

4

Lab

Truyền thông với Web Server trong C#

Working with Web Server in C#

Thực hành Lập trình mạng căn bản - NT106
Giảng viên biên soạn: ThS. Đỗ Thị Hương Lan

Tháng 03/2023
Lưu hành nội bộ

A. TỔNG QUAN

1. Mục tiêu

- Nắm được cách giao tiếp với Web Server trong C# và vận dụng để viết các ứng dụng gửi, nhận dữ liệu, truyền thông với Web Server
- Làm quen, tương tác với Web Service, Web API

2. Môi trường/Công cụ

- Môi trường: IDE Microsoft Visual Studio

3. Liên quan

- Các kiến thức nền tảng về lập trình, ngôn ngữ lập trình C#, Windows Forms Application
- Các kiến thức nền tảng về cơ chế hoạt động của Web Client, Web Server, Web Service, Web API
- Tham khảo tài liệu (Mục E) để có kiến thức cơ bản về C#, Winforms

B. CÀI ĐẶT VÀ VẬN HÀNH WEB SERVER VỚI IIS

Yêu cầu:

Cài đặt một Web Server và publish một trang web đơn giản

Lưu ý: Ví dụ sau đây sử dụng **IIS (Internet Information Services)** để dựng một Web Server. Sinh viên có thể sử dụng các giải pháp thay thế như sau: Apache, Nginx, v.v...

1. Cách tạo một Website đơn giản chạy trên localhost

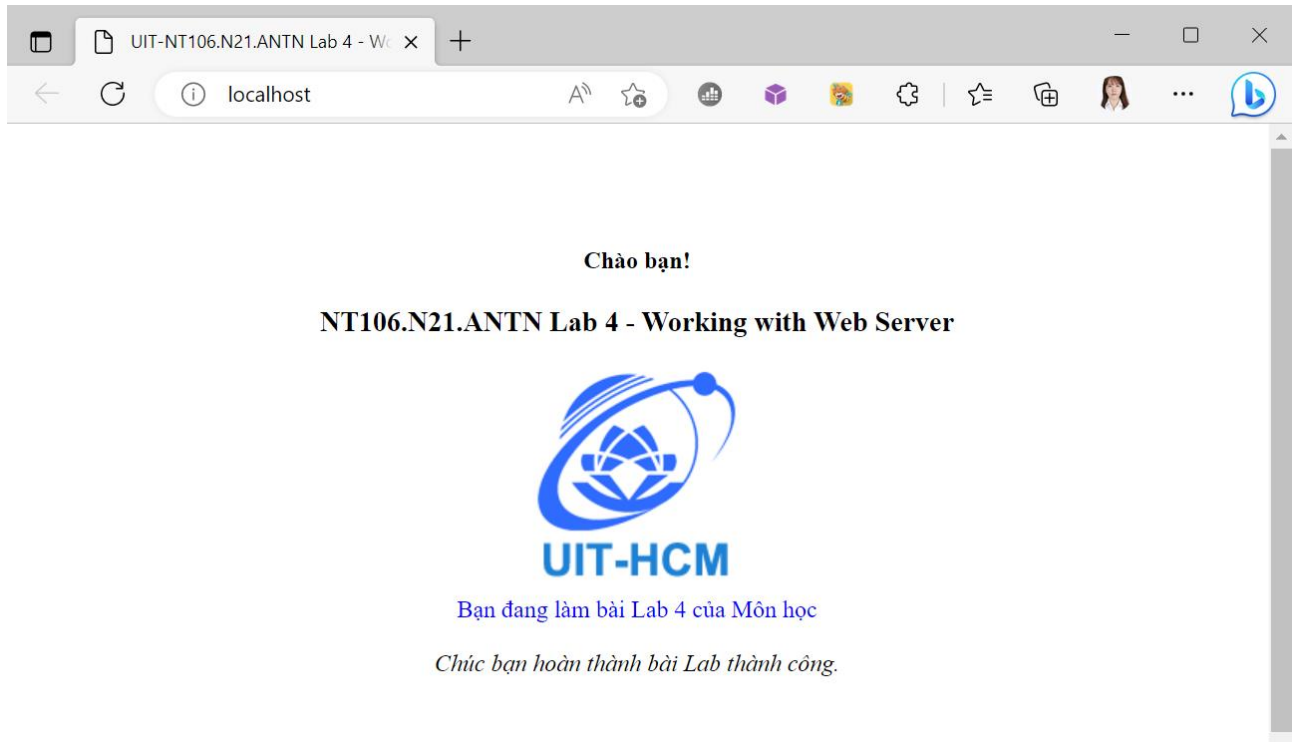
Bước 1: Mở một IDE hỗ trợ soạn thảo, viết mã nguồn HTML (Notepad, Visual Studio ...), và tạo một file với đuôi là .html. Ví dụ: index.html

Bước 2: Viết mã nguồn để tạo một Website đơn giản. Tham khảo file bên dưới

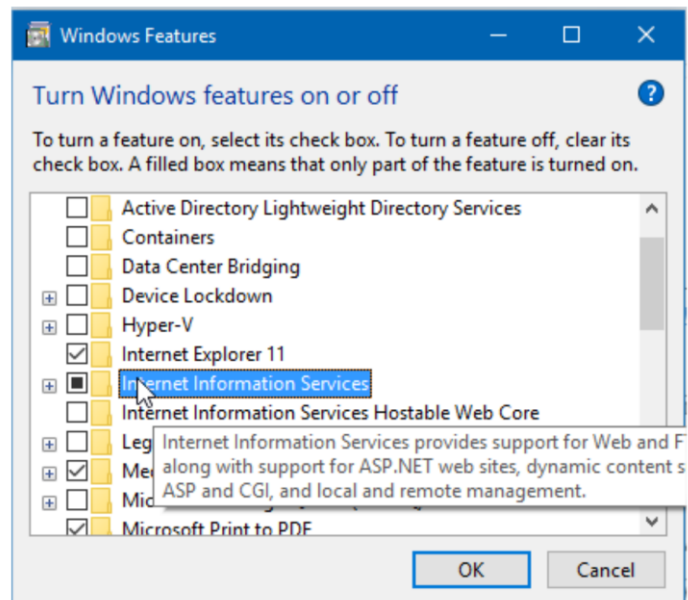
Lưu ý: Sinh viên có thể tìm hiểu thêm về HTML tại: <http://www.w3schools.com/> hoặc tự tạo một trang web khác tương đương bằng HTML, CSS.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>UIT-NT106.N21.ANTN Lab 4 - Working with Web Server</title>
  <meta name="description" content="Lab4" />
  <meta name="keywords" content="Lab4" />
</head>
<body class='pushmenu-push' id="page">
  <div class="dialog" style="width:60%;margin:10% auto;text-align:center;">
    <div class="dialogBox">
      <h4>Chào bạn!</h4>
      <h3>NT106.N21.ANTN Lab 4 - Working with Web Server</h3>
      
      <div style="color:blue;">Bạn đang làm bài Lab 4 của Môn học</div>
      <br/>
      <i>Chúc bạn hoàn thành bài Lab thành công.</i>
    </div>
  </div>
</body>
```

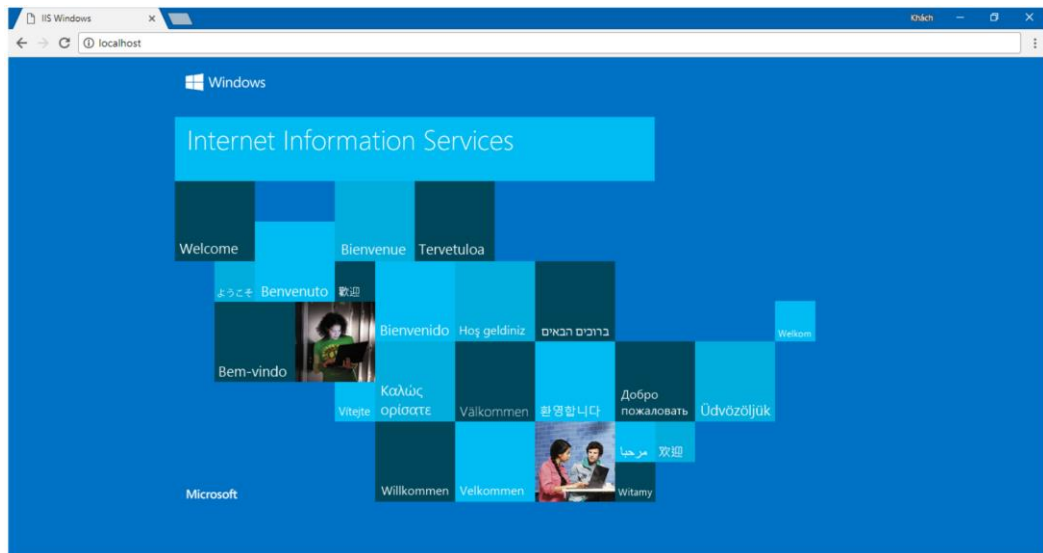
Đoạn mã nguồn trên, khi chạy trên trình duyệt sẽ hiển thị dưới dạng như sau:



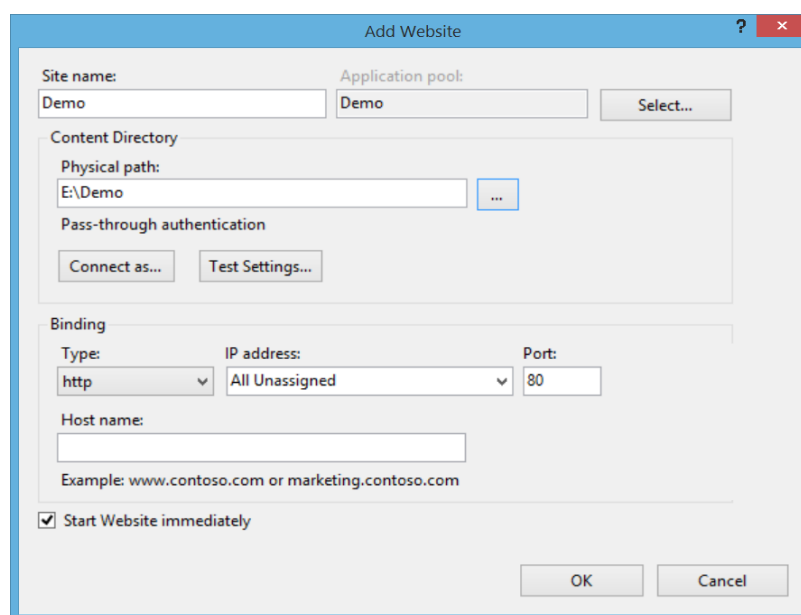
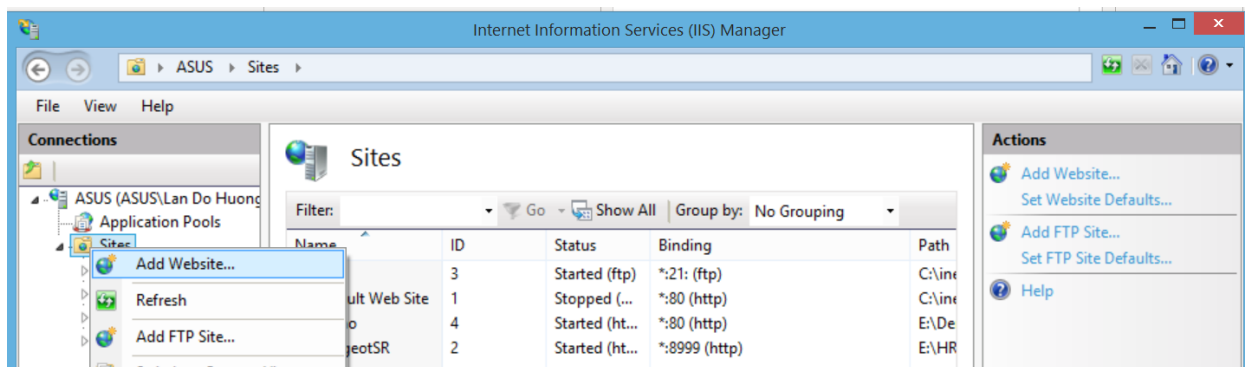
Bước 3: Bật dịch vụ **IIS** (dịch vụ để tạo máy chủ Web/FTP,...) trên **Windows 7/8/10** với các bước như sau: **Control Panel -> Programs and features -> Turn Windows Features on or off -> Chọn Internet Information Service -> OK**



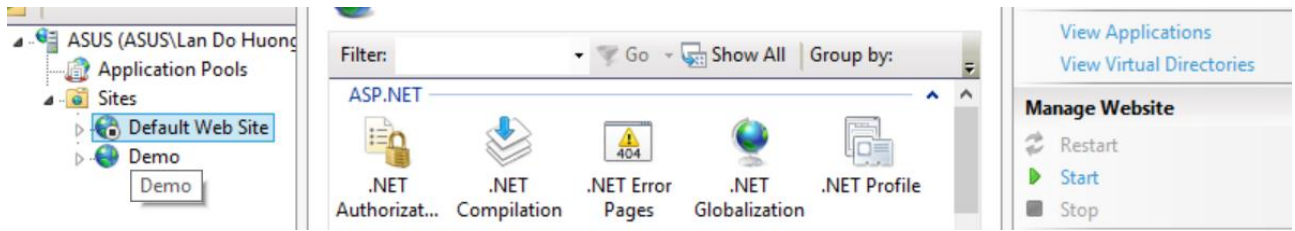
Bước 3: Sau khi bật dịch vụ IIS, kiểm tra dịch vụ đã hoạt động hay chưa bằng cách sử dụng một trình duyệt bất kỳ, và gõ **localhost**, nếu một trang Web mặc định của IIS hiện ra là thành công.



Bước 4: Lúc này ta đã có một máy chủ web đang hoạt động ở địa chỉ localhost. Mở IIS, để chạy một website trên địa chỉ này, nhấp chuột phải vào Site à Add Website. Với Physical path là tên folder chứa file html đã tạo ở Bước 1,2



Lưu ý: Dừng chạy (Stop) trang mặc định của IIS và Chạy (Start) trang vừa khởi tạo để địa chỉ localhost chạy web site mới



Bước 5: Truy cập vào trang web với đường dẫn <http://localhost/index.html> hoặc <http://localhost>. Truy cập thử trang web của sinh viên khác từ trình duyệt bằng các gõ URL như sau: <http://A.B.C.D/index.html> với A.B.C.D là địa chỉ IP của máy tính mà bạn mình sử dụng. Lưu ý: sinh viên và bạn mình phải sử dụng chung mạng Local (LAN)

C. NỘI DUNG THỰC HÀNH

1. Bài 01 – Chương trình tải, hiển thị nội dung và header

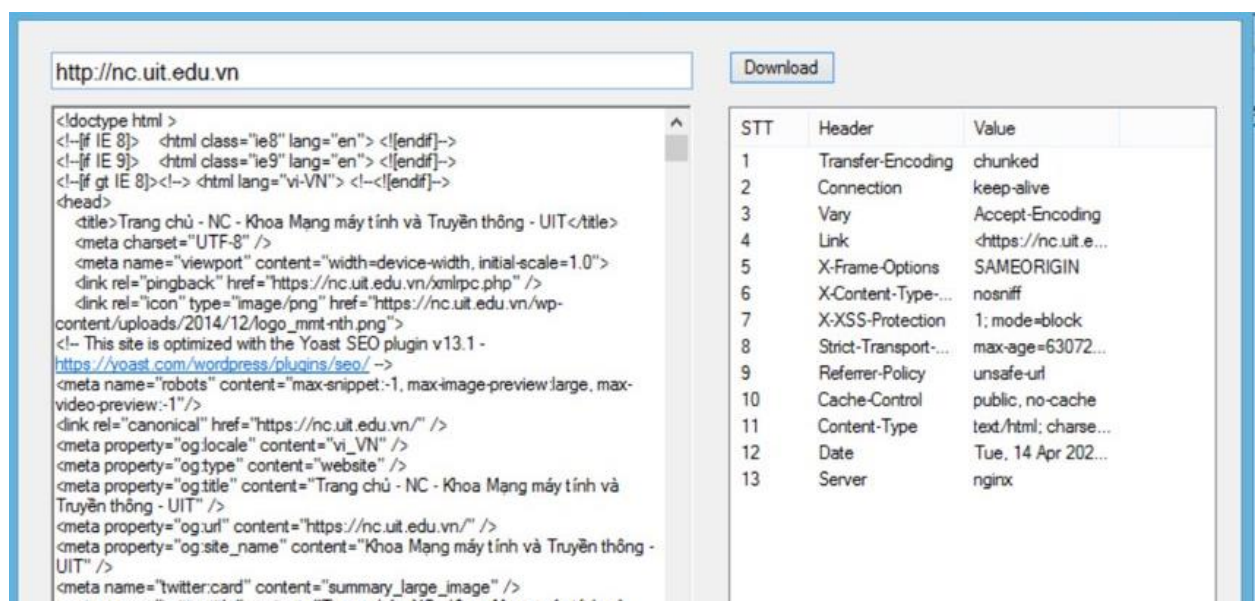
Yêu cầu:

Viết chương trình hiển thị nội dung HTML của một Website và các trường header kèm theo mà Server phản hồi

Yêu cầu chi tiết:

- Cho phép nhập URL trang web cần tải nội dung
- Nhấn nút “Download” để lấy nội dung HTML và Header về
- Hiển thị nội dung HTML
- Hiển thị các trường header (Name, Value) thành danh sách (Bảng/Grid/List)

Giao diện tham khảo:



Gợi ý: Có thể sử dụng lớp TCPClient hoặc các lớp WebRequest, WebClient hoặc HttpClient

Cách 1: Sử dụng WebRequest và WebResponse

- Khởi tạo WebRequest với đường link web cần làm việc
- Dùng Stream để đón nhận dữ liệu từ WebResponse trả về
- Tải nội dung thông qua phương thức đọc từ Stream. Ví dụ: ReadToEnd()
- Tải Header kèm theo dựa vào thuộc tính WebResponse.Headers

Tham khảo thêm tại:

<https://learn.microsoft.com/en-us/dotnet/api/system.net.webresponse.headers?view=net-7.0>

Code tham khảo: Hàm lấy nội dung sử dụng WebRequest

```
private string getHTML(string szURL)
{
    // Create a request for the URL.
    WebRequest request = WebRequest.Create(szUrl);
    // Get the response.
    WebResponse response = request.GetResponse();
    // Get the stream containing content returned by the server.
    Stream dataStream = response.GetResponseStream();
    // Open the stream using a StreamReader for easy access.
    StreamReader reader = new StreamReader(dataStream);
    // Read the content.
    string responseFromServer = reader.ReadToEnd();
    // Close the response.
    response.Close();
    return responseFromServer;
}
```

Cách 2: Sử dụng WebClient

- Dùng phương thức DownloadData
- Lấy ResponseHeader dưới dạng một WebHeaderCollection

```
byte[] response = myClient.DownloadData(url);
richTextBox1.Text = Encoding.UTF8.GetString(response);
WebHeaderCollection whc = myClient.ResponseHeaders;
```

Nâng cao:

Cấu hình IIS, che giấu/làm giả một số thông tin

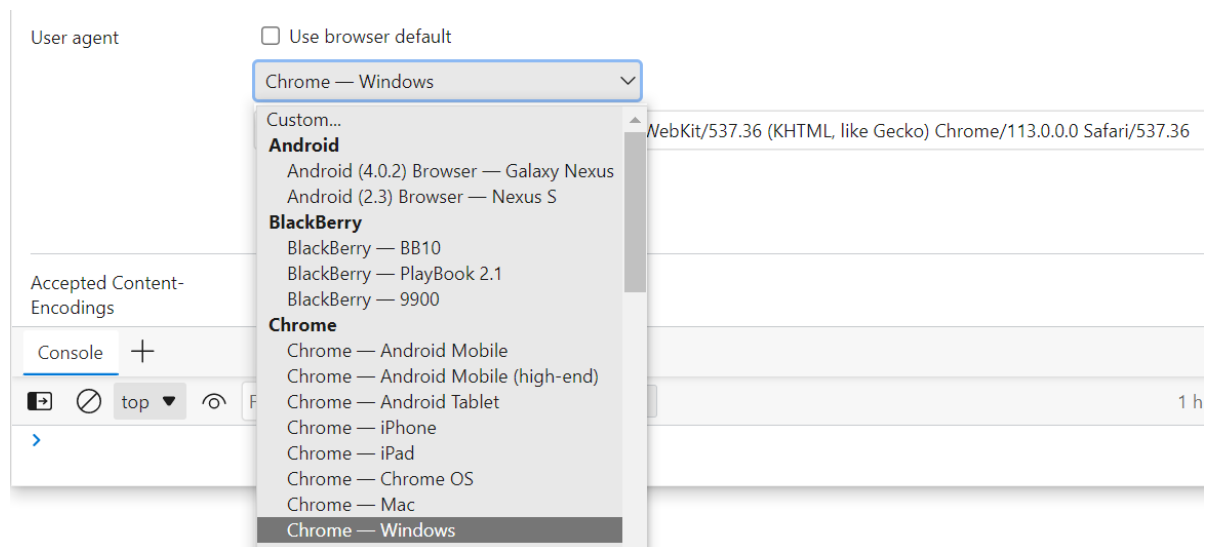
2. Bài 02 – Chương trình xem web kèm tiện ích giả lập – Device Emulator

Yêu cầu:

Viết chương trình xem nội dung trang web bất kỳ, tích hợp tính năng cho phép giả lập chuyển đổi kích thước màn hình, loại thiết bị

Yêu cầu chi tiết:

- Cho phép xem nội dung Website (Raw HTML/Render HTML)
- Viết tiện ích User-Agent Switcher: cho phép người dùng chọn loại nền tảng giả lập



Gợi ý: Tham khảo thêm về các chuỗi User Agent

- Sử dụng công cụ **Dev Tools** của các trình duyệt (Ví dụ: Chrome, Edge, v.v...).
- Tại **Network conditions**, bỏ chọn “Select automatically” bên cạnh User Agents. Chọn một User Agent từ danh sách để tham khảo chuỗi User Agent tương ứng

Ví dụ:

Safari - Iphone	Mozilla/5.0 (iPhone; CPU iPhone OS 13_2_3 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/13.0.3 Mobile/15E148 Safari/604.1
Safari - Ipad	Mozilla/5.0 (iPad; CPU iPhone OS 13_2_3 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/13.0.3 Mobile/15E148 Safari/604.1
Chrome - Windows	Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 Safari/537.36

3. Bài 03 – Làm việc với API/Web Service

Yêu cầu:

Viết chương trình làm việc/tương tác với WebAPI/Service để lấy dữ liệu

Yêu cầu chi tiết:

- Lấy dữ liệu cần (với các định dạng như XML, JSON, v.v ...) từ API, Webservices
- Trích xuất dữ liệu, xử lý và thiết kế giao diện để hiển thị thông tin một cách trực quan

Gợi ý:

- Sử dụng Free API <https://jsonplaceholder.typicode.com/>
- Lấy và hiển thị danh sách 100 photos/100 comments/10 users

Tham khảo thêm tại:

<https://learn.microsoft.com/en-us/dotnet/fundamentals/networking/http/httpclient>

Nâng cao:

Tự viết/xây dựng và vận hành Web Service/API

Tham khảo thêm tại:

- Web Service với C#:

<https://docs.microsoft.com/en-us/troubleshoot/developer/visualstudio/csharp/general/write-web-service>

- Web API với C#:

<https://docs.microsoft.com/en-us/aspnet/web-api/overview/getting-started-with-aspnet-web-api/tutorial-your-first-web-api>

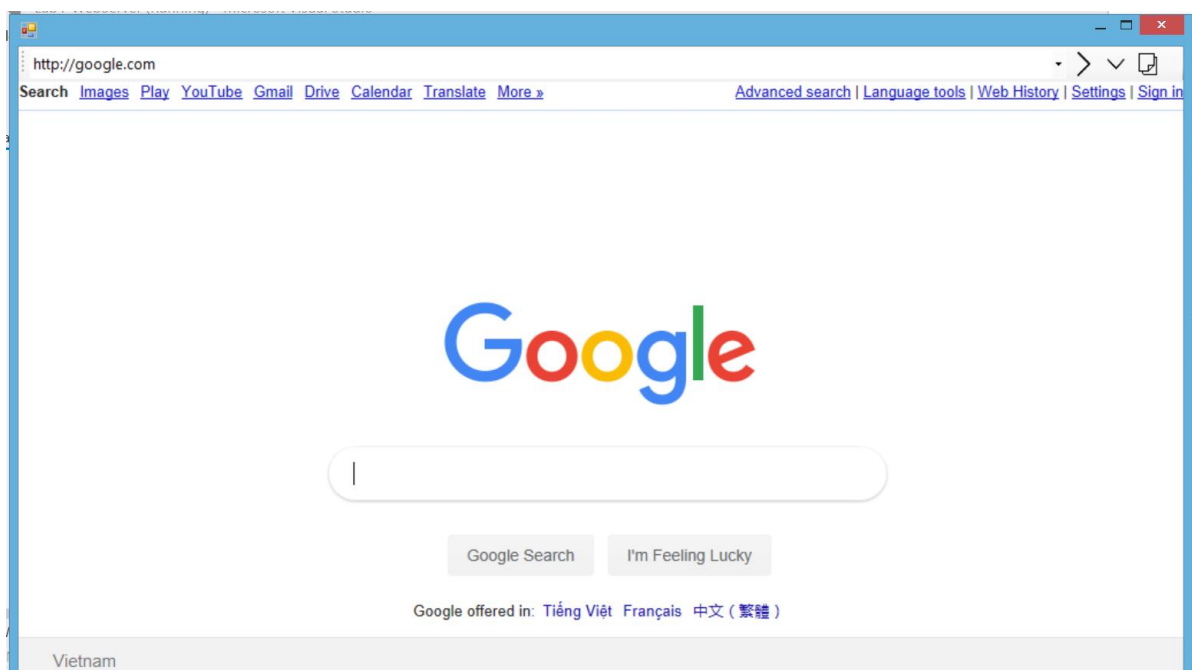
4. Bài 04 – Web Browser cơ bản

Yêu cầu:

Viết chương xem hoạt động như một web browser cơ bản

Yêu cầu chi tiết:

- Xem nội dung Website (Render HTML)
- Download Source (Lưu trữ mã nguồn)
- Xem Source (Xem mã nguồn), Header Response, cấu hình thông tin Header Request

Giao diện minh họa:**Gợi ý:**

- Có thể sử dụng Control WebBrowser để hiển thị nội dung của Website (Render)
- Tính năng Download: tải toàn bộ Source, bao gồm hình ảnh, các file liên quan bằng cách sử dụng **HTMLAgilityPack** để xử lý các file dựa trên các thẻ HTML
- Hiển thị cửa sổ Xem Source bằng cách tạo một Form mới và parse dữ liệu sang, hiển thị Form với dữ liệu
- Để sử dụng HTMLAgilityPack, sử dụng Nuget Packet để thêm vào Project. Tham khảo cách thêm và sử dụng một packet tại:

<https://docs.microsoft.com/en-us/nuget/quickstart/install-and-use-a-package-in-visual-studio>

D. YÊU CẦU & ĐÁNH GIÁ

1. Yêu cầu

- Sinh viên thực hành và nộp bài **cá nhân/nhóm** theo thời gian quy định.
- Bài nộp: **Source code** (nén) và 1 **File .pdf** báo cáo tóm lược các bài tập liên quan kèm hình ảnh minh họa

Toàn bộ project đặt vào 1 file nén (.rar/.zip)	File .PDF screenshot
LabX-MSSV-HọTênSV	LabX-MSSV-HọTênSV.pdf
Ví dụ: Lab1-16520901-NguyenVanA	Ví dụ: Lab1-16520901-NguyenVanA.pdf

1. Đánh giá kết quả

- Tiêu chí đánh giá:
 - Chương trình chạy được, hoàn thành các yêu cầu cơ bản: **+70%**.
 - Có kiểm tra các điều kiện ràng buộc khi nhập dữ liệu, code “sạch” [2], đặt tên biến rõ ràng: **+30%**.
 - Nộp bài không đầy đủ; lỗi, không chạy được; nộp trễ; sao chép code bạn khác, nguồn có sẵn: *xử lý tùy theo mức độ* (- **10 → 100%**)

E. THAM KHẢO

[1] Microsoft (2018). C# Guide. [Online] Available at: <https://docs.microsoft.com/en-us/dotnet/csharp/>

[2] Martin, R. C. (2009). *Clean code: a handbook of agile software craftsmanship*. Pearson Education.

HẾT