

Name: Phan Phạm Trọng Phúc

ID: 21520402

Class: NT216.N21.ANTN

Lập trình mạng căn bản LAB 4'S REPORT

Giao diện DashBoard để di chuyển giữa các bài:



Code:

```

namespace Lab04_21520482_PhanPhamTrongPhuc
{
    3 references
    public partial class DashBoard : Form
    {
        1 reference
        public DashBoard()
        {
            InitializeComponent();
        }

        1 reference
        private void btnBai1_Click(object sender, EventArgs e)
        {
            var bai1 = new Bai_1();
            bai1.Show();
        }

        1 reference
        private void btnBai2_Click(object sender, EventArgs e)
        {
            var bai2 = new Bai_2();
            bai2.Show();
        }

        1 reference
        private void btnBai3_Click(object sender, EventArgs e)
        {
            var bai3 = new Bai_3();
            bai3.Show();
        }

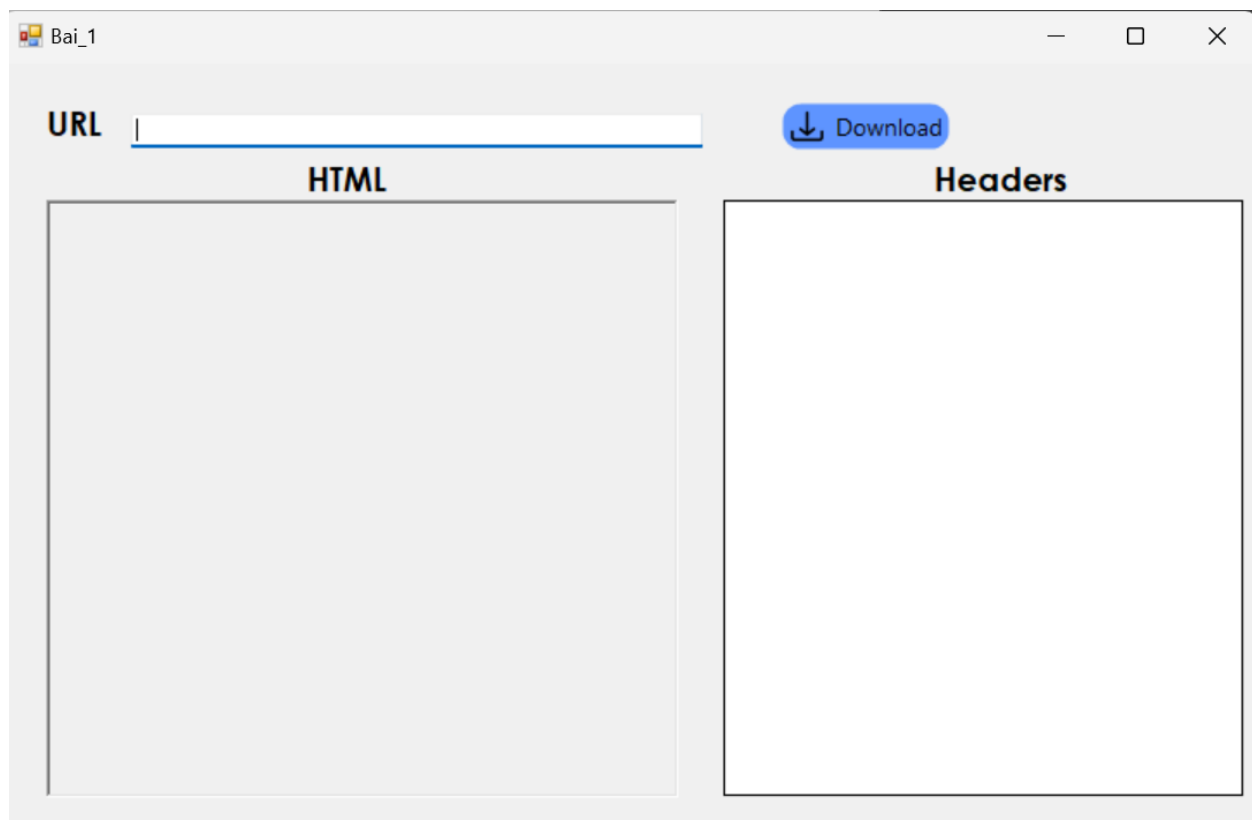
        1 reference
        private void btnBai4_Click(object sender, EventArgs e)
        {
            var bai4 = new Bai_4();
            bai4.Show();
        }
    }
}

```

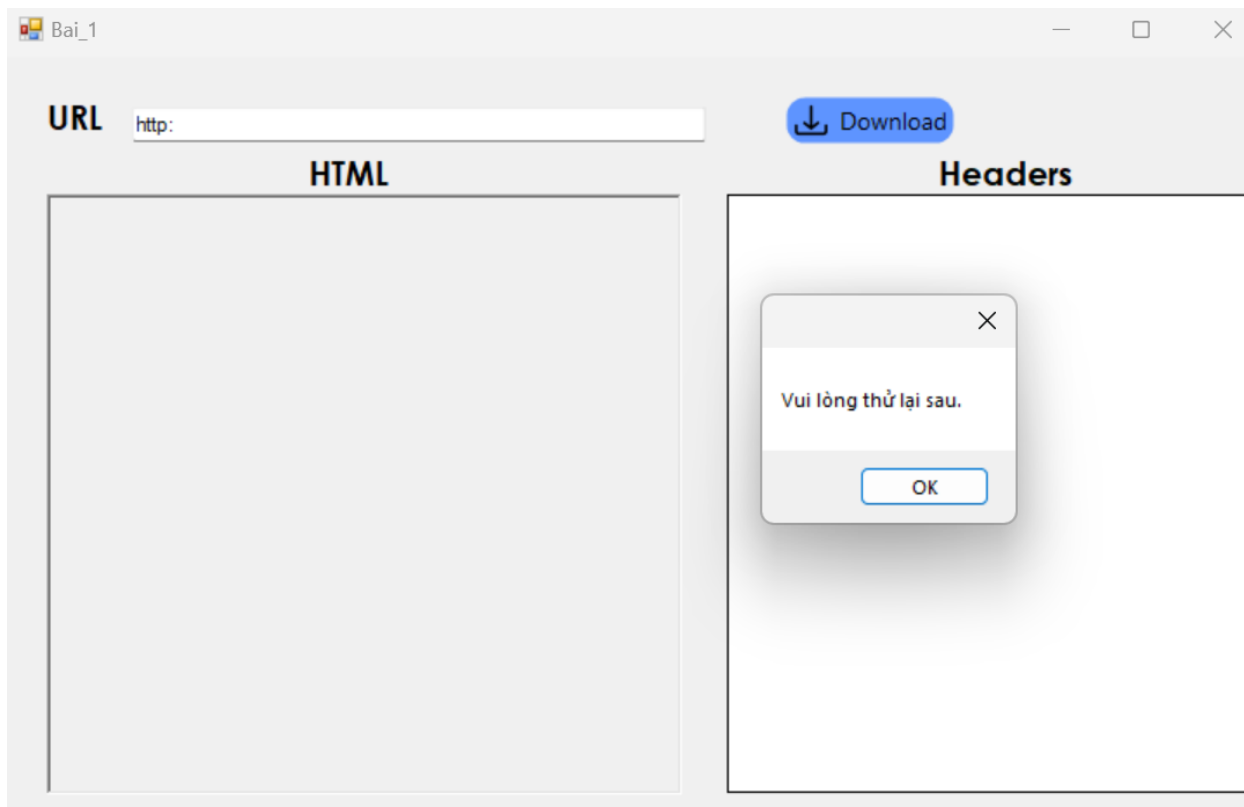
Tạo ra các form mới và di chuyển tương ứng nếu người dùng chọn bài.

Bài 1:

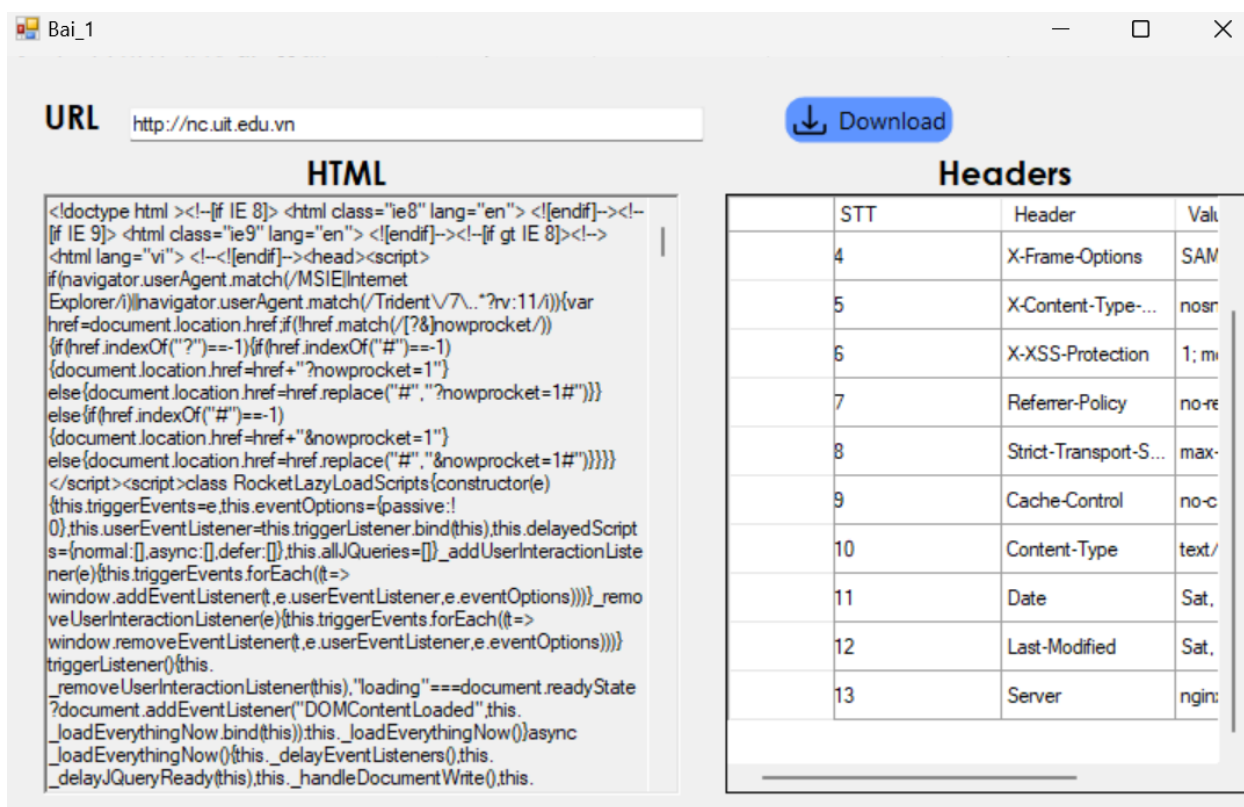
Giao diện:



Kiểm tra textbox URL do người dùng nhập vào nếu là chuỗi rỗng hoặc không thể lấy được source code thì sẽ hiển thị “Vui lòng thử lại sau”:



Khi người dùng nhập đúng URL và có thể lấy được sourcecode:



Chương trình sẽ lấy sourcecode của trang web, hiển thị trong richtextbox và xử lý các responseheader sau đó hiển thị trong datagridview.

Code:

```
1 reference
private void btnDownload_Click(object sender, EventArgs e)
{
    string url = tbUrl.Text;
    if (url == "")
    {
        MessageBox.Show("Vui lòng nhập URL!");
    }
    else
    {
        using (WebClient myClient = new WebClient())
        {
            // Try receive data
            try
            {
                byte[] response = myClient.DownloadData(url);
                string sourceCode = Encoding.UTF8.GetString(response);

                // Display the source code
                rtbHTML.Text = sourceCode;

                // Process the response headers as needed
                WebHeaderCollection whc = myClient.ResponseHeaders;
                DataTable headerTable = ConvertToDataTable(whc);
                dataGridView.DataSource = headerTable;
                dataGridView.Show();
            }
            catch (Exception err)
            {
                Console.WriteLine(err.ToString());
                MessageBox.Show("Vui lòng thử lại sau.");
            }
        }
    }
}
```

- Dùng phương thức DownloadData của WebClient để lấy source code của trang web về.
- Lấy các ResponseHeader dưới dạng WebHeaderCollection và sau đó hiển thị ở datagridview.
- Hàm xử lý và hiển thị các ReponseHeader trong datagridview.

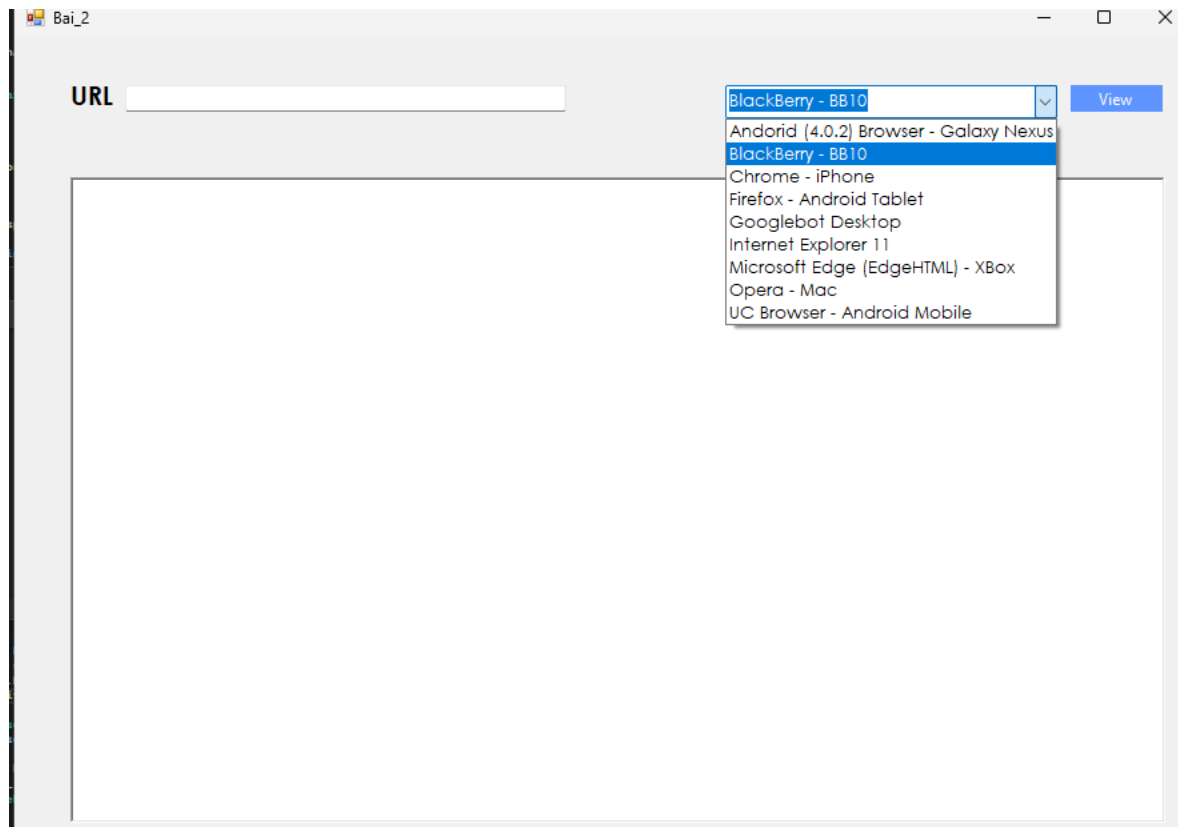
```
1 reference
private DataTable ConvertToDataTable(WebHeaderCollection headers)
{
    // Create datagridview table
    DataTable table = new DataTable();
    table.Columns.Add("STT", typeof(int));
    table.Columns.Add("Header", typeof(string));
    table.Columns.Add("Value", typeof(string));

    for (int i = 0; i < headers.Count; i++)
    {
        string headerName = headers.GetKey(i);
        string headerValue = headers.Get(i);
        table.Rows.Add(i + 1, headerName, headerValue);
    }

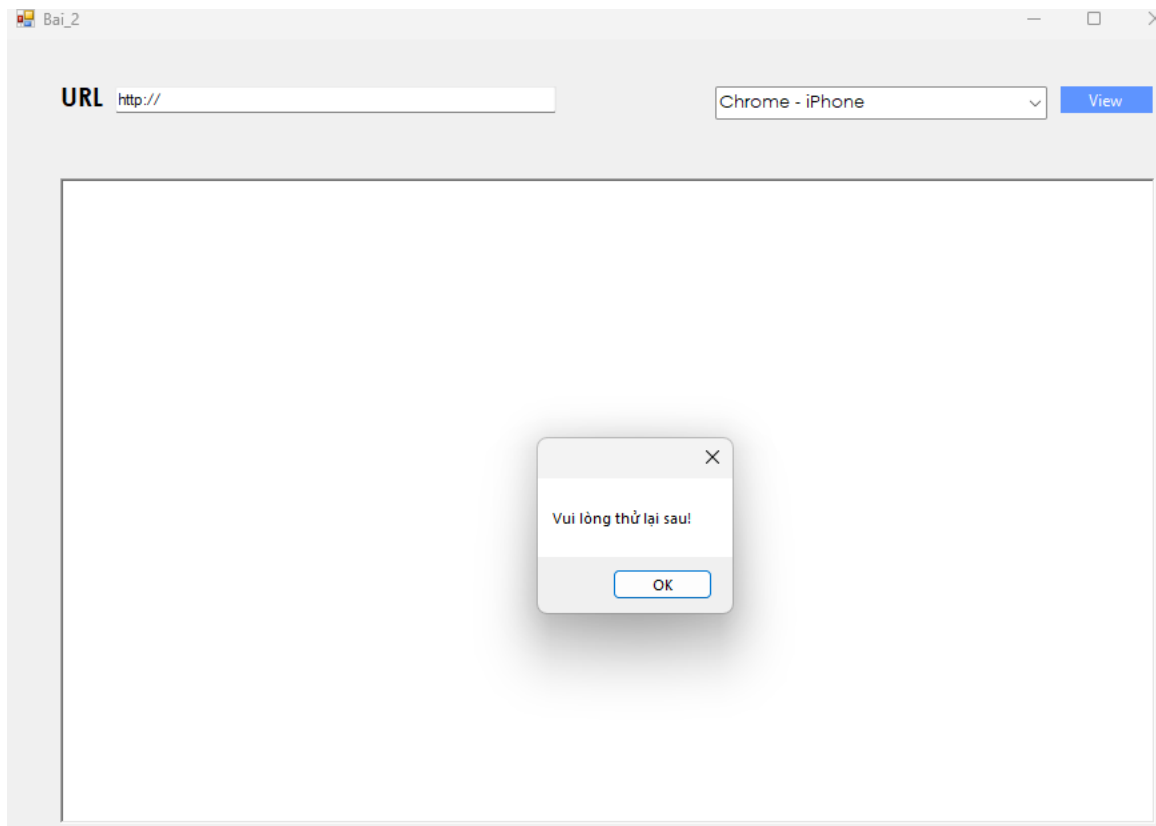
    return table;
}
```

Bài 2:

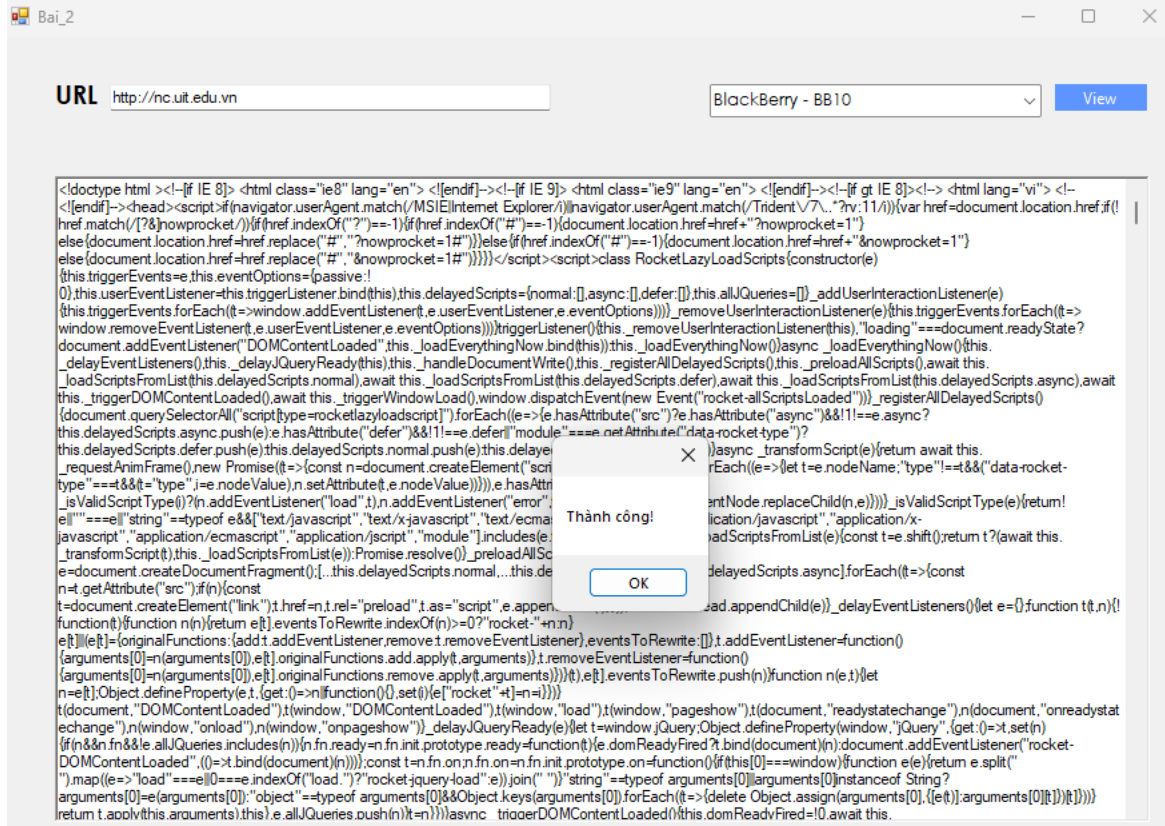
Giao diện:



Chương trình cho phép chọn User-Agent



Kiểm tra URL



Hiện thị sourcecode lấy từ trang web tương ứng

```
'Lab04_21520402_PhanPhamTrongPhuc.exe' (CLR v4.0.30319: Lab04_21520402_PhanPhamTrongPhuc.exe): Loaded 'C:\WINDOWS\Microsoft.Net\v4.0.30319\mscorlib.dll'
Selected Key: BlackBerry - BB10
Selected Value: Mozilla/5.0 (BB10; Touch) AppleWebKit/537.1+ (KHTML, like Gecko) Version/10.0.0.1337 Mobile Safari/537.1+
User-Agent: Mozilla/5.0 (BB10; Touch) AppleWebKit/537.1+ (KHTML, like Gecko) Version/10.0.0.1337 Mobile Safari/537.1+
Selected Key: Internet Explorer 11
Selected Value: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko
The program '[23644] Lab04_21520402_PhanPhamTrongPhuc.exe' has exited with code 0 (0x0).
```

In ra màn hình để kiểm tra

Code:

```
private void btnDisplay_Click(object sender, EventArgs e)
{
    Dictionary<string, string> valueDictionary = new Dictionary<string, string>()
    {
        {"Android (4.0.2) Browser - Galaxy Nexus", "Mozilla/5.0 (Linux; U; Android 4.0.2; en-us; Galaxy Nexus Build/ICL53F) AppleWebKit/534.30 (KHTML, like Gecko) Version/4.0 Mobile Safari/534.30"},
        {"BlackBerry - BB10", "Mozilla/5.0 (BB10; Touch) AppleWebKit/537.1+ (KHTML, like Gecko) Version/10.0.0.1337 Mobile Safari/537.1+"},
        {"Chrome - iPhone", "Mozilla/5.0 (iPhone; CPU iPhone OS 11_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/113.0.0.0 Mobile/15E148 Safari/604.1"},
        {"Firefox - Android Tablet", "Mozilla/5.0 (Android 4.4; Tablet; rv:70.0) Gecko/70.0 Firefox/70.0"},
        {"Googlebot Desktop", "Mozilla/5.0 AppleWebKit/537.36 (KHTML, like Gecko; compatible; Googlebot/2.1; +http://www.google.com/bot.html) Chrome/113.0.0.0 Safari/537.36"},
        {"Internet Explorer 11", "Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko"},
        {"Microsoft Edge (EdgeHTML) - Xbox", "Mozilla/5.0 (Windows NT 10.0; Win64; x64; Xbox; Microsoft Edge/18.10240.16234.0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36 Edge/18.19841"},
        {"Opera - Mac", "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3984.97 Safari/537.36 OPR/65.0.3467.48"},
        {"UC Browser - Android Mobile", "Mozilla/5.0 (Linux; U; Android 8.1.0; en-US; Nexus 6P Build/OPM7.181205.001) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/57.0.2987.108 UCBrowser/12.11.1.1197 Mobile Safari/537.36"},
    };

    string selectedKey = cbDevices.SelectedItem.ToString();

    // Retrieve the corresponding value from the dictionary for testing
    if (valueDictionary.TryGetValue(selectedKey, out string selectedValue))
    {
        // Display the selected key and value
        Console.WriteLine($"Selected Key: {selectedKey}");
        Console.WriteLine($"Selected Value: {selectedValue}");
    }
}
```

Dùng Dictionary để lưu cặp key – value tương ứng của từng chuỗi User-Agent. Ở đây em chỉ lấy mỗi User-Agent một đại diện.

```
try
{
    HttpClient httpClient = new HttpClient();
    string url = tbUrl.Text;
    httpClient.DefaultRequestHeaders.Add("User-Agent", selectedValue);
    Console.WriteLine("User-Agent: " + selectedValue);

    HttpResponseMessage response = httpClient.GetAsync(url).Result;
    if (response.IsSuccessStatusCode && url != "" && selectedKey != "")
    {
        string htmlContent = response.Content.ReadAsStringAsync().Result;
        rtbHTML.Text = htmlContent;
        MessageBox.Show("Thành công!");
    }
    else
    {
        rtbHTML.Text = "Mã lỗi: " + response.StatusCode;
    }
}
catch (Exception err)
{
    Console.WriteLine(err.ToString());
    MessageBox.Show("Vui lòng thử lại sau!");
}
```

Sau đó dùng phương thức Get để lấy sourcecode trang web về và hiển thị. Trong phương thức Get có thêm một Header là User-Agent và chuỗi tương ứng với người dùng chọn.

Bài 3:

Giao diện:

Bai_3

Select city....

Update

Sunrise: Sunset:

Location

No infor

No infor

Description: Pressure: Humidity: Wind speed:

Country: Longitude: Latitude: Date:

Cho phép người dùng chọn và hiển thị thông tin thời tiết của 5 thành phố.

Bai_3

Select city....

Update

Sunrise: Sunset:

No infor

Now

No infor

Description: Pressure: Humidity: Wind speed:

Country: Longitude: Latitude: Date:

Vui lòng chọn thành phố.

OK

Kiểm tra trường hợp người dùng không chọn thành phố.

Location:

Sunrise: Sunset:

Clouds

Now

28,97 °C

Description:

Pressure:

Humidity:

Wind speed:

Location

Current time in Ho Chi Minh City 2:58 PM

Country:

Longitude:

Latitude:

Date:

Hiển thị thông tin thời tiết của Sài Gòn.

Location:

Sunrise: Sunset:

Clear

Now

14,2 °C

Description:

Pressure:

Humidity:

Wind speed:

Location

Current time in New York 3:58 AM

Country:

Longitude:

Latitude:

Date:

Hiển thị thông tin thời tiết của New York.

Code:

Sử dụng API của openweatherAPI để lấy json về xử lý và hiển thị thông tin.

← ↻ ⚠ Not secure | api.openweathermap.org/data/2.5/weather?q=Hanoi&appid=a2cd151ba1553e11f3e7c0e24cb2d80

```

{"coord":{"lon":105.8412,"lat":21.0245},"weather":[{"id":804,"main":"Clouds","description":"overcast clouds","icon":"04d"}],"base":"stations","main":{"temp":306.15,"feels_like":309.73,"temp_min":306.15,"temp_max":306.15,"pressure":1005,"humidity":51,"sea_level":1005,"grnd_level":1003},"visibility":10000,"wind":{"speed":3.51,"deg":169,"gust":4.94},"clouds":{"all":91},"dt":1685260686,"sys":{"type":1,"id":9308,"country":"VN","sunrise":1685225733,"sunset":1685273547},"timezone":25200,"id":1581130,"name":"Hanoi","cod":200}

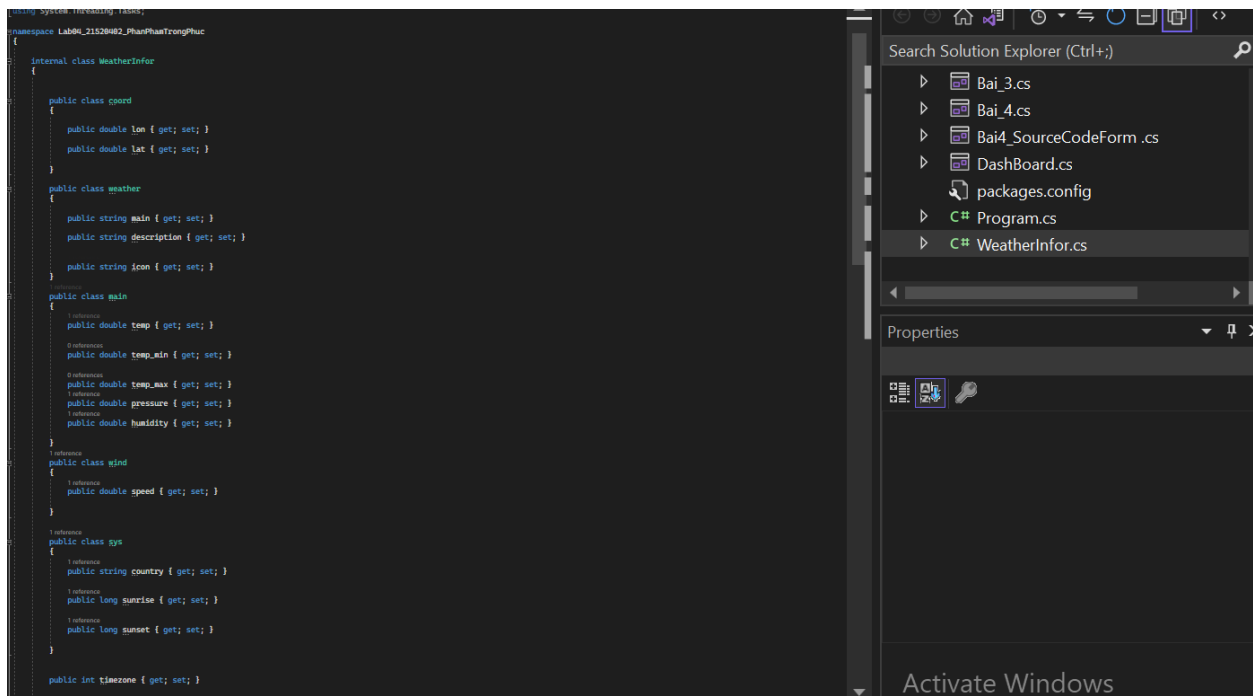
```

```

1 reference
private void btnUpdate_Click(object sender, EventArgs e)
{
    // Check combobox value
    if (cboCity.SelectedItem == null)
    {
        MessageBox.Show("Vui lòng chọn thành phố.");
    }
    else
    {
        string city = cboCity.SelectedItem.ToString();
    }
}

```

Sử dụng combobox để lấy thông tin thành phố người dùng chọn.



Tạo class `WeatherInfo` để xử lý các trường dữ liệu lấy từ file json.

```

using (WebClient web = new WebClient())
{
    // Create URL for HTTP Request
    string url = "http://api.openweathermap.org/data/2.5/weather?q=" + city.ToString() + "&appid=a2cd151ba1553e11f3e7c9e24cb2d80";
    var json = web.DownloadString(url);

    try
    {
        WeatherInfor.root info = JsonConvert.DeserializeObject<WeatherInfor.root>(json);

        // coord's values
        tbLongi.Text = info.coord.lon.ToString();
        tbLatitude.Text = info.coord.lat.ToString();

        // weather's values
        tbDescription.Text = info.weather[0].description.ToString();
        tbMain.Text = info.weather[0].main.ToString();
        pbIcon.ImageLocation = "http://openweathermap.org/img/wn/" + info.weather[0].icon + ".png";

        // main's values
        double tempInC = Math.Round((info.main.temp - 273.15), 2);
        lbTemp.Text = tempInC.ToString() + " °C";
        tbPressure.Text = info.main.pressure.ToString() + " hPa";
        tbHumidity.Text = info.main.humidity.ToString() + " %";

        // wind's values
        tbSpeed.Text = info.wind.speed.ToString() + " m/s";

        // sys's values
        tbCountryCity.Text = info.sys.country;

        // Get time of the city from API
        int timeZoneOffset = info.timezone;
        TimeSpan offset = TimeSpan.FromSeconds(timeZoneOffset);

        // Created TimeZoneInfo from time
        TimeZoneInfo timeZone = TimeZoneInfo.CreateCustomTimeZone(
            "Custom Time Zone", offset, "Custom Time Zone", "Custom Time Zone");

        // Converted time from UTC to local time
        DateTimeOffset sunriseDateTimeOffset = DateTimeOffset.FromUnixTimeSeconds(info.sys.sunrise);
        DateTimeOffset sunsetDateTimeOffset = DateTimeOffset.FromUnixTimeSeconds(info.sys.sunset);

        DateTimeOffset localSunriseDateTimeOffset = TimeZoneInfo.ConvertTime(sunriseDateTimeOffset, timeZone);
        DateTimeOffset localSunsetDateTimeOffset = TimeZoneInfo.ConvertTime(sunsetDateTimeOffset, timeZone);
        // Display the time
        tbSunrise.Text = localSunriseDateTimeOffset.ToString("h:mm tt", CultureInfo.InvariantCulture);
        tbSunset.Text = localSunsetDateTimeOffset.ToString("h:mm tt", CultureInfo.InvariantCulture);

        // timezone's values
        TimeSpan timeSpan = TimeSpan.FromSeconds(info.timezone);
        TimeZoneInfo timeZoneInfo = TimeZoneInfo.CreateCustomTimeZone(info.name, timeSpan, info.name, info.name);

        // Get local time of the city
        DateTimeOffset utcTime = DateTimeOffset.UtcNow;
        DateTimeOffset localTime = TimeZoneInfo.ConvertTime(utcTime, timeZoneInfo);

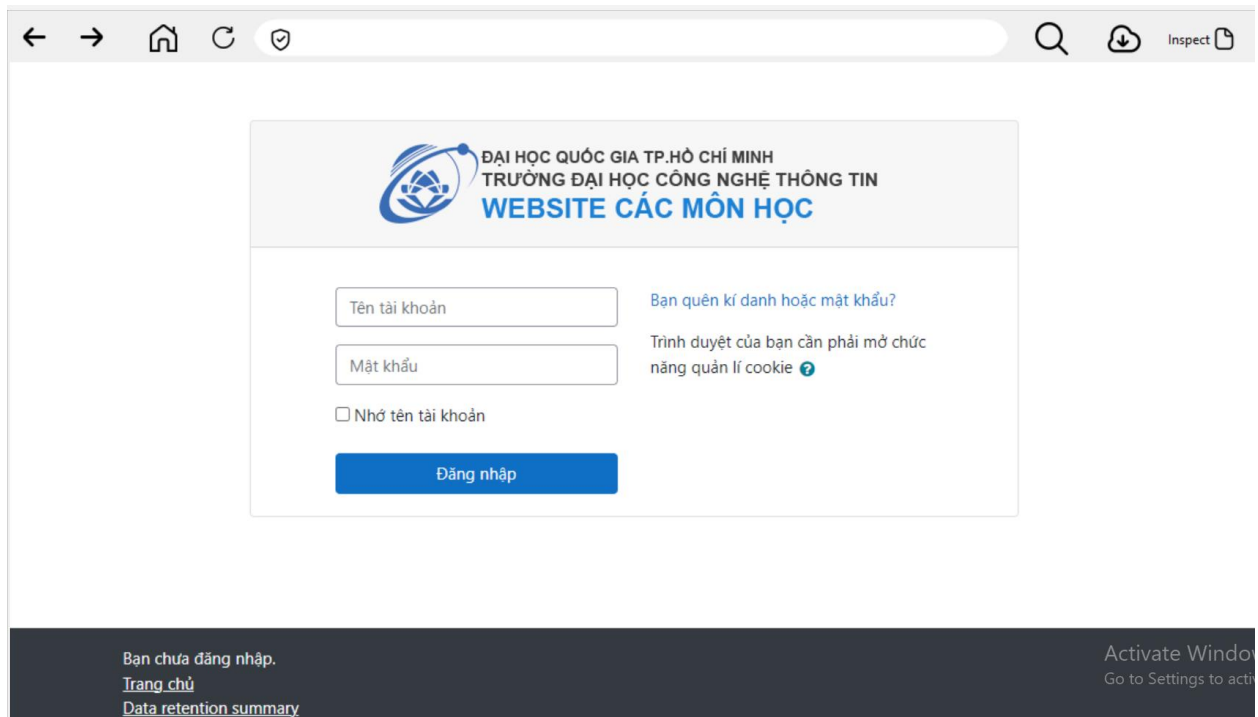
        tbDate.Text = localTime.ToString("dd/MM/yyyy ");
        lbTime.Text = "Current time in " + info.name + " " + localTime.ToString("h:mm tt", CultureInfo.InvariantCulture);
    }
}

```

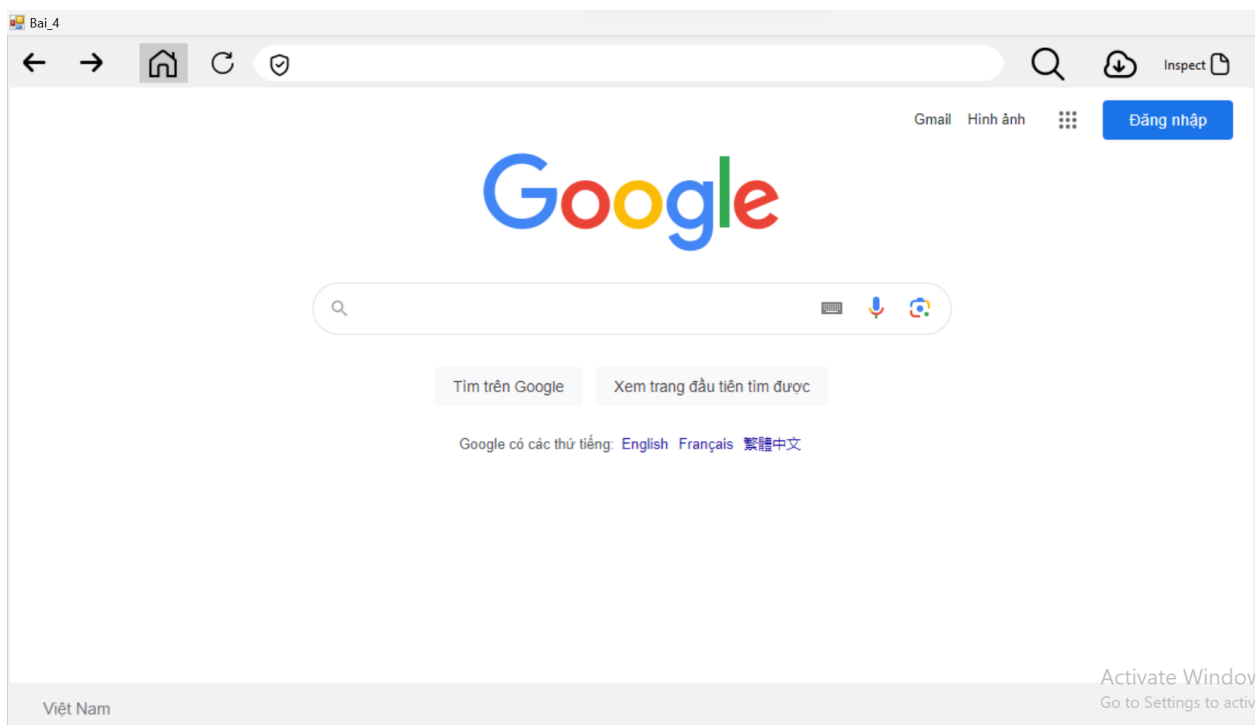
Xử lý các dữ liệu liên quan để hiển thị ra màn hình.

Bài 4:

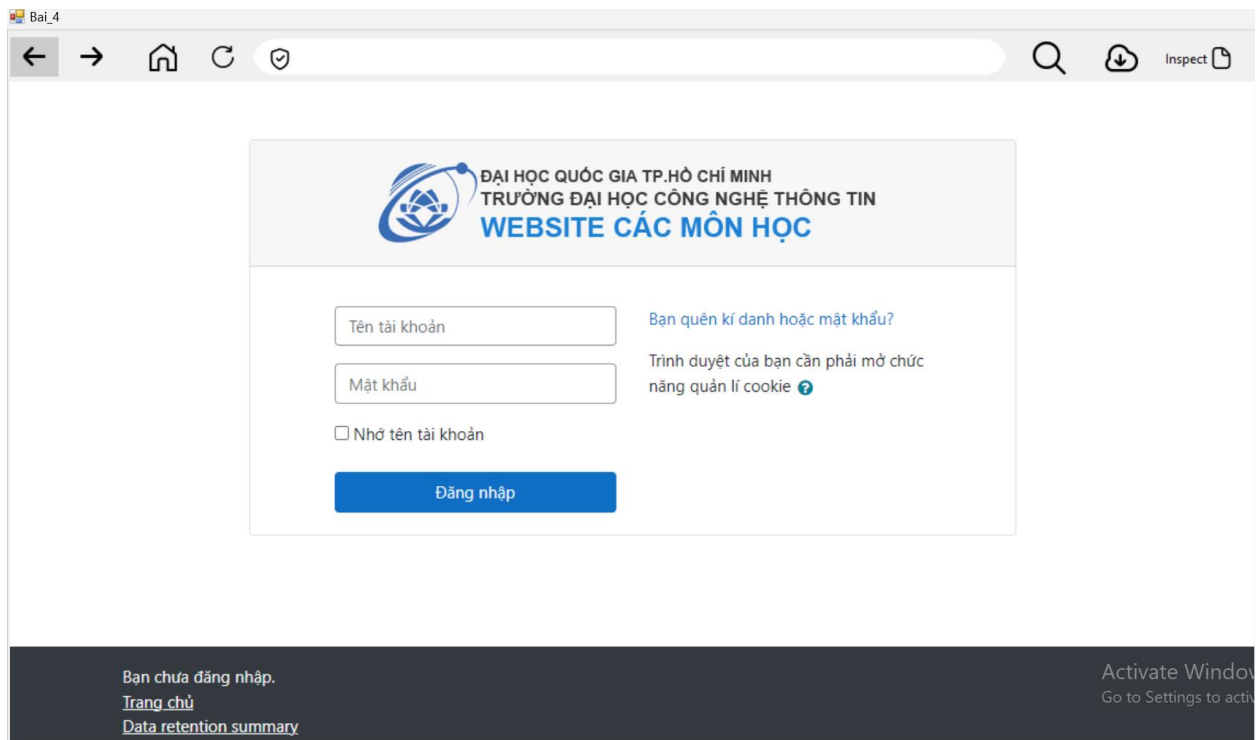
Giao diện:



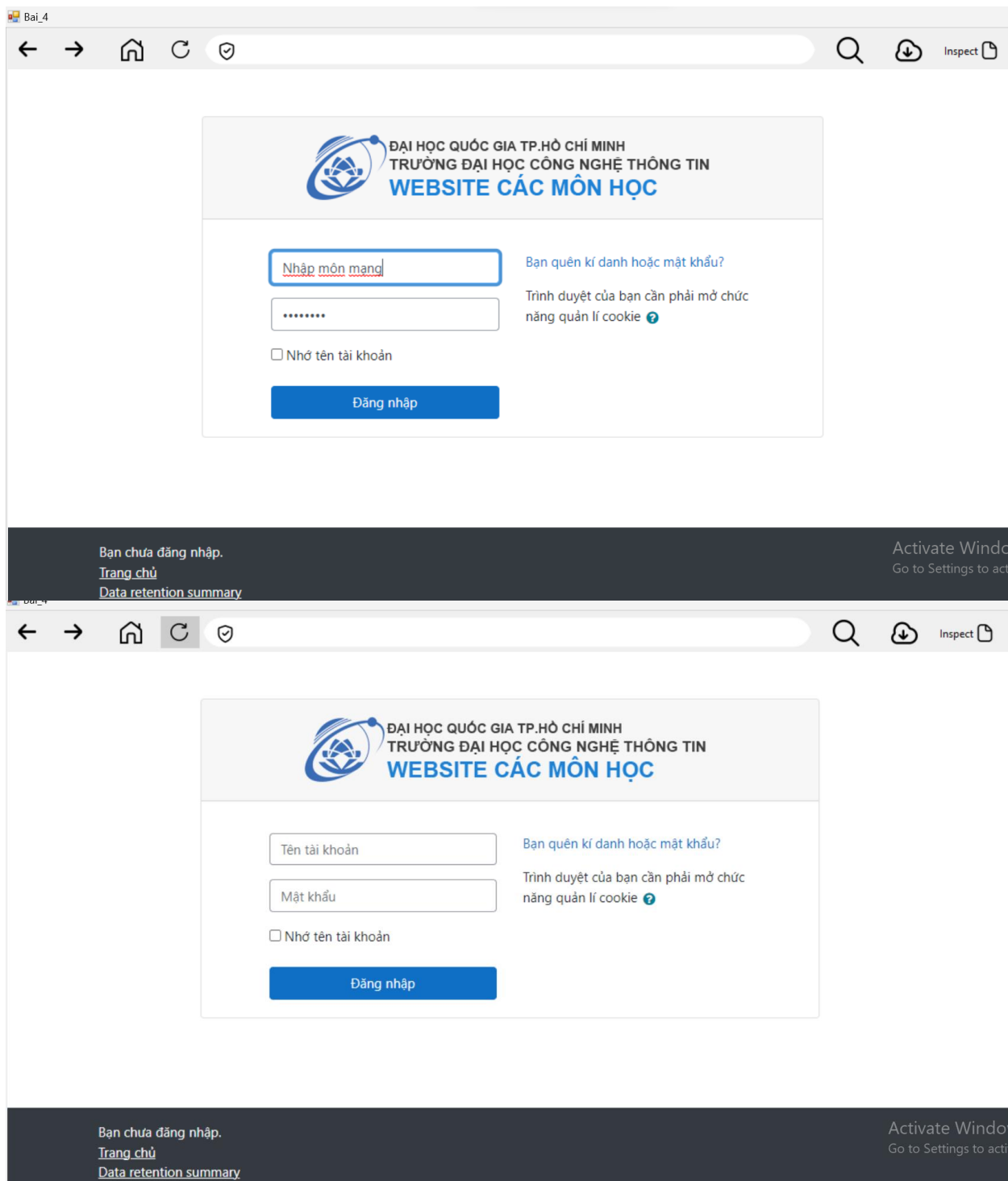
Chương trình mặc định sẽ hiển thị trang web Course UIT sử dụng webview2.



Khi nhấn phím Home sẽ chuyển về trang google.com



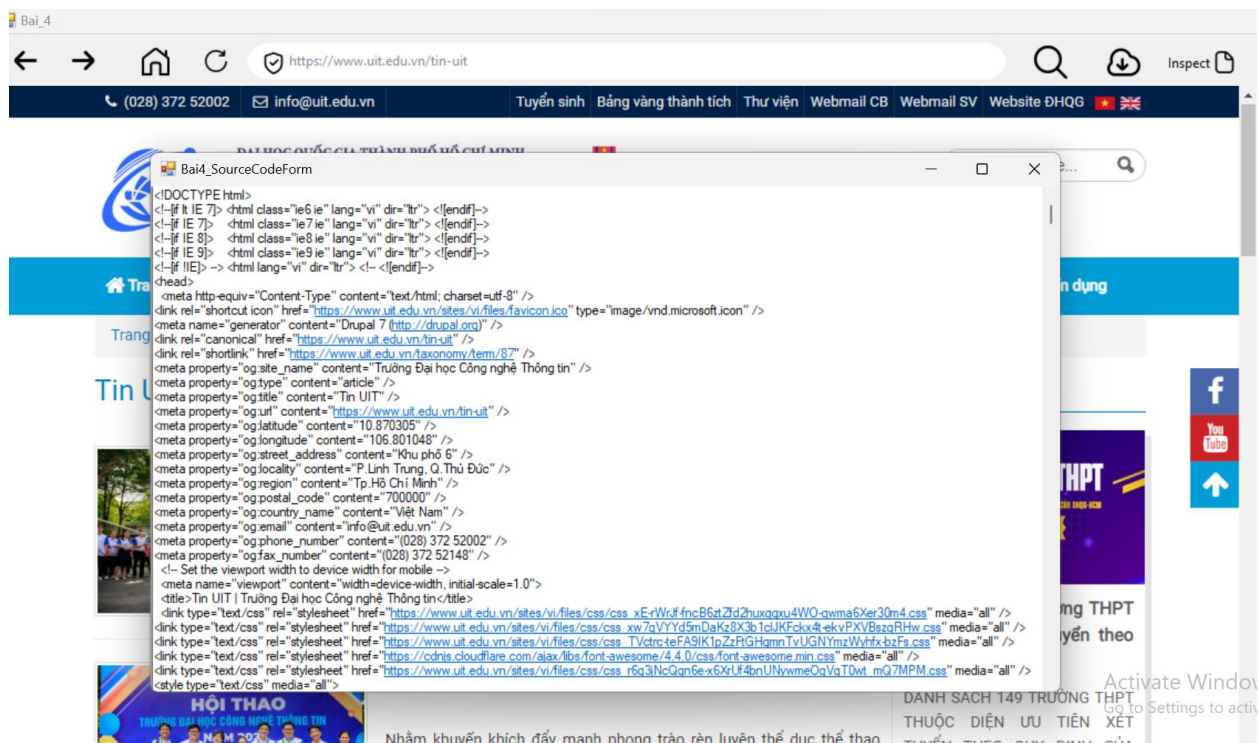
Nhấn nút quay lại. Chương trình kiểm tra nếu tồn tại trang trước đó thì chương trình sẽ có thể quay lại.



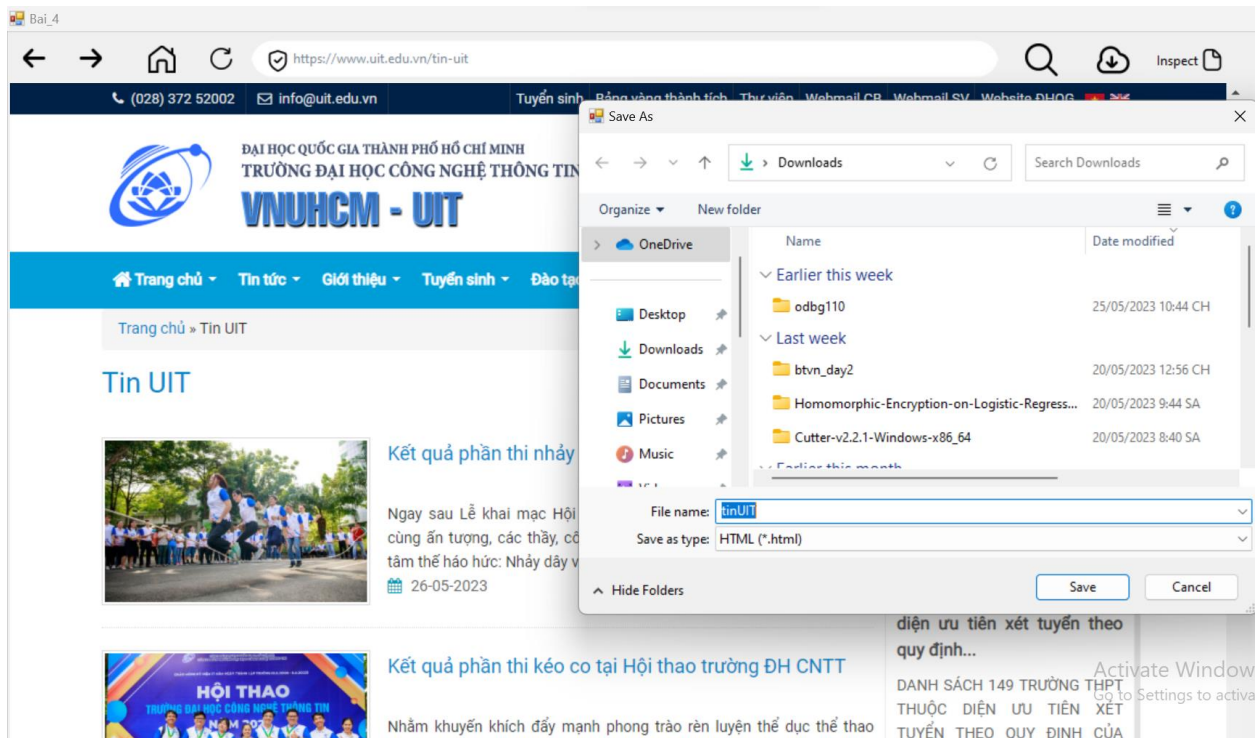
Khi nhấn phím reload thì web sẽ reload lại.



Khi nhập địa chỉ URL thì trang sẽ render ra nội dung trang web đó.



Khi nhấn nút inspect thì sẽ hiển thị source code của trang web.





Cho phép tải source code của trang web về máy.

Code:

```

public partial class Bai_4 : Form
{
    1 reference
    public Bai_4()
    {
        InitializeComponent();
    }

    // Default will display website UIT
    1 reference
    private void Bai_4_Load(object sender, EventArgs e)
    {
        InitBrowser();
    }

    1 reference
    private async Task initizated()
    {
        await webview2.EnsureCoreWebView2Async(null);
    }
    1 reference
    public async void InitBrowser()
    {
        await initizated();
        webview2.CoreWebView2.Navigate("https://courses.uit.edu.vn/login/index.php");
    }

    // Home function. Display google.com
    1 reference
    private void btnHome_Click(object sender, EventArgs e)
    {
        webview2.CoreWebView2.Navigate("https://google.com");
    }

    // Reload function
    1 reference
    private void btnReload_Click(object sender, EventArgs e)
    {
        webview2.CoreWebView2.Reload();
    }

    // Go back function
    1 reference
    private void btnBack_Click(object sender, EventArgs e)
    {
        if (webview2.CanGoBack) { webview2.GoBack(); }
    }

    // Go foward function
    1 reference
    private void btnFoward_Click(object sender, EventArgs e)
    {
        if (webview2.CanGoForward) { webview2.GoForward(); }
    }
}

```

Các chức năng chính:

```

// Download function
1 reference
private async void btnDownload_Click(object sender, EventArgs e)
{
    if (getSource(tbURL.Text) == "URL không tồn tại!")
    {
        MessageBox.Show("Vui lòng nhập URL!");
    }
    else
    {
        using (SaveFileDialog sfd = new SaveFileDialog() { Filter = "HTML|*.html", ValidateNames = true })
        {
            if (sfd.ShowDialog() == DialogResult.OK)
            {
                using (StreamWriter sw = new StreamWriter(sfd.FileName))
                {
                    await sw.WriteLineAsync(getSource(tbURL.Text));
                    MessageBox.Show("Download completed!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);
                }
            }
        }
    }
}

// Inspect function
1 reference
private void btnInspect_Click(object sender, EventArgs e)
{
    // Create an instance of SourceCodeForm
    Bai4_SourceCodeForm sourceCodeForm = new Bai4_SourceCodeForm();

    // Get the source code of the web page
    string sourceCode = getSource(tbURL.Text);

    // Set the source code in the SourceCodeForm
    sourceCodeForm.SetSourceCode(sourceCode);

    // Display the SourceCodeForm
    sourceCodeForm.ShowDialog();
}

3 references
private string getSource(string strURL)
{
    try
    {
        HtmlAgilityPack.HtmlWeb web = new HtmlAgilityPack.HtmlWeb();
        HtmlAgilityPack.HtmlDocument doc = new HtmlAgilityPack.HtmlDocument();
        doc = web.Load(strURL);
        string s = doc.Text;
        return s;
    }
    catch
    {
        return "URL không tồn tại!";
    }
}
}

```

- Chức năng Download sẽ lưu file về máy dưới dạng tệp .HTML
- Chức năng Inspect sẽ hiển thị source code trang web trong một windows form mới.