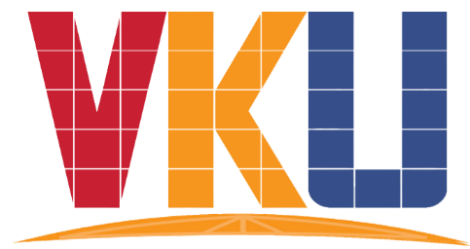
****

**ĐẠI HỌC ĐÀ NẴNG**

**ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**VIỆT HÀN**



**ĐỒ ÁN CƠ SỞ 4**

**ĐỀ TÀI: PHÁT TRIỂN DỊCH VỤ CLIENT-SERVER BẰNG GIAO THỨC FTP**

**GVHD: TS.NGUYỄN SĨ THÌN**

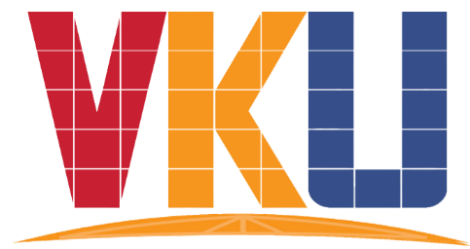
**SINH VIÊN: DƯƠNG THỊ MỸ LINH**

**LỚP:** **18IT5**

*Đà Nẵng, tháng năm 2020*

**ĐẠI HỌC ĐÀ NẴNG**

**ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT HÀN**



**ĐỒ ÁN CƠ SỞ 4**

**ĐỀ TÀI: PHÁT TRIỂN DỊCH VỤ CLIENT-SERVER BẰNG GIAO THỨC FTP**

**GVHD:** **TS. NGUYỄN SĨ THÌN**

**SINH VIÊN: DƯƠNG THỊ MỸ LINH**

**LỚP: 18IT5**

*Đà Nẵng, tháng năm 2020*

**NHẬN XÉT**

**(**của giảng viên hướng dẫn**)**

**Giảng viên hướng dẫn**

(ký tên)

**TS. NGUYỄN SĨ THÌN**

[**LỜI MỞ ĐẦU** 4](#_Toc60831735)

[**CHƯƠNG I: TỔNG QUAN VỀ GIAO THỨC FTP** 5](#_Toc60831736)

[**1.1. Giới thiệu về giao thức FTP.** 5](#_Toc60831737)

[**1.2. Mục tiêu đề tài.** 6](#_Toc60831738)

[**CHƯƠNG II: CƠ SỞ LÝ THUYẾT** 7](#_Toc60831739)

[**2.1. Giới thiệu mô hình Client – Server**. 7](#_Toc60831740)

[**2.2. Mô hình cơ bản trong FTP** 7](#_Toc60831741)

[**2.2.1 Kết nối TCP trong FTP** 7](#_Toc60831742)

[**2.2.2. Mô hình FTP** 8](#_Toc60831743)

[**2.3. Chức năng từng phần trong mô hình FTP** 9](#_Toc60831744)

[**2.4. Nguyên lí hoạt động của FTP** 10](#_Toc60831745)

[**2.5. Các phương thức truyền dữ liệu trong FTP** 11](#_Toc60831746)

[**CHƯƠNG III: KẾT QUẢ THỰC HIỆN** 13](#_Toc60831747)

[**CHƯƠNG IV: KẾT QUẢ ĐẠT ĐƯỢC** 14](#_Toc60831748)

[**4.1. Ưu điểm** 14](#_Toc60831749)

[**4.2. Nhược điểm** 14](#_Toc60831750)

[**4.3. Hướng phát triển** 14](#_Toc60831751)

# **LỜI MỞ ĐẦU**

# Internet đóng vai trò rất lớn trong việc phát triển kinh tế xã hội, là nhân tố thúc đẩy xã hội ngày càng phát triển. Không những phục vụ cho nhu cầu giải trí, giao lưu văn hóa, internet dạng là công cụ chính trong tất cả các lĩnh vực của đời sống xã hội, từ kinh doanh đến quản lý từ dịch vụ cho đến hệ thống.

# Các tổ chức, công ty hay các cơ quan cần phải xây dựng hệ thống mạng máy tính cho riêng mình để trao đổi dữ liệu giữa các bộ phận. Dữ liệu được truyền đi trên mạng phải đảm bảo: dữ liệu được chuyển tới đích nhanh chóng và đúng đắn. Hầu hết dữ liệu được truyền qua mạng là truyền dưới dạng file.Từ những gói tin dữ liệu được gởi đi và về, về dung lượng và tốc độ. Nên các giao thức, các mô hình được phát triển phù hợp. Bên cạnh đó, vấn đề trao đổi file là không thể thiếu đối với người dùng đầu cuối. Và với giao thức FTP (giao thức truyền tập tin – thường được dùng để trao đổi tập tin qua mạng lưới truyền thông dùng giao thức TCP/IP đang làm nền tảng cho sự ổn định, phát triển để dáp ứng nhu cầu của người dùng.

Trên cơ sở kiến thức đã học ở trường em đã tìm hiểu, nghiên cứu, phát triển dịch vụ Client-Server bằng giao thức FTP giúp chúng ta hình dung rõ nhất về giao thức trên. Thông qua đề tài em muốn học hỏi các kĩ năng cũng như trao dồi kiến thức lập trình.

Em xin chân thành cảm ơn!

# **CHƯƠNG I: TỔNG QUAN VỀ GIAO THỨC FTP**

## **1.1. Giới thiệu về giao thức FTP.**

FTP (viết tắt của File Transfer Protocol dịch ra là "Giao thức truyền tập tin") thường được dùng để trao đổi tập tin qua mạng lưới truyền thông dùng giao thức TCP/IP. Hoạt động của FTP cần có hai máy tính, một máy chủ (server) và một máy khách (Client). Máy chủ FTP, dùng chạy phần mềm cung cấp dịch vụ FTP, gọi là trình chủ, lắng nghe yêu cầu về dịch vụ của các máy tính khác trên mạng lưới. Máy khách chạy phần mềm FTP dành cho người sử dụng dịch vụ, gọi là trình khách, thì khởi đầu một liên kết với máy chủ. Một khi hai máy đã liên kết với nhau, máy khách có thể xử lý một số thao tác về tập tin, như tải tập tin lên máy chủ, tải tập tin từ máy chủ xuống máy của mình, đổi tên của tập tin, hoặc xóa tập tin ở máy chủ v.v. Vì giao thức FTP là một giao thức chuẩn công khai, cho nên bất cứ một công ty phần mềm nào, hay một lập trình viên nào cũng có thể viết trình chủ FTP hoặc trình khách FTP.

Hầu như bất cứ một nền tảng hệ điều hành máy tính nào cũng hỗ trợ giao thức FTP. Điều này cho phép tất cả các máy tính kết nối với một mạng lưới có nền TCP/IP, xử lý tập tin trên một máy tính khác trên cùng một mạng lưới với mình, bất kể máy tính ấy dùng hệ điều hành nào (nếu các máy tính ấy đều cho phép sự truy cập của các máy tính khác, dùng giao thức FTP. FTP thường chạy trên hai cổng 20 và 21, và chỉ chạy riêng trên nền của TCP. Trình chủ FTP lắng nghe các yêu cầu dịch vụ từ những kết nối vào máy của các trình khách FTP, trên cổng 21. Đường kết nối trên cổng 21 này tạo nên một dòng truyền điều khiển, cho phép các dòng lệnh được chuyển qua trình chủ FTP. Để truyền tải tập tin qua lại giữa hai máy, chúng ta cần phải có một kết nối khác. Tùy thuộc vào chế độ truyền tải được sử dụng, trình khách (ở chế độ năng động - active mode) hoặc trình chủ (ở chế độ bị động - passive mode) đều có thể lắng nghe yêu cầu kết nối đến từ đầu kia của mình.

Trong trường hợp kết nối ở chế độ năng động, (trình chủ kết nối với trình khách để truyền tải dữ liệu), trình chủ phải trước tiên đóng kết vào cổng 20, trước khi liên lạc và kết nối với trình khách. Trong chế độ bị động, hạn chế này được giải tỏa và việc đóng kết trước là một việc không cần phải làm. Trong khi dữ liệu được truyền tải qua dòng dữ liệu, dòng điều khiển đứng im. Tình trạng này gây ra một số vấn đề, đặc biệt khi số lượng dữ liệu đòi hỏi được truyền tải là một số lượng lớn, và đường truyền tải chạy thông qua những bức tường lửa. Bức tường lửa là dụng cụ thường tự động ngắt các phiên giao dịch sau một thời gian dài im lặng. Tuy tập tin có thể được truyền tải qua hoàn thiện, song dòng điều khiển do bị bức tường lửa ngắt mạch truyền thông giữa quãng, gây ra báo lỗi.

## **1.2. Mục đích của giao thức FTP**

Mục đích của giao thức FTP, như được phác thảo trong bản RFC, là:

1. Khuyến khích việc dùng chung tập tin (như chương trình ứng dụng hoặc dữ liệu)
2. Khuyến khích việc sử dụng máy tính ở xa một cách gián tiếp / âm thầm (implicit).
3. Che đậy sự khác biệt về hệ thống lưu trữ tập tin giữa các máy chủ, hầu cho người dùng không cần phải quan tâm đến những sự khác biệt riêng tư của chúng.
4. Truyền tải dữ liệu một cách đáng tin cậy và có hiệu quả cao.

## **1.3. Mục tiêu đề tài.**

Xây dựng cách thức hoạt động của giao thức FTP. Một giao thức trao đổi file khá phổ biến hiện nay. Mô phỏng cách thức hoạt động của giao thức FTP bằng chương trình được viết bởi Java. Trong phần mô phỏng, ứng dụng FTP sẽ có những chức năng cơ bản như: đăng nhập User, download file từ server về máy client, upload file từ client về server, xóa file trong folder Client hoặc xóa file trong folder Server

**FTP có 3 phương thức truyền dữ liệu**

1. Stream Mode: dữ liệu được truyền đi vơi dạng các byte có cấu trúc không liên tiếp
2. Block mode: chia dữ liệu thành các gói nhỏ và đóng gói thành các FTP block

Mỗi block chứ thông tin về khối dữ liệu được gởi đi.

1. Compressed Mode: sử dụng phương thức run-length encoding giúp phát hiện và loại bỏ các gói dữ liệu bị trùng lặp, giảm chiều dài của toàn bộ thông điệp cần gởi đi

# **CHƯƠNG II: CƠ SỞ LÝ THUYẾT**

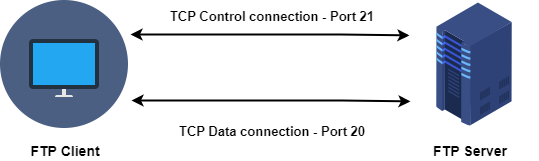
## **2.1. Giới thiệu mô hình Client – Server**.

Các ứng dụng mạng thường hoạt động theo mô hình client - server như Email, gửi nhận tập tin, game online, ... Mô hình này gồm có một chương trình đóng vai trò là client và một chương trình đóng vai trò là server. Hai chương trình này sẽ giao tiếp với nhau thông qua mạng. Chương trình server đóng vai trò cung cấp dịch vụ. Chương trình này luôn luôn lắng nghe các yêu cầu từ phía client, rồi tính toán và đáp trả kết quả tương ứng. Chương trình client cần một dịch vụ và gởi yêu cầu dịch vụ tới chương trình server và đợi đáp trả từ server.

Như vậy, quá trình trao đổi dữ liệu giữa client - server bao gồm: Truyền một yêu cầu từ chương trình client tới chương trình server. Yêu cầu được server xử lý truyền đáp ứng cho client

## **2.2. Mô hình cơ bản trong FTP**

### **2.2.1 Kết nối TCP trong FTP**



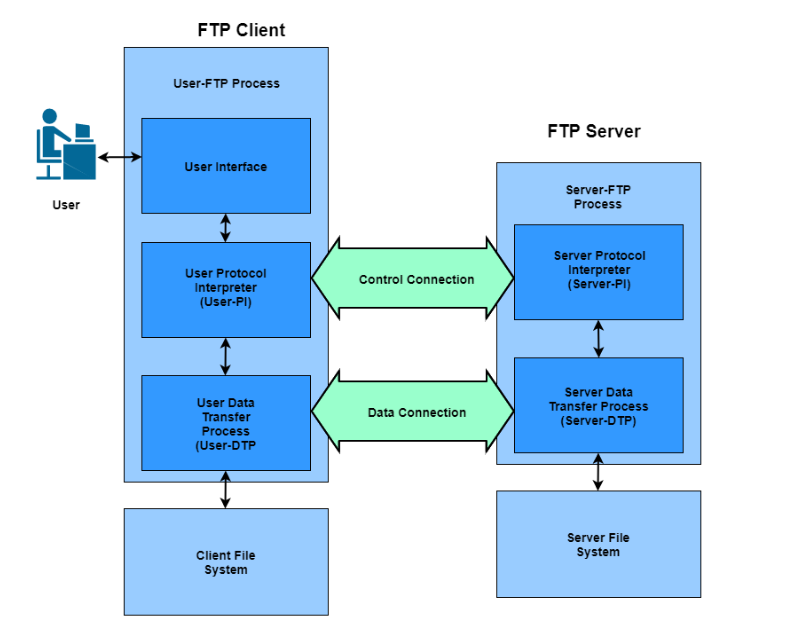
**Hình 1: kết nối TCP trong FTP**

Giống như hầu hết các giao thức TCP/IP, FTP dựa trên mô hình Client – Server. Tuy nhiên, khác với các ứng dụng khác chạy trên nền TCP/IP, FTP cần tới 2 kết nối TCP:

* Control connection (sử dụng port 21 – trên server): Đây là kết nối TCP logic chính được tạo ra khi phiên làm việc được thiết lập. Nó được thực hiện giữa các quá trình điều khiển. Nó được duy trì trong suốt phiên làm việc và chỉ cho các thông tin điều khiển đi qua như lệnh hay response(phản hồi)
* Data connection (sử dụng port 20 – trên server): Kết nối này sử dụng các quy tắc rất phức tạp vì các loại dữ liệu có thể khác nhau. Nó được thực hiện giữa các quá trình truyền dữ liệu. Kết nối này mở khi có lệnh chuyển tệp và đóng khi tệp truyền xong.

### **2.2.2. Mô hình FTP**

Sơ đồ minh họa:



**Hình 2: Mô hình FTP**

Do chức năng điều khiển và dữ liệu được truyền tải bằng cách sử dụng các kênh riêng biệt nên mô hình FTP chia mỗi thiết bị thành 2 phần giao thức logic chịu trách nhiệm cho mỗi kết nối ở trên:

* **Protocol interpreter (PI):** Là thành phần quản lý kênh điều khiển, phát và nhận lệnh và trả lời.
* **Data transfer process (DTP**)**:** chịu trách nhiệm gửi và nhận dữ liệu giữa client và server.

**2.3. Chức năng từng phần trong mô hình FTP**

* **Phía Server**
* **Server Protocol Interpreter (Server-PI)**: Chịu trách nhiệm quản lí Control Connection trên Server. Nó lắng nghe yêu cầu kết nối hướng từ User trên cổng 21. Khi kết nối được thiết lập, nó nhận lệnh từ User-PI, gửi phản hồi và quản lí tiến trình truyền dữ liệu trên Server.
* **Server Data Transfer Process (Server-DTP)**: chịu trách nhiệm nhận và gửi file từ User-DTP. Server-DTP vừa làm nhiệm vụ thiết lập Data Connection và lắng nghe Data Connection của User thông qua cổng 20. Nó tương tác với Server File System trên hệ thống cục bộ để đọc và chép file.
* **Phía Client**
* **User Interface**: Đây là chương trình được chạy trên máy tính, nó cung cấp giao diện xử lí cho người dùng, chỉ có trên phía Client. Nó cho phép người dùng sử dụng những lệnh đơn giản để điều khiển các session FTP, từ đó có thể theo dõi được các thông tin và kết quả xảy ra trong quá trình.
* **User Protocol Interpreter (User-PI)**: Chịu trách nhiệm quản lí Control Connection phía Client. Nó khởi tạo phiên kết nối FTP bằng việc phát hiện ra Request tới Server-PI. Sau khi kết nối được thiết lập, nó xử lí các lệnh nhận được trên User Interface, gửi chúng tới Server-PI rồi đợi nhận Response trở lại. Nó cũng quản lí các tiến trình trên Client.
* **User Data Transfer Process (User-DTP)**: Có nhiệm vụ gửi hoặc nhận dữ liệu từ Server-DTP. User-DTP có thể thiết lập hoặc lắng nghe DataConnection từ Server thông qua cổng 20. Nó tương tác với Client File System trên Client để lưu trữ file.

## **2.4. Nguyên lí hoạt động của FTP**

Cần có 2 kết nối TCP trong phiên làm việc của FTP: TCP Data connection trên cổng 20, TCP Control connection trên cổng 21.

* Control connection: luôn được mở ở mọi thời điểm khi dữ liệu hoặc lệnh được gửi.
* Data connection: chỉ được mở khi có trao đổi dữ liệu thực.

**Trình tự chung của FTP hoạt động như sau:**

1. FTP Client mở Control connection đến FTP server (trên port 21) và chỉ định 1 cổng trên Client để Server gửi lại phản hồi. Đường kết nối này dùng để truyền lệnh và không phải là dữ liệu. Control connection sẽ mở trong suốt thời gian của phiên làm việc (telnet giữa 2 hệ thống)
2. Client chuyển tiếp thông tin như username, password tới Server để thực hiện xác thực (authentication). Server sẽ trả lời bằng mã chấp nhận hay từ chối của các request.
3. Client gửi thêm các lệnh với tên tệp, kiểu dữ liệu,… để vận chuyển, thêm luồng dữ liệu(tức là chuyển tập tin từ máy khách đến máy chủ hoặc ngược lại). Server sẽ phản hồi với mã (reply code) chấp nhận hoặc từ chối.
4. Khi dữ liệu đã sẵn sàng, 2 bên sẽ mở kết nối TCP trên cổng 20.
5. Dữ liệu có thể được vận chuyển giữa Client và Server trên cổng 20. Dữ liệu vận chuyển được mã hóa theo 1 số định dạng bao gồm NVT-ASCII hoặc nhị phân(binary)
6. Khi quá trình vận chuyển dữ liệu được hoàn thành, phiên làm việc của FTP Server sẽ đóng lại Data Connection trên cổng 20. Nhưng vẫn giữ Control Connection trên công 21.
7. Control connection có thể được sử dụng để thiết lập truyền dữ liệu khác hoặc đóng liên kết.

## **2.5. Các phương thức truyền dữ liệu trong FTP**

Khi Client-DTP và Server-DTP thiết lập xong kênh dữ liệu, dữ liệu sẽ được truyền trực tiếp từ phía Client tới phía Server, hoặc ngược lại, tùy theo các lệnh được sử dụng. Do thông tin điều khiển được gửi đi trên kênh điều khiển, nên toàn bộ kênh dữ liệu có thể được sử dụng để truyền dữ liệu. FTP có ba phương thức truyền dữ liệu, đó là: stream mode, block mode, và compressed mode.

**1. Stream mode**

Dữ liệu truyền đi liên tiếp dưới dạng các byte không cấu trúc.

Thiết bị gửi chỉ đơn thuần đẩy luồng dữ liệu qua kết nối TCP tới phía nhận.

Không có trường tiêu đề nhất định

Không có cấu trúc dạng Header, nên việc báo hiệu kết thúc file sẽ đơn giản được thực hiện khi thiết bị gửi ngắt kênh kết nối dữ liệu khi đã truyền dữ liệu xong.

Được sử dụng nhiều nhất trong 3 phương thức trong triển khai FTP thực tế. Do

Là phương thức mặc định và đơn giản nhất.

Là phương thức phổ biến nhất, vì nó xử lí các file chỉ đơn thuần là xử lí dòng byte, mà không cần để ý tới nội dung.

Không tốn 1 lượng byte “overload” nào để thông báo Header.

**2. Block mode**

Phương thức truyền dữ liệu mang tính quy chuẩn hơn.

Dữ liệu được chia thành nhiều khối nhỏ và đóng gói thành các FTP block.

Mỗi block có 1 trường Header 3 byte: báo hiệu độ dài, và chứa thông tin về các khối dữ liệu đang được gửi.

Một thuật toán đặc biệt được sử dụng để kiểm tra các dữ liệu đã truyền đi. Và để phát hiện, khởi tạo lại đối với 1 phiên truyền dữ liệu đã bị ngắt kết nối.

**3. Compressed mode (Chế độ nén)**

Phương thức truyền dữ liệu sử dụng 1 kỹ thuật nén đơn giản, là “run-lenght encoding (mã hóa chiều dài)” – có tác dụng phát hiện và xử lí các đoạn lặp trong dữ liệu được truyền đi để giảm chiều dài của toàn bộ thông điệp.

Thông tin sau khi được nén, sẽ được xử lí như Block mode, với trường Header.

Trong thực tế, việc nén dữ liệu thường được thực hiện ở chỗ khác, làm cho phương thức Compressed mode trở nên không cần thiết.

# **CHƯƠNG III: KẾT QUẢ THỰC HIỆN**

# **CHƯƠNG IV: KẾT QUẢ ĐẠT ĐƯỢC**

## **4.1. Ưu điểm**

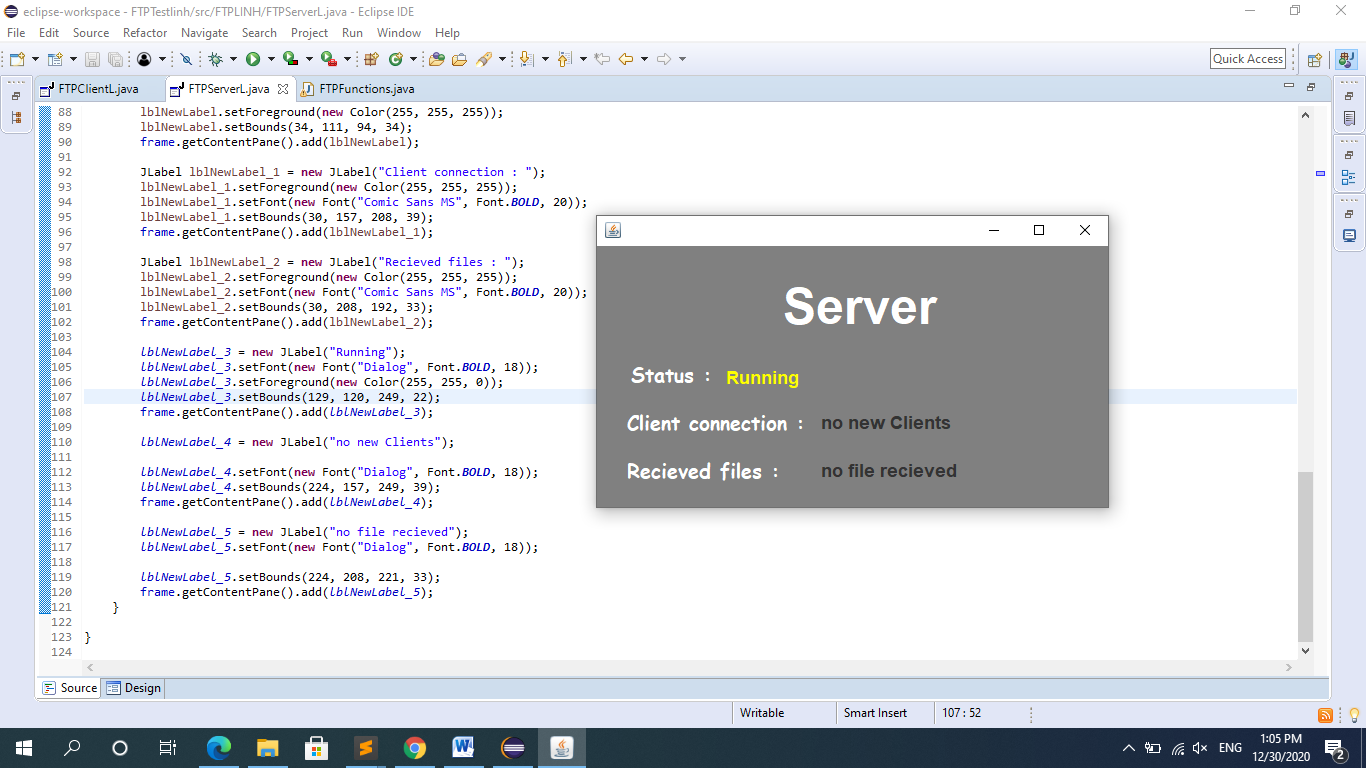
* Giao diện dễ nhìn
* Làm dược chức năng cơ bản
* Học hỏi và nâng cao tinh thần nghiên cứu

## **4.2. Nhược điểm**

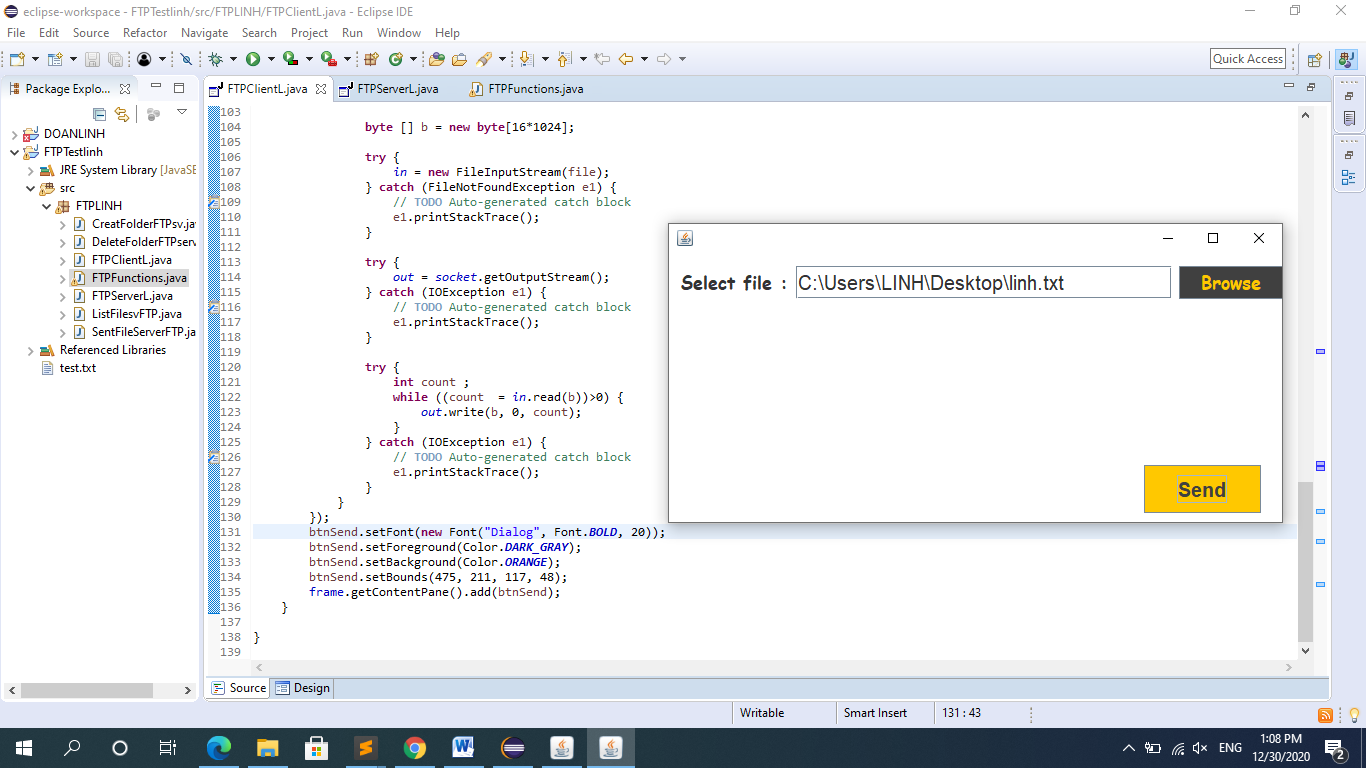
* Do đề tài phong phú và đa dạng, nên trong quá trình phân tích và thiết kế không tránh khỏi các sai sót và hạn chế nhất định.
* Mô hình còn một số chức năng chưa hoàn thiện và chính xác.
* Đòi hỏi máy tính mạnh và ổn định.
* Gặp nhiều lỗi trong quá trình làm ứng dụng.

## **4.3. Hướng phát triển**

* Xử lý các lỗi chặt chẽ hơn trước khi đưa ra sử dụng.
* Mở rộng bài toán.
* Linh hoạt hơn trong sử lý thông tin và truyền File.
* Bổ sung thêm nhiều chức năng mới.
* Phát triển thêm các tính năng chưa hoàn thiện.



**Hình 3: ServerFTP**



**Hình 4: ClientFTP**

**TÀI LIỆU THAM KHẢO**

* **Tài liệu giáo trình**

<https://ebookxanh.com/tai-lieu/do-an-ung-dung-client-server-mo-phong-chuc-nang-cua-cac-ung-dung-ftp-truyen-va-nhan-file-908283.html>

<https://tailieu.vn/doc/do-an-ung-dung-client-server-mo-phong-chuc-nang-cua-cac-ung-dung-ftp-truyen-va-nhan-file-1604877.html>

<https://stackoverflow.com/questions/16707816/ftp-client-server-model-for-file-transfer-in-java>

<http://www.dailyfreecode.com/code/ftp-client-server-1250.aspx>

<https://www.codeproject.com/Articles/293391/File-Transfer-Protocol-FTP-Client>

https://github.com/gudipati/FTP-ClientServer

* **Tài liệu số (Youtube)**

<https://www.youtube.com/watch?v=GQZCxN3zrMM> <https://www.youtube.com/watch?v=vrxlCS8vlXU&list=PLMYF6NkLrdN9myVt8qSDQqhBjGwwLuEzG>

<https://www.youtube.com/watch?v=VVqh0BCgFpM>