TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN MÔN WEB NÂNG CAO**

**WEB SOCKET**

*Người hướng dẫn*: **Ths. MAI VĂN MẠNH**

*Người thực hiện*: **NGUYỄN MINH NHỰT – 518H0545**

**NGUYỄN BRANDON TUẤN – 518H0074**

Lớp **: 18H50203**

Khoá  **: 22**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN MÔN WEB NÂNG CAO**

**WEB SOCKET**

*Người hướng dẫn*: **Ths. MAI VĂN MẠNH**

*Người thực hiện*: **NGUYỄN MINH NHỰT – 518H0545**

**NGUYỄN BRANDON TUẤN – 518H0074**

Lớp **: 18H50203**

Khoá  **: 22**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

# LỜI CẢM ƠN

Lời đầu tiên, chúng em xin được gửi lời cảm ơn chân thành đến với Khoa CNTT. Chúng em xin chân thành cảm ơn sự hỗ trợ, giúp đỡ trong quá trình thực hiện bài báo cáo môn “Web nâng cao” này.

Chúng em xin chân thành cảm ơn thầy Mai Văn Mạnh – thầy đã trực tiếp hướng dẫn và chỉ bảo chúng chúng em hoàn thành bài báo cáo này.

Chúng em xin chân thành cảm ơn những người bạn đang học tập và làm việc tại trường Đại học Tôn Đức Thắng và gia đình đã đồng hành, động viên, tạo điều kiện và giúp đỡ chúng em trong suốt quá trình này.

Do năng lực thực tế của chúng em còn non yếu nên chúng em đảm bảo còn nhiều thiếu sót, mong quý thầy/cô bỏ qua. Đồng thời chúng em mong nhận được nhiều ý kiến đóng góp từ nhiều nguồn khác nữa, điều này sẽ giúp chúng em tích lũy thêm kinh nghiệm để hoàn thành bài báo cáo tốt nghiệp sắp tới đạt kết quả tốt hơn.

# ĐỒ ÁN ĐƯỢC HOÀN THÀNH

# TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Em xin cam đoan đây là sản phẩm đồ án của riêng chúng em và được sự hướng dẫn của ThS Mai Văn Mạnh. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào chúng em xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do em gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 01 tháng 05 năm 2021*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

**

*Nguyễn Minh Nhựt*

**

*Nguyễn Brandon Tuấn*

# PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

# TÓM TẮT

# MỤC LỤC

[LỜI CẢM ƠN i](#_Toc71479165)

[ĐỒ ÁN ĐƯỢC HOÀN THÀNH ii](#_Toc71479166)

[TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG ii](#_Toc71479167)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iv](#_Toc71479168)

[TÓM TẮT vi](#_Toc71479169)

[MỤC LỤC 1](#_Toc71479170)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 2](#_Toc71479171)

[CHƯƠNG 1 – GIỚI THỆU ĐỀ TÀI 3](#_Toc71479172)

[1.1 Đặt vấn đề và giới thiệu 3](#_Toc71479173)

[1.2 Lý do lựa chọn đề tài 5](#_Toc71479174)

[1.3 Ý nghĩa của đề tài 5](#_Toc71479175)

[CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT WEBSOCKET 6](#_Toc71479176)

[2.1 Tổng quan chương 6](#_Toc71479177)

[2.2 Network Socket là gì? 6](#_Toc71479178)

[2.2.1 Socket là gì? 6](#_Toc71479179)

[2.2.2 Phân loại socket: 9](#_Toc71479180)

[2.2 Web Socket là gì? 16](#_Toc71479181)

[2.2.1 Giới thiệu sơ lược 16](#_Toc71479182)

[2.2.2 Lịch sử phát triển của Web Socket 16](#_Toc71479183)

[2.3 Cách thức hoạt động của WebSocket 17](#_Toc71479184)

[2.3.1 Truyền toàn song công (Full-duplex) 17](#_Toc71479185)

[2.3.2 HTTP Long Polling là gì? 18](#_Toc71479186)

[CHƯƠNG 3 – SƠ LƯỢC VỀ THƯ VIỆN VÀ NGÔN NGỮ THỰC NGHIỆM 20](#_Toc71479187)

[CHƯƠNG 4 – PHÂN TÍCH THIẾT KẾ VÀ THỰC NGHIỆM 21](#_Toc71479188)

[CHƯƠNG 5 – TỔNG KẾT 22](#_Toc71479189)

[TÀI LIỆU THAM KHẢO 23](#_Toc71479190)

# DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH**

**No table of figures entries found.**

**DANH MỤC BẢNG**

**No table of figures entries found.**

# CHƯƠNG 1 – GIỚI THỆU ĐỀ TÀI

## Đặt vấn đề và giới thiệu

Như đã đề cập ở trên, nhóm chúng em sẽ nghiên cứu về Web Socket, chúng ta sẽ dùng thuật ngữ này xuyên suốt bài báo cáo này vì chưa có thuật ngữ tiếng Việt nào phù hợp để phiên dịch cho phù hợp với tên gọi.

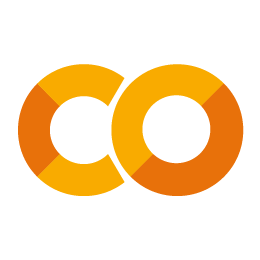
Như chúng ta cũng đã biết, hiện nay trên thị trường đã tồn tại rất nhiều những công cụ giúp cho đời sống cũng như công việc của chúng ta trở nên dễ dàng và tốt hơn, ví dụ như:

* *Multiplayer gaming* – trò chơi nhiều người chơi trực tuyến, giúp chúng ta giải trí sau những giờ làm, học tập căng thẳng và gặp gỡ bạn bè.



#### Hình 1: iChess là ví dụ cho trò chơi trực tuyến nhiều người chơi

* *Collaborative editing/coding* – đây là những trang web giúp cho đội nhóm có thể hoạt động, chia sẻ và thực hiện những thao tác trên 1 nguồn chung mà ở đây mọi thứ được cập nhật thường xuyên, ví dụ như: Google Colab, Google Sheet, Doc và những sản phẩm khác nữa…



#### Hình 2: Google Colab – công cụ soạn thảo đa tuyến

* *Multimedia chat* – đây là 1 ứng dụng phổ biến và thường xuyên được bắt gặp nhất. Một trang web giúp chúng ta trò chuyện trực tuyến, hoàn toàn miễn phí là một phát mình không hề nhỏ, ví dụ như: Messenger, Discord, … Hơn nữa, trong bài làm lần này, chúng em sẽ viết một ứng dụng chat trực tuyến sử dụng giao thức Web socket.



#### Hình 3: Facebook Messenger – một multimedia chat online không còn xa lạ

Điểm chung dễ nhận thấy mà các ứng dụng trên đưa ra là tất cả mọi hoạt động đều được cập nhật thường xuyên, khi có một tín hiệu gửi đi sẽ có tín hiệu nhận về, có sửa đổi thì sẽ có cập nhật. Đây là một đặc thù mà Web Socket đem lại cho chúng ta và cụ thể chúng như thế nào thì ở ***CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT***, chúng em sẽ cố gắng trình bày cụ thể và rõ ràng hơn.

## Lý do lựa chọn đề tài

Chúng em chọn đề tài này là vì:

1. Đây là một chủ đề phổ biến, một trong những kĩ năng và kiến thức quan trọng mà ai cũng nên có khi học về lập trình Web và ứng dụng nói chung và Node.js nói riêng
2. Cụ thể hơn nữa, đây là một chủ đề mà nhóm chúng em chưa tìm hiểu trước đây nên đây sẽ là cơ hội để chúng em nghiên cứu

## Ý nghĩa của đề tài

Với đề tài này, chúng em có thể biết thêm về một giao thức mới mà với giao thức này chúng ta có thể thực hiện được rất nhiều công tác bổ ích như chúng em đã đề cập như trên. Hiện nay, với xu hướng toàn cầu hóa cùng với dịch bệnh COVID 19, tất cả mọi hoạt động sẽ được diễn ra trên nền tảng số, như là: nhắn tin trực tuyến, các dịch vụ lớp học, cuộc hợp trực tuyến …

# CHƯƠNG 2 – CƠ SỞ LÝ THUYẾT WEBSOCKET

## 2.1 Tổng quan chương

Trong chương này, chúng em sẽ trình bày các kiến thức mà chúng em đã tìm hiểu trong suốt thời gian làm báo cáo. Theo đó, chúng ta sẽ có các mục sau

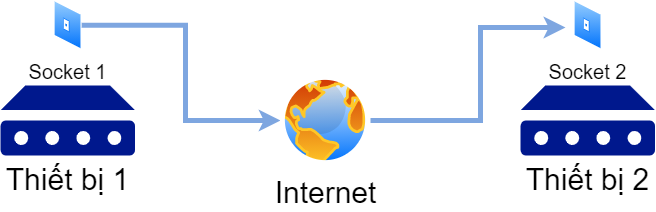
* 1. Network Socket là gì?
  2. Web Socket là gì?
  3. Cách thức hoạt động của Web Socket

## 2.2 Network Socket là gì?

### Socket là gì?

Socket là điểm cuối (end-point) của liên kết giao tiếp 2 chiều (2-way communication) giữa 2 chương trình chạy trên mạng. Tức, 1 socket được sử dụng để cho phép 1 process giao tiếp với 1 process khác.

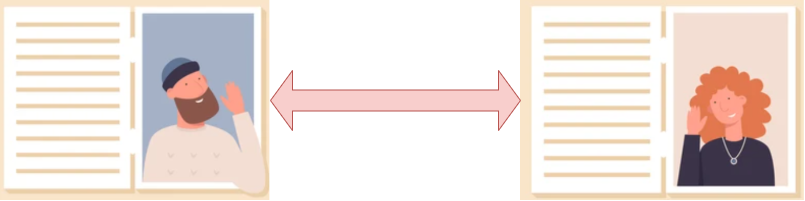
Socket chủ yếu được gắn với 1cổng mạng để chou lớp TCP có thể định dạng chương trình mà dữ liệu sẽ phải truyền tới. Trong đó, điểm cuối đó là tập hợp địa chỉ IP và số cổng mạng (port number).



#### Hình 4: Mô hình dẫn truyền dữ liệu qua Socket

Để trình bày một cách dễ hiểu hơn thì chúng ta sẽ sử dụng ví dụ như sau:

Tưởng tượng rằng mình đang ở trong nhà khi phải giãn cách xã hội và không được phép ra ngoài. Một ngày nọ, bạn quyết định đi ra khung cửa sổ nhà mình dể ngắm cảnh nhưng hàng xóm của bạn cũng xuất hiện ở khung cửa sổ của cô ấy thế là bạn và hàng xóm đã vẫy tay chào nhau rồi sau đó trò chuyện một lúc.



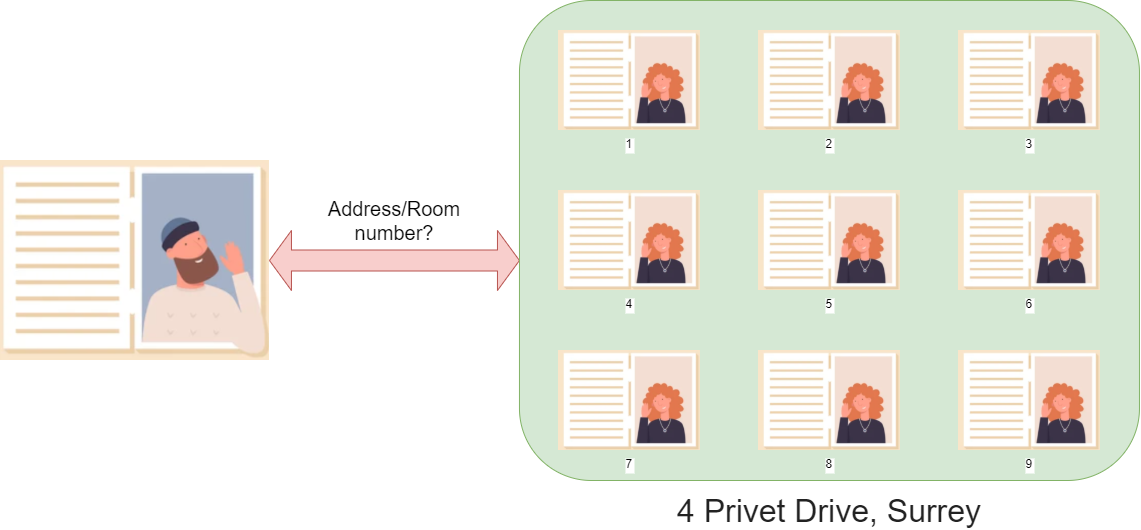
#### Hình 5: Việc giao tiếp qua cửa sổ

Qua ví dụ trên, chúng ta có thể phân so sánh như sau:

* Ngôi nhà bạn và hàng xóm = Máy tính nguồn và máy tính đích
* Khung cửa sổ = socket – là điểm cuối mà chúng ta có thể gửi đi những thông tin
* Cuộc trò chuyện = Data

Như đã đề cập ở trên, còn việc socket là tập hợp của địa chỉ IP và số cổng mạng sẽ được hiểu như thế nào?

Chúng ta sẽ dùng lại mẫu truyện trên để làm ví dụ nhưng nâng cấp hơn, đó là khi căn nhà bây giờ có n phòng và n thành viên trong gia đình sống trong đó. Tất cả các phòng đều có cửa sổ và số phòng (từ 1 đến n).

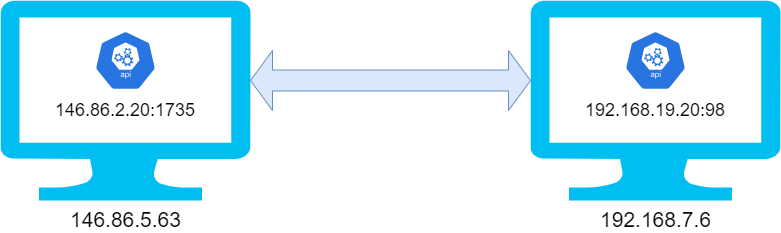


#### Hình 6: Việc giao tiếp bây giờ đòi hỏi địa chỉ nhà và số phòng

Khi này đây bạn cần phải biết 2 thứ để có thể giao tiếp với người hàng xóm mà mình mong muốn đó là số phòng, ngoài ra còn biết cả địa chỉ nhà để chắc chắn rằng mình đang nói chuyện với đúng nhà.

Cũng giống như máy tính vậy, vì bây giờ bộ xử lý hiện nay đã là đa lõi nên việc nhiều chương trình được thực thi là dễ hiểu. Mỗi socket phải được gắn với 1 cổng – thứ sẽ đại diện cho chương trình. Như vậy, Tầng mạng – Network layer mới có thể hiểu nên gửi data đi đến đâu?

Vì có các cổng, chúng ta có thể biết rằng chương trình nào sẽ nhận được data nhưng trong 1 mạng, chúng ta dễ nhận thấy có rất nhiều máy tính khác nhau nên việc biết được địa chỉ cụ thể của máy mà ta đang giao tiếp là điều cần thiết.



#### Hình 7: Mô hình đầy đủ Network Socket

### Phân loại socket:

Hiện tại chúng ta có thể phân socket thành 3 loại phổ biến sau: Datagram socket, Stream socket và Raw socket.

Ngoài ra chúng ta còn có các socket loại khác sẽ được sử dụng

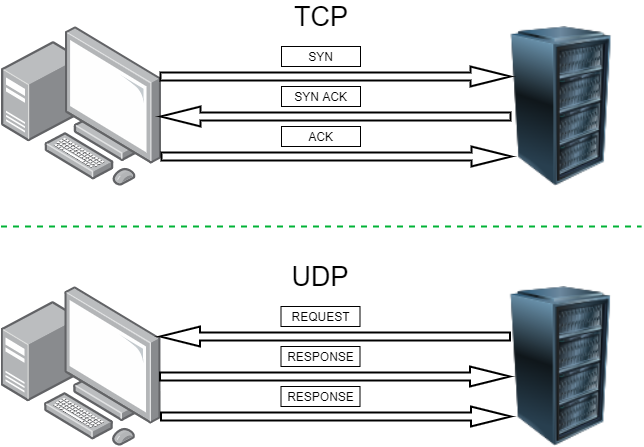
#### **2.2.2.1 Datagram Socket**

Loại đầu tiên, **Datagram socket**, ngoài ra ta có thể gọi nó là socket vô hướng kết nối, dựa trên giao thức UDP (User Datagram Protocol) việc truyền dữ liệu không yêu cầu có sự thiết lập kết nối giữa 2 process. Tức là nó cung cấp connection-less point cho việc gửi và nhận packets.

Trước hết, chúng ta hãy tìm hiểu sơ lược về giao thức mạng UDP – User Datagram Protocol.

UDP hay Giao thức dữ liệu người dùng là 1 giao thức dùng để thay thế cho giao thức TCP. Chủ yếu giao thức này để sử dụng cho việc thiết lập kết nối có độ trễ thấp và khả năng chịu mất mát giữa các ứng dụng trên internet.

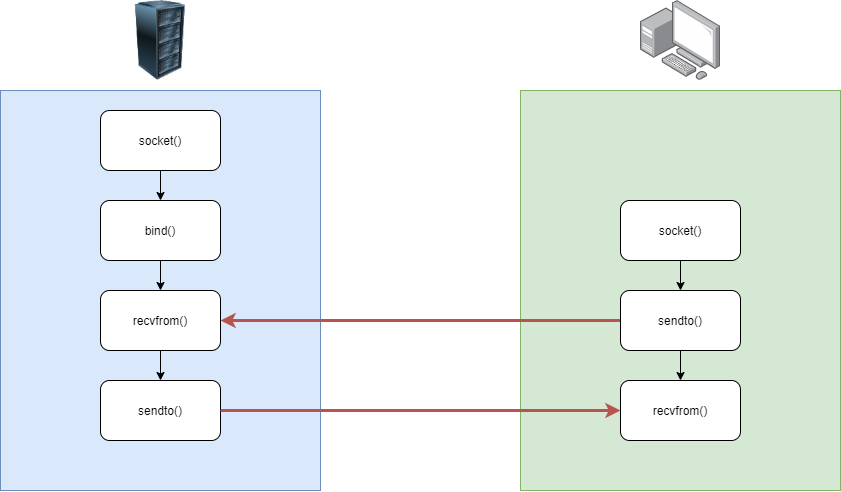
Cả UDP và TCP đều chạy trên IP và đôi khi được gọi là UDP / IP hoặc TCP / IP, tuy nhiên, có sự khác biệt quan trọng giữa hai giao thức này là UDP cho phép giao tiếp quá trình với quá trình, trong khi TCP hỗ trợ giao tiếp từ máy chủ đến máy chủ lưu trữ.



##### Hình 8: Mô hình hoạt động của UDP

Ngược lại, UDP được coi là một giao thức không kết nối vì nó không yêu cầu thiết lập mạch ảo trước khi bất kỳ quá trình truyền dữ liệu nào xảy ra. Giao thức truyền thông chỉ gửi các gói tin, có nghĩa là nó có chi phí và độ trễ băng thông thấp hơn nhiều. Với UDP, các gói có thể đi theo các con đường khác nhau giữa người gửi và người nhận, và kết quả là một số gói có thể bị mất hoặc nhận không đúng thứ tự.

Kế thừa các đặc điểm trên nên Datagram Socket không yêu cầu thiết lập kết nối và không có những cơ chế phức tạp nên tốc độ giao thức khá nhanh, thuận tiện cho các ứng dụng truyền dữ liệu nhanh như chat, game online…



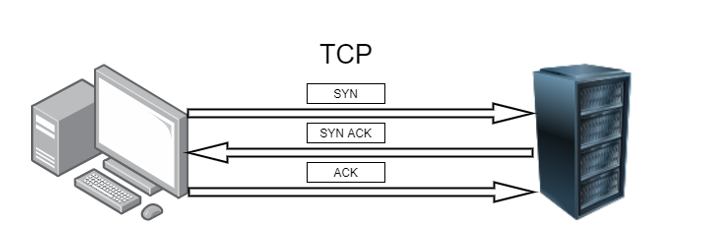
##### Hình 9: Mô hình dẫn truyền data của Datagram Socket

#### **2.2.2.2 Stream Socket**

Loại thứ 2, **Stream Socket**, dựa trên giao thức TCP (Tranmission Control Protocol), stream socket thiết lập giao tiếp 2 chiều theo mô hình client và server. Được gọi là socket hướng kết nối.

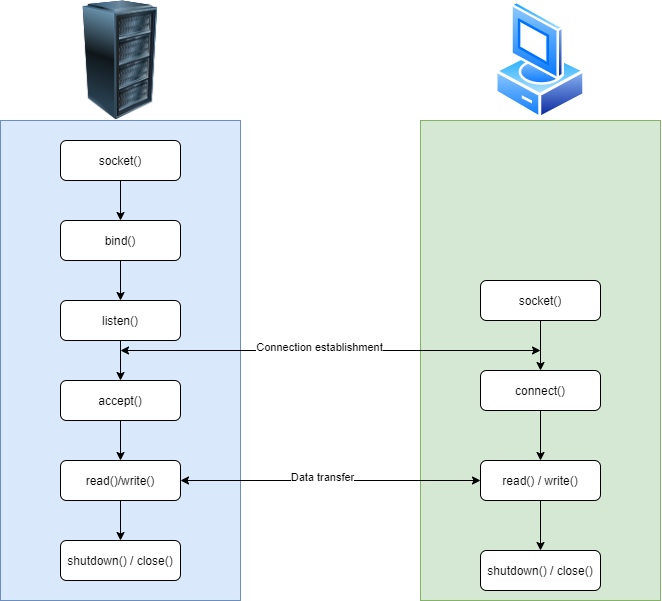
Như trên, để hiểu được cách thức dẫn truyền data của Stream Socket, hãy tìm hiểu sơ lược giao thức TCP.

TCP - Tranmission Control Protocol hay Giao thức điều khiển truyền vận. Các ứng dụng trên các máy chủ được nối mạng có thể tạo các "kết nối" với nhau, mà qua đó chúng có thể trao đổi dữ liệu hoặc các gói tin. Ngược lại với UDP, giao thức này đảm bảo chuyển giao dữ liệu tới nơi nhận một cách đáng tin cậy và đúng thứ tự. TCP còn phân biệt giữa dữ liệu của nhiều ứng dụng chạy trên cùng 1 máy chủ.



##### Hình 10: Mô hình hoạt động giao thức TCP

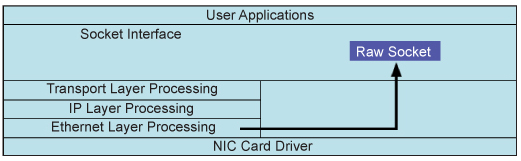
Kế thừa các tính chất trên của TCP mà Stream Socket được xem là giao thức này đảm bảo dữ liệu được truyền đến nơi nhận một cách đáng tin cậy, đúng tuần tự nhờ vào cơ chế quản lý luồng lưu thông trên mạng và cơ chế chống tắc nghẽn.



##### Hình 11: Mô hình giao tiếp server – client định hướng kết nối bằng stream socket

Cách dẫn truyền dữ liệu có phần khác so với Datagram socket nhưng chúng đều giống nhau ở chỗ là chúng đều phải đặt cho socket vừa tạo 1 cái tên thông qua lệnh *bind()* và sau đó hiển thị số cổng mà server muốn truyền dữ liệu qua.

#### **2.2.2.3 Raw Socket**



##### Hình 12: Mô hình hoạt động của Raw Socket (Nguồn: opensourceforu.com)

Các loại socket khác như Datagram Socket hay là Stream socket, sẽ chỉ nhận data từ tầng vận chuyển – Transport Layer và chúng không chứa thông tin header mà chỉ có thông tin cần chuyển. Điều này cho thấy không có thông tin về địa chỉ IP và địa chỉ MAC.

Còn Raw socket, mục đích dẫn truyền của nó hoàn toàn khác. Nó cho phép chương trình có thể truy cập tới những giao thức cấp thấp hơn như TCP, UDP, IP hay ICMP, khi này, Raw socket nhận những gói tin thuần mà chưa qua giải nén. Ngoài ra, chúng ta cũng không cần phải cung cấp địa chỉ IP hay số cổng mạng cho Raw Socket.

## 2.2 Web Socket là gì?

### 2.2.1 Giới thiệu sơ lược

Nếu chúng ta đã nghe nói về HTML5 và các tính năng nổi trội của nó. Hai trong số các tính năng mới của HTML5 ứng dụng trực tiếp cho công nghệ web thời gian thực và giao tiếp client – server.

Chúng ta có thể dùng Server-Sent Events hay là EventSource API cho HTTP Streaming nhưng chúng ta còn 1 cách khác, đó là WebSocket.



##### Hình 13: Logo của HTML5

Như cũng đã đề cập ở *CHƯƠNG 1 – GIỚI THỆU ĐỀ TÀI*, đôi lúc, các ứng dụng như Messenger hay iChess sẽ cần đến sự giao tiếp 2 chiều (2-way communication)từ cả 2 phía client và server.

Chúng ta cũng đã đọc qua các giao thức mạng ở phần trên thì giờ đây WebSocket là 1 giao thức mạng và nó ứng dụng giao thức TCP để khởi tạo giao tiếp dạng truyền toàn song công (full-duplex).

### 2.2.2 Lịch sử phát triển của Web Socket

WebSocket lần đầu tiên được giới thiệu với tên là TCPConnection, là một đặc tả của HTML5.

Vào tháng 6 năm 2008, một chuỗi các tham luận được khởi xướng bởi Michael Carter



#### Hình 13: Chân dung Michael Carter

Tên "WebSocket" được đặt bởi Ian Hickson và Michael Carter ngay sau sự hợp tác phát triển ứng dụng chatroom #whatwg IRC, và sau đó được Ian Hickson đưa vào đặc tả HTML5. Vào tháng 12 năm 2009, Google Chrome 4 là trình duyệt đầu tiên có thể đáp ứng mọi tiêu chuẩn của WebSocket và được đặt bằng mặc định. Sau đó, công tác phát triển được chuyển từ nhóm W3C và WHATWG sang IETF vào tháng 2 năm 2010 và là tác giả của hai bản sửa đổi dưới thời Ian Hickson.

Sau khi giao thức được vận chuyển và được kích hoạt theo mặc định cho nhiều trình duyệt hơn, WebSocket được chuẩn hóa bởi IETF (RFC 6455) vào năm 2011 và WebSocket API đang được chuẩn hóa được chuẩn hóa bởi W3C.

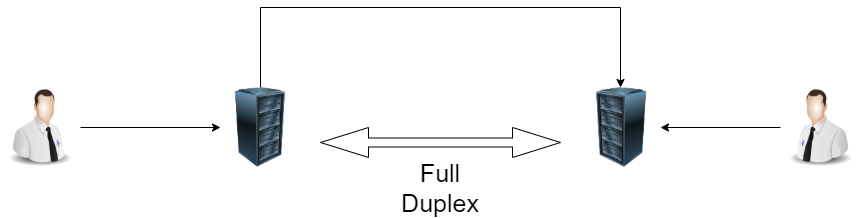
## 2.3 Cách thức hoạt động của WebSocket

Như đã đề cập ở trên, thật là khó hiểu nếu đi một mạch vào cách thức hoạt động của giao thức này. Vậy, chúng ta nên hiểu thế nào là giao tiếp dạng truyền toàn song công (full-duplex), HTTP long polling và AJAX polling.

### 2.3.1 Truyền toàn song công (Full-duplex)

Đi vào khái niệm đầu tiên là truyền toàn song công – full duplex.

Trên một môi trường truyền dẫn (ví dụ trên một sợi cáp quang) thông tin lan truyền giữa các thiết bị mạng có thể được thực hiện theo nhiều dạng thức khác nhau nhưng trong trường hợp môi trường truyền và các thiết bị mạng có thể hoạt động song song cùng lúc để quá trình T1 và T2 xảy ra đồng thời ta có dạng thức truyền toàn song công (Full duplex).



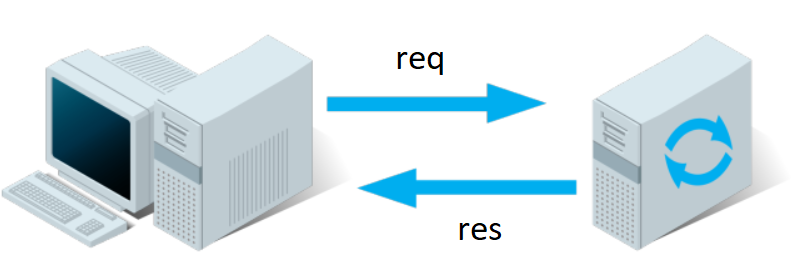
#### Hình 14: Minh họa truyền toàn song công

Truyền toàn song công – full duplex cho phép dữ liệu truyền đồng thời trên cả hai đường. Một thiết bị có thể đồng thời vừa Phát lại vừa Thu tín hiệu. Các modem máy tính, bộ chuyển đổi quang điện đều hoạt động theo phương thức này, mạng LAN sử dụng toàn thiết bị tập trung lớp 2 hoặc 2 máy tính kết nối trực tiếp với nhau có thể sử dụng dạng thức fullduplex.

Hãy nghĩ về việc chúng ta gọi điện cho nhau, cả 2 người nói và nghe đều có thể giao tiếp cùng lúc.

### 2.3.2 HTTP Long Polling là gì?

Trước hết, Regular HTTP Polling là gì? Chúng ta hiểu đơn giản, nó chỉ là việc client gửi 1 request lên cho server và sau đó server sẽ xử lý và trả về 1 response về cho client sau đó đợi request tiếp theo để xử lý.



#### Hình 15: Minh họa regular HTTP Polling

Với tính thụ động của mô hình này. HTTP Long Polling ra đời.

WebSocket được sinh ra là để thay thế cho kĩ thuật long polling. Trong trường hợp này, client gửi một yêu cầu và chờ đến khi phản hồi được nhận về. Lợi ích của cách tiếp cận này là kết nối luôn mở và khi nào dữ liệu sẵn có, nó sẽ gửi đi ngay lập tức. Tuy nhiên, kết nối có thể bị timed-out và cần phải thiết lập lại để tiếp tục quá trình.

# CHƯƠNG 3 – SƠ LƯỢC VỀ THƯ VIỆN VÀ NGÔN NGỮ THỰC NGHIỆM

# CHƯƠNG 4 – PHÂN TÍCH THIẾT KẾ VÀ THỰC NGHIỆM

# CHƯƠNG 5 – TỔNG KẾT

# TÀI LIỆU THAM KHẢO

**Tiếng Việt**

1. Blog TopDev - [*Socket là gì? WebSocket là gì? Hiểu hơn về Websocket*](https://topdev.vn/blog/socket-la-gi-websocket-la-gi/#:~:text=Websocket%20l%C3%A0%20giao%20th%E1%BB%A9c%20h%E1%BB%97,k%C3%AC%20lo%E1%BA%A1i%20%E1%BB%A9ng%20d%E1%BB%A5ng%20n%C3%A0o.)
2. Wikipedia tiếng Việt – [*TCP*](https://vi.wikipedia.org/wiki/TCP#:~:text=TCP%20(Transmission%20Control%20Protocol%20%2D%20%22,li%E1%BB%87u%20ho%E1%BA%B7c%20c%C3%A1c%20g%C3%B3i%20tin.)
3. Rohit Rai – [*Socket.IO real-time Web Application Development*.](https://www.packtpub.com/product/socket-io-real-time-web-application-development/9781782160786)
4. InfoWorld – [*9 killer uses for Web Sockets*](https://www.infoworld.com/article/2071232/9-killer-uses-for-websockets.html)
5. Wikipedia – [*Network Socket*](https://en.wikipedia.org/wiki/Network_socket)
6. Rahul Dwivedi – [*What is network socket*](https://dev.to/rdrahul/what-is-a-network-socket-5d4)
7. Prof. Douglas Comer - [*Fundamentals Of Computer Networking And Internetworking*](https://netbook.cs.purdue.edu/Lecture_Notes.pdf)
8. Oracle Documentation - [*Connection-Oriented Communication Using Stream Sockets*](https://docs.oracle.com/cd/E19120-01/open.solaris/817-4415/sockets-12/index.html)
9. Wikipedia – [*Web Socket*](https://en.wikipedia.org/wiki/WebSocket)
10. Scott Sanderson - [*A Simple Start to jQuery, JavaScript, and Html5 for Beginners*](https://pdfhost.io/v/d~XM9Phjv_JavaScript__A_Simple_Start_to_jQuery_JavaScript_and_HTML5_PDFDrive_pdf.pdf)
11. Vanessa Wang, Frank Salim, Peter Moskovits - [*The Definitive Guide to HTML5 WebSocket*](https://pdfhost.io/v/sisYree25_The_Definitive_Guide_to_HTML5_WebSocket.pdf)