

**BÁO CÁO CUỐI KÌ**

**ĐỀ TÀI:**

**XÂY DỰNG CHƯƠNG TRÌNH**

**CHAT ROOM**

**Sinh viên thực hiện:**

**Trần Nhựt Phi 15067621**

**Nguyễn Tiến Khoa 15050851**

**Nguyễn Quang Khải 15076041**

**Link github: https://github.com/tuankhoa2009/NguyenKhoa**

**GVHD: ThS: Huỳnh Thái Học**

**Mục lục:**

[**1.Tổng quan 2**](#_Toc500181418)

[1.1. Xây dựng chương trình Chat room bằng Java và nhiệm vụ 2](#_Toc500181419)

[**2.Chi tiết chương trình Chat room 4**](#_Toc500181420)

[2.1.Khái niệm giao thức TCP/IP 4](#_Toc500181421)

[2.2. Khái niệm Client/Server 6](#_Toc500181422)

[2.3. Khái niệm cơ chế Socket trong Java 7](#_Toc500181423)

[2.4. Thread 7](#_Toc500181424)

[2.5. Source code 7](#_Toc500181424)

[**3. Kết luận 13**](#_Toc500181425)

[**Tài liệu tham khảo: 13**](#_Toc500181426)

**BÀI BÁO CÁO**

# 1.Tổng quan

### 1.1. Xây dựng chương trình Chat room bằng Java và nhiệm vụ

**Bảng phân công công việc**

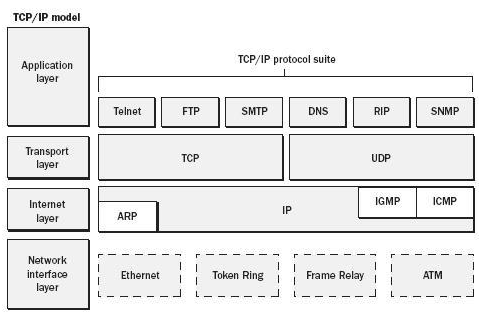
|  |  |  |
| --- | --- | --- |
| STT | Nội dung | Người thực hiện |
| Tuần 1 | -Tìm và đọc tài liệu TCP/IP  -Tìm và đọc tài liệu Socket  -Tìm và đọc tài liệu Client/Server | Khải  Khoa  Phi |
| Tuần 2 | -Tiếp tục nghiên cứu về công việc tuần 1 | Cả nhóm |
| Tuần 3 | -Tìm hiểu Socket TCP viết bằng JAVA.Chat Ping-Pong gửi message Client-Server  -Viết class Server  -Viết class Client | Cả nhóm  Phi  Khải |
| Tuần 4 | -Tìm hiểu Thread.  -Cải tiến bài Ping-Pong Server thep Thread | Cả nhóm  Khoa |
| Tuần 5 | -Cải tiến Thread Để cho 1 server có thể kết nối nhiều client | Khoa |
| Tuần 6 | -Server làm nhiệm vụ  trung gian cho client có thể gửi Message đến các client khác. | Phi  Khoa |
| Tuần 7 | -Test chỉnh sửa , bổ sung những thứ cần thiết cho các class.  -Thiết kế giao diện | Khoa, Phi  Khải |
| Tuần 8 | -Nghiên cứu lấy IP để kết nối 2 máy trong mạng LAN | Khoa  Phi |
| Tuần 9 | -Tiếp tục công việc tuần 8 | Khoa,Phi |
| Tuần 10 | -Test, chỉnh sửa chương trình | Cả nhóm |

**1.2.Tiến độ thực hiện**

|  |  |  |
| --- | --- | --- |
| STT | Nội dung | Tiến độ |
| Tuần 1 | -Tìm và đọc tài liệu TCP/IP  -Tìm và đọc tài liệu Socket  -Tìm và đọc tài liệu Client/Server | Hoàn thành |
| Tuần 2 | -Tiếp tục nghiên cứu về công việc tuần 1 | Hoàn thành |
| Tuần 3 | -Tìm hiểu Socket TCP viết bằng JAVA.Chat Ping-Pong gửi message Client-Server  -Viết class Server  -Viết class Client | Hoàn thành |
| Tuần 4 | -Tìm hiểu Thread.  -Cải tiến bài Ping-Pong Server thep Thread | Hoàn thành |
| Tuần 5 | -Cải tiến Thread Để cho 1 server có thể kết nối nhiều client | Hoàn thành |
| Tuần 6 | -Server làm nhiệm vụ  trung gian cho client có thể gửi Message đến các client khác. | Hoàn Thành |
| Tuần 7 | -Test chỉnh sửa , bổ sung những thứ cần thiết cho các class.  Thiết kế giao diện | Hoàn Thành |
| Tuần 8 | - Nghiên cứu lấy IP để kết nối 2 máy trong mạng LAN | Hoàn Thành |
| Tuần 9 | -Tiếp tục công việc tuần 8 | Hoàn Thành |
| Tuần 10 | -Test, chỉnh sửa chương trình | Hoàn Thành |

# 2.Chi tiết chương trình Chat room

### 2.1.Khái niệm giao thức TCP/IP



-Giao thức TCP/IP được phát triển từ mạng ARPANET và Internet và được dùng như giao thức mạng và vận chuyển trên mạng Internet. TCP (Transmission Control Protocol) là giao thức thuộc tầng vận chuyển và IP (Internet Protocol) là giao thức thuộc tầng mạng của mô hình OSI. Giao thức TCP/IP hiện nay là giao thức được sử dụng rộng rãi nhất để liên kết các máy tính và các mạng.

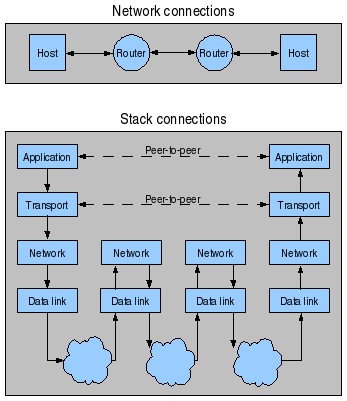
-TCP/IP được xem như giản lược của mô hình tham chiếu OSI với 4 tầng như sau:

o Tầng Liên Kết (Datalink Layer)

o Tầng Mạng (Internet Layer)

o Tầng Giao Vận (Transport Layer)

o Tầng Ứng Dụng (Application Layer)

[](http://www.vnpro.vn/wp-content/uploads/2015/11/C%C3%A1c-t%E1%BA%A7ng-trong-B%E1%BB%99-giao-th%E1%BB%A9c-TCP-IP.jpg)

Các tầng trong Bộ giao thức TCP/IP

***+Tầng liên kết:*** Tầng liên kết (còn được gọi là tầng liên kết dữ liệu hay tầng giao tiếp mạng) là tầng thấp nhất trong mô hình TCP/IP, bao gồm các thiết bị giao tiếp mạng và các chương trình cung cấp các thông tin cần thiết để có thể hoạt động, truy nhập đường truyền vật lý qua các thiết bị giao tiếp mạng đó.

***+Tầng Internet:*** Tầng Internet ( hay còn gọi là tầng Mạng) xử lý quá trình truyền gói tin trên mạng, các giao thức của tầng này bao gồm : IP ( Internet Protocol) , ICMP ( Internet Control Message Protocol) , IGMP ( Internet Group Message Protocol )

***+Tầng giao vận:*** Tầng giao vận phụ trách luồng dữ liệu giữa 2 trạm thực hiện các ứng dụng của tầng trên, tầng này có 2 giao thức chính là TCP ( Transmisson Control Protocol) và UDP ( User Datagram Protocol )

***+Tầng ứng dụng*** là tầng trên của mô hình TCP/IP bao gồm các tiến trình và các ứng dụng cung cấp cho người sử dụng để truy cập mạng.

### 2.2. Khái niệm Client/Server

**2.2.1. Server**

-Là một máy tính cho phép thực hiện yêu cầu của một hoặc nhiều người dùng từ phía client. Khi có yêu cầu từ phía người dùng, server sẽ chịu trách nhiệm xử lý và trả về kết quả cho người dùng, các kết quả này có thể là một tài nguyên nào đó nằm trên server hay một cái gì đó ví dụ như kết quả của một phép tính. Xét ở một khía cạnh khác thì server có thể được định nghĩa như một máy tính có nhiều người sử dụng vì một server phải xử lý rất nhiều các yêu cầu khác nhau từ client khác nhau vì vậy server sẽ hoạt động tốt hơn nếu như có thể xử lý đa nhiệm, tức là các tính năng hoạt động một cách độc lập và song song với nhau.

**2.2.2. Client**

  -Client là máy tính chỉ được sử dụng bởi một người dùng, máy client có thể sử dụng các hệ điều hành như Windows, MacOs, Ubuntu ...v.v.v và đóng vai trò tương tác giữa người dùng và Server. Bản thân mỗi client được tích hợp nhiều tính năng và khi thông qua kết nối với Server, Client còn có thể sử dụng thêm những tính năng mà Server cung cấp, Client chỉ cần nhập các thông tin cần thiết (các tham số đầu vào) và thực hiện gửi yêu cầu lên Server, sau khi Server xử lý xong sẽ trả về kết quả cho Client. Client và Server có thể chia sẻ tài nguyên trên máy cho nhau và Client được coi là người sử dụng dịch vụ do một hoặc nhiều Server cung cấp.

### 2.3. Khái niệm cơ chế Socket trong Java

-**Socket** là một cổng logic mà một chương trình sử dụng để kết nối với một chương trình khác chạy trên một máy tính khác trên Internet. Chương trình mạng có thể sử dụng nhiều Socket cùng một lúc, nhờ đó nhiều chương trình có thể sử dụng Internet cùng một lúc. Có 2 loai:

**+ Stream Socket**: Dựa trên giao thức TCP( Tranmission Control Protocol) việc truyền dữ liệu chỉ thực hiện giữa 2 quá trình đã thiết lập kết nối. Giao thức này đảm bảo dữ liệu được truyền đến nơi nhận một cách đáng tin cậy, đúng thứ tự nhờ vào cơ chế quản lý luồng lưu thông trên mạng và cơ chế chống tắc nghẽn.

**+ Datagram Socket**: Dựa trên giao thức UDP( User Datagram Protocol) việc truyền dữ liệu không yêu cầu có sự thiết lập kết nối giữa 2 quá trình. Ngược lại với giao thức TCP thì dữ liệu được truyền theo giao thức UDP không được tin cậy, có thế không đúng trình tự và lặp lại. Tuy nhiên vì nó không yêu cầu thiết lập kết nối không phải có những cơ chế phức tạp nên tốc độ nhanh…ứng dụng cho các ứng dụng truyền dữ liệu nhanh như chat, game…..

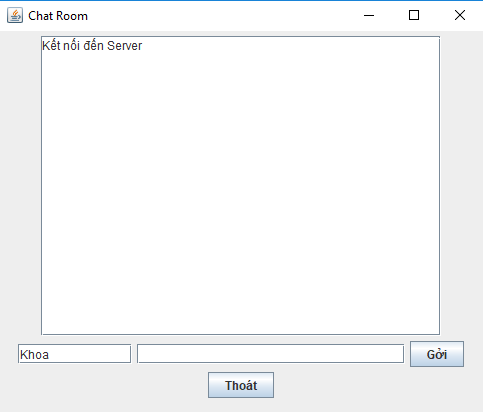
### 2.4. Thread

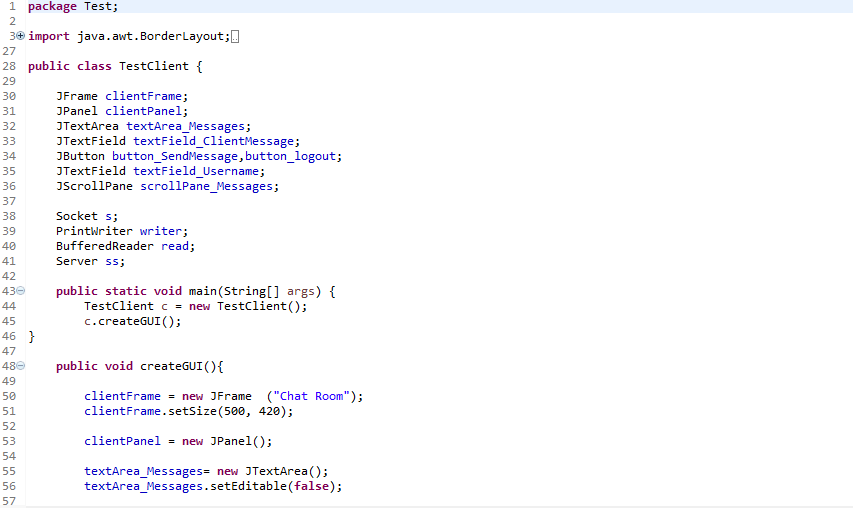
**\*Đa luồng trong javaMultithreading**

Multithreading: Là khả năng thực hiện đồng thời nhiều phần khác nhau của một chương trình được gọi là thread. Ví dụ trong Microsoft Excel chúng ta có thể làm việc đồng thời với các sheet khác nhau.

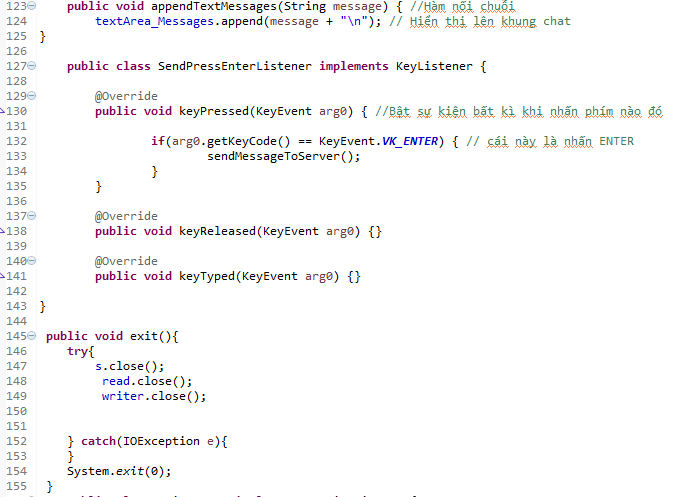
**2.5. Source code**

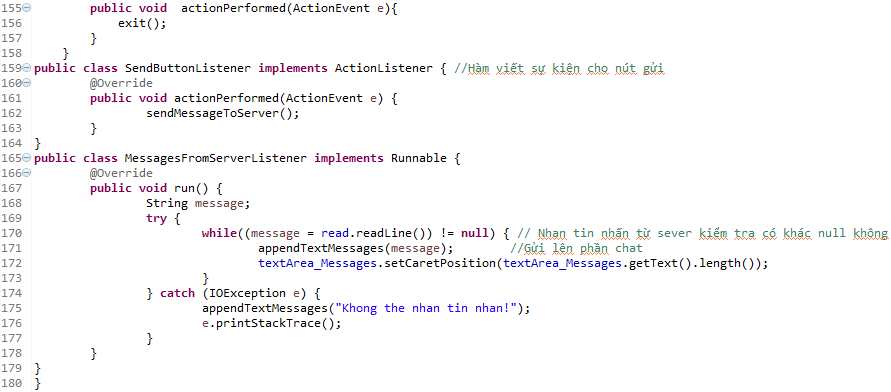
Client:



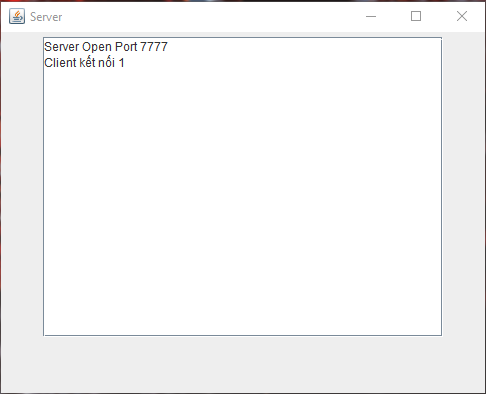






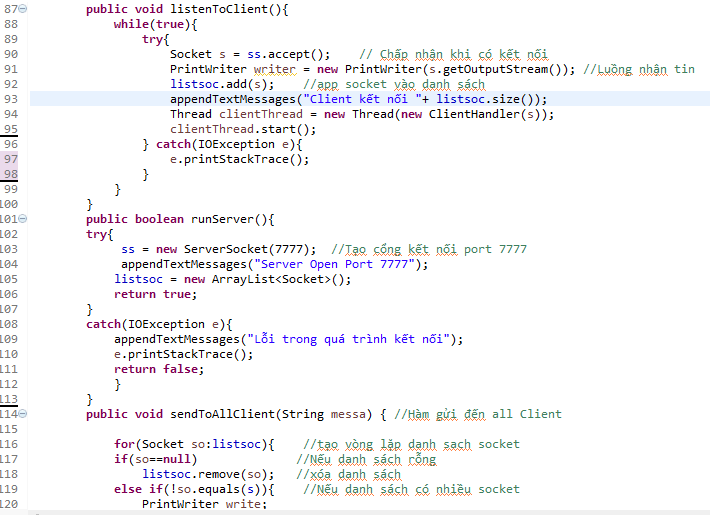


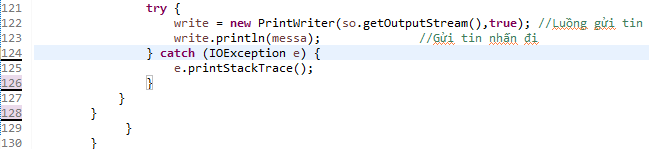
Server:











# 3. Kết luận

-Hoàn thành được 90% chương trình

-10% hạn chế còn lại:

+Chưa sửa được lỗi gỏ không có dấu

-Hướng phát triển ứng dụng:

+ Thêm icon;

+ Gửi hình, video, âm thanh;

+ Gửi tệp;

+ Gọi video trong ứng dụng;

------------------------------------------------------------------------------------------------------------

# Tài liệu tham khảo:

<http://www.toancau.vn/vietnam/4/5/32/giao-thuc-tcpip-la-gi>

<http://www.dreamincode.net/forums/topic/301933-multithreaded-chatroom-server/>

<https://stackoverflow.com/questions/20807941/server-client-chat-room-in-java>

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