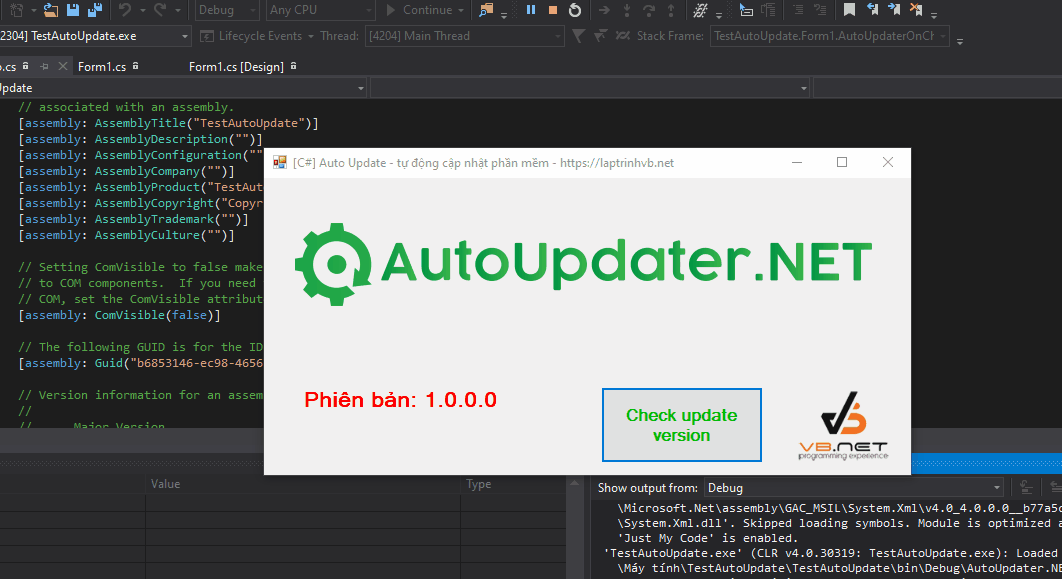
**[C#] Auto Update Application Winform**

Nếu phần mềm của bạn bị lỗi, hay bạn muốn cập nhật tính năng nào mới, bạn chỉ cần copy những file cập nhật mới lên một web server hoặc ftp server.

Khi người dùng, mở ứng dụng lên, sẽ báo cho người dùng biết có phiên bản mới, nếu người dùng bấm OK, phần mềm sẽ tự động tải về và cập nhật.

**Demo ứng dụng tự động cập nhật phần mềm C# Winform**:



Trong bài viết này mình sẽ giới thiệu đến các bạn thư viện [AutoUpdater.NET.](https://github.com/ravibpatel/AutoUpdater.NET)

Với thư viện này, các bạn sẽ dễ dàng tích hợp module cập nhật ứng dụng cho App của mình.

Đầu tiên, các bạn cần import thư viện **AutoUpdater.NET** vào project bằng Nuget Console, phiên bản hiện tại là 1.6.3

PM> Install-Package Autoupdater.NET.Official -Version 1.6.3

C#

Copy

Để xây dựng module này các bạn tạo cho mình 1 file update.xml với nội dung như sau và sau đó copy lên Server Web của mình nhé.

<?xml version="1.0" encoding="UTF-8"?>

<item>

<version>2.0.0.0</version>

<url>https://laptrinhvb.net/TestAutoUpdate.zip</url>

<changelog>https://laptrinhvb.net</changelog>

<mandatory>false</mandatory>

</item>

Markup

Copy

**Version**: 2.0.0.0 là để cho máy client đọc phiên bản mới nhất để so sánh với phiên bản hiện tại.

**url**: là đường dẫn để cho thư viện tải phiên bản cập nhật về và tự động giải nén.

Các bạn nhớ nén thành file .ZIP nhé

**Changelog**: đường dẫn đến website hoặc file html để thông báo cho người dùng phiên bản mới có những tính năng nào mới hoặc fix bug gì.

**Mandatory:** nếu bạn set = true, thì bắt buộc người dùng phải cập nhật phần mềm, bỏ qua skip update



Và dưới đây là **source code ứng dụng tự động cập nhật phần mềm C#**, Winform:

using AutoUpdaterDotNET;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace TestAutoUpdate

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

System.Reflection.Assembly assembly = System.Reflection.Assembly.GetExecutingAssembly();

System.Diagnostics.FileVersionInfo fvi = System.Diagnostics.FileVersionInfo.GetVersionInfo(assembly.Location);

AutoUpdater.CheckForUpdateEvent += AutoUpdaterOnCheckForUpdateEvent;

string version = fvi.FileVersion;

label1.Text = "Phiên bản: " + version;

AutoUpdater.DownloadPath = "update";

System.Timers.Timer timer = new System.Timers.Timer

{

Interval = 15 \* 60 \* 1000,

SynchronizingObject = this

};

timer.Elapsed += delegate

{

AutoUpdater.Start("https://laptrinhvb.net/update.xml");

};

timer.Start();

}

private void AutoUpdaterOnCheckForUpdateEvent(UpdateInfoEventArgs args)

{

if (args.IsUpdateAvailable)

{

DialogResult dialogResult;

dialogResult =

MessageBox.Show(

$@"Bạn ơi, phần mềm của bạn có phiên bản mới {args.CurrentVersion}. Phiên bản bạn đang sử dụng hiện tại {args.InstalledVersion}. Bạn có muốn cập nhật phần mềm không?", @"Cập nhật phần mềm",

MessageBoxButtons.YesNo,

MessageBoxIcon.Information);

if (dialogResult.Equals(DialogResult.Yes) || dialogResult.Equals(DialogResult.OK))

{

try

{

if (AutoUpdater.DownloadUpdate(args))

{

Application.Exit();

}

}

catch (Exception exception)

{

MessageBox.Show(exception.Message, exception.GetType().ToString(), MessageBoxButtons.OK,

MessageBoxIcon.Error);

}

}

}

else

{

MessageBox.Show(@"Phiên bản bạn đang sử dụng đã được cập nhật mới nhất.", @"Cập nhật phần mềm",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

private void button1\_Click(object sender, EventArgs e)

{

AutoUpdater.Start("https://laptrinhvb.net/update.xml");

}

}

}

C#

Copy

Chi tiết và cách sử dụng thêm các bạn có thể vào**trang AutoUpdater.NET** để tham khảo tài liệu của nó nhé.

Thanks for watching!