Tên: Trần Hưng Thuận MSSV: 0812508

Tên: Trần Nguyễn Hoàng Thông MSSV: 0812502

***Báo Cáo Cuối Kỳ Môn Hệ Điều Hành Nâng Cao***

1. ***Mô Hình Cài Đặt:***
2. Server có cài đặt 3 lớp sau:
3. Lớp File:

* Các Thuộc Tính:
* String m\_sTenFile: chứa tên file.
* String m\_sUserGiuFile: chứa tên user giữ file đó
* String m\_bTinhTrangDownLoad: tình trạng download file này. Nếu =”Co”, file này đang có user khác đang download. Và nếu =”khong” thì ngược lại.
* String m\_bTinhTrangUser: tình trạng user chứ file này. Nếu =”online” thì user này đang có kết nối với server. Và nếu =”offline” thì ngược lại.
* Queue<String> HangDoiDownLoad: dùng để chứa tên các user muốn download file này trong khi file này được downlaod bởi 1 user khác.
* Các Hàm:
* File(): khời tạo
* Công Dụng:
* Dùng để lưu trữ thông tin của 1 file được chia sẻ trên server.

1. Lớp User:

* Các Thuộc Tính:
* string m\_sTenUser.
* string m\_sPassword.
* System.Net.IPAddress m\_IpAddress: dia chi ip của user này.
* int m\_iPort: so port socket dùng lắng nghe.
* Các Hàm:
* Hàm khởi tạo: public User();
* Hàm gán giá trị cho 1 đối tượng File: void LayThongTin(string name, string pass, string ip, string port)
* Công Dụng:
* Dùng để lưu trữ thông tin của 1 user đã đãng ký lên server.

1. Lớp Server:

* Các Thuộc Tính:
* static List<File> m\_arrDanhSachFile: dùng để lưu trữ danh sách các đối tượng File đang được các client chia sẻ . (thêm static để dùng chung các thread)
* static List<User> m\_arrDanhSachUser: danh sách các đối tượng user đã đăng ký cho server. (thêm static để dùng chung các thread)
* static Mutex ms: dùng để bảo vệ miền đụng độ
* Socket m\_socServer: lắng nghe các kết nối từ các user.
* int m\_iPort: số port của socket m\_socServer.
* int m\_iSoClient: so user kết nối với server.
* Socket[] m\_socLamViec: 1 mang các socket. Khi có 1 client kết nối đến thì sẽ được cấp 1 socket phục vụ riêng cho clinet đó.
* Các Hàm:
* public static OleDbConnection KetNoi(): dùng để kết nối cơ sở dữ liệu và gọi lại hàm này khi viết các hàm thao tác trên cơ sở dữ liệu khác. Cơ sở dữ liệu là file CSDL.mdb nằm cùng thư mục với file .exe của chương trình. (thêm static để dùng chung các thread)
* public static List<User> LayDanhSachUser(): lấy danh sách các user từ cơ sờ dữ liệu. (thêm static để dùng chung các thread).
* public static List<File> layDanhSachFile(): lấy danh sách các file được lưu trên cơ sở dữ liệu.(thêm static để dùng chung các thread).
* public static bool themUserMoi(User UserNew): thêm thông tin 1 user mới vào cơ sở dữ liệu khi có 1 user đăng ký lên server. (thêm static để dùng chung các thread)
* public static bool themFileTrenCoSoDuLieu(File FileNew): thêm thông tin 1 file mới vào cơ sở dữ liệu khi 1 user thêm chia sẻ file mới. (thêm static để dùng chung các thread)
* public static bool xoaFileTrenCoSoDuLieu(string User, string File): xóa thông tin của 1 file trên cơ sở dữ liệu khi 1 user thông báo xóa 1 file chia sẻ. (thêm static để dùng chung các thread)
* public Server():hàm dựng khởi tạo mặc định các thuộc tính trên đối tượng server.
* public void LangNghe():tạo ra socket để lắng nghe các kết nối đến server.Sử dụng kiểu lắng nghe non-blocking(không đồng bộ)
* public void ClientKetNoi(IAsyncResult asyn):hàm xử lí khi có kết nối đến server,sẽ tạo ra 1 thread và gọi hàm LamViec() để phục vụ cho client kết nối tới và gọi lại hàm lắng nghe cho server.
* public void LamViec():phục vụ cho client kết nối tới.
* public string PhanTichGoiTin(string s,Socket soc):phân tích thông tin từ client gửi đến thông qua các kí tự đầu tiên làm cờ:

1-user's name-password : là gửi yêu cầu đăng nhập

2-tên file-user's name yeu cau :là gửi yêu cầu tìm kiếm

3-user's name-password-port : là gửi yêu cầu đăng kí

4-user's name: la gửi yêu cầu đăng xuất

5-ten file-user's name: gửi yêu cầu thêm 1 file cho chia sẻ.

6-ten file-user's name: gui yeu cau xoa 1 file ko chia se nua

7-ten file-user's name: gửi trả lời đã cho down load xong file này

8-ten file-user's name chưá file-user name mún down load: gưỉ yêu câù down load

9-user's name-da online:thong bao xac nhan da online chua(chinh xac la de xac nhan da mo cong lang nghe chua)

Và gửi thông báo lại cho client.

* public void DownOffline(object name):phục vụ cho việc download offline.
* public string DangNhap(string s,Socket soc):xử lí khi có yêu cầu đăng nhập từ client.
* public List<File> TimKiemFile(string s):trả ra danh sách các file chứa chuỗi tìm kiếm.
* public string GuiTraLoiChoUser(List<File> s):gửi danh sách file cho user
* public string DangKy(string s,Socket soc):xử lí khi có yêu cầu đăng kí gửi đến server
* public string DangXuat(string s,Socket soc): xử lí khi có yêu cầu đăng xuất gửi đến server
* public string NhanFile(string s): xử lí khi có yêu cầu chia sẽ thêm file gửi đến server
* public string XoaFile(string s): xử lí khi có yêu cầu xóa file đã chia sẽ gửi đến server
* public string NhanXong(string s): xử lí khi có thông báo đã upload xong 1 file gửi đến server.

1. Client:
2. Lớp MyFile:

* Các thuộc tính:
* string m\_sTenFile
* string m\_sUserGiuFile;
* string m\_bTinhTrangDownLoad: tình trạng download cùa file này.
* string m\_bTinhTrangUser: tình trạng user chứa file này.
* Công dụng:
* Dùng chứa thông tin 1 file khi server gửi đến.

1. Lớp SocketServer:

* Các thuộc tính:
* public IPEndPoint ipeServer: chứa thông tin ip và port của server.
* public Socket sServer = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp): dùng dể kết nối đến server.
* Các hàm:
* public void taoKetNoiDenServer(string ipServer, int portServer): gửi yêu cầu tạo kết nối đến server với thông tin ip=ipServer và port = portServer.
* public string dangNhapUser(string User, string Pass,string port):dùng để gửi yêu cầu đăng nhập 1 user.
* public string taoUser(string User, string Pass1, string Pass2,string port): dùng để gửi yêu cầu tạo 1 user mới.
* public string themFile(string User, string TenFile): dùng để gửi yêu cầu chia sẻ thêm 1 file mới.
* public string xoaFile(string User, string TenFile): dùng để gửi yêu cầu xóa 1 file không cho chia sẻ.
* public string timFile(string User,string TenFile): gửi yêu cầu tìm kiếm 1 file với thông tin là tên file.
* public void yeuCauDownload(string User, string UserGiu, string TenFile): gửi yêu cầu download 1 file.
* public void DangXuat(string user): gửi yêu cầu đóng kết nối với server.
* Các hàm trên thực ra gửi 1 chuỗi với 1 cờ để phân biệt ở đầu chuỗi đến server với các cờ tương ứng sau:

1-user's name-password : là gửi yêu cầu đăng nhập

2-tên file-user's name yeu cau :là gửi yêu cầu tìm kiếm

3-user's name-password-port : là gửi yêu cầu đăng kí

4-user's name: la gửi yêu cầu đăng xuất

5-ten file-user's name: gửi yêu cầu thêm 1 file cho chia sẻ.

6-ten file-user's name: gui yeu cau xoa 1 file ko chia se nua

7-ten file-user's name: gửi trả lời đã cho down load xong file này

8-ten file-user's name chưá file-user name mún down load: gưỉ yêu câù down load

9-user's name-da online:thong bao xac nhan da online chua(chinh xac la de xac nhan da mo cong lang nghe chua)

* Sau đó, server sẽ gửi 1 chuỗi dể trả lời.Từ đó client sẽ xử lý các trường hợp tương ứng.
* Các hàm taoKetNoiDenServer, dangNhapUser, taoUser, được sử dung ở các form KetNoi, Dang Nhap , Dang Ky tương ứng.
* Form ChiaSeFile sẽ gọi các hàm themFile, xoaFile, timFile, yeuCauDownload với các nút button tương ứng.

Trong Form ChiaSeFile có khai báo 2 lớp:

1. Lớp MyUser:

* Các thuộc tính:
* string m\_sTenUser: tên user giữ file.
* IPAddress m\_IPAddress: dia chi ip cua user này.
* int m\_iPort: port lắng nghe của user này
* string m\_sFilename: ten file.
* Các hàm:
* public void LuuThongTin(string s): giống như hàm dựng các thuộc tính của đối tượng MyUser khi nhận 1 chuoi s từ server sẽ phân tích ra các giá trị của thuộc tính.
* Công dụng:
* dùng để lưu trữ các thông tin của user giữ file mà mình muốn download.

1. SocketClient:

* Trong lớp SocketClient có xây dựng lớp FileChiaSe để lư trữ thông tin 1 file mà user này chia sẻ: tên file và đường dẫn của file này.
* Các Thuộc Tính:
* Socket m\_socLangNghe: dùng để lắng nghe các yêu cầu từ các user khác.
* Socket[] m\_socLamViec: 1 mảng các socket phục vụ cho từng user khác.
* IPAddress ipServer: chứa ip của server.
* int m\_iSoClient: số user kết nối đến user này.
* int portServer: số port của Server
* int m\_iPort: port lắng nghe của user này.
* string MyName: tên user này
* List<FileChiaSe> m\_DanhSachFileChiaSe=layDanhSachFile(): danh sách các đối tượng FileChiaSe mà user cho phép chia sẻ và gán nó bằng với hàm đọc cơ sở dữ liệu layDanhSachFile() được định nghĩa ở trên.
* Các Hàm:
* public static OleDbConnection KetNoi(): kết nối với file CSDL.mdb chứa cơ sở dữ liệu của client.
* public static List<FileChiaSe> layDanhSachFile(): lấy danh sách các file mà user này cho chia sẻ được lưu trên cơ sở dữ liệu.
* public static bool themFileChiaSe(File FileNew): thêm thông tin 1 file mới vào cơ sở dữ liệu khi user này thêm chia sẻ file mới.
* public static bool xoaFileChiaSe(string User, string File): xóa thông tin của 1 file trên cơ sở dữ liệu khi user này không muốn chia sẻ.
* public void LayThongTin(IPAddress ipserver,int portsv,int myport,string name):gán giá trị cho thuộc tính ipServer, portServer, m\_iPort, MyName của đối tượng SocketClient.
* public void LangNghe(int port): tạo ra socket để lắng nghe các kết nối đến server.Sử dụng kiểu lắng nghe non-blocking(không đồng bộ).
* public void ClientKetNoi(IAsyncResult asyn): xử lí khi có kết nối đến server,sẽ tạo ra 1 thread và gọi hàm LamViec() để phục vụ cho client kết nối tới và gọi lại hàm lắng nghe cho server.
* public void LamViec():phục vụ cho client kết nối tới.
* public string PhanTichGoiTin(string s, Socket soc): phân tích thông tin từ client gửi đến thông qua các kí tự đầu tiên làm cờ:

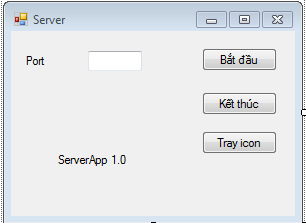
1-vui long doi : gửi thông báo đến cho user(tu server)

2-ten file-user's name-dia chi ip-port : gui file va thong tin cua user giu file duoc phep download(tu server)

3-ten file: nhan yeu cau down load tu user khac(tu user khac)

* public string DownLoadFile(Socket soc, string filename): gọi hàm này khi muốn download 1 file.
* public string UpLoadFile(string filename, Socket soc): gọi hàm này khi nhận dược yêu cầu downlaod file.
* public string LuuFile(string s): lưu thông tin của 1 file lên List m\_DanhSachFileChiaSe, và lư xuống cơ sở dữ liệu
* public bool XoaFile(string s): xóa thông tin của 1 file ra khỏi List m\_DanhSachFileChiaSe, và xóa nó trong cơ sở dữ liệu.

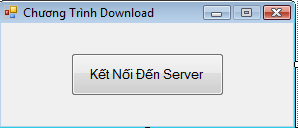
1. ***Thiết Kế Chương Trình:***
2. Server: chỉ có 1 form và có chức năng tray icon



* Button Bắt đầu :dùng để bắt dầu lắng nghe với port nhập ở textbox.
* Button kết thúc:dùng để kết thúc chương trình.Đóng socket lắng nghe.
* Button Tray icon:dùng để đưa chương trình xuống thanh task bar

1. Client:

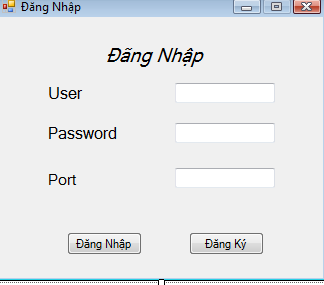
* Chương trình gồm 5 form:
* Form được chạy khi chạy file.exe là form: FormMain.



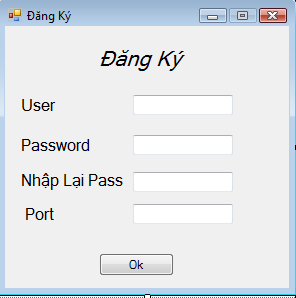
* Trong FormMain, Bấm buttton “Kết Nối Đến Server” sẽ hiện lên FormKetNoi.



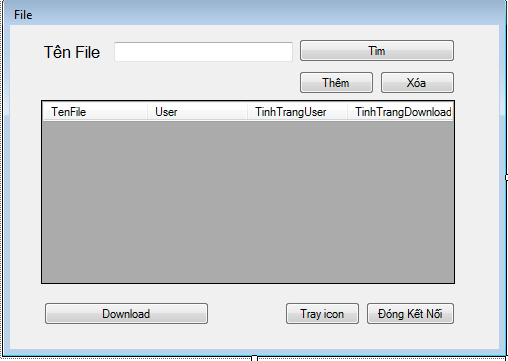
* Trong Form này, ta sẽ điền ip và port của server mà mình muốn kết nối. bấm button “Kết nối” nếu server này đang online thì sẽ hiện ra FormDangNhap.



* Trong FormDanngNhap, nếu chưa có tài khoản thì bấm buttton “Đăng Ký”.



* nếu có rồi thì ta điền User, Password, port và bắm button “Đăng Nhập”. Nếu thành công, thì sẽ hiện lên FormChiaSe.



* Trong FromChiaSe, ta có textbox điền tên file mà mình muồn tìm trên server, khi ta bắm button “Tìm” thì danh sách các file thỏa yêu cầu tìm sẽ hiện lên
* Bảng Data Gird View .từ đó, ta chọn 1 dòng thông tin file trong đó để thực hiện việc download với button “Download”.
* Bắm button “Them” ,ta sẽ chọn 1 file cho phép chia sẽ với các user khác. Sau đó, sẽ gửi yêu cầu thêm file lên server.
* Bấm button “Xoa”, ta sẽ xóa file với thông tin tên file lấy ở txtboxTenFile để gủi yêu cầu xóa 1 file lên server.
* Button” tray icon”: sẽ tạo 1 tray icon ở thanh task và đóng cửa sổ hiện hành.
* Button” Đóng Kết Nối”: sẽ gửi yêu cầu đăng xuất lên server.

1. ***Kịch Bản Trao Đổi Dữ Liệu:***

* Khi 1 user connect vào server thì sẽ gửi user’s name và pass tới server để chứng thực (với cờ là 1).Server sẽ kiểm tra user name trước :
* nếu chưa có trong danh sách user thì bắt user phải đăng kí.Gửi lại thông báo “nhap sai user hoac user chua dang ky”
* nếu có thì kiểm tra pass để chứng thực cho user :

nếu nhập sai pass sẽ hiện thông báo : nhap sai password

nếu nhập sai port sẽ hiện thông báo : nhap sai port da dang ki

nếu đăng nhập trên máy khác,không giống địa chỉ IP lúc đăng kí thì sẽ hiện thông báo :” Khong dung ip luc dang ky.Moi ban dang ky lai hoac chuyen may.”

* Bước tiếp theo:

Sau đó user sẽ gửi 1 thông báo đến cho Server để thông báo mình đã online và thực hiện lắng nghe kết nối. Server sẽ cấp phát thêm 1 Thread để kiểm tra hang đợi download những file của user vừa đăng nhập

* Nếu lần đầu kết nối thì không có file nào mà user này chia sẽ,user sẽ gửi danh sách file có thể chia sẻ trên máy tính của mình cho server (với cờ 5).Server xác nhận sẽ gửi lại thông báo “Nhan file thanh cong” hoặc sẽ gửi lại thông báo "file da duoc ban chia se roi hoac trung ten"nếu file đã có trên server.
* Ngược lại thì Server sẽ gửi tiếp những yêu cầu download đã yêu cầu trong lúc user này offline và thực hiện download/upload các file này.Và user có thể gửi thêm danh sách file mới có thể chia sẻ cũng như danh sách các file trước đó và giờ không cho chia sẻ nữa.Server xác nhận việc xóa file chia sẽ sẽ gửi lại thông báo "Da Xoa" hoặc "khong tim thay file can xoa" nếu không tìm thấy file cần xóa. Server sẽ cập nhật bảng danh sách file của mình.
* Quá trình upload và download:

User giữ file khi nhận được yêu cầu download từ user khác thì sẽ kiểm tra xem file đ1o còn tồn tại hay không và thông báo cho user muốn down.

* Nếu user gửi yêu cầu tìm kiếm tên 1 file lên server. Server sẽ tra trong bảng danh sách file xem có tên file đó hay không:
* Nếu có,gửi lại danh sách các file thỏa tên đó cho client cùng với các thông tin: tên file, user chứa file, user có online hay không, tình trạng download của file đó.
* Nếu không, thì server sẽ gửi thông báo cho user đó là file đó không tồn tại.
* Nếu có, client đó chọn 1 file trong danh sách đó để download thì server sẽ kiểm tra xem:
* Nếu user đó online và hiện không có user nào download file đó thì sẽ chấp nhận việc download và cập nhật cột tình trạng download của file đó là có.Sau đó Server sẽ gửi thông tin của user đang giữ file đó đến cho user muốn download.User muốn down sẽ kết nối đến user đang giữ file thông qua port lắng nghe của user này và gửi thông báo về việc muốn download 1 file.User giữ file sẽ nhận kết nối và kiểm tra file đó còn tồn tại hay không,nếu còn 2 user bắt đầu thực hiện truyền file,nếu không còn tồn tại thì user muốn download sẽ nhận được thông báo “file khong ton tai hoac da bi xoa”.Quá trinh truyền file:
* User muốn down sẽ gọi hàm Receive để chờ user giữ file gửi dung lượng của file đó.Nếu nhận được thì trả về thông báo cho user giữ file để bắt đầu việc truyền file.Sau đó tạo 1 buffer byte[1024] để nhận từng KB dữ liệu từ user giữ file
* User giữ file đầu tiên sẽ gửi dung lượng file đó qua cho user muốn downn và chờ nhận được thông báo bắt đầu truyền file.Sau khi nhận được thông báo thì tạo ra 1 buffer là byte[1024] để truyền từng KB của file.
* Ngược lại, nếu 1 trong 2 trường hợp này không chấp nhận việc download: user đó đang offline, file đó đang có user khác đang download thì server sẽ đưa tên của user muốn download vào hang đợi của file đó.

* Việc gửi hay nhận 2 file là hoàn toàn tách biệt vì mỗi khi gửi sẽ cấp 1 thread riêng nên có thể cho phép nhiều user download nhiều file cùng 1 lúc ở cùng 1 user.
* Nếu trong quá trình download xảy ra sự cố thì việc download sẽ hủy và sẽ xử lý như trường hợp user yêu cầu download offline.
* Khi việc download thực hiện xong:
  + user chứa file đó sẽ gửi thông báo hoàn thành việc download đến server.
* Và lúc đó, server sẽ kiểm tra xem hang đợi của file đó còn chứa user nào không?Nếu có thì lấy user tiếp theo trong hàng đợi ra và thưc hiện như trên. Còn nếu không thì server sẽ cập nhật lại cột tình trạng download của file đó là không.
* Còn user yêu cầu download xong:
* Nếu có yêu cầu download file thì thực hiện các bước download như trên.
* Nếu không, thì user đó có thể offline. Khi đó server sẽ cập nhật trình trạng của user đó trong bảng danh sách file.

1. ***Đa tiểu trình,đồng bộ hóa và lien lạc giữa các tiểu trình:***

**ở Server:**

* Đa tiểu trình:

Form sẽ giữ thread chính và gọi hàm LangNghe() của server.

mỗi client kết nối tới sẽ được cấp phát 1 thread để phục vụ.Thread này gọi hàm LamViec().Các thread này sử dụng biến dùng chung là List<File> m\_arrDanhSachFile, List<User> m\_arrDanhSachUser; dùng chung các hàm truy xuất cơ sở dữ liệu và 1 biến Mutex để đồng bộ hóa (Mutex ms)

Hàm LamViec() sẽ phân tích các thông tin mà client gửi tới và xử lí chúng.Nếu là đăng nhập thì server sẽ cấp 1 thread khác để kiếm tra xem các file của client đó có được ai đợi để download không ,nếu có thì thực hiện việc download đó lun.(Download offline)

* Đồng bộ hóa:

Việc truy xuất tài nguyên dùng chung có thể xảy ra đụng độ xuất hiện ở các hàm sau :

public string DangNhap(string s,Socket soc)

public List<File> TimKiemFile(string s)

public string DangXuat(string s,Socket soc)

public string XoaFile(string s)

public string NhanXong(string s)

ở những hàm này thì vòng for để tìm kiếm và truy xuất tài nguyên dùng chung (là bảng danh sách file) sẽ được bảo vệ bởi biến Mutex để tránh các trường hợp đụng độ giữa các thread.

Trong quá trình Download offline, Server gửi thông tin cho user muốn down thì sẽ chạy đồng thời việc User này gửi yêu cầu cho User giữ file và việc User giữ file thực hiện việc lắng nghe các kết nối.

Nếu user giữ file chưa kịp thực hiện lắng nghe kết nối mà User muốn down đã gửi yêu cầu thì yêu cầu đó không tới được User giữ file.Vì thế việc User giữ thực hiện lắng nghe phải được thực hiện trước khi user muốn down gửi yêu cầu.

Thực hiện:

Server sẽ chờ 1 thông báo từ User giữ file là đã online(đã thực hiện lắng nghe) rồi mới gửi thông tin về User đó cho user muốn download.Vì thế đảm bảo việc User giữ file thực hiện lắng nghe gòi thì User muốn down mới gửi thông tin

* Liên lạc giữa các tiểu trình:

Mỗi thread được cấp để phục vụ cho 1 client kết nối tới sẽ giống như 1 bản sao của server,hoạt động khá độc lập với nhau,chỉ truy xuất biến dùng chung là bảng danh sách file.

Khi mà có yêu cầu đăng nhập,thì thread được cấp để phục vụ kết nối đó sẽ phát sinh ra 1 thread nữa phục vụ cho việc kiểm tra hàng đợi của các file của user vừa đăng nhập,phục vụ cho việc Download offline.Thread này sẽ được thread cha truyền qua 1 biến string là tên của user vừa đăng nhập để kiếm tra các file của user này.

**Ở Client:**

* Đa tiểu trình :

FormChiaSeFile giữ thread chính ,sẽ gọi hàm LangNghe của lớp Socketclient để lắng nghe các kết nối đến.

Khi có kết nối đến thì sẽ cấp 1 thread để phục vụ cho kết nối đó.Thread này sẽ gọi hàm LamViec()

* Đồng bộ hóa :

Không cần đồng bộ

* Liên lạc giữa các tiểu trình.

Khi cần xuất thông báo ra màn hình thì thread con sẽ lien lạc với thread chính giữ form và xuất ra màn hình thông báo

Giữa các thread con thì hoạt đông độc lập

1. ***Các TestCase:***

* **Trường hợp 1:**

Khi 1 user gửi yêu cầu tìm kiếm file thì server sẽ truy xuất bảng danh sách file(chạy vòng for để tìm kiếm)và ngay lúc đó thì user đang giữ file thỏa yêu cầu tìm kiếm vừa đăng nhập hoặc đăng xuất (server cũng truy xuất vào bảng danh sách file và chạy vòng for để thay đổi thuộc tính tình trạng user).

**Ví dụ cụ thể:**

User1 muốn down file có chứa chuỗi “sinh”.

User2 đang giữ 2 file là “sinhhoc.txt” và “sinhviên.txt” và đang online

**Không đồng bộ** :

Trong khi user1 muốn down chạy vòng for để tìm kiếm file (ví dụ file có chứa chuỗi “sinh”) và tìm được file “sinhhoc.txt” với tình trạng của user2 là online thì bị ngắt CPU và trả về cho vòng for chạy để thay đổi thuộc tính tình trạng user của file đó (ví dụ ở đây là chuyển về offline) .sau khi chạy xong lại trả CPU về cho user1 để tiếp tục tìm kiếm và sẽ tìm thấy file “sinhvien.txt” với tình trạng của user2 là offline sẽ dẫn đến kết quả tìm kiếm trả về cho user đó sẽ là 2 file “sinhhoc.txt” với tình trạng user2 là online và file “sinhvien.txt” với tình trạng user2 là offline.

**Có đồng bộ:**

Trong khi user1 muốn down chạy vòng for để tìm kiếm (truy xuất vô miền găng là bảng danh sách file) thì sẽ được biến mutex bảo vệ.Hoặc user2 đăng nhập hay đăng xuất thì cũng sẽ truy xuất bảng danh sách file(miền găng) để thay đổi thuộc tính cũng sẽ được biến mutex bảo vệ.Vì vậy kết quả trả ra sẽ đúng.

* **Trường hợp 2**

Khi 1 user gửi yêu cầu tìm kiếm file thì server sẽ truy xuất bảng danh sách file(chạy vòng for để tìm kiếm)và ngay lúc đó thì user đang giữ file thỏa yêu cầu tìm kiếm xóa hoặc thêm file thỏa yêu cầu (chạy vòng for để tìm file cần xóa)

**Ví dụ cụ thể:**

User1 muốn down file có chứa chuỗi “sinh”.

User2 đang giữ 2 file là “sinhhoc.txt” và “sinhviên.txt” và đang online

**Không đồng bộ:**

Trong lúc user2 xóa file sinhhoc.txt (đang chạy vòng for để tìm kiếm file đó)nhưng chưa kịp xóa thì đã bị ngắt CPU để cho user1 tìm kiếm với chuỗi “sinh” thì tìm được file “sinhhoc.txt”.Sau đó CPU lại được trả về cho user2 để tìm kiếm và xóa file.Vì thế kết quả trả về vẫn có file “sinhhoc.txt” trong khi file đã bị xóa trên server.

**Đồng bộ:**

Lúc user2 truy xuất miền găng để xóa file “sinhhoc.txt” thì được bảo vệ đến khi xóa xong.Vì thế khi user1 tiến hành tìm kiếm thì sẽ không tìm thấy file “sinhhoc.txt”.

* **Trường hợp 3:**

Khi 1 user gửi thông báo đã download xong 1 file cho server và server sẽ truy xuất bàng danh sách file để tìm kiếm file đó cập nhập lại thuộc tính “tình trạng download” cùng lúc có user khác tìm kiếm file đó

**Ví dụ cụ thể:**

User1 gửi thông báo đã upload xong file “a.txt”

User2 tìm kiếm file với chuỗi “t”

**Không đồng bộ:**

Trong khi server đang tìm kiếm file a.txt để cập nhập lại thuộc tính “tình trang download” thì bị ngắt CPU.Cu được chuyển qua phục vụ user2 tìm kiếm file với chuỗi “t” và đã tìm thấy file “a.txt” ở tình trạng đang download.Vì thế kết quả trả về sẽ bị sai.

**Có đồng bộ:**

Khi server truy xuất bảng danh sách file để tìm kiếm file “a.txt” thì sẽ được bảo vệ đến khi làm xong việc cập nhập lại thuộc tính cho file “a.txt”.Sau đó user2 tìm kiếm sẽ trả về kết quả đúng

1. ***Phân Công:***

* Trần Nguyễn Hoàng Thông,MSSV:0812502 :
* Về code:
* **Ở Client:**

Viết class SocketClient,class Myuser,thực hiện đa tiểu trình ,….

* **Ở Server:**

Viết toàn bộ trong Server trừ phần truy xuất cơ sở dữ liệu

* Về báo cáo:
* Viết mô hình cài đặt và thiết kế chương trình của Server
* Viết kịch bản trao đổi dữ liệu,đa tiến trình,đồng bộ hóa,liên lạc giữa các tiểu trình của cả Server và Client
* Viết các testcase

* Trần Hưng Thuận,MSSV:0812508 :
* Về code:
* Thiết kế cơ sở dữ liệu , viết các hàm thao tác với cơ sở dữ liệu cho server và client.
* Thiết kế chương trình cho client.
* Viết các lớp MyFile và lớp SocketServer của client: tạo kết nối đến server.
* Về báo cáo:
* Viết mô hình cài đặt và thiết kế chương trình của Client