mở rộng vô cùng tốt. Chỉ cần người dùng cần sử dụng bất cứ lúc nào thì họ cũng có thể tăng được số lượng tài nguyên của mình. Ví dụ như số Client hoặc Server. Nhờ đó mà chúng ta có thể tăng kích thước của Server một cách dễ dàng mà không bị gián đoạn nhiều.
- **Khả năng truy cập:** Hoàn toàn không hề có sự phân biệt giữa các vị trí hay nền tảng với nhau. Tất cả mọi Client đều có khả năng đăng nhập được vào hệ thống mạng máy tính. Điều này sẽ giúp cho tất cả các nhân viên đều có thể truy cập thông tin của công ty một cách dễ dàng mà không cần phải dùng một terminal mode hoặc một bộ xử lý nào khác.

Nhược điểm:

- **Tắc nghẽn lưu lượng:** Trong trường hợp có quá nhiều Client tạo request từ cùng một Server thì nó có thể sẽ làm cho kết nối chậm hơn. Trong trường hợp xấu nhất còn có thể xuất hiện hiện tượng crash. Khi một server bị quá tải thì sẽ tạo ra nhiều vấn đề khi truy cập thông tin.

- **Độ bền:** khi Server chính xảy ra sự cố hoặc bị nhiễu thì cũng đồng nghĩa với việc toàn bộ hệ thống mạng sẽ bị gián đoạn.
- **Chi phí cao:** Vì các hệ thống mạng có sức mạnh rất lớn cũng đồng nghĩa với việc giá để chỉ cho việc này là rất đắt. Chính vì vậy, không phải ai cũng có khả năng chi trả và sử dụng.
- **Cần bảo trì thường xuyên:** Khi các Server thực hiện triển khai để làm việc thì nó cũng sẽ hoạt động một cách không ngừng nghỉ. Điều này đồng nghĩa với việc chúng ta cần phải quan tâm đến việc bảo trì hệ thống đúng mức. Khi xảy ra bất cứ vấn đề gì cũng cần phải giải quyết ngay lập tức.

2.2.2 Peer-to-Peer

P2P – Peer-to-peer (mạng ngang hàng) là một mô hình mạng phi tập trung với các bên có các cấu trúc phiên giao tiếp giống nhau. Trong đó, mỗi nút hoạt động giống như một máy khách và máy chủ của hệ thống cho phép chia sẻ các phương tiện truyền thông với nhau dễ dàng hơn, nhanh chóng hơn. P2P có nhiều tính năng như cung cấp môi trường tính toán song song, lưu trữ phân tán và định tuyến ẩn danh lưu lượng mạng.

Trong kiến trúc P2P thì mỗi máy tính sẽ có các nhiệm vụ và tính năng hoạt động như nhau. Các máy tính sẽ kết nối trực tiếp với nhau tạo thành một nhóm làm việc nhỏ có nhiệm vụ chia sẻ tệp, máy in và truy cập Internet. Vì vậy mà P2P được sử dụng phổ biến trong phạm vi làm việc nhỏ như gia đình, văn phòng hoặc trường học có các PC hoạt động như một máy trạm độc lập. Cho phép lưu trữ các dữ liệu trên ổ cứng riêng và có khả năng chia sẻ các dữ liệu đó tới các PC khác trên cùng một mạng.

Ưu điểm:

- Không cần sử dụng tới máy chủ.
- Mỗi một thiết bị máy tính là một người dùng quản lý riêng.
- Không yêu cầu bất kỳ các kiến thức kỹ thuật chuyên ngành phức tạp nào.
- Một mạng P2P thích hợp với môi trường gia đình và doanh nghiệp nhỏ.
- Sử dụng ít lưu lượng truy cập mạng.

Nhược điểm:

- Các thông tin trên máy không thể thực hiện sao lưu tập trung.
- Việc cho phép truy cập cùng một lúc bởi nhiều thiết bị máy tính làm giảm hiệu suất hoạt động.
- Các tệp không được sắp xếp khoa học mà được lưu trữ trên máy tính cá nhân gây khó khăn trong việc xác định vị trí của chúng.
- Việc đảm bảo an toàn cho hệ thống mạng là việc của tất cả người dùng.
- Chỉ cung cấp một số quyền cơ bản và không có bảo mật nâng cao.

3 Mô tả chức năng

3.1 Tính năng cơ bản của server và client

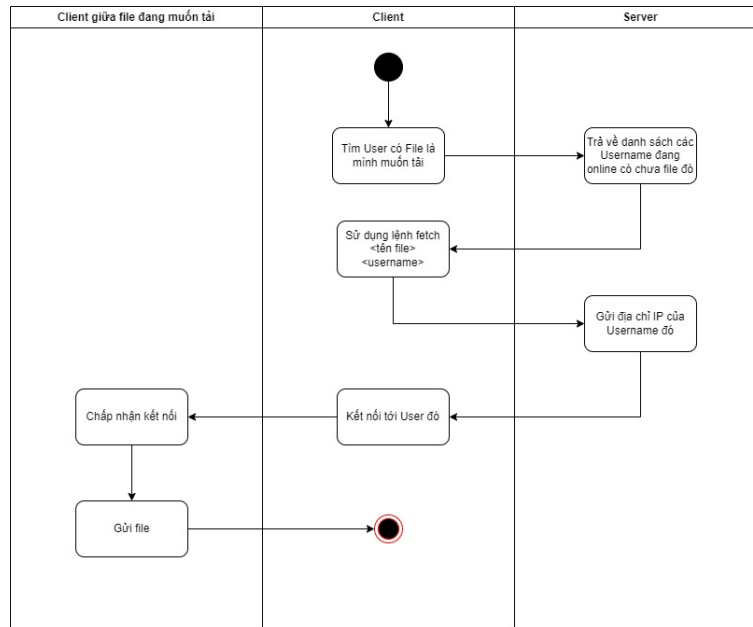
Server: Server của hệ thống này không có chức năng lưu trữ file nhưng vẫn có các chức năng quản lý và có khả năng cung cấp dịch vụ cho client như một server bình thường. Khi khởi động server sẽ lắng nghe các kết nối của các client, sau đó tạo cho mỗi user một luồng thực thi riêng rồi sẽ quản lý các tác vụ đăng nhập/đăng ký của người dùng. Sau khi người dùng đã login vào hệ thống, server chờ cho client thực hiện các lệnh như nếu user muốn in ra các file đang được publish thì Server sẽ tìm trong database xong sau đó gửi toàn bộ thông tin cho user, còn nếu như user muốn file một file từ ai đó thì Server cũng sẽ trao địa chỉ IP của người đó cho user, rồi 2 client này tự trao đổi file với nhau và Server sẽ không can thiệp vào nữa. Bên cạnh việc cung cấp các dịch vụ cho client, server cũng có các hàm riêng dành cho việc quản lý như có thể kiểm tra user nào online/offline, xem tất cả các file mà user publish,...

- Server có thể xem tất cả các file mà mỗi client đang sử dụng
- Server có thể kiểm tra xem Client đó còn online hay đã Offline
- Server cũng có thể liệt kê các User và file của họ đang publish
- Server cũng có thể thấy địa chỉ IP của các User đang kết nối với mình
- Server có thể gửi thông tin như cho các client

Client: Về phía của client, client khi vào ứng dụng thì phải login/register, nếu là người dùng mới thì phải tạo thư mục repository để lưu trữ ngay sau đó. Sau đã có repository, user sẽ có thể thực hiện các truy vấn như: xem các file nào mà mình có thể download, publish file cho các người dùng khác có thể tải, gửi file cho các người dùng khác, cũng như là download file từ người dùng khác về máy của mình.

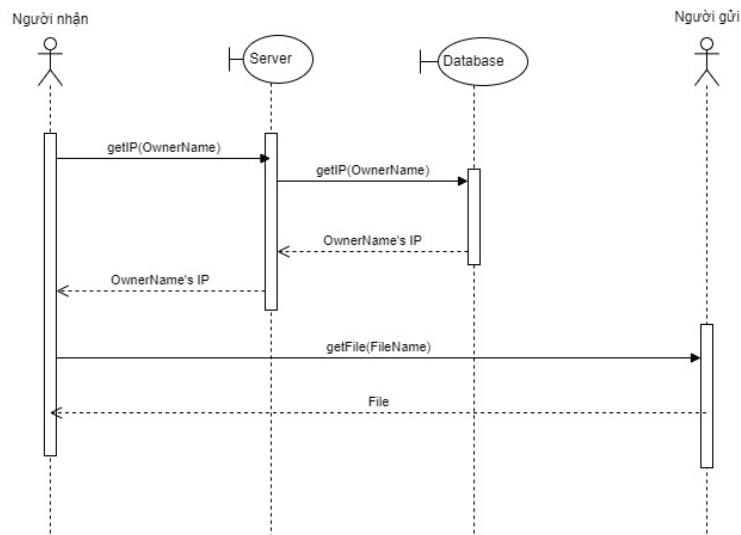
- Client đăng kí và đăng nhập vô tài khoản của mình
- Client tạo local repository trên máy chia sẻ cũng như tải các file của người khác về đó
- Client cũng có thể xem các User khác đang publish file gì
- Client có thể tìm ai là chủ sở hữu của file mà mình đang cần
- Client có thể tải file mà người khác publish về máy của mình

3.2 Activity Diagram



Hình 1: Fetch Activity Diagram

3.3 Sequence Diagram



Hình 2: Fetch Sequence Diagram

3.4 Các hàm chức năng

3.4.1 Server



Specific functions	Description
discover <username>	Hiện tất cả các file trong local repo của <username>
ping <username>	Kiểm tra <username> có đang kết nối với server hay không
listUser	Liệt kê tất cả các user
listUser -o	Liệt kê tất cả các user đang online
listUser -a	Liệt kê tất cả các user đang online kèm tất cả các file trong local repo của user đó
getUserInform <username>	Lấy địa chỉ IP của <username>
getOwner <filename>	Liệt kê tất cả các users chứa file tên <filename> gửi về phía client
sendOwnerInform <username1> <username2>	Gửi địa chỉ IP của <username1> cho <username2>

3.4.2 Client

Specific functions	Description
register <username> <password>	Đăng ký tài khoản với tên đăng nhập <username> và mật khẩu <password>
login <username> <password>	Đăng nhập tài khoản với tên đăng nhập <username> và mật khẩu <password>
create repository <path>	Tạo local repo tên repository ở máy của user
publish <path_to_filename1> <filename2>	Thêm file có tên <filename1> từ trên máy của user vào local repo dưới tên <filename2>
listUser -a	Liệt kê tất cả các user đang online kèm tất cả các file trong local repo của user đó
findOwner <filename>	Xuất danh sách các user có <filename> trên local repo
fetch <filename> <username>	Download <filename> từ <username> về local repo

3.5 Luồng thực thi cho từng hàm

3.5.1 Server

getOwner <filename>: Giao thức TCP được sử dụng để truyền thông địa chỉ IP của người dùng.

Luồng thực thi:

- Khi nhận được yêu cầu tìm file từ phía client.
- Server sẽ truy vấn trong database và gửi trả danh sách các users chứa file mà client cần tìm thông qua giao thức TCP.

sendOwnerInform <username1> <username2>: Giao thức TCP được sử dụng để truyền thông địa chỉ IP của người dùng.

Luồng thực thi:

- Server xác định địa chỉ IP của người dùng <username1> thông qua database.
- Server gửi địa chỉ IP này cho người dùng <username2> qua giao thức TCP.

3.5.2 Client

register <username> <password>: Giao thức TCP được sử dụng để đăng ký tài khoản và truyền thông tin đăng ký.

Luồng thực thi:

- Client gửi yêu cầu "register <username> <password>" đến Server.
- Sau đó server xác nhận và gửi kết quả đăng ký ("successful" hoặc "failed") trở lại cho Client.

login <username> <password>: Giao thức TCP được sử dụng để đăng nhập và truyền thông tin đăng nhập.

Luồng thực thi:

- Client gửi yêu cầu "login <username> <password>" đến Server.
- Server xác nhận thông tin đăng nhập và gửi kết quả đăng nhập ("successful" hoặc "failed") trở lại cho Client.

create repository <path>: *Luồng thực thi:*

- Client tạo repository trên máy của mình rồi gửi path của repo cho server quản lý.
- Server nhận path của repository và lưu trữ vào database.

publish <path_to_filename1> <filename2>:

Luồng thực thi:

- Client sao chép file từ đường dẫn tới filename1 trên máy client vào repository đã tạo với tên mới là <filename2>, rồi gửi tên <filename2> để server biết client vừa publish file này và sẵn sàng cho người khác download.
- Nhận được tên file sẽ lưu vào trong database.

findOwner <filename>: Giao thức TCP được sử dụng để truyền thông tin về tệp cần tìm.

Luồng thực thi:

- Client gửi yêu cầu "findOwner <filename>" đến Server..
- Server tìm kiếm và liệt kê người dùng chứa tệp <filename> và gửi danh sách trở lại cho Client.

fetch <filename> <username>: Giao thức TCP được sử dụng để tải tệp từ người dùng khác và truyền thông tin tải tệp.

Luồng thực thi:

- Client gửi yêu cầu "fetch <filename> <username>" đến Server qua giao thức TCP.
- Server sẽ kiểm tra filename và username có hợp lý không, rồi sau đó gửi trả địa chỉ IP của username cho client đang yêu cầu.
- Sau khi nhận được địa chỉ IP từ server, client sẽ kết nối tới người dùng đang chứa file muốn tải (tạm gọi là client B).
- Client B sau đó sẽ gửi file đó về lại cho client mà thực hiện truy vấn.

4 Manual

4.1 Server Side

Trước tiên chúng ta cần chạy server bằng command trên terminal hoặc cmd:

python server.py

Sau đó chúng ta nhập host và port. Trong đó, host đối với trong quy mô bài tập lớn này sẽ là địa chỉ IP public của máy tính chạy file server.py và port sẽ tùy ý. Sau khi khởi động server sẽ có giao diện như sau:



Hình 1: Giao diện Server Side khi vừa khởi động

Quản trị server có thể nhấn vào nút "Chạy" để biết danh sách các lệnh có thể thực thi:



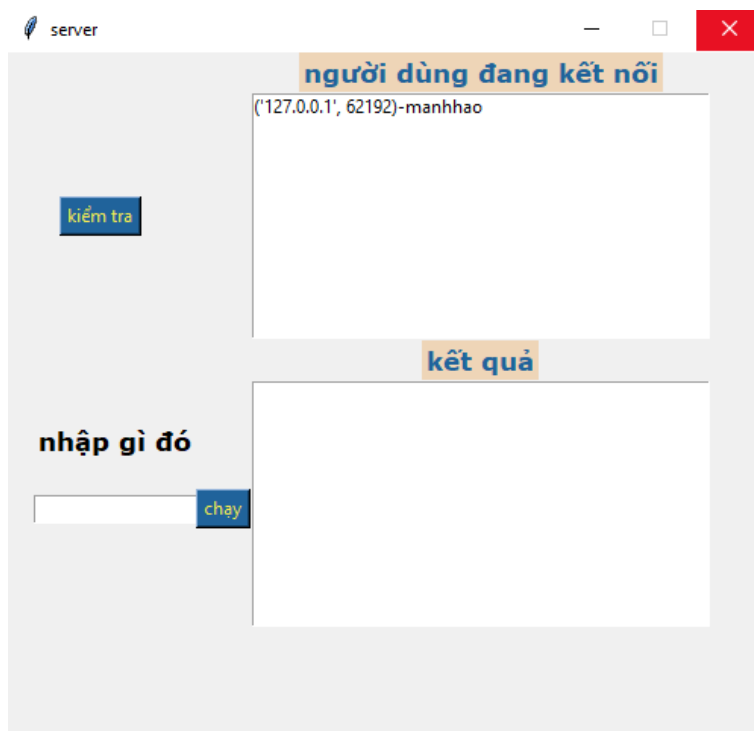
Hình 2: Server với danh sách các lệnh hợp lệ

Sau khi nhập lệnh hợp lệ và chạy (ví dụ lệnh "listUser"), kết quả sẽ được trả vào phần "Kết quả" (các lệnh khác đều tương tự):



Hình 3: Giao diện Server với lệnh "listUser"

Khi quản trị server muốn biết client nào đang kết nối tới server, quản trị viên nhấn vào nút "Kiểm tra", danh sách người dùng đang kết nối sẽ được in ra bảng kế bên:



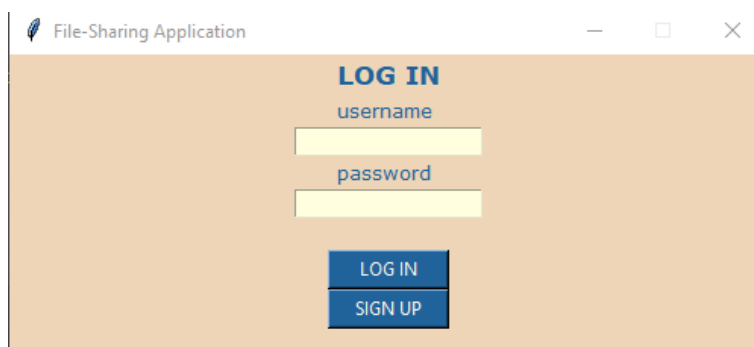
Hình 4: Server kiểm tra người dùng nào đang kết nối

4.2 Client Side

Khi server đã khởi động, ta chạy client bằng cách nhập lệnh:

python client.py

Và tương tự như server, chúng ta sẽ nhập tiếp host và port, với host là địa chỉ IP public của máy tính chạy file server.py và port sẽ là port của server đã set ở trên. Sau khi đã khởi động Client sẽ có giao diện như sau:



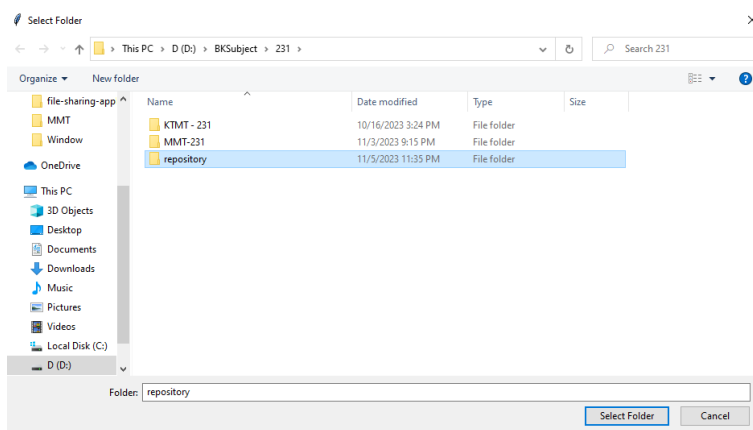
Hình 5: Giao diện đăng nhập/ đăng ký

Người dùng đăng nhập bằng username và password đã tạo từ trước và nhấn "LOG IN". Nếu chưa có, người dùng có thể nhập username và password của tài khoản muốn tạo và nhấn "SIGN UP". Sau khi đăng nhập/ đăng ký thành công sẽ có giao diện:



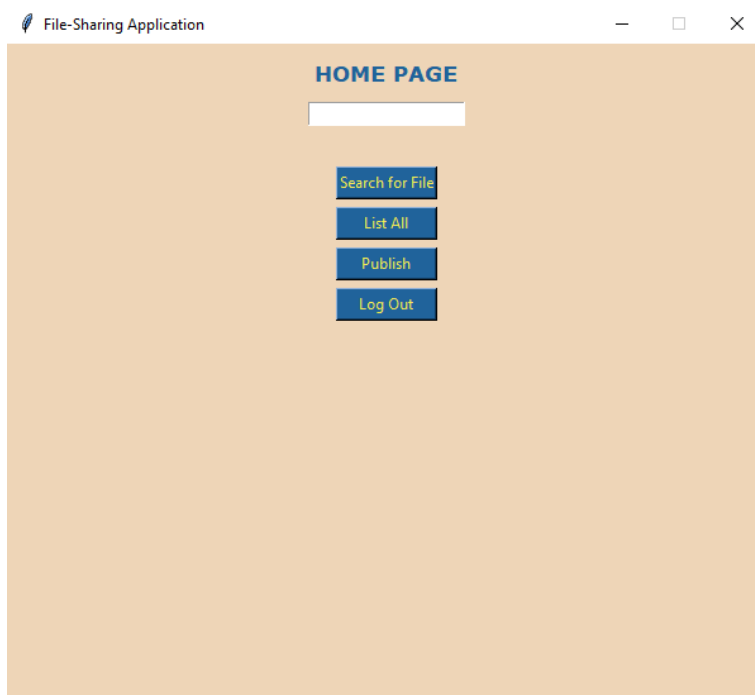
Hình 6: User sau khi đăng nhập/ Đăng ký thành công

Người dùng nhấn chọn "Chọn thư mục" và chọn tiếp một folder để làm local repository, nếu chưa có thì người dùng có thể nhấn chuột phải tạo một folder mới:



Hình 7: Chọn thư mục làm Local repository

Sau khi chọn thư mục làm local repository thành công, client sẽ quay lại giao diện chính:



Hình 8: Màn hình chính

Nếu người dùng muốn publish file từ máy người dùng vào local repository vừa tạo, người dùng nhấn chọn "Publish", giao diện sẽ hiển thị đường dẫn tới local repository vừa tạo, sau đó người dùng chọn file và đường dẫn tới file đó sẽ được in ra trên màn hình:

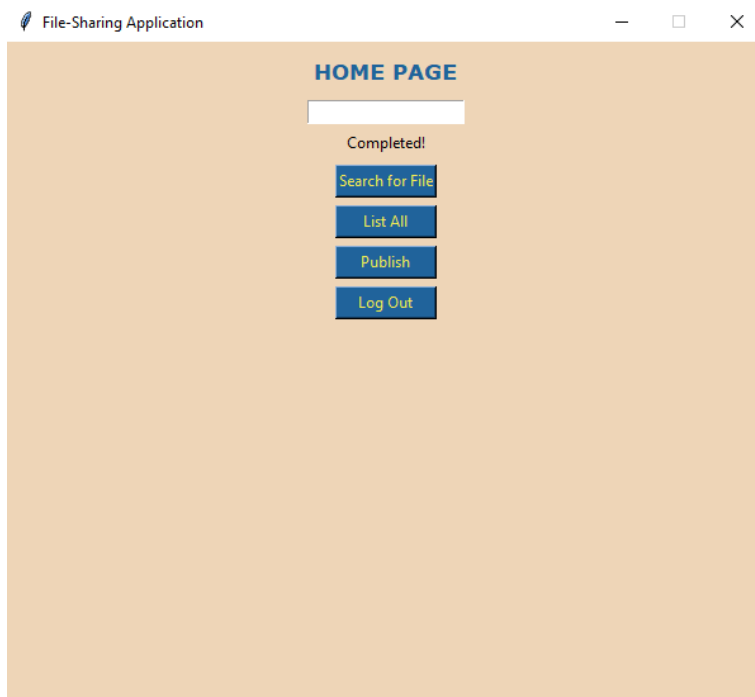


Hình 9: Đường dẫn tới local repository



Hình 10: Đường dẫn tới file vừa chọn

Sau khi chọn giao diện sẽ như hình dưới:

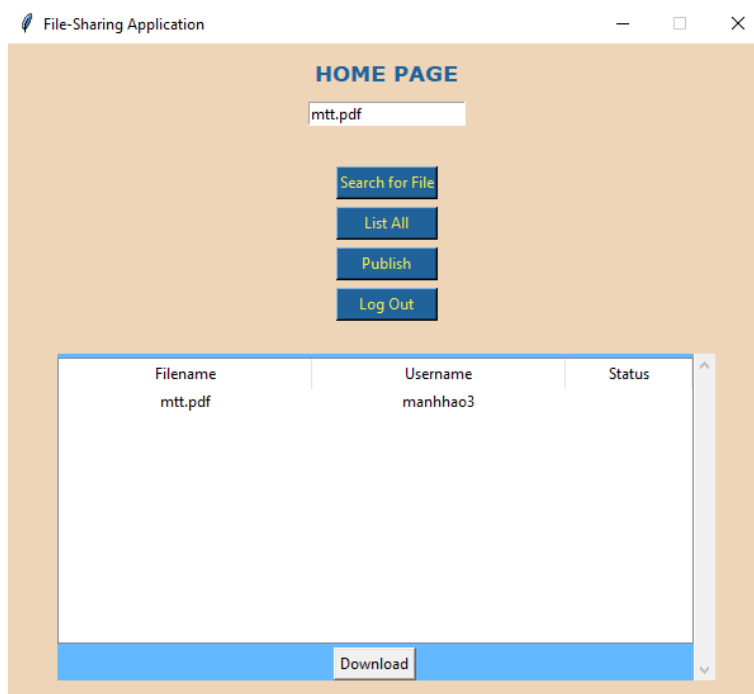


Hình 11: Thêm file vào local repository thành công

Nếu người dùng muốn tìm kiếm file muốn tải từ các client khác, người dùng nhập tên file cần tìm và nhấn "Search for file", nếu thất bại sẽ in ra thông báo thất bại, nếu thành công sẽ trả về bảng chứa danh sách các user có file cần tải, người dùng nhấn download và file sẽ được tải về máy người dùng



Hình 12: Tìm kiếm file thất bại



Hình 13: Tìm kiếm và tải file thành công



5 Source code

Link github chứa source code của nhóm: <https://github.com/mhaoo/file-sharing-application>

6 Tài liệu tham khảo

References

- [1] MySQL Workbench Tutorial: <https://www.oracle.com/technetwork/mysql-hands-on-lab-403032.pdf>
- [2] Python Tutorial: <https://docs.python.org/3/tutorial/index.html>
- [3] Tkinter Python Tutorial: <https://docs.python.org/3/library/tk.html>
- [4] Networking Tutorial: https://gaia.cs.umass.edu/kurose_oss/ppt.php