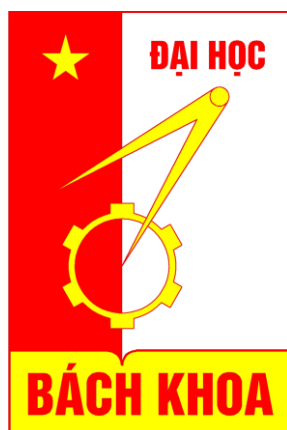


TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG



BÁO CÁO BÀI TẬP LỚN
Môn: Thực hành lập trình mạng

ĐỀ TÀI:

XÂY DỰNG DỊCH VỤ QUẢN LÝ FILE

Giảng viên : Ts. Trần Hải Anh

Sinh viên thực hiện:

Nguyễn Hữu Tiến - 20133945
Lê Tất Tùng - 20134394

Hà Nội, tháng 12 năm 2016

Mục lục

Phân tích yêu cầu	4
Tóm tắt nội dung báo cáo.....	5
1. Cơ sở lý thuyết	6
1.1 Lập trình socket trên Linux	6
1.2 Kết nối với database (mysql) sử dụng ngôn ngữ lập trình C	7
2. Xây dựng chương trình	8
2. 1 Tổng quan chương trình	8
2.2 Tổ chức của thông tin gửi nhận giữa server - client.....	8
3. Cài đặt và kiểm thử	10
3. 1 Cài đặt chương trình	10
3. 2 Kiểm thử và mô phỏng chương trình	11
4. Kết luận và hướng phát triển.....	16
4.1 Kết luận.....	16
Về chức năng.....	16
Về hiệu năng	16
Về quá trình làm việc	17
4.2 Hướng phát triển.....	17

LỜI CẢM ƠN

Trên thực tế không có sự thành công nào mà không gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp của người khác. Trong suốt thời gian từ khi bắt đầu học tập ở giảng đường đại học đến nay, em đã nhận được rất nhiều sự quan tâm, giúp đỡ của quý Thầy Cô, gia đình và bạn bè. Với lòng biết ơn sâu sắc nhất, em xin gửi đến quý Thầy Cô ở viện Công Nghệ Thông Tin trường đại học Bách Khoa Hà Nội đã cùng với tri thức và tâm huyết của mình để truyền đạt vốn kiến thức quý báu cho chúng em trong suốt thời gian học tập tại trường. Và đặc biệt, trong học kỳ này, việc đã tổ chức cho chúng em được tiếp cận với môn học mà theo em là rất hữu ích đối với sinh viên công nghệ thông tin. Đó là môn học “thực hành lập trình mạng”.

Chúng em xin chân thành cảm ơn TS. Trần Hải Anh đã tận tâm hướng dẫn chúng em qua từng buổi học trên lớp cũng như những buổi nói chuyện, thảo luận về môn học. Nếu không có những lời hướng dẫn, dạy bảo của thầy thì em nghĩ bài project này của em rất khó có thể hoàn thiện được. Một lần nữa, chúng em xin chân thành cảm ơn thầy. Bước đầu đi vào thực tế, tìm hiểu về socket, connect mysql, kiến thức của em còn hạn chế và còn nhiều ngỡ ngàng. Do vậy, không tránh khỏi những thiếu sót là điều chắc chắn, em rất mong nhận được những ý kiến đóng góp quý báu của quý Thầy Cô và các bạn học cùng lớp để kiến thức của em trong môn học này được hoàn thiện hơn.

Phân tích yêu cầu

Tên đề tài là:

XÂY DỰNG DỊCH VỤ QUẢN LÝ FILE.

Yêu cầu chương trình phải cung cấp được các chức năng sau :

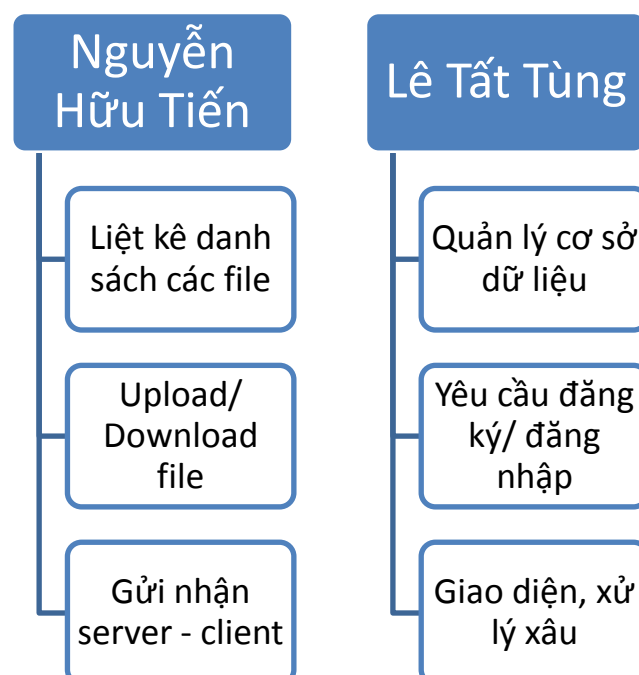
I. Server:

1. Tổ chức lưu trữ các file
2. Lưu trữ danh sách file, gửi cho client nếu được yêu cầu
3. Quản lý cơ sở dữ liệu (mysql) để lưu trữ thông tin đăng ký cho các user
4. Gửi file mà user yêu cầu.
5. Nhận và lưu trữ lại file mà user upload lên.

II. Client:

1. Yêu cầu đăng ký/đăng nhập
2. Liệt kê danh sách file mà server quản lý
3. Cho người dùng chọn và gửi yêu cầu tải file về

Phân công nhiệm vụ



Tóm tắt nội dung báo cáo

Báo cáo gồm có 4 chương:

1. Chương 1: Cơ sở lý thuyết.
2. Chương 2: Xây dựng chương trình.
3. Chương 3: Cài đặt và kiểm thử.
4. Chương 4: Kết luận.

Cụ thể nội dung từng chương như sau:

Chương	Nội dung
Cơ sở lý thuyết	Lập trình socket trên Linux Kết nối mysql bằng ngôn ngữ C
Xây dựng chương trình	Tổng quan, phương thức xây dựng chương trình
Cài đặt và kiểm thử.	1. Cài đặt chương trình 2. Hướng dẫn chạy chương trình 3. Kiểm thử
Kết luận	Nhận xét về chức năng – hiệu năng và quá trình làm việc.

1. Cơ sở lý thuyết

1.1 Lập trình socket trên Linux

Socket

Socket là một phương pháp để thiết lập kết nối truyền thông giữa một chương trình yêu cầu dịch vụ (client) và một chương trình cung cấp dịch vụ (server) trên mạng LAN, WAN hay Internet và đôi lúc là giữa những quá trình ngay bên trong máy tính. Mỗi socket có thể được xem như một điểm cuối trong một kết nối. Một socket trên máy yêu cầu dịch vụ có địa chỉ mạng được cấp sẵn để “gọi” một socket trên máy cung cấp dịch vụ. Một khi socket đã được thiết lập phù hợp, hai máy tính có thể trao đổi dịch vụ và dữ liệu.

Một Socket Address bao gồm: host name và port. Ví dụ: socket(127.0.0.1; 80)//cổng dịch vụ của HTTP

Trên Windows, thư viện Winsock được nhiều người sử dụng để lập trình mạng trên Windows. Tuy nhiên trong yêu cầu của chương trình là làm việc trong môi trường mã nguồn mở. Em xin được trình bày những nội dung về lập trình C Socket trên UNIX, với thư viện:

```
#include <sys/types.h>
```

```
#include <sys/socket.h>
```

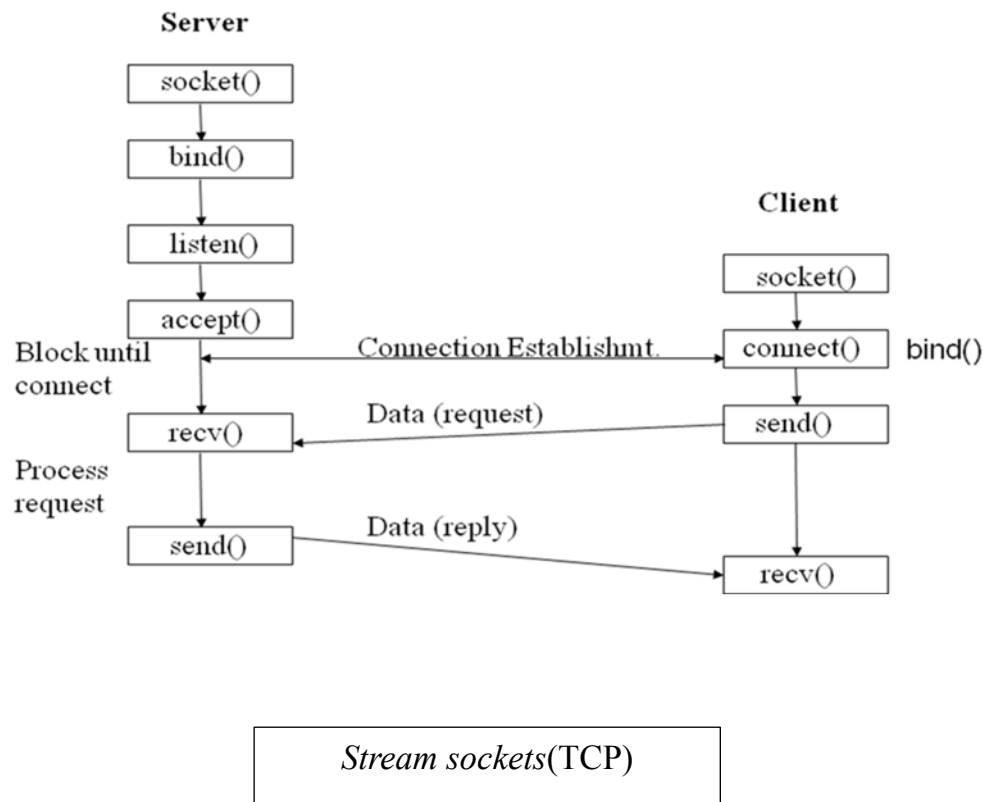
```
#include <netinet/in.h>
```

```
#include <arpa/inet.h>
```

Sử dụng socket trong project

Có hai loại socket: stream sockets(SOCK_STREAM) và datagram sockets(SOCK_DGRAM). Trong project này, nhóm chúng em sử dụng SOCK_STREAM. Stream sockets sử dụng giao thức TCP(Transmission Control Protocol) nên rất đáng tin cậy. Stream thường được dùng để chuyển các gói dữ liệu nối tiếp nhau(thứ tự của các gói dữ liệu(packets) là điều quan trọng nhất). Ví dụ, bạn có 3 gói dữ liệu "A,B,C", stream sockets sẽ chuyển theo đúng thứ tự "A,B,C".

Mô hình stream sockets :



1.2 Kết nối với database (mysql) sử dụng ngôn ngữ lập trình C

MySQL là một hệ thống quản lý cơ sở dữ liệu mã nguồn mở hàng đầu. Nó là một đa người dùng, hệ thống quản lý cơ sở dữ liệu đa luồng. MySQL là đặc biệt phổ biến trên web. Nó là một phần của nền tảng LAMP rất phổ biến bao gồm Linux, Apache, MySQL, và PHP. MySQL hiện thuộc sở hữu của Oracle. cơ sở dữ liệu MySQL có sẵn trên hầu hết các nền tảng hệ điều hành quan trọng. Nó chạy trên BSD Unix, Linux, Windows. Những trang web quản lý hàng triệu truy vấn mỗi ngày. MySQL có hai phiên bản: hệ thống máy chủ MySQL và hệ thống nhúng MySQL.

\$ Sudo apt-get install libmysqlclient-dev

Để có thể biên dịch ví dụ C, chúng ta cần phải cài đặt các thư viện phát triển MySQL C. Dòng trên cho thấy cách chúng ta có thể làm điều đó trên dựa trên Debian Linux.

Các thư viện MYSQL C:

```
#include <my_global.h>
```

```
#include <mysql.h>
```

2. Xây dựng chương trình

2.1 Tổng quan chương trình

Trong bài làm nhóm em sử dụng stream sockets(SOCK_STREAM) vì các lý do sau:

- Dung lượng các thông tin truyền không quá lớn, không đòi hỏi cao về tốc độ.
- Tuy nhiên tính tin cậy cần phải được đảm bảo để các thông tin gửi nhận giữa server - client được đầy đủ và toàn vẹn, không làm mất mát hay sai lệch thông tin.

2.2 Tổ chức của thông tin gửi nhận giữa server - client

Trong project, nhóm em quyết định sử dụng một struct Object để làm đối tượng lưu trữ thông tin gửi nhận giữa server - client. Cấu trúc Object được tổ chức như sau:

```
typedef struct Obj {  
    char method[METHOD_LEN];  
    char typeData[TYPE_DATA_LEN];  
    char fileName[FILE_NAME_LEN];  
    char file[FILE_LEN];  
    int fileSize;  
    char listFile[ARRAY_FILE_LEN][FILE_NAME_LEN];  
};
```



```

int sizeListFile;

int userID;

char userName[USER_NAME_LEN];

char passWord[PASS_WORD_LEN];

int status;

} Object;

```

Trong đó :

+ method:

- Ứng với các chức năng Login, SignUp, Upload file thì method là “POST” -tức là đẩy dữ liệu đầu vào từ client lên server.
- Ứng với các chức năng lấy danh sách file đã upload, download file thì method là “GET” - tức là lấy dữ liệu yêu cầu từ server gửi xuống cho client.

+typeData:

- Ứng với chức năng Login thì typeData là “LOG_IN”.
- Ứng với chức năng Sign up thì typeData là “SIGN_UP”.
- Ứng với chức năng lấy danh sách file client đã gửi thì typeData là “DATAS”.
- Ứng với chức năng Download file thì typeData là “DATA”.
- Ứng với chức năng Upload file thì typeData là “DATA”.

+fileName là tên của file trong quá trình gửi nhận.

+file : nội dung của file gửi nhận

+listFile: danh sách tên các file mà user upload

+sizeListFile: số lượng file mà user đã upload

- +userID: id của user lưu trong database
- +userName: tên đăng nhập của account user trong database
- +passWord: pass word đăng nhập của account user trong database.
- +status: trạng thái của account. Nếu status = 1 tức là account đang trong trạng thái đăng nhập, status = 0 tức là account đã đăng xuất.
- +Note: Các hằng số quy định về độ dài, số lượng được lưu trữ trong define.c

3. Cài đặt và kiểm thử

3.1 Cài đặt chương trình

- Hệ thống được phát triển trên Linux với trình biên dịch GCC và thư viện lập trình Socket có sẵn là:

```
#include <sys/types.h>
```

```
#include <sys/socket.h>
```

```
#include <netinet/in.h>
```

```
#include <arpa/inet.h>
```

- Ngoài ra còn sử dụng các thư viện hỗ trợ cho việc kết nối database ở MYSQL. Đó là:

```
#include <my_global.h>
```

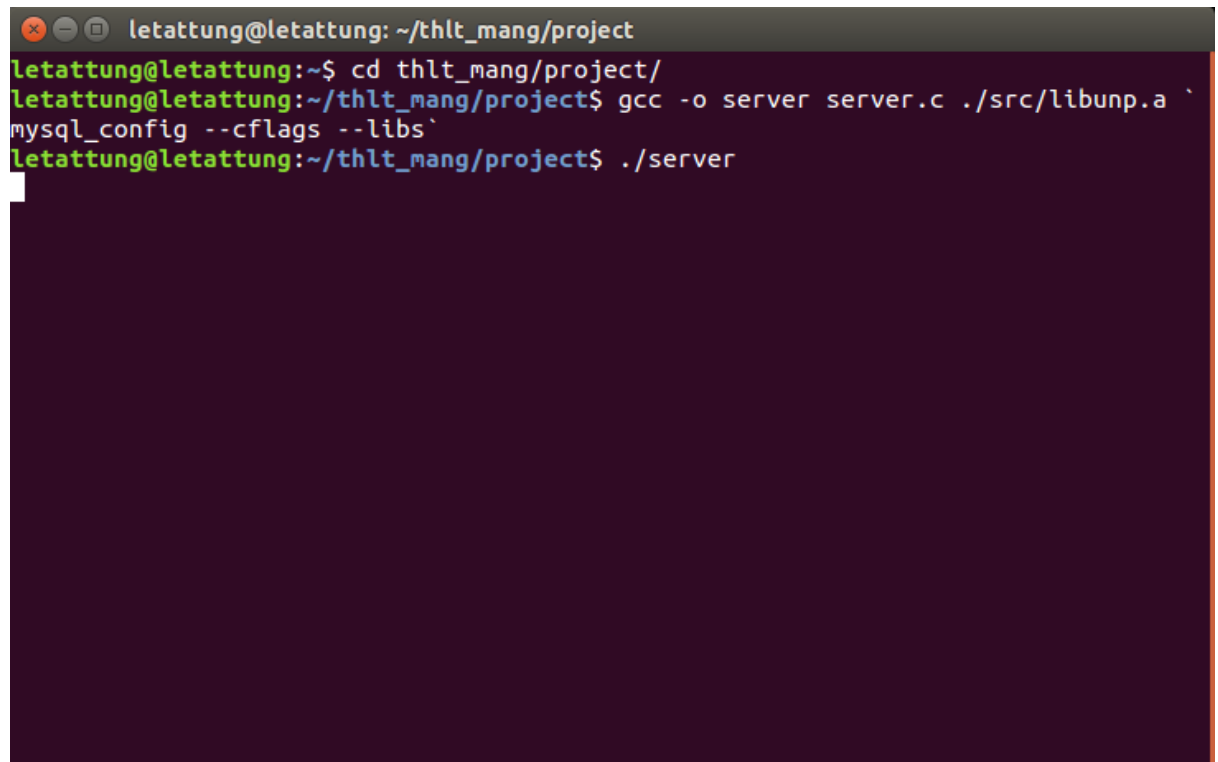
```
#include <mysql.h>
```

- Phân gửi nhận theo từng chức năng yêu cầu được cài đặt trong file server.c và client.c
- Phần khai báo các hằng số, độ dài xâu, các thông tin hệ thống được khai báo trong file define.c
- Phần nhập và kiểm tra đầu vào của chương trình cài đặt trong file input.c

- Phần kết nối, truy vấn đến database được cài đặt trong sql.c
- Phần hỗ trợ liên quan đến nhập liệu, xử lý các đường dẫn file được cài đặt trong file support.c
- Phần tạo ra các câu lệnh truy vấn database được cài đặt trong string_handling.c

3.2 Kiểm thử và mô phỏng chương trình

1. Khởi tạo server



```

letattung@letattung: ~/thlt_mang/project
letattung@letattung:~$ cd thlt_mang/project/
letattung@letattung:~/thlt_mang/project$ gcc -o server server.c ./src/libunp.a `
mysql_config --cflags --libs`
letattung@letattung:~/thlt_mang/project$ ./server

```

2. Khởi tạo client

```
letattung@letattung: ~/thlt_mang/project
letattung@letattung:~$ cd thlt_mang/project/
letattung@letattung:~/thlt_mang/project$ gcc -o client client.c
letattung@letattung:~/thlt_mang/project$ ./client
Trang dang nhap
1. Dang ki tai khoan
2. Dang nhap
3. Thoat
Ban chon: 
```

3. Thực hiện chức năng sign up (đăng ký tài khoản mới)

```
letattung@letattung: ~/thlt_mang/project
letattung@letattung:~$ cd thlt_mang/project/
letattung@letattung:~/thlt_mang/project$ gcc -o client client.c
letattung@letattung:~/thlt_mang/project$ ./client
Trang dang nhap
1. Dang ki tai khoan
2. Dang nhap
3. Thoat
Ban chon: 1
Nhap thong tin cho tai khoan
Nhap username: nguyenuutien
Nhap password: tien123
Dang ki thanh cong!!

Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 
```

4. Tài khoản mới đã được cập nhật vào database

+ Options

Id	UserName	PassWord
1	letattung	tung123
2	tung123	tung123
3	nguyenuutien	tien123

5. Thực hiện chức năng xem danh sách các file đã upload

```
Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 1
Khong co file nao!!

Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 
```

Do tài khoản mới vừa được tạo ra và chưa upload file nào nên hệ thống không tìm thấy file nào đã được upload.

6. Thực hiện chức năng upload file

```
Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 3
Nhap ten file: ./test/text.txt
Up file thanh cong!!

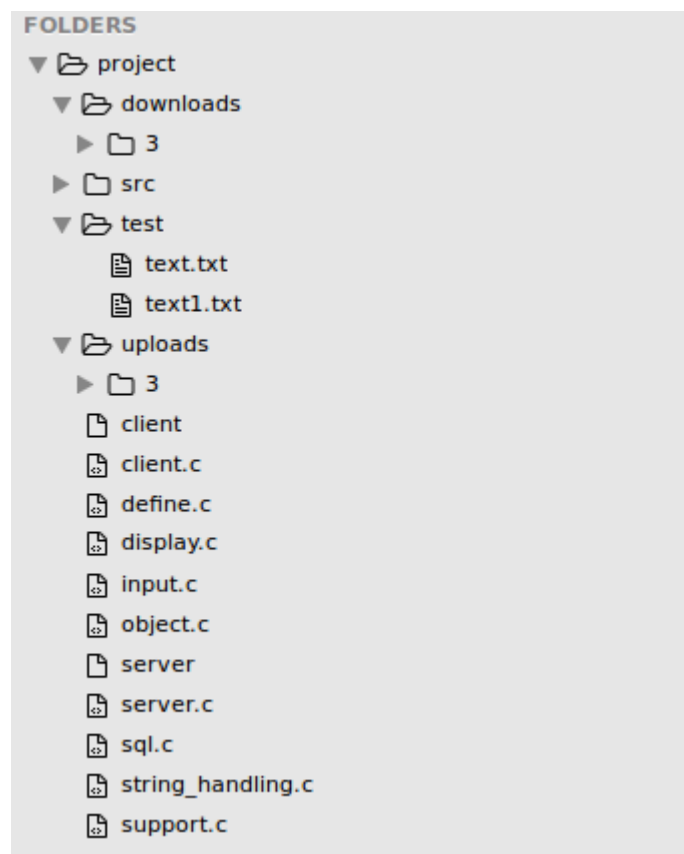
Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 
```

7. Thực hiện chức năng download file

```
Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 2
Nhap ten file: text.txt
Down file thanh cong!!

Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 
```

8. Thư mục download và upload



Mỗi lần client thực hiện download hoặc upload file thành công, hệ thống sẽ tự động thêm file vào folder có tên chính là id của account vừa thực hiện yêu cầu.

9. Thực hiện chức năng login (với một tài khoản đã tồn tại)

```
Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 4
Trang dang nhap
1. Dang ki tai khoan
2. Dang nhap
3. Thoat
Ban chon: 2
Nhap thong tin cho tai khoan
Nhap username: letattung
Nhap password: tung123
Dang nhap thanh cong!!

Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 
```

10. Thực hiện chức năng đăng xuất và kết thúc chương trình

```
Trang quan ly file
1. Danh sach cac file ma ban da upload
2. Download file
3. Upload file
4. Dang xuat
Ban chon: 4
Trang dang nhap
1. Dang ki tai khoan
2. Dang nhap
3. Thoat
Ban chon: 3
letattung@letattung:~/thlt_mang/project$ 
```

4. Kết luận và hướng phát triển

4.1 Kết luận

Về chức năng

Chương trình đã thực hiện đầy đủ các yêu cầu mà bài toán đặt ra. Cụ thể:

I. Server:

- Tổ chức lưu trữ các file
- Lưu trữ danh sách file, gửi cho client nếu được yêu cầu
- Quản lý cơ sở dữ liệu (mysql) để lưu trữ thông tin đăng ký cho các user
- Gửi file mà user yêu cầu.
- Nhận và lưu trữ lại file mà user upload lên.

II. Client

- Yêu cầu đăng ký/đăng nhập
- Liệt kê danh sách file mà server quản lý
- Cho người dùng chọn và gửi yêu cầu tải file về

Về hiệu năng

I. Ưu điểm:

- Server có thể phục vụ được nhiều Client truy cập tới.
- File mã hóa có kích thước lớn vẫn không ảnh hưởng nhiều đến thời gian trả file sau khi Server đã mã hóa hay giải mã.

II. Nhược điểm:

- Chương trình còn thiếu giao diện đồ họa để có thể tối đa sức mạnh tương tác người dùng.

Về quá trình làm việc

Đã có sự tái sử dụng mã nguồn , phân chia các chức năng, cách tổ chức file và định dạng gửi nhận rõ ràng. Cụ thể:

- Các chức năng login, signup, download, upload , liệt kê danh sách file được phân chia cụ thể trong server và client
- Các hàm có chức năng tương tự nhau được lưu trữ và cài đặt trong các file riêng biệt, tiện lợi cho việc phát triển và bảo trì như : `string_handling.c`, `object.c`, `sql.c`, `input.c`, `display.c`

4.2 Hướng phát triển

Xây dựng đồ họa cho chương trình. Nhóm em đề xuất 1 giải pháp để thực hiện phần đồ họa là sử dụng GTK – phần mềm mã nguồn mở được sử dụng nhiều trong việc thiết kế đồ họa cho các chương trình chạy trên Linux.

TÀI LIỆU THAM KHẢO

[1] Sile bài giảng và vở ghi môn học Thực hành lập trình mạng do thầy giáo Trần Hải Anh giảng dạy.

[2] Các trang web về lập trình C, lập trình socket, c connect to mysql

<http://stackoverflow.com/questions/3453168/c-program-mysql-connection>

<http://zetcode.com/db/mysqlc/>