

Báo cáo đề tài ứng dụng truyền nhận file qua mạng

Thành viên nhóm

Họ tên	Mã số sinh viên
Đoàn Khuê	1612311
Nguyễn Văn Linh	1612340
Mai Chí Cường	1612070
Phạm Trung Tín	1612695

- **Mục lục**
 - 1. Đánh giá chức năng đã hoàn thành
 - 2. Phân công việc
 - 3. Tổng quan hệ thống
 - 3.1. Mô hình hoạt động
 - 3.2. Thiết kế giao thức
 - 3.2.1. PING method
 - 3.2.2. SYNC method
 - 3.2.3. INFO pmethod
 - 3.2.4. GET method
 - 3.2.5. LIST method
 - 3.3. Chi tiết vận hành
 - 3.3.1 Master
 - 3.3.2 Fileserver
 - 3.3.3 Client
 - 4. Usage
 - 5. Demo
 - 6. Tham khảo

1. Đánh giá chức năng đã hoàn thành

1.1. Master server

Chức năng	Hoàn thành	Ghi chú
Có địa chỉ IP:port cố định. Chứa thông tin về các file được chia sẻ bởi file server, thông tin IP:port của file server quản lý tương ứng.	✓	
Cung cấp service (#1) để ghi nhận thông tin mà file server gửi lên gồm: danh sách các file, IP:port của file server.	✓	
Cung cấp service (#2) để client có thể lấy thông tin danh sách các file được chia sẻ, kèm theo IP:port của file server quản lý file.	✓	

Chức năng	Hoàn thành	Ghi chú
Cho phép nhiều client, nhiều file server kết nối tới cùng một thời điểm.	✓	

1.2. File server

Chức năng	Hoàn thành	Ghi chú
Chứa các files có thể chia sẻ được với client.	✓	
Khi file server khởi động, nó kết nối đến master server và gọi service (1) để gửi thông tin của chính nó lên file server gồm: danh sách file có thể chia sẻ, địa chỉ IP, port mà client có thể kết nối tới để tải file.	✓	
Cung cấp service (#3) để client tải file với input là tên file cần tải. Service này sử dụng giao thức UDP tại tầng Transport.	✓	
Cung cấp chức năng hiện danh sách file hiện có trên các file server	✓	Chức năng nhóm tự thêm vào
Cho phép nhiều client kết nối đến cùng một thời điểm.	✓	
Khi file server ngừng hoạt động, master server cần loại bỏ danh sách file của file server tương ứng.	✓	

1.3. Client

Chức năng	Hoàn thành	Ghi chú
Có thể sử dụng service (2) do master server cung cấp và service (3) do file server cung cấp.	✓	
Có thể tải nhiều file cùng một thời điểm.	✓	

1.4. Các yêu cầu khác

Chức năng	Hoàn thành	Ghi chú
Thiết kế giao thức tại tầng Application để đảm bảo file được truyền nhận theo giao thức UDP (service #3) có độ tin cậy.	✗	Nhóm không áp dụng UDP tin cậy được nên sử dụng TCP

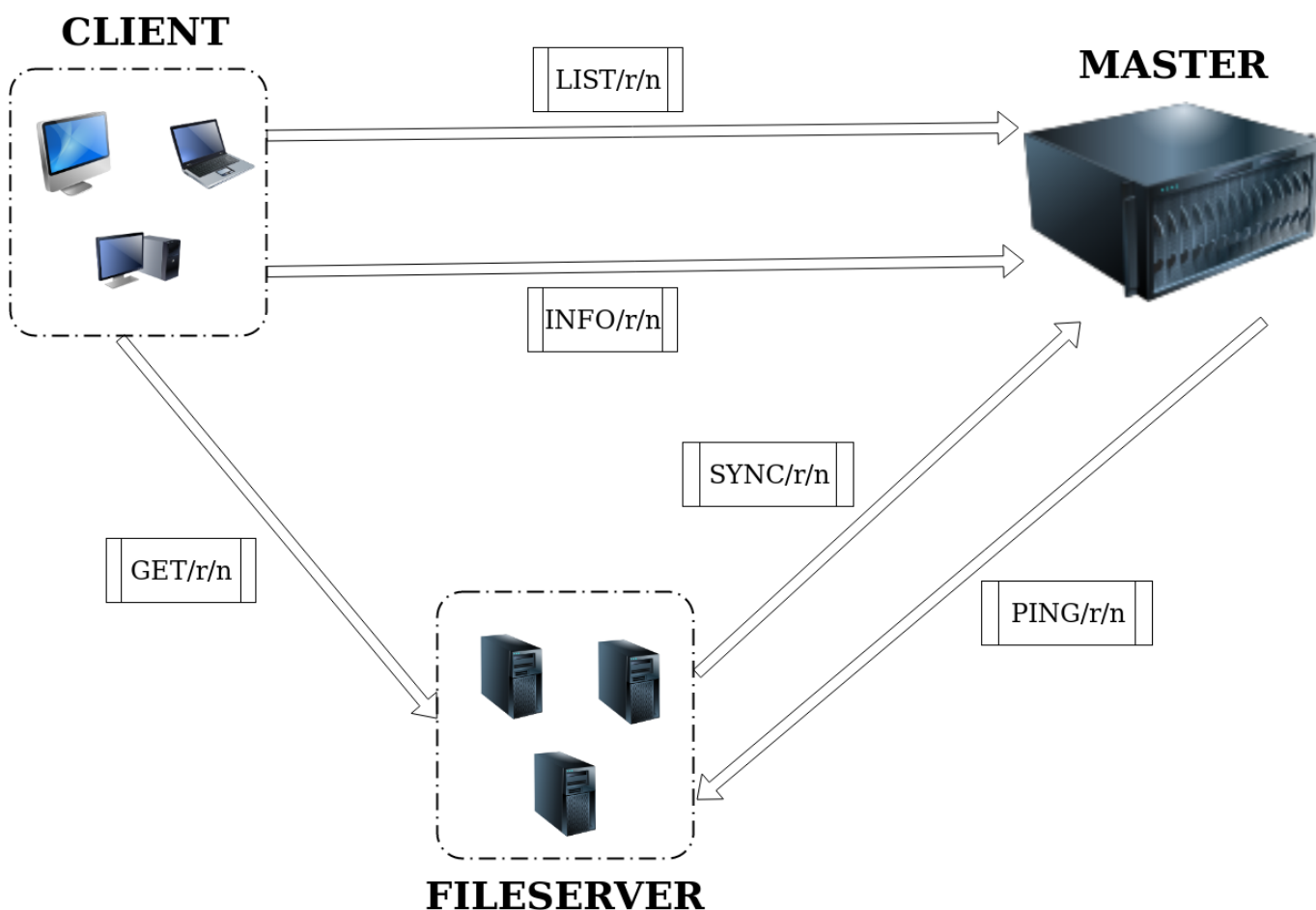
2. Phân công việc

Chức năng	Người thực hiện	Mã số sinh viên	Mức độ hoàn thành	Ghi chú
Master	Nguyễn Văn Linh	1612340	100%	
File Server	Phạm Trung Tín	1612695	90%	Không triển khai được UDP tin cậy
Client	Đoàn Khuê	1612311	90%	Không triển khai được UDP tin cậy
Test và ghi báo cáo	Mai Chí Cường	1612070	100%	

Cả nhóm tham gia thiết kế giao thức.

3. Tổng quan hệ thống

3.1. Mô hình hoạt động



- **Centralize**

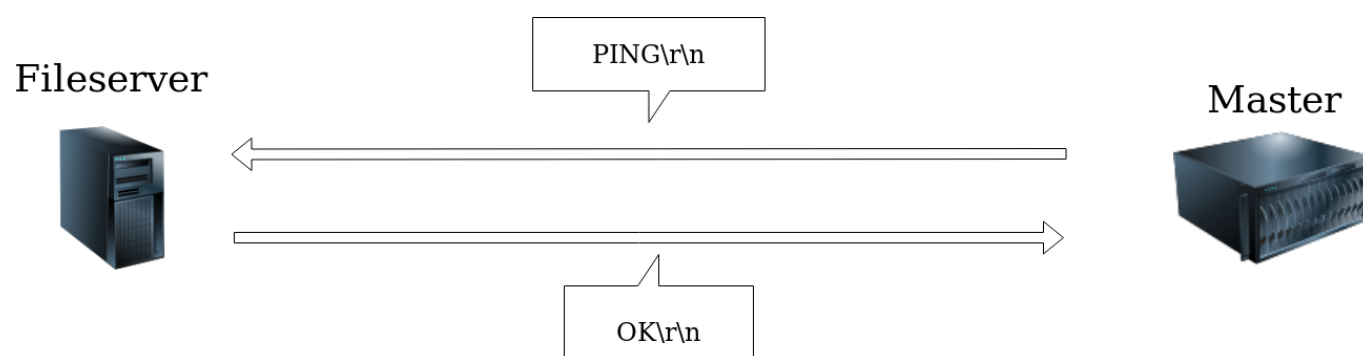
- Xử lý tập trung, toàn bộ thông tin về file được lưu trên master server
- Client connect đến master để lấy IP PORT fileserver sau đó kết nối đến để lấy data, các fileserver phân tán ở mọi nơi (scalable), tránh trường hợp masterserver bị quá tải
- Client có thể lấy danh sách toàn bộ các file đang được lưu thông tin trên master

- **Multithreading**

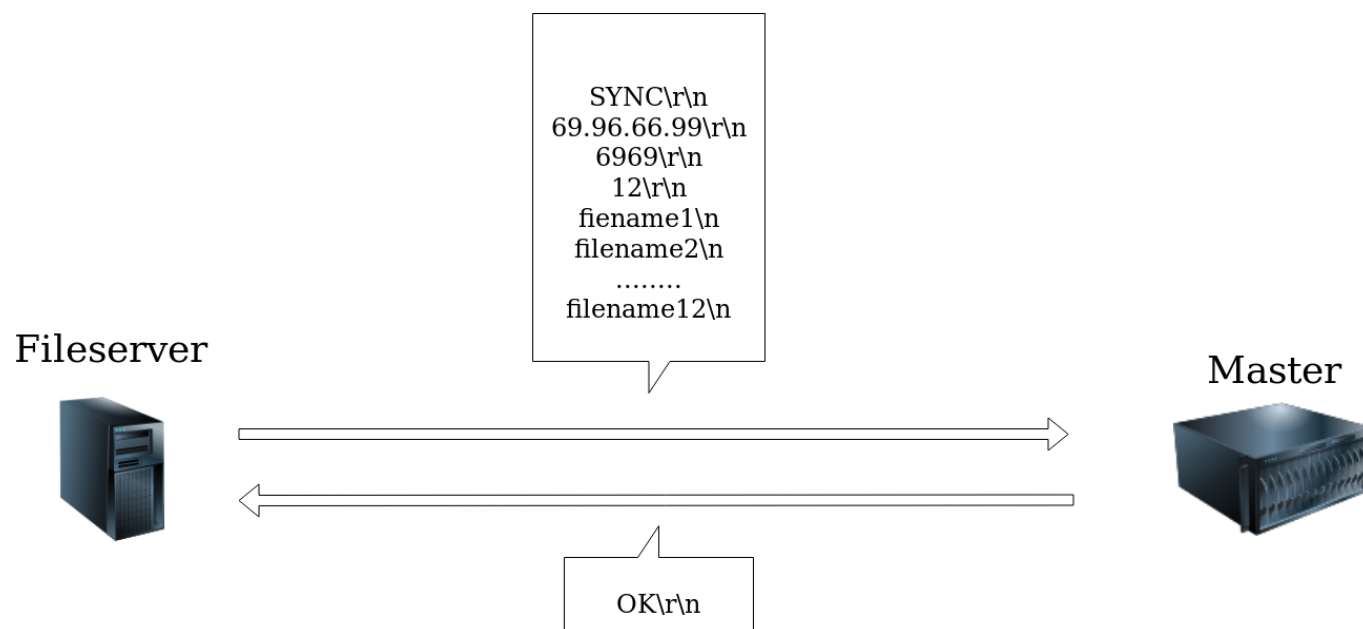
- Đồng bộ danh sách các node fileserver (linked list) bằng mutex (multithread thêm xóa sửa trên critical section)
- Masterserver, fileserver có khả năng xử lý nhiều request 1 lúc
- Client sử dụng multithread tạo nhiều connection => có khả năng tải nhiều file cùng 1 lúc
- **check keep alive**
 - Masterserver sử dụng 1 thread chạy vòng lặp sau mỗi 2s, gửi PING request đến ip và port, check heartbeat của từng fileserver trong list, nếu bị mất kết nối sẽ xóa fileserver ra khỏi list

3.2. Thiết kế giao thức

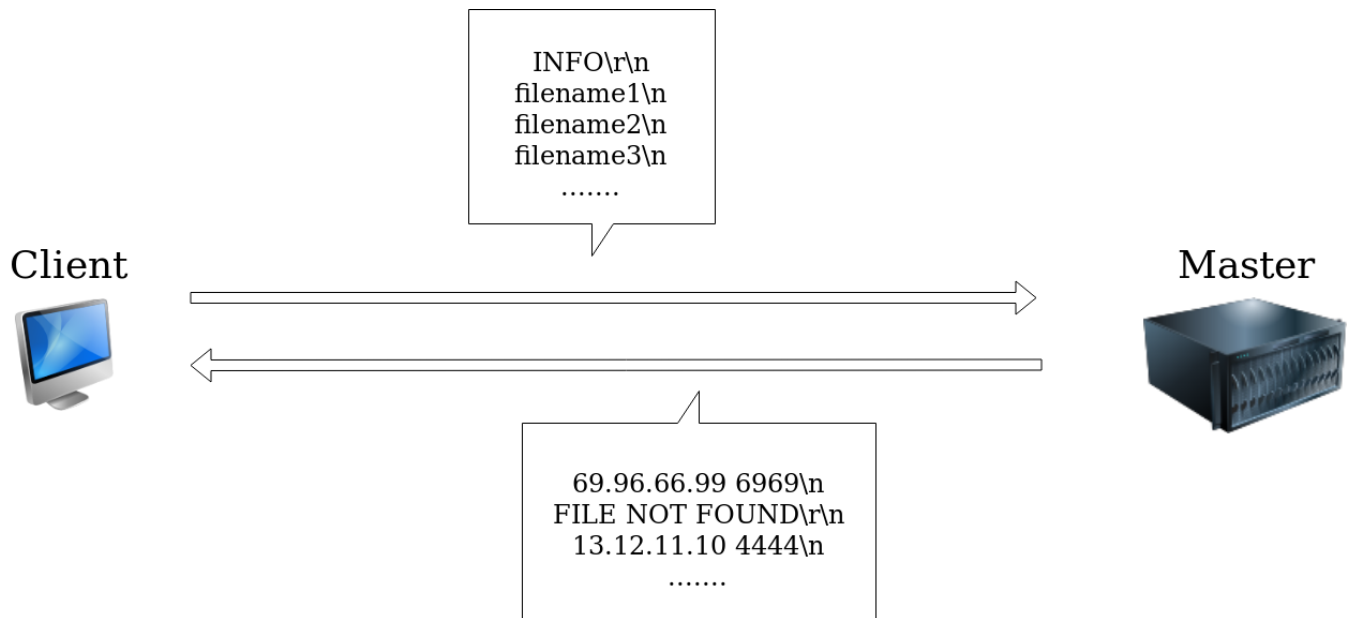
3.2.1. PING method



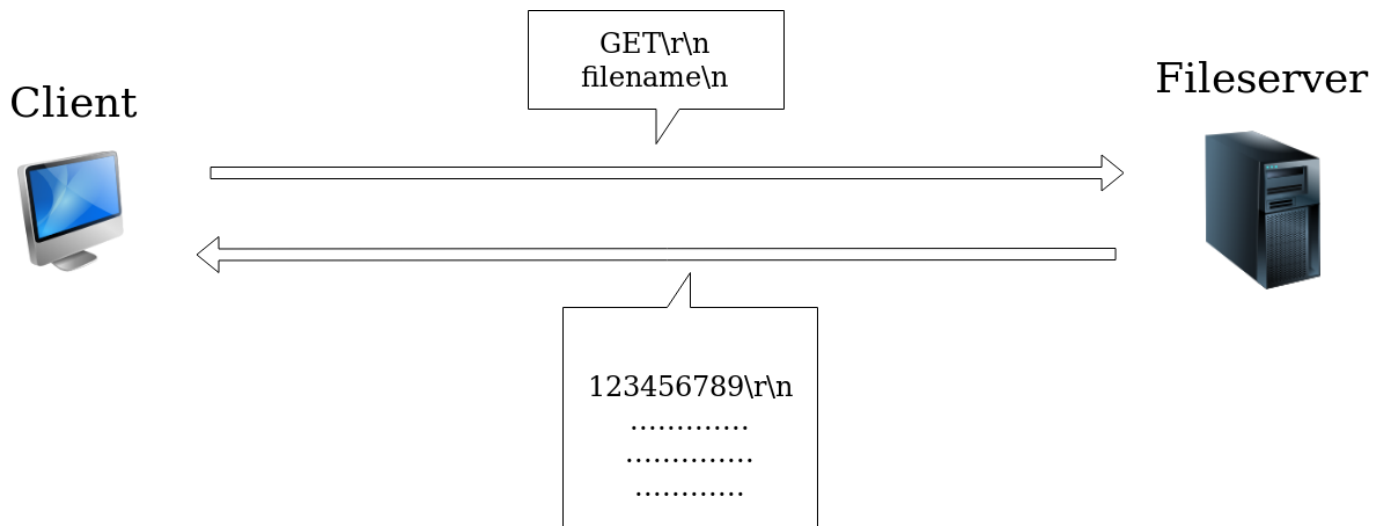
3.2.2. SYNC method



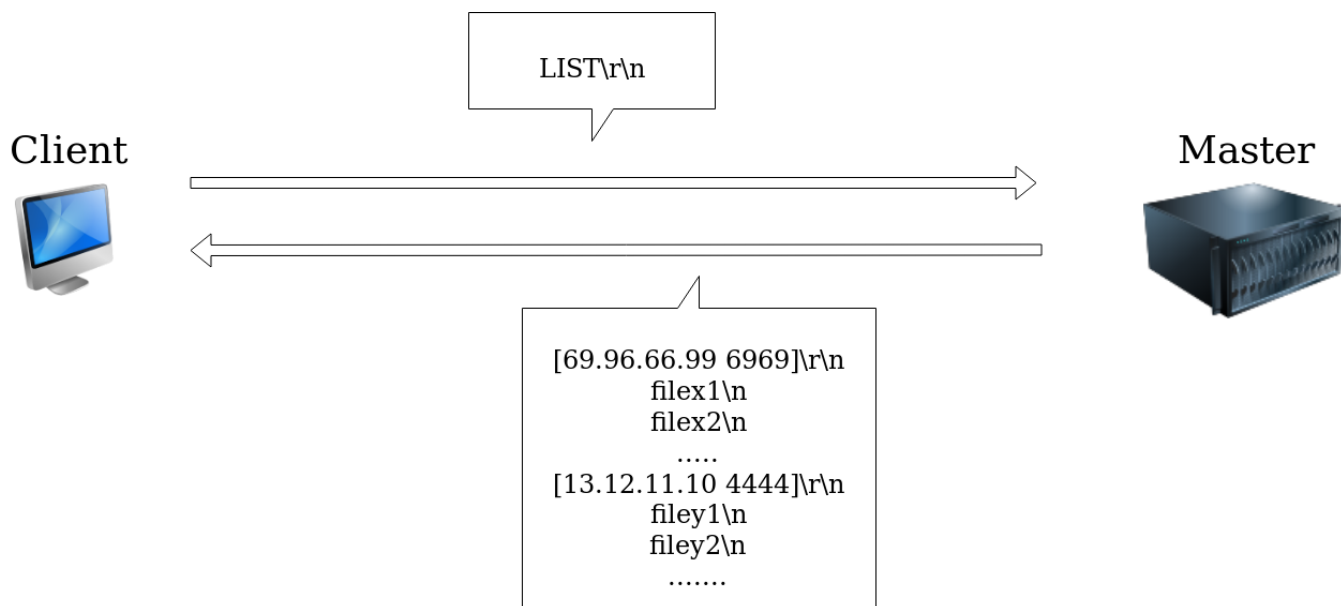
3.2.3. INFO method



3.2.4. GET method

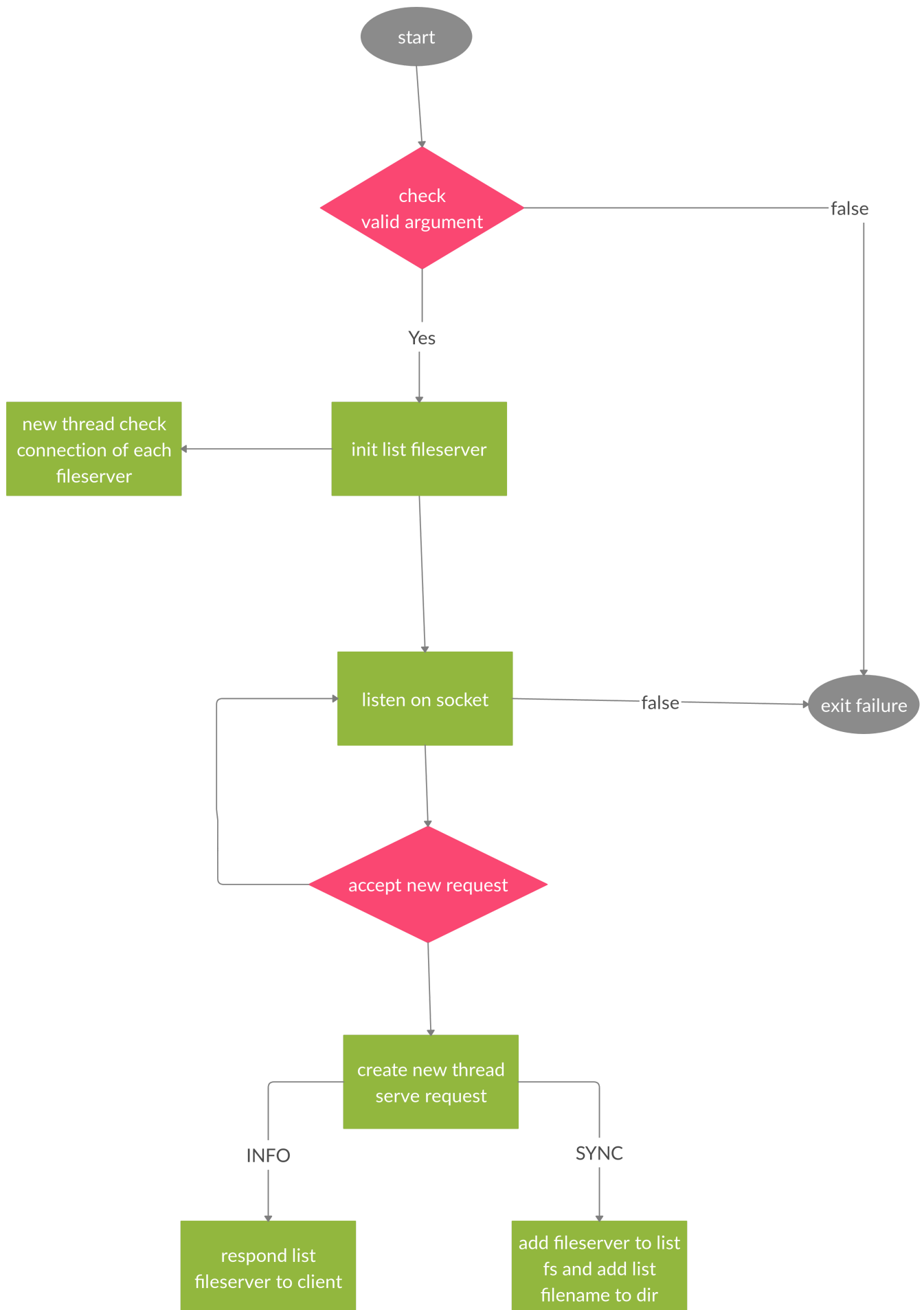


3.2.5. LIST method



3.3. Chi tiết vận hành

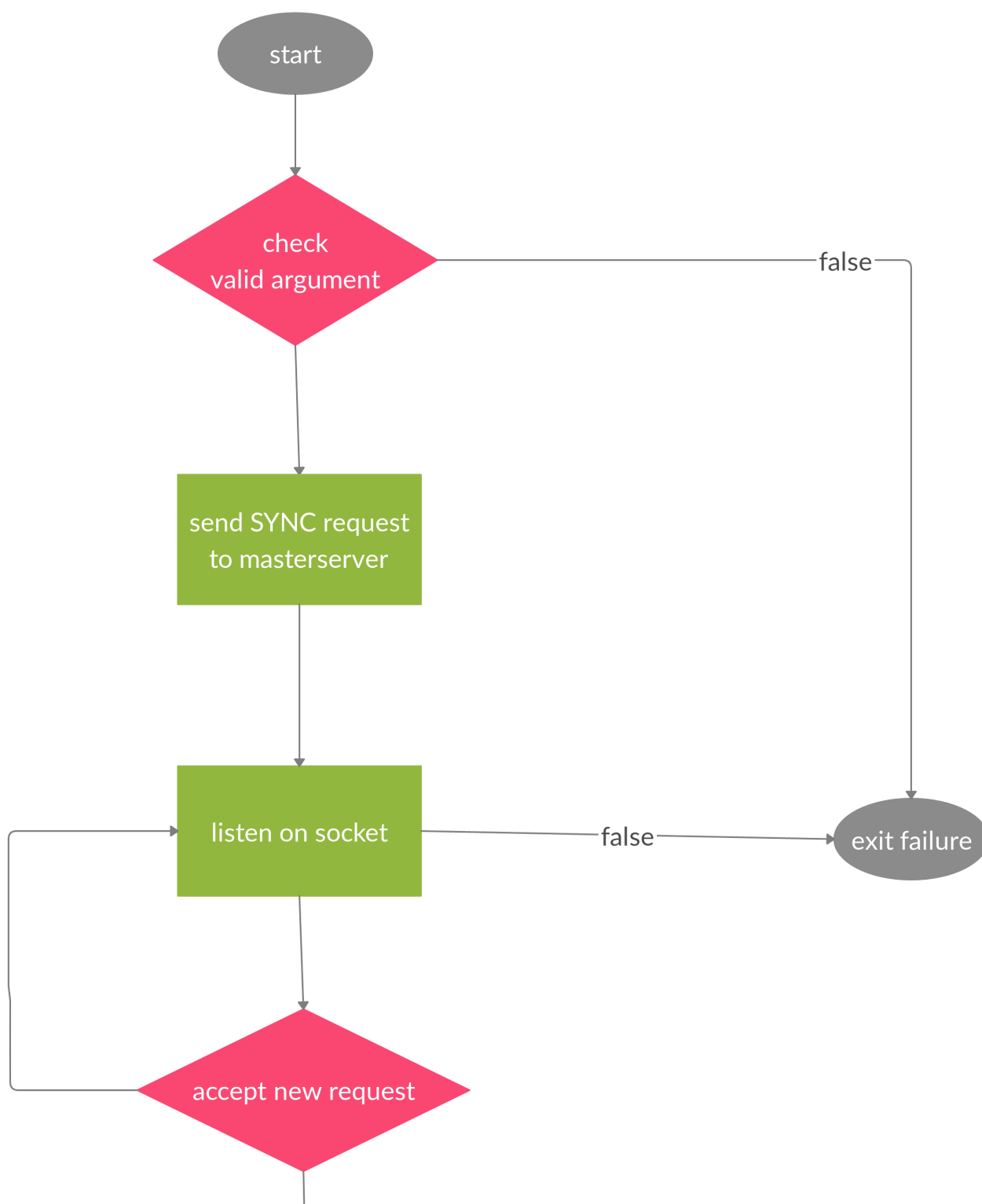
3.3.1 Master

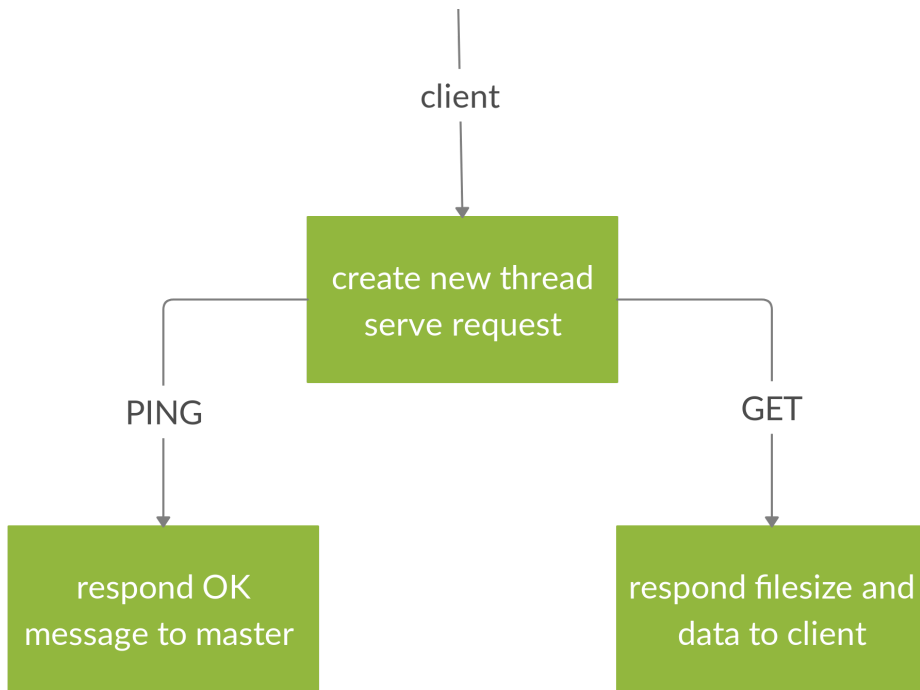


- Fileserver nhận tham số dòng lệnh, lấy thông tin port và thư mục sẽ lưu thông tin fileserver

- Tạo list_fs (list fileserver) là một linked list
- tạo 1 thread dùng để check connection của từng fileserver trong list_fs
- Tạo socket, listen trên port đã xác định
- Nhận request, tạo 1 thread detach khỏi main thread để phục vụ:
 - Nếu là SYNC request, add fileserver vào list_fs, tạo thư mục định dạng là IP_PORT để chứa danh sách các file
 - Nếu là INFO request, trả về client thông tin các fileserver tương ứng với danh sách file
 - Nếu là LIST request, lấy toàn bộ thông tin các fileserver cùng với filename tương ứng trả về cho client

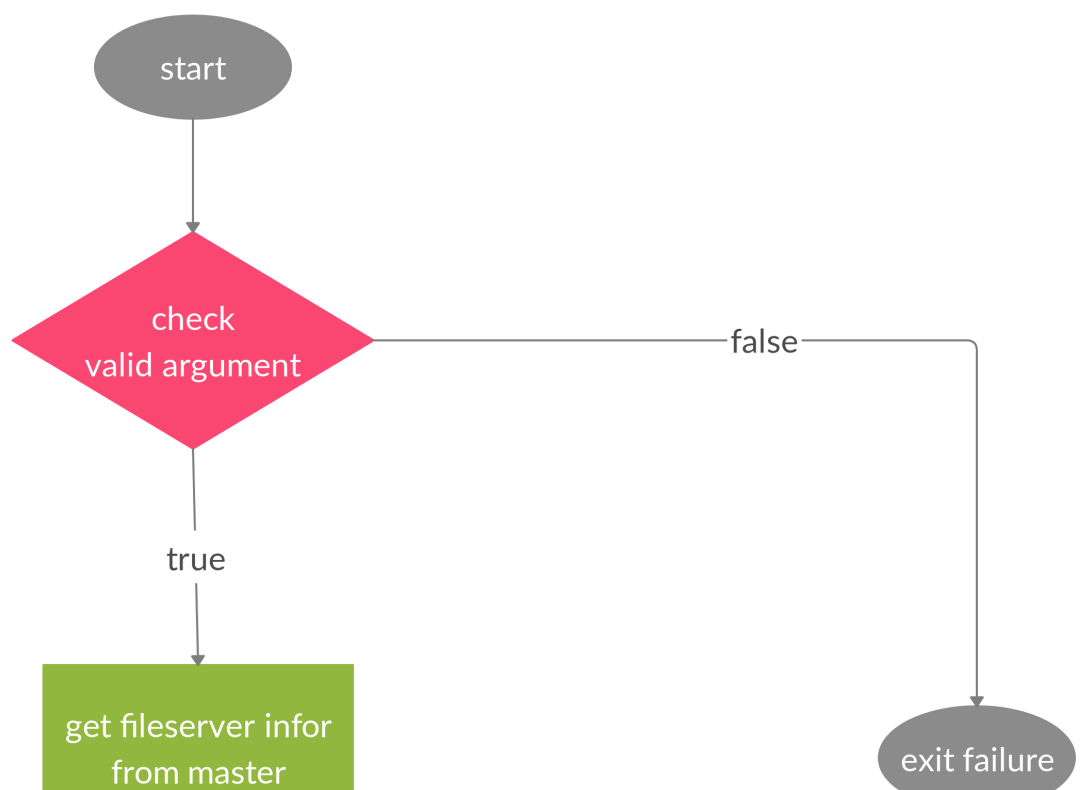
3.3.2 Fileserver

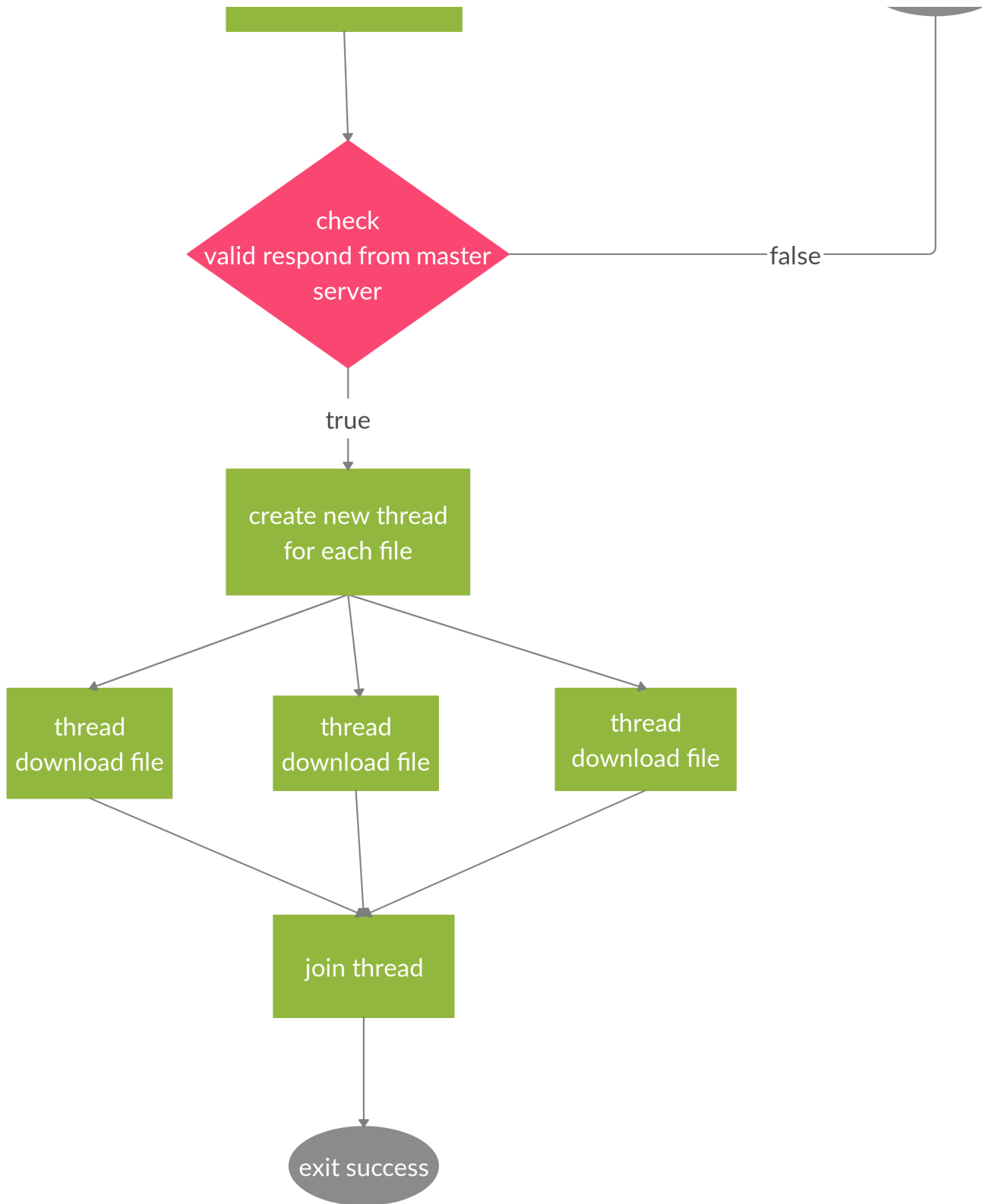




- Fileserver nhận tham số dòng lệnh, lấy thông tin port sẽ listen, master ip port và thư mục sẽ chia sẻ file
- Tạo socket, bind, listen trên port đã xác định
- Gửi SYNC request đến master server để đồng bộ file, cấu trúc là thông tin IP_PORT cùng với danh sách các file cần đồng bộ
- Nhận request, tạo 1 thread detach khỏi main thread để phục vụ:
 - Nếu là masterserver PING thì respond OK
 - Nếu là client GET thì trả về filesize cùng với data, trường hợp ko tìm thấy file trả về FNF (file not found)

3.3.3 Client





- Client nhận tham số dòng lệnh từ user, lấy thông tin master server ip, port và danh sách các file cần tải
- Client nhận thông tin các fileserver, tạo nhiều thread (connection) để tải từng file
- Sau khi các thread đã tải xong, join thread và kết thúc chương trình

4. Usage

Chúng ta có thể sử dụng cờ **-h** để xem hướng dẫn sử dụng.

Master:

```
$ ./master -h
Usage: master -p <PORT>
Master server centralize fileserver info storage

-p, --port          listen port  (default 55555)
-d, --dir           directory to store fileserver info (must is empty dir)
                    (default $HOME/.master)-h, --help        display help
message then exit
-v, --version       output version information then exit
```

Fileserver:

```
$ ./fileserver -h
fileserver [-i <IP>] [-m <PORT>] [-p <PORT>] -d <DIR>
allow client download file in storage directory

-i, --master-ip     master server ip (default 127.0.0.1)
-m, --master-port   master server listening port (default 55555)
-p, --listen-port   listen port  (default 44444)
-d, --dir           storage directory to download
-h, --help          display help message then exit
-v, --version       output version information then exit
```

Client:

```
$ client -h
client [-i <IP>] [-p <PORT>] [-o <OUTPATH>] <FILE> ...
Get fileserver info from master server then download file from fileserver

-i, --host-ip       host ip of master server (default 127.0.0.1)
-p, --port          listening port of master server (default 55555)
-o, --out-dir       out dir for downloaded file (default ./)
-l, --list          list all file on master server
-h, --help          display help message then exit
-v, --version       output version information then exit

By default, this software can not download directory, just file only :)
This version still not support hostname yet :D
To download a file whose name start with a '-', for example '-foo', use
command:
client [OPTIONS] -- -foo
```

5. Demo

Link YouTube:

<https://youtu.be/xiSovxvSRFk>

6. Tham khảo

- <https://dev.to/mattcanello/multithreaded-programming-lmh>
- <http://www.kegel.com/c10k.html#strategies>
- <https://linux.die.net/man/2/setsockopt>
- <https://www.geeksforgeeks.org/mutex-lock-for-linux-thread-synchronization/>
- <http://www.cs.kent.edu/~ruttan/sysprog/lectures/multi-thread/multi-thread.html>
- <https://stackoverflow.com/questions/612097/how-can-i-get-the-list-of-files-in-a-directory-using-c-or-c>
- <https://stackoverflow.com/questions/4553012/checking-if-a-file-is-a-directory-or-just-a-file>
- <https://www.cs.rpi.edu/~moorthy/Courses/os98/Pgms/socket.html>