

BÁO CÁO THỰC HÀNH

Môn học: Lập trình mạng căn bản

Kỳ báo cáo: Buổi 04 (Session 04)

Tên chủ đề:

GVHD: Nghi Hoàng Khoa

Ngày báo cáo: 03/04/2022

Nhóm: NH (ghi số thứ tự nhóm)

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT106.M21.ATCL

STT	Họ và tên	MSSV	Email
1	Võ Khoa Nam	19521877	19521877@gm.uit.edu.vn
2	Trịnh Minh Hoàng	19521547	19521547@gm.uit.edu.vn
3	Phạm Trung Hiếu	19521512	19521512@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

STT	Công việc	Kết quả tự đánh giá
1	Hiển thị nội dung HTML của một trang web bất kỳ	100%
2	Viết chương trình gửi dữ liệu bất kỳ lên địa chỉ	100%
3	Download nội dung trang web	100%
4	Viết chương trình hoạt động như Web Browser	100%

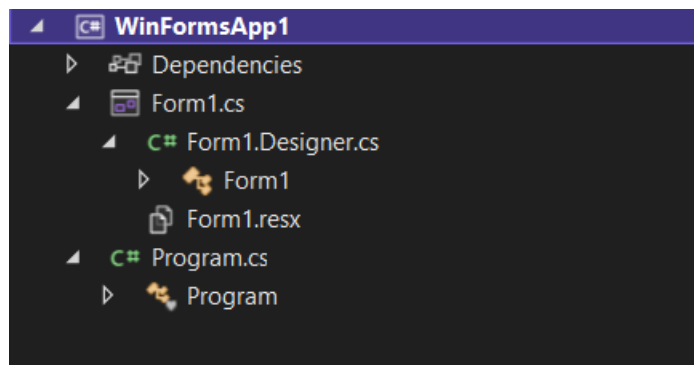
Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

¹ Ghi nội dung công việc, các kịch bản trong bài Thực hành

BÁO CÁO CHI TIẾT

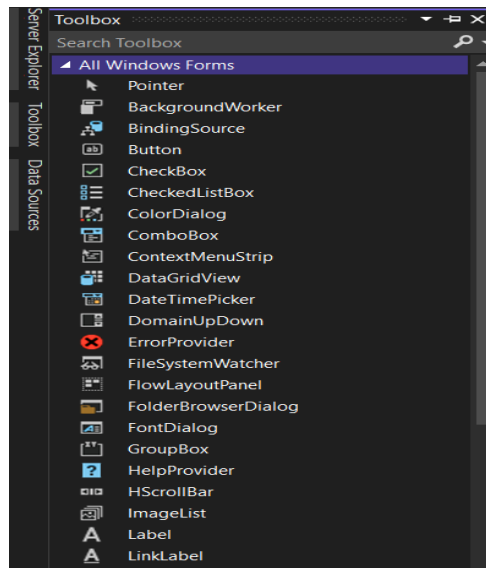
1. Hiện thị nội dung của một trang web bất kỳ

- Tài nguyên:
- Mô tả/mục tiêu:
 - Mô tả: Sử dụng WebRequest và WebResponse để hiển thị nội dung của một trang web bất kỳ
 - Mục tiêu: hiển thị được nội dung của một trang web ra màn hình
- Các bước thực hiện/ Phương pháp thực hiện (Ảnh chụp màn hình, có giải thích)
 - Bước 1: Tạo project trong Visual Code bằng cách chọn File -> New -> Project -> Visual C# -> Windows Form Application. Đặt tên cho cho project.

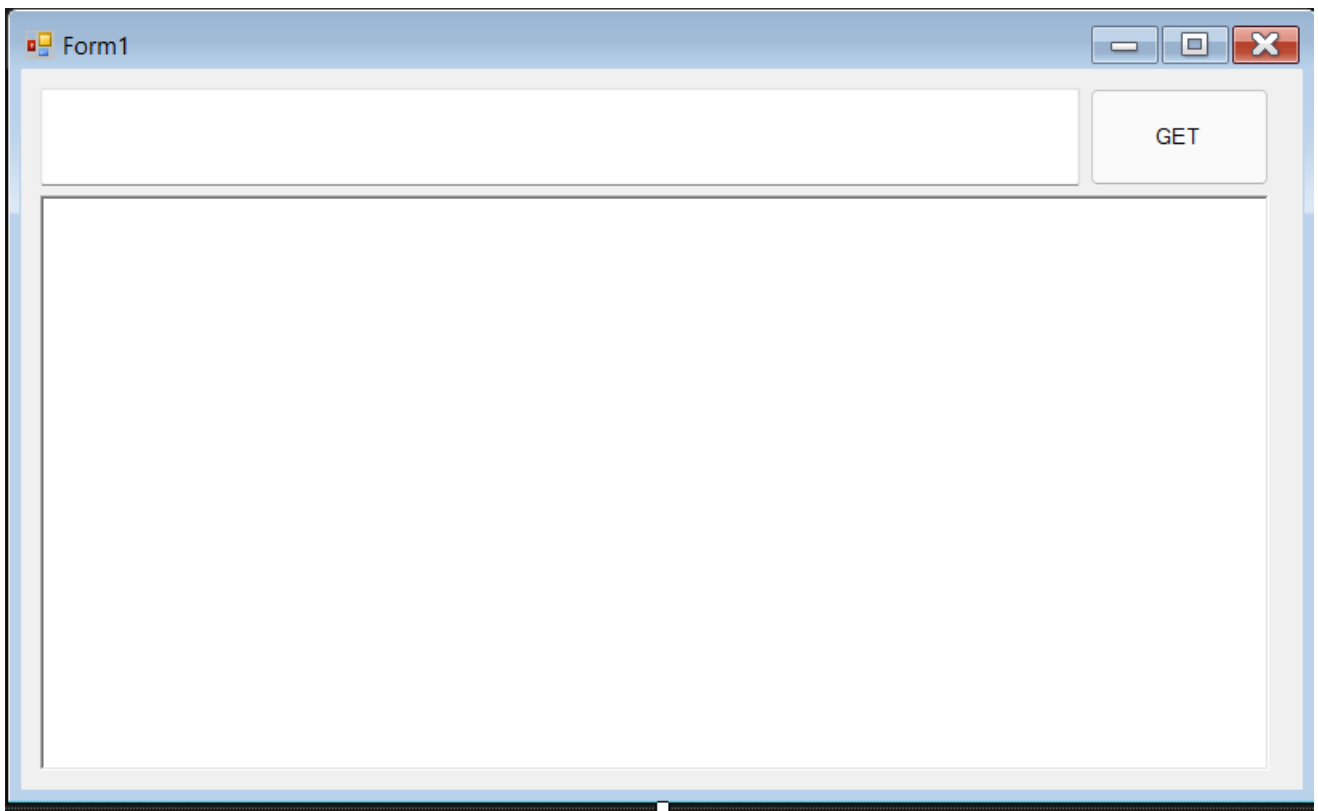


Hình 1: Sau khi tạo project sẽ được tổ chức gồm các file trong hình.

- Bước 1: Tạo form giao diện theo yêu cầu bằng cách sử dụng toolbox trong Visual Studio. Sử dụng button và richTextBox để tạo ra giao.



Hình 2: Thanh toolbox chứa các thẻ label, input text, button, ... Để tạo giao diện form



Hình 3: Giao diện windowsform của ứng dụng.

- Bước 2: Lập trình sự kiện và chức năng theo yêu cầu bài toán gửi và nhận dữ liệu.
 - o Để lấy được nội dung của một trang web bằng URL n click vào nút “GET” trên giao diện. Vì vậy ta phải bắt sự kiện click cho button này.
 - o Nhấp đúp chuột vào button “GET”. Chương trình sẽ khởi tạo 1 hàm tương ứng với sự kiện click vào button này.

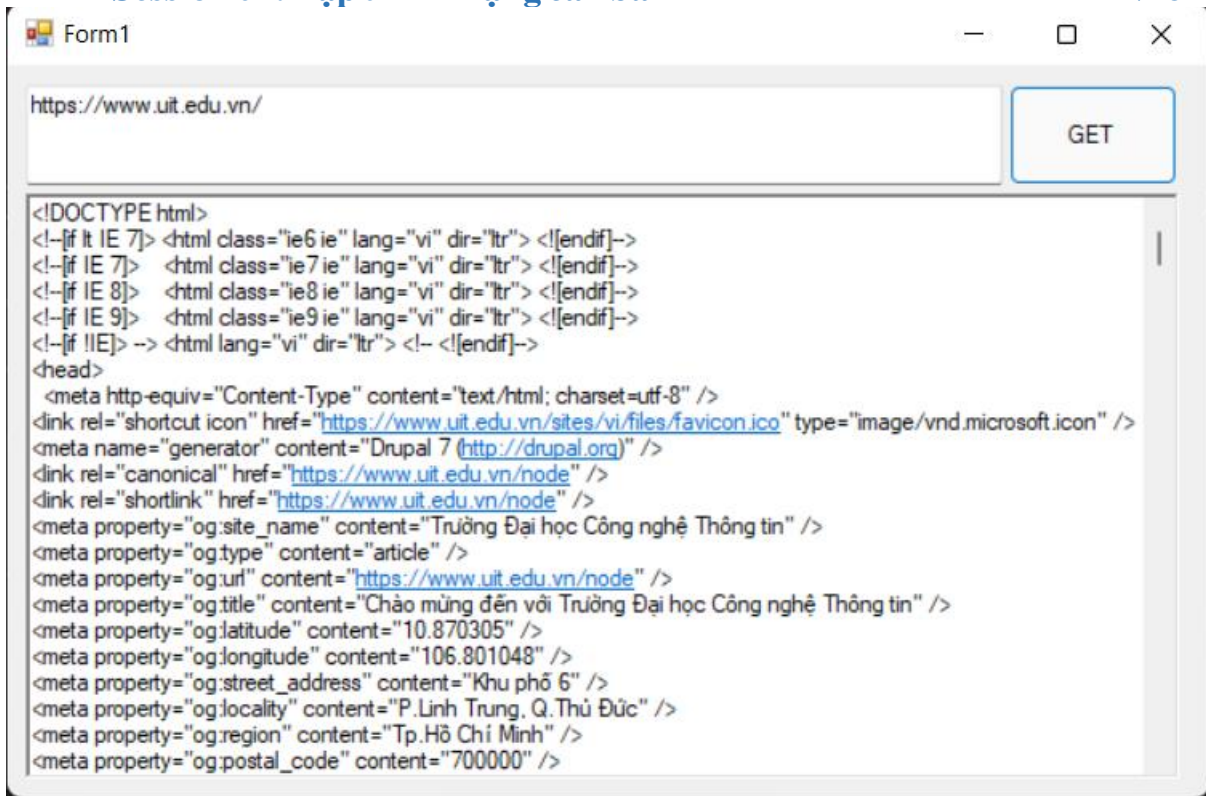
- Thêm code vào hàm để thực hiện chức năng

```
1 reference
private void button1_Click(object sender, EventArgs e)
{
    try
    {
        richTextBox1.Text = getHTML(textBox1.Text);
    }
    catch
    {
        MessageBox.Show("URL không tồn tại!", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
        textBox1.Text = "http://";
    }
}
```

```
1 reference
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;
}
```

Hình 5: Đoạn mã thực hiện việc lấy và hiển thị nội dung của trang web ra màn hình.

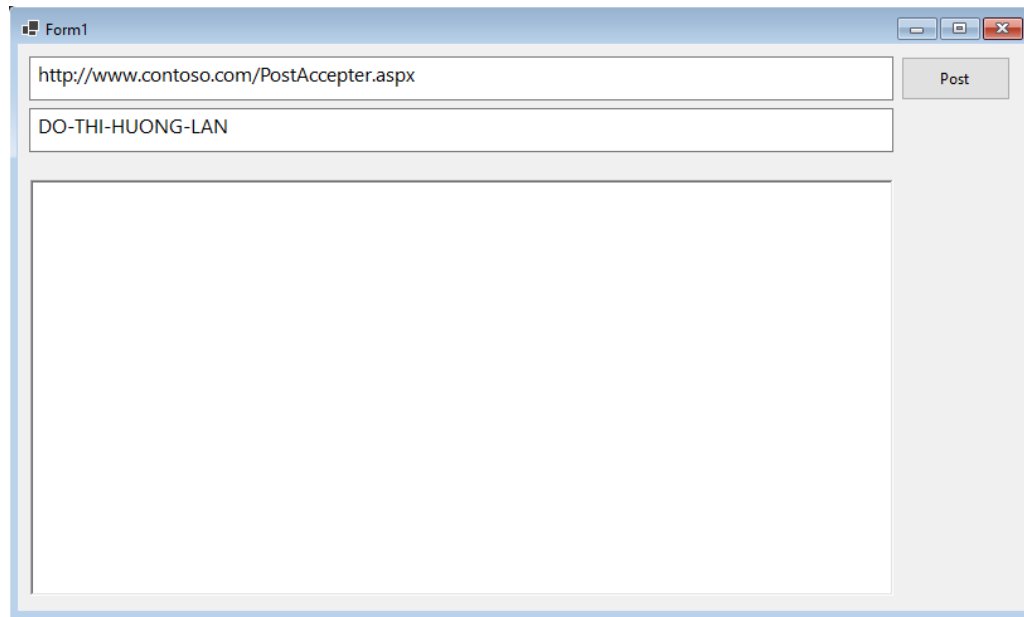
- Tạo ra hàm getHTML để có thể sử dụng URL truy cập và lấy nội dung của trang web
 - Khởi tạo WebRequest với nội dung là đường dẫn của trang web lấy từ nội dung nhập vào của người dùng
 - Sử dụng Stream để đón nhận dữ liệu từ WebResponse trả về
 - Tải nội dung trang web và tiến hành lưu vào biến responseFromServer.
 - Hiển thị nội dung của trang web ra richTextBox
- Kết quả thực hiện chương trình:



Hình 7: Kết quả sau khi chạy chương trình

2. Viết chương trình gửi dữ liệu bất kỳ lên địa chỉ

- Tài nguyên:
- Mô tả/mục tiêu:
 - Mô tả:
 - Mục tiêu: viết được chương trình gửi dữ liệu bất kỳ lên địa chỉ: <http://www.contoso.com/PostAcceptor.aspx> bằng phương thức POST
- Các bước thực hiện/ Phương pháp thực hiện (Ảnh chụp màn hình, có giải thích)
 - Bước 1: Tạo project trong Visual Code bằng cách chọn File -> New -> Project -> Visual C# -> Windows Form Application. Đặt tên cho project.
 - Bước 2: Tạo form giao diện theo yêu cầu bằng cách sử dụng toolbox trong Visual Studio. Sử dụng textbox để tạo ô chứa địa chỉ URL được gửi dữ liệu có tên 'txtUrl', và textbox để chứa nội dung có tên 'txtContent'. Sử dụng button để tạo nút "POST" có tên 'btnPost' sau khi chạy chương trình ta Click và nút POST chương trình sẽ gửi dữ liệu lên địa chỉ URL ở 'txtUrl'. Sử dụng richtextbox để chứa nội dung cần Post lên trang web có URL trong ô 'txtUrl'.



Hình 9: Tạo form giao diện theo yêu cầu

- Bước 2: Lập trình sự kiện và chức năng theo yêu cầu
 - Để thực hiện gửi và hiển thị các nội dung khi dữ liệu được gửi thì người dùng phải tiến hành click vào nút “POST” vì vậy ta tiến hành bắt sự kiện khi người dùng click vào nút này.
 - Nhấp đúp chuột vào button “POST”. Chương trình sẽ khởi tạo 1 hàm tương ứng với sự kiện click vào button này.

```
private void btnPost_Click(object sender, EventArgs e)
{
    ...
}
```

Hình 10: Hàm khởi tạo sao khi click vào nút button POST



- Thêm code để thực hiện yêu cầu của bài toán vào

```
private void btnPost_Click(object sender, EventArgs e)
{
    if (txtContent.Text != string.Empty)
    {
        rtbResponse.Text = getResponse(txtURL.Text);
    }
    else
    {
        MessageBox.Show("Nhập gì đó vào nội dung!", "Thông báo",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}
```

- Thực hiện bắt sự kiện cho nút btnPost

Để thực hiện được yêu cầu của bài toán:

- Thực hiện viết hàm getResponse

```
private string getResponse(string strURL)
{
    ~
}
```

- Khởi tạo request với địa chỉ URL sử dụng phương thức POST

```
WebRequest request = WebRequest.Create(strURL);
request.Method = "POST";
```

- Lấy dữ liệu từ form, sau đó convert sang mảng byte

```
string postData = txtContent.Text;
byte[] byteArray = UTF32Encoding.UTF32.GetBytes(postData);

request.ContentType = "application/x-www-form-urlencoded";
request.ContentLength = byteArray.Length;

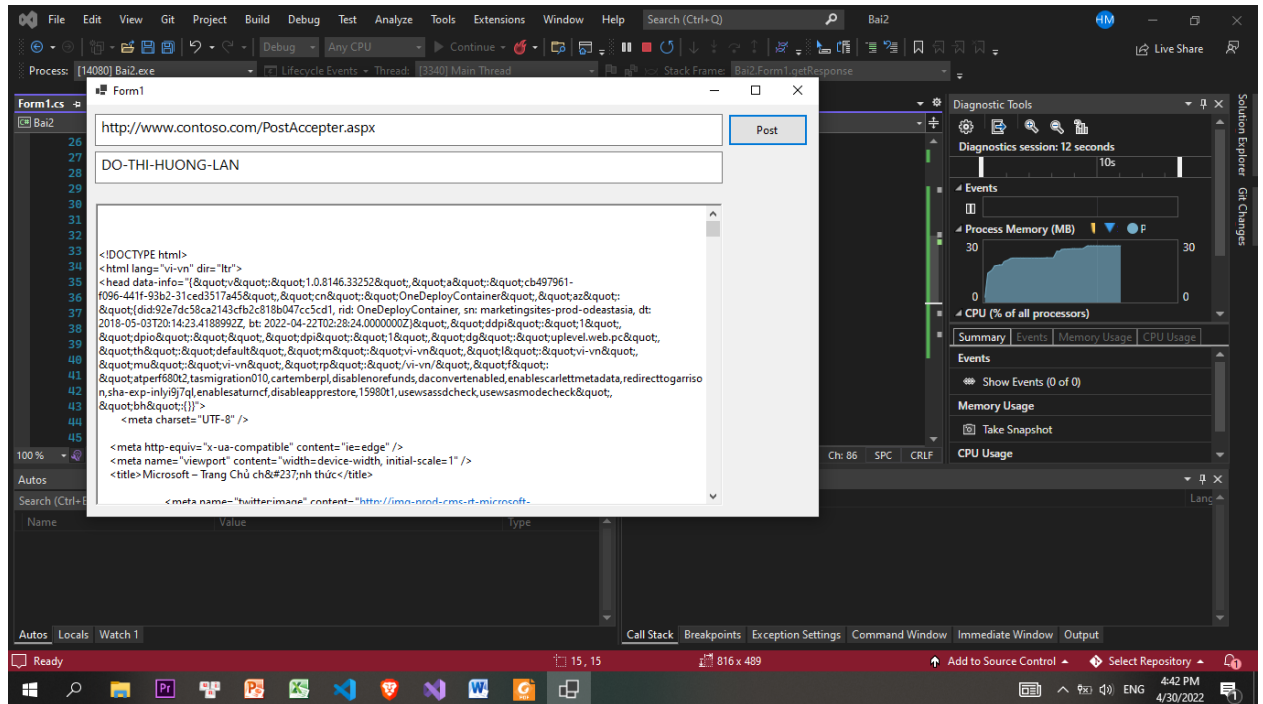
Stream dataStream = request.GetRequestStream();
dataStream.Write(byteArray, 0, byteArray.Length);
dataStream.Close();
```

- Lấy dữ liệu từ form, sau đó convert sang mảng byte Get response từ WebRequest và gán vào response stream, đọc dữ liệu từ response stream.


```
dataStream = response.GetResponseStream();
StreamReader reader = new StreamReader(dataStream);
string responseFromServer = reader.ReadToEnd();
response.Close();
return responseFromServer;
```

Hình 11: Đoạn mã thực hiện yêu cầu

Sau khi nhấn POST nội dung trả về của trang web được hiển thị trong richtextbox

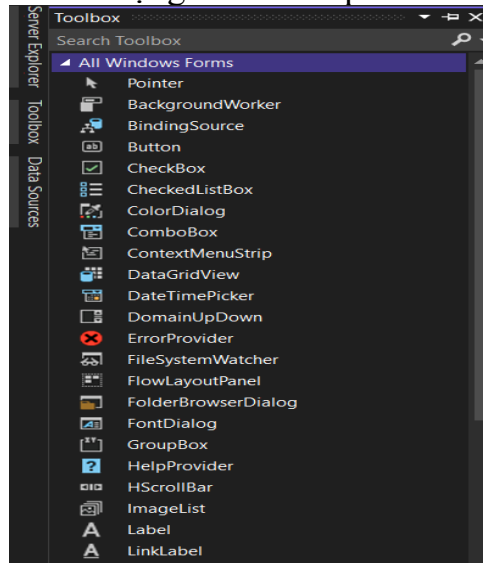


Hình 12: Kết quả hoạt động của chương trình

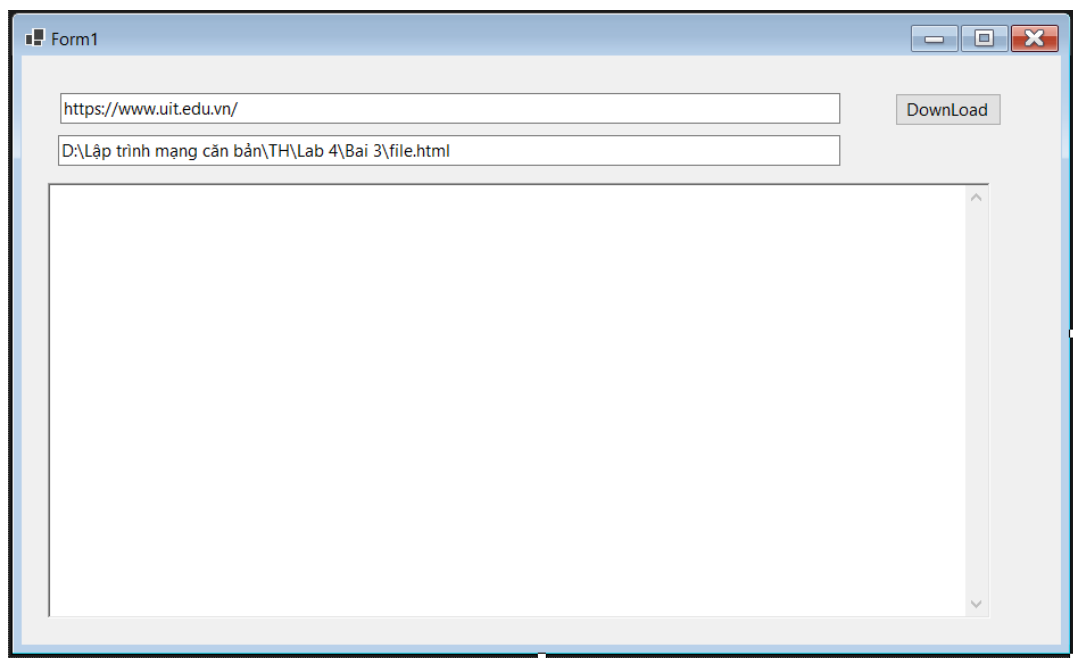
3. Download nội dung trang web

- Tài nguyên:
- Mô tả/mục tiêu:
 - Mô tả: viết chương trình download nội dung trang wb bất kỳ từ một địa chỉ url bất kỳ và lưu thành file html, sau đó hiển thị nội dung lên form
 - Mục tiêu: viết được chương trình, thực hiện được các chức năng mô tả
- Các bước thực hiện/ Phương pháp thực hiện (Ảnh chụp màn hình, có giải thích)

- Bước 1: Tạo form giao diện theo yêu cầu bằng cách sử dụng toolbox trong Visual Studio. Sử dụng label và input text để tạo ra giao diện.



Hình 13: Thanh tool box chứa các thành phần để tạo giao diện form



Hình 14: Giao diện windows form khởi tạo theo yêu cầu

- Bước 2: Lập trình sự kiện và chức năng theo yêu cầu bài toán thực hiện các phép tính

```

1 reference
private void btnDown_Click(object sender, EventArgs e)
{
    string url = txtUrl.Text;
    string fileurl = txtFileUrl.Text;
    WebClient myClient = new WebClient();
    Stream response = myClient.OpenRead(url);
    myClient.DownloadFile(url, fileurl);

    StreamReader reader = new StreamReader(response);
    string responseFromServer = reader.ReadToEnd();
    rtxtHtml.Text = responseFromServer;
}

```

- Khởi tạo 1 WebClient
 - Sử dụng OpenRead để đọc nội dung trang web
 - Dùng downloadFile để tải nội dung trang web về đường dẫn file html đã chỉ định
 - StreamReader để đọc phản hồi từ web
 - Cuối cùng hiển thị nội dung ra richTextBox
- Bước 3: Kết quả thực hiện:

The screenshot shows a Windows application window titled "Form1". It contains a web browser interface with a URL field, a "Download" button, and a file path field. Below these fields is a text area displaying the HTML content of the downloaded page.

URL:

File Path:

Download button:

HTML Content:

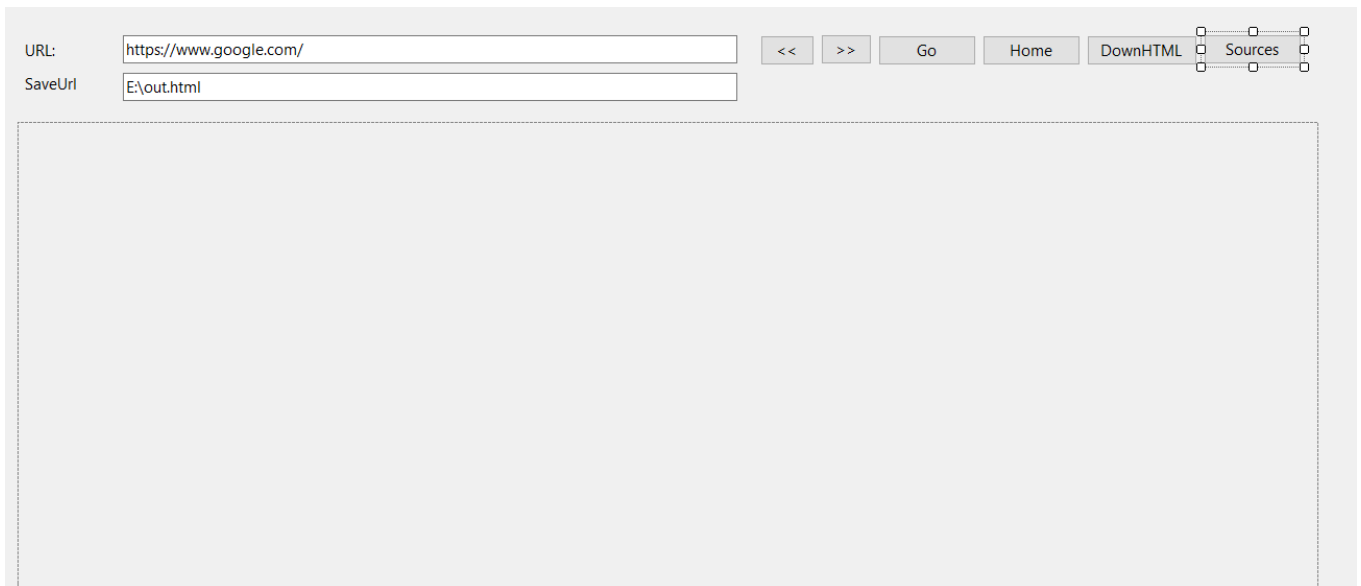
```

<!DOCTYPE html>
<!--[if lt IE 7]> <html class="ie6 ie" lang="vi" dir="ltr"> <![endif-->
<!--[if IE 7]> <html class="ie7 ie" lang="vi" dir="ltr"> <![endif-->
<!--[if IE 8]> <html class="ie8 ie" lang="vi" dir="ltr"> <![endif-->
<!--[if IE 9]> <html class="ie9 ie" lang="vi" dir="ltr"> <![endif-->
<!--[if !IE]> --> <html lang="vi" dir="ltr"> <!-- <![endif-->
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <link rel="shortcut icon" href="https://www.uit.edu.vn/sites/vi/files/favicon.ico" type="image/vnd.microsoft.icon" />
  <meta name="generator" content="Drupal 7 (http://drupal.org)" />
  <link rel="canonical" href="https://www.uit.edu.vn/node" />
  <link rel="shortlink" href="https://www.uit.edu.vn/node" />
  <meta property="og:site_name" content="Trường Đại học Công nghệ Thông tin" />
  <meta property="og:type" content="article" />
  <meta property="og:url" content="https://www.uit.edu.vn/node" />
  <meta property="og:title" content="Chào mừng đến với Trường Đại học Công nghệ Thông tin" />
  <meta property="og:latitude" content="10.870305" />
  <meta property="og:longitude" content="106.801048" />
  <meta property="og:street_address" content="Khu phố 6" />

```

4. Viết chương trình hoạt động như Web Browser

- Tài nguyên:
- Mô tả/mục tiêu:
 - Viết chương trình hoạt động như Web Browser có các tính năng cơ bản
 - Chương trình có các tính năng như: xem nội dung website, download file html, xem source
- Các bước thực hiện/ Phương pháp thực hiện (Ảnh chụp màn hình, có giải thích)
 - Bước 1: Tạo form



- URL: địa chỉ web
- SaveUrl: địa chỉ lưu thành file html
- Go: chuyển web browser đến đường link textbox
- <<,>>: có chức năng go back và go forward
- Home: về trang chủ của web browser
- DownHtml: download file html
- Source: hiển thị html đã download được
- Bước 2: Lập trình chức năng

```
3 references
public partial class Form1 : Form
{
    WebBrowser webBrowser1;
    1 reference
    public Form1()
    {
        InitializeComponent();
        webBrowser1 = new WebBrowser();
        webBrowser1.Width = pnlWeb.Width;
        webBrowser1.Height = pnlWeb.Height;
        pnlWeb.Controls.Add(webBrowser1);
    }
}
```

- Tạo 1 web browser và add vào panel

```
1 reference
private void btnGo_Click(object sender, EventArgs e)
{
    string link = txtUrl.Text;
    webBrowser1.Navigate(link);
}
```

- Chuyển tới đã được nhập trong txtUrl.Text

```
1 reference
private void btnBack_Click(object sender, EventArgs e)
{
    webBrowser1.GoBack();
}

1 reference
private void btnForward_Click(object sender, EventArgs e)
{
    webBrowser1.GoForward();
}

1 reference
private void btnHome_Click(object sender, EventArgs e)
{
    webBrowser1.GoHome();
}
```

- Đoạn code thực hiện các chức năng GoBack, GoForward, GoHome của web browser

```
1 reference
private void btnDown_Click(object sender, EventArgs e)
{
    string url = webBrowser1.Url.ToString();
    var html = new HtmlWeb();
    var document = html.Load(url);
    File.WriteAllText(txtSave.Text, document.DocumentNode.InnerHtml);
    MessageBox.Show("Done");
}
```

- Sử dụng HtmlAgilityPack để lấy các thẻ của html
- File.WriteAllText để lưu các thẻ html vào địa chỉ đã chỉ định

```
1 reference
private void btnSources_Click(object sender, EventArgs e)
{
    string url = webBrowser1.Url.ToString();
    var html = new HtmlWeb();
    var document = html.Load(url);
    sources child = new sources(document.DocumentNode.InnerHtml);
    child.Show();
}
```

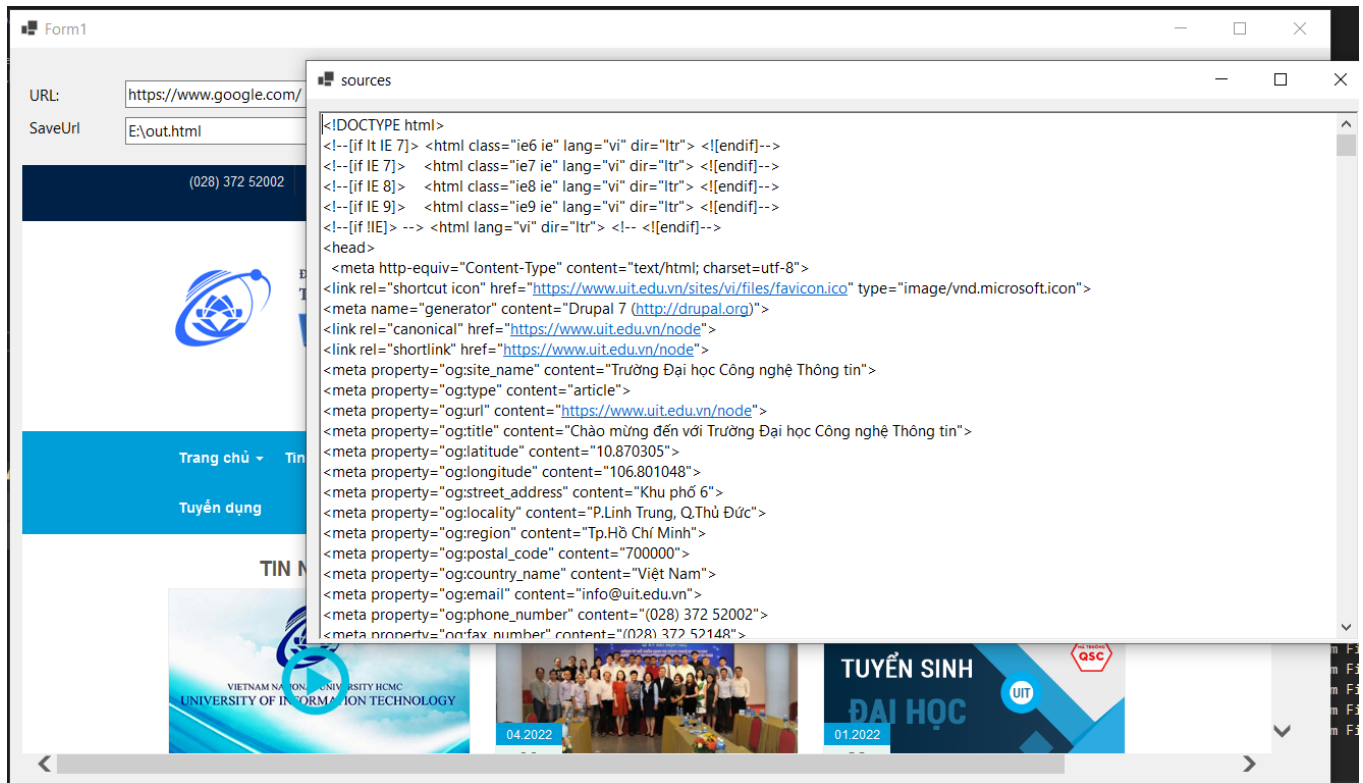
- Truyền file html đã lấy được bằng HtmlAgilityPack vào form 2 có tên source để hiển thị

```
1 reference
public sources(string html): this()
{
    _message = html;
    rtxtHtml.Text = _message;
}
```

- Tạo hàm khởi tạo ở form 2 lấy dữ liệu truyền từ form 1 để hiển thị

- Khi nhận send client sẽ tin cho server dưới dạng mảng byte

- Kết quả thực hiện chương trình



Link Source Code: <https://github.com/khoanamgit/NT106.M21.ATCL-NH/tree/main/Lab04>

YÊU CẦU CHUNG

- Sinh viên tìm hiểu và thực hành theo hướng dẫn.
- Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Sinh viên báo cáo kết quả thực hiện và nộp bài.

Báo cáo:

- File **.PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Nội dung trình bày bằng **Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Neo Sans Intel/UTM Viet Sach)– cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.**
- Đặt tên theo định dạng: [Mã lớp]-SessionX_GroupY. (trong đó X là Thứ tự buổi Thực hành, Y là số thứ tự Nhóm Thực hành đã đăng ký với GVHD-TH).

Ví dụ: [NT101.K11.ANTT]-Session1_Group3.

- Nếu báo cáo có nhiều file, nén tất cả file vào file **.ZIP** với cùng tên file báo cáo.
- **Không đặt tên đúng định dạng – yêu cầu, sẽ KHÔNG chấm điểm bài Lab.**
- Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

Đánh giá: Sinh viên hiểu và tự thực hiện được bài thực hành. Khuyến khích:

- Chuẩn bị tốt.
- Có nội dung mở rộng, ứng dụng trong kịch bản phức tạp hơn, có đóng góp xây dựng bài thực hành.

Bài sao chép, trễ, ... sẽ được xử lý tùy mức độ vi phạm.

HẾT