ĐẠI HỌC ĐÀ NẴNG

KHOA CÔNG NGHỆ THỒN TIN VÀ TRUYỀN THÔNG

**ĐỒ ÁN MÔN HỌC**

**ĐỒ ÁN CƠ SỞ 4**



**Đề tài: NGHIÊN CỨU VÀ XÂY DỰNG CHƯƠNG TRÌNH DUYỆT WEB BẰNG HTTP**

**(**Simple Browser**)**

Sinh viên thực hiện : **Phạm Trung Hiếu**

Lớp : **18IT3**

Mã sinh viên : **18IT135**

Giảng viên hướng dẫn : **TS. Hồ Văn Phi**

***Đà nẵng, tháng 12 năm 2020***

*MỤC LỤC*

[PHẦN MỞ ĐẦU 4](#_Toc59976139)

[**CẤU TRÚC CỦA BÁO CÁO** 5](#_Toc59976140)

[CHƯƠNG I: CƠ SỞ LÍ THUYẾT 6](#_Toc59976141)

[1. Tổng quan về lập trình mạng 6](#_Toc59976142)

[1.1 Lập trình mạng là gì? 6](#_Toc59976143)

[1.2 Các giao thức và truyền thông 7](#_Toc59976144)

[1.3 Mô hình OSI 8](#_Toc59976145)

[1.4 Giao thức TCP và UDP 13](#_Toc59976146)

[Giao thức internet 14](#_Toc59976147)

[**2. Sơ lược về ngôn ngữ lập trình** 16](#_Toc59976148)

[2.1 Java là gì? 17](#_Toc59976149)

[2.2 Ứng dụng của java 18](#_Toc59976150)

[2.3 Những đặc điểm cơ bản của java 18](#_Toc59976151)

[**CHƯƠNG II: PHÂN TÍCH THIẾT KẾ HỆ THỐNG TRÌNH DUYỆT WEB** 21](#_Toc59976152)

[1 Giới thiệu bài toán 21](#_Toc59976153)

[1.1 Giới thiệu về các browser 21](#_Toc59976154)

[1.2 Công nghệ và ngôn ngữ sử dụng 23](#_Toc59976155)

[1.3 Phân tích yêu cầu 26](#_Toc59976156)

[**2 CÀI ĐẶT** 27](#_Toc59976157)

[2.1 Giao diện 29](#_Toc59976158)

[2.2 Phương thức tìm kiếm url 30](#_Toc59976159)

[2.3 Code browser 31](#_Toc59976160)

[2.4 Đặt home cho browser của bạn: 33](#_Toc59976161)

[2.5 Thay đổi home 33](#_Toc59976162)

[2.6 Tạo và hiển thị biểu mẩu 34](#_Toc59976163)

[**CHƯƠNG III: DEMO CHƯƠNG TRÌNH** 34](#_Toc59976164)

[1 Chạy demo browser 34](#_Toc59976165)

[2 Kết quả đạt được 36](#_Toc59976166)

[4 Đánh giá 38](#_Toc59976167)

[IV Kết Luận 39](#_Toc59976168)

******

# PHẦN MỞ ĐẦU

Tất cả chúng ta đều đã được tiếp cận với các ứng dụng để lướt web từ lâu. Sớm nhất là có yahoo, edgle, google chrome, firefox hay là ở Việt Nam mình có cốc cốc và một vài browser khác.

Em muốn thử sức mình ở lĩnh vực lập trình mạng và tạo ra một browser của riêng bản thân. Tuy khá đơn giản và chưa có gì thực sự nổi bật để so sanh với các browser khác nhưng đây là dự án đầu tiên của bản thân. Em vẫn chưa làm được thật sự tốt, mong thầy góp ý và đưa ra đánh giá để giúp em có thể hoàn thiện dự án này thật sự tốt.

Em có tham khảo 1 vài youtuber và 1 vài trang mạng về lập trình. Em tạo nên browser với giao diện đơn giản và tham khảo giao diện của cốc cốc.

Sau khi tìm hiểu và học hỏi em đã hoàn thiện được browser của bản thân. Mặc dù không được đẹp hay đầy đủ chức năng như các browser khác nhưng vẫn đáp ứng đầy đủ các chức năng mà các browser khác hiện có là tìm kiếm và chạy được các url.

Vì đây là đề tài đồ án môn học làm một người nên em sẽ cố gắng phát triển tốt nhất trong khả năng của bản thân. Dự kiến sẽ hoàn thành trước tháng 10-2020 và sẽ cố gắng phát triển tiếp mong sau này có thể đi vào hoạt động chính thức.

Em cũng xin được cảm ơn thầy Hồ Văn Phi đã hướng dẫn và giúp đỡ em trong đồ án cơ sở 4 trong suốt thời gian thực hiện đồ án.

Họ và tên sinh viên:

***Hiếu***

**Phạm Trung Hiếu**

# **CẤU TRÚC CỦA BÁO CÁO**

**Chương I: CƠ SỞ LÍ THUYẾT**

- Tổng quan về lập trình mạng.

- Sơ lược về ngôn ngữ lập trình

**Chương II: PHÂN TÍCH THIẾT KẾ HỆ THỐNG TRÌNH DUYỆT WEB**

* Đưa ra đặc tả yêu cầu của ứng dụng
* Phân tích và thiết kế hệ thống
* Thiết kế giao diện

**Chương III: DEMO CHƯƠNG TRÌNH**

* Chạy demo chương trình
* Tổng kết những gì đề án làm được và chưa làm được
* Đưa ra hướng cho đề tài

# CHƯƠNG I: CƠ SỞ LÍ THUYẾT

1. Tổng quan về lập trình mạng

### 1.1 Lập trình mạng là gì?

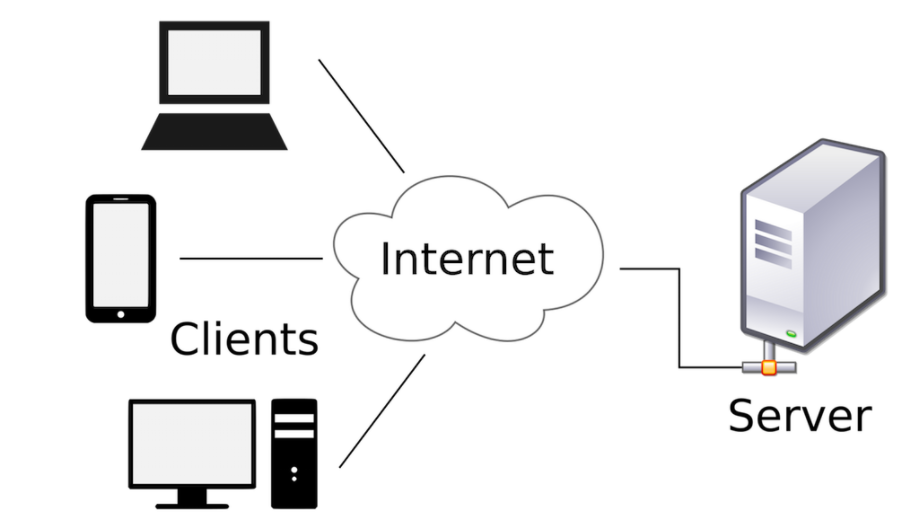
Ngày này khi nói đến phát triển các ứng dụng phần mềm, đa số là người ta muốn nói đến chương trình có khả năng làm việc trong môi trường mạng tích hợp nói chung và mạng máy tính nói riêng. Từ các chương trình kế toán doanh nghiệp, quản lý, trò chơi, điều khiển... đều là các chương trình ứng dụng mạng.

Vấn đề lập trình mạng liên quan đế nhiều lĩnh vực kiến thức khác nhau. Từ kiến thức sử dụng ngôn ngữ lập trình, phân tích thiết kế hệ thống, kiến thức hệ thống mạng, mô hình xây dựng chương trình ứng dụng mạng, kiến thức về cơ sở dữ liệu... cho đến kiến thức truyền thông, các kiến thức các lĩnh vực liên quan khác như mạng điện thoại di động, PSTN, hệ thống GPS, các mạng như BlueTooth, WUSB, mạng sensor.... Nhưng có thể nói vấn đề lập trình mạng có 3 vấn đề chính cốt lõi tích hợp trong lập trình ứng dụng mạng và được thể hiện như hình

Hay nói cách khác, vấn đề lập trình mạng có thể được định nghĩa với công thức sau:

LTM = KTM + MH + NN

* LTM: Lập trình mạng
* KTM: Kiến thức mạng truyền thông (mạng máy tính, PSTN,...)
* MH: Mô hình lập trình mạng
* NN: Ngôn ngữ lập trình mạng



### 1.2 Các giao thức và truyền thông

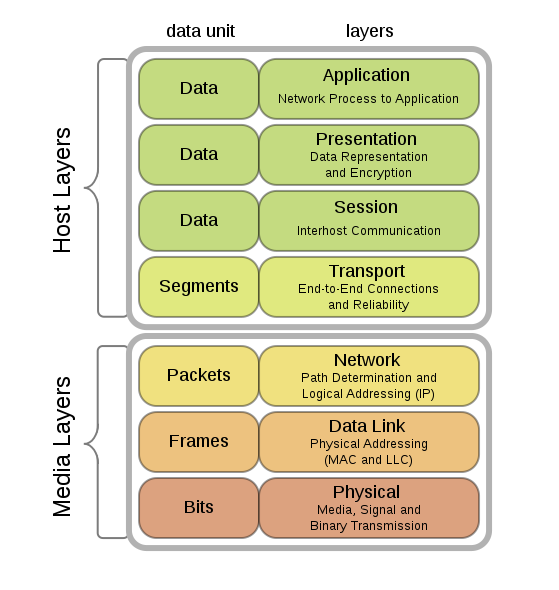
* Giao thức truyền thông (Communication protocol) hay Giao thức liên mạng, Giao thức truyền thông, Giao thức trao đổi thông tin - trong công nghệ thông tin gọi tắt là giao thức (protocol) - là một tập hợp các quy tắc chuẩn dành cho việc biểu diễn dữ liệu, phát tín hiệu, chứng thực và phát hiện các lỗi dữ liệu. Đó là những việc rất cần thiết để gửi thông tin qua các kênh truyền thông, nhờ đó mà các thiết bị có thể kết nối và trao đổi thông tin với nhau.
* Các giao thức dùng cho việc truyền thông tin tín hiệu số trong mạng máy tính có nhiều tính năng để đảm bảo việc trao đổi dữ liệu một các đáng tin cậy tqua một kênh truyền thông không hoàn hảo.
* Có các mô hình giao thức mang tính lý thuyết, được ngành công nghệ thông tin tôn trọng một cách tương đối như mô hình OSI. Cũng có những giao thức được triển khai và ứng dụng rộng rãi như TCP, IP, HTTP, FTP, SMTP, POP3...
* Có nhiều giao thức được sử dụng để giao tiếp hoặc truyền đạt thông tin trên Internet, dưới đây là một số các giao thức tiêu biểu:
  + TCP (*Transmission Control Protocol*): thiết lập kết nối giữa các máy tính để truyền dữ liệu. Nó chia nhỏ dữ liệu ra thành những gói (*packet*) và đảm bảo việc truyền dữ liệu thành công.
  + IP (*Internet Protocol*): định tuyến (*route*) các gói dữ liệu khi chúng được truyền qua Internet, đảm bảo dữ liệu sẽ đến đúng nơi cần nhận.
  + HTTP (*HyperText Transfer Protocol*): cho phép trao đổi thông tin (chủ yếu ở dạng siêu văn bản) qua Internet.
  + FTP (*File Transfer Protocol*): cho phép trao đổi tập tin qua Internet.
  + SMTP (*Simple Mail Transfer Protocol*): cho phép gởi các thông điệp thư điện tử (*e-mail*) qua Internet.
  + POP3 (*Post Office Protocol*, phiên bản 3): cho phép nhận các thông điệp thư điện tử qua Internet.
  + MIME (*Multipurpose Internet Mail Extension*): một mở rộng của giao thức SMTP, cho phép gởi kèm các tập tin nhị phân, phim, nhạc,... theo thư điện tử.
  + WAP (*Wireless Application Protocol*): cho phép trao đổi thông tin giữa các thiết bị không dây, như điện thoại di động.
  + Bitcoin: Cho phép thanh toán trực tiếp qua thiết bị kết nối Internet mà không cần thông qua bất kỳ tổ chức tài chính trung gian nào, với đơn vị tiền tệ là bitcoin.

### 1.3 Mô hình OSI

##### 1.3.1Tổng quan của mô hình OSI

**Mô hình OSI** (*Open Systems Interconnection Reference Model*, viết ngắn là *OSI Model* hoặc *OSI Reference Model*) - tạm dịch là **Mô hình tham chiếu kết nối các hệ thống mở** - là một thiết kế dựa vào nguyên lý tầng cấp, lý giải một cách trừu tượng kỹ thuật kết nối truyền thông giữa các máy vi tính và thiết kế giao thức mạng giữa chúng. Mô hình này được phát triển thành một phần trong kế hoạch Kết nối các hệ thống mở (*Open Systems Interconnection*) do ISO và IUT-T khởi xướng. Nó còn được gọi là **Mô hình bảy tầng của OSI**.

##### 1.3.2Tường trình các tầng cấp của mẫu hình OSI



**Tầng 1: Tầng vật lý (*Physical Layer*)**

Tầng vật lý định nghĩa tất cả các đặc tả về điện và vật lý cho các thiết bị. Trong đó bao gồm bố trí của các chân cắm(*pin*), các hiệu điện thế, và các đặc tả về cáp nối (*cable*). Các thiết bị tầng vật lý bao gồm Hub, bộ lặp (*repeater*),thiết bị chuyển đổi tín hiệu(Converter), thiết bị tiếp hợp mạng (*network adapter*) và thiết bị tiếp hợp kênh máy chủ (*Host Bus Adapter*)- (HBA dùng trong mạng lưu trữ (*Storage Area Network*)). Chức năng và dịch vụ căn bản được thực hiện bởi tầng vật lý bao gồm:

* Thiết lập hoặc ngắt mạch kết nối điện (*electrical connection*) với một môi trường truyền dẫn phương tiện truyền thông (*transmission medium*).
* Tham gia vào quy trình mà trong đó các tài nguyên truyền thông được chia sẻ hiệu quả giữa nhiều người dùng. Chẳng hạn giải quyết tranh chấp tài nguyên (*contention*) và điều khiển lưu lượng.
* Điều chế (*modulation*), hoặc biến đổi giữa biểu diễn dữ liệu số (*digital data*) của các thiết bị người dùng và các tín hiệu tương ứng được truyền qua kênh truyền thông (*communication channel*).

Cáp (*bus*) SCSI song song hoạt động ở tầng cấp này. Nhiều tiêu chuẩn khác nhau của Ethernet dành cho tầng vật lý cũng nằm trong tầng này; Ethernet nhập tầng vật lý với tầng liên kết dữ liệu vào làm một. Điều tương tự cũng xảy ra đối với các mạng cục bộ như Token ring, FDDI và IEEE 802.11.]]

**Tầng 2: Tầng liên kết dữ liệu (*Data-Link Layer*)**

Tầng liên kết dữ liệu cung cấp các phương tiện có tính chức năng và quy trình để truyền dữ liệu giữa các thực thể mạng (truy cập đường truyền, đưa dữ liệu vào mạng), phát hiện và có thể sửa chữa các lỗi trong tầng vật lý nếu có. Cách đánh địa chỉ mang tính vật lý, nghĩa là địa chỉ (địa chỉ MAC) được mã hóa cứng vào trong các thẻ mạng (*network card*) khi chúng được sản xuất. Hệ thống xác định địa chỉ này không có đẳng cấp (*flat scheme*). *Chú ý:* Ví dụ điển hình nhất là Ethernet. Những ví dụ khác về các giao thức liên kết dữ liệu (*data link protocol*) là các giao thức HDLC; ADCCP dành cho các mạng điểm-tới-điểm hoặc mạng chuyển mạch gói (*packet-switched networks*) và giao thức Aloha cho các mạng cục bộ. Trong các mạng cục bộ theo tiêu chuẩn IEEE 802, và một số mạng theo tiêu chuẩn khác, chẳng hạn FDDI, tầng liên kết dữ liệu có thể được chia ra thành 2 tầng con: tầng MAC (*Media Access Control* - Điều khiển Truy nhập Đường truyền) và tầng LLC (*Logical Link Control* - Điều khiển Liên kết Logic) theo tiêu chuẩn IEEE 802.2.

Tầng liên kết dữ liệu chính là nơi các thiết bị chuyển mạch (switches) hoạt động. Kết nối chỉ được cung cấp giữa các nút mạng được nối với nhau trong nội bộ mạng.

**Tầng 3: Tầng mạng (*Network Layer*)**

Tầng mạng cung cấp các chức năng và quy trình cho việc truyền các chuỗi dữ liệu có độ dài đa dạng, từ một nguồn tới một đích, thông qua một hoặc nhiều mạng, trong khi vẫn duy trì chất lượng dịch vụ (*quality of service*) mà tầng giao vận yêu cầu. Tầng mạng thực hiện chức năng định tuyến,.Các thiết bị định tuyến (*router*) hoạt động tại tầng này — gửi dữ liệu ra khắp mạng mở rộng, làm cho liên mạng trở nên khả thi (còn có thiết bị chuyển mạch (*switch*) tầng 3, còn gọi là chuyển mạch IP). Đây là một hệ thống định vị địa chỉ lôgic (*logical addressing scheme*) – các giá trị được chọn bởi kỹ sư mạng. Hệ thống này có cấu trúc phả hệ. Ví dụ điển hình của giao thức tầng 3 là giao thức IP.

**Tầng 4: Tầng giao vận (*Transport Layer*)**

Tầng giao vận cung cấp dịch vụ chuyên dụng chuyển dữ liệu giữa các người dùng tại đầu cuối, nhờ đó các tầng trên không phải quan tâm đến việc cung cấp dịch vụ truyền dữ liệu đáng tin cậy và hiệu quả. Tầng giao vận kiểm soát độ tin cậy của một kết nối được cho trước. Một số giao thức có định hướng trạng thái và kết nối (*state and connection orientated*). Có nghĩa là tầng giao vận có thể theo dõi các gói tin và truyền lại các gói bị thất bại. Một ví dụ điển hình của giao thức tầng 4 là TCP. Tầng này là nơi các thông điệp được chuyển sang thành các gói tin TCP hoặc UDP. Ở tầng 4 địa chỉ được đánh là address ports, thông qua address ports để phân biệt được ứng dụng trao đổi.

**Tầng 5: Tầng phiên (*Session layer*)**

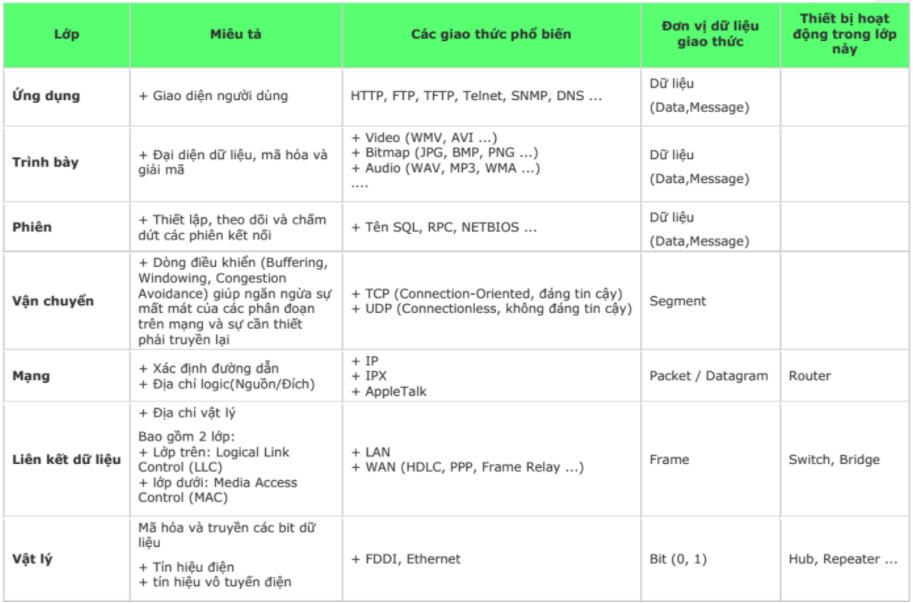
Tầng phiên kiểm soát các (phiên) hội thoại giữa các máy tính. Tầng này thiết lập, quản lý và kết thúc các kết nối giữa trình ứng dụng địa phương và trình ứng dụng ở xa. Tầng này còn hỗ trợ hoạt động song công (*duplex*) hoặc bán song công (*half-duplex*) hoặc đơn công (*Single*) và thiết lập các quy trình đánh dấu điểm hoàn thành (*checkpointing*) - giúp việc phục hồi truyền thông nhanh hơn khi có lỗi xảy ra, vì điểm đã hoàn thành đã được đánh dấu - trì hoãn (*adjournment*), kết thúc (*termination*) và khởi động lại (*restart*). Mô hình OSI uỷ nhiệm cho tầng này trách nhiệm "ngắt mạch nhẹ nhàng" (*graceful close*) các phiên giao dịch (một tính chất của giao thức kiểm soát giao vận TCP) và trách nhiệm kiểm tra và phục hồi phiên, đây là phần thường không được dùng đến trong bộ giao thức TCP/IP.

**Tầng 6: Tầng trình diễn (*Presentation layer*)**

Lớp trình diễn hoạt động như tầng dữ liệu trên mạng. lớp này trên máy tính truyền dữ liệu làm nhiệm vụ dịch dữ liệu được gửi từ tầng Application sang dạng Fomat chung. Và tại máy tính nhận, lớp này lại chuyển từ Fomat chung sang định dạng của tầng Application. Lớp thể hiện thực hiện các chức năng sau: - Dịch các mã ký tự từ ASCII sang EBCDIC. - Chuyển đổi dữ liệu, ví dụ từ số interger sang số dấu phảy động. - Nén dữ liệu để giảm lượng dữ liệu truyền trên mạng. - Mã hoá và giải mã dữ liệu để đảm bảo sự bảo mật trên mạng.

**Tầng 7: Tầng ứng dụng (*Application layer*)**

Tầng ứng dụng là tầng gần với người sử dụng nhất. Nó cung cấp phương tiện cho người dùng truy nhập các thông tin và dữ liệu trên mạng thông qua chương trình ứng dụng. Tầng này là giao diện chính để người dùng tương tác với chương trình ứng



### 1.4 Giao thức TCP và UDP

##### 1.4.1 TCP

* TCP (*Transmission Control Protocol* - "Giao thức điều khiển truyền vận") là một trong các giao thức cốt lõi của bộ giao thức TCP/IP. Sử dụng TCP, 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. 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 (chẳng hạn, dịch vụ Web và dịch vụ thư điện tử) đồng thời chạy trên cùng một máy chủ.
* TCP hỗ trợ nhiều giao thức ứng dụng phổ biến nhất trên Internet hiện nay như WWW, Email và Secure Shell.
* Dữ liệu được gửi từ máy nguồn thành các dòng (stream) bao gồm các byte (8 bit) tới TCP đến chuyển qua mạng. TCP chia các dòng byte này thành các đoạn (segment) có kích thước thích hợp ( quyết định dựa theo kích thước của đơn vị truyền dẫn tối đa MTU) của tầng Data Link.
* TCP đảm bảo tính toàn vẹn của thông tin bằng cách gắn cho mỗi gói tin một số thứ tự ( sequence number). Số thứ tự này còn dùng để đảm bảo dữ liệu được truyền từ nguồn đến đích theo đúng thứ tự. Máy đích nhận thông tin sẽ gửi lại "tin báo nhận" (acknowledgement) cho các gói tin đã nhận thành công. Một "đồng hồ" (timer) tại máy nguồn sẽ báo lỗi time-out nếu không nhận được "tin báo nhận" trong một khoảng thời gian bằng một round-trip time (RTT), và dữ liệu - lúc đó được coi là bị thất lạc - sẽ được gửi lại.
* TCP sử dụng giá trị kiểm tra (checksum) để kiểm tra xem có byte nào bị hỏng trong quá trình truyền hay không. Giá trị này được tính toán cho mỗi khối dữ liệu trước khi gửi đi tại nơi gửi và được kiểm tra lại tại nơi nhận.
* Là một giao thức đảm bảo tính toàn vẹn thông tin và khả năng bảo mật cao, TCP được sử dụng rộng rãi trong HTTP/HTTPS, FPT (truyền file), SMTP/POP3/IMAP (email).

##### 1.4.2 UDP

* UDP (User Datagram Protocol) là một phần của bộ giao thức Internet, được sử dụng bởi các chương trình chạy trên các máy khác nhau trong mạng. UDP hoạt động tương tự như TCP nhưng nó không bao gồm quá trình kiểm tra lỗi khi truyền gói tin, vì vậy, nó không đảm bảo tính toàn vẹn của dữ liệu.
* Do bản chất không bao gồm quá trình kiểm tra nên UDP nhanh hơn, hữu dụng với các việc trả lời các truy vấn nhỏ với lượng lớn người yêu cầu.
* UDP được sử dụng phổ biến trong các ứng dụng stream media, Voice over IP và game online.

##### 1.4.3 Sự khác nhau giữa TCP và UDP

|  |  |
| --- | --- |
| **TCP** | **UDP** |
| Không cho phép mất gói tin | Cho phép mất dữ liệu |
| Tốc độ truyền thấp hơn UDP | Tốc độ truyền cao |

### Giao thức internet

##### 1.5.1 TCP/IP

**Bộ giao thức TCP/IP**, (tiếng Anh: *Internet protocol suite* hoặc *IP suite* hoặc *TCP/IP protocol suite* - bộ giao thức liên mạng), là một bộ các giao thức truyền thông cài đặt chồng giao thức mà Internet và hầu hết các mạng máy tính thương mại đang chạy trên đó. Bộ giao thức này được đặt tên theo hai giao thức chính của nó là TCP (*Giao thức Điều khiển Giao vận*) và IP (*Giao thức Liên mạng*). Chúng cũng là hai giao thức đầu tiên được định nghĩa.

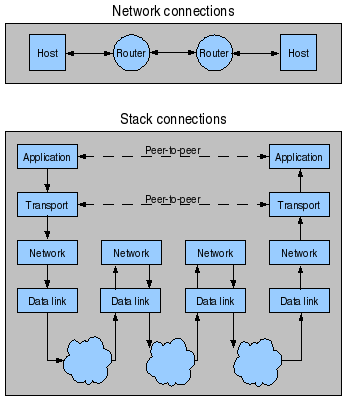
Như nhiều bộ giao thức khác, bộ giao thức TCP/IP có thể được coi là một tập hợp các tầng, mỗi tầng giải quyết một tập các vấn đề có liên quan đến việc truyền dữ liệu, và cung cấp cho các giao thức tầng cấp trên một dịch vụ được định nghĩa rõ ràng dựa trên việc sử dụng các dịch vụ của các tầng thấp hơn. Về mặt lôgic, các tầng trên gần với người dùng hơn và làm việc với dữ liệu trừu tượng hơn, chúng dựa vào các giao thức tầng cấp dướiđể biến đổi dữ liệu thành các dạng mà cuối cùng có thể được truyền đi một cách vật lý.

|  |  |
| --- | --- |
| **Tầng ứng dụng** | DNS, TFTP, TLS/SSL, FTP, HTTP, IMAP, IRC, NNTP, POP3, SIP, SMTP, SNMP, SSH, TELNET, ECHO, BitTorrent, RTP, PNRP, rlogin, ENRP, … |
|  | Các giao thức định tuyến như BGP và RIP, vì một số lý do, chạy trên TCP và UDP - theo thứ tự từng cặp: BGP dùng TCP, RIP dùng UDP - còn có thể được coi là một phần của tầng ứng dụng hoặc tầng mạng. |
| **Tầng giao vận** | TCP, UDP, DCCP, SCTP, IL, RUDP, … |
|  | Các giao thức định tuyến như OSPF (*tuyến ngắn nhất được chọn đầu tiên*), chạy trên IP, cũng có thể được coi là một phần của tầng giao vận, hoặc tầng mạng. ICMP (*Internet control message protocol| - tạm dịch là Giao thức điều khiển thông điệp Internet*) và IGMP (*Internet group management protocol - tạm dịch là Giao thức quản lý nhóm Internet*) chạy trên IP, có thể được coi là một phần của tầng mạng. |
| **Tầng mạng** | IP (IPv4, IPv6) |
|  | ARP (*Address Resolution Protocol| - tạm dịch là Giao thức tìm địa chỉ*) và RARP (*Reverse Address Resolution Protocol - tạm dịch là Giao thức tìm địa chỉ ngược lại*) hoạt động ở bên dưới IP nhưng ở trên tầng liên kết (*link layer*), vậy có thể nói là nó nằm ở khoảng trung gian giữa hai tầng. |
| **Tầng liên kết** | Ethernet, Wi-Fi, Token ring, PPP, SLIP, FDDI, ATM, Frame Relay, SMDS, … |

Mô hình OSI miêu tả một tập cố định gồm 7 tầng mà một số nhà sản xuất lựa chọn và nó có thể được so sánh tương đối với bộ giao thức TCP/IP. Sự so sánh này có thể gây nhầm lẫn hoặc mang lại sự hiểu biết sâu hơn về bộ giao thức TCP/IP.

##### 1.5.2 Các tầng trong chồng giao thức của bộ giao thức TCP/IP

Bộ giao thức IP dùng sự đóng gói dữ liệu hòng trừu tượng hóa (*thu nhỏ lại quan niệm cho dễ hiểu*) các giao thức và các dịch vụ. Nói một cách chung chung, giao thức ở tầng cao hơn dùng giao thức ở tầng thấp hơn để đạt được mục đích của mình. Chồng giao thức Internet gần giống như các tầng cấp trong mô hình của Bộ quốc phòng Mỹ:



## **2. Sơ lược về ngôn ngữ lập trình**

Nói chung tất cả các ngôn ngữ lập trình đều có thể sử dụng để lập trình mạng. Nhưng mỗi ngôn ngữ có những ưu, nhược điểm khác nhau và được hỗ trợ thư viện API ở các mức độ khác nhau. Tuỳ từng ứng dụng mạng cụ thể, hệ điều hành mạng cụ thể và thói quen lập trình mà người lập trình có thể chọn ngôn ngữ phù hợp để phát triển các ứng dụng mạng. Các ngôn ngữ lập trình phổ biến hiện nay gồm những ngôn ngữ sau:

* Hợp ngữ (Assembly Language)
* C/C++
* VC++, VB, Delphi
* Java
* .NET
* ASP
* Javascript

Đối với phát triển ứng dụng mạng hiện nay có 2 ngôn ngữ lập trình được sử dụng phổ biến nhất, đó là .NET và Java. Người lập trình có thể sử dụng thành thạo một trong 2 dòng ngôn ngữ đó để phát triển ứng dụng mạng. Tuy nhiên ngày nay với sự xuất hiện của Javascript đã tạo ra một xu hướng lập trình Full-Stack thay thế cho các công nghệ lập trình front-end hay back-end truyền thống.

### 2.1 Java là gì?

Java là một ngôn ngữ lập lập trình, được phát triển bởi **Sun Microsystem** vào năm 1995, là ngôn ngữ kế thừa trực tiếp từ C/C++ và là một ngôn ngữ lập trình hướng đối tượng.

Vì sao ngôn ngữ này lại được đặt tên là Java? Java là tên một hòn đảo ở Indonesia - hòn đảo nổi tiếng với loại coffee Peet và cũng là loại nước uống phổ biến của các kỹ sư Sun. Ban đầu Ngôn ngữ này được đặt tên là "Oak" (có nghĩa là "Cây sồi" - 1991), nhưng các luật sư của Sun xác định rằng tên đó đã được đăng ký nhãn hiệu nên các nhà phát triển đã phải thay thế bằng một tên mới -  và cũng vì lý do trên mà cái tên Java đã ra đời và trở thành tên gọi chính thức của Ngôn ngữ này - Ngôn ngữ Lập trình Java.

### 2.2 Ứng dụng của java

Ngày nay Java được sử dụng với các mục đích sau:

* Phát triển ứng dụng cho các thiết bị điện tử thông minh, các ứng dụng cho doanh nghiệp với quy mô lớn.
* Tạo các trang web có nội dung động *(web applet*), nâng cao chức năng của server.
* Phát triển nhiều loại ứng dụng khác nhau: Cơ sở dữ liệu, mạng, Internet, viễn thông, giải trí,...

### 2.3 Những đặc điểm cơ bản của java

Tiêu chí hàng đầu của Ngôn ngữ Lập trình Java là **"Write Once, Run Anywhere"** (*Viết một lần, chạy mọi nơi*), nghĩa là Java cho phép chúng ta viết code một lần và thực thi được trên các hệ điều hành khác nhau. Ví dụ, chúng ta viết code trên Hệ điều hành Windows và nó có thể thực thi được trên các Hệ điều hành Linux và Mac OS...

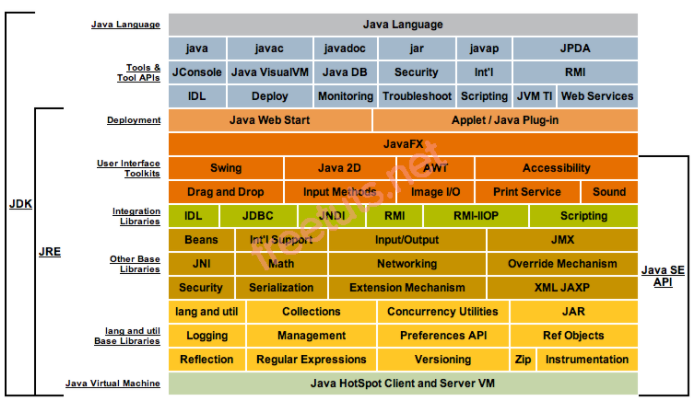
Với đặc điểm nổi bật đó, Java có những đặc điểm cơ bản như sau:

* Đơn giản và quen thuộc: Vì Java kế thừa trực tiếp từ C/C++ nên nó có những đặc điểm của ngôn ngữ này, Java đơn giản vì mặc dù dựa trên cơ sở C++ nhưng Sun đã cẩn thận lược bỏ các tính năng khó nhất của của C++ để làm cho ngôn ngữ này dễ sử dụng hơn.
* Hướng đối tượng và quen thuộc.
* Mạnh mẽ *(thể hiện ở cơ chế tự động thu gom rác - Garbage Collection*) và an toàn.
* Kiến trúc trung lập, độc lập nền tảng và có tính khả chuyển (*Portability*).
* Hiệu suất cao.
* Máy ảo (biên dịch và thông dịch).
* Phân tán.
* Đa nhiệm: Ngôn ngữ Java cho phép xâ dựng trình ứng dụng, trong đó nhiều quá trình có thể xảy ra đồng thời. Tính đa nhiệm cho phép các nhà lập trình có thể biên soạn phần mềm đáp ứng tốt hơn, tương tác tốt hơn và thực hiện theo thời gian thực.

#### 2.3.1 Các platform cơ bản của java

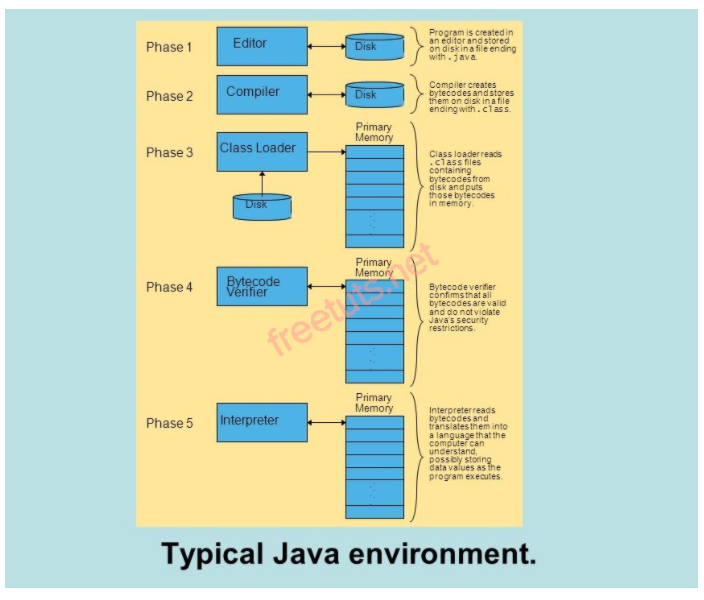
Java Platform gồm có 3 thành phần chính:

* Java Virtual Machine (*Java VM*): Máy ảo Java.
* Java Application Programming Interface (*Java API*).
* Java Development Kit (*JDK*) gồm trình biên dịch, thông dịch, trợ giúp, soạn tài liệu... và các thư viện chuẩn.

****

#### 2.3.2Tiêu chuẩn của 1 môi trường java điển hình

Thông thường, các chương trình Java trải qua 5 giai đoạn chính:

* **Editor**: Lập trình viên viết chương trình và được lưu vào máy tính với định dạng .java.
* **Compiler**: Biên dịch chương trình thành bytecodes (*định dạng .class*) - nhờ bước trung gian này mà Java được viết 1 lần và chạy trên các hệ điều hành khác nhau.
* **Class Loader**: Đọc file .class chứa mã bytecodes và lưu vào trong bộ nhớ.
* **Bytecode Verifier**: Đảm bảo rằng mã bytecodes là hợp lệ và không vi phạm các vấn đề về bảo mật của Java.
* **Intepreter**: Biên dịch bytecodes thành mã máy để máy tính có thể hiểu được và sau đó thực thi chương trình.
* 

# **CHƯƠNG II: PHÂN TÍCH THIẾT KẾ HỆ THỐNG TRÌNH DUYỆT WEB**

## 1 Giới thiệu bài toán

### Giới thiệu về các browser

##### **Web browser – trình duyệt web?**

1. Một **web browser** là một phần mềm ứng dụng để truy xuất, trình diễn và chuyển các nguồn thông tin (information resource) trên mạng hệ thống mạng toàn cầu (World Wide Web). Một nguồn thông tin được nhận dạng bởi một Uniform Resource Identifier (URI) và có thể là một trang web, phim - video, hình ảnh (images) hoặc các mẫu thông tin khác.
2. Mặc dù các trình duyệt với mục đích là để truy cập vào hệ thống mạng toàn cầu, các trình duyệt còn được sử dụng để truy cập các thông tin được cung cấp bởi các web servers (máy chủ web) trong hệ thống mạng riêng hoặc các tài liệu (files) đến các hệ thống file (file system). Hoặc cũng được dùng để tiết kiệm tài nguyên thông tin cho các hệ thống lưu trữ file.

##### **Lịch sử về browser**

1. **Lịch sử của các trình duyệt Web** bắt đầu vào cuối thập niên 1980, khi một loạt các công nghệ đặt nền móng cho các trình duyệt Web đầu tiên, WorldWideWeb, do Tim Berners-Lee vào năm 1991. Sự phát triển này dựa trên các công nghệ phần mềm và phần cứng hiện có, ngoài ra cũng mở ra nhiều công nghệ mới bắt đầu tư đây.
2. Ted Nelson và Douglas Engelbart phát triển khái niệm siêu văn bản trước khi Berners-Lee và CERN khá lâu. Web browser đã trở thành cốt lõi của World Wide Web. Berners-Lee thừa nhận sự đóng góp của Engelbart.
3. Việc giới thiệu trình duyệt Mosaic NCSA Web vào năm 1993 - một trong những trình duyệt web với giao diện đồ họa đầu tiên - đã dẫn tới sự bùng nổ trong việc dùng web. Marc Andreessen, người lãnh đạo của đội Mosaic tại NCSA, sớm bắt đầu công ty riêng của mình, đặt tên là Netscape, và phát hành các phiên bản Mosaic Netscape Navigator vào năm 1994, và sau đó nhanh chóng trở thành trình duyệt phổ biến nhất thế giới, chiếm 90% của tất cả các sử dụng Web đỉnh cao của nó.
4. Microsoft đã đáp trả bằng trình duyệt Internet Explorer của mình trong năm 1995 (cũng bị ảnh hưởng nhiều từ Mosaic), và đầu cuộc chiến trình duyệt đầu tiên. Bằng việc kết hợp Internet Explorer với Windows, Microsoft đã có thể tận dụng ưu thế của nó trong thị trường hệ điều hành để quản lý thị trường trình duyệt Web; Internet Explorer sử dụng đạt vị trí trên 95% người dùng vào năm 2002. Internet Explorer chiếm 60% người sử dụng trình duyệt vào Tháng Tư năm 2010 theo thống kê của Net Applications.
5. Opera đầu tiên xuất hiện vào năm 1996; mặc dù nó đã không bao giờ đạt được sử dụng rộng rãi, chiếm 2% thị phần trình duyệt vào tháng tư năm 2010, tuy nhiên Opera chiếm một phần đáng kể của thị trường đang phát triển nhanh là trình duyệt Web điện thoại di động, Oepra được cài đặt sẵn trên trên 40.000.000 điện thoại. Opera cũng có sẵn trên một số hệ thống nhúng khác, bao gồm video game console Wii của Nintendo.
6. Năm 1998, Netscape triển khai Mozilla Foundation trong một nỗ lực để sản xuất một trình duyệt cạnh tranh bằng cách sử dụng mô hình phần mềm mã nguồn mở. Đó là trình duyệt sau nay phát triển thành Firefox, sau đó thì trình duyệt này được phát triển khá tốt trong khi vẫn còn trong giai đoạn beta, ngay sau khi phát hành Firefox 1.0 vào cuối năm 2004, Firefox (mọi phiên bản) đã chiếm 7,4% thị phần sử dụng trình duyệt. Và đến tháng tư năm 2010, Firefox đã có một thị phần 25%.
7. Safari của Apple đã phát hành phiên bản beta đầu tiên vào tháng một năm 2003; vào tháng 10 năm 2009, đã chi phối thị phần của trình duyệt web dựa trên việc phát triển từ các ứng dụng của Apple, chiếm dưới 5% thị trường trình duyệt toàn bộ vào tháng 4 năm 2010. Nó được xem như là công cụ cho web (còn được gọi là WebKit) để trình diễn các tính năng và là nền tảng được ứng dụng nhiều trên thiết bị di động, bao gồm cả hệ điều hành iPhone, Google Android, Nokia S60 và Palm WebOS.
8. Vào tháng 9 năm 2008, Google bắt đầu nhảy vào cuộc chiến các trình duyệt bằng trình duyệt web Google Chrome. Đến tháng 4 năm 2010, Google Chrome đã chiếm 7% thị phần.

##### **1.1.3 Chức năng**

**Mục đích chính của một trình duyệt web** là để mang lại nguồn thông tin cho người dùng. Quá trình này bắt đầu khi một người sử dụng nhập vào (URI) hay tạm gọi là một đường dẫn, Ví dụ như http://google.com.vn. Các tiền tố của URI sẽ xác định cho phương thức truy cập và dữ liệu nhận được sẽ được biên dịch như thế nào. Các loại URI thông dụng nhất bắt đầu với http: định một nguồn tài nguyên để được lấy dựa trên Hypertext Transfer Protocol (HTTP) (tôi tạm gọi là giao thức truyền dữ liệu cho các siêu văn bản). Nhiều trình duyệt cũng hỗ trợ một loạt các tiền tố khác, chẳng hạn như https: (cũng tương tự như HTTP nhưng dữ liệu được mã hóa và bảo mật hơn) hay ftp: cho File Transfer Protocol (giao thức truyền tải file), và file: cho các tập tin lưu trữ nội bộ. Các trình duyệt web có thể không trực tiếp xử lý mà thường chuyển qua các úng dụng khác xử lý. Ví dụ, mailto: URI thường sẽ được chuyển qua chương trình ứng dụng mail mặc định mà người dùng đang sử dụng, hoặc news: sẽ gọi đến các chương trình đọc tin tức.

Tuy nhiên trình duyệt cũng có cơ chế mở và được hỗ trợ bởi các plugins của nó để có thể hiển thị được các loại tài liệu trực tiếp trên web browser. Hầu hết các trình duyệt có thể hiển thị hình ảnh, âm thanh, video, và các tập tin XML, và thường có một số plug-in mặc định để hỗ trợ các ứng dụng Flash và Java applet. Khi gặp phải một tập tin không được hỗ trợ thì tập tin này sẽ được tải về thay vì hiển thị trực tiếp trên web, lúc này trình duyệt sẽ nhắc người dùng để lưu tập tin vào đĩa.

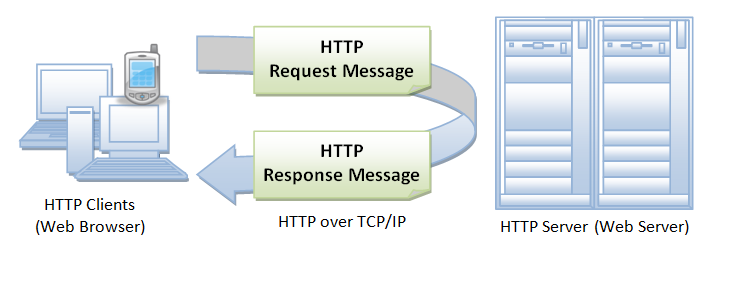
Tính tương tác trong một trang web có thể được ứng dụng bởi javascript mà thường không đòi hỏi sự hỗ trợ thêm của plugin. javascript có thể được sử dụng cùng với các công nghệ khác để cho phép tương tác trực tiếp với máy chủ trang web, ví dụ: tương tác thông qua AJAX.

Các thông tin trên trang web có thể chứa siêu liên kết (hyperlinks) tới các nguồn thông tin khác. Mỗi liên kết có chứa các URI để đi đến các nguồn thông tin khác. Khi một liên kết được nhấp, trình duyệt điều hướng người dùng tới trang thông tin khác.

### 1.2 Công nghệ và ngôn ngữ sử dụng

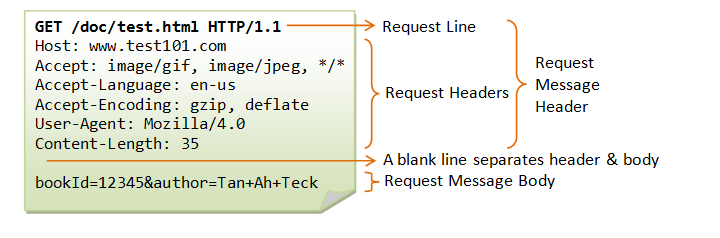
##### 1 Hyper Text Transfer Protocol (HTTP)

HTTP (Hyper Text Transfer Protocol - Giao thức truyền tải siêu văn bản) là một trong năm giao thức chuẩn về mạng Internet, được dùng để liên hệ thông tin giữa Máy chủ cung cấp dịch vụ (Web server) và Máy sử dụng dịch vụ (Web client), là giao thức Client - Server dùng cho World Wide Web (www). HTTP là một giao thức ứng dụng của bộ giao thức TCP/IP (các giao thức nền tảng cho Internet)



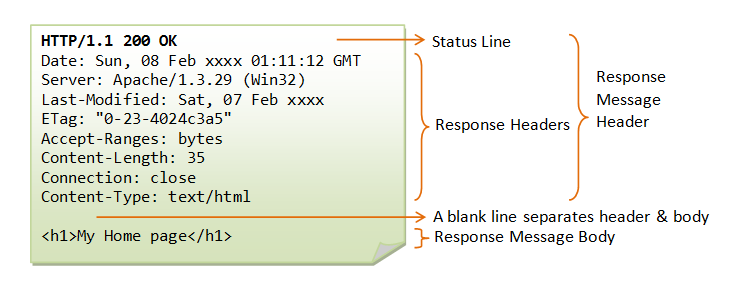
##### 2 HTTP request

* Để bắt đầu trao đổi dữ liệu, phía client khởi tạo một HTTP session bằng cách mở một kết nối TCP đến HTTP server sau đó gửi request đến server này. Request có thể được tạo bằng nhiều cách, trực tiếp khi người dùng nhấp vào một liên kết trên trình duyệt hoặc gián tiếp, ví dụ như một video được đính kèm trong một website và việc request đến website này sẽ dẫn đến một request tới video ấy.



* Bắt đầu của HTTP Request sẽ là dòng Request-Line bao gồm 3 thông tin đó là:  
  + Method: là phương thức mà HTTP Request này sử dụng, thường là GET, POST, ngoài ra còn một số phương thức khác như HEAD, PUT, DELETE, OPTION, CONNECT. Trong ví dụ trên là GET
  + URI: là địa chỉ định danh của tài nguyên. Trong tường hợp này URI là / - tức request cho tài nguyên gốc, nếu request không yêu cầu một tài nguyên cụ thể, URI có thể là dấu \*.
  + HTTP version: là phiên bản HTTP đang sử dụng, ở đây là HTTP 1.1.
* Tiếp theo là các trường request-header, cho phép client gửi thêm các thông tin bổ sung về thông điệp HTTP request và về chính client. Một số trường thông dụng như:
  + Accept: loại nội dung có thể nhận được từ thông điệp response. Ví dụ: text/plain, text/html…
  + Accept-Encoding: các kiểu nén được chấp nhận. Ví dụ: gzip, deflate, xz, exi…
  + Connection: tùy chọn điều khiển cho kết nối hiện thời. Ví dụ: keep-alive, Upgrade…
  + Cookie: thông tin HTTP Cookie từ server.
  + User-Agent: thông tin về user agent của người dùng.

##### 3 HTTP Response

* Cấu trúc HTTP response gần giống với HTTP request, chỉ khác nhau là thay vì Request-Line, thì HTTP có response có Status-Line. Và giống như Request-Line, Status-Line cũng có ba phần như sau:
  + HTTP-version: phiên bản HTTP cao nhất mà server hỗ trợ.
  + Status-Code: mã kết quả trả về.
  + Reason-Phrase: mô tả về Status-Code.
  + 

Nắm bắt và tiếp thu kiến thức đã và đang học trong chương trình, với nền tảng kiến thức môn Lập trình mạng và kiến thức về ngôn ngữ java. Bằng phần mềm esclipse em đã tạo nên browser của riêng mình. Em có tham khảo một vài youtuber về mảng lập trình mạng và tìm hiểu thêm về các giao thức HTTP để hoàn thành tốt đề tài đồ án này.

### 1.3 Phân tích yêu cầu

##### 1.Yêu cầu chức năng:

- Trình duyệt web, chạy các Url, đường dẫn…

- Xem video, nghe nhạc, đọc báo, lướt web, các trang mạng xã hội, chơi game đơn giản trên browser…

- xem, sửa lịch sử duyệt web.

##### 2.Yêu cầu phi chức năng:

-Có thể mở tối đa 20 tab mới.

-luôn luôn hoạt động mọi lúc, mọi điều kiện, 24/7.

-Chứa tối đa 10000 lượt truy cập cùng 1 thời điểm.

##### 3.Yêu cầu hệ thống

* Có thể tương thích tất cả các hệ diều hành như macos, window, linus,…
* Không yêu cầu cấu hình quá cao, cụ thể: + Ram: lớn hơn 512mb.

+ Rom: Trống trên 5gb.

+SSD: không bắt buộc, dùng HDD vẫn có thể cho ra tốc độ tốt nhất.

+CPU: có tần số trên 1.0 ghz.

* Nhẹ, thuận tiện và dễ sử dụng.

3.1. Yêu cầu tối thiểu.

-Ram: trên 512mb.

-Rom: Trống trên 5gb.

-CPU: Intel core i3-5th gen 1.0ghz.

3.2.Yêu cầu đề cử.

-Ram: Trên 1gb.

-Rom: Trống trên 10gb.

-CPU: Intel core i3-9th gen 2.4ghz.

-VGA: intel graphics 520s.

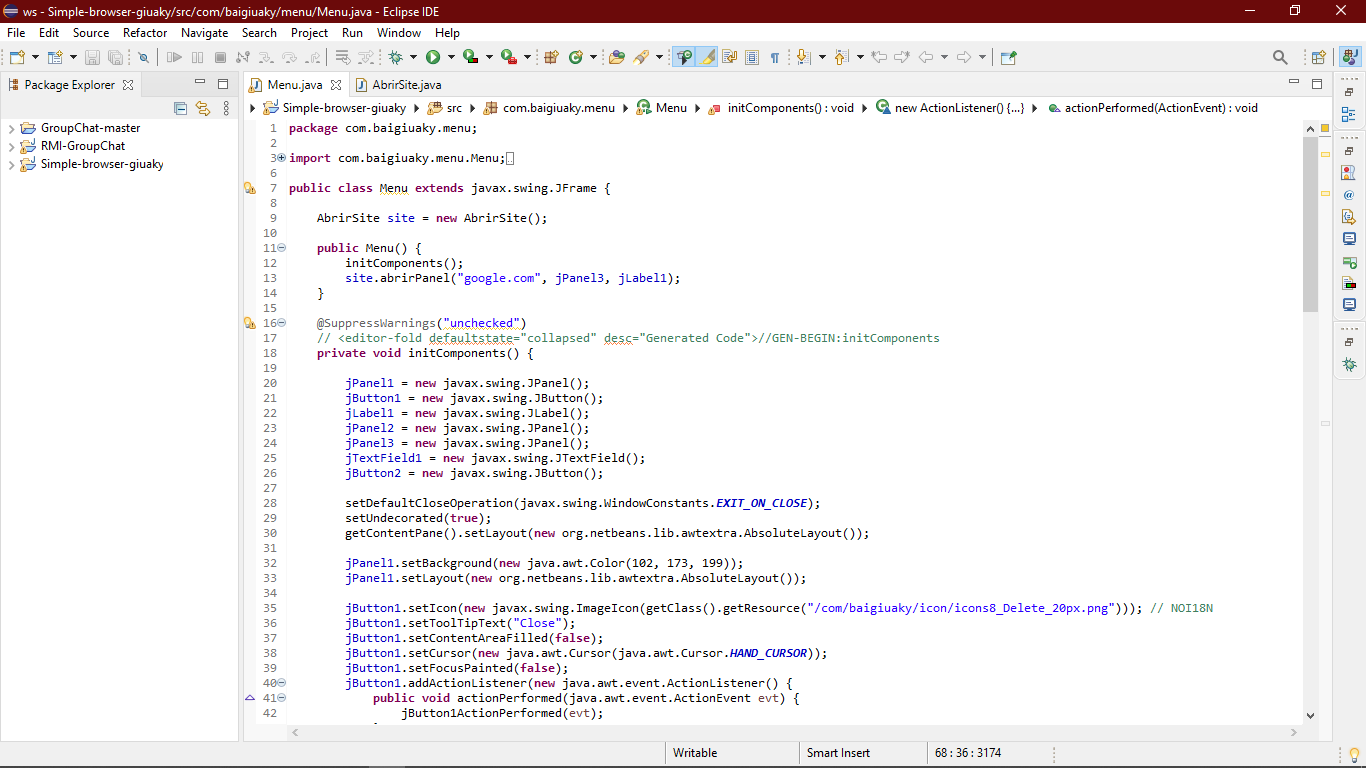
## **2 CÀI ĐẶT**

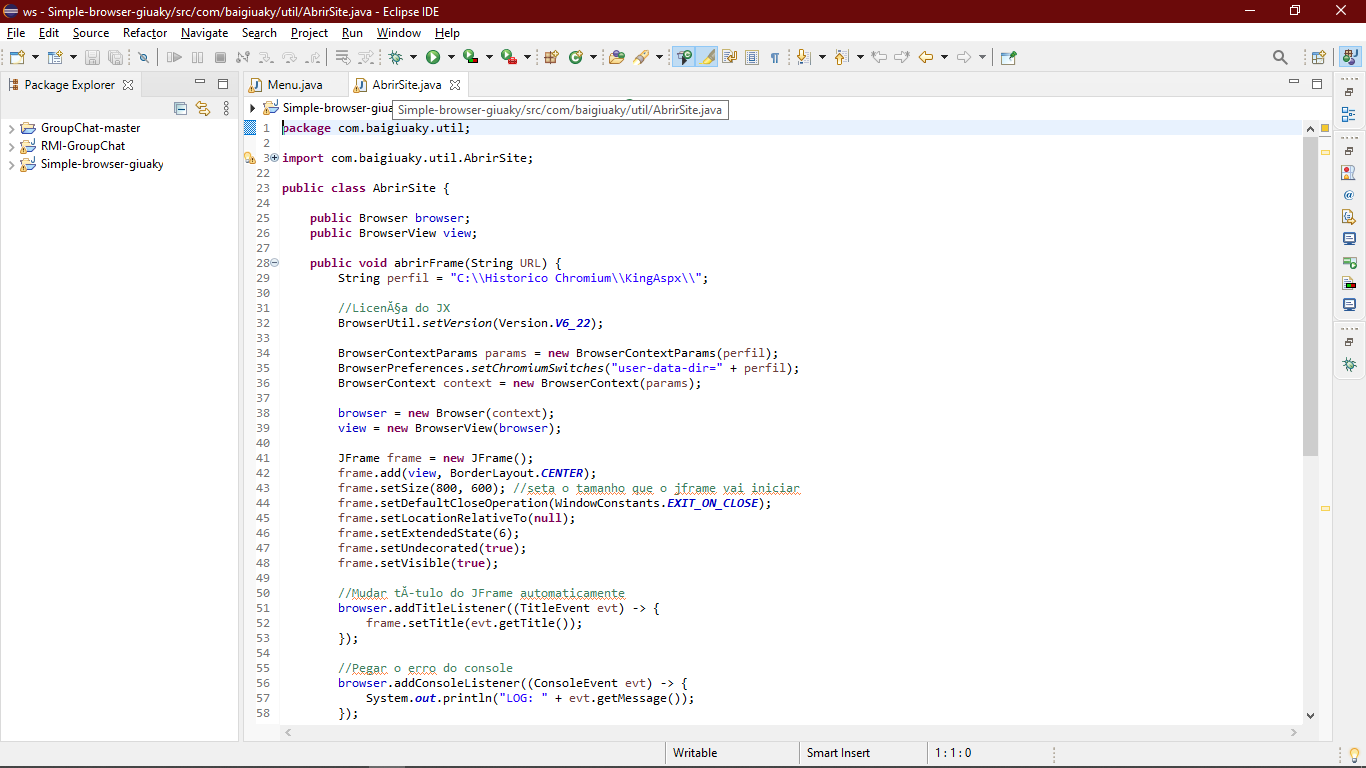
Sử dụng thư viện JXBrowser.

* JXBrowser - Thư viện JXBrowser
* Tài liệu JXBrowser - https://jxbrowser.support.teamdev.com/support/solutions
* Hướng dẫn - https://www.youtube.com/watch?v=7rVChaNN33U

Browser sử dụng hai lớp:

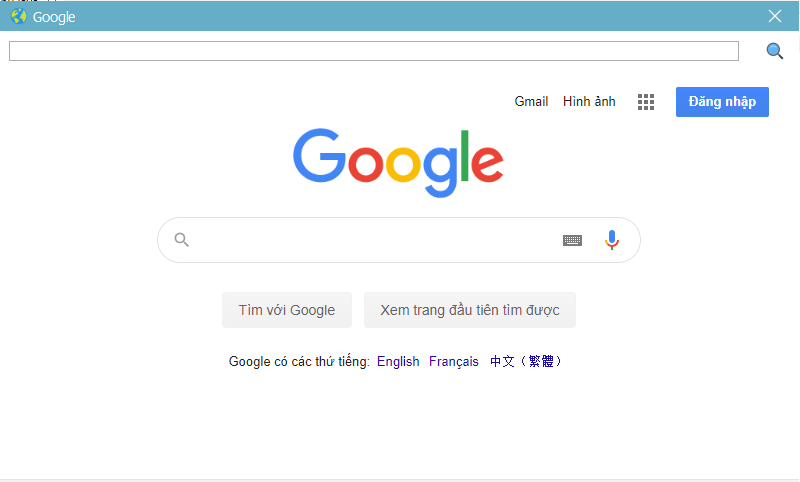
Class Menu: Tôi sẽ tạo một đối tượng JTextField và JEditorPane để phục vụ như một trường địa chỉ và một cửa sổ. Người dùng sẽ nhập URL vào trường địa chỉ và nhấn nút enter và chương trình sẽ dẫn người dùng đến trang web, hiển thị nội dung trong cửa sổ.

CLass AbrirSite: Chứa phương thức chính để chạy các hướng dẫn. Hướng dẫn này được thực hiện với giả định rằng người dùng đã làm theo các hướng dẫn trước đó và quen thuộc với các thuật ngữ mã hóa được sử dụng trong đó, đồng thời có kiến thức lập trình Java cơ bản. 



2.1 Giao diện

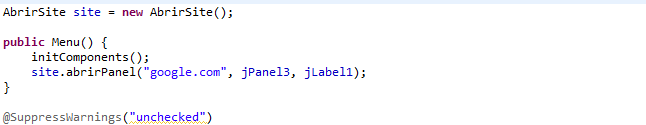
**- giao diện tổng quát:**

****

Hình 2.2. 1 giao diện browser

**Giao diện mặc định:**

Sẽ đặt mặc định google là home của browser và các bạn chỉ cần nhập url ở ô url hoặc tìm kiếm trực tiếp trên google.

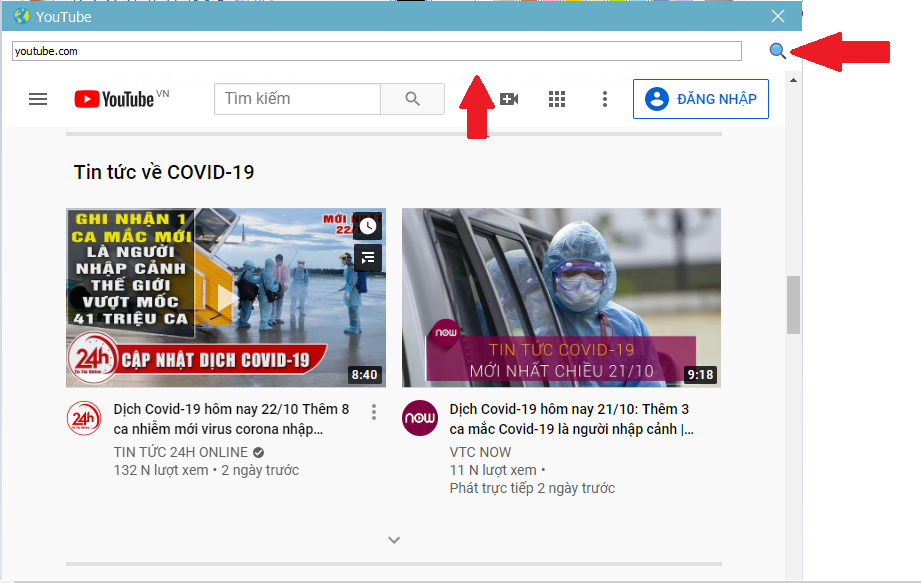


Hình 2.2.1b: Code set homesite

### 2.2 Phương thức tìm kiếm url

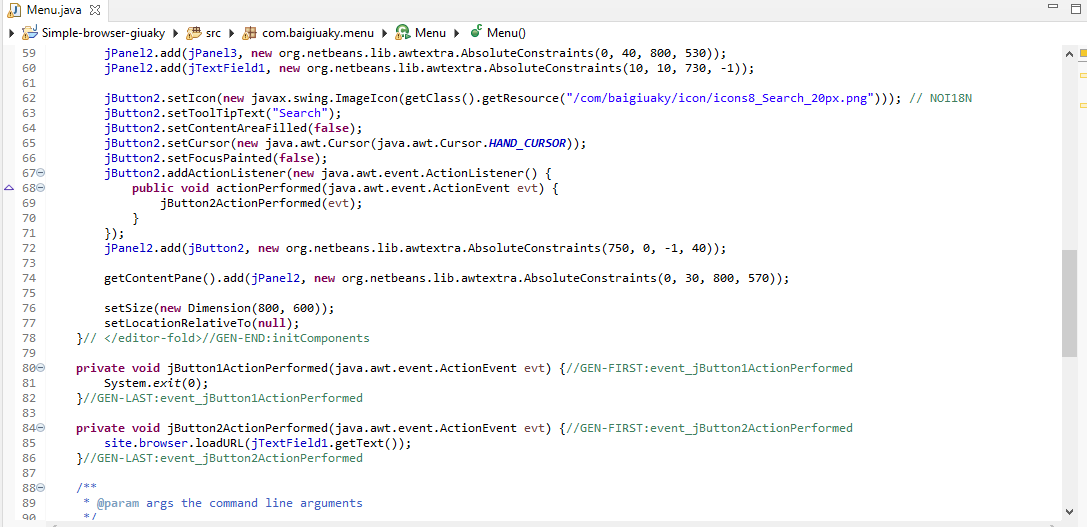
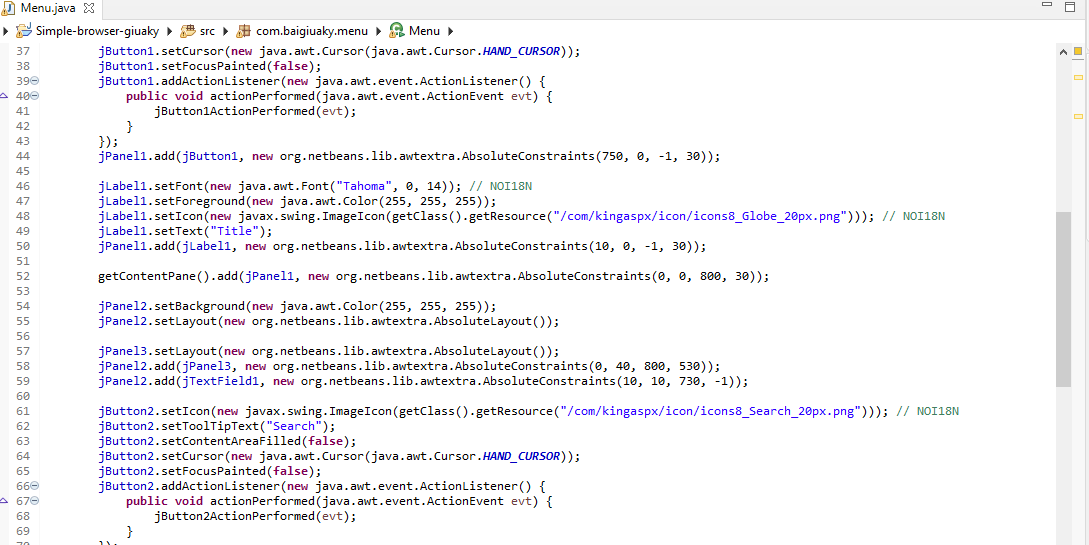


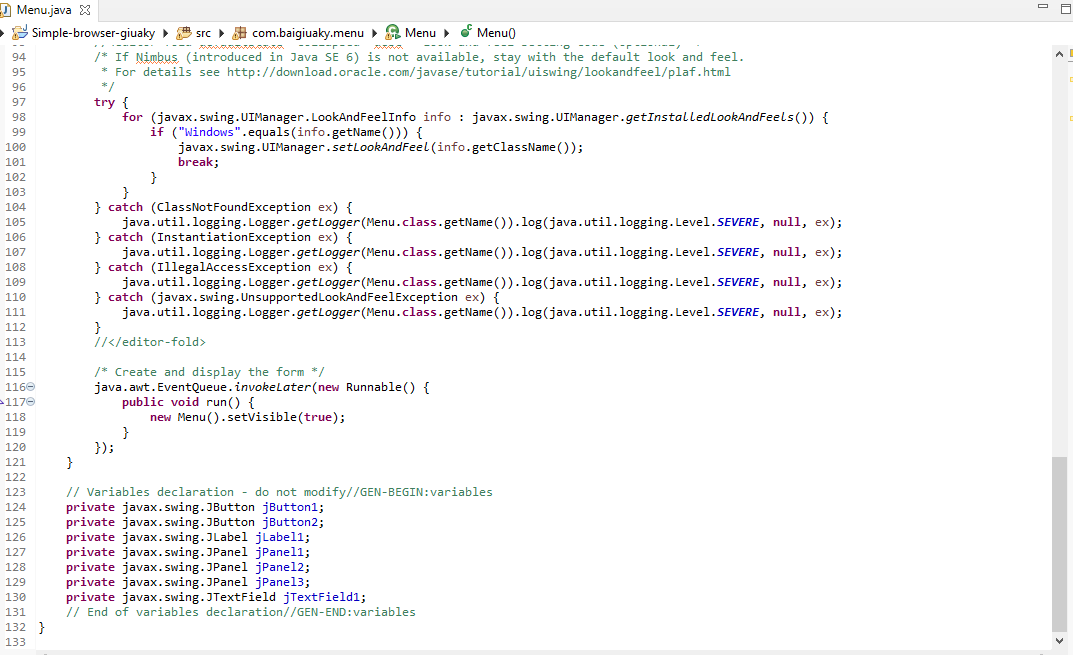
Giao diện ActionListener được liên kết với đối tượng để phát hiện sự kiện khi người dùng nhấn vào icon “SEARCH” ở bên cạnh thanh tìm kiếm. Phương thức actionPerformed được thực hiện.



Hình 2.2. 2 nhập URL và tìm kiếm

### 2.3 Code browser



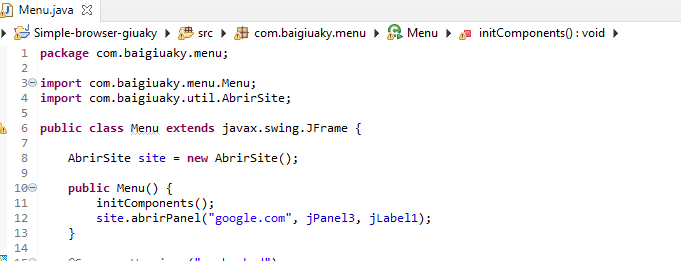


**Hình 2.2. 3 Code**

**Mô tả hoạt động:**

* Code sẽ đặt hiển thị giao diện, vị trí các nút, khung và các button.
* Các button sẽ được gán hoạt động ở đây và set icon.
* Bao gồm cả các private.
* Code sẽ hoạt động khi nhập URL hợp lệ (<https://google.com> hoặc google.com).
* Sẽ báo lỗi nếu nhập URL không hợp lệ (google, youtube,express,…).
* Giao diện của bài sẽ được đặt dưới dạng cửa sổ 1080\*800.

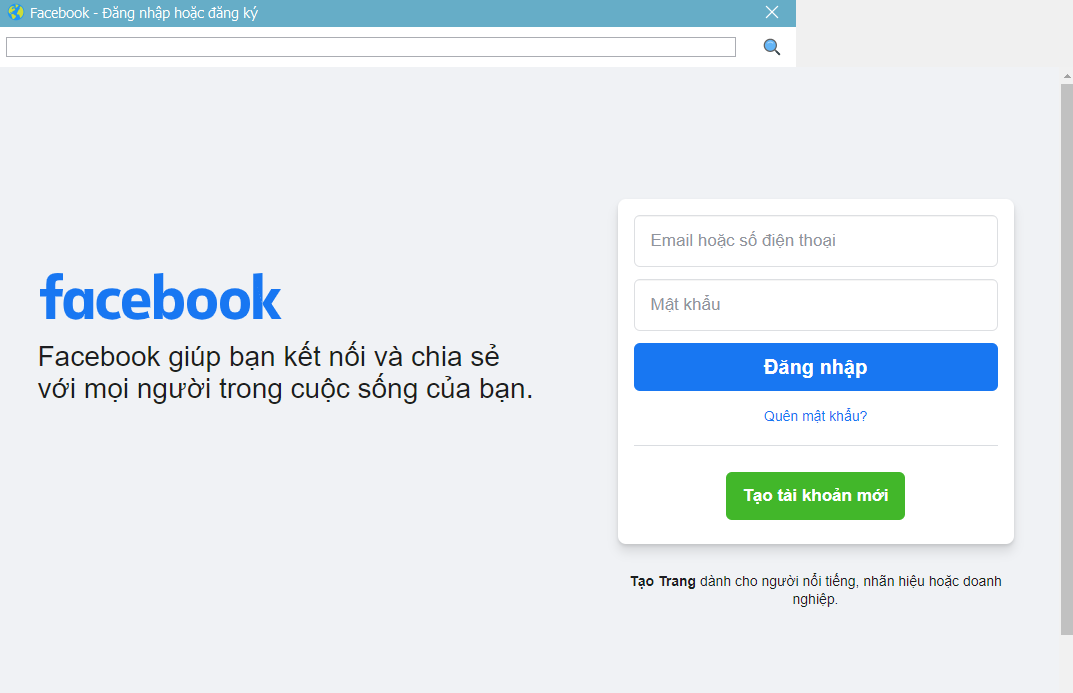
### 2.4 Đặt home cho browser của bạn:



**Hình 2.2. 4 code đặt home cho browser**

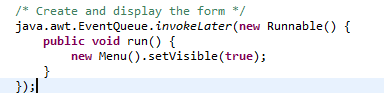
### 2.5 Thay đổi home

**Sau khi thay đổi home thành facebook.com**



**Hình 2.2.5 giao diện home sau khi đã đổi thành facebook**

### 2.6 Tạo và hiển thị biểu mẩu

****

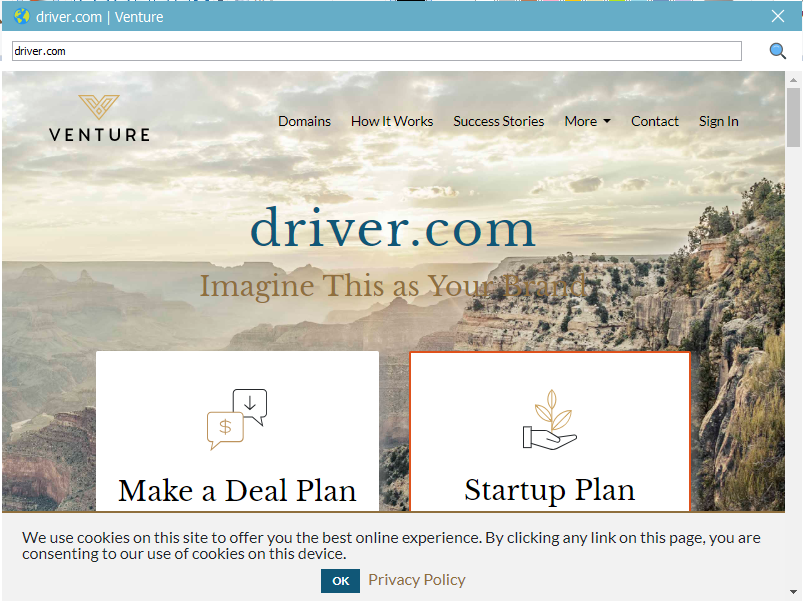
**Hình 2.2.6 tạo và hiển thị biểu mẫu**

# **CHƯƠNG III: DEMO CHƯƠNG TRÌNH**

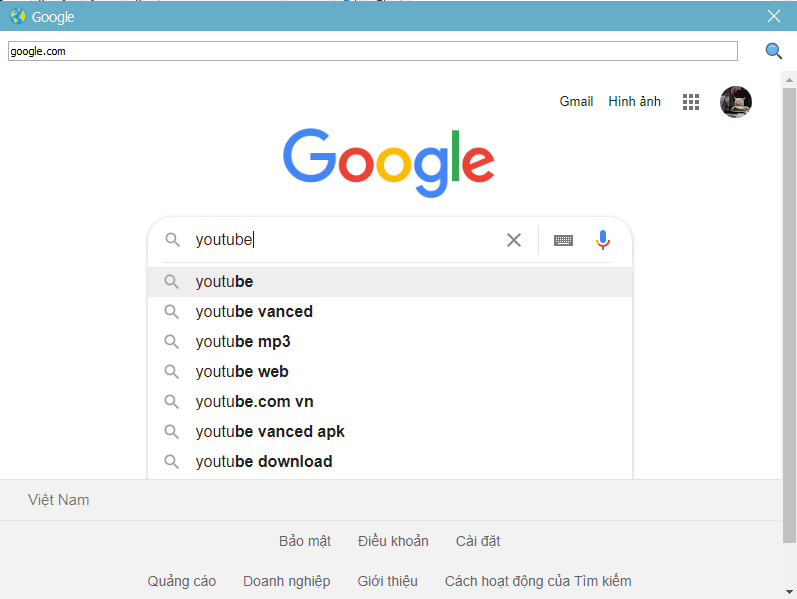
### 1 Chạy demo browser

-Nhập driver.com vào ô tìm kiếm url, sau đó ấn vào icon search.

-Hiển thị ở bên dưới là kết quả của tìm kiếm.



-Hoặc có thể tìm kiếm ở home browser (google) có thể đáp ứng được tất cả các chức năng cơ bản của các browser khác.

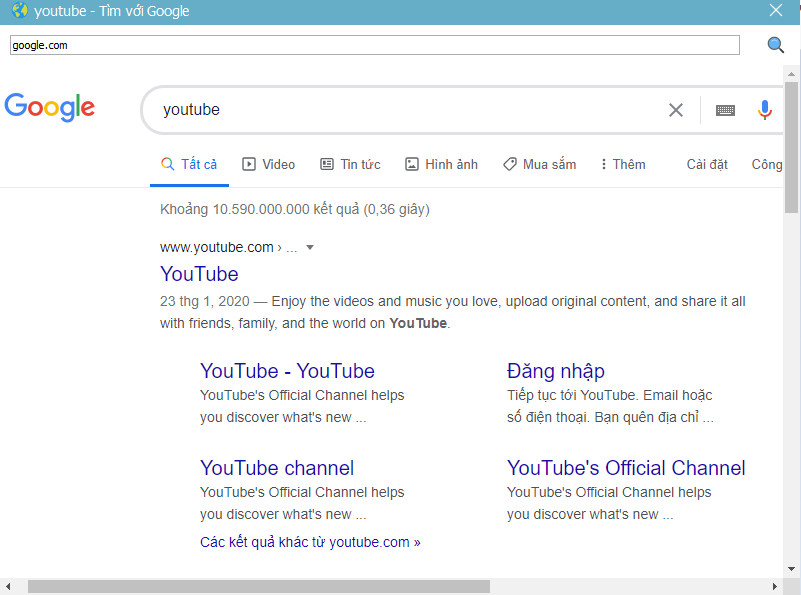


Chạy các bài kiểm tra

Nếu bạn muốn có một URL cố định, đừng đặt Chuỗi làm tham số, hãy thay đổi Chuỗi thành URL thực, ví dụ: www.google.com. Nếu bạn muốn tạo thứ gì đó tương tác, hãy sử dụng Chuỗi.

//browser.loadURL(URL);

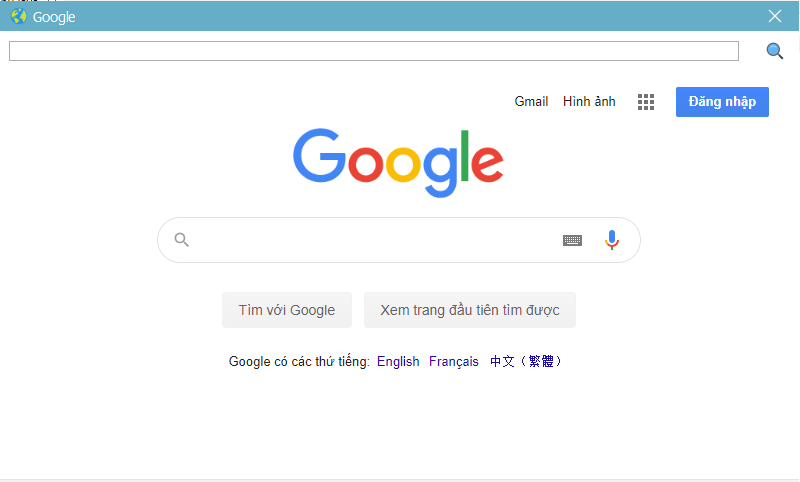
browser.loadURL("www.google.com");

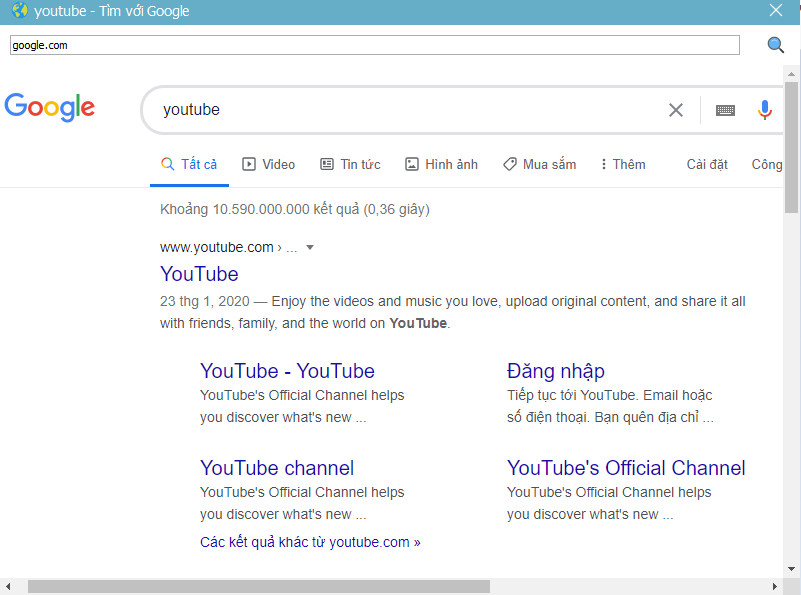


* Đây là kết quả sau khi gọi ra google.com và tìm kiếm trong google.
* Vẫn hoạt động bình thường như các browser khác.
* Đã thử tìm kiếm bằng yahoo.com hay streuser.com và hoặt động rất tốt.

## 2 Kết quả đạt được

Sau khi hoàn thành ứng dụng đã có thể hoạt đọng như 1 browser bình thường khác. Có thể chạy tìm kiếm URL hoặc tìm kiếm trên home browser (google) có thể tìm kiếm và hiển thị các kết quả lên khu vực hiển thị kết quả.

****



## Đánh giá

* 1. Ưu điểm.

Về mặt cơ bản thì ứng dụng đã đáp ứng phần nào được những tính năng cần thiết của một browser đơn giản và có thể hoạt động tốt mà không gặp bất kỳ lỗi nào. Thêm vào đó là ứng dụng không có bất cứ yêu cầu nào quá cao về phần cứng lẫn phần mềm, nếu so sánh sang với các đàn anh như google chrome, opera,…. Thì ứng dụng tiêu tốn rất ít hiệu năng của ram và bộ nhớ, chỉ bằng 1/5 so với các đàn anh đã nêu trên. Nhưng vẫn có 1 vài điểm chưa tốt.

* 1. Hạn chế

Như đã nêu ở trên, mặc dù hoạt động rất tốt nhưng vẫn có một vài hạn chế mà cần thêm thời gian để phát triển và khắc phục thêm. Đây là một vài các hạn chế dễ nhận biết nhất của dự án:

* Giao diện chưa đẹp mắt
* Chưa có một số tính năng đặc biệt
* Ứng dụng chưa có tính chuyên nghiệp cao
* Hơi đơn giản và chưa có đặc điểm riêng.

Có thể vẫn còn một vài hạn chế khác mà sau khi trải qua một thời gian sử dụng mới có thể phát hiện ra. Nhưng đây là dự án đầu tiên của em có mang tính chất phát triển, em mong sau này từng bước một sẽ cố gắng hoàn thiện và cho ra sản phẩm có thể mang đến nhiều tiện ích và hữu dụng cho cuộc sống giúp mọi người và bản thân. Có thể giúp đỡ mọi người không còn khó khăn trong việc tìm kiếm thông tin và kết nối trong qua dự án của em.

# CHƯƠNG IV: KẾT LUẬN

Tuy đây là kết quả của em sau 1 thời gian học tập, tìm hiểu và tự phát triển và chưa được bắt mắt hay có các tính năng đặc trưng và mở rộng của 1 browser nhưng em sẽ cố gắng để hoan thiện nhất có thể để browser của em có thể hoạt động tốt và không gặp bất cứ vấn đề nào, em vẫn còn cần phải học hỏi thêm tuy nhiên em rất tự hào vì có thể tạo ra 1 phần mềm có tính ứng dụng thực tế như thế này.

Trong thời gian tới, em sẽ tiếp tục phát triển ứng dụng để giúp ứng dụng được hoàn thiện tốt hơn. Sẽ phát triển thêm mảng giao diện và một vài tính năng đặc biệt khác để có thể chính thức là một browser có thể đi vào hoạt động. Em sẽ cố tạo ra một vài tính năng đặc trưng mà chỉ browser của em mới có, tuy rất khó khăn nhưng em vẫn sẽ cố để đạt được kết quả mà bản thân mong muốn.

Đây là một chương trình ứng dụng thực tế lớn và phức tạp, cần có một nhóm các chuyên gia cùng thực hiện. Vì vậy trong khuôn khổ đề án này chỉ có thể đưa ra những chức năng cơ bản, mô phỏng một phần nhỏ yêu cầu của thực tế. Rất mong nhận được sự đóng góp ý kiến của các thầy cô giáo và bạn bè để chương trình được tiếp tục hoàn thiện, đáp ứng một cách tốt nhất như là 1 browser đơn giản.

***Em xin chân thành cảm ơn!***