**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**PBL4: DỰ ÁN HỆ ĐIỀU HÀNH & MẠNG MÁY TÍNH**

**Đề tài : Xây dựng chương trình ứng dụng nhắn tin**

**SINH VIÊN THỰC HIỆN:**

**Nguyễn Hữu Phúc LỚP: 21TCLC\_DT3 NHÓM: 21N13**

**Hồ Anh Dũng LỚP: 21TCLC\_DT3 NHÓM: 21N13**

**GIẢNG VIÊN HƯỚNG DẪN: Nguyễn Thị Lệ Quyên**

**Đà Nẵng 4/2/ 2024**

MỤC LỤC

[1](#_Toc153545410)

[MỤC LỤC 2](#_Toc153545411)

[DANH SÁCH HÌNH VẼ 3](#_Toc153545412)

[GIỚI THIỆU ĐỀ TÀI 4](#_Toc153545413)

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 5](#_Toc153545414)

[1.1. Tổng quan về Ứng Dụng Nhắn Tin 5](#_Toc153545415)

[1.1.1. Tổng quan 5](#_Toc153545416)

[1.2. Lập trình Socket với TCP 7](#_Toc153545418)

[1.2.1. Mô hình Client-Server 7](#_Toc153545419)

[1.2.2. Giao thức TCP 10](#_Toc153545420)

[1.2.3. Mô hình truyền tin Socket 12](#_Toc153545421)

[CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 15](#_Toc153545422)

[2.1. Biểu đồ use case( Use case diagram) 15](#_Toc153545423)

[2.2. Biểu đồ hoạt động (activity diagram) 16](#_Toc153545424)

[2.3. Mô tả chức năng hệ thống 22](#_Toc153545425)

[CHƯƠNG 3. TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ 22](#_Toc153545426)

[3.1. Giao diện chương trình 22](#_Toc153545427)

[3.2. Đánh giá kết quả 25](#_Toc153545428)

[CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 25](#_Toc153545429)

[4.1. Kết luận 25](#_Toc153545430)

[4.2. Hướng phát triển 25](#_Toc153545431)

[TÀI LIỆU THAM KHẢO 26](#_Toc153545432)

DANH SÁCH HÌNH VẼ

[Hình 1: Mô hình Server 6](#_Toc153545089)

[Hình 2: Giới thiệu TCP 8](#_Toc153545090)

[Hình 3: Cách thức hoạt động của Socket 11](#_Toc153545091)

[Hình 4: Minh họa socket 12](#_Toc153545092)

[Hình 5: Sơ đồ usecase 13](#_Toc153545093)

[Hình 6: Biểu đồ hoạt động đăng ký tài khoản 15](#_Toc153545094)

[Hình 7: Biểu đồ hoạt động đăng nhâp 16](#_Toc153545095)

[Hình 8: Biểu đồ hoạt động download Image 18](#_Toc153545098)

[Hình 9: Biểu đồ hoạt động download File 20](#_Toc153545099)

[Hình 10: Giao diện chính Server 21](#_Toc153545100)

[Hình 11: Giao diện đăng nhập 22](#_Toc153545101)

[Hình 12: Giao diện trang chủ 22](#_Toc153545102)

[Hình 13: Chức năng gửi ảnh 23](#_Toc153545103)

[Hình 14: Chức năng gửi file 23](#_Toc153545104)

[Hình 15:Chức năng gửi emoji 23](#_Toc153545105)

[Hình 16: Chức năng dowload file 24](#_Toc153545106)

[Hình 17: Chức năng xem và dowload image 24](#_Toc153545107)

GIỚI THIỆU ĐỀ TÀI

Ngày nay, với sức mạnh của công nghệ hiện đại, con người ngày càng thoát khỏi những khó khăn trong cuộc sống và công việc tấp nập. Để chủ động hội nhập vào thế giới nhanh chóng phát triển này, chúng ta cần sử dụng các công cụ hiện đại, đặc biệt là trong lĩnh vực công nghệ thông tin.

Một số ứng dụng quan trọng của công nghệ thông tin là trong các ngành khoa học và kỹ thuật, nơi mà sự hỗ trợ từ các phần mềm và ứng dụng là không thể thiếu. Điển hình là nhu cầu truy cập internet để cung cấp và lấy thông tin qua các máy chủ. Dựa trên nhu cầu này, nhóm chúng tôi quyết định thực hiện đề tài "Triển khai ứng dụng nhắn tin bằng lập trình Socket".

Trong thời gian gần đây, sự phát triển của công nghệ thông tin đã mang lại nhiều công cụ hữu ích. Đồ án của chúng tôi tập trung vào triển khai một ứng dụng nhắn tin sử dụng lập trình Socket, một công nghệ mạnh mẽ cho việc xây dựng các ứng dụng mạng.

Mục tiêu chính của đồ án là hiểu rõ cơ chế hoạt động của ứng dụng nhắn tin và thực hiện nó thông qua việc sử dụng lập trình Socket. Chúng tôi sẽ xây dựng một ứng dụng nhắn tin đơn giản có khả năng gửi và nhận tin nhắn giữa các máy khách. Đồ án sẽ trình bày các bước thiết kế, cài đặt, kiểm thử và đánh giá ứng dụng nhắn tin bằng lập trình Socket.

Chúng tôi hi vọng rằng đồ án không chỉ mang lại kiến thức vững về lập trình mạng mà còn giúp hiểu rõ hơn về cách triển khai các ứng dụng truyền thông giữa các máy khách một cách hiệu quả. Đồ án sẽ đem lại trải nghiệm học thuật giá trị và thú vị cho chúng tôi.

Chân thành cảm ơn sự quan tâm và hỗ trợ của giảng viên Nguyễn Thị Lệ Quyên đối với đồ án này. Chúng tôi hy vọng rằng hành trình này sẽ đầy đủ những khám phá mới và kỹ năng lập trình mạng thêm sâu rộng.

# CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

## Tổng quan về Ứng Nhắn tin.

**1.1.1.tổng quan :**

Ứng dụng trên điện thoại quá nhiều, người dùng sẽ cần đến một công cụ để có thể chứa đựng tất cả và ứng dụng tin nhắn hứa hẹn trở thành cổng thông tin duy nhất trong tương lai gần.

• Khả năng trao đổi tin nhắn văn bản từ người này sang người khác thông qua loạt ứng dụng được gắn kết với nhau, các tính năng bổ sung như group chat, hình ảnh, nhãn dán và video hay voice cũng có thể dễ dàng lan truyền. Điển hình như WhatsApp, ứng dụng phổ biến bên ngoài nước Mỹ và các quốc gia Đông Á, nơi mà lượng tin nhắn chiếm hơn hơn 50% so với toàn bộ hệ thống SMS toàn cầu. Bên cạnh đó cũng có Facebook Messenger đang hoạt động khá tốt.

• Sự bùng nổ ứng dụng xuất phát từ hành vi hay tâm lý có thể tạo nên trào lưu, khiến một ứng dụng có thể lan truyền đi như virus, ví dụ như Instagram, Snapchat, Yo, Yik Yak, Secret hay Meerkat. Thách thức đối với những ứng dụng này là dựa quá nhiều vào hành vi của người sử dụng để tạo ra sự tăng trưởng. Khi ứng dụng không còn gì mới lạ hay không đủ chiều sâu thì thường chỉ chỉ tạo ra tính nhất thời và khó có khả năng đi xa trên thị trường. Phương thức hoạt động của Snapchat là một trong những minh chứng cho đặc điểm trên.

• Ứng dụng tin nhắn như một nền tảng là công cụ điển hình trong việc tạo và mở rộng trải nghiệm dành cho người dùng. Điều này có thể phụ thuộc vào môi trường xã hội, điển hình như Wechat với 500 triệu người sử dụng, phần lớn ở Trung Quốc, trong khi Line tại Nhật Bản, Zalo ở Việt Nam và Kik ở Mỹ là những công cụ chủ đạo**.**

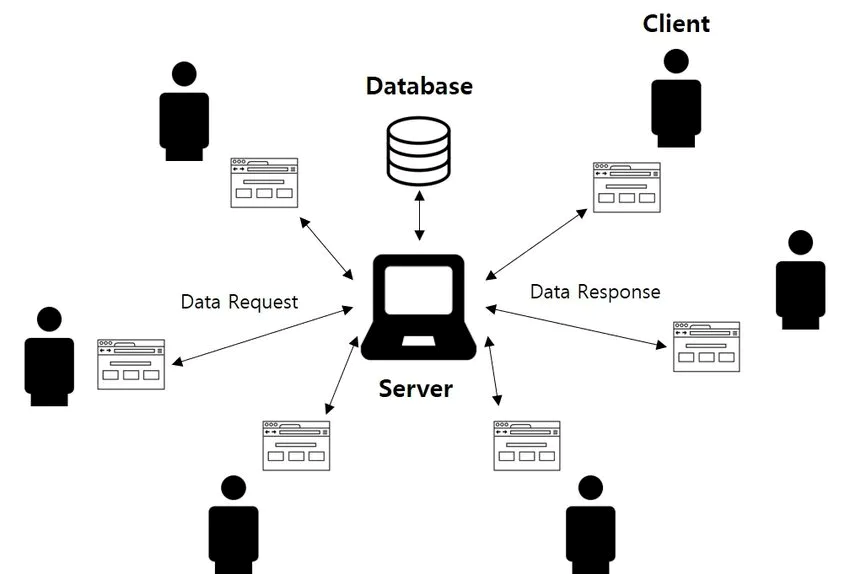
## Lập trình Socket với TCP

### Mô hình Client-Server

#### Khái niệm

Mô hình client server (mô hình mạng máy khách – máy chủ) là một cấu trúc ứng dụng phân tán. Nó phân vùng các nhiệm vụ hay workload giữa các nhà cung cấp tài nguyên hoặc dịch vụ, gọi là server, và người yêu cầu dịch vụ (Client).

Trong kiến trúc Client Server, khi máy Client gửi yêu cầu dữ liệu đến Server thông qua Internet, server sẽ chấp nhận quy trình được yêu cầu. Sau đó gửi các gói dữ liệu được yêu cầu trở lại client. Client không chia sẻ bất kỳ tài nguyên nào của họ.



Hình 1: Mô hình Server

#### Nguyên tắc

* Client :Khi nói đến Client (khách hàng), thì nó có nghĩa là một người hay một tổ chức sử dụng một dịch vụ cụ thể nào đó. Trong thế giới kỹ thuật số cũng tương tự như vậy. Client là một máy tính (Host), tức là có khả năng nhận thông tin hoặc sử dụng một dịch vụ cụ thể từ các nhà cung cấp dịch vụ (Server).
* Server: Server còn được định nghĩa như là một máy tính nhiều người sử dụng (multiuser computer). Vì một máy chủ phải quản lý nhiều yêu cầu từ các máy khách trên mạng cho nên nó hoạt động sẽ tốt hơn nếu hệ điều hành của nó là đa nhiệm với các tính năng hoạt động độc lập song song với nhau như hệ điều hành UNIX, Windows.

Server cung cấp và điều khiển các tiến trình truy cập vào tài nguyên của hệ thống. Các ứng dụng chạy trên máy chủ phải được tách rời nhau để một lỗi của ứng dụng này không làm hỏng ứng dụng khác. Tính đa nhiệm đảm bảo một tiến trình không sử dụng toàn bộ tài nguyên hệ thống.

Máy chủ đóng vai trò như là một nhà cung cấp dịch vụ cho các máy khách yêu cầu tới khi cần, các dịch vụ như cơ sở dữ liệu, in ấn, truyền tệp, hệ thống... Các ứng dụng máy chủ cung cấp các dịch vụ mang tính chức năng để hỗ trợ cho các hoạt động trên các máy khách có hiệu quả hơn. Sự hỗ trợ của các dịch vụ này có thể là toàn bộ hoặc chỉ một phần thông qua IPC.

Để đảm bảo tính an toàn trên mạng cho nên máy chủ này còn có vai trò như là một nhà quản lý toàn bộ quyền truy cập dữ liệu của các máy khách, nói cách khác đó là vai trò quản trị mạng. Có rất nhiều cách thức hiện nay nhằm quản trị có hiệu quả, một trong những cách đang được sử dụng đó là dùng tên đăng nhập và mật khẩu Ưu nhược điểm

* Ưu điểm:
* Tập trung

Ưu điểm chính của mô hình mạng khách chủ là khả năng kiểm soát tập trung (Centralization) được tích hợp sẵn. Với mô hình này, tất cả thông tin cần thiết đều được đặt ở một vị trí duy nhất. Việc này rất hữu ích cho những quản trị viên mạng. Vì họ có được toàn quyền quản lý và điều hành

Bằng tính năng này, mọi sự cố trong mạng đều có thể được giải quyết ở một nơi duy nhất. Và do đó, việc cập nhật tài nguyên, dữ liệu cũng trở nên dễ dàng hơn.

* Bảo mật

Trong mạng Client Server, dữ liệu được bảo vệ tốt do kiến trúc tập trung của mạng. Nó có thể được áp dụng các biện pháp kiểm soát truy cập, sao cho chỉ những người dùng được ủy quyền mới có thể truy cập.Một trong những cách để làm vậy là áp đặt thông tin đăng nhập, chẳng hạn như Username hay Password. Hơn nữa, nếu dữ liệu

bị mất, các file có thể được khôi phục dễ dàng từ chỉ một bản sao lưu duy nhất.

* Khả năng mở rộng

Mô hình mạng Client server có khả năng mở rộng tốt. Bất cứ khi nào người dùng cần, họ có thể tăng số lượng tài nguyên, chẳng hạn như số Client hay Server. Do đó có thể dễ dàng tăng kích thước của Server mà không bị gián đoạn nhiều.

* Khả năng truy cập

Không có sự phân biệt giữa các vị trí hay nền tảng khác nhau, mọi client đều có thể đăng nhập vào hệ thống. Bằng cách này, tất cả nhân viên đều có thể truy cập thông

tin của công ty của họ, không cần phải dùng một terminal mode hay một bộ xử lý nào.

* Nhược điểm:
* Tắc nghẽn lưu lượng

Nhược điểm chính của mô hình mạng Client Server là tắc nghẽn lưu lượng. Nếu có quá nhiều Client tạo request từ cùng một Server, nó có thể làm chậm kết nối. Hoặc tệ hơn là dẫn đến hiện tượng crash. Một server bị quá tải có thể tạo ra nhiều vấn đề trong

việc truy cập thông tin.

* Độ bền

Như ta đã biết, mạng Client Server là mạng tập trung. Nên nếu Server chính xảy ra sự cố hay bị nhiễu, toàn bộ hệ thống mạng sẽ bị gián đoạn. Do đó, các mạng client server sẽ thiếu tính ổn định và độ bền.

* Chi phí

Chi phí thiết lập và bảo trì server trong các mạng client server thường khá cao. Vì các hệ thống mạng có sức mạnh lớn có thể có giá rất đắt. Do đó, không phải tất cả người dùng đều có thể chi trả được.

* Bảo trì

Khi các Server được triển khai, nó sẽ hoạt động không ngừng nghỉ. Có nghĩa là nó cầnđược quan tâm đúng mức nếu có bất kỳ vấn đề gì thì phải giải quyết ngay. Vì vậy, cầncó một nhà quản lý mạng chuyên biệt để duy trì hoạt động của Server.

* Tài nguyên

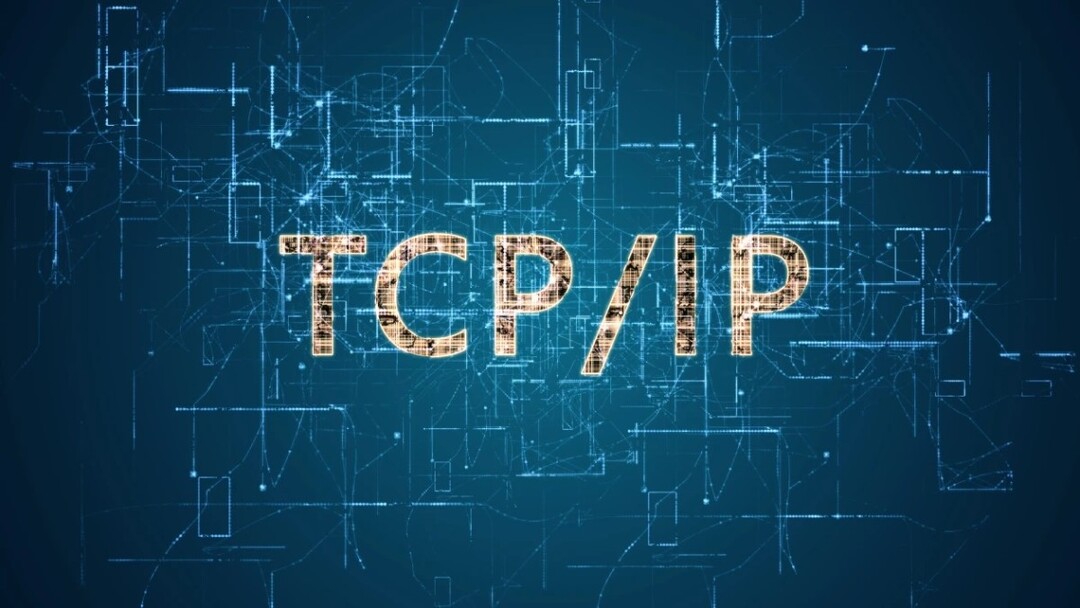
Không phải tất cả tài nguyên hiện có ở trên Server đều có thể sử dụng được. Ví dụ như bạn không thể in trực tiếp tài liệu trên web, hoặc chỉnh sửa bất kỳ thông tin nào trên ổ cứng của Client

### Giao thức TCP

#### Khái Niệm và Đặc Điểm Cơ Bản

TCP, hay Transmission Control Protocol là một giao thức mạng quan trọng được sử dụng trong việc truyền dữ liệu qua một mạng nào đó. Một giao thức trong phạm vi mạng là một tập hợp các quy tắc và trình tự kiểm soát việc thực hiện truyền dữ liệu sao cho tất cả mọi người trên thế giới bất kể vị trí địa lý, bất kể ứng dụng, phần mềm họ đang sử dụng đều có thể thao tác theo cùng một phương thức giống nhau được gọi là TCP.

TCP thường kết hợp với IP (Giao thức Internet) theo một cặp được gọi là TCP/IP. Bạn có thể bắt gặp cụm thuật ngữ này trong phần network setting trên máy tính, điện thoại thông minh hoặc các thiết bị cầm tay của mình. IP sẽ xử lý việc gán địa chỉ và chuyển tiếp các gói tin từ nguồn đến đích trong khi TCP kiểm soát độ tin cậy của truyền dẫn.



Hình 2: Giới thiệu TCP

* Tính Chất Chính

- Đáng Tin Cậy: TCP đảm bảo rằng mọi dữ liệu được gửi từ một bên sẽ được nhận đúng và đầy đủ bởi bên kia, thông qua việc sử dụng các cơ chế như xác nhận (acknowledgments) và tái tạo dữ liệu khi cần.

- Dựa Trên Kết Nối: Mọi truyền thông bằng TCP bắt đầu bằng việc thiết lập một kết nối logic giữa máy gửi và máy nhận. Quá trình này bao gồm ba bước: thiết lập kết nối (three-way handshake), truyền dữ liệu, và đóng kết nối (four-way handshake).

- Kiểm Soát Luồng (Flow Control): TCP sử dụng cơ chế kiểm soát luồng để đảm bảo rằng máy nhận không bị quá tải với dữ liệu, giữ cho quá trình truyền thông diễn ra một cách hiệu quả.

- Tái Tạo Dữ Liệu (Data Retransmission): Khi dữ liệu bị mất trên đường truyền, TCP tự động tái tạo lại dữ liệu đó để đảm bảo tính toàn vẹn và đúng đắn của dữ liệu.

#### Chức Năng và Ứng Dụng

- Truyền Tải Dữ Liệu: TCP chủ yếu được sử dụng để truyền tải dữ liệu giữa các máy tính trên mạng. Điều này bao gồm truyền tải dữ liệu trong các ứng dụng web, email, truyền thông đa phương tiện, và nhiều ứng dụng mạng khác.

- Ứng Dụng Trong Giao Thức HTTP: Giao thức HTTP (Hypertext Transfer Protocol), sử dụng TCP để truyền tải dữ liệu giữa máy chủ web và trình duyệt của người dùng, đảm bảo tính ổn định và đáng tin cậy của trang web.

- Truyền Tải Dữ Liệu Tự Do Lưu Ý (FTP): Khi bạn tải về hoặc tải lên tệp tin thông qua giao thức FTP, TCP đảm bảo rằng dữ liệu được truyền tải một cách an toàn và đáng tin cậy.

TCP/IP được sử dụng để kết nối thông tin trong Internet. Người dùng có thể thực hiện cung cấp thông tin từ xa, gửi mail, truyền file, ảnh hoặc phân phối web trên mạng Internet. Phương thức này cũng cho phép truy cập máy chủ từ xa, thay đổi trạng thái thông tin truyền trong môi trường Internet. Với giao thức TCP/IP, người dùng có thể dễ dàng thay đổi cách biểu thị thông tin thông qua các giao thức cơ bản hoặc giao thức ở mỗi lớp khi thông tin được truyền qua. Nhờ đó, thao tác truyền thông tin sẽ chính xác và hiệu quả hơn.

Giao thức TCP đóng vai trò quan trọng trong việc đảm bảo tính toàn vẹn và ổn định của truyền thông mạng. Sự ổn định này làm cho TCP trở thành một trong những lựa chọn hàng đầu cho các ứng dụng đòi hỏi độ tin cậy cao trên môi trường mạng.

### Mô hình truyền tin Socket

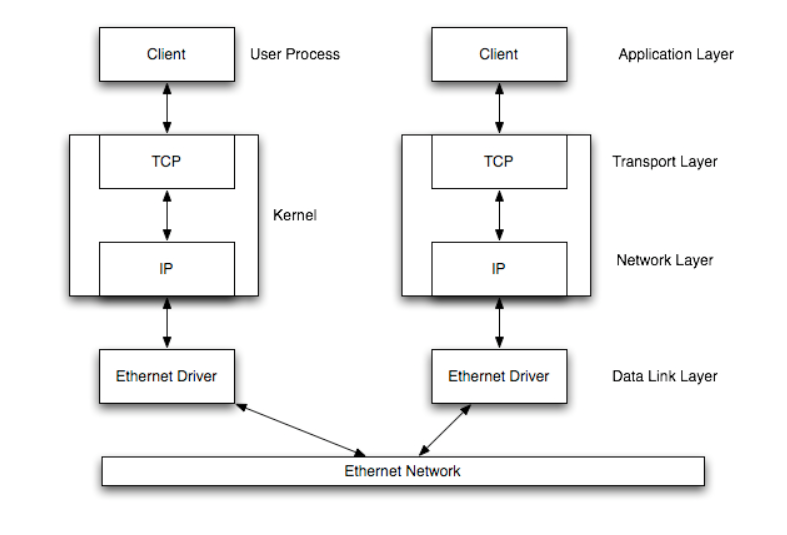
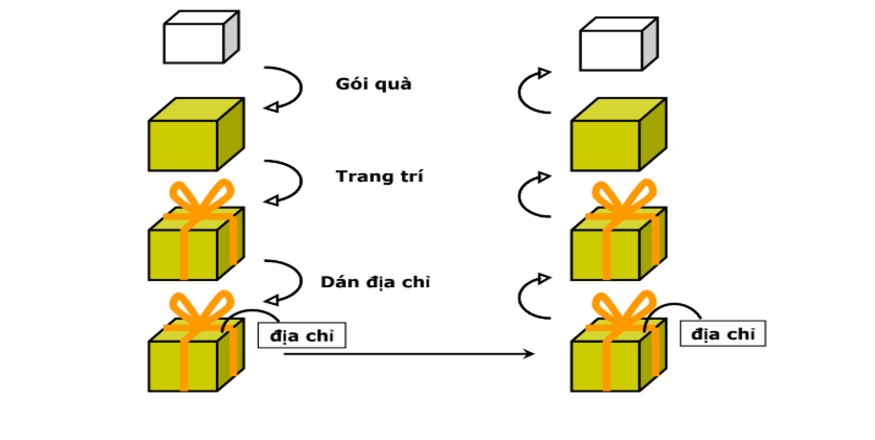
#### ****1. Tổng quan về Socket****

Trong hệ thống mạng máy tính tồn tại những mô hình tham chiếu có kiến trúc phần tầng (OSI, TCP/IP…) nhằm hỗ trợ chức năng trao đôi thông tin giữa các ứng dụng ở nhiều máy tính khác nhau.

#### https://codelearnstorage.s3.amazonaws.com/Media/Default/Users/tria3ltt_40gmail_2Ecom/socketJavaImage/Annotation%202020-08-08%20132406.jpg

Hình 3: Cách thức hoạt động của Socket

Dữ liệu bên gửi sẽ được đóng gói (Encapsulation) từ tầng trên đến tầng cuối là tầng vật lí (Physical Layer), sau đó nhờ tầng vật lí này chuyển dữ liệu đến tầng vật lí máy bên nhận, bên nhận tiến hành giải mã (decapsulation) gói dữ kiện từ tầng dưới lên tầng trên cùng, là tầng ứng dụng (application layer).

Hình 4: Minh họa socket

Ở đây, Socket chính là cửa giao tiếp giữa tầng ứng dụng và tầng giao vận (Transport layer). Nói cách khác, Socket là giao diện do ứng dụng tạo ra trên máy trạm, quản lí bởi hệ điều hành qua đó các ứng dụng có thể gửi/nhận thông điệp đến/từ các ứng  dụng khác. Ở đó, Socket sẽ được ràng buộc với một mã số cổng (Port Number) để giúp tầng giao vận định danh được ứng dụng nhận/gửi thông điệp.

Các bạn có thể thấy ở hình ảnh trên, tầng giao vận có 2 phương thức là TCP (Ttransmission Control Protocol) và UDP (User Datagram Protocol), như vậy socket cơ bản là có 2 loại: Stream Socket sử dụng TCP truyền dòng bytes và Datagram Socket dử dụng UDP truyền gói tin. Với ngôn ngữ lập trình Java, chúng ta được cung cấp 3 loại khác nhau của sockets:

1. Stream Socket (TCP) : Tạo luồng dữ liệu hai chiều, đáng tin cậy, có trình tự và không trùng lặp, dữ liệu chỉ được gửi/nhận khi có đã có liên kết. Dùng với Socket Class của java.
2. Datagram Socket (UDP): Có thể nhận dữ liệu không theo tình tự, trùng lặp. Dùng với DatagramSocket Class.
3. Multicast Socket : cho phép dữ liệu được gửi đến nhiều bên nhận một lúc. Dùng với DatagramSocket Class.

Socket được hỗ trợ trên nhiều ngôn ngữ như C, Java, Pearl,  Python,….

#### Các bước để tạo lên 1 chương trình

* Phía server

Tạo socket với hàm socket các tham số trong đó theo thứ tự là họ giao thức, kiểu socket, kiểu giao thức.

Gán địa chỉ cho socket bind các tham số lần lượt là mô tả socket vừa tạo, con trỏ chỏ đến địa chỉ socket, độ lớn địa chỉ

Chỉ định socket lắng nghe kết nối listen trong đó socket là mô tả socket vừa tạo, số lượng tối đa các kết nối đang chờ

Chờ/chấp nhận kết nối accept lần lượt có các tham số là mô tả socket vừa tạo, con trỏ tới cấu trúc địa chỉ socket của tiến trình kết nối đến, độ lớn cấu trúc địa chỉ

Thiết lập kết nối với máy chủ TCP .

* Phía client

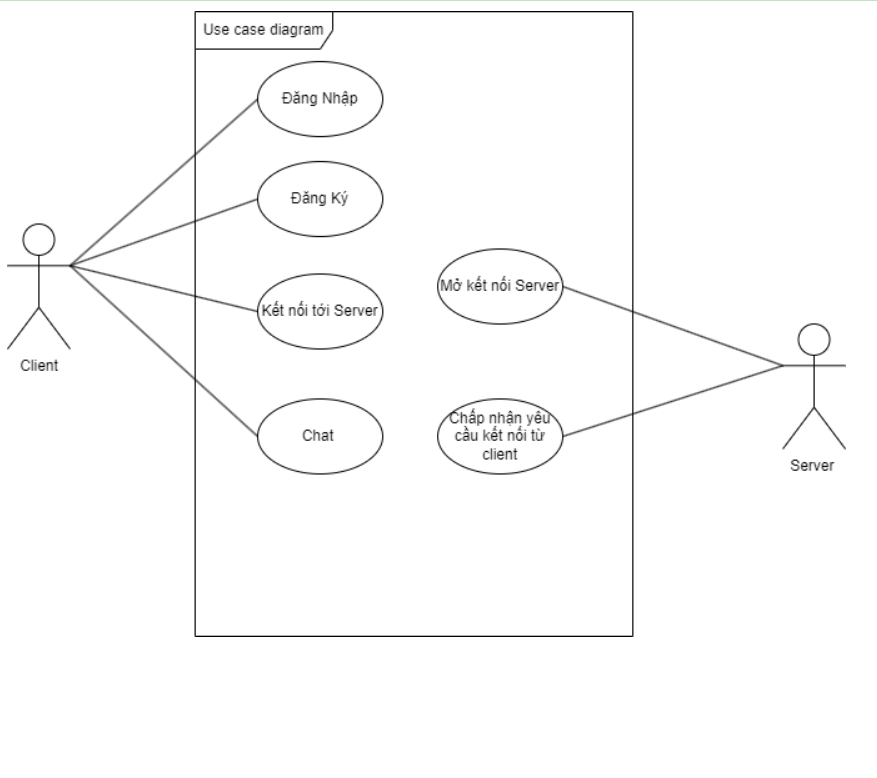
Tạo socket với hàm socket, các tham số trong đó theo thứ tự là họ giao thức, kiểu socket, kiểu giao thức.

Connect tới địa chỉ server với hàm connect

Đọc dữ liệu từ server và ghi vào biến.

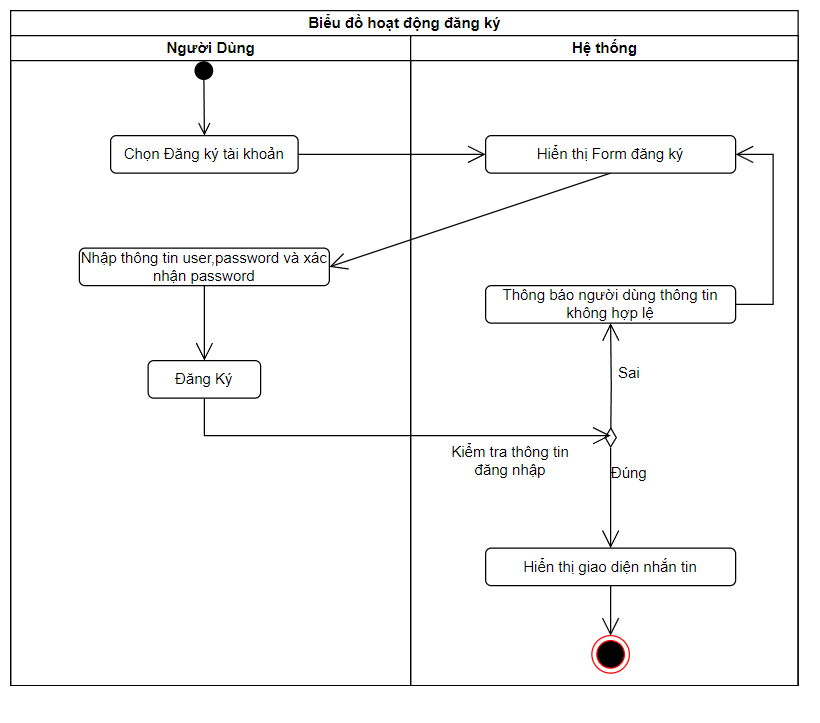
# CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Biểu đồ use case( Use case diagram)



Hình 5: Sơ đồ usecase

## Biểu đồ hoạt động (activity diagram)



Hình 6: Biểu đồ hoạt động đăng ký tài khoản

Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, hàng

Mô tả được tạo tự động

Hình 7: Biểu đồ hoạt động đăng nhập

Ảnh có chứa văn bản, biểu đồ, ảnh chụp màn hình, hàng

Mô tả được tạo tự động

Hình 8: Biểu đồ hoạt động download image

Ảnh có chứa văn bản, biểu đồ, ảnh chụp màn hình, Kế hoạch

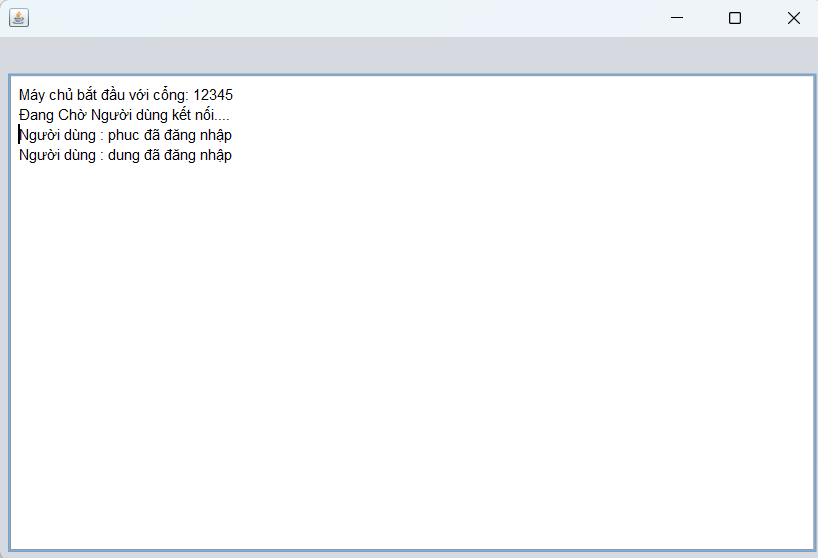
Mô tả được tạo tự động

Hình 9: Biểu đồ hoạt động download File

## Mô tả chức năng hệ thống

# CHƯƠNG 3. TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ

## Giao diện chương trình



Hình 10: Giao diện chính Server

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Hệ điều hành

Mô tả được tạo tự động

Hình 11: Giao diện đăng nhập

Ảnh có chứa ảnh chụp màn hình, văn bản, phần mềm, màn hình

Mô tả được tạo tự động

Hình 12: Giao diện Chat

Ảnh có chứa ảnh chụp màn hình, văn bản, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

Hình 13: Chức năng gửi ảnh

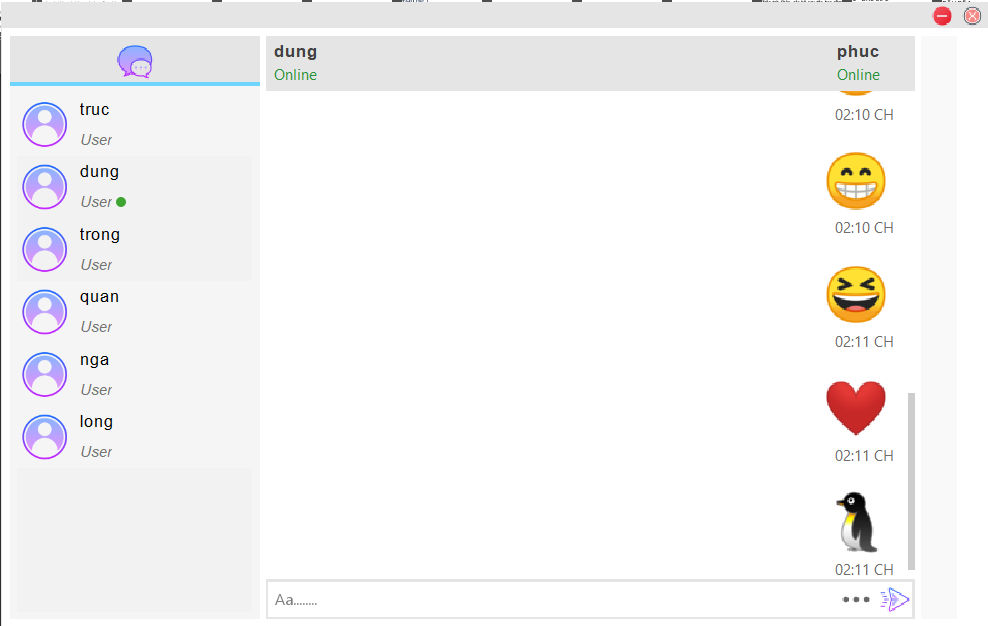
Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Mô tả được tạo tự động

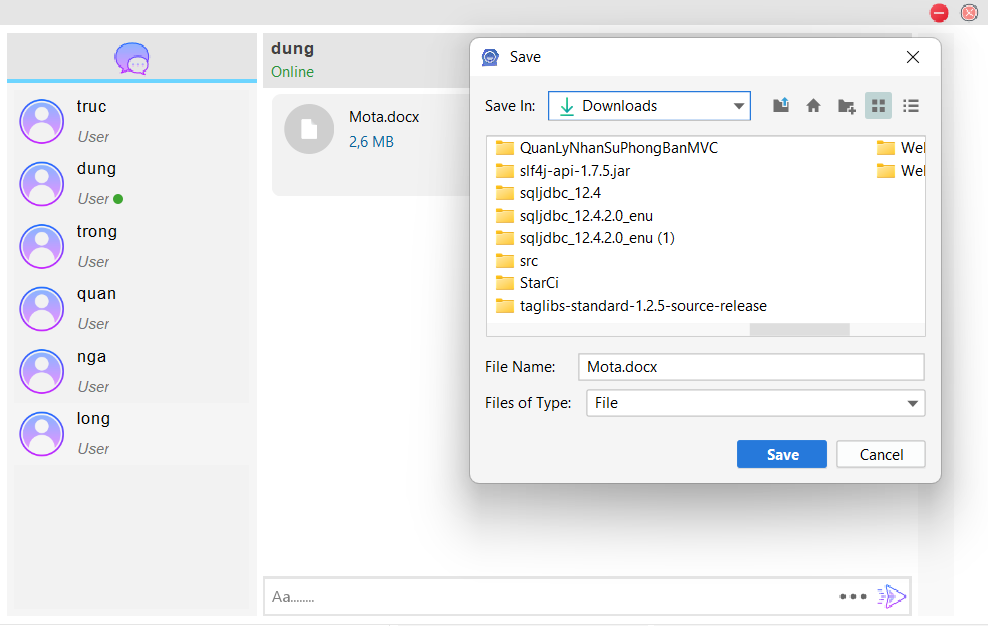
Hình 14: Chức năng gửi file



Ảnh có chứa ảnh chụp màn hình, văn bản, phần mềm, Biểu tượng máy tính

Mô tả được tạo tự động

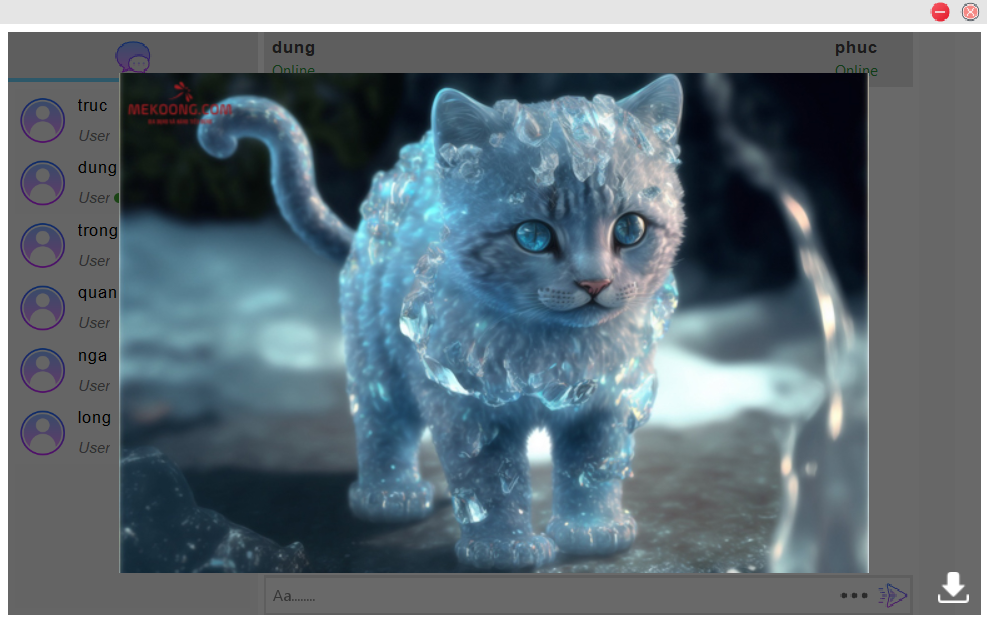
Hình 15:Chức năng gửi emoji

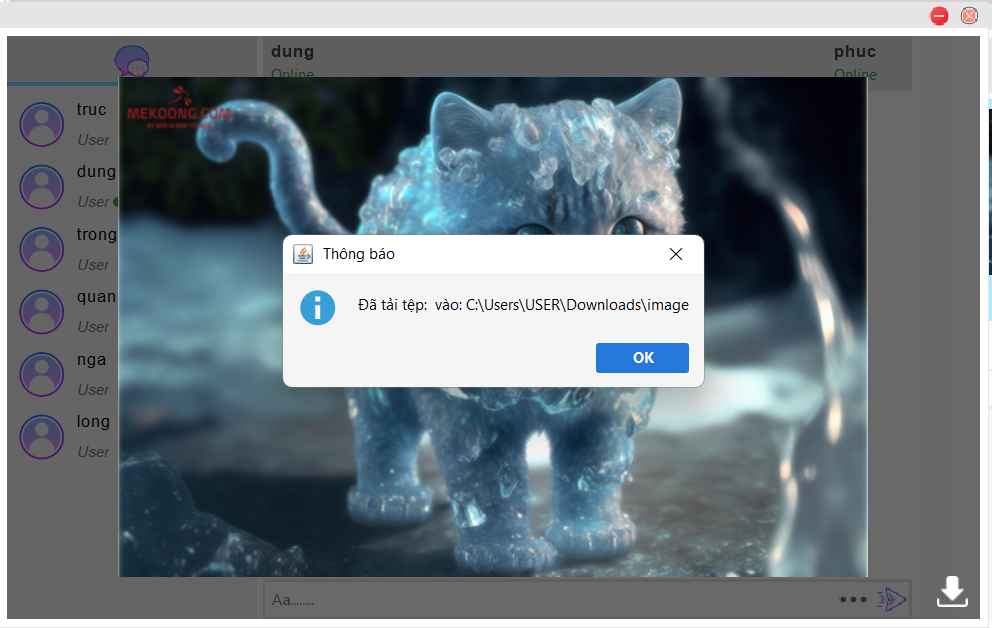


Ảnh có chứa văn bản, phần mềm, Biểu tượng máy tính, Phần mềm đa phương tiện

Mô tả được tạo tự động

Hình 16: Chức năng download File





Hình 7: Chức năng xem và download image

## Đánh giá kết quả

• Giao diện thiết kế dễ nhìn, dễ tiếp cận và sử dụng.

• Các thao tác đơn giản, dễ hiểu.

• Các chức năng làm được:

- Chức năng đăng kí, đăng nhập.

- Chức năng gửi tin nhắn ,gửi file,gửi ảnh,gửi emoji.

- Chức năng năng tải file của người dùng khi người dùng khác gửi file đến .

- Kết nối được nhiều Client cùng một lúc.

# CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 4.1. Kết luận

Trong thời gian tìm hiểu, nghiên cứu cơ sở lý thuyết và triển khai ứng dụng công nghệ, đồ án đã đạt được những kết quả sau:

Về mặt lý thuyết, đã sử dụng được tốt được công nghệ Client-Server, SQL, Socket để xây dựng nên một ứng dụng hoàn chỉnh với giao diện rõ ràng và các chức năng cơ bản.

Về mặt thực tiễn ứng dụng, đồ án đã đạt được hỗ trợ việc gửi ảnh ,cũng như truyền file dễ dàng và nhanh chóng.

Tuy nhiên, đồ án còn tồn tại các vấn đề như sau:

- Tốc độ phản hồi của Server còn chậm, xử lí chậm.

- Tốc độ upload và download file chưa nhanh, phụ thuộc vào tốc độ internet và tốc độ xử lí của máy Client.

## 4.2. Hướng phát triển

Một số hướng phát triển như sau:

-Giao diện đẹp mắt hơn, dễ sử dụng hơn.

- Tăng tốc độ xử lí và phản hồi.

- Nâng cao trải nghiệm người dùng.

TÀI LIỆU THAM KHẢO

[1] Mai Văn Hà, Bài giảng Công Nghệ Web, tài liệu lưu hành nội bộ

[2] Pham Minh Tuấn, Bài giảng Lập Trình Mạng, tài liệu lưu hành nội bộ

[3] <https://vi.wikipedia.org/wiki/FTP>

[4] [Lập Trình Socket Cơ Bản Với TCP/IP Trong Java (codelearn.io)](https://codelearn.io/sharing/lap-trinh-socket-co-voi-tcpip-ava)

[5] [Java Socket Programming - Socket Server, Client example | DigitalOcean](https://www.digitalocean.com/community/tutorials/java-socket-programming-server-client)