

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



MẠNG MÁY TÍNH (CO3049)

BÀI TẬP LỚN 1 (Phase 1)

DEVELOP A NETWORK APPLICATION (FILE-SHARING APPLICATION)

GVHD: Lê Bảo Khánh
SV: Đào Duy Long - 2113928
Trương Thuận Hưng - 2113619
Nguyễn Phạm Thiên Phúc - 2114445
Trần Minh Hiếu - 2113363

TP. HỒ CHÍ MINH, THÁNG 10/2023

Mục lục

1	Giới thiệu	2
1.1	Bối cảnh	2
1.2	Mục tiêu	2
2	Mô tả hệ thống	3
2.1	Phân tích yêu cầu	3
2.1.1	Yêu cầu chức năng	3
2.1.2	Yêu cầu phi chức năng	3
	2.1.2.a Cho toàn hệ thống:	3
	2.1.2.b Cho chức năng:	3
2.2	Giao thức	4
2.2.1	Giao thức Client-Server	4
2.2.2	Giao thức Peer-to-Peer	5
3	Diagram	5
3.1	Use Case Diagram	5

1 Giới thiệu

1.1 Bối cảnh

Ứng dụng chia sẻ file P2P là một mạng máy tính trong đó các máy tính kết nối trực tiếp với nhau để truyền dữ liệu. Điều này trái ngược với mô hình truyền thống, trong đó dữ liệu được truyền qua một máy chủ trung tâm. Với sự phân bổ tài nguyên hợp lý tránh sự quá tải của Server, mô hình P2P chia sẻ file ra đời giúp tránh phụ thuộc quá nhiều vào Server và tài nguyên của Server dẫn đến tắt nghẽn do lượng truy cập quá nhiều một thời điểm.

- **Chia sẻ file giữa các cá nhân:** Ứng dụng chia sẻ file P2P có thể được sử dụng để chia sẻ file giữa các cá nhân, chẳng hạn như bạn bè, gia đình hoặc đồng nghiệp. Điều này có thể được thực hiện qua mạng cục bộ hoặc qua Internet.
- **Chia sẻ file trong doanh nghiệp:** Ứng dụng chia sẻ file P2P có thể được sử dụng để chia sẻ file trong doanh nghiệp. Điều này có thể được sử dụng để chia sẻ tài liệu, mã nguồn hoặc các loại file khác giữa các nhân viên.
- **Chia sẻ file trong các tổ chức phi lợi nhuận:** Ứng dụng chia sẻ file P2P có thể được sử dụng để chia sẻ file trong các tổ chức phi lợi nhuận. Điều này có thể được sử dụng để chia sẻ tài liệu, hình ảnh hoặc video giữa các thành viên của tổ chức.

1.2 Mục tiêu

Mục tiêu chính của dự án là phát triển một ứng dụng chia sẻ tệp tin (File-sharing application) đơn giản, sử dụng giao thức TCP/IP. Hệ thống bao gồm một máy chủ trung tâm (Server) và các máy khách (Client) kết nối với máy chủ. Các máy khách sẽ thông báo cho máy chủ về các tệp tin có trong local repository của mình và có thể yêu cầu truy xuất các tệp tin từ các máy khách khác.

Đối với Server:

- Server nắm giữ thông tin về các Client tham gia vào hệ thống.
- Server quản lý danh sách các file mà các Client chia sẻ.
- Khi nhận được yêu cầu tải file từ Client, Server phải có khả năng xác định file đó đang thuộc Client nào và phản hồi lại.

Đối với Client:

- Client có khả năng gửi yêu cầu tải file lên Server và nhận được phản hồi từ Server.
- Client có thể tương tác với Client khác theo cơ chế Peer-to-Peer để có thể gửi và nhận file giữa các Client.
- Client có thể upload file từ máy vào repository của mình và thông báo cho Server khi hoàn tất.
- Tương tự, Client cũng có thể xóa file khỏi repository và thông báo cho Server khi hoàn tất.
- Multithread: Cho phép nhiều Client cùng tải một file của một Client đích ở cùng một thời điểm.

2 Mô tả hệ thống

2.1 Phân tích yêu cầu

2.1.1 Yêu cầu chức năng

- **Đăng ký và tạo tài khoản:** Ứng dụng phải cung cấp chức năng cho người dùng đăng ký và tạo tài khoản để tham gia hệ thống.
- **Đăng nhập và đăng xuất:** Ứng dụng phải hỗ trợ chức năng đăng nhập và đăng xuất cho người dùng.
- **Xem danh sách hostname các client:** Máy chủ (Server) phải có khả năng xem danh sách hostname của tất cả các client đã tham gia vào ứng dụng.
- **Kiểm tra trạng thái của một client ("Ping hostname"):** Máy chủ (Server) sẽ được cung cấp chức năng kiểm tra trạng thái của một client.
- **Xem thông tin các file đang chia sẻ của một client ("Discover hostname"):** Máy chủ (Server) được cung cấp khả năng xem thông tin về các file đang được chia sẻ bởi các client.
- **Upload file vào repository ("publish lname fname"):** Client phải có khả năng upload file từ máy của họ vào repository của client trong ứng dụng và sau khi hoàn thành, thông báo sẽ được gửi đến máy chủ (Server).
- **Xem danh sách file và tìm kiếm file:** Client phải có chức năng xem danh sách các file đang được chia sẻ trong ứng dụng.
- **Tải file từ client khác ("fetch file"):** Client phải có khả năng tải file từ client khác về repository của họ. Để có thể tải file thì client sẽ gửi yêu cầu tải file đến máy chủ (Server), máy chủ sẽ gửi lại cho client thông tin của một client đang nắm giữ file cần tải và bắt đầu tải file từ client đó.
- **Xóa file khỏi repository:** Client có khả năng xóa file khỏi repository chia sẻ của họ và gửi thông báo đến máy chủ (Server).

2.1.2 Yêu cầu phi chức năng

2.1.2.a Cho toàn hệ thống:

- Đảm bảo khả năng tải file thành công 90%: Hệ thống phải có khả năng đảm bảo rằng tối thiểu 90% các tải file được thực hiện thành công mà không gặp lỗi.
- Tỷ lệ mất gói tin dưới 10%: Hệ thống phải đảm bảo rằng tỷ lệ mất gói tin trong quá trình truyền dữ liệu không vượt quá 10%.
- Đảm bảo việc khách hàng có thể sử dụng trong 5 phút tìm hiểu về ứng dụng: Người dùng mới phải có khả năng sử dụng cơ bản của ứng dụng một cách dễ dàng trong vòng 2 phút sau khi làm quen với ứng dụng.

2.1.2.b Cho chức năng:

- Với chức năng gửi thông báo đến Server: Thời gian server nhận được thông báo tính từ khi client publish file lên repository hoàn tất sẽ không quá 2s.
- Với chức năng upload file chia sẻ "**publish lname fname**": Tên fname của file trong repository của client phải là khác với các tên file đã được chia sẻ có trong hệ thống.
- Multithread cho việc từ hai client nguồn trở lên có thể tải file đồng thời tại một client đích: Hệ thống cần hỗ trợ chạy nhiều luồng (**multithread**) để cho phép nhiều client cùng tải file đồng thời đến một client đích.

2.2 Giao thức

2.2.1 Giao thức Client-Server

Xác định giao thức dùng cho các chức năng liên quan đến mô hình Client-Server:

- Chức năng Đăng kí và tạo tài khoản:

- Giao thức sử dụng là: **HTTP**.
- Lí do sử dụng: HTTP được sử dụng cho chức năng đăng ký do tính phổ biến và tiện lợi. Giao thức này cho phép gửi thông tin đăng ký từ máy khách đến máy chủ một cách dễ dàng thông qua yêu cầu HTTP POST. Hơn nữa, HTTP hỗ trợ tích hợp bảo mật thông qua HTTPS, giúp đảm bảo tính bảo mật của thông tin đăng ký.

- Chức năng Đăng nhập và đăng nhập:

- Giao thức sử dụng là: **HTTP**.
- Lí do sử dụng: HTTP cũng được sử dụng cho chức năng đăng nhập và đăng xuất vì tính phổ biến và đơn giản của giao thức. Người dùng có thể gửi thông tin đăng nhập (tên đăng nhập hoặc email và mật khẩu) thông qua yêu cầu HTTP POST và đăng xuất thông qua yêu cầu HTTP GET hoặc POST

- Chức năng Kiểm tra trạng thái của một client ("Ping hostname"):

- Giao thức sử dụng là: **ICMP**.
- Lí do sử dụng: ICMP (Internet Control Message Protocol) là giao thức chuyên dụng để kiểm tra trạng thái của mạng và các thiết bị mạng. Nó thường được sử dụng để kiểm tra kết nối và tính khả dụng của một máy client thông qua gửi gói tin kiểm tra (ping) và nhận phản hồi.

- Chức năng Xem thông tin các file đang chia sẻ của một client ("Discover hostname"):

- Giao thức sử dụng là: **HTTP**.
- Lí do sử dụng: Giao thức HTTP được sử dụng cho chức năng này vì tính phổ biến và tích hợp dễ dàng. Máy chủ có thể gửi yêu cầu HTTP để lấy thông tin về các file đang chia sẻ từ máy client, và máy client có thể phản hồi thông tin thông qua phản hồi HTTP.

- Chức năng Upload file vào repository ("publish lname fname"):

- Giao thức sử dụng là: **HTTP**.
- Lí do sử dụng: Giao thức HTTP là để cho phép máy khách gửi thông tin (tên và kích thước) của file sau khi đã tải lên thành công vào repository. Khi máy khách hoàn tất việc tải lên, nó sử dụng yêu cầu HTTP POST để truyền thông tin về file (tên, kích thước) lên máy chủ, và sau đó máy chủ lưu trữ thông tin này trong database.

- Chức năng Xóa file khỏi repository:

- Giao thức sử dụng là: **HTTP**.
- Lí do sử dụng: Giao thức HTTP là để cho phép máy khách gửi thông báo đến máy chủ sau khi thực hiện thành công việc xóa file từ repository. Khi máy khách hoàn tất việc xóa file, nó sử dụng yêu cầu HTTP POST để gửi thông tin về việc xóa file lên máy chủ. Máy chủ sau đó có thể cập nhật thông tin trong database.

2.2.2 Giao thức Peer-to-Peer

Xác định giao thức dùng cho chức năng liên quan đến mô hình Peer-to-Peer:

- Tải file từ client khác ("fetch file"):

- Giao thức sử dụng là: **WebRTC**.
- Lí do sử dụng: WebRTC (Web Real-Time Communication) được sử dụng cho chức năng tải file từ máy client khác vì tính năng truyền tải dữ liệu trực tiếp giữa các máy client. WebRTC cho phép tạo các kết nối trực tiếp giữa các máy client, thực hiện việc truyền tải dữ liệu dưới dạng các luồng dữ liệu (data streams). Điều này giúp tối ưu hóa trải nghiệm tải file nhanh chóng và hiệu quả.

3 Diagram

3.1 Use Case Diagram

