

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



Assignment

Computer Networking

Chủ đề 1:

Develop a network application

GVHD: BÙI XUÂN GIANG

SV thực hiện:	Nguyễn Chí Thiết	2213242
Các thành viên khác:	Nguyễn Lê Gia Kiệt	2211761
	Nguyễn Huy Hoàng	2211091

TP. HỒ CHÍ MINH, THÁNG 12/2024

Mục lục

1	GIỚI THIỆU ĐỀ TÀI	2
2	MÔ TẢ ĐỀ TÀI	2
2.1	Yêu cầu chức năng và phi chức năng của đề tài	2
3	HỆ THỐNG ỨNG DỤNG	3
3.1	Tổng quan ứng dụng	3
3.2	Kết quả cần thiết khi thực hiện xong đề tài	4
4	ĐẶC TẢ CÁC HÀM VÀ PHƯƠNG THỨC ĐÃ ĐƯỢC SỬ DỤNG	4
4.1	Phía server	4
4.2	Phía khách hàng	5
5	USE CASE	7
5.1	Đăng nhập	7
5.2	Kết nối với Server	8
5.3	Upload tập tin	9
5.4	Download tập tin	10
5.5	Server	11
6	ACTIVITY DIAGRAM	12
6.1	Đăng nhập	12
6.2	Kết nối với Server	13
6.3	Upload tập tin	15
6.4	Download tập tin	16
6.5	Server	17
7	GIAO DIỆN CỦA SẢN PHẨM	18
7.1	Peer	18
7.1.1	Giao diện bắt đầu ứng dụng	18
7.1.2	Giao diện đăng nhập	19
7.1.3	Giao diện kết nối với Server	20
7.1.4	Giao diện chính của ứng dụng	21
7.1.5	Giao diện upload tập tin	22
7.1.6	Giao diện download tập tin	23
7.2	Server	24
7.2.1	Giao diện bắt đầu ứng dụng	24
7.2.2	Giao diện đăng nhập	25
7.2.3	Giao diện chính của ứng dụng	26
7.2.4	Giao diện hiển thị các tập tin	27

Tài liệu tham khảo	28
--------------------	----

1 GIỚI THIỆU ĐỀ TÀI

Môn học Mạng máy tính đóng một vai trò quan trọng trong lĩnh vực công nghệ thông tin, nhờ môn học này chúng ta hiểu cách kết nối, chia sẻ giữa các thiết bị điện tử. Không những thế, môn học không chỉ giúp sinh viên nhận thức vai trò quan trọng của mạng máy tính trong cuộc sống hiện đại và trong phát triển của công nghệ thông tin.

Trong thời đại kỹ thuật số hiện nay, việc chia sẻ dữ liệu giữa các thiết bị ngày càng trở nên phổ biến và cần thiết hơn bao giờ hết. Để đáp ứng nhu cầu này mà không cần sự trung gian của máy chủ trung gian, việc phát triển ứng dụng chia sẻ file ngang hàng (P2P) trên mạng máy tính đang trở thành một ưu tiên quan trọng trong lĩnh vực công nghệ thông tin.

Báo cáo này sẽ tập trung vào quá trình thiết kế và triển khai của một ứng dụng P2P cho phép người dùng truyền tải dữ liệu trực tiếp giữa các thiết bị thông qua mạng máy tính. Mục tiêu của ứng dụng là tạo ra một môi trường truyền tải dữ liệu an toàn, nhanh chóng và hiệu quả, mà không cần sự can thiệp của máy chủ trung gian. Trên cơ sở những tính năng và yếu tố trên, báo cáo sẽ trình bày quy trình thiết kế và triển khai của ứng dụng P2P, cùng với các thách thức và giải pháp trong quá trình phát triển.

2 MÔ TẢ ĐỀ TÀI

Trong bài tập lớn này, các tính năng chính của ứng dụng bao gồm:

- Kết nối trực tiếp giữa các thiết bị: Ứng dụng cho phép người dùng thiết lập kết nối trực tiếp với các thiết bị khác trong cùng mạng máy tính, mà không cần thông qua máy chủ trung gian.
- Chia sẻ dữ liệu một cách an toàn: Dữ liệu được truyền tải giữa các thiết bị thông qua giao thức mã hóa để đảm bảo tính riêng tư và bảo mật.
- Quản lý tải xuống và tải lên: Ứng dụng cung cấp các tính năng quản lý tải xuống và tải lên, cho phép người dùng dễ dàng theo dõi tiến trình chia sẻ dữ liệu.
- Giao diện người dùng thân thiện: Giao diện người dùng được thiết kế đơn giản và dễ sử dụng, giúp người dùng dễ dàng truy cập và sử dụng các tính năng của ứng dụng.
- Tích hợp tính năng tìm kiếm: ứng dụng cung cấp các công cụ tìm kiếm và khám phá dữ liệu trên mạng máy tính, giúp người dùng dễ dàng tìm kiếm và truy cập vào các tệp dữ liệu phù hợp.

2.1 Yêu cầu chức năng và phi chức năng của đề tài

2.1.1. Giao tiếp với máy chủ trung tâm thông qua máy chủ trung gian (Tracker)

- Gửi thông tin về các tập tin có sẵn trong kho lưu trữ cục bộ của máy khách đến máy chủ trung tâm thông qua giao thức TCP/IP.
- Yêu cầu tập tin từ máy chủ trung tâm khi máy khách cần tập tin mà nó không có trong kho lưu trữ cục bộ của mình.

2.1.2. Tải xuống từ các máy chủ phụ (Peers)

- Kết nối với nhiều máy chủ phụ cùng một lúc để tải xuống từ chúng.
- Yêu cầu các khối dữ liệu từ máy chủ phụ và tải xuống các khối đó.

2.1.3. Tải lên cho các máy khách phụ (Peers)

- Cho phép các máy khách khác kết nối và tải tập tin từ máy chủ của mình sau khi tải xuống xong.
- Thực hiện phương pháp "tit-for-tat" để chia sẻ tập tin với các máy khách khác.

2.1.4. Giao diện người dùng

- Cho phép người dùng chọn tập tin để tải xuống, vị trí lưu trữ, và xem thống kê về tải xuống và tải lên hiện tại.
- Cung cấp giao diện dòng lệnh hoặc giao diện đồ họa để tương tác với ứng dụng.

2.1.5. Yêu cầu phi chức năng

- Ứng dụng phải hoạt động một cách ổn định và hiệu quả, có khả năng xử lý nhiều kết nối cùng một lúc.
- Đảm bảo không có lỗi cơ bản và cải thiện hiệu suất tải xuống và tải lên.

Bảo mật:

- Đảm bảo tính an toàn của dữ liệu trong quá trình tải xuống và tải lên.
- Xác thực người dùng và các máy khách để tránh các hành động gian lận và tấn công.

Tài liệu và hướng dẫn:

- Cung cấp tài liệu và hướng dẫn chi tiết cho người sử dụng và các nhà phát triển.
- Hỗ trợ các hướng dẫn về cách sử dụng ứng dụng và cách triển khai các tính năng bổ sung (nếu có).

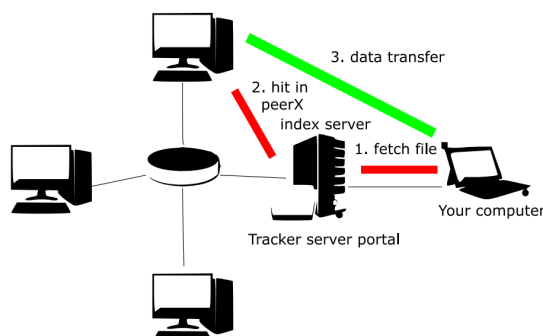
Phát triển và mở rộng:

- Duy trì và phát triển tính năng của ứng dụng sau khi hoàn thành yêu cầu cơ bản.
- Khả năng mở rộng và tích hợp các tính năng mới theo yêu cầu của người dùng và thị trường (nếu có).

3 HỆ THỐNG ỨNG DỤNG

3.1 Tổng quan ứng dụng

- Một máy chủ trung tâm giữ thông tin về các máy khách đang kết nối tới các tệp tin.
- Thông qua giao thức theo dõi (*tracker protocol*), một máy khách thông báo cho máy chủ về các tệp tin nằm trong kho lưu trữ cục bộ của nó nhưng không truyền dữ liệu tệp tin thực sự đến máy chủ.
- Khi một máy khách yêu cầu một tệp tin không thuộc về kho lưu trữ cục bộ của nó, một yêu cầu được gửi đến máy chủ.
- Nhiều máy khách có thể đang tải xuống các tệp tin khác nhau từ một máy khách mục tiêu tại một thời điểm nhất định. Điều này đòi hỏi mã máy khách phải hỗ trợ đa luồng.



Hình 1: Minh họa hệ thống chia sẻ thông tin

3.2 Kết quả cần thiết khi thực hiện xong đề tài

Ứng dụng mạng được phát triển hoàn chỉnh với khả năng chia sẻ và tải xuống tập tin thông qua giao thức BitTorrent. Giao diện người dùng được thiết kế thân thiện và dễ sử dụng, cho phép người dùng quản lý tập tin một cách dễ dàng và theo dõi tiến trình tải xuống và tải lên. Ngoài ra, ứng dụng cũng có khả năng kết nối với máy chủ trung tâm và tải xuống từ các máy chủ phụ, cũng như chia sẻ tập tin với các máy khách khác trên mạng.

Ứng dụng đảm bảo tính ổn định và hiệu suất cao, và có thể xử lý nhiều kết nối cùng một lúc và cung cấp trải nghiệm người dùng mượt mà; dữ liệu được truyền tải qua mạng được bảo mật, đảm bảo tính an toàn và bảo mật cho người dùng, cung cấp tài liệu và hướng dẫn chi tiết để người dùng có thể sử dụng và triển khai ứng dụng một cách dễ dàng.

4 ĐẶC TẢ CÁC HÀM VÀ PHƯƠNG THỨC ĐÃ ĐƯỢC SỬ DỤNG

4.1 Phía server

Máy chủ trong mã này có vai trò quản lý danh sách các tập tin và các máy khách chứa các tập tin tương ứng để phục vụ việc chia sẻ tệp tin. Vai trò cụ thể của máy chủ bao gồm:

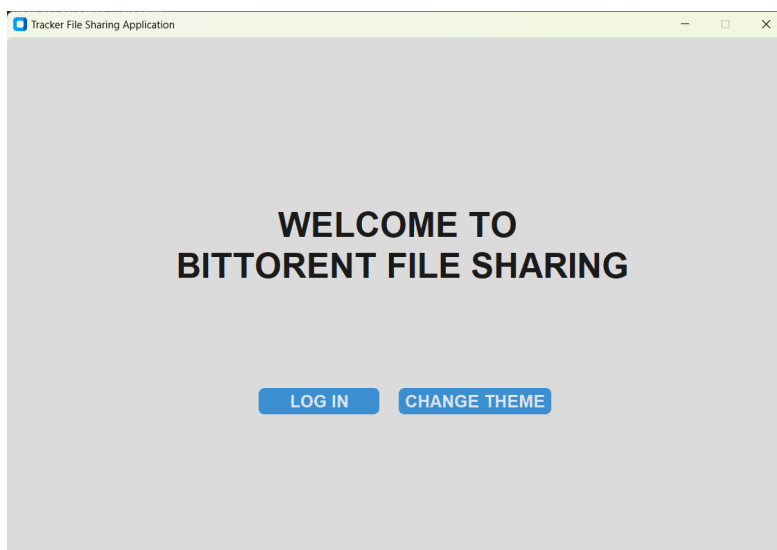
Server_FrontEnd (Giao diện Máy chủ)

- **init:** Khởi tạo giao diện người dùng của máy chủ, bao gồm các khung và nút.
- **switch_frame:** Chuyển đổi giữa các khung trên giao diện người dùng.
- **changeTheme:** Thay đổi chủ đề của giao diện.
- **initialPage:** Hiển thị giao diện bắt đầu của ứng dụng và các nút để đăng nhập hoặc thay đổi chủ đề.
- **executeLoginButton:** Hiển thị giao diện đăng nhập và kiểm tra thông tin đăng nhập.
- **check_login:** Kiểm tra thông tin đăng nhập và chuyển đến trang chính nếu thông tin hợp lệ.

- **mainPage:** Hiển thị trang chính của ứng dụng, bao gồm thông tin máy chủ và các nút điều hướng.
- **showPeers:** Hiển thị danh sách các peer kết nối với máy chủ.
- **listFilesOnSystem:** Hiển thị danh sách các tệp trên hệ thống.
- **showListFileOnSystem:** Hiển thị danh sách các tệp trên hệ thống.
- **showStatusCenter:** Hiển thị trạng thái trung tâm trên giao diện người dùng.

Server_Backend (Máy chủ Backend)

- **init:** Khởi tạo máy chủ và các biến đối tượng.
- **implementDownload:** Triển khai quá trình tải xuống tệp từ máy chủ.
- **implementListenPeer:** Lắng nghe các kết nối từ các peer.
- **implementSharing:** Triển khai quá trình chia sẻ tệp từ peer lên máy chủ.
- **threadListenPeer:** Quản lý việc lắng nghe các kết nối từ peer.

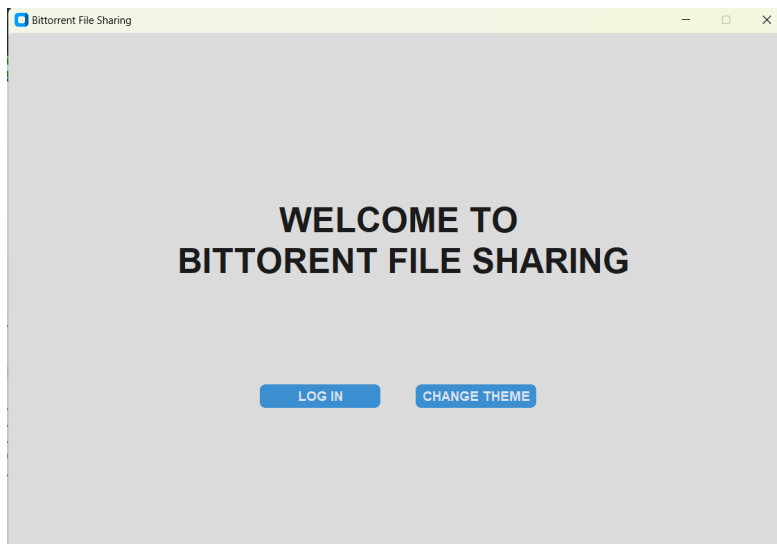


Hình 2: Giao diện của server

4.2 Phía khách hàng

Một peer có khả năng chia sẻ dữ liệu bằng cách tải xuống từ những peer khác trong mạng và cung cấp các phần của tệp đó cho những peer khác. Quá trình chia sẻ này giúp giảm tải cho máy chủ trung tâm và tăng tốc độ tải xuống tổng thể cho mỗi người dùng. Đồng thời, các peer cũng có thể tạo ra các tệp cục bộ để chia sẻ với những người dùng khác trên mạng. Hơn nữa, các peer cũng có vai trò trong việc quản lý mạng BitTorrent bằng cách trao đổi thông tin về các phần của tệp và trạng thái kết nối. Họ tham gia vào quá trình trao đổi dữ liệu thông qua các giao thức được xác định trước để đảm bảo tính hiệu quả và tin cậy của hệ thống:

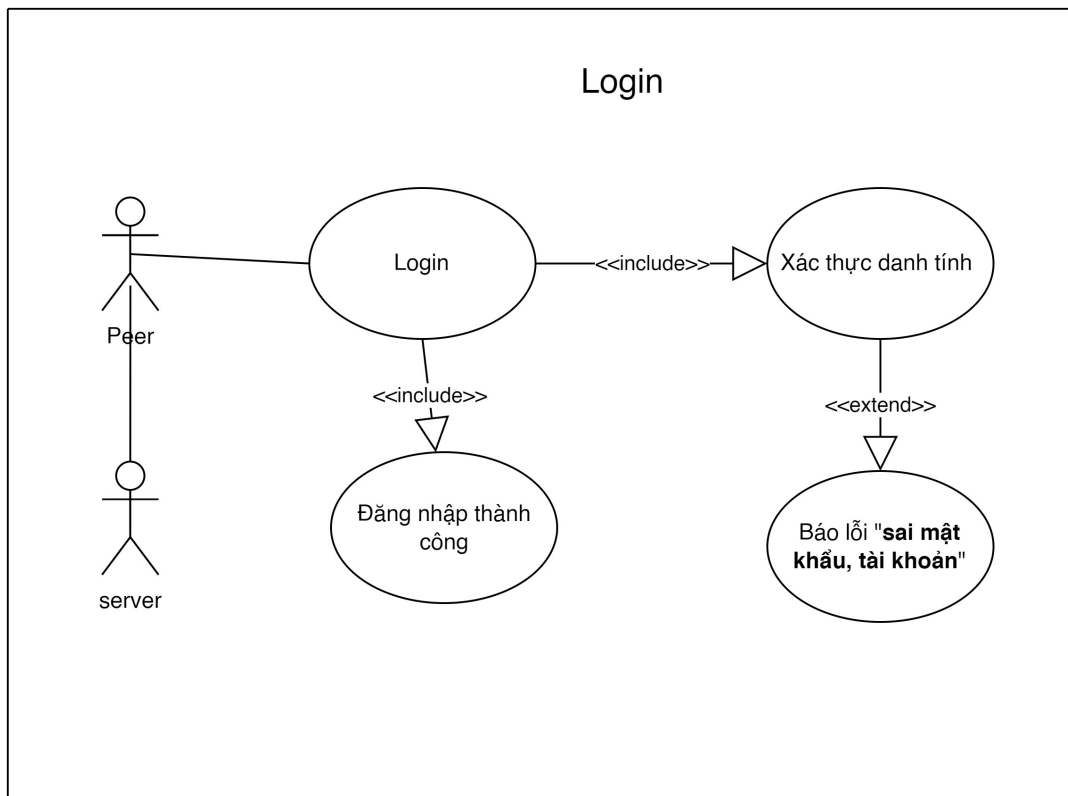
- **initialPage**: Hiển thị trang chính với nút đăng nhập và nút thay đổi chủ đề trong giao diện người dùng.
- **executeLoginButton**: Hiển thị trang đăng nhập với các trường nhập và nút xác nhận trong giao diện người dùng.
- **connectToServer**: Hiển thị trang kết nối đến máy chủ với các trường nhập cho host và cổng trong giao diện người dùng.
- **mainPage**: Hiển thị trang chính sau khi đăng nhập thành công, hiển thị thông tin về máy đồng tốt và các nút chức năng trong giao diện người dùng.
- **executeUploadButton**: Hiển thị trang tải lên với khung nhập đường dẫn và nút tải lên trong giao diện người dùng.
- **executeDownloadButton**: Hiển thị trang tải xuống với danh sách các tệp tồn tại và khung nhập tên tệp để tải xuống trong giao diện người dùng.
- **threadListenServerOrPeers**: Một luồng được tạo ra để lắng nghe các yêu cầu từ máy chủ hoặc đồng hành khác và xử lý chúng.
- **listenServerOrPeers**: Tạo một socket lắng nghe trên máy đồng hành để chấp nhận các kết nối từ máy chủ hoặc các đồng hành khác.
- **implementDownload**: Thực hiện việc tải xuống tập tin từ máy chủ hoặc các đồng hành khác, gửi yêu cầu tải xuống và nhận thông tin về tệp cần tải xuống.



Hình 3: Giao diện của các peers

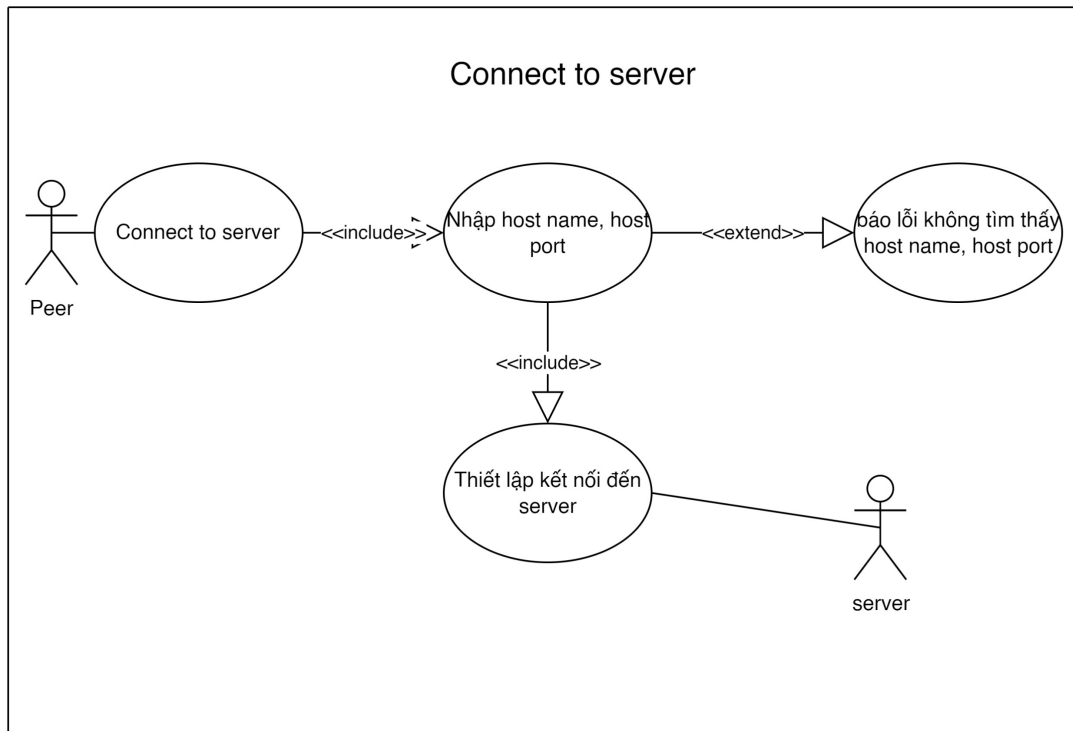
5 USE CASE

5.1 Đăng nhập



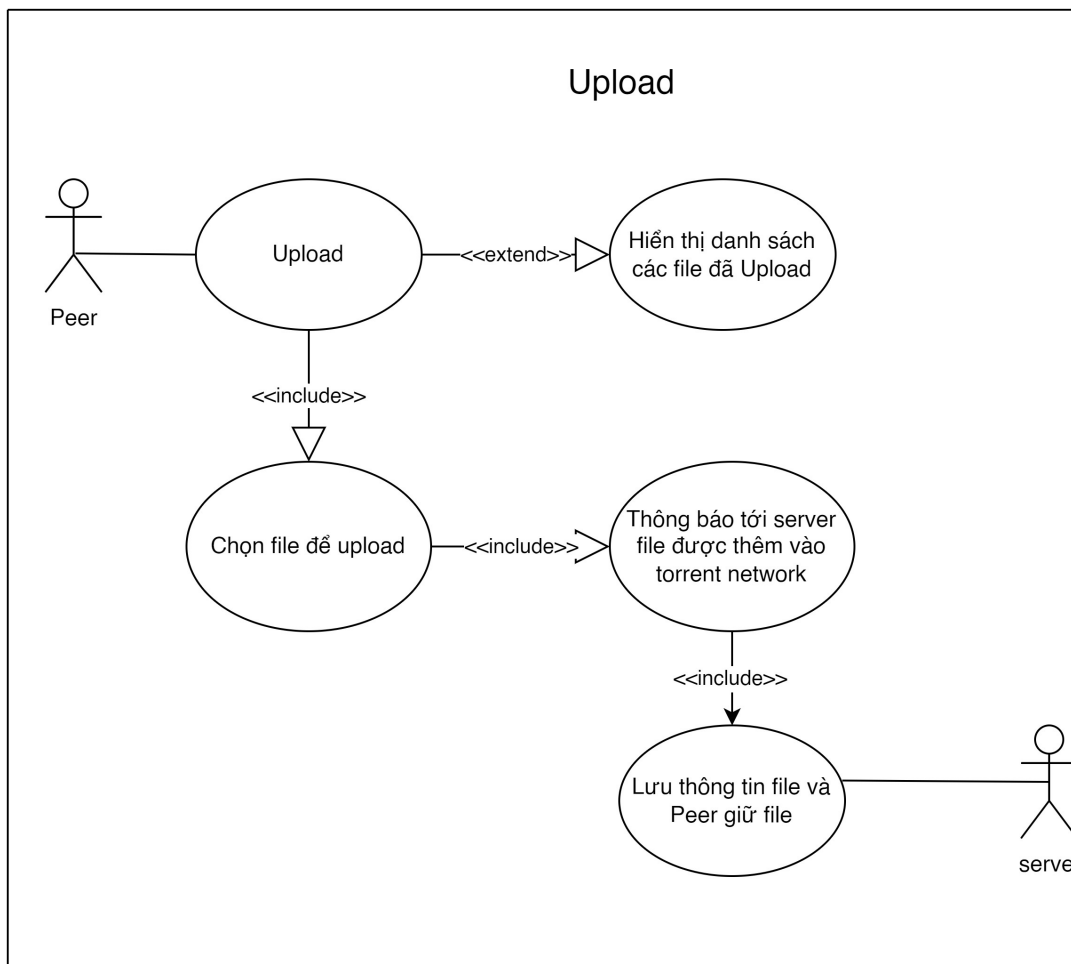
- **Precondition:** Thông tin đăng nhập tồn tại
- **Postcondition:** Danh tính người dùng được xác thực, khởi động phiên hoạt động mới và điều hướng vào giao diện chính của hệ thống.
- **Basic path:**
 1. Người dùng nhập thông tin tài khoản để đăng nhập: tên đăng nhập, mật khẩu.
 2. Hệ thống kiểm tra tính chính xác của thông tin đăng nhập
- **Exceptional path:** Tại bước thứ 2, nếu thông tin đăng nhập không hợp lệ thì sẽ hiển thị thông báo cho người dùng, đăng nhập thất bại.

5.2 Kết nối với Server



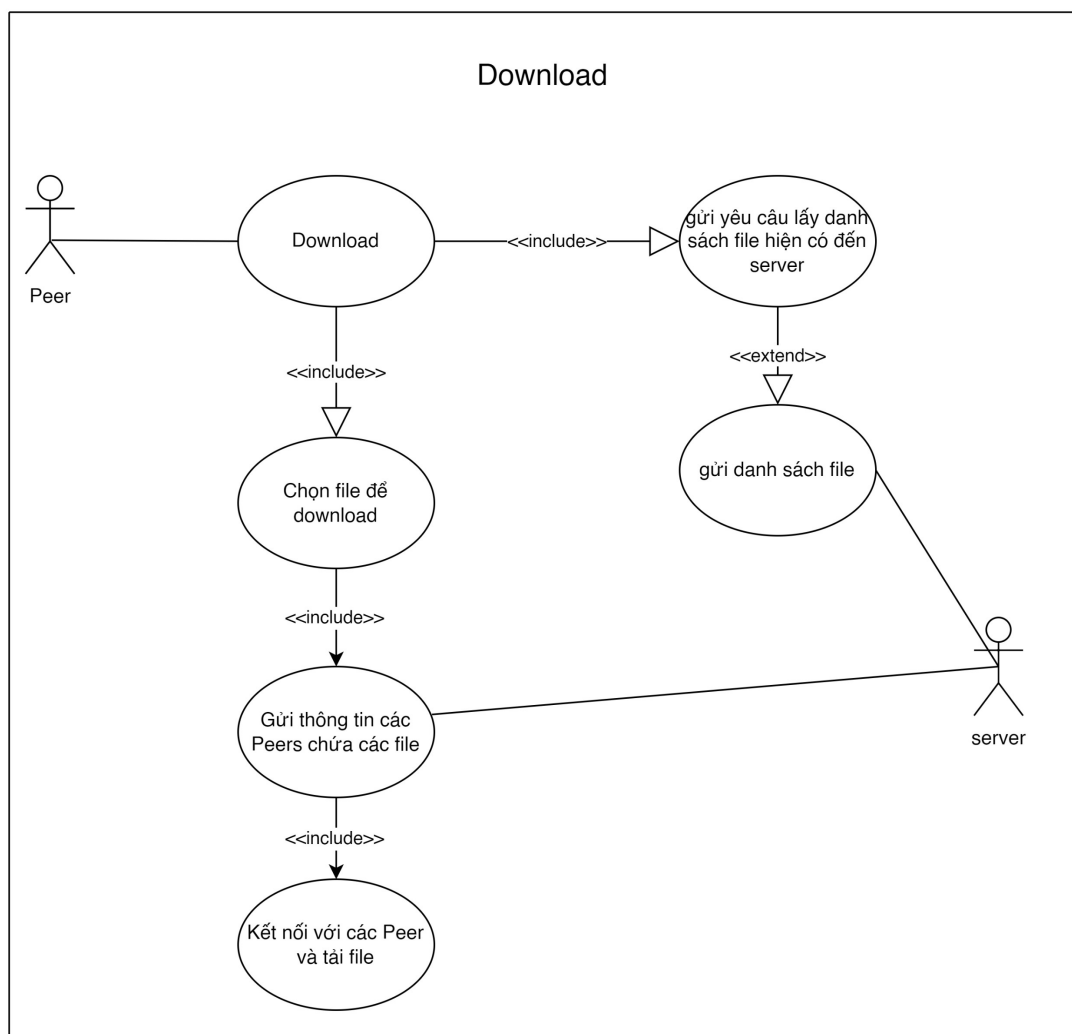
- **Precondition:** Người dùng login thành công và Server đang active
- **Postcondition:** Peer kết nối với server thành công
- **Basic path:**
 1. Người dùng nhập thông tin gồm Host name và host port.
 2. Hệ thống kiểm tra xem host name và host port có tồn tại hay không
 - Nếu đang tồn tại server khớp với thông tin trên thì peer sẽ kết nối với server
 - Nếu không tồn tại server khớp với thông tin trên thì báo lỗi không tìm thấy server

5.3 Upload tập tin



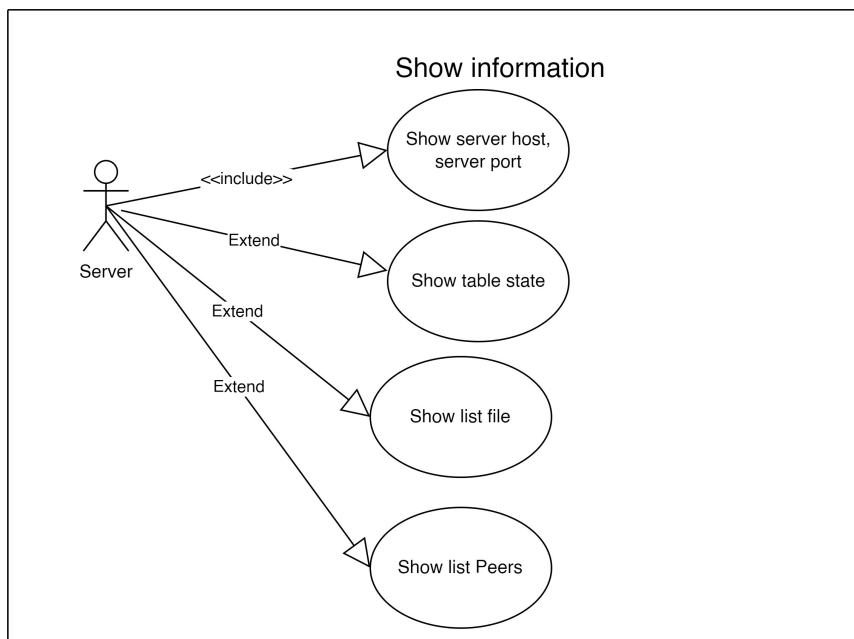
- **Precondition:** Người dùng đã kết nối với server
- **Postcondition:** File được thêm vào torrent network
- **Basic path:**
 1. Người dùng nhập đường dẫn của file cần upload
 2. Peer gửi thông tin file upload đến server
 3. server lưu thông tin file và peer đang nắm giữ file đó vào torrent

5.4 Download tập tin



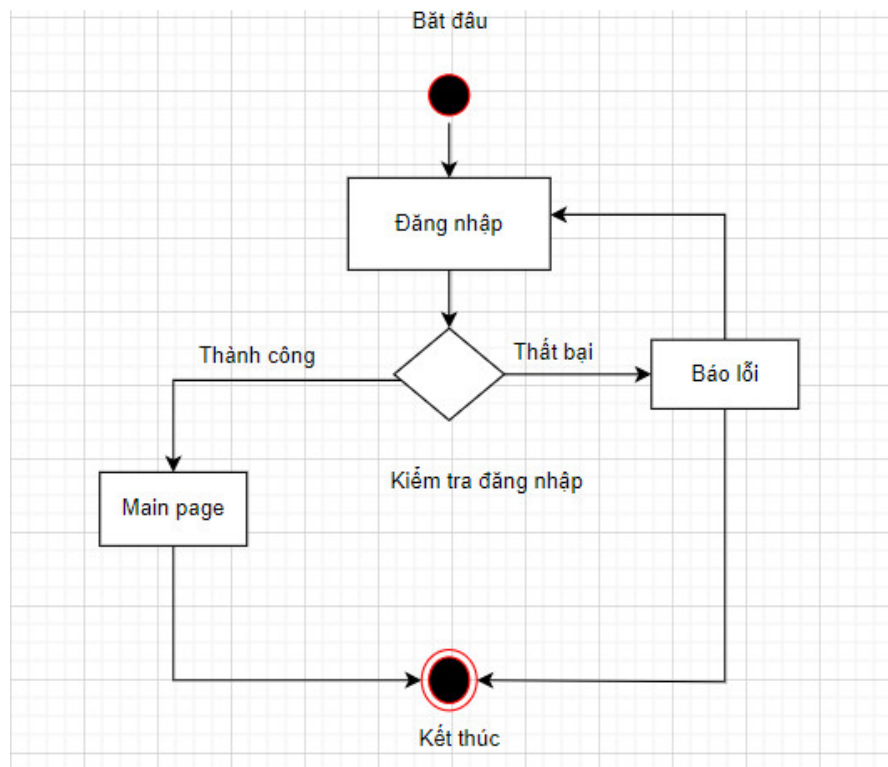
- **Precondition:** Người dùng đã kết nối với server và có file đã upload
- **Postcondition:** File được tải về local
- **Basic path:**
 1. Người dùng ấn vào nút Download
 2. Server gửi về danh sách file hiện có
 - Nếu đã có file upload thì hiển thị danh sách file hiện có, tiếp tục bước 3
 - Nếu chưa có file thì không hiển thị gì, quay lại bước 1
 3. Người dùng nhập tên file muốn download
 4. Server gửi thông tin các Peers chứa các file
 5. Kết nối với các Peers và tải file

5.5 Server



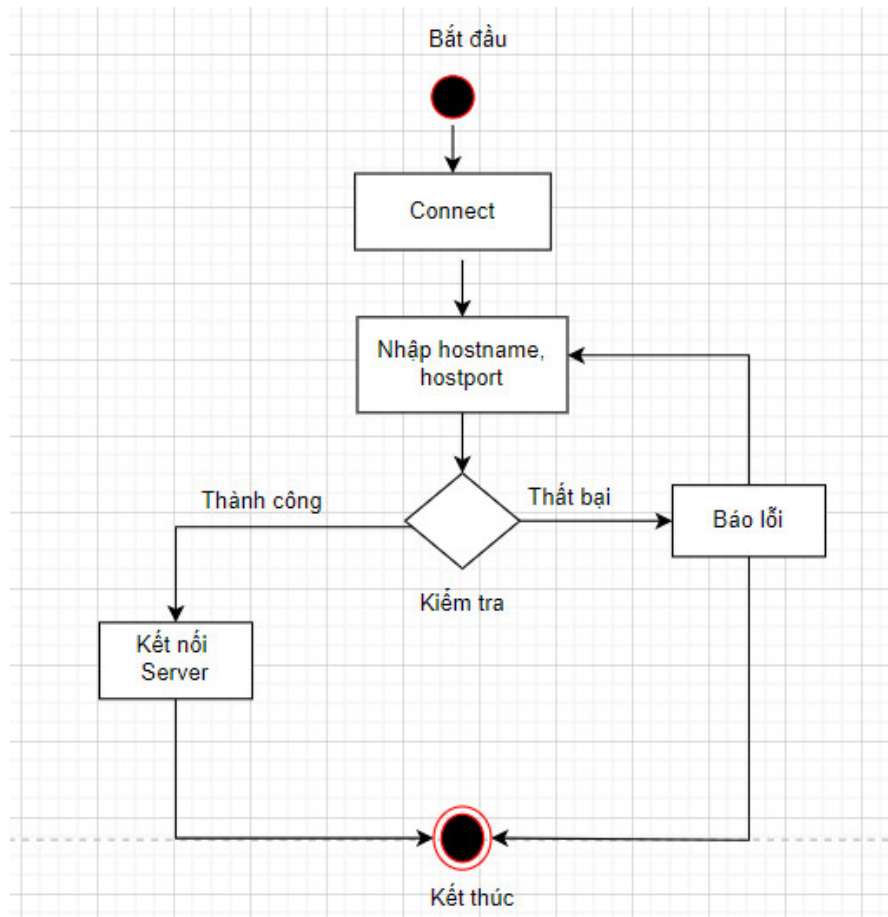
6 ACTIVITY DIAGRAM

6.1 Đăng nhập



Hình 4: Activity diagram Login

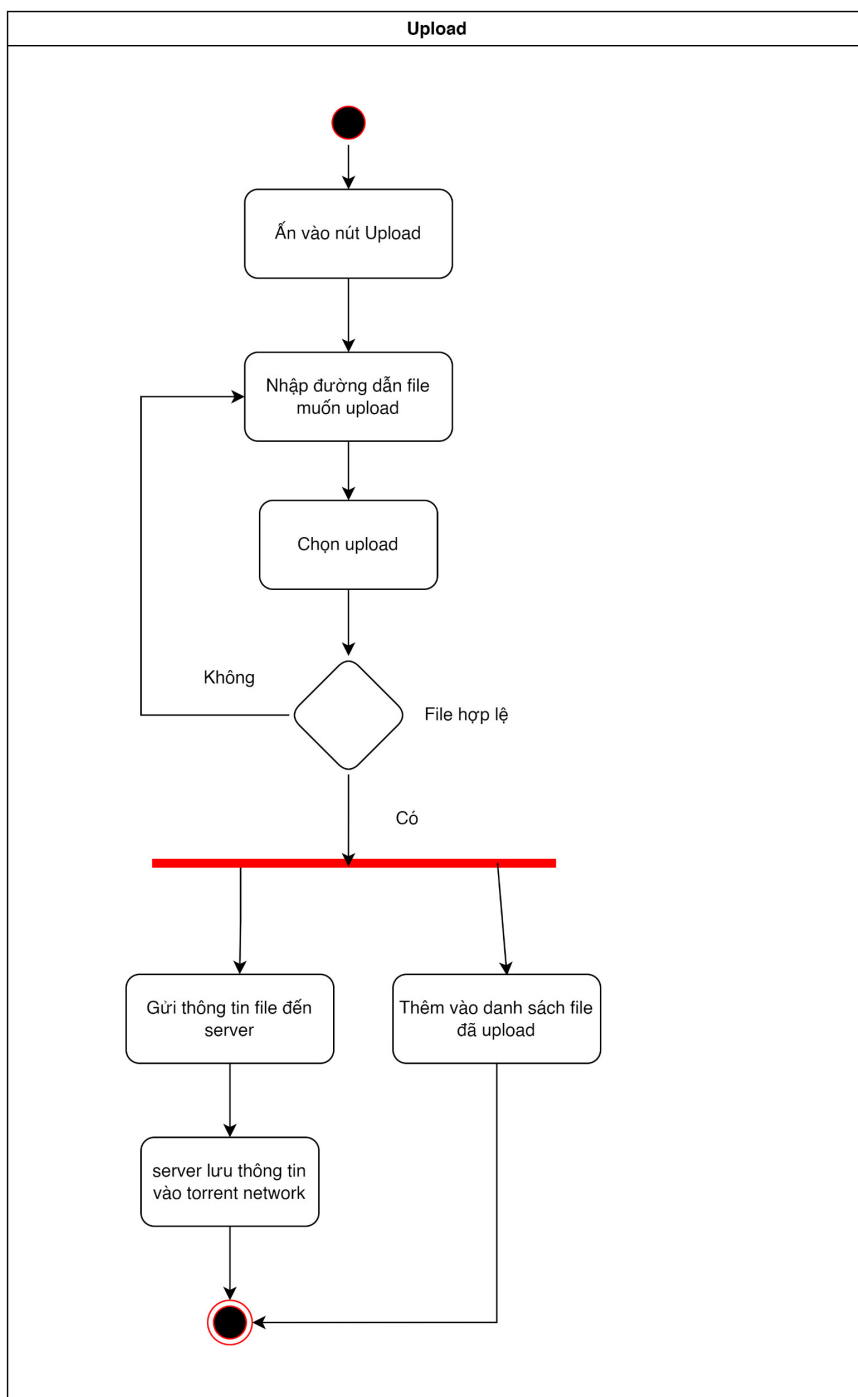
6.2 Kết nối với Server



Hình 5: Activity diagram Connect Server

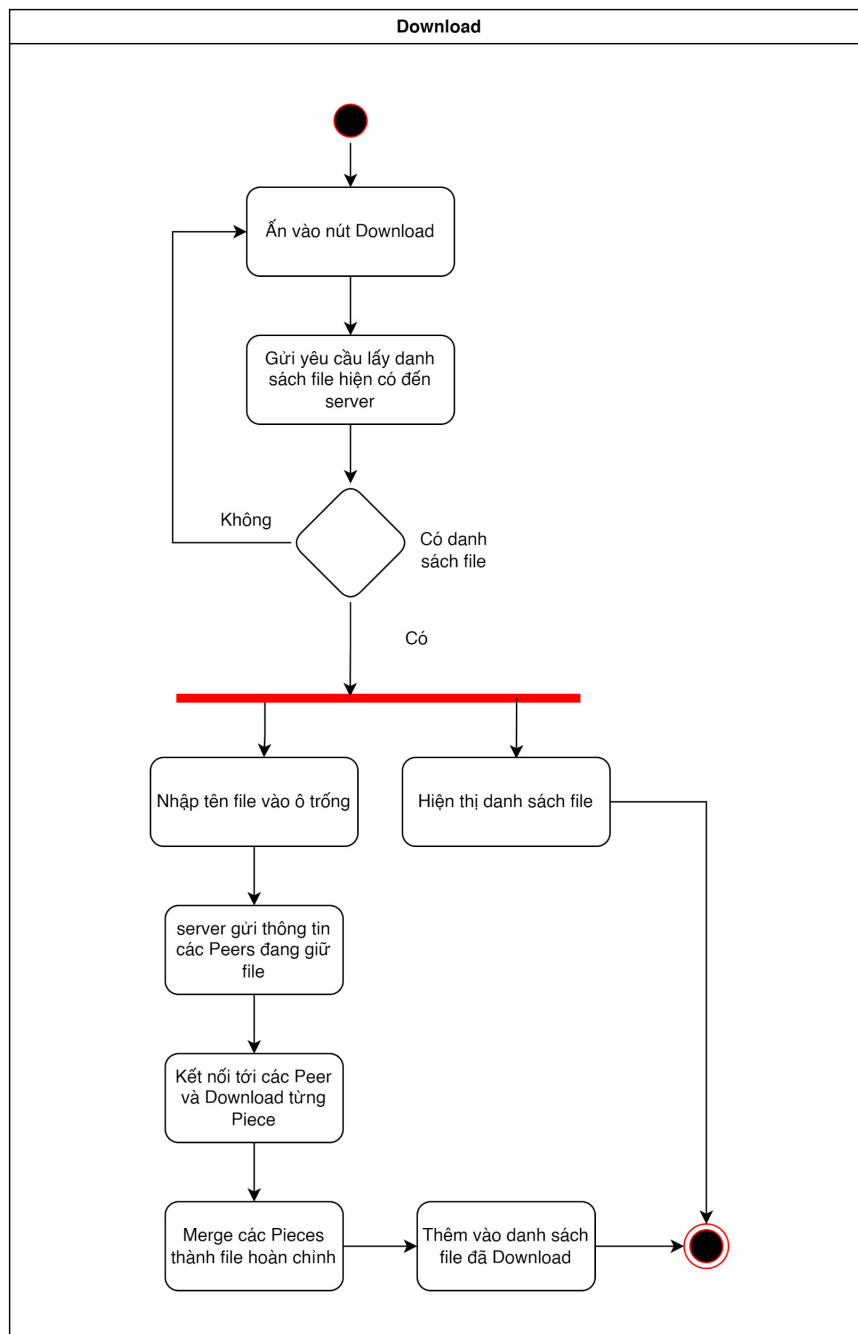


6.3 Upload tập tin



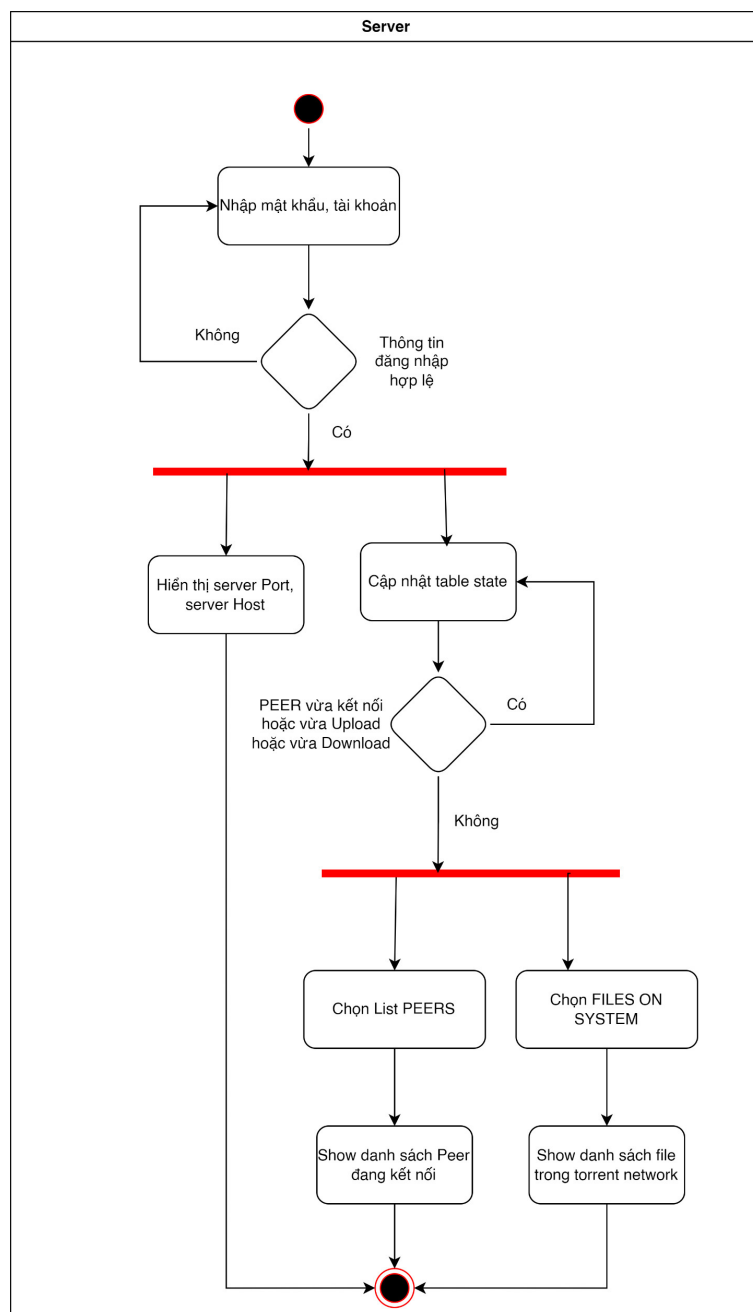
Hình 6: Activity diagram Upload

6.4 Download tập tin



Hình 7: Activity diagram Download

6.5 Server

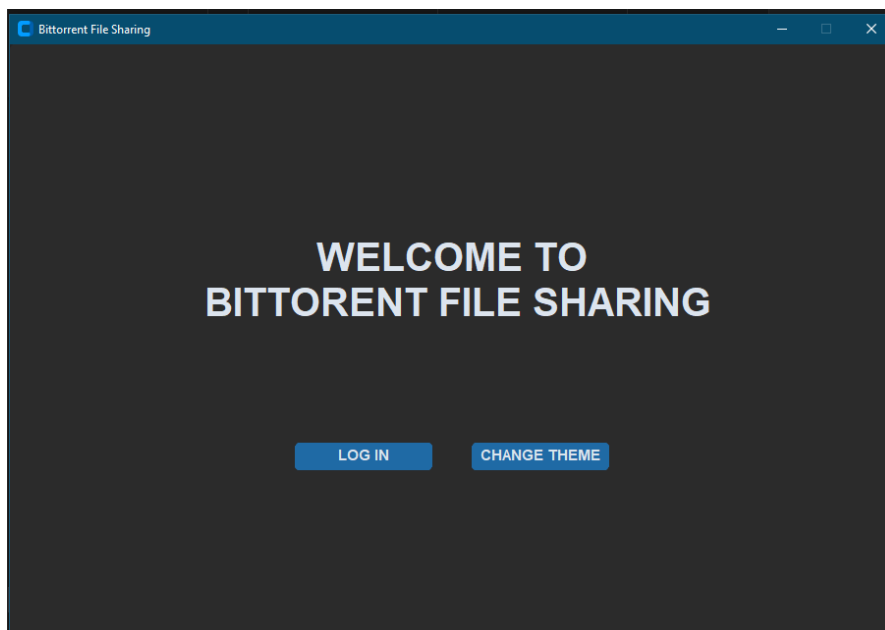


Hình 8: Activity diagram cho Server

7 GIAO DIỆN CỦA SẢN PHẨM

7.1 Peer

7.1.1 Giao diện bắt đầu ứng dụng



Hình 9: Giao diện bắt đầu ứng dụng.



7.1.2 Giao diện đăng nhập

Bittorrent File Sharing

HOME PAGE

LOG IN

Username

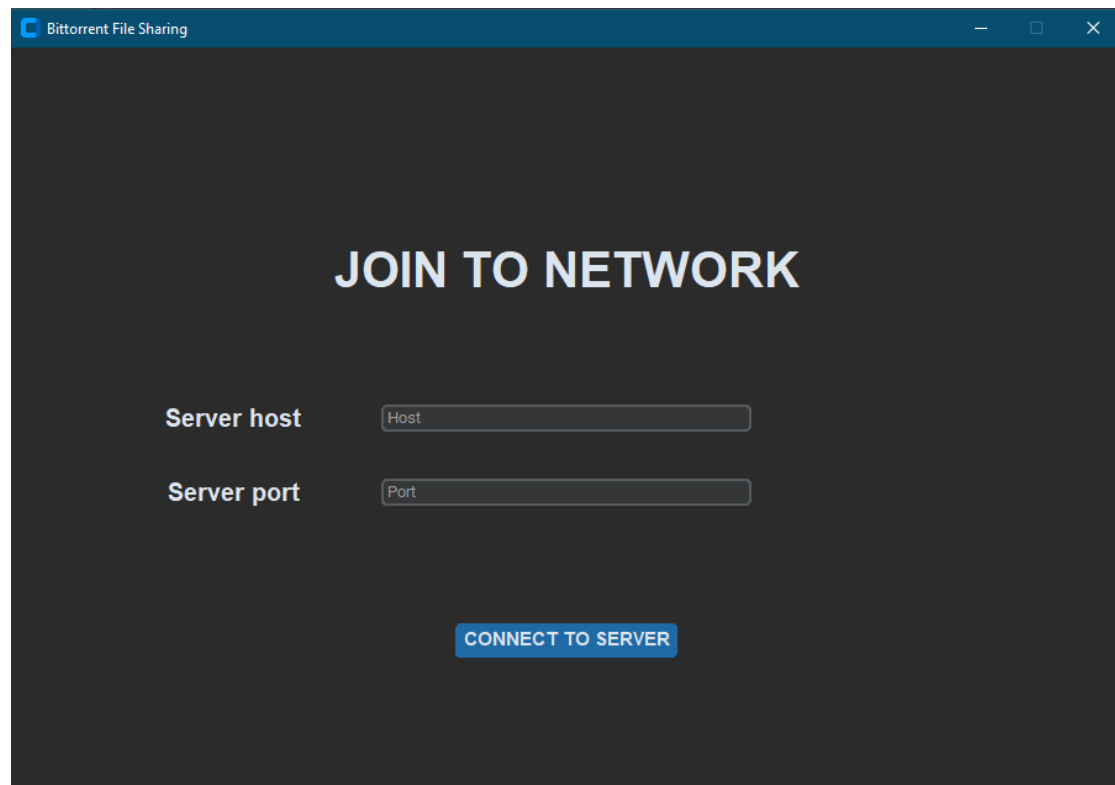
Password

CONFIRM

Hình 10: Giao diện đăng nhập.



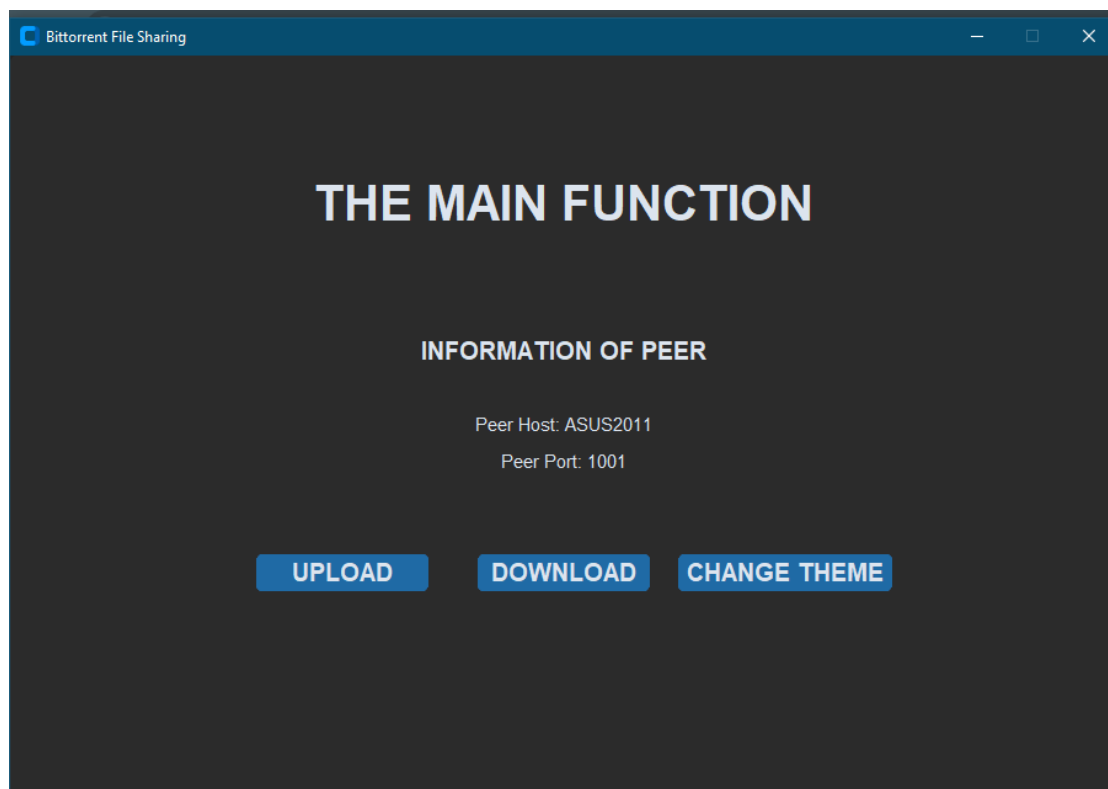
7.1.3 Giao diện kết nối với Server



Hình 11: Giao diện kết nối với Server.



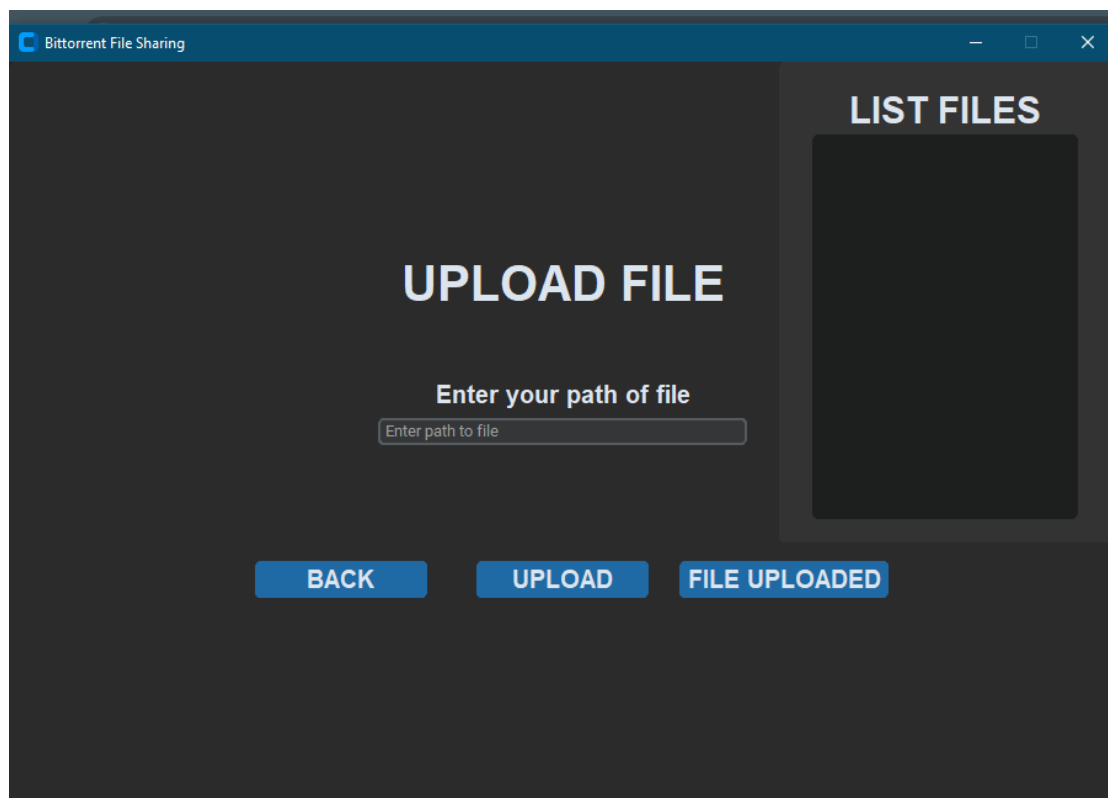
7.1.4 Giao diện chính của ứng dụng



Hình 12: Giao diện chính của ứng dụng.



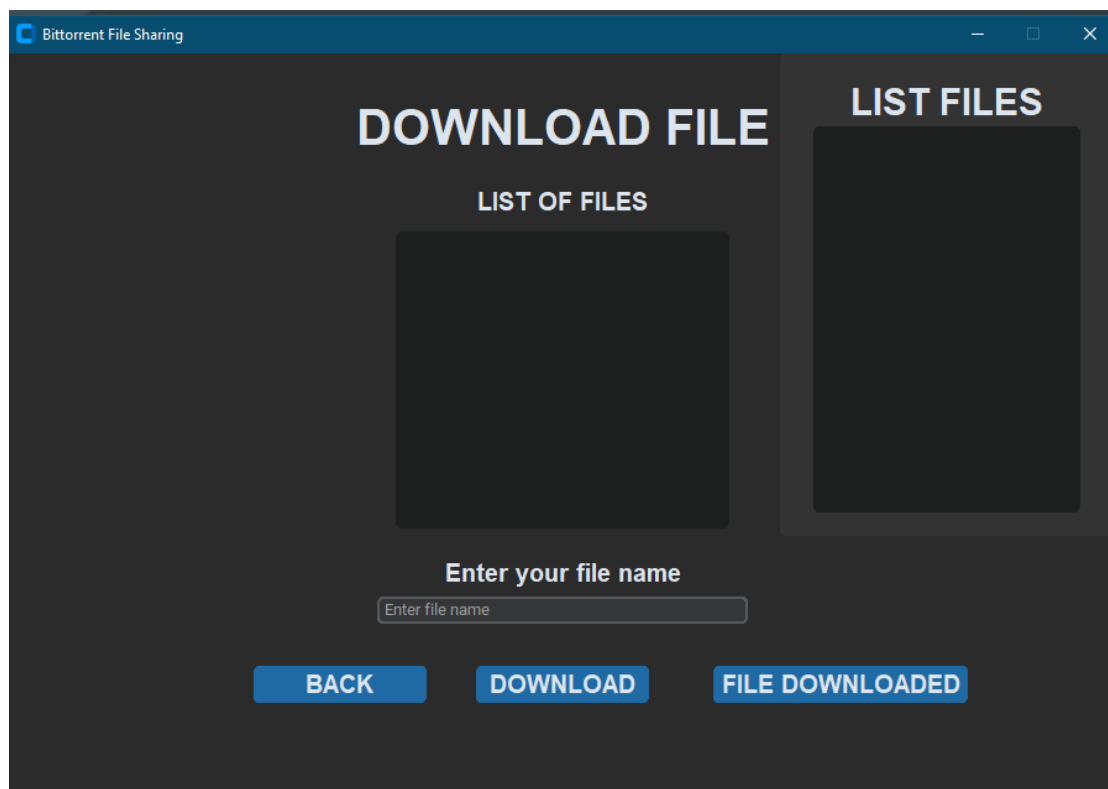
7.1.5 Giao diện upload tập tin



Hình 13: Giao diện upload tập tin.



7.1.6 Giao diện download tập tin

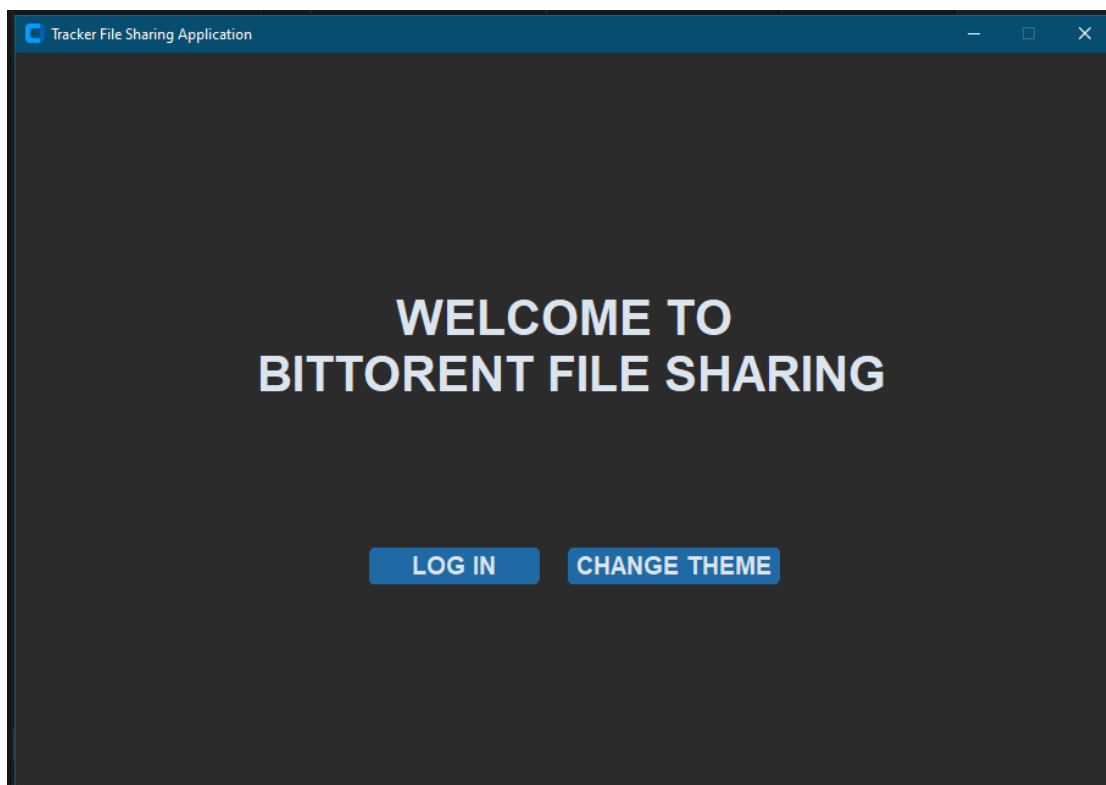


Hình 14: Giao diện download tập tin.



7.2 Server

7.2.1 Giao diện bắt đầu ứng dụng



Hình 15: Giao diện bắt đầu ứng dụng.



7.2.2 Giao diện đăng nhập

Tracker File Sharing Application

HOME PAGE

LOG IN

Username

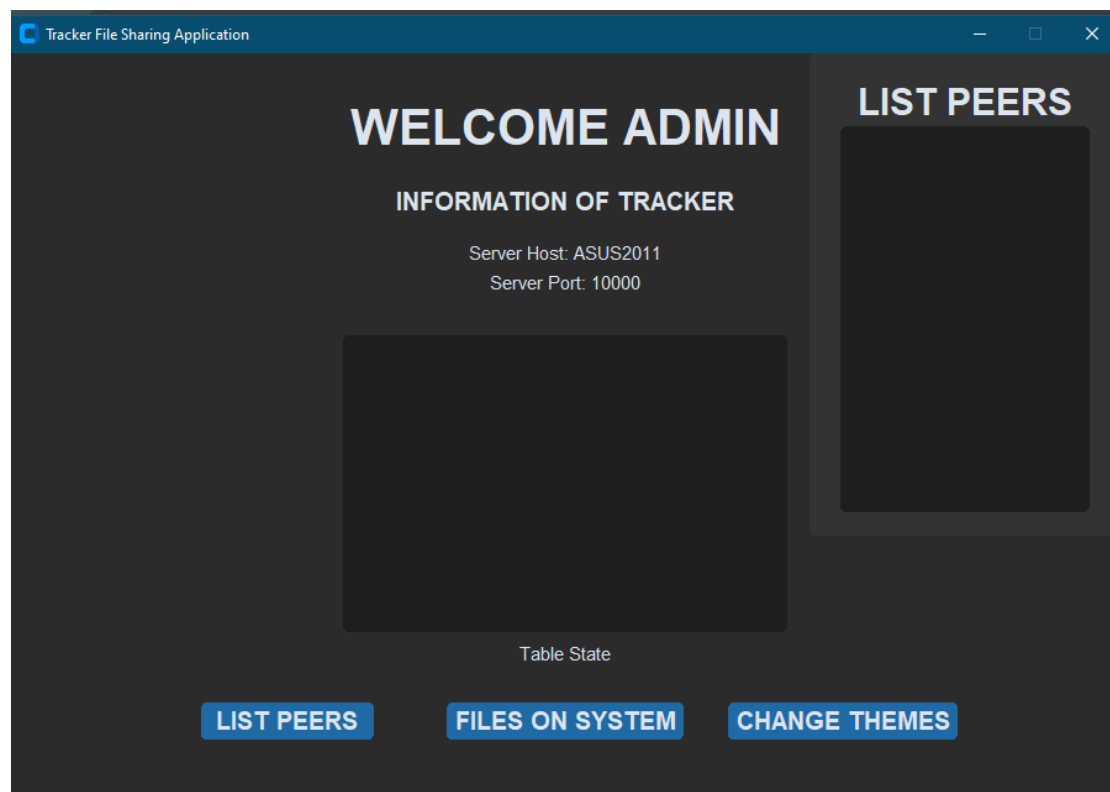
Password

CONFIRM

Hình 16: Giao diện đăng nhập.



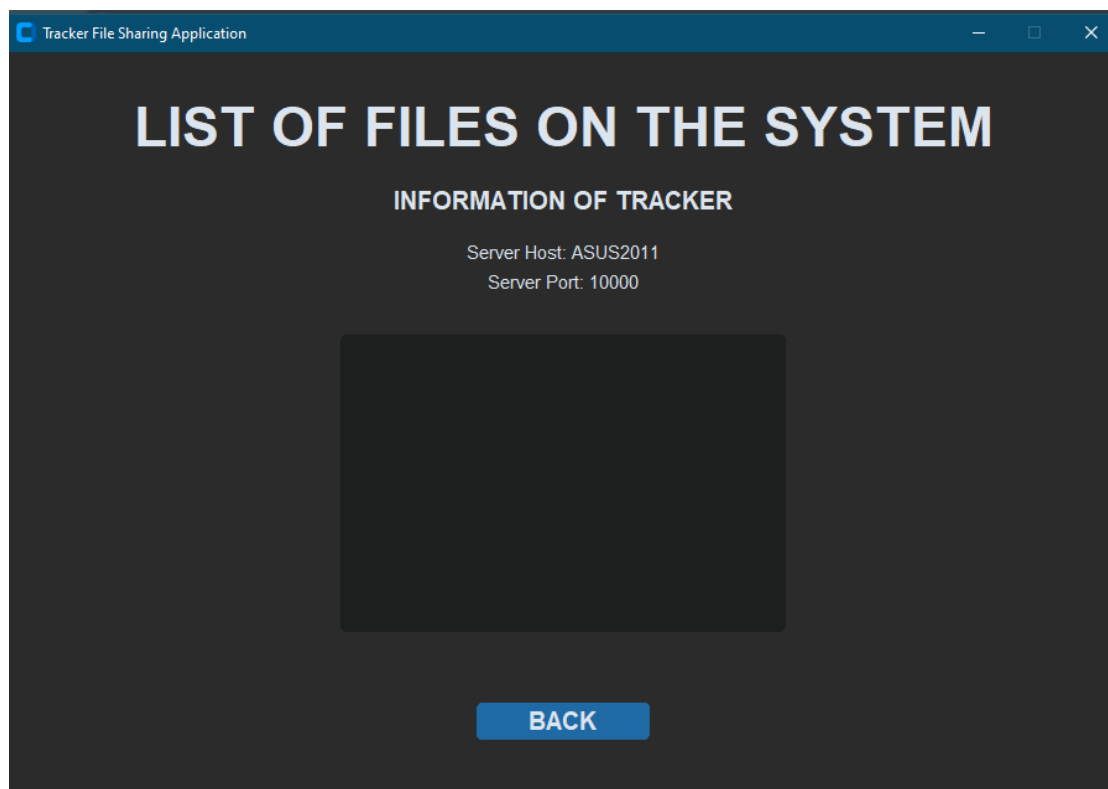
7.2.3 Giao diện chính của ứng dụng



Hình 17: Giao diện chính của ứng dụng.



7.2.4 Giao diện hiển thị các tập tin



Hình 18: Giao diện hiển thị các tập tin.



TÀI LIỆU THAM KHẢO

1. Protocols in Application Layer: <https://www.geeksforgeeks.org/protocols-application-layer>.
2. Introduction Socket.IO: <https://socket.io/docs/v4/>
3. P2P (Peer To Peer) File Sharing: <https://www.geeksforgeeks.org/p2p-peer-to-peer-file-sharing/>
4. BitTorrentSpecification: <https://wiki.theory.org/BitTorrentSpecification>