

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING



COMPUTER NETWORKING

Assignment

DEVELOP A NETWORK APPLICATION

Advisor: La Quốc Nhật Huân
Students: Võ Nguyễn Đức Phát - 2212540
Nguyễn Đăng Cường -
Trần Phước Nhật -
Trần Uy -

HO CHI MINH CITY, NOVEMBER 2024



Mục lục

1	Cơ sở lý thuyết	3
1.1	Mô hình Peer To Peer là gì?	3
2	Phase 1	4
2.1	Đăng ký tệp, tải tệp với tracker	4
2.2	Yêu cầu file, tải file xuống từ tracker	5
2.3	Đăng nhập vào hệ thống để sử dụng	5
3	Phase 2	6
3.1	Các chức năng, hàm trong Bài tập lớn	6
3.2	Diagram	6
3.3	Quản lý trên Database	6
4	Demo Code	7
4.1	Các chức năng, hàm trong Bài tập lớn	7
4.2	Diagram	7
4.3	Quản lý trên Database	7



Member list & Workload

No.	Fullname	Student ID	Problems	Percentage of work
1	Nguyễn Văn A	19181716	- Relation & Counting: 1, 2, 3 Bonus: 1, 2, 3. - Probability: 1, 2, 3.	30%
2	Nguyễn Văn B	19181717	- Relation & Counting: 4, 5, 6 Bonus: 4, 5, 6. - Graph: 1, 2, 3, Bonus: 1, 2, 3.	20%

1 Cơ sở lý thuyết

1.1 Mô hình Peer To Peer là gì?

Mô hình **Peer-to-Peer (P2P)** là một cấu trúc mạng theo dạng phân tán, trong đó các thiết bị hoặc *peer* đóng vai trò vừa là máy khách (client) vừa là máy chủ (server) trong quá trình trao đổi tài nguyên với nhau. Khác với kiểu mô hình truyền thống Client-Server, nơi mà chỉ có một máy chủ tập trung phục vụ các máy khách, mô hình P2P không yêu cầu một máy chủ trung tâm để quản lý dữ liệu. Thay vào đó, mỗi peer có khả năng cung cấp và nhận dữ liệu trực tiếp với các peer khác trong mạng. Điều này giúp cải thiện hiệu quả sử dụng băng thông, tăng khả năng mở rộng một cách dễ dàng cũng như khả năng chịu lỗi của hệ thống.

Trong bối cảnh của Bài tập lớn này, cụ thể là mạng chia sẻ dữ liệu *BitTorrent*, mô hình P2P cho phép các peer chia sẻ các phần của tệp tin với nhau, ngay khi chúng vừa tải về thành công. Cơ chế này giúp tối ưu hóa tốc độ tải tệp, vì một peer không cần phải tải toàn bộ tệp từ một nguồn duy nhất. Nó có thể nhận dữ liệu từ nhiều peer khác nhau đồng thời, giao tiếp với các peer đó. Ngoài ra, một yếu tố quan trọng trong BitTorrent là sự có mặt của **tracker**, giúp quản lý và cung cấp thông tin về các peer đang chia sẻ tệp. Tracker không lưu trữ tệp tin mà chỉ đóng vai trò cung cấp thông tin để các peer tìm thấy nhau dễ dàng hơn trong mạng.

Kết hợp giữa tính năng phân phối và cấu trúc phân tán của mô hình P2P, BitTorrent đã trở thành một trong những phương thức chia sẻ tệp phổ biến nhất trên thế giới, nổi bật với khả năng giảm tải cho các máy chủ trung tâm và tăng tốc độ truyền tải cho người dùng. Bằng lẽ đó, trong bài tập lớn này sẽ mô tả lại cách hiện thực 1 ứng dụng áp dụng mạng chia sẻ dữ liệu BitTorrent cơ bản.

2 Phase 1

Xác định và mô tả các chức năng của ứng dụng và các giao thức

Trong mô hình BitTorrent, các chức năng chính của ứng dụng và giao thức được thực hiện qua sự phối hợp giữa các peer và tracker để chia sẻ tệp tin hiệu quả. Các chức năng cơ bản bao gồm: *đăng ký tệp* với tracker, *yêu cầu tệp* từ tracker, và *cung cấp thông tin các peer đang nắm giữ tệp*.

2.1 Đăng ký tệp, tải tệp với tracker

Khi một peer muốn chia sẻ tệp tin, bước đầu tiên là **register - đăng ký tệp** với tracker. Quá trình này diễn ra khi peer gửi một thông báo đến tracker để báo rằng nó đang nắm giữ một bản sao của tệp tin cần chia sẻ. Tracker sẽ lưu trữ thông tin về peer này, bao gồm địa chỉ của peer và thông tin về tệp tin mà nó nắm giữ. Điều này cho phép các peer khác dễ dàng tìm thấy tệp khi cần. Quá trình đó diễn ra cụ thể như sau:

- Chọn tệp cần chia sẻ: Người dùng sử dụng giao diện của ứng dụng để chọn các tệp hoặc thư mục mà họ muốn chia sẻ với người dùng khác trên mạng. Đây có thể là các tệp đã có sẵn trên máy tính của họ. Lưu ý là phải ghi rõ về định dạng của tệp cần chia sẻ (Ví dụ: muốn chia sẻ file pdf, gọi rõ ra là "Data.pdf")
- Xác định phần trích dẫn (hash): Trước khi tải tệp lên, ứng dụng sẽ tạo ra một phần trích dẫn hoặc hash cho các phần của tệp. Hash này được sử dụng để xác minh tính toàn vẹn và đánh dấu riêng biệt của tệp trong quá trình tải xuống.
- Tệp được chia thành các phần nhỏ có kích thước cố định, gọi là các chunk. Mỗi chunk được gắn với một mã hash để xác thực và nhận diện duy nhất phần dữ liệu đó.
- Tạo tệp torrent: Ứng dụng sẽ tạo một tệp metainfo (.torrent file) chứa thông tin về tệp, các phần trích dẫn (hash) của các chunk, và các thông tin khác như tên tệp, kích thước để khi 1 peer muốn download thì tracker sẽ dựa trên .torrent file này và đưa thông tin cho peer cần download.
- Chia sẻ tệp torrent: Sau khi tạo tệp torrent, người dùng có thể chia sẻ tệp này với người dùng khác thông qua email, trang web, hoặc các kênh khác trên mạng. Tệp torrent này chứa thông tin cần thiết để người dùng khác có thể tìm kiếm và tải tệp từ nguồn chia sẻ.
- Kết nối và chia sẻ thông tin: Khi người dùng khác tìm thấy tệp torrent và muốn tải xuống, họ sẽ kết nối với máy tính của người chia sẻ thông qua mạng BitTorrent. Từ đó, các máy tính có giữ tệp đó sẽ gửi các phần nhỏ của tệp (chunk) cho người tải xuống từ máy tính của họ.

Đối với việc giao tiếp như này, giao thức truyền thông được sử dụng là: **HTTP** - gửi yêu cầu đăng ký đến tracker, thông báo rằng nó có một tệp tin cụ thể và sẵn sàng chia sẻ. Yêu cầu này bao gồm cả IP của các peer, mã port. Ngoài ra, còn sử dụng giao thức BitTorrent để kết nối trực tiếp giữa các peer.

2.2 Yêu cầu file, tải file xuống từ tracker

Khi một peer khác cần tải tệp, nó sẽ thực hiện chức năng **yêu cầu tệp** từ tracker. Peer này gửi yêu cầu đến tracker, bao gồm thông tin về tên, định dạng của tệp tin cần tải. Tracker sẽ kiểm tra database và tìm kiếm các peer khác đang có bản sao (cụ thể là các chunk) của file này. Đây là bước rất cần thiết để peer yêu cầu biết được các peer nào có thể cung cấp file. Cụ thể như sau:

- **Tìm kiếm tệp:** Người dùng sử dụng CLI của ứng dụng để nhập các thông tin tìm kiếm như tên tệp, định dạng để báo cho tracker biết. Sau đó, tracker sẽ tìm kiếm file .torrent trong database để có được thông tin về các peer và đang nắm giữ file này.
- **Phản hồi lại yêu cầu tìm kiếm:** Tracker nhận yêu cầu tìm kiếm và phản hồi bằng danh sách các peer đang nắm giữ tệp tin cần tìm. Danh sách này bao gồm địa chỉ IP, cổng kết nối, và các thông tin về các chunk cụ thể mà mỗi peer đang nắm giữ của tệp đó, giúp peer yêu cầu có thể kết nối và tải dữ liệu một cách hiệu quả.
- **Kết nối và tải xuống:** Khi đã có được thông tin chi tiết về file cần tải, ứng dụng sẽ có thuật toán phù hợp để tự động tải về theo 1 cách tối ưu và hiệu quả nhất cho người dùng.
- **Hoàn thành tải xuống:** Khi tất cả các phần của tệp đã được tải xuống, ứng dụng tự động kiểm tra và xác minh tính toàn vẹn của tệp, sau đó thông báo cho người dùng rằng quá trình tải xuống đã hoàn tất.

Giao thức truyền thông được sử dụng ở chức năng này là **HTTP** - Sử dụng để tìm kiếm và tải xuống tệp thông qua IP, port. Tương tự chức năng trên, cũng sử dụng mạng BitTorrent để các peer có thể giao tiếp trực tiếp với nhau.

2.3 Đăng nhập vào hệ thống để sử dụng

Chức năng này cho phép người dùng đăng nhập vào hệ thống thông qua giao diện dòng lệnh (CLI) trên terminal. Sau khi đăng nhập thành công, người dùng sẽ có quyền truy cập vào cơ sở dữ liệu để thực hiện các thao tác liên quan đến tracker trong mô hình BitTorrent.

- Khi chạy chương trình Python qua terminal, người dùng sẽ được yêu cầu nhập tên người dùng và mật khẩu.
- Hệ thống kiểm tra thông tin đăng nhập từ cơ sở dữ liệu để xác thực. Nếu thông tin hợp lệ, quá trình đăng nhập thành công và người dùng có quyền truy cập vào hệ thống.
- Sau khi đăng nhập, người dùng có thể gọi tracker để register file hoặc request file của các peer nắm giữ tệp cần thiết.

Trong chức năng đăng nhập và kết nối với cơ sở dữ liệu để gọi tracker qua CLI, các giao thức chính bao gồm:

- PostgreSQL: Sử dụng giao thức TCP để kết nối qua cổng mặc định 5432.



3 Phase 2

Hiện thực trong Bài tập lớn

3.1 Các chức năng, hàm trong Bài tập lớn

3.2 Diagram

3.3 Quản lý trên Database



4 Demo Code

Hiện thực trong Bài tập lớn

4.1 Các chức năng, hàm trong Bài tập lớn

4.2 Diagram

4.3 Quản lý trên Database



Tài liệu

[1] ...

[2] ...