

XÂY DỰNG CÁC ỨNG DỤNG TRAO ĐỔI THÔNG TIN GIỮA CÁC MÁY TÍNH SỬ DỤNG GIAO THỨC TCP VÀ UDP

GVHD:
Thầy Bùi Quý Anh

Link Video Demo:

https://drive.google.com/drive/folders/1cx01gJ-SBwEB9EdnpzcbTfo4e2_2JsrP?usp=sharing

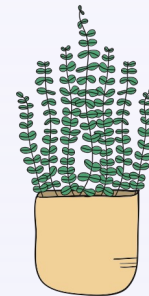
Người thực hiện: Nguyễn Thị Anh Thư



Mục lục

- 1 Cơ sở lý thuyết
- 2 Phân tích yêu cầu
- 3 Chương trình demo
- 4 Mở rộng
- 5 Kết luận

Cơ sở lý thuyết



URLs, URIs Class

- Sự khác nhau giữa URLs và URIs

URL	URI
URL được sử dụng để mô tả danh tính của một mặt hàng.	URI cung cấp một kỹ thuật để xác định danh tính của một mục.
URL liên kết một trang web, một thành phần của trang web hoặc một chương trình trên trang web với sự trợ giúp của các phương thức truy cập như giao thức.	URI được sử dụng để phân biệt tài nguyên này với tài nguyên khác bất kể phương thức được sử dụng.
URL cung cấp thông tin chi tiết về loại giao thức nào sẽ được sử dụng.	URI không chứa đặc tả giao thức.
URL là một loại URI.	URI là siêu tập hợp của URL.
Nó bao gồm giao thức, tên miền, đường dẫn, hàm băm, ...	Nó bao gồm lược đồ, quyền hạn, đường dẫn, truy vấn và nhiều hơn nữa.

Socket Client/ Server

Client Sockets

Khi kết nối được thiết lập, các chủ thể địa phương và từ xa lấy các luồng đầu vào và đầu ra từ socket và sử dụng các luồng đó để gửi dữ liệu cho nhau.

Khi việc truyền dữ liệu hoàn tất, một hoặc cả hai bên sẽ đóng kết nối.

Socket là điểm cuối của liên kết giao tiếp hai chiều giữa hai chương trình đang chạy trên mạng.

Lớp Socket được sử dụng để biểu diễn kết nối giữa máy khách và máy chủ.

Thực hiện bảy hoạt động cơ bản sau:

- Kết nối với một máy từ xa
- Gửi dữ liệu
- Nhận dữ liệu
- Đóng một mối liên hệ
- Liên kết với một cổng
- Nghe xem có nhận được dữ liệu không
- Chấp nhận kết nối từ các máy từ xa trên cổng giới hạn

Chương trình tạo ra một socket mới với một thiết bị tạo dựng. Socket cố gắng kết nối với từ xa.

Socket Option

Vị trí	Tên Option	Kiểu dữ liệu	Get/Set	Mô tả Option
SOL-SOCKET	SO_ACCEPTCON N	BOOL	Get	Socket có lắng nghe không?
	SO_BROADCAST	BOOL	Get/Set	Cho phép phát tin nhắn broadcast trên socket.
	SO_DONTLINGER	BOOL	Get/Set	Cho phép hoặc vô hiệu hóa trả về ngay lập tức từ closeSocket ()
	SO_KEEPALIVE	BOOL	Get/Set	Gửi tin nhắn keep-alive
	SO_LINGER	struct linger	Get/Set	Cho phép hoặc vô hiệu hóa trả về ngay lập tức từ closeSocket ()
	SO_OOBINLINE	BOOL	Get/Set	Out-of-band data nằm trong data stream thông thường
	SO_REUSEADDR	BOOL	Get/Set	Cho phép hoặc vô hiệu hóa việc tái sử dụng một socket bị ràng buộc
	SO_SECURE	DWORD	Get/Set	Cho phép hoặc vô hiệu hóa mã hóa SSL trên socket
	SO_SNDBUF	Int	Get/Set	Kích thước của bộ đệm được phân bổ để gửi dữ liệu
	SO_TYPE	Int	Get	Kiểu socket

Socket Option

Vị trí	Tên Option	Kiểu dữ liệu	Get/Set	Mô tả Option
IPPROTO_TCP	TCP_NODELAY	BOOL	Get/Set	Bật/Tắt thuật toán NAGLE
IPPROTO_IP	IP_MULTICAST_TTL	Int	Get/Set	Thời gian để duy trì một gói phát đa hướng
	IP_MULTICAST_IF	unsigned long	Get/Set	Địa chỉ giao diện phát đa hướng đi
	IP_ADD_MEMBERSHIP	struct ip_mreq	Get/Set	Thêm socket vào một nhóm phát đa hướng
	IP_DROP_MEMBERSHIP	struct	Get/Set	Loại bỏ socket từ một nhóm phát đa hướng

The Object Methods

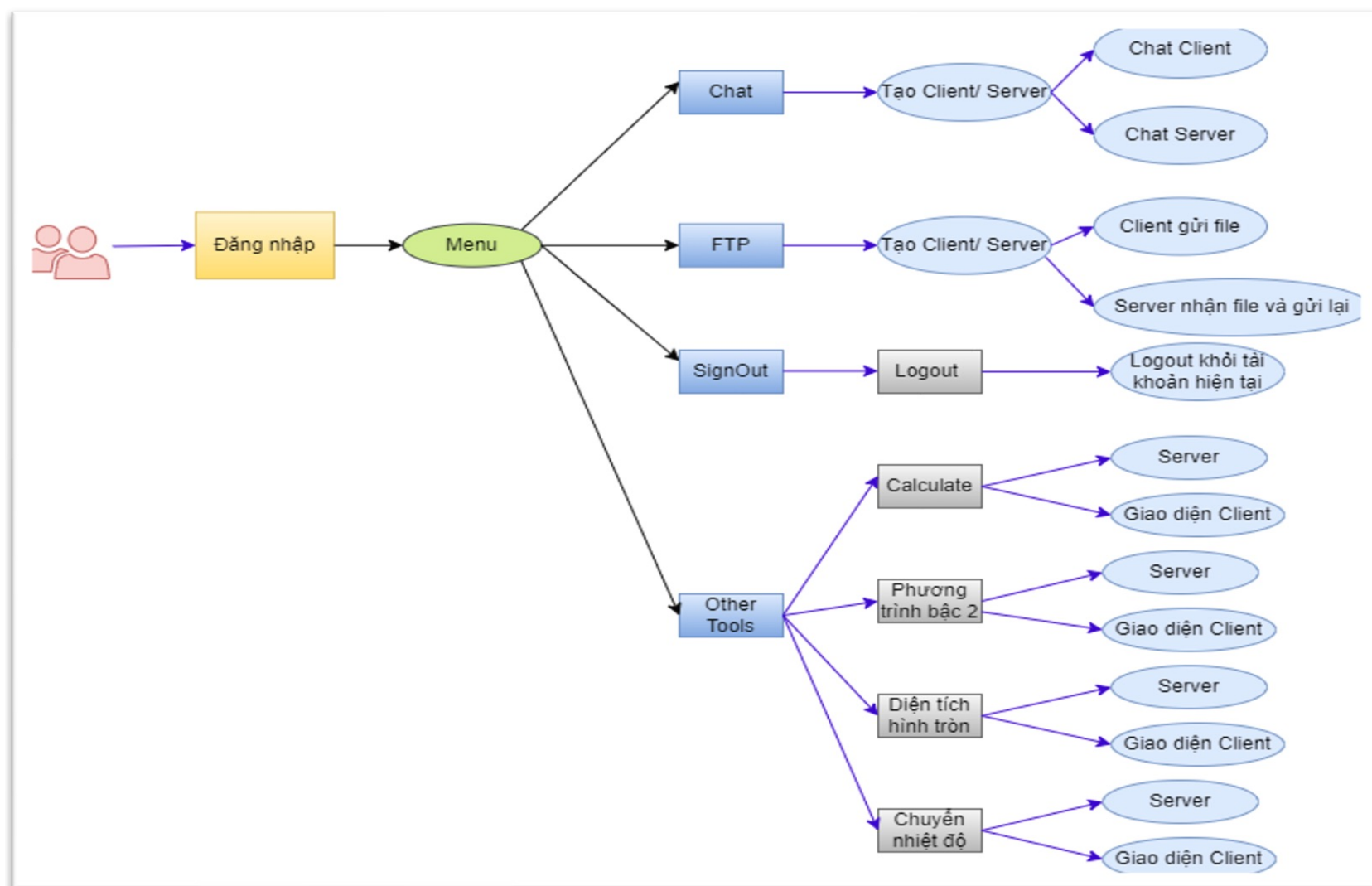
STT	Tên phương thức	Mô tả
1	<code>public Socket accept() throws IOException</code>	Phương thức này chặn cho đến khi hoặc một client kết nối với server trên cổng được chỉ định hoặc socket time-out giả định rằng giá trị time-out được thiết lập theo phương thức <code>setSoTimeout()</code> . Nếu không thì phương thức này chặn vô thời hạn.
2	<code>public InetAddress getInetAddress()</code>	Phương thức này trả về địa chỉ của máy tính khác mà socket này được kết nối.
3	<code>public int getPort()</code>	Trả về cổng socket được liên kết với máy từ xa.
4	<code>public int getLocalPort()</code>	Trả về cổng socket được liên kết với máy cục bộ.
5	<code>Public SocketAddress getRemoteSocketAddress()</code>	Trả về địa chỉ của socket từ xa.

The Object Methods

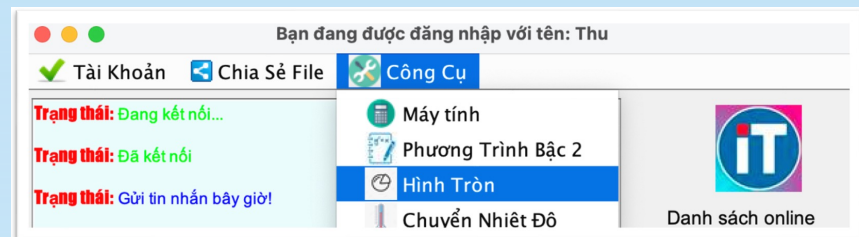
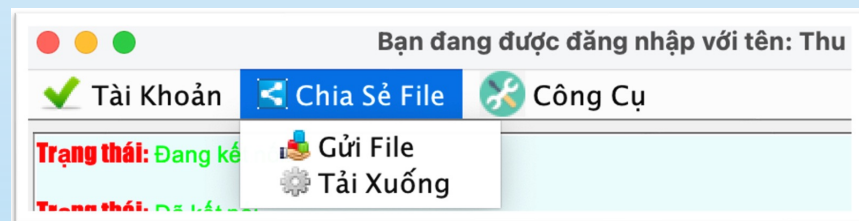
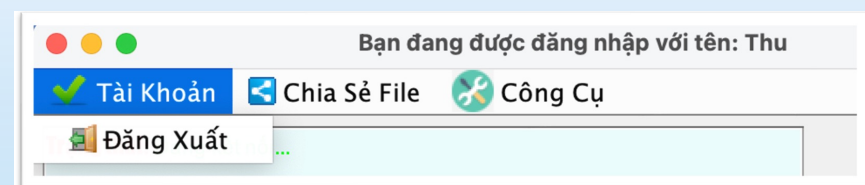
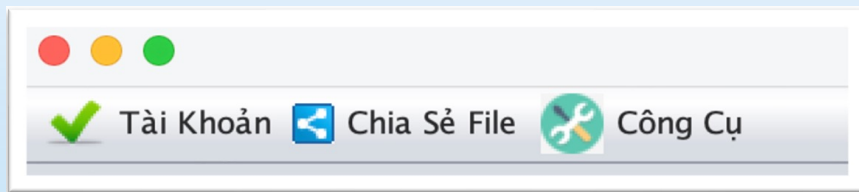
STT	Tên phương thức	Mô tả
6	<code>public InputStream getInputStream() throws IOException</code>	Trả về input stream của socket. Input stream được kết nối với output stream của socket từ xa.
7	<code>public OutputStream getOutputStream() throws IOException</code>	Trả về output stream của socket. Output stream được kết nối với luồng đầu vào của socket từ xa.
8	<code>public void close() throws IOException</code>	Đóng socket, giúp socket object này không còn có khả năng kết nối lại với bất kỳ server nào.
9	<code>public void setSocketTimeot(int timeout)</code>	Đặt giá trị time-out cho server socket trong bao lâu để chờ client accept().

Phân tích yêu cầu

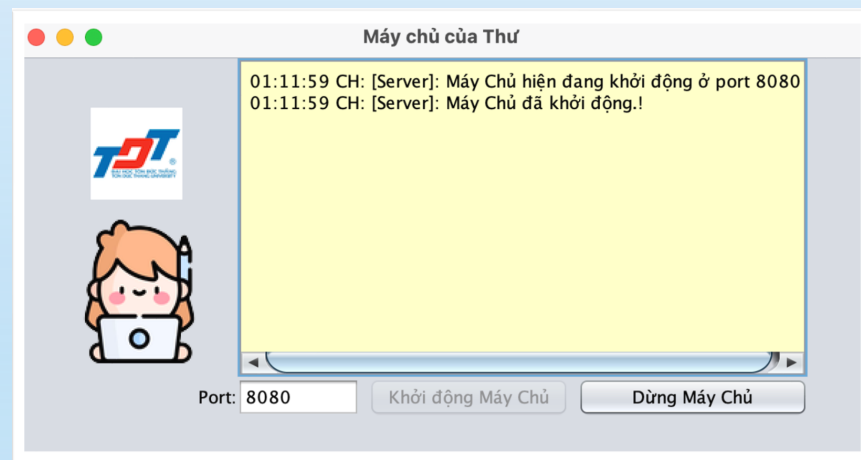




Các chức năng chính



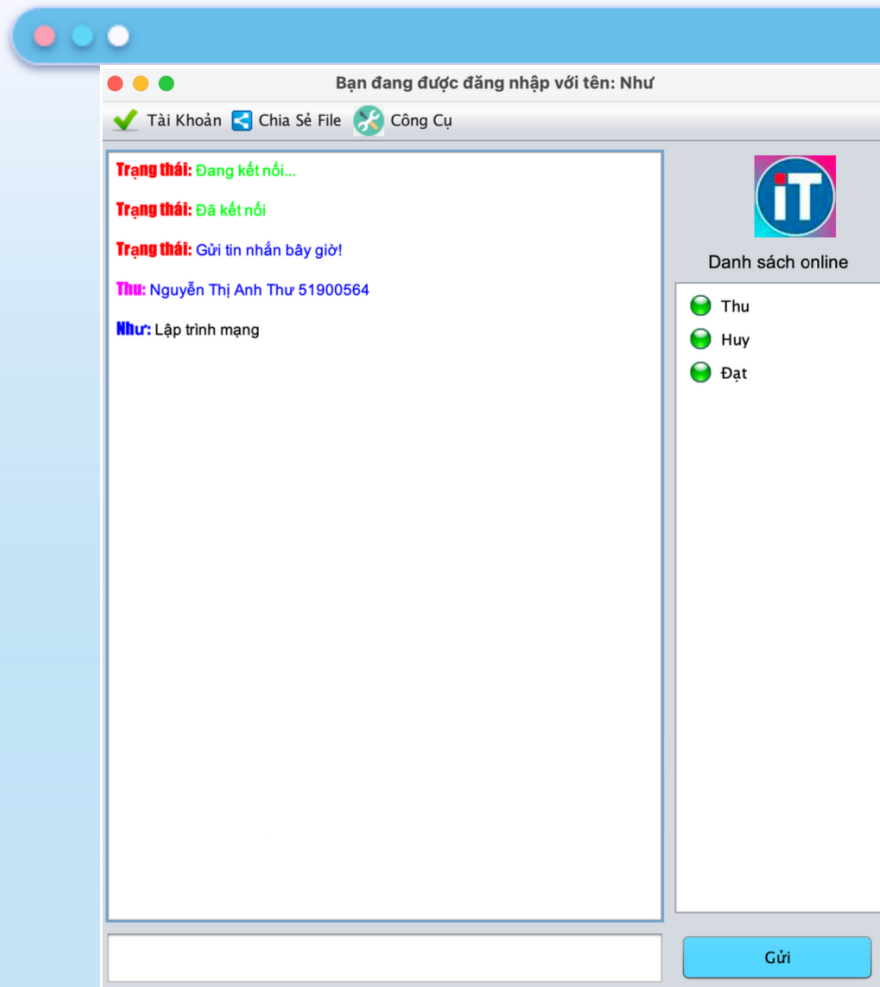
Phần demo



Giao diện máy chủ



Giao diện Client



Giao diện đăng nhập
vào Chat

Mở rộng

Thêm một vài tính năng mới như:

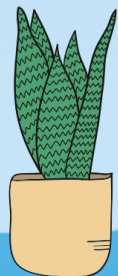
- Video call
- Chụp ảnh
- ...



Kết luận

Nhìn chung, mô hình ứng dụng chat client server cũng đã hoàn thiện. Toàn bộ hệ thống đã được thiết kế trên java swing. Mô hình đáp ứng được các yêu cầu được đưa ra như các client và server có thể chat được với nhau và có các ứng dụng cơ bản.

Các chức năng được thêm: Đăng nhập/ đăng xuất, Chuyển gửi/ nhận file, tính toán số học, tính phương trình bậc 2, tính chu vi và diện tích hình tròn, chuyển đổi độ C và F. Có kèm âm thanh khi thực hiện gửi tin và nhận tin.





Tài liệu tham khảo

- [1]. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun(2015)
<https://arxiv.org/abs/1512.03385>
 - [2]. Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger (2016) -
<https://arxiv.org/abs/1608.06993>
 - [3]. Stackoverflow (2017) <https://stackoverflow.com/questions/48938739/inception-resnet-v2-model-consists-of-how-many-layers>
 - [4]. Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam(2017)
<https://arxiv.org/abs/1704.04861>
- 



Thanks for watching!