 **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

\*\*\*\*

**BÁO CÁO GIỮA KỲ**

**MÔN PHÁT TRIỂN ỨNG DỤNG (CÓ ĐỒ ÁN)**

**ĐỀ TÀI**

***FILE TRANSFER BETWEEN COMPUTERS***

**Sinh viên thực hiện:**

* Phan Anh Duy (MSSV: 15021501 )
* Nguyễn Tấn Hiển (MSSV: 15014891)-Nhóm trưởng
* Nguyễn Phan Minh Chí (MSSV:15021481)

**Link Github:** https://github.com/ptud2017/-n-PTUD-Topic-5-

**Giảng viên hướng dẫn:** ThS.Huỳnh Thái Học

**I.Tổng quan**

**1.1.Đề tài nghiên cứu (*FILE TRANSFER BETWEEN COMPUTERS*** )

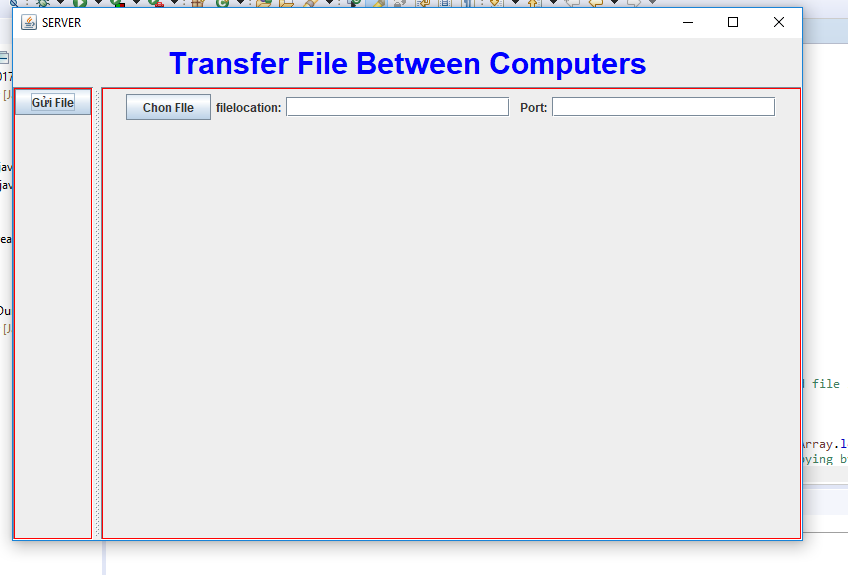
**Bảng kế hoạch thực hiện dề tài trong 10 tuần:**

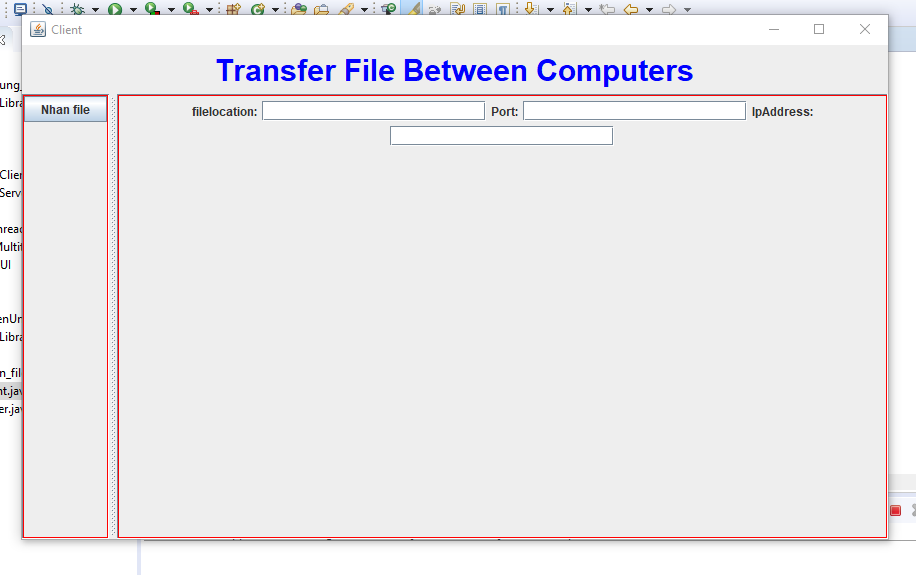
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tuần** | Công việc | Nguyễn Tấn Hiển | Nguyễn PhanMinh Chí | Phan Anh Duy |
| **1** | * Căn bản về ngôn ngữ lập trình java, ôn tập java căn bản | + Đọc thêm tài liệu java, thực hành tạo giao diện  + Lập kế hoạch cho 9 tuần tiếp theo | | |
| **2** | * Căn bản về mạng máy tính:   + Về giao thức UDP  + Mô hình hoạt động clients/server  + Stream | Tìm hiểu về Stream | Tìm hiểu về UDP | Tìm hiểu về client/serer |
| **3** | * Lập trình Socket UDP theo ví dụ (send 1 chuỗi) | Tìm hiểu về lập trình socket | | |
| Viết code phần UDP Server kết nối | Kiểm tra,thêm chú thích ,test chương trình | Viết code phần Client kết nối |
| **4** | * Đọc file và gửi file   + Tìm hiểu về đọc và gửi file  + Viết chương trình gửi file đơn giản cho Client và Server | - Đọc tài liệu  - Viết Code gửi file máy Server | - Đọc tài liệu  - Viết Code nhận file máy Client | - Đọc tài liệu  - Kiểm tra,sửa lỗi và chạy chương trình gửi file cơn bản |
| **5** | * Thực hành về thread | Tìm hiểu về thread trong lập trình socket java | | |
| Lập trình thread, ứng dụng vào mô hình clien/server,code | Lập trình thread, ứng dụng vào mô hình clien/server,code | Lập trình thread, ứng dụng vào mô hình clien/server,code |
| **6** | -Multithreaded Server trong Java | Đọc hiểu làm theo ví dụ | | |
| **7** | * Tạo giao diện cho chương trình client/server | Viết giao diện cho chương trình máy server | Sửa lỗi và hiệu chỉnh giao diện them nút chức năng | Viết giao diện cho chương trình máy client |
| **8** | * Chạy chương trình, kiểm tra, sửa lỗi | Hiệu chỉnh chương trình | | |
| **9** | * Chạy chương trình, kiểm tra, sửa lỗi |
| **10** | * Hoàn thiện chương trình và nộp sản phẩm | Kiểm tra chương trình và nộp sản phẩm | | |

* 1. **Tiến độ thực hiện đề tài:**

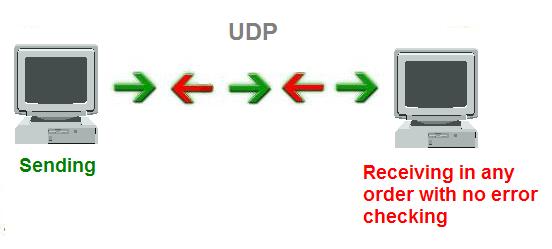
Bảng công việc hoàn thành của các thành viên

|  |  |  |
| --- | --- | --- |
| **Phan Anh Duy**  **(15021501)** | **Nguyễn Tấn Hiển (15014891)** | **Nguyễn Phan Minh Chí (15021481)** |
| -Viết code phần Client kết nối  - Kiểm tra,sửa lỗi và chạy chương trình gửi file cơ bản  - Sửa lỗi và hiệu chỉnh giao diện thêm nút chức năng gửi file(phía server) và nhận file( phía client) | -Viết code phần UDP Server kết nối  - Đọc tài liệu ,viết Code gửi file máy Server  - Viết giao diện cho chương trình máy Server | - Kiểm tra,thêm chú thích ,test chương trình  - Đọc tài liệu,viết Code nhận file máy Client  - Viết giao diện cho chương trình máy client |

**Giao diện Server**

**Giao diện client**

**II.Chi tiết thực hiện đề tài**: ***Truyền tập tin giữa hai máy tính sử dụng mạng Lan***

**1.Sơ lược giao thức UDP :** UDP (User Datagram Protocol) là giao thức truyền nhận dữ liệu dưới dạng các gói tin độc lập giữa các máy tính với nhau nhưng không đảm bảo sự toàn vẹn dữ liệu (không tin cậy). UDP là giao thức phi kết nối 

***Hình 1***

***Chế độ không kết nối (UDP)***

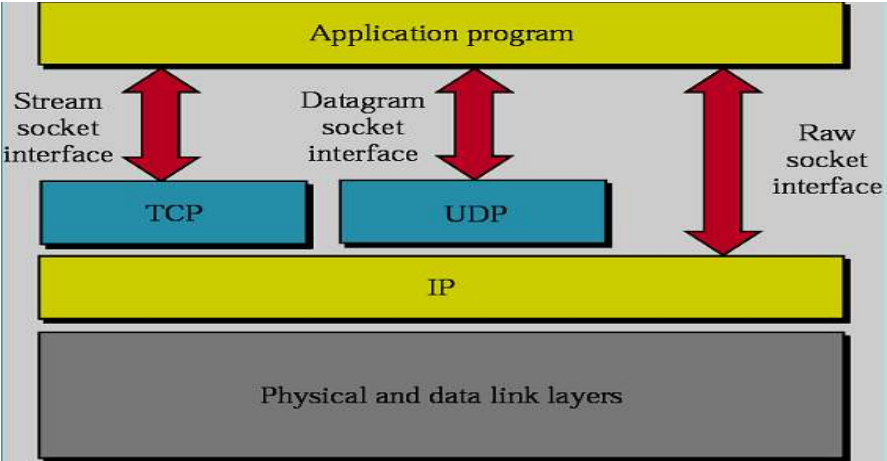
* Không tồn tại kênh giao tiếp ảo giữa hai bên giao tiếp
* Dữ liệu được gởi đi theo chế độ không bảo đảm: Không kiểm tra lỗi, không phát hiện không truyền lại gói tin bị lỗi hay mất, không bảo đảm thứ tự đến của các gói tin . . .
* Dữ liệu không chính xác, tốc độ truyền nhanh.
* Thích hợp cho các ứng dụng cần tốc độ, không cần chính xác cao: truyền âm thanh, hình ảnh . .

**2.Sơ lược về Socket:**

* Socket API – Socket Application Programming Interface
* Ðược giới thiệu ở BSD4.1 UNIX, 1981
* Ðược ứng dụng khởi tạo, sử dụng và hủy bỏ.
* Dùng cơ chế client/server
* Cung cấp hai dịch vụ chuyển dữ liệu thông qua socket API: unreliable datagram (UDP); reliable, byte stream-oriented (TCP)

Socket là giao diện giữa chương trình ứng dụng với tầng vận chuyển. Nó cho phép ta chọn giao thức sử dụng ở tầng vận chuyển là TCP hay UDP cho chương trình ứng dụng của minh.

Tầng vận chuyển giúp chuyển tiếp các thông điệp giữa các chương trình ứng dụng với nhau. Nó có thể hoạt động theo hai chế độ:Giao tiếp có nối kết, nếu sử dụng giao thức TCP hoặc giao tiếp không nối kết, nếu sử dụng giao thức UDP



**Hình 2:Mô hình socket**

* Stream Socket: dùng cho connection-oriented protocol như TCP.
* Datagram Socket: dùng cho connectionless protocol như UDP.
* Raw Socket: dùng cho một số protocol của một số ứng dụng đặc biệt, dùng các dịch vụ trực tiếp của lớp IP.

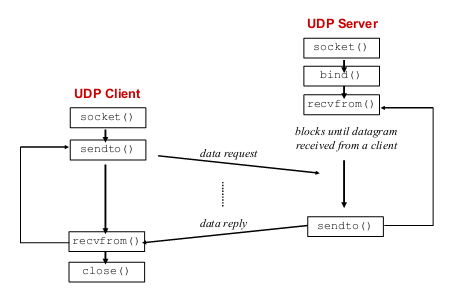
**3.Mô hình Client-Server sử dụng Socket(UDP)**

* Mô hình được phổ biến nhất trong các hệ thống phân tán là mô hình client/server.
* Mô hình truyền tin client/server hướng tới việc cung cấp dịch vụ.
* Quá trình trao đổi dữ liệu bao gồm:

+ Truyền một yêu cầu từ tiến trình client tới tiến trình server

+ Yêu cầu được server xử lý

+ Truyền đáp ứng cho client



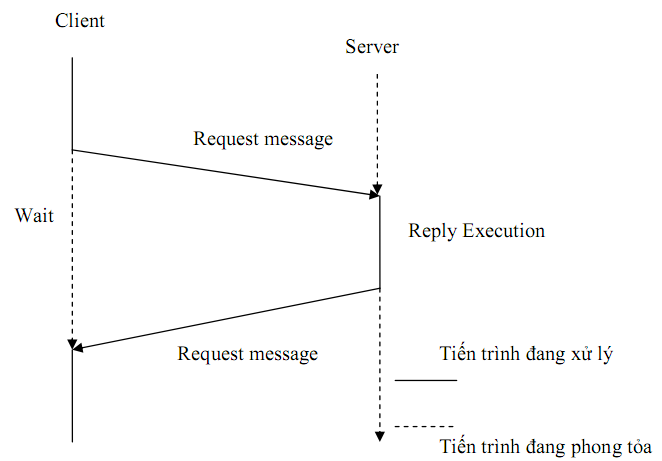
***Hình 3***

* *socket(): Server yêu cầu tạo một socket để có thể sử dụng các dịch vụ của tầng vận chuyển.*
* *bind(): Server yêu cầu gán số hiệu cổng cho socket*

Sau khi tạo Socket xong, Client và Server có thể trao đổi thông tin qua lại với nhau thông qua hai hàm sendto() và recvfrom(). Đơn vị dữ liệu trao đổi giữa Client và Server là các Datagram Package (Gói tin thư tín). Protocol của ứng dụng phải định nghĩa khuôn dạng và ý nghĩa của các Datagrame Package. Mỗi Datagrame Package có chứa thông tin về địa chỉ người gởi và người nhận (IP, Port).

**\*Quá trình trao đổi dữ liệu theo client/server**

Mô hình client/server thường được cài đặt dựa trên các thao tác cơ bản là gửi (send) và nhận (receive).



***Hình 4***

**III.Kết luận:**

- Cơ bản nhóm đã tạo được giao diện: máy client và máy server

- Các nút chức năng gửi và nhận file đơn giản trên giao diện client và server (Gửi nhận file trên cùng ổ đĩa).

**Công việc tuần còn lại:**

* Nhóm sẽ tiếp tục hoàn thiện code gửi và nhận file.
* Xem lại cách kết nối giữa hai máy.

**\*Link hình:**

<http://dulieu.tailieuhoctap.vn/books/cong-nghe-thong-tin/quan-tri-mang/file_goc_777812.pptx>

**\*Các link tài liệu tham khảo :**

1.Opp

<https://www3.ntu.edu.sg/home/ehchua/programming/java/J3a_OOPBasics.html#zz-3.1>

<http://docs.oracle.com/javase/tutorial/java/concepts/index.html>

<http://congdongjava.com/forum/threads/slide-oop-l%E1%BA%ADp-tr%C3%ACnh-h%C6%B0%E1%BB%9Bng-%C4%91%E1%BB%91i-t%C6%B0%E1%BB%A3ng-trong-java.953/>

2.Stream

<http://www.javabeginner.com/learn-java/java-threads-tutorial>

<http://tutorials.jenkov.com/java-concurrency/creating-and-starting-threads.html>

3.UDP-Client/Server

<http://www.cit.ctu.edu.vn/~dtnghi/netprog/4a-Socket.pdf>

4.Lập Trình socket

<http://o7planning.org/vi/10393/huong-dan-lap-trinh-java-socket>

http://www.codingdevil.com/2015/08/transfer-file-using-java-socket.html

5.Thread

<http://www.javabeginner.com/learn-java/java-threads-tutorial>

<http://tutorials.jenkov.com/java-concurrency/creating-and-starting-threads.html>

<http://laptrinhvb.net/bai-viet/chuyen-de-vb-net/Gioi-thieu-va-huong-dan-su-dung-Thread-va-Multi-Thread,-Process-trong-visual-dot-net/a54766b0b5180aa5.html>