

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



BÁO CÁO

Bài tập lớn

DEVELOP A NETWORK APPLICATION

GV hướng dẫn: Lê Bảo Khánh
Members: Nguyễn Minh Hiếu - 2153343
Trần Tường Khang - 2252313
Diệp Vũ Minh - 2252472
Nguyễn Quốc Lộc - 2252458
Vũ Đức Toàn - 2252822

Hồ Chí Minh City, 10/2024



Contents

1	Tổng quan dự án	2
1.1	Thông tin dự án	2
1.2	Phạm vi	2
1.3	Phân bổ nhiệm vụ	2
2	Bối cảnh lý thuyết và giải pháp đề xuất	2
2.1	Centralized Peer-to-Peer (P2P) Model	2
2.2	Phát triển ứng dụng	2
2.3	Tóm tắt công nghệ	2
3	Giới thiệu	3
3.1	Mô tả dự án	3
3.2	Mục tiêu	3
4	Chức năng của hệ thống	3
4.1	Functional requirements	3
4.2	Non-functional requirements	4
4.3	Data Management Requirements	5
4.3.1	Giao thức giữa Tracer và Node	5
4.3.2	Giao thức giữa các Node (Peer-to-Peer TCP/IP)	5
4.3.3	Phương pháp truyền dữ liệu đa chiều (MDDT)(TCP/IP)	5
5	Thiết kế hệ thống	5
6	Use case diagram	6
7	Kiểm thử	6
8	Hướng dẫn sử dụng	6



1 Tổng quan dự án

1.1 Thông tin dự án

Tiêu đề: Hệ thống Chia Sẻ Tập Tin Trung Tâm

Thời gian thực hiện: Bắt đầu 15/10 - Kết thúc 31/10

Kích thước nhóm: 5

Mục tiêu: Phát triển ứng dụng chia sẻ tài liệu peer-to-peer (P2P) sử dụng mô hình mạng tập trung và giao thức TCP/IP.

1.2 Phạm vi

Product: [Product Scope Description]

Network Model: [Description of the network model used]

Document Storage: [Description of document storage setup]

1.3 Phân bổ nhiệm vụ

Tên	MSSV	Nhiệm vụ được giao
Trần Tường Khang	2252313	Task 4,5,6
Diệp Vũ Minh	2252472	Task 3,4,5
Nguyễn Quốc Lộc	2252458	Task 1,2,3

2 Bối cảnh lý thuyết và giải pháp đề xuất

2.1 Centralized Peer-to-Peer (P2P) Model

Overview: [Brief explanation of the centralized P2P model]

Data Transfer Protocol: [Description of protocols used, e.g., HTTP and TCP]

2.2 Phát triển ứng dụng

Technologies: List and describe all technologies and tools used, including front-end and back-end frameworks, database systems, etc.

2.3 Tóm tắt công nghệ

Objective	Solution
Connection Model	[e.g., Centralized P2P]
Data Transfer Protocol	[e.g., HTTP, TCP]
Version Control & Source Management	[e.g., Git, GitHub]
Front-end	[e.g., ReactJS]
Back-end	[e.g., Fastify, Prisma]

3 Giới thiệu

3.1 Mô tả dự án

Hệ thống chia sẻ tệp kiểu torrent đơn giản mà nhóm chúng mình xây dựng có hai thành phần chính: Tracker và Node. Tracker đóng vai trò như một máy chủ trung tâm để quản lý danh sách các node và theo dõi thông tin về các file mà mỗi node sở hữu. Node là các máy khách tham gia vào mạng, có thể tải lên hoặc tải xuống tệp từ các node khác trong hệ thống.

Hệ thống hỗ trợ tính năng truyền dữ liệu đa chiều (MDDT), cho phép một node có thể tải xuống nhiều tệp từ nhiều nguồn cùng lúc. Hệ thống sử dụng giao thức TCP/IP cho các kết nối mạng và thực hiện chia sẻ tệp theo kiểu P2P.

3.2 Mục tiêu

Mục tiêu của dự án là tạo ra một hệ thống cho phép nhiều máy tính (Nodes) kết nối với một máy chủ trung tâm (tracker) để có khả năng chia sẻ và tìm kiếm tệp tin dễ dàng. Dự án này nhấn mạnh vào việc quản lý thông tin về các tệp tin trong mạng và cung cấp cơ hội cho các máy tính trong mạng truy cập và chia sẻ tệp tin với nhau một cách hiệu quả.

4 Chức năng của hệ thống

4.1 Functional requirements

Server (Tracker)

- Quản lý thông tin file: Tracker lưu trữ thông tin về các file đã có tại các node, bao gồm những mảnh tệp (pieces) mà mỗi node sở hữu.
- Xử lý yêu cầu từ node: Tracker nhận và xử lý các yêu cầu từ node, bao gồm việc cung cấp thông tin về các peer khác có tệp mà node cần tải xuống.
- Trả về danh sách peer: Khi một node yêu cầu tải xuống tệp, tracker sẽ trả về danh sách các peer mà node có thể kết nối để tải tệp.
- Chia sẻ thông tin tải và seeding: Tracker theo dõi trạng thái tải của các node, như số byte đã tải và thông báo khi một node bắt đầu hoặc hoàn tất việc chia sẻ (seeding) tệp.

Client (Node)

- Gửi thông tin tới tracker: Khi khởi động, node gửi thông tin về các tệp mà nó sở hữu và các tệp cần tải đến tracker, bao gồm các mảnh tệp (pieces) đã có.
- Tải xuống tệp: Node gửi yêu cầu tới tracker để nhận danh sách các peer có file cần thiết và kết nối với các peer đó để tải xuống các mảnh tệp còn thiếu.
- Chia sẻ tệp (Seeding): Sau khi hoàn tất tải xuống, node trở thành "seeder" và bắt đầu chia sẻ tệp với các node khác. Nó sẽ thông báo với tracker rằng tệp đã sẵn sàng để chia sẻ.

4.2 Non-functional requirements

Yêu cầu hiệu suất:

- Thời gian phản hồi: Tracker phải phản hồi yêu cầu từ node trong vòng 5 giây cho các tác vụ phổ biến như lấy danh sách các peer hoặc cập nhật trạng thái chia sẻ.
- Tốc độ xử lý yêu cầu: Tracker phải xử lý được tối thiểu 500 yêu cầu từ các node đồng thời mà không làm giảm hiệu suất.
- Khả năng mở rộng: Hệ thống cần có khả năng xử lý tải cao và hỗ trợ ít nhất 1,000 node kết nối đồng thời.

Yêu cầu bảo mật:

- Xác thực: Hệ thống cần xác thực các node trước khi chấp nhận yêu cầu, đảm bảo chỉ các node hợp lệ mới được tham gia vào mạng chia sẻ.
- Bảo mật dữ liệu: Thông tin trao đổi giữa tracker và node phải được mã hóa để ngăn chặn rò rỉ hoặc tấn công man-in-the-middle.

Yêu cầu về khả năng sử dụng:

- Thao tác đơn giản: Giao diện và quy trình hoạt động của node cần dễ sử dụng, cho phép người dùng kết nối và bắt đầu chia sẻ hoặc tải tệp mà không cần nhiều bước phức tạp.
- Hướng dẫn rõ ràng: Hệ thống cần cung cấp hướng dẫn đơn giản để người dùng có thể nắm bắt nhanh chóng cách sử dụng hệ thống và quản lý các tệp của mình.

Yêu cầu tin cậy:

- Khả năng hoạt động liên tục: Hệ thống phải đảm bảo tính khả dụng 24/7, với tỷ lệ uptime tối thiểu 99% để các node luôn có thể kết nối với tracker.
- Khôi phục sau sự cố: Trong trường hợp xảy ra sự cố, tracker cần có khả năng khôi phục và hoạt động bình thường trở lại trong vòng 2 giờ.

Yêu cầu tổ chức:

- Yêu cầu phát triển: Hệ thống cần có kiến trúc mở rộng dễ dàng để tăng số lượng node và các file chia sẻ mà không làm ảnh hưởng đến hiệu suất. Tracker cần hỗ trợ việc thêm node mới vào mạng chia sẻ trong vòng tối đa 5 phút kể từ khi kết nối.
- Công nghệ sử dụng: Hệ thống sử dụng các công nghệ như (thêm công nghệ) để phát triển.

Yêu cầu môi trường: Tương thích trình duyệt: Node cần hoạt động tương thích với các trình duyệt phổ biến như Google Chrome, Firefox và Microsoft Edge để tối ưu trải nghiệm cho người dùng.

Yêu cầu bên ngoài:

- Yêu cầu pháp lý: Hệ thống phải tuân thủ các quy định pháp lý liên quan đến xử lý và lưu trữ dữ liệu cá nhân.
-
- Yêu cầu đạo đức: Hệ thống phải đảm bảo bảo mật thông tin cá nhân của người dùng, chỉ sử dụng thông tin này cho mục đích chia sẻ tệp mà người dùng đã cho phép.

4.3 Data Management Requirements

Data Storage: Outline storage location and management for both server and client data.

4.3.1 Giao thức giữa Tracer và Node

Mô tả: Giao thức này được sử dụng để node có thể liên lạc với tracker và nhận thông tin về các peer khác trong mạng.

Yêu cầu từ node tới tracker:

- Khi node khởi động, nó gửi một yêu cầu bao gồm magnet text (chứa hash code(mã băm) dẫn tới tệp metainfo) tới tracker.
- Node cũng gửi yêu cầu để thông báo về các tệp mà nó sở hữu và các file còn thiếu.
- Yêu cầu bao gồm các trạng thái như started, stopped, hoặc completed khi cần thiết, cùng với số byte đã tải về.

Phản hồi từ tracker:

- Tracker sẽ trả về danh sách các peer, mỗi peer bao gồm: ID, địa chỉ IP, và port.
- Ngoài ra, tracker cũng có thể trả về thông báo lỗi hoặc cảnh báo nếu có vấn đề xảy ra trong quá trình giao tiếp.

4.3.2 Giao thức giữa các Node (Peer-to-Peer TCP/IP)

Mô tả: Giao thức này được sử dụng để các node trong mạng có thể trao đổi các file với nhau.

Quá trình tải xuống: Sau khi nhận danh sách các peer từ tracker, node sẽ kết nối với các peer này và bắt đầu tải xuống các file mà nó chưa có. Node cần đảm bảo rằng không yêu cầu tải các file mà peer không có hoặc các file mà nó đã có.

Quá trình tải lên: Sau khi tải xong một tệp, node sẽ bắt đầu chia sẻ (seeding) tệp này với các node khác. Node cần quản lý luồng dữ liệu từ nhiều peer cùng lúc, đảm bảo tính hiệu quả trong việc tải lên và tránh bị quá tải.

4.3.3 Phương pháp truyền dữ liệu đa chiều (MDDT)(TCP/IP)

Mô tả: Node sử dụng phương pháp đa luồng để tải xuống các file từ nhiều nguồn khác nhau cùng lúc.

Quản lý yêu cầu:

Node sẽ giữ một hàng đợi các yêu cầu tải tệp từ các peer khác nhau và theo dõi trạng thái của từng file để tránh việc yêu cầu trùng lặp.

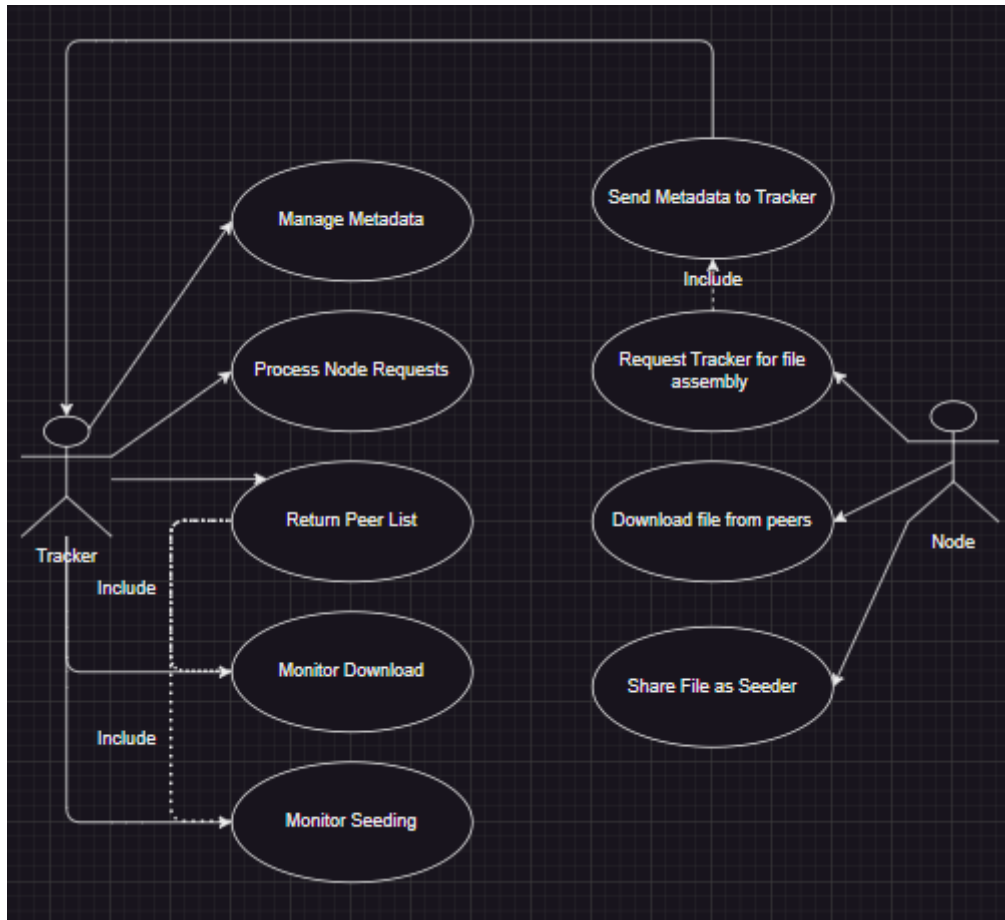
Khi có lỗi kết nối hoặc ngắt kết nối, node sẽ chuyển sang các peer khác có cùng file và tiếp tục tải xuống.

5 Thiết kế hệ thống

System Overview: [Description of the system architecture]

Server Details: [Details on server functionalities]

Client Details: [Description of client-side functionalities and data flow]



Hình 1: Use-case diagram

6 Use case diagram

7 Kiểm thử

System Setup Tests: Outline test cases for system setup and client connectivity.

Functionality Testing: Provide test cases for client publishing, unpublishing, and document retrieval.

8 Hướng dẫn sử dụng

Installation Guide: Provide steps for setting up the server and client.

System Usage: Step-by-step guide for both administrative and user actions on the system.



References

- [1] Computer Networking: A Top-Down Approach - Jim Kurose, Keith Ross, 8th Global Edition, Pear-son, 2021.
- [2] Computer Networks - Andrew S. Tanenbaum, Nick Feamster, David J. Wetherall, 6th Edi-tion, Pear-son, 2021.
- [3] The Illustrated Network: How TCP/IP Works in a Modern Network - Walter Goralski, Second Edi-tion, Morgan Kaufman, 2017.