**Giới thiệu**

Một ứng dụng được triển khai bằng ClickOne sẽ có các đặc điểm:

–          Bạn có thể tùy chọn triển khai ứng dụng offline hoặc online.

–          Ứng dụng sau khi cài đặt sẽ được nằm trên Start Menu và trong Control Panel/Add-Remove Programs.

–          Khi thực thi ứng dụng, người dùng sẽ chạy file manifest (ClickOnce Application Reference) có đuôi là .appref-ms. File này sẽ chịu trách nhiệm chạy ứng dụng thực sự (file .exe) và kiểm tra cập nhật. Việc kiểm tra cập nhật sẽ được dựa vào mã hash của các tập tin để xác định phiên bản mới.

**Ứng dụng auto-update với giao diện OneClick trong Visual Studio**

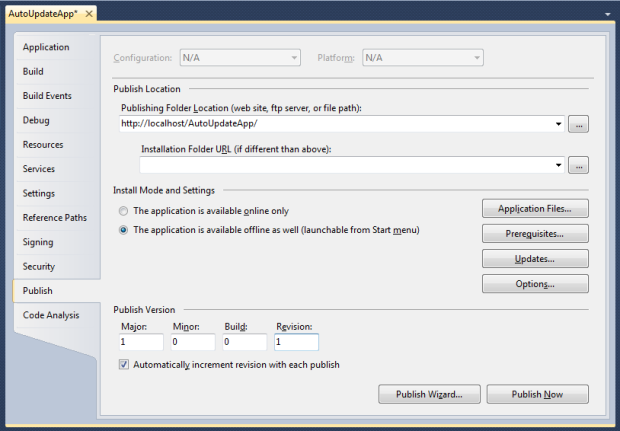
Để bắt đầu, tôi tạo một ứng dụng Windows Form với tên AutoUpdateApp (bạn có thể tạo các loại ứng dụng desktop khác như Console hoặc WPF), sau đó vào phần Project Properties trong Visual Studio và chuyển qua thẻ Publish.

Tại phần Publish Location:

–          **Publishing Folder Location (web site, ftp server, or file path):**đường dẫn mà các tập tin dùng để cài đặt sẽ được tạo ra. Ở đây tôi sử dụng IIS để tạo server và đặt là <http://localhost/AutoUpdateApp/>

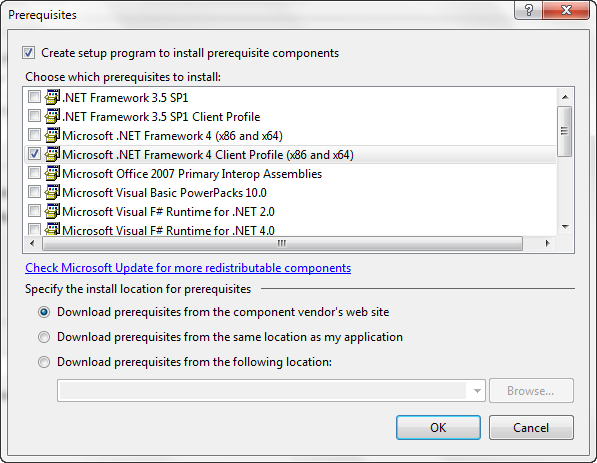
–          **Installation Folder URL (if different than above):** để trống mục này vì tôi sử dụng thư mục publish cùng với thư mục để cài đặt.

Phần Install Mode and Settings, tôi chọn “**The application is available offline as well …**” để cho phép ứng dụng chạy offline và cho phép sử dụng tính năng update.

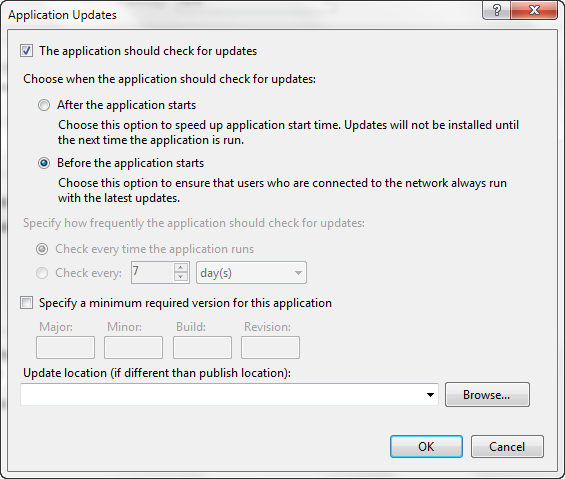
[](https://yinyangit.wordpress.com/2011/10/06/visual-studio-2010-trien-khai-ung-dung-tu-cap-nhat-auto-update-voi-oneclick/visual-studio-publish-tab/)

Nhấn nút **Application Files…** và lựa chọn các file bạn muốn được copy chung vào thư mục ứng dụng.

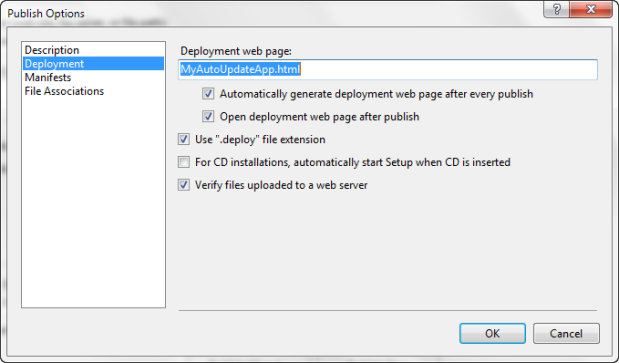
Nhấn nút **Prerequisities…** để lựa chọn các thành phần cần thiết để ứng dụng hoạt động, mặc định VS sẽ tự động lựa chọn dựa vào dự án của bạn.

[](https://yinyangit.wordpress.com/2011/10/06/visual-studio-2010-trien-khai-ung-dung-tu-cap-nhat-auto-update-voi-oneclick/visual-studio-prerequisites/)

Nhấn nút **Updates…,** trong bảng này bạn chọn “**The application should check for updates**”, chọn thời điểm kiểm tra update và đặt Update location chỉ trong trường hợp đường dẫn này khác với Publish location.

[](https://yinyangit.wordpress.com/2011/10/06/visual-studio-2010-trien-khai-ung-dung-tu-cap-nhat-auto-update-voi-oneclick/visual-studio-application-updates/)

Nhấn nút **Options…**, thiết lập các thông tin trong phần Description để hiển thị lên trang web cài đặt. Trong phần Deployment, thiết lập tên web page và tùy chọn đánh dấu hai tùy chọn ngay bên dưới để VS tự động sinh ra và mở trang web cài đặt.

[](https://yinyangit.wordpress.com/2011/10/06/visual-studio-2010-trien-khai-ung-dung-tu-cap-nhat-auto-update-voi-oneclick/visual-studio-publish-options/)

**Note**: Thay vì thiết lập theo cách này, bạn có thể nhấn nút **Publish Wizard** bên dưới để thực hiện từng bước theo trình wizard. Sau khi thiết lập hoàn tất, bạn chỉ cần nhấn nút **Publish Now** sau mỗi lần ứng dụng thay đổi.

Sau khi public, trang web cài đặt sẽ được mở ra để bản tải về file setup. File này sẽ tự động cài ứng dụng vào hệ thống và lên Start Menu.

[A screenshot of a computer

Description automatically generated](https://yinyangit.wordpress.com/2011/10/06/visual-studio-2010-trien-khai-ung-dung-tu-cap-nhat-auto-update-voi-oneclick/visual-studio-auto-generated-oneclick-web-page/)

Khi publish phiên bản mới, chương trình sẽ hiển thị thông điệp yêu cầu cập nhật.

[A screenshot of a computer error

Description automatically generated](https://yinyangit.wordpress.com/2011/10/06/visual-studio-2010-trien-khai-ung-dung-tu-cap-nhat-auto-update-voi-oneclick/visual-studio-oneclick-app-autoupdate-alert/)

**Ứng dụng auto-update với OneClick Deployment API**

Để sử dụng cách này, bạn vẫn phải thiết lập các thông tin cơ bản và đường dẫn cập nhật. Tuy nhiên bạn không bật chức năng tự động cập nhật mà chỉ cần gọi phương thức sau tại thời điểm cần cập nhật (Nguồn: [How to: Check for Application Updates Programmatically Using the ClickOnce Deployment API](http://msdn.microsoft.com/en-us/library/ms404263.aspx))

C# code:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68 | **private** **void** InstallUpdateSyncWithInfo()  {      UpdateCheckInfo info = **null**;    **if** (ApplicationDeployment.IsNetworkDeployed)      {          ApplicationDeployment ad = ApplicationDeployment.CurrentDeployment;    **try**          {              info = ad.CheckForDetailedUpdate();            }  **catch** (DeploymentDownloadException dde)          {              MessageBox.Show("The new version of the application cannot be downloaded at this time. \n\nPlease check your network connection, or try again later. Error: " + dde.Message);  **return**;          }  **catch** (InvalidDeploymentException ide)          {              MessageBox.Show("Cannot check for a new version of the application. The ClickOnce deployment is corrupt. Please redeploy the application and try again. Error: " + ide.Message);  **return**;          }  **catch** (InvalidOperationException ioe)          {              MessageBox.Show("This application cannot be updated. It is likely not a ClickOnce application. Error: " + ioe.Message);  **return**;          }    **if** (info.UpdateAvailable)          {              Boolean doUpdate = **true**;    **if** (!info.IsUpdateRequired)              {                  DialogResult dr = MessageBox.Show("An update is available. Would you like to update the application now?", "Update Available", MessageBoxButtons.OKCancel);  **if** (!(DialogResult.OK == dr))                  {                      doUpdate = **false**;                  }              }  **else**              {                  // Display a message that the app MUST reboot. Display the minimum required version.                  MessageBox.Show("This application has detected a mandatory update from your current " +                      "version to version " + info.MinimumRequiredVersion.ToString() +                      ". The application will now install the update and restart.",                      "Update Available", MessageBoxButtons.OK,                      MessageBoxIcon.Information);              }    **if** (doUpdate)              {  **try**                  {                      ad.Update();                      MessageBox.Show("The application has been upgraded, and will now restart.");                      Application.Restart();                  }  **catch** (DeploymentDownloadException dde)                  {                      MessageBox.Show("Cannot install the latest version of the application. \n\nPlease check your network connection, or try again later. Error: " + dde);  **return**;                  }              }          }      }  } |

Khi phát hiện bản cập nhật mới, chương trình sẽ hiển thị thông điệp khác và yêu cầu restart lại ứng dụng:

[A screenshot of a computer error

Description automatically generated](https://yinyangit.wordpress.com/2011/10/06/visual-studio-2010-trien-khai-ung-dung-tu-cap-nhat-auto-update-voi-oneclick/visual-studio-oneclick-app-custom-autoupdate-alert/)

**Chương trình minh họa**

Sử dụng phương pháp này, bạn sẽ không được phép lựa chọn đường dẫn cài đặt vì vậy tôi thêm chức năng mở thư mục ứng dụng vào chương trình. Và thêm một ListView để hiển thị các thông tin assembly liên quan, khi bạn thay đổi Target framework, các thông tin này cũng được cập nhật và thay đổi theo:

[A screenshot of a computer

Description automatically generated](https://yinyangit.wordpress.com/2011/10/06/visual-studio-2010-trien-khai-ung-dung-tu-cap-nhat-auto-update-voi-oneclick/autoupdateapplication-demo/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50 | **using** System;  **using** System.Windows.Forms;  **using** System.Reflection;  **using** System.Diagnostics;  **using** System.Deployment.Application;    **namespace** AutoUpdateApp  {  **public** **partial** **class** Form1 : Form  {  **public** Form1()      {          InitializeComponent();    **try**          {              // chi chay duoc sau khi ung dung da cai dat  **this**.Text = Application.ProductName + " " + ApplicationDeployment.CurrentDeployment.CurrentVersion;          }  **catch**          { }            label1.Text = Application.StartupPath;            Assembly asm = Assembly.GetExecutingAssembly();            AssemblyName[] asmNames = asm.GetReferencedAssemblies();    **foreach** (AssemblyName nm **in** asmNames)          {              ListViewItem item = **new** ListViewItem(nm.Name);              item.SubItems.Add(nm.FullName);              listView1.Items.Add(item);          }        }    **private** **void** btnExit\_Click(**object** sender, EventArgs e)      {  **this**.Close();      }    **private** **void** btnOpbenFolder\_Click(**object** sender, EventArgs e)      {  **string** arg = "/select, " + Application.ExecutablePath;          Process.Start("explorer", arg);      }    }  } |

[https://yinyangit.wordpress.com](https://yinyangit.wordpress.com/)