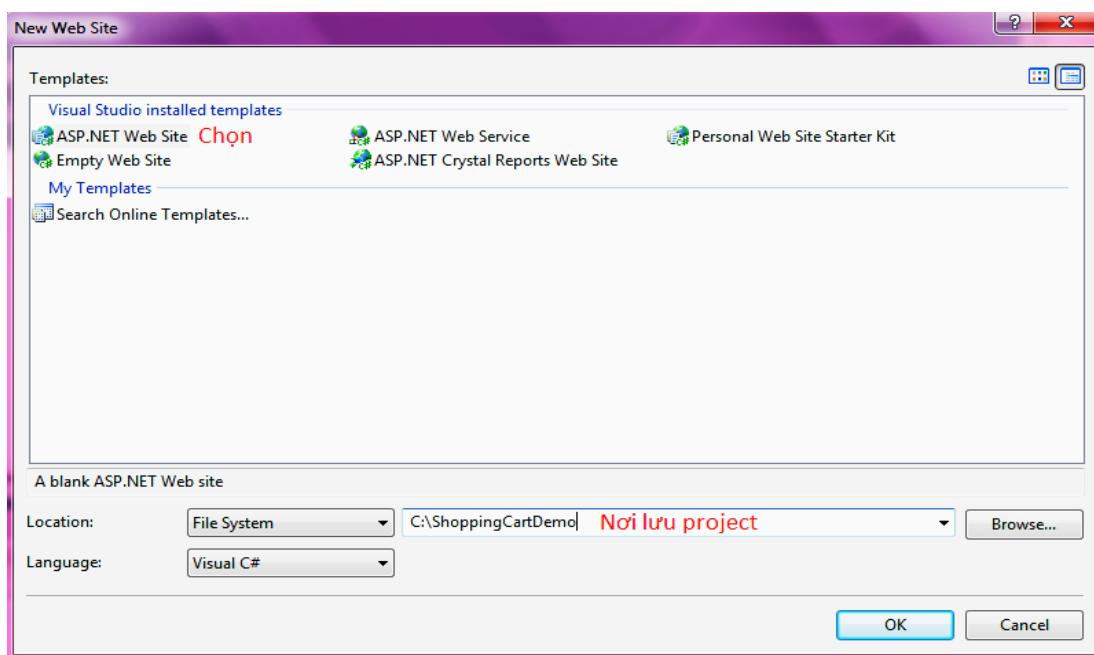
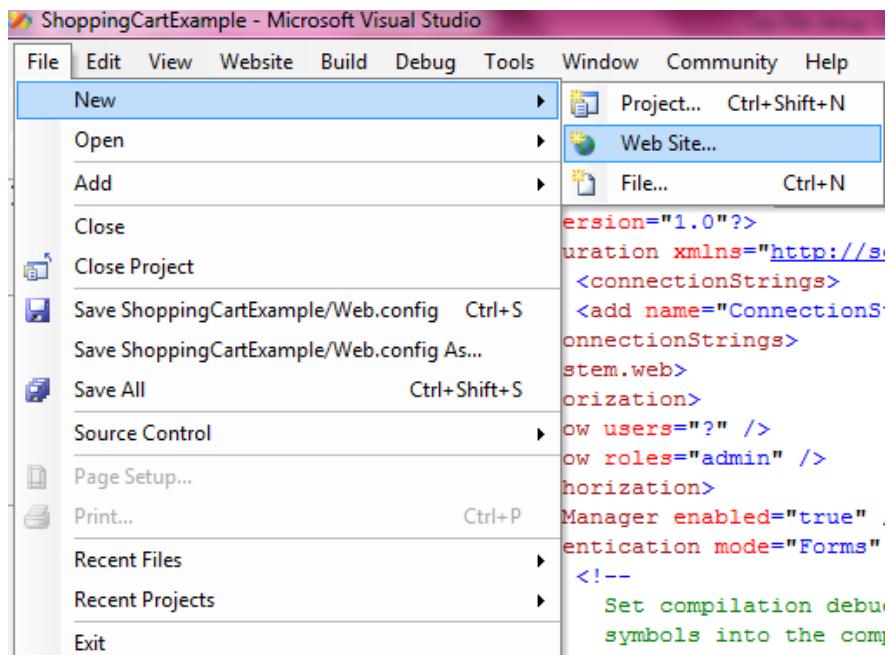


## PHỤ LỤC

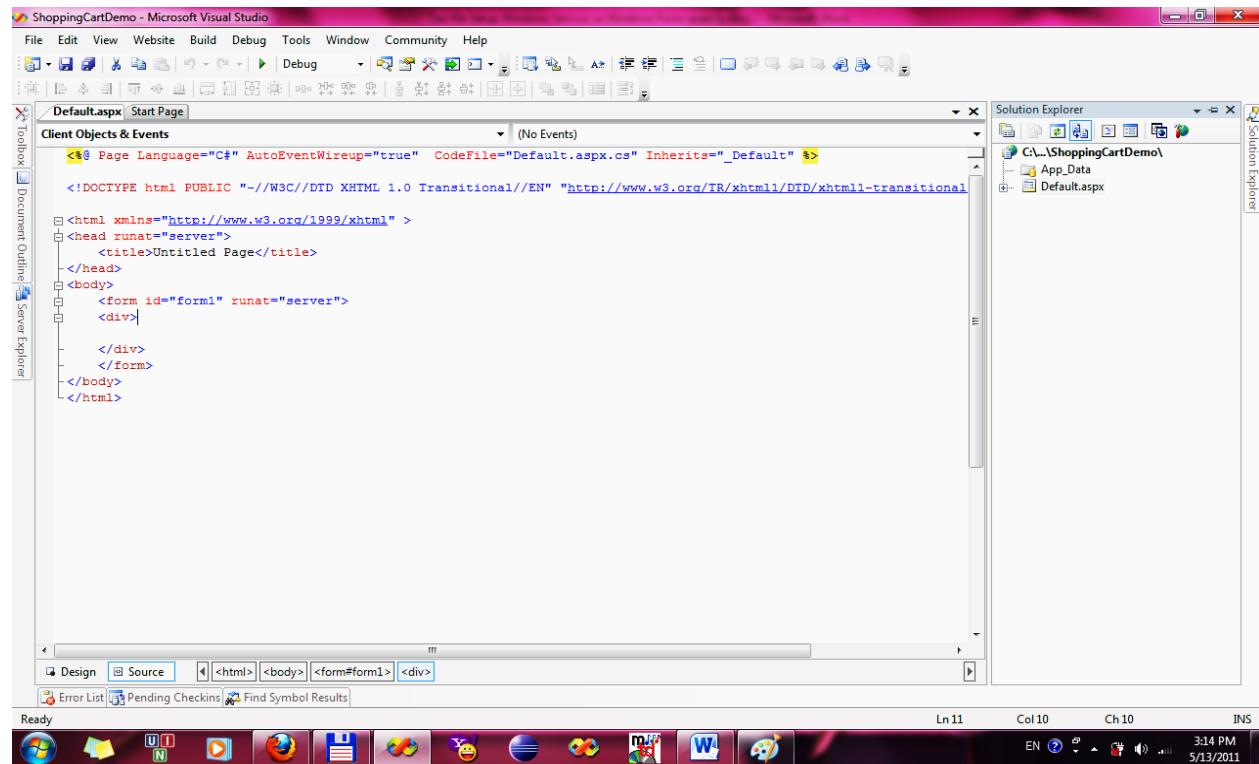
### HƯỚNG DẪN 1: SHOPPING CART

**Bước 1:** Đầu tiên mở chương trình Microsoft Visual Studio lên.

**Bước 2:** Tạo một project dạng website File → New → Web Site.

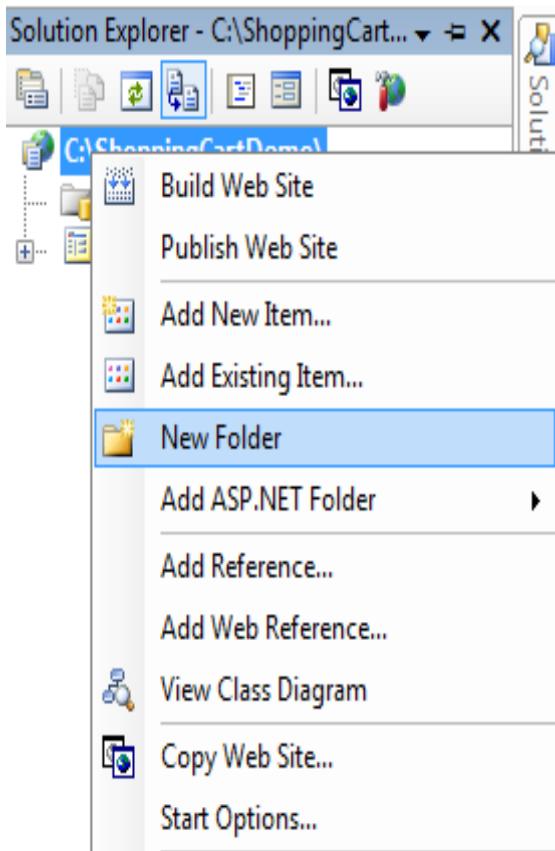


Sau khi bấm OK. Bạn đã có một website ban đầu



#### **Bước 4:** Tạo folder chứa các hình ảnh của sản phẩm.

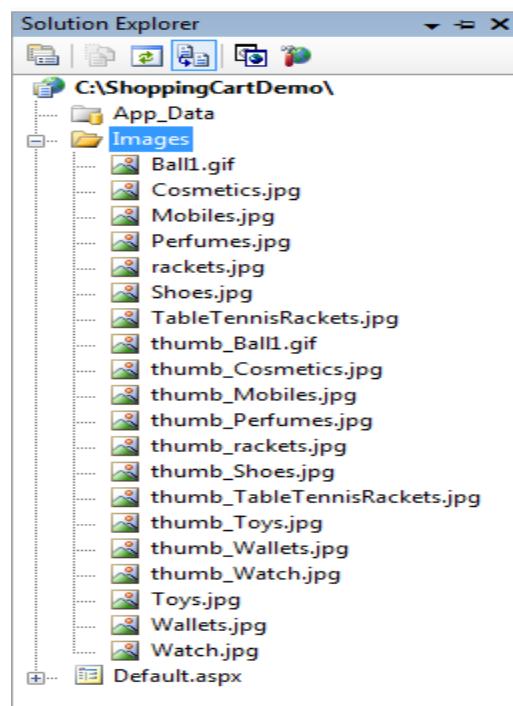
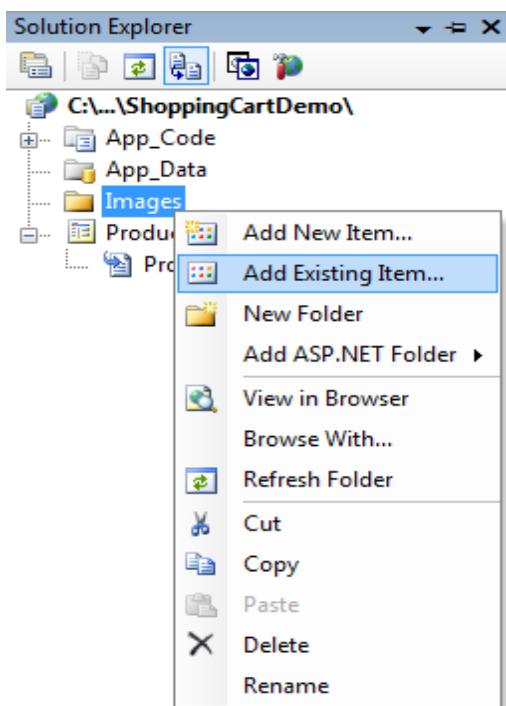
Click chuột phải vào C:\ ShoppingCartDemo\ → New Folder. Rồi đặt tên là Images.  
Rồi chép các hình ảnh sản phẩm vào thư mục này.



### Documents library

Images
Name
Ball1.gif
Cosmetics.jpg
Mobiles.jpg
Perfumes.jpg
rackets.jpg
Shoes.jpg
TableTennisRackets.jpg
thumb_Ball1.gif
thumb_Cosmetics.jpg
thumb_Mobiles.jpg
thumb_Perfumes.jpg
thumb_rackets.jpg
thumb_Shoes.jpg
thumb_TableTennisRackets.jpg
thumb_Toys.jpg
thumb_Wallets.jpg
thumb_Watch.jpg
Toys.jpg
Wallets.jpg
Watch.jpg

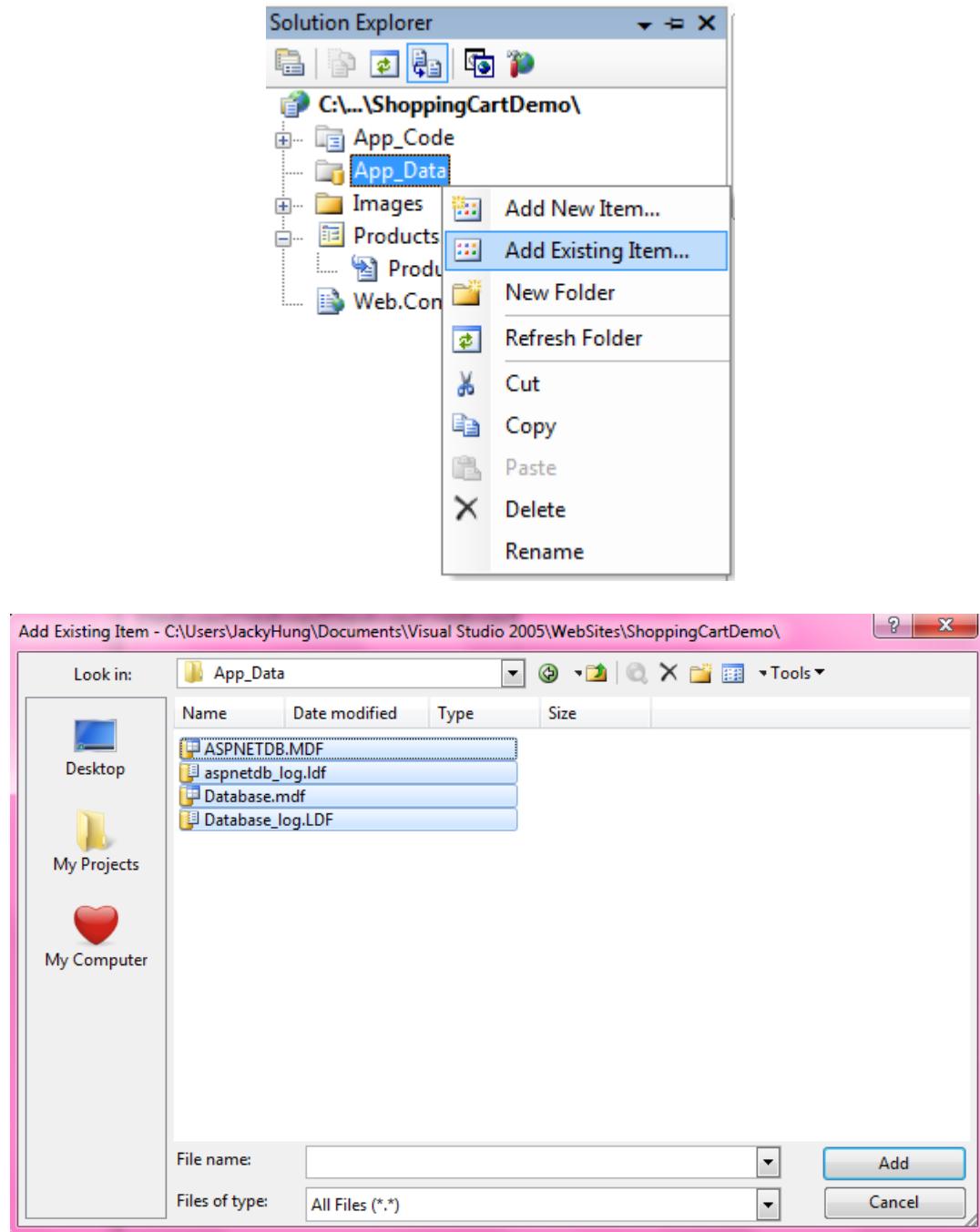
Click chuột phải vào Images → Add Existing Item. Rồi chọn hết hình ảnh sản phẩm ở trên vào.



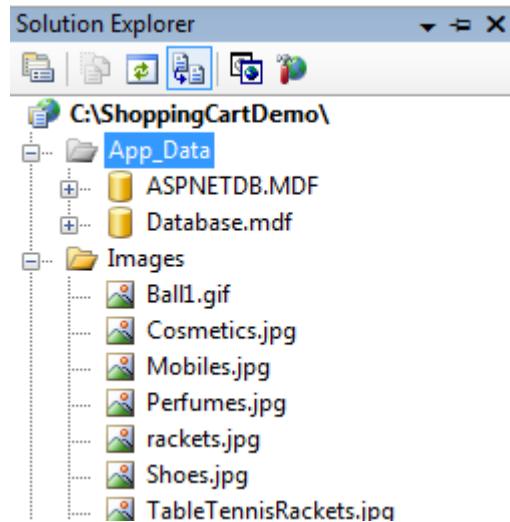
**Bước 5:** Add database cho web.

Chép các database vào thư mục App\_Data.

Click chuột phải vào App\_Data → Add Existing Item. Rồi chọn database của web attach vào.

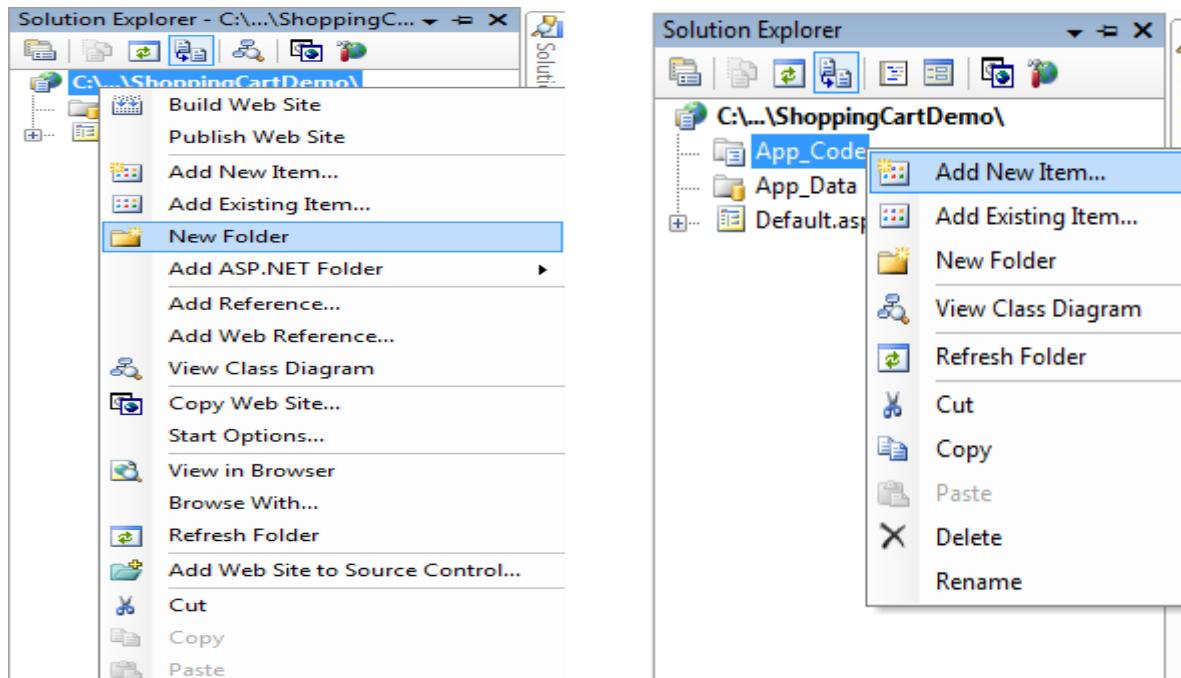


Bấm Add. Được kết quả

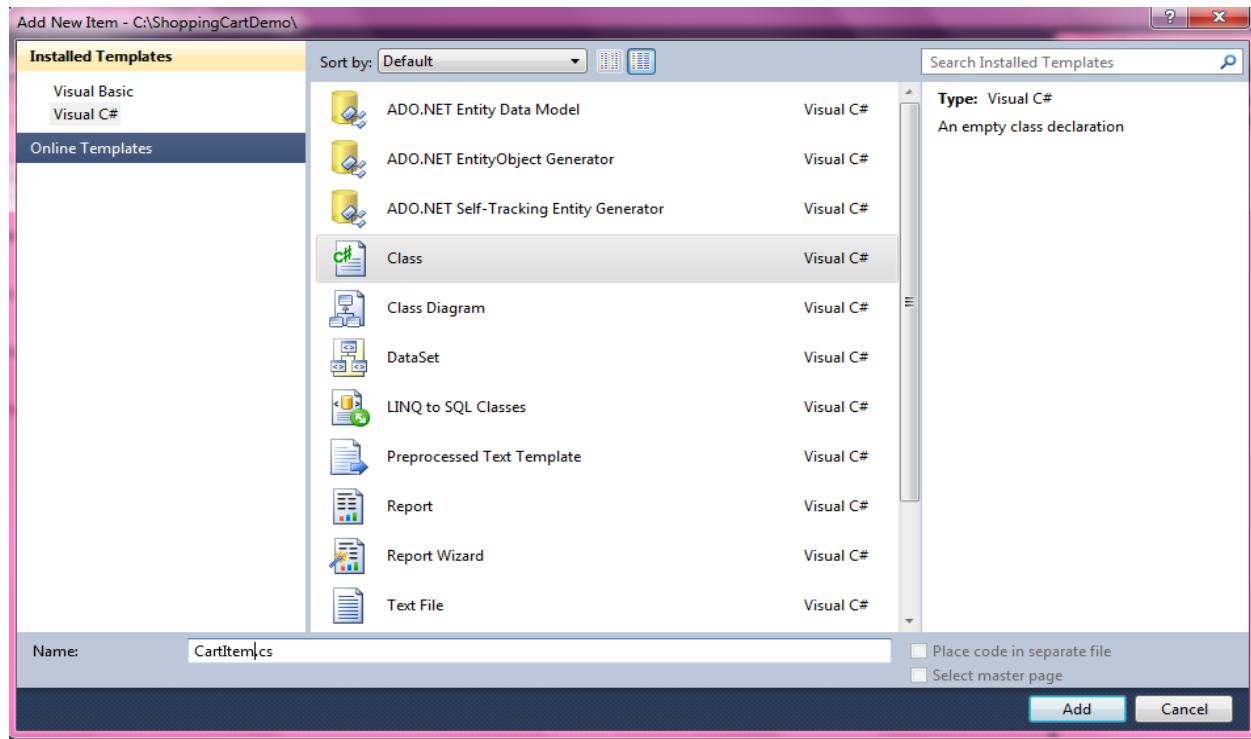


**Bước 6:** Tạo folder chứa code dạng cs.

Click chuột phải vào C:\ShoppingCartDemo\ → New Folder. Rồi đặt tên là App\_Code. Rồi click chuột phải vào folder App\_Code. Chọn Add New Item....



Rồi chọn như sau:



**Bước 7:** Trong file CartItem.cs sửa lại như sau.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

[Serializable]

public class CartItem
{
    private int _productID; // mã sản phẩm
    private string _productName; //tên sản phẩm
    private string _imageUrl; // đường dẫn hình ảnh sản phẩm
    private int _quantity; // số lượng
    private double _price; // đơn giá
}

```

```
private double _subTotal; // thành tiền = đơn giá * số lượng

public CartItem() // constructor
{
}

public CartItem(int ProductID, string ProductName, string ImageUrl, int
Quantity, double Price)

{
    _productID = ProductID;
    _productName = ProductName;
    _imageUrl = ImageUrl;
    _quantity = Quantity;
    _price = Price;
    _subTotal = Quantity * Price;
}

public int ProductID // các phương thức set và get cho thuộc tính
{
    get { return _productID; }
    set { _productID = value; }
}

public string ProductName
{
    get { return _productName; }
    set { _productName = value; }
}

public string ImageUrl
{
```

```
get { return _imageUrl; }

set { _imageUrl = value; }

}

public int Quantity

{

    get { return _quantity; }

    set { _quantity = value; }

}

public double Price

{

    get { return _price; }

    set { _price = value; }

}

public double SubTotal

{

    get { return _quantity * _price; }

}

}
```

**Bước 8:** Tương tự tạo file Cart.cs và có code như sau.

```
using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;
```

[Serializable]

```
public class Cart

{
    private DateTime _dateCreated; // ngày tạo giỏ hàng
    private DateTime _lastUpdate; // ngày cập nhập giỏ hàng
    private List<CartItem> _items; // danh sách mặt hàng trong giỏ

    public Cart() // constructor
    {
        if (this._items == null)
        {
            this._items = new List<CartItem>();
            this._dateCreated = DateTime.Now;
        }
    }

    public List<CartItem> Items // phương thức set và get cho mặt hàng
    {
        get { return _items; }
        set { _items = value; }
    }

    public void Insert(int ProductID, double Price, int Quantity, string ProductName,
string ImageUrl)// thêm một mặt hàng vào giỏ hàng
    {
        int ItemIndex = ItemIndexOfID(ProductID);
        if (ItemIndex == -1)
        {
```

```
CartItem newItem = new CartItem(); // tạo mặt hàng

    newItem.ProductID = ProductID;
    newItem.Quantity = Quantity;
    newItem.Price = Price;
    newItem.ProductName = ProductName;
    newItem.ImageUrl = ImageUrl;
    _items.Add(newItem); // thêm hàng vào giỏ

}

else

{

    _items[RowIndex].Quantity += 1;

}

_lastUpdate = DateTime.Now; // cập nhập lại ngày cập nhật của giỏ hàng

}

public void Update(int RowID, int ProductID, int Quantity, double Price) // cập
nhập lại mặt hàng tại dòng thứ rowid trong giỏ hàng

{

    CartItem Item = _items[RowIndex];
    Item.ProductID = ProductID;
    Item.Quantity = Quantity;
    Item.Price = Price;
    _lastUpdate = DateTime.Now;

}

public void DeleteItem(int rowID) // xóa mặt hàng thứ rowID trong giỏ hàng

{
```

```
_items.RemoveAt(rowID);

_lastUpdate = DateTime.Now;

}

private int ItemIndexOfID(int ProductID) // tìm mặt hàng có mã là ProductID
nằm ở dòng thứ mấy trong giỏ hàng

{

    int index = 0;

    foreach (CartItem item in _items)

    {

        if (item.ProductID == ProductID)

        {

            return index;

        }

        index += 1;

    }

    return -1;

}

public double Total // tổng tiền của giỏ hàng

{

    get

    {

        double t = 0;

        if (_items == null)

        {

            return 0;
```

```
    }

    foreach (CartItem Item in _items)

    {
        t += Item.SubTotal;

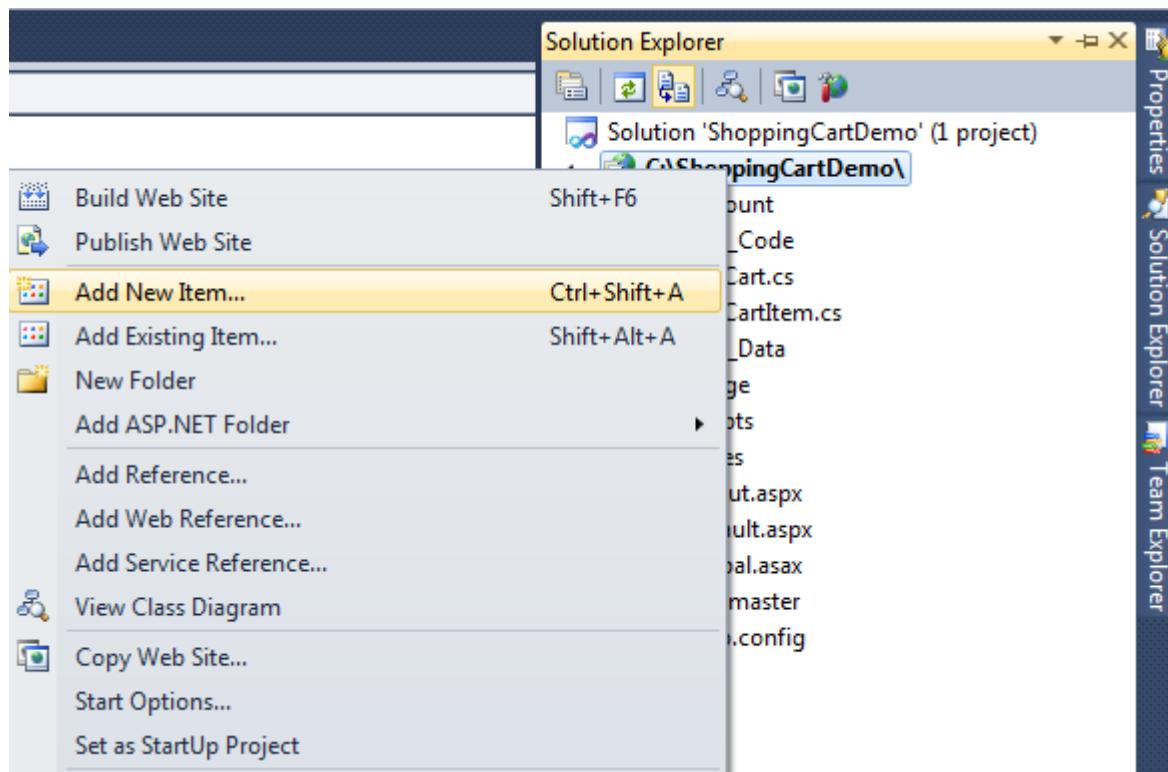
    }

    return t;
}

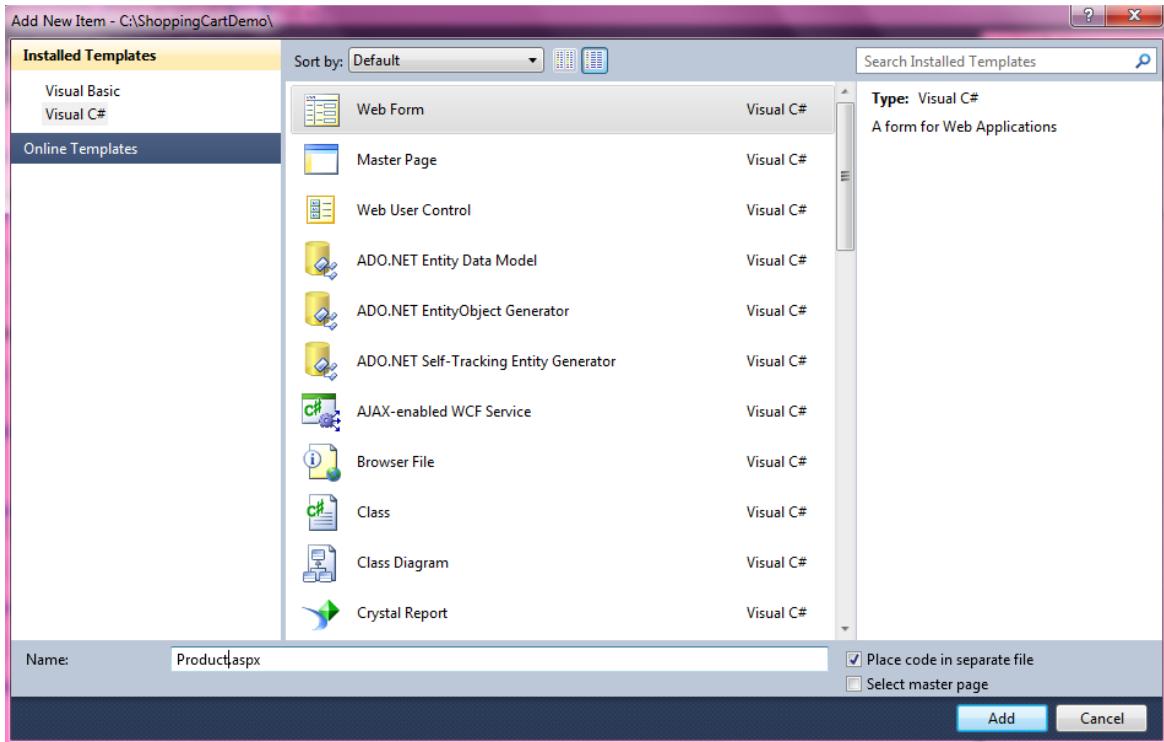
}
```

### **Bước 9:** Tạo ra trang products page.

Click chuột phải vào project và chọn add new item như sau:



Sau đó chọn Web form. Rồi đặt tên trang là Product.aspx



C# phát sinh ra trang có nội dung như sau

```

Product.aspx  X  App_Code/Cart.cs*  App_Code/CartItem.cs  Default.aspx
Client Objects & Events  (No Events)
1  <%@ Page Language="C#" AutoEventWireup="true" CodeFile="Product.aspx.cs" Inherits="Product" %>
2
3  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
4
5  <html xmlns="http://www.w3.org/1999/xhtml">
6  <head runat="server">
7      <title></title>
8  </head>
9  <body>
10 <form id="form1" runat="server">
11     <div>
12
13     </div>
14 </form>
15 </body>
16 </html>

```

Bạn thêm đoạn sau vào trong thẻ form như sau:

```

<div>

    <asp:SqlDataSource           ID="SqlDataSource1"           runat="server"
ConnectionString=<%$ ConnectionStrings:ConnectionString %>

        SelectCommand="SELECT [ProductID], [Name], [Description], [Price],
[ImageUrl] FROM [Products]">

    </asp:SqlDataSource>

```

```

</div>

<asp:DataList ID="DataList1" runat="server"
DataSourceID="SqlDataSource1" RepeatColumns="4"
RepeatDirection="Horizontal">

<ItemTemplate>

    <asp:ImageButton ID="ImageButton1" runat="server"
ImageUrl='<%# Eval("ImageUrl", "Images\\thumb_{0}") %>' PostBackUrl='<%#
Eval("ProductID", "ProductDetails.aspx?ProductID={0}") %>' /><br />
        <asp:Label ID="NameLabel" runat="server" Text='<%# Eval("Name")%>'></asp:Label><br />
        <asp:Label ID="PriceLabel" runat="server" Text='<%# Eval("Price",
"{0:C}") %>'></asp:Label><br />
        <br />
    </ItemTemplate>
</asp:DataList>
<br />
<asp:HyperLink ID="CartLink" runat="server"
NavigateUrl="~/UserCart.aspx">View Shopping Cart</asp:HyperLink><br />

```

Tại dòng code `ConnectionString="<%$ ConnectionStrings:ConnectionString %>"`

Bạn thay đổi chuỗi kết nối đến cơ sở dữ liệu của trang web. Ví dụ ở đây là

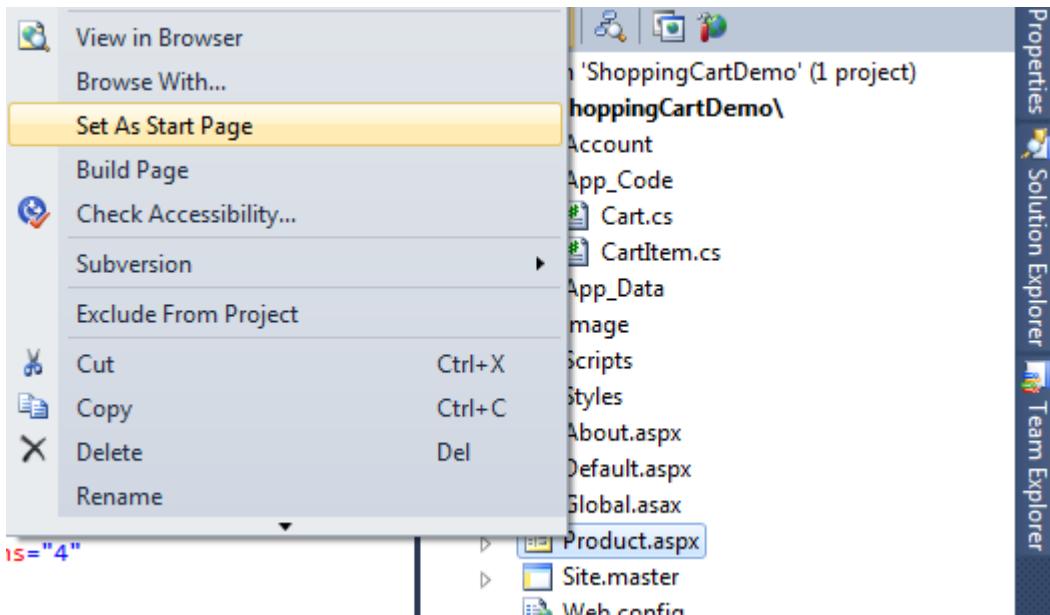
```

ConnectionString="Data Source=JACKYHUNG-PC\SQLEXPRESS;
AttachDbFilename=C:\ShoppingCartDemo\App_Data\Database.mdf;Integrated
Security=True;User Instance=True"

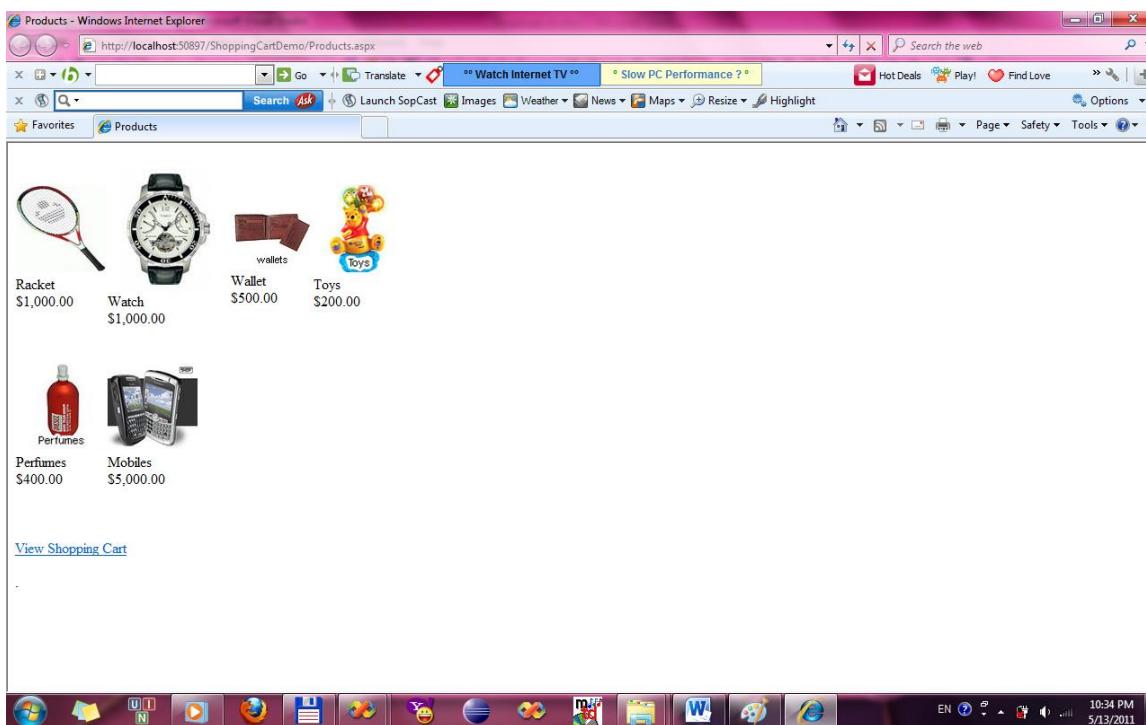
```

Bạn thiết lập page Product.aspx là page chính khi chạy project lên như sau:

Click chuột phải vào Product.aspx ➔ Set as start page.

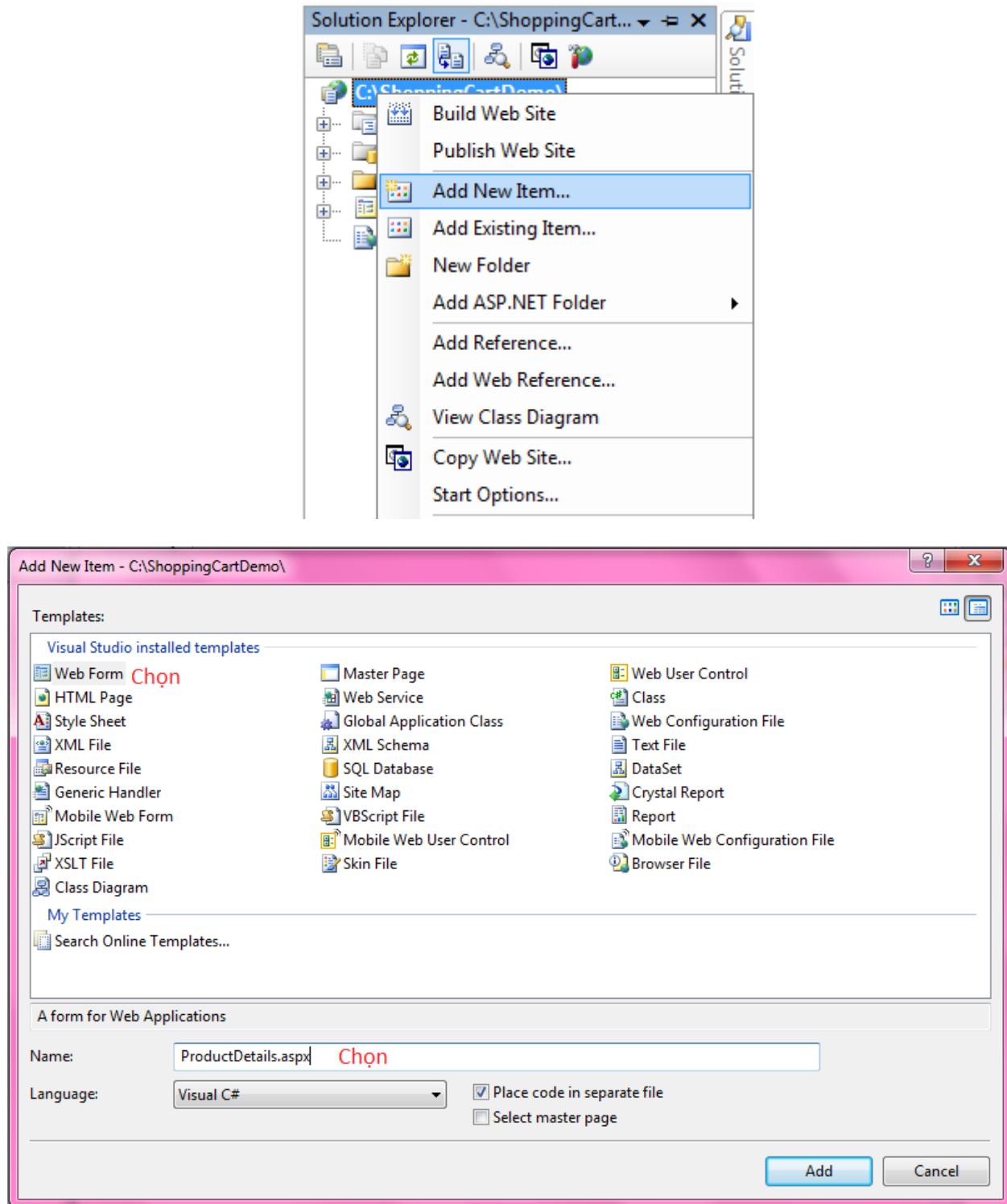


### Bước 10: Chạy và xem thử kết quả.



### Bước 11: Tạo ra trang product details page.

Click chuột phải vào C:\ ShoppingCartDemo \ → Add New Item.



Trong file ProductDetails.aspx. bạn thay đổi code như sau

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="ProductDetails.aspx.cs" Inherits="ProductDetails" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title>Product Details</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%$ ConnectionStrings:ConnectionString %>" 
SelectCommand="SELECT [ProductID], [Name], [Description], [Price],
[ImageUrl] FROM [Products] WHERE ([ProductID] = @ProductID)">
                <SelectParameters>
                    <asp:QueryStringParameter Name="ProductID"
QueryStringField="ProductID" Type="Decimal" />
                </SelectParameters>
            </asp:SqlDataSource>
        </div>
    </form>
</body>
</html>
```

```
<asp:DataList ID="DataList1" runat="server"
DataSourceID="SqlDataSource1">

    <ItemTemplate>
        <asp:Image ID="Image1" runat="server" ImageUrl='<%# Eval("ImageUrl", "~/Images\\{0}") %>' /><br />
        <asp:Label ID="ImageUrlLabel" runat="server" Text='<%# Eval("ImageUrl") %>' Visible="False"></asp:Label><br />
        <asp:Label ID="NameLabel" runat="server" Text='<%# Eval("Name") %>'></asp:Label><br />
        <asp:Label ID="DescriptionLabel" runat="server" Text='<%# Eval("Description") %>'></asp:Label><br />
        <asp:Label ID="PriceLabel" runat="server" Text='<%# Eval("Price", "{0:#0.00}") %>'></asp:Label><br />
        <br />
        <br />
    </ItemTemplate>
</asp:DataList><br />
<asp:Button ID="btnAdd" runat="server" OnClick="Button1_Click" Text="Add to Cart" /><br />
<br />
    &nbsp;<asp:HyperLink ID="HyperLink1" runat="server" NavigateUrl="~/Products.aspx">Return to Products Page</asp:HyperLink>
</form>
</body>
</html>
```

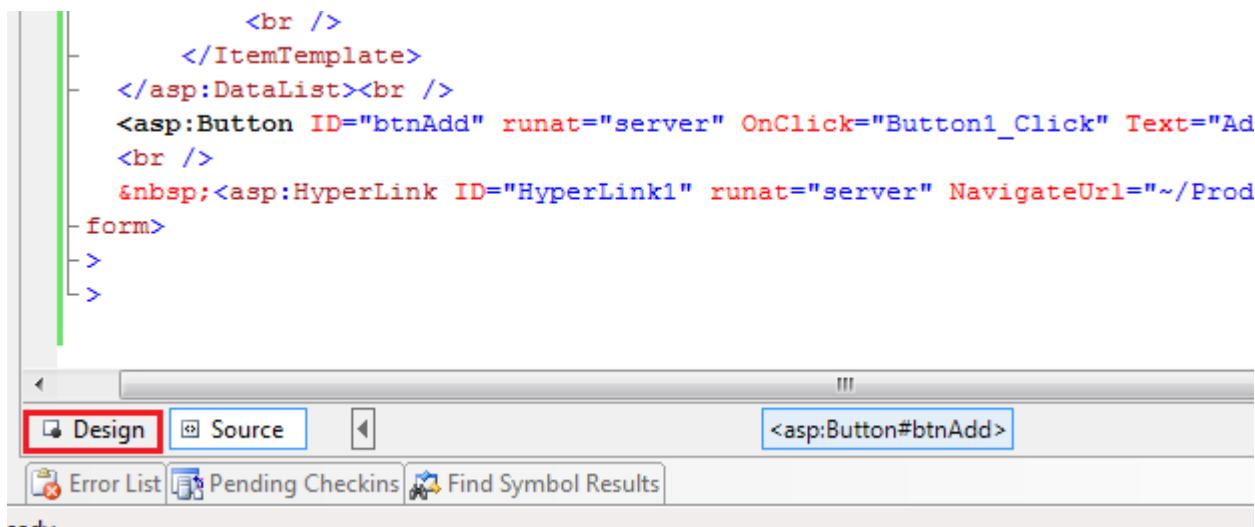
Tại dòng code `ConnectionString="<%$ ConnectionStrings:ConnectionString %>"`

Bạn thay đổi chuỗi kết nối đến cơ sở dữ liệu của trang web. Ví dụ ở đây là

```
ConnectionString="Data Source=JACKYHUNG-
PC\SQLEXPRESS;AttachDbFilename=C:\ShoppingCartDemo\App_Data\Database.mdf;
Integrated Security=True;User Instance=True"
```

**Bước 12:** Viết code xử lý cho button Add To Cart của trang ProductDetails.aspx

Chuyển sang phần design



Rồi kéo xuống dưới. Có nút button: Add to Cart.



Click chuột đúp chuột vào button này. Thêm đoạn code sau vào

```
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;

public partial class ProductDetails : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        double Price = double.Parse(((Label)
            DataList1.Controls[0].FindControl("PriceLabel")).Text);
        string ProductName = ((Label)
            DataList1.Controls[0].FindControl("NameLabel")).Text;
        string ProductImageUrl = ((Label)
            DataList1.Controls[0].FindControl("ImageUrlLabel")).Text;
        int ProductID = int.Parse(Request.QueryString["ProductID"]);
        if (Profile.SCart == null)
        {
            Profile.SCart = new ShoppingCartExample.Cart();
        }
        Profile.SCart.Insert
            (ProductID, Price, 1, ProductName, ProductImageUrl);
        Server.Transfer("Products.aspx");
    }
}
```

```
protected void Button1_Click(object sender, EventArgs e)
{
    double Price = double.Parse(((Label)
        DataList1.Controls[0].FindControl("PriceLabel")).Text);

    string ProductName = ((Label)
        DataList1.Controls[0].FindControl("NameLabel")).Text;

    string ProductImageUrl = ((Label)
        DataList1.Controls[0].FindControl("ImageUrlLabel")).Text;

    int ProductID = int.Parse(Request.QueryString["ProductID"]);

    if (Profile.SCart == null)
```

```

{
    Profile.SCart = new Cart();

}

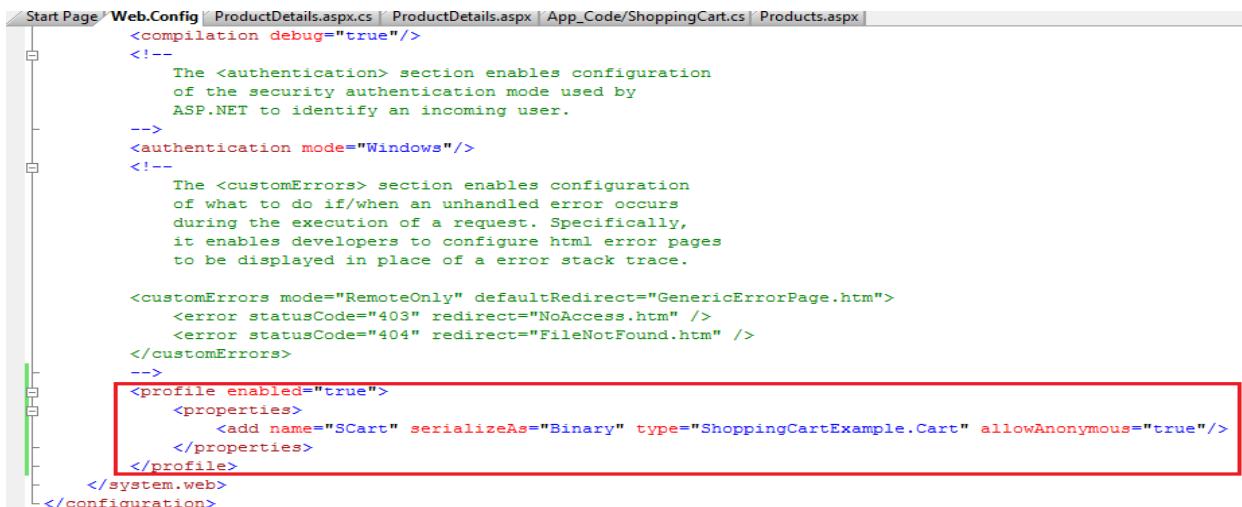
Profile.SCart.Insert
    (ProductID, Price, 1, ProductName, ProductImageUrl);

Server.Transfer("Products.aspx");

}

```

**Bước 13:** Mở file Web.config. Thêm đoạn code sau vào:

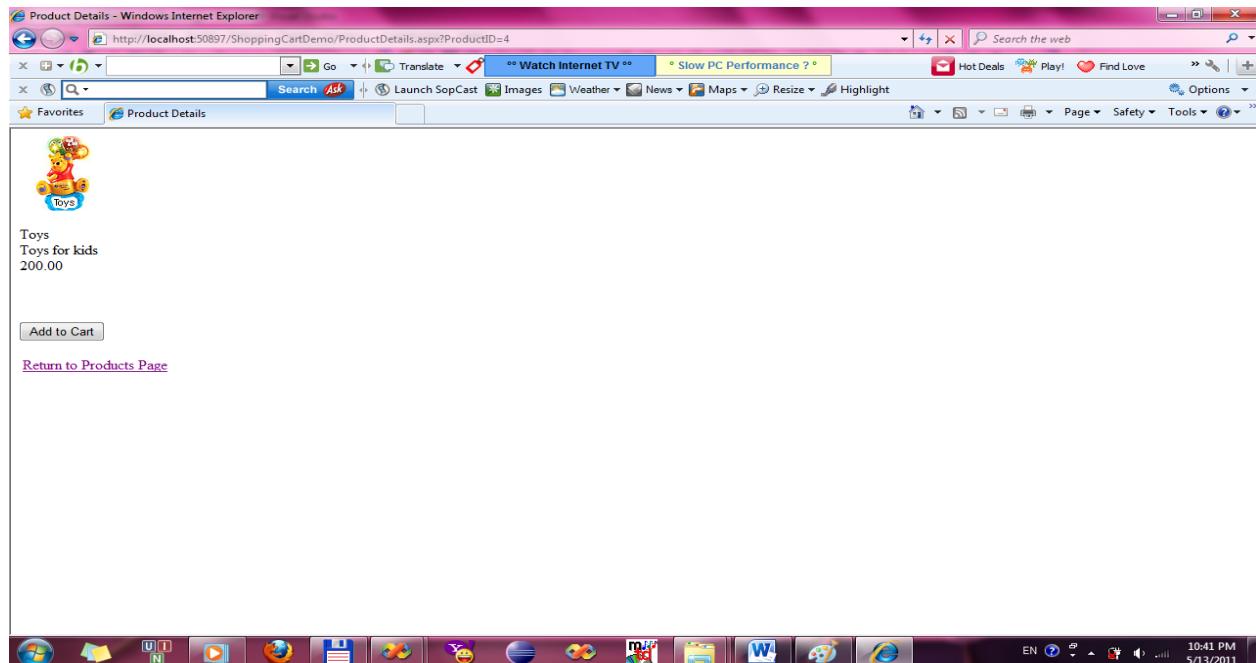


```

<profile enabled="true">
    <properties>
        <add name="SCart" serializeAs="Binary" type="Cart"
allowAnonymous="true"/>
    </properties>
</profile>

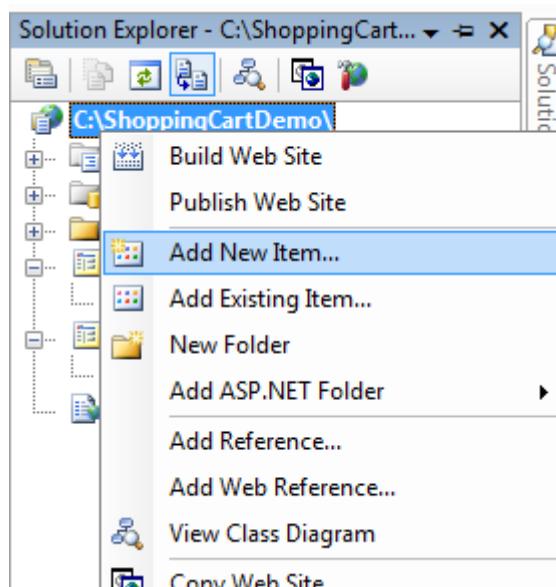
```

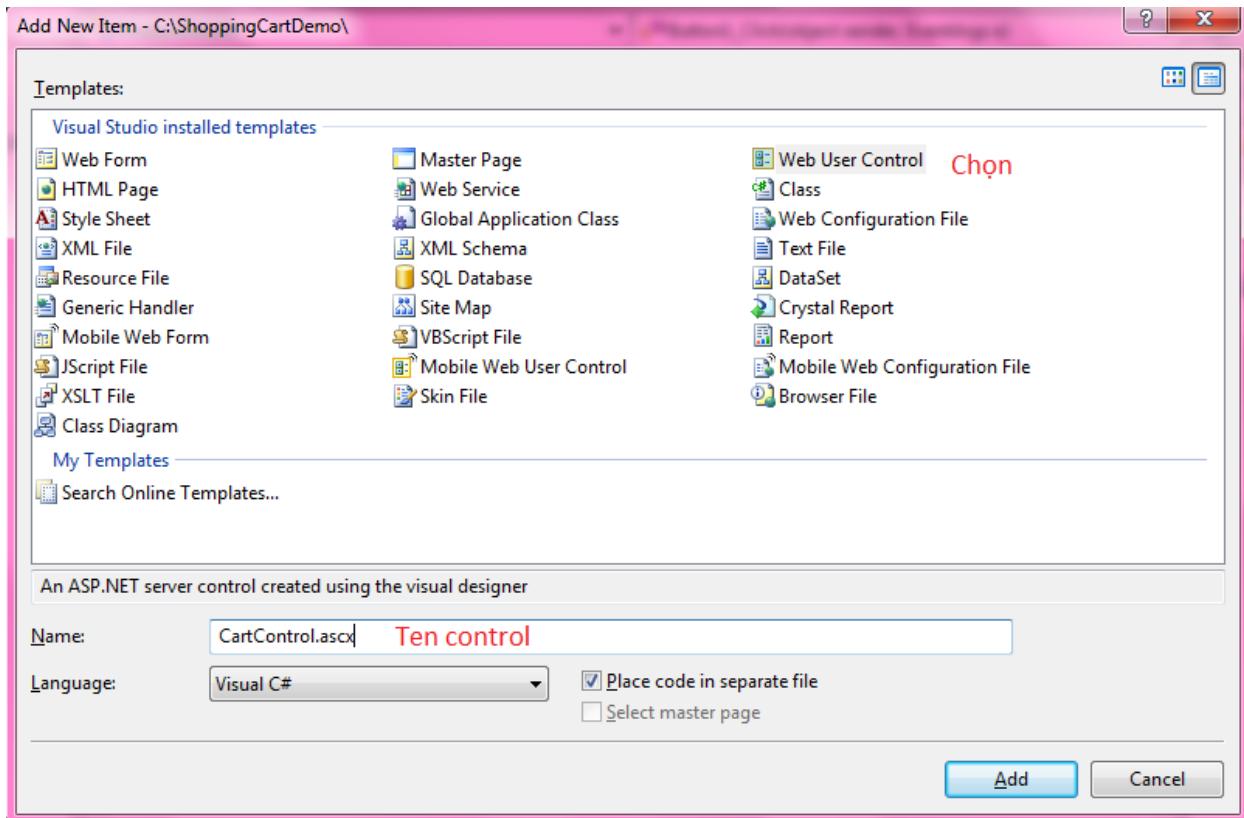
**Bước 14:** Chạy file Products.aspx rồi click vào một sản phẩm. Xem kết quả.



### **Bước 15:** Bây giờ thêm một web user control vào

Right click chuột vào ShoppingCartDemo → Add new Item.





Trong file source của nó thay đổi như sau

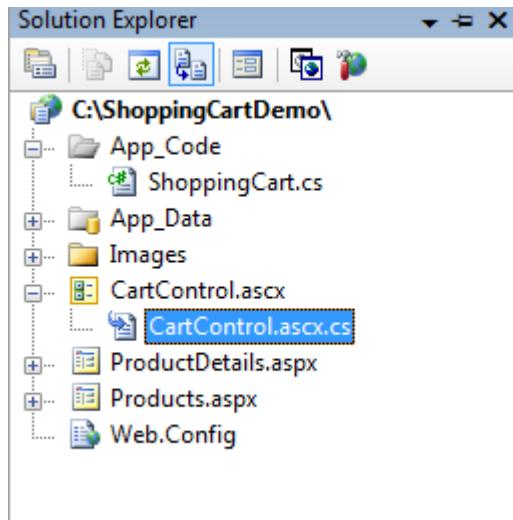
```
<%@ Control Language="C#" AutoEventWireup="true"
CodeFile="CartControl.ascx.cs" Inherits="CartControl" %>

<asp:GridView ID=" grdCart" runat="server"
    AutoGenerateColumns="False"
    DataKeyNames="ProductID"
    OnRowCancelingEdit=" grdCart_RowCancelingEdit"
    OnRowDeleting=" grdCart_RowDeleting"
    OnRowEditing=" grdCart_RowEditing"
    OnRowUpdating=" grdCart_RowUpdating">

<Columns>
    <asp:TemplateField>
        <ItemTemplate>
```

```
<asp:Image ID="Image1" runat="server"
    ImageUrl='<%#Eval("ImageUrl","~/Images/thumb_{0}")%>'>
</ItemTemplate>
</asp:TemplateField>
<asp:BoundField DataField="ProductName"
    HeaderText="Product" ReadOnly="True"/>
<asp:BoundField DataField="Quantity" HeaderText="Quantity"/>
<asp:BoundField DataField="Price" DataFormatString="{0:c}"
    HeaderText="Price" ReadOnly="True" />
<asp:BoundField DataField="SubTotal" DataFormatString="{0:c}"
    HeaderText="Total" ReadOnly="True" />
<asp:CommandField ShowDeleteButton="True"
    ShowEditButton="True"/>
</Columns>
<EmptyDataTemplate>
Your Shopping Cart is empty, add items
<a href="Products.aspx">Add Products</a>
</EmptyDataTemplate>
</asp:GridView>
<asp:Label ID="TotalLabel" runat="server"></asp:Label>
```

**Bước 16:** Mở file CartControl.ascx.cs



Và thay đổi như sau

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
public partial class CartControl : System.Web.UI.UserControl
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Profile.SCart == null)
    {
```

```
Profile.SCart = new Cart();

}

if (!Page.IsPostBack)

{

    ReBindGrid();

}

if (Profile.SCart.Items == null)

{

    TotalLabel.Visible = false;

}

}

protected void grdCart_RowEditing

    (object sender, GridViewEditEventArgs e)

{

    grdCart.EditIndex = e.NewEditIndex;

    ReBindGrid();

}

protected void grdCart_RowUpdating

    (object sender, GridViewUpdateEventArgs e)

{

    TextBox txtQuantity = (TextBox)

        grdCart.Rows[e.RowIndex].Cells[2].Controls[0];

    int Quantity = Convert.ToInt32(txtQuantity.Text);

    if (Quantity == 0)

{
```

```
    Profile.SCart.Items.RemoveAt(e.RowIndex);

}

else

{

    Profile.SCart.Items[e.RowIndex].Quantity

        = Quantity;

}

grdCart.EditIndex = -1;

ReBindGrid();

}

protected void grdCart_RowCancelingEdit

    (object sender, GridViewCancelEventArgs e)

{

    grdCart.EditIndex = -1;

    ReBindGrid();

}

protected void grdCart_RowDeleting

    (object sender, GridViewDeleteEventArgs e)

{

    Profile.SCart.Items.RemoveAt(e.RowIndex);

    ReBindGrid();

}

private void ReBindGrid()

{

    grdCart.DataSource = Profile.SCart.Items;
```

```
        DataBind();

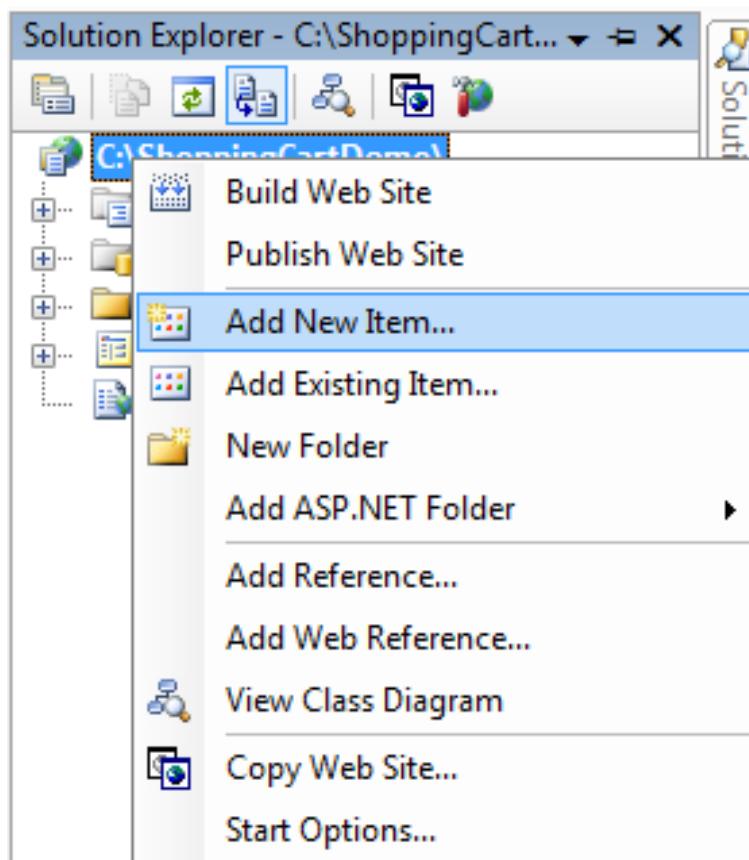
        TotalLabel.Text = string.Format("Total:{0,19:C}",
            Profile.SCart.Total);

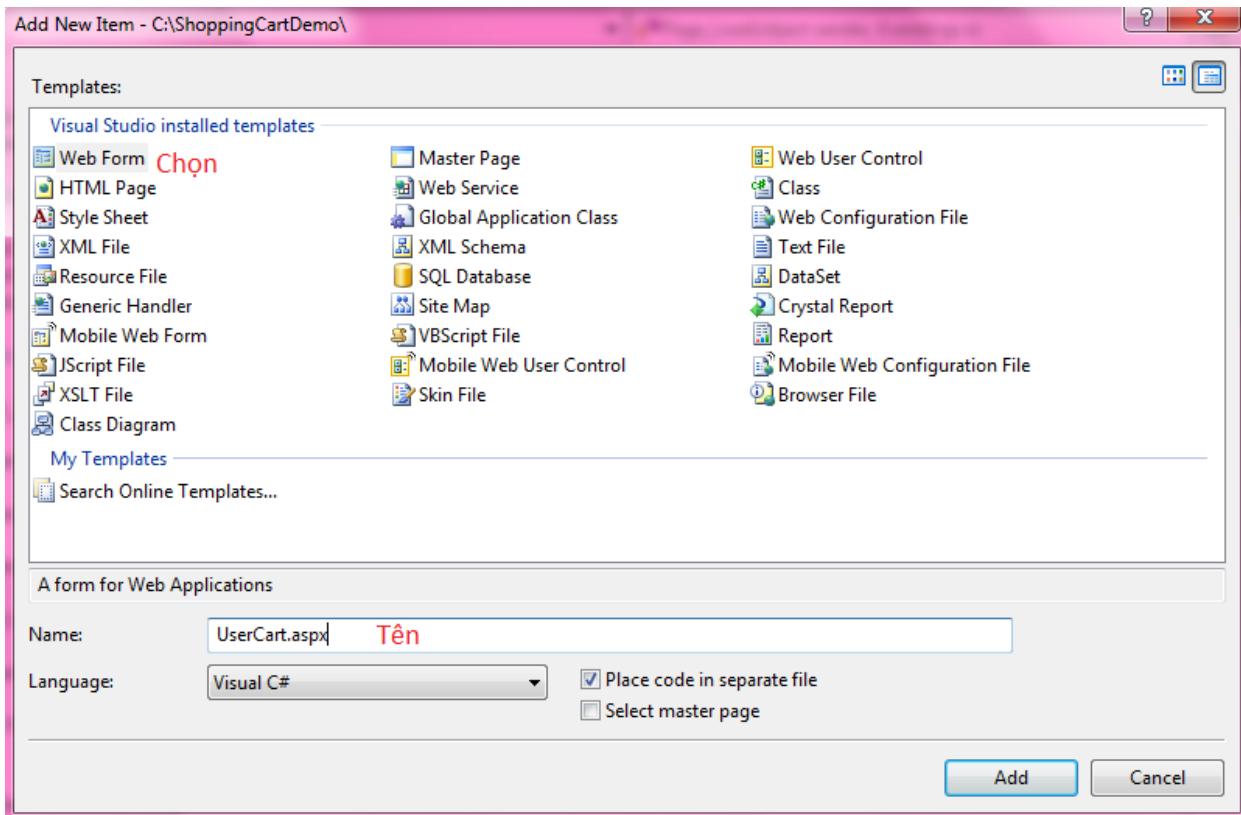
    }

}
```

**Bước 17:** Tạo ra trang UserCart page.

Click chuột phải vào C:\ShoppingCartDemo\ → Add New Item.





Trong file UserCart.aspx. bạn thay đổi code như sau

```

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="UserCart.aspx.cs"
Inherits="UserCart" %>

<%@ Register Src="CartControl.ascx" TagName="CartControl" TagPrefix="uc1"
%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >

<head id="Head1" runat="server">
    <title>Untitled Page</title>
</head>
<body>
    <form id="form1" runat="server">

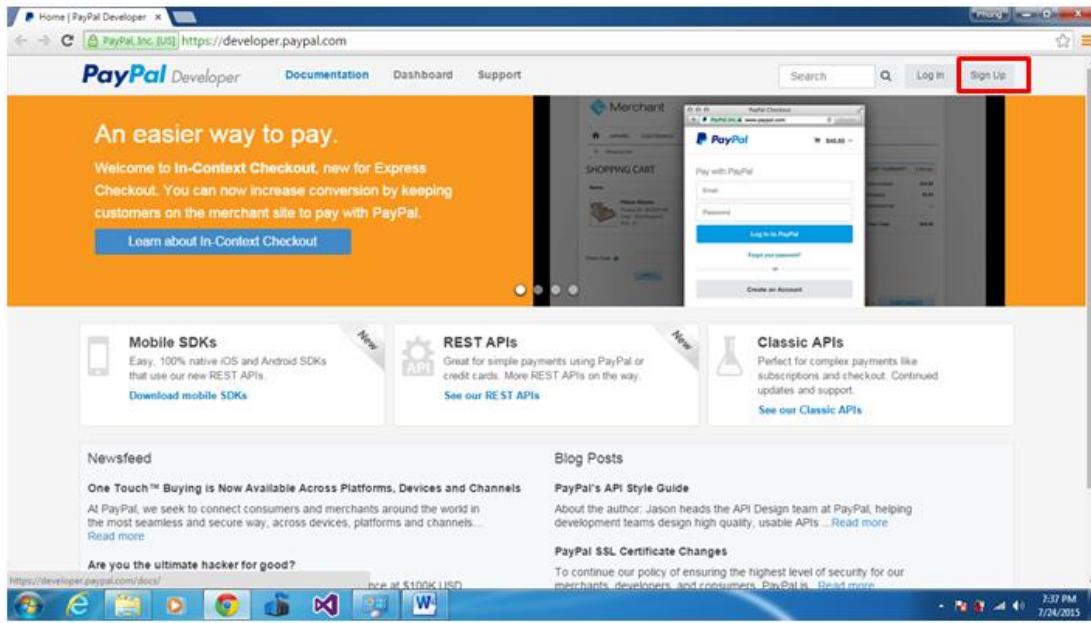
```

```
<div>
    <uc1:CartControl ID="CartControl1" runat="server" />
    <br />
    <br />
    <br />
    <asp:HyperLink ID="HyperLink1" runat="server"
        NavigateUrl("~/Products.aspx")>Return to Products Page</asp:HyperLink></div>
</form>
</body>
</html>
```

**Bước 18:** Chạy và thử nghiệm kết quả.

# HƯỚNG DẪN 2: TÍCH HỢP VÀ SỬ DỤNG PAYPAL

**Bước 1:** Vào trang <https://developer.paypal.com/>



**Bước 2:** Nhấn vào Sign Up Now. Sau đó điền đầy đủ các thông tin cần thiết

Sign up for a business account

Currently, we only offer live REST API credentials to those with a U.S. Business account. Non-U.S users can test our new REST APIs or use Classic APIs after creating an account on paypal.com.

Enter your email address and choose a password.

**Địa chỉ mail của bạn**

Mật khẩu phải ít nhất 8 ký tự và phải ít nhất 1 ký tự là số hoặc ký tự đặc biệt

Điền thông tin ảo cũng được

ZIP code: 10001

By clicking the button below, I agree to PayPal's User Agreement, Privacy Policy, Acceptable Use Policy, Electronic Communications Delivery Policy, PayPal Developer Agreement.

Agree and Continue

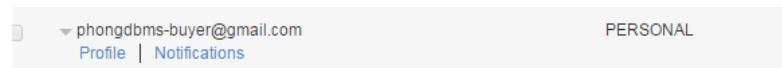
Rồi bấm Agree and Continue.

**Bước 3:** Bây giờ login vào trang <https://developer.paypal.com/> với account và password bạn đã đăng ký .

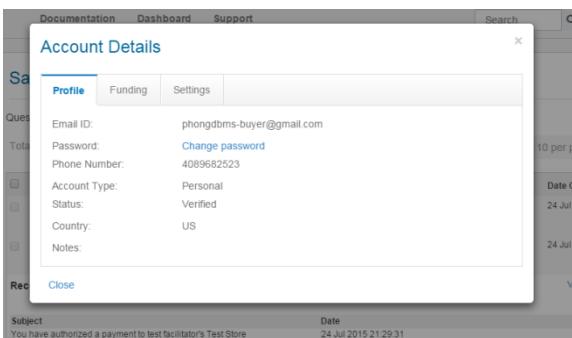
**Bước 4:** Tạo account cho buyer và merchant trên tài khoản của bạn , vào Dashboard rồi chọn tab account .Khi tạo tài khoản mặc định đã có account merchant và account buyer rồi chỉ việc đổi lại mật khẩu và lưu các thông tin để làm các bước sau.

Email Address	Type	Country	Date Created
phongdbms-facilitator@gmail.com	BUSINESS	US	24 Jul 2015
phongdbms-buyer@gmail.com	PERSONAL	US	24 Jul 2015

**Bước 5** Chọn tài khoản người mua hàng (account buyer) chỉnh sửa thông tin cần thiết để có thể đăng nhập và kiểm tra



Chọn profile



Chọn change password và đổi mật khẩu. Có thể xem thông tin về tài khoản ở các thẻ còn lại

**Account Details**

Profile	Funding	Settings
<b>Bank Account</b>		
Account Number:	4463206	
Routing Number:	325272063	
<b>Credit Card</b>		
Credit Card Number:	4032030434342367	
Credit Card Type:	VISA	
Expiration Date:	08/2020	
<b>PayPal</b>		
Balance:	9999.00 USD	

[Close](#)

**Bước 6** Chọn tài khoản người người bán hàng (account merchant) chỉnh sửa thông tin cần thiết để có thể đăng nhập và kiểm tra

Email Address	Type
phongdbms-facilitator@gmail.com	BUSINESS

[Profile](#) | [Notifications](#)

Chọn profile

[Documentation](#) [Dashboard](#) [Support](#) [Search](#)

**Account Details**

Profile	API Credentials	Funding	Settings
<b>Classic TEST API credentials</b>			
Username:	phongdbms-facilitator_api1@gmail.com		
Password:	K97VNBF43EBDVFZ <b>Thông tin cần ghi nhớ cho bước sau</b>		
Signature:	A-T8IQvi-5JjaG7wA8kr4goinFU1A-Bf.jFdZ-rCJoT7yshGG9R0Q8rk		

[Close](#)

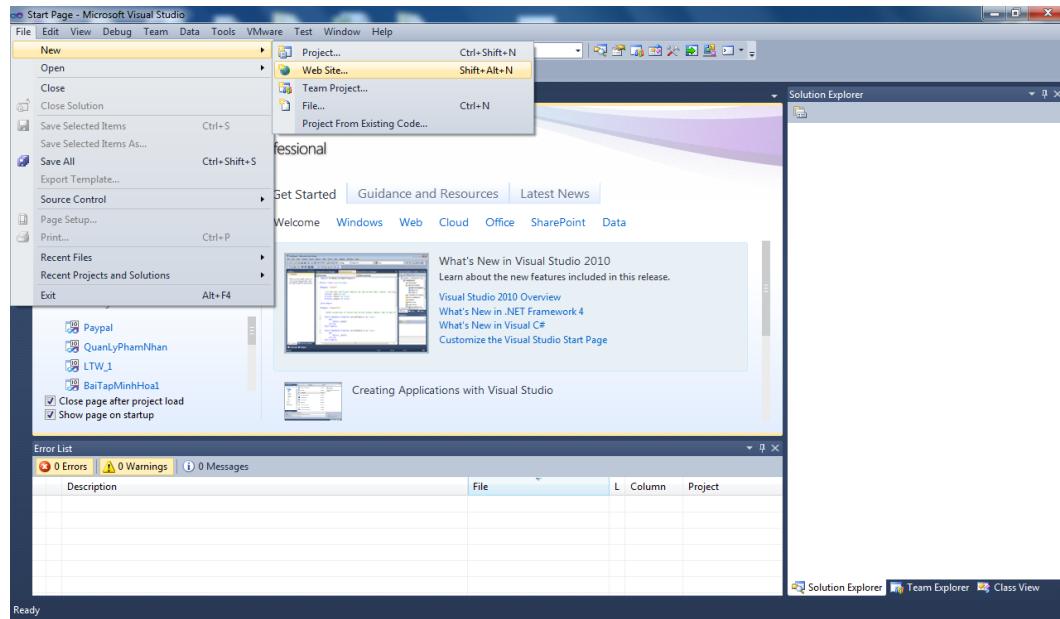
[phongdbms-buyer@gmail.com](#) [PERSONAL](#) [US](#)

[Profile](#) | [Notifications](#)

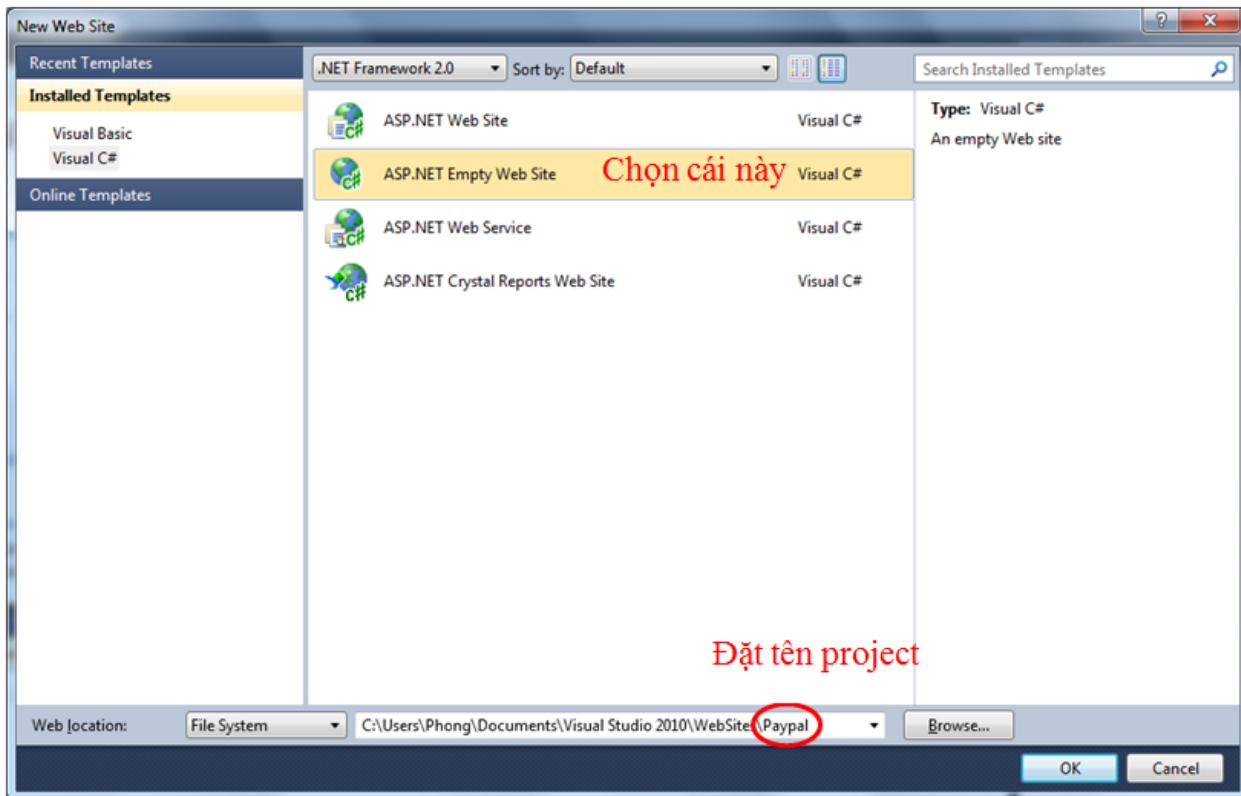
## Xây dựng trang web có Thanh toán trực tuyến qua paypal

**Bước 1:** Đầu tiên mở chương trình Microsoft Visual Studio lên.

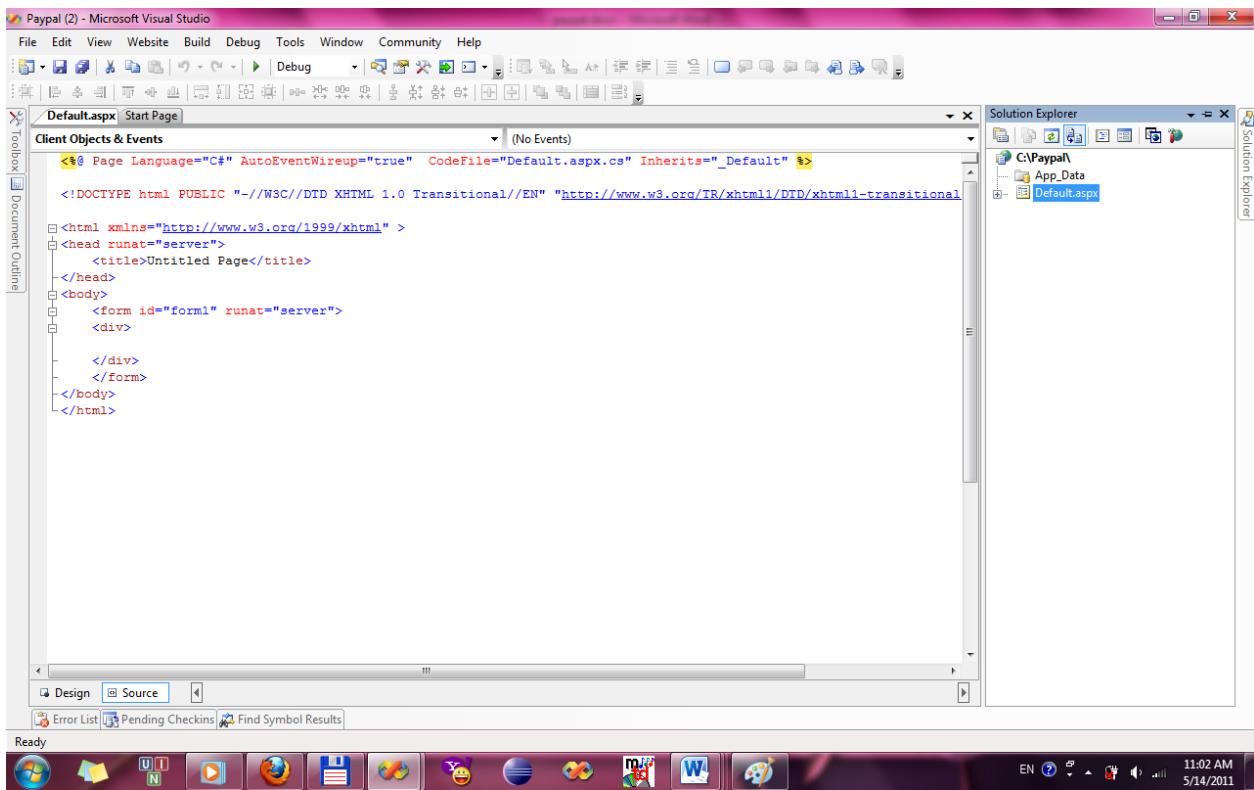
**Bước 2:** Tạo một project dạng website File → New → Web Site.



Rồi làm như sau

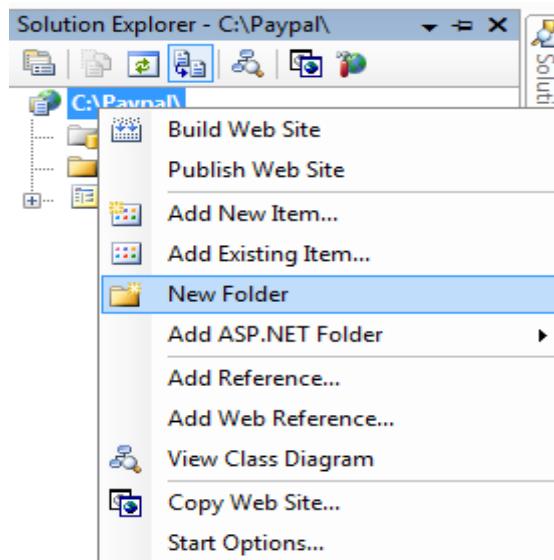


Sau khi bấm OK. Ta được project như sau

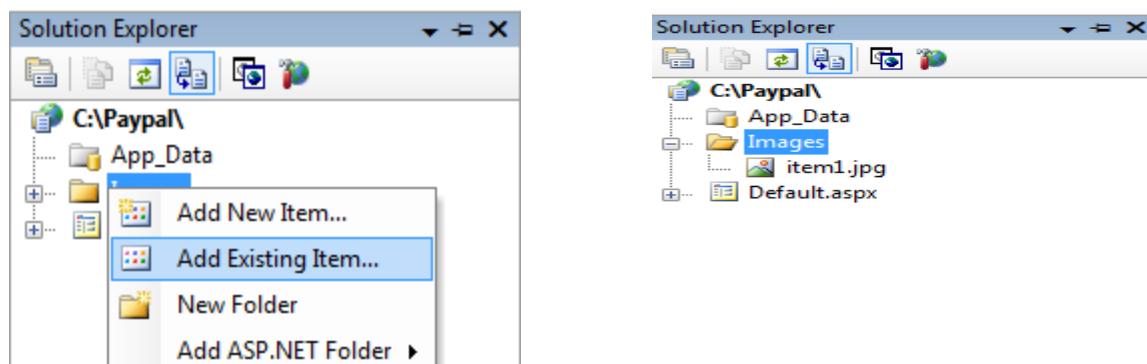


**Bước 3:** Tạo folder chứa các hình ảnh của sản phẩm.

Click chuột phải vào C:\Paypal\ → New Folder. Rồi đặt tên là Images. Rồi chép các hình ảnh sản phẩm vào thư mục này.

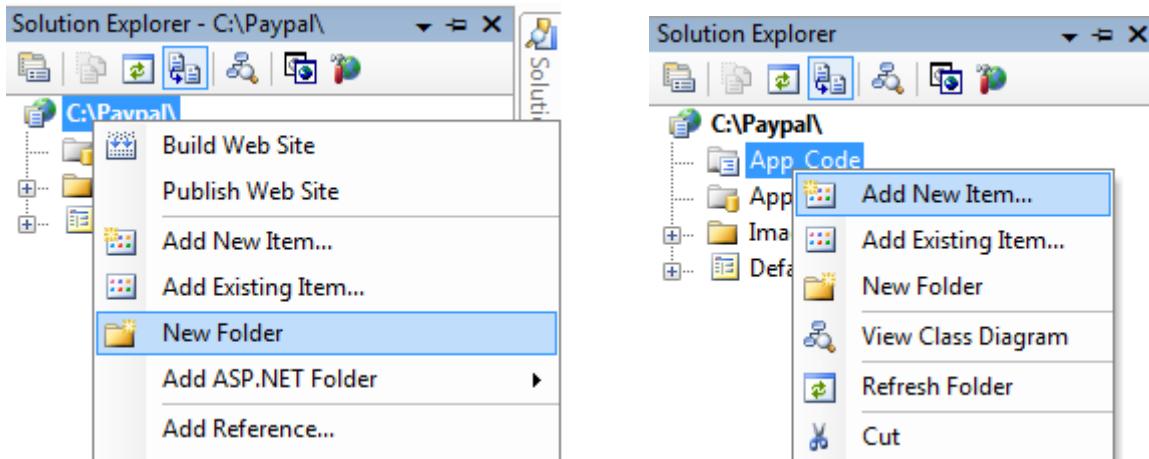


Click chuột phải vào Images → Add Existing Item. Rồi chọn hết hình ảnh sản phẩm ở trên vào.

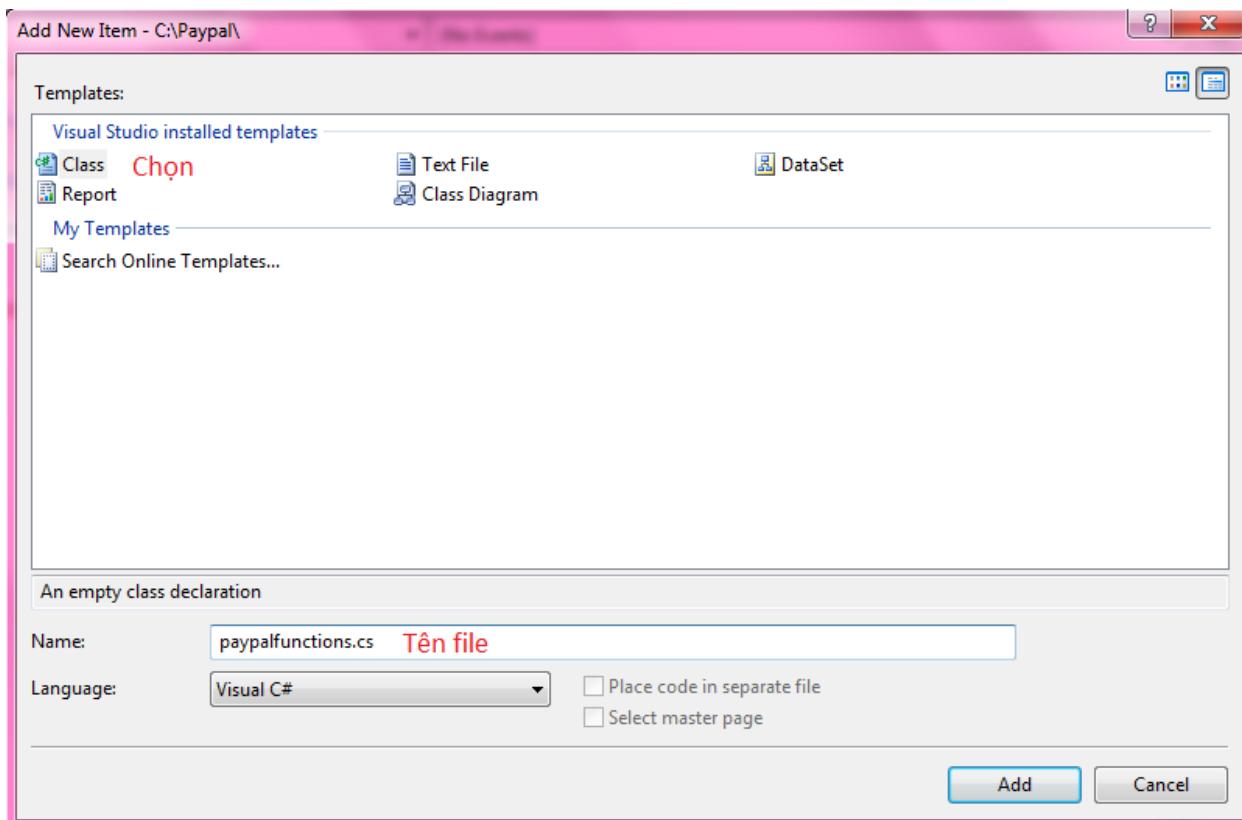


**Bước 4:** Tạo folder chứa code dạng cs.

Click chuột phải vào C:\Paypal\ → New Folder. Rồi đặt tên là App\_Code. Rồi click chuột phải vào folder App\_Code. Chọn Add New Item....



Rồi chọn như sau:



**Bước 5:** Bạn chép đè file paypalfunction.cs lên file paypalfunction.cs trong project của bạn.

Trong file paypalfunction.cs đó bạn tìm nơi có nội dung sau

```
//Replace <API_USERNAME> with your API Username  
  
//Replace <API_PASSWORD> with your API Password  
  
//Replace <API_SIGNATURE> with your Signature  
  
public string APIUsername = "seller_1253264174_biz_api1.gmail.com";  
  
private string APIPassword = "1253264187";  
  
private string APISignature =  
"AiPC9BjkCyDFQXbSkoZcgqH3hpacA68FSy0cFjJYJyay53u6YEI15ImN";
```

Bạn thay đổi lại nội dung thành của bạn (Phần API Credentials trên paypal của bạn ở trên)

```
public string APIUsername = "phongdbms-facilitator_api1.gmail.com";  
  
private string APIPassword = "K97VNBXF43EBDVFZ";  
  
private string APISignature = "A-T8IQvi-5JjaG7wA8kr4goinFU1A-Bf.jFdZ-  
rCJoT7yshGG9R0Q8rk"
```

Trong file paypalfunction.cs đó bạn lại tìm nơi có nội dung sau:

```
string returnUrl = "http://localhost:2545/Paypal/BillingInformation.aspx";  
  
string cancelURL = "http://localhost:2545/Paypal/ReturnPage.aspx";
```

Thay đổi chữ Paypal thành tên project của bạn.

```
string returnUrl = "http://localhost:2545/Hung/BillingInformation.aspx";  
  
string cancelURL = "http://localhost:2545/Hung/ReturnPage.aspx";
```

Và bạn chú ý, port ở đây là 2545, và 2 trang BillingInformation.aspx và ReturnPage.aspx mà paypal sẽ liên kết với ứng dụng/web của bạn thông qua port này và dùng 2 trang này. Quá trình change port và tạo 2 trang BillingInformation.aspx và ReturnPage.aspx sẽ hướng dẫn ở sau.

**Bước 6:** Tạo ra trang default page.

Tại trang Default.aspx, trong phần thẻ body bạn thay đổi code lại như sau

```
<div>

    <h1> Your Shopping Cart.</h1>

</div>

<div>

    <table border="0" cellpadding="0" cellspacing="0" style="font-family: Arial; font-size: 12px">

        <tr>

            <td colspan="2"><span>Items</span></td>

            <td ><span>Qty</span></td>

            <td ><span>Ship</span></td>

            <td><span >Price</span></td>

            <td><span >Total</span></td>

        </tr>

        <tr>

            <td></td>

            <td>MacBook Pro MC118ZP/A</td>

            <td><input type="text" value="1" size="2" />

                <a href="javascript:void(0)">Remove</a>

            </td>

            <td>Usually ships within 1 to 2 days.</td>

            <td>$1.200</td>

            <td>$1.200</td>

        </tr>

        <tr>

            <td colspan="6" style="text-align: right; border-bottom:dashed 1px #CCCCCC;">&nbsp;</td>

        </tr>

    </table>

</div>
```

```
</tr>

<tr>

    <td colspan="5" style="text-align: right">Items Total:</td>
    <td>$1.200</td>
</tr>

<tr>

    <td colspan="5" style="text-align: right">Shipping:</td>
    <td>$220</td>
</tr>

</table>

</div>

<div style="text-align:left; padding-top:20px;">

    <form      action="expresscheckout.aspx"      method="post"      id="form1"
runat="server">

        <input          type='image'          name='submit'
src='https://www.paypal.com/en_US/i/btn/btn_xpressCheckout.gif'
border='0' align='top' alt='PayPal' style="text-align: right" />
    </form>
</div>
```

Đây là nội trong giỏ hàng của bạn, nó chỉ có 1 cái laptop thôi ☺.

Trong hàm Page\_Load của file default.aspx.cs thay đổi như sau:

```
protected void Page_Load(object sender, EventArgs e)
{
    Session["payment_amt"] = "500.00";
}
```

**Bước 7:** Tạo ra trang expresscheckout.aspx page.

Tại trang expresscheckout.aspx bạn thay đổi code như sau

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="expresscheckout.aspx.cs" Inherits="PayPalEC" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title></title>
</head>
<body>
</body>
</html>
```

Trong file expresscheckout.aspx.cs thay đổi như sau

```
using System;
using System.Web;
using System.Web.UI;
public partial class PayPalEC : System.Web.UI.Page
{
    protected void ShowMessage(string msg)
    {
        ClientScriptManager cs = Page.ClientScript;
        if (!cs.IsStartupScriptRegistered(this.GetType(), "Message"))
        {
            string scriptBlock = "alert(\"" + msg + "\");";
            cs.RegisterStartupScript(this.GetType(), "Message", scriptBlock);
        }
    }
}
```

```
        cs.RegisterStartupScript(this.GetType(), "Message", scriptBlock, true);

    }

}

protected void Page_Load(object sender, EventArgs e)

{

    NVPAPICaller test = new NVPAPICaller();

    string retMsg = "";

    string token = "";

    if (Session["payment_amt"] != null)

    {

        ShowMessage("not null");

        string amt = Session["payment_amt"].ToString();

        bool ret = test.ShortcutExpressCheckout(amt, ref token, ref retMsg);

        if (ret)

        {

            Session["token"] = token;

            Response.Redirect(retMsg);

        }

        else

        {

            Response.Redirect("APIError.aspx?" + retMsg);

        }

    }

    else

    {
```

```
    ShowMessage("null");

    Response.Redirect("APIError.aspx?ErrorCode=AmtMissing");

}

}

}
```

**Bước 8:** Tạo ra trang BillingInformation.aspx page.

Tại trang BillingInformation.aspx, trong phần body bạn thêm đoạn code sau

```
<form id="form1" runat="server" >

<div>

    <h1>Order Review.</h1>

</div>

<div id="div_Shipping" runat="server">

</div>

<div>

    <table border="0" cellpadding="0" cellspacing="0" style="font-family: Arial;
font-size: 12px">

        <tr>

            <td colspan="2"><span >Items</span> </td>

            <td><span >Qty</span></td>

            <td><span >Ship</span></td>

            <td><span >Price</span></td>

            <td><span >Total</span></td>

        </tr>

        <tr>

            <td></td>
```

```
<td>MacBook Pro MC118ZP/A</td>

<td><input type="text" value="1" size="2" />

    <a href="javascript:void(0)">Remove</a></td>

<td>Usually ships within 1 to 2 days.</td>

<td>$1.200</td>

<td>$1.200</td>

</tr>

<tr>

    <td colspan="6" style="text-align: right; border-bottom: dashed 1px #CCCCCC;"> &ampnbsp</td>

</tr>

<tr>

    <td colspan="5" style="text-align: right">Items Total:</td>

    <td>$1.200</td>

</tr>

<tr>

    <td colspan="5" style="text-align: right">Shipping:</td>

    <td>$220</td>

</tr>

</table>

</div>

<div>

    <asp:Button ID="btn_ConfirmOrder" runat="server" Text="Confirm Order" OnClick="btn_ConfirmOrder_Click" />

</div>
```

```
<div id="div_Transaction" runat="server">  
</div>  
</form >
```

Trong file BillingInformation.aspx.cs bạn thêm các thông tin sau

Hàm ShowMessage:

```
protected void ShowMessage(string msg)  
{  
    ClientScriptManager cs = Page.ClientScript;  
    if (!cs.IsStartupScriptRegistered(this.GetType(), "Message"))  
    {  
        string scriptBlock = "alert(\"" + msg + "\");";  
        cs.RegisterStartupScript(this.GetType(), "Message", scriptBlock, true);  
    }  
}
```

Hàm Page\_Load thay đổi nội dung như sau

```
protected void Page_Load(object sender, EventArgs e)  
{  
    NVPAPICaller test = new NVPAPICaller();  
    string retMsg = "";  
    string token = "";  
    string payerId = "";  
    string shippingAddress = "";  
    token = Session["token"].ToString();  
    bool ret = test.GetShippingDetails(token, ref payerId, ref shippingAddress, ref  
    retMsg);
```

```
if (ret)
{
    Session["payerId"] = Request.QueryString["PayerID"];
    div_Shipping.InnerHtml = shippingAddress;
}
else
{
    Response.Redirect("APIError.aspx?" + retMsg);
}
}
```

Thêm hàm xử lý sự kiện button ConfirmOrder trên trang web

```
protected void btn_ConfirmOrder_Click(object sender, EventArgs e)
{
    NVPAPICaller test = new NVPAPICaller();
    string retMsg = "";
    string token = "";
    string finalPaymentAmount = "";
    string payerId = "";
    NVPCodec decoder = new NVPCodec();
    token = Session["token"].ToString();
    payerId = Session["payerId"].ToString();
    finalPaymentAmount = Session["payment_amt"].ToString();
    bool ret = test.ConfirmPayment(finalPaymentAmount, token, payerId, ref
decoder, ref retMsg);
    if (ret)
```

{

// Unique transaction ID of the payment. Note: If the PaymentAction of the request was Authorization or Order, this value is your AuthorizationID for use with the Authorization & Capture APIs.

```
string transactionId = decoder["TRANSACTIONID"];
```

// The type of transaction Possible values: I cart I express-checkout

```
string transactionType = decoder["TRANSACTIONTYPE"];
```

// Indicates whether the payment is instant or delayed. Possible values: I none I echeck I instant

```
string paymentType = decoder["PAYMENTTYPE"];
```

// Time/date stamp of payment

```
string orderTime = decoder["ORDERTIME"];
```

// The final amount charged, including any shipping and taxes from your Merchant Profile.

```
string amt = decoder["AMT"];
```

// A three-character currency code for one of the currencies listed in PayPay-Supported Transactional Currencies. Default: USD.

```
string currencyCode = decoder["CURRENCYCODE"];
```

// PayPal fee amount charged for the transaction

```
string feeAmt = decoder["FEEAMT"];
```

// Amount deposited in your PayPal account after a currency conversion.

```
string settleAmt = decoder["SETTLEAMT"];
```

// Tax charged on the transaction.

```
string taxAmt = decoder["TAXAMT"];
```

// Exchange rate if a currency conversion occurred. Relevant only if you are billing in their non-primary currency. If

```

        string exchangeRate = decoder["EXCHANGERATE"];

        string s_Transaction = string.Empty;

        s_Transaction += "transaction ID: " + transactionId + "<br />";

        s_Transaction += "transaction Type: " + transactionType + "<br />";

        s_Transaction += "payment Type: " + paymentType + "<br />";

        s_Transaction += "order Time: " + orderTime + "<br />";

        s_Transaction += "amount: " + amt + "<br />";

        s_Transaction += "currency Code: " + currencyCode + "<br />";

        s_Transaction += "fee Amt: " + feeAmt + "<br />";

        s_Transaction += "exchange Rate: " + exchangeRate;

        div_Transaction.InnerHtml = s_Transaction;

    }

    else

    {

        Response.Redirect("APIError.aspx?" + retMsg);

    }

}

```

**Bước 9:** Tạo ra trang APIError.aspx page.

Tại trang APIError.aspx bạn thay đổi code như sau

```

<%@ Page Language="c#" AutoEventWireup="false" CodeFile="APIError.aspx.cs"
Inherits="APIError" %>

<html>

    <head>

        <title>PayPal - Error Page</title>

    </head>

```

```
<body>

    <table class="api" id="Table1">

        <tr>

            <td class="field"></td>

            <td><%=Request.QueryString.Get("ErrorCode")%></td>

        </tr>

        <tr>

            <td class="field"></td>

            <td><%=Request.QueryString.Get("Desc")%></td>

        </tr>

        <tr>

            <td class="field"></td>

            <td><%=Request.QueryString.Get("Desc2")%></td>

        </tr>

    </table>

</body>

</html>
```

**Bước 10:** Tạo ra trang ReturnPage.aspx page.

Tại trang ReturnPage.aspx, trong phần body bạn thay đổi code như sau

```
<body>

    <form id="form1" runat="server">

        <div>

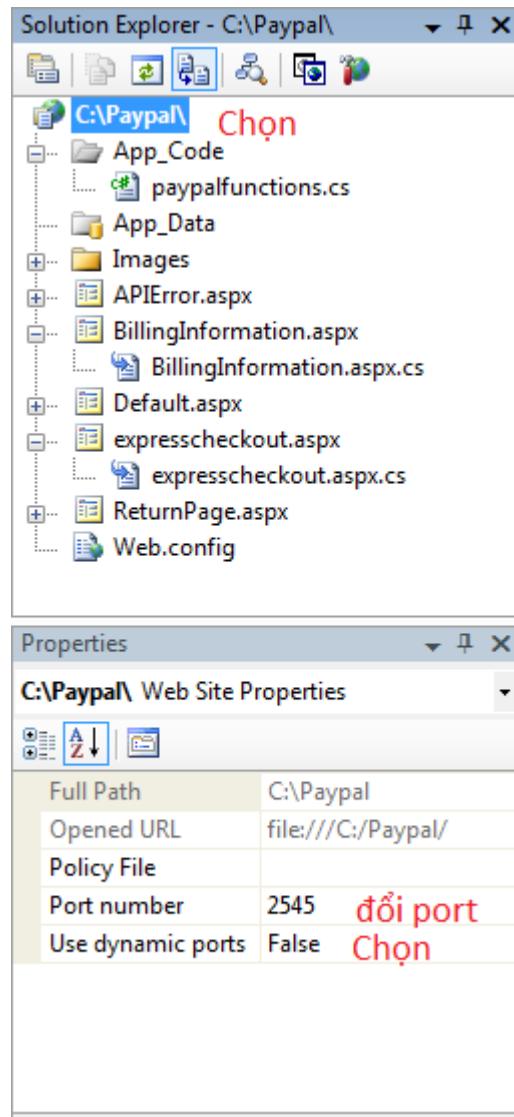
            Nothing.

        </div>

    </form>
```

```
</body>
</html>
```

**Bước 11:** Đổi port chạy project sang port 2545



Xong rồi đó chạy và test.

### Chạy và test Express Checkout

**Bước 1:** Chú ý bạn phải login vào trang paypal với tài khoản thật trước tức là tài khoản ở **bước 2** ở trên lúc đăng ký.

**Bước 2:** Sau đó, Chạy trang default.aspx.

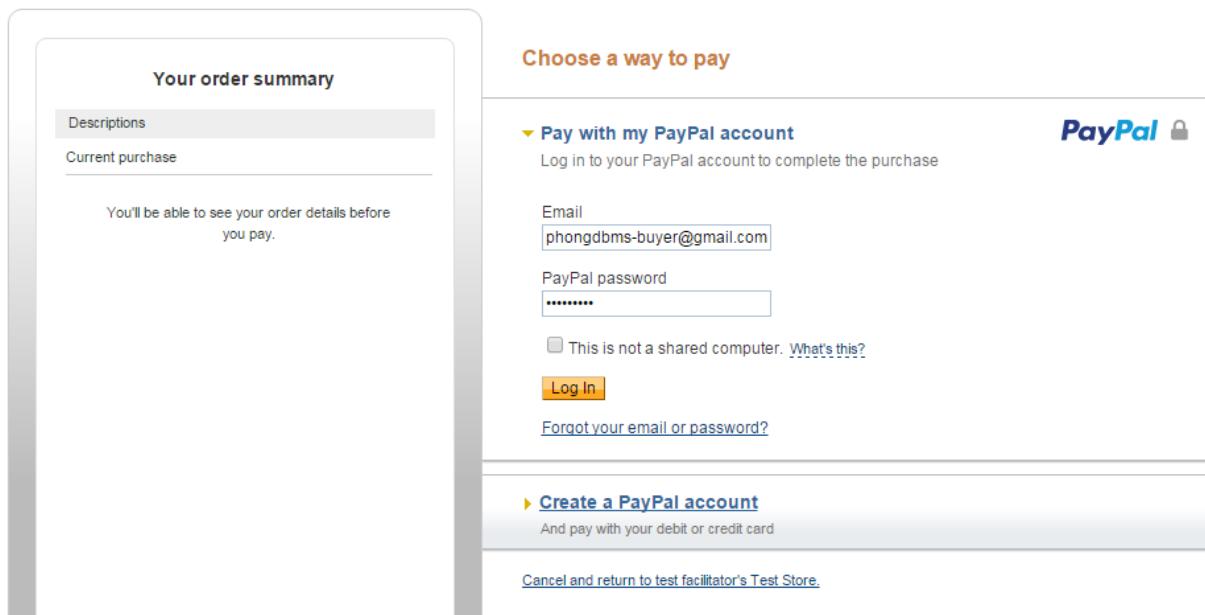
## Your Shopping Cart.

Items	Qty	Ship	Price	Total
	1		\$1.200	\$1.200
MacBook Pro MC118ZP/A 1 <a href="#">Remove</a>				Usually ships within 1 to 2 days.
				Items Total:\$1.200 Shipping:\$220

**PayPal Check out**  
The safer, easier way to pay

**Bước 3:** Click button “**Check out with PayPal**”. Sau đó hiện ra trang login. Bạn login vào với tài account buyer và password mà bạn đã đặt ban đầu

test facilitator's Test Store



Your order summary

Descriptions  
Current purchase

You'll be able to see your order details before you pay.

Choose a way to pay

▼ Pay with my PayPal account  Log in to your PayPal account to complete the purchase

Email: phongdbms-buyer@gmail.com

PayPal password:   This is not a shared computer. [What's this?](#)

**Log In** [Forgot your email or password?](#)

► Create a PayPal account  
And pay with your debit or credit card

[Cancel and return to test facilitator's Test Store.](#)

Ở đây, login với user phongdbms-buyer@gmail.com

Chọn Log In để được hình sau

### test facilitator's Test Store

**Your order summary**

Descriptions  
Current purchase

You'll be able to see your order details before you pay.

**PayPal Electronic Communications Delivery Policy Consent**

This will just take a minute and then you can complete your transaction.

We've updated our Electronic Communications Delivery Policy. Please read it and consent so we can send your account information electronically, including your payment confirmation.

Please confirm that:

Yes, I've read and agree to the [Electronic Communications Delivery Policy](#). I understand that PayPal will provide me with information about my account electronically. I confirm that I can access emails, web pages, and PDF files.

**Agree and Continue**

[Cancel and return to test facilitator's Test Store.](#)

Check yes và chọn Agree and Continue được hình sau

### test facilitator's Test Store

**Your order summary**

Descriptions  
Current purchase

You'll be able to see your order details before you pay.

**Review your information**

**Shipping address** [Change](#)

test buyer  
1 Main St  
San Jose, CA 95131  
United States

**Payment methods** [Change](#)

PayPal Balance

If funds are added to your PayPal balance before this transaction completes, the additional balance may be used for this transaction. [Learn more](#)

[View PayPal policies](#) and your payment method rights.

**Contact information**  
phongdbms-buyer@gmail.com

**Continue**

Click **Continue** để đến trang **Order Review** (trang do mình cấu hình – không phải Paypal ).

**Bước 4:** Bấm confirm order**Order Review.**

First Name test

Last Name buyer

Shipping Address

Name test buyer

Street1 1 Main St

Street2

City San Jose

State CA

Zip 95131

Items	Qty	Ship	Price	Total
-------	-----	------	-------	-------



MacBook Pro MC118ZP/A 1

[Remove](#)

Usually ships within 1 to 2 days.

\$1.200

\$1.200

Items Total:\$1.200

Shipping:\$220

[Confirm Order](#)

transaction ID: 05151078MM5471519

transaction Type: expresscheckout

payment Type: instant

order Time: 2015-07-24T14:29:19Z

amount: 500.00

currency Code: USD

fee Amt:

exchange Rate:

**Bước 6:** Bạn vào [https://www.sandbox.paypal.com/vn/cgi-bin/webscr?cmd=\\_login-run](https://www.sandbox.paypal.com/vn/cgi-bin/webscr?cmd=_login-run) login vào với account merchant bạn sẽ thấy thông tin về transaction như sau:

The screenshot shows the PayPal Sandbox Test Site interface. At the top, there's a navigation bar with links for My Account, Send Money, Request Money, Merchant Services, and Products & Services. Below the navigation is a sub-menu with Overview, Add Money, Withdraw, History, Resolution Center, and Profile.

The main content area displays a welcome message: "Welcome, test facilitator". It shows basic account information: Business Name: test facilitator's Test Store | Account Type: Business | Status: Verified | Account Limits: View Limits. Below this, it shows a balance of \$0.00 USD.

The "My recent activity" section lists transactions from Jul 17, 2015 to Jul 24, 2015. One transaction is shown in detail:

Date	Type	Name/Email	Payment status	Details	Order status/Actions	Gross
Jul 24, 2015	Authorization From	test buyer	Pending	<a href="#">Details</a>	<a href="#">Capture</a>	\$500.00 USD

A red circle highlights the "Capture" button in the "Order status/Actions" column. To the right of the main content, there are two sidebar boxes: "Notifications" and "My account tools".

Do chọn hình thức là Authorization nên số tiền thanh toán sẽ được giữ lại (tối đa là 3 ngày, nếu quá thì transaction sẽ bị hủy) chỉ đến khi nào nhấn button Capture thì khi đó mới hoàn tất quá trình thanh toán. (Nếu chọn hình thức là Sale thì khi đó sẽ thanh toán trực tiếp luôn mà không cần giữ lại.)

**Bước 7:** Bạn nhấn vào capture. Rồi điền thông tin vào

**P PayPal**

My Account Send Money Request Money Merchant Services Products & Services

Overview Add Money Withdraw History Resolution Center Profile

### Capture Funds

Enter the capture details below to move funds from your buyer's account to your account. To capture multiple transactions at once, use our [batch capture](#) tool.

Authorization ID: 05151078MM5471519  
 Name: test buyer  
 Email: phongdbms-buyer@gmail.com  
 Authorization Amount: \$500.00 USD  
 Capture Amount: **500** 500

Additional Capture Possible:  Yes (option to capture additional funds on this authorization if needed)  
 No (no additional capture needed; close authorization after this capture)

**Note to merchant:** Your buyer test buyer authorized a payment of \$500.00 USD. If you capture an amount different than this, you should notify your buyer of this change.

PayPal strongly recommends that you explain any unique circumstances (e.g. multiple captures, changes in item availability) to your buyer in detail below. Your buyer will see this note in the Transaction Details.

**Note to Buyer:**  
 (Optional)

1000 characters left

**P PayPal**

My Account Send Money Request Money Merchant Services Products & Services

Overview Add Money Withdraw History Resolution Center Profile

### Confirm Capture of Funds

Please verify that the information displayed below is correct and click **Capture Funds**. The funds will then be transferred to your account.

Authorization ID: 05151078MM5471519  
 Name: test buyer  
 Email: phongdbms-buyer@gmail.com  
 Authorization Amount: \$500.00 USD  
 Capture Amount: \$500.00 USD  
 Additional Capture Possible: No  
 Note to Buyer:

**Capture Funds** Edit Cancel

Bấm Continue. Rồi bấm Capture Funds

PayPal™ Sandbox

Test Site

My Account | Send Money | Request Money | Merchant Services | Products & Services

Overview Add Money Withdraw History Resolution Center Profile

**Confirm Capture of Funds**

Please verify that the information displayed below is correct and click **Capture Funds**. The funds will then be transferred to your account.

Authorization ID:	23223936UT614261S
Name:	buyeraccount buyeraccount
Email:	buyeraccount01@yahoo.com
Authorization Amount:	\$500.00 USD
Capture Amount:	\$500.00 USD
Additional Capture Possible:	No
Note to Buyer:	

**Capture Funds** | Edit | Cancel

### Quay lại trang overview

My Account | Send Money | Request Money | Merchant Services | Products & Services

Overview Add Money Withdraw History Resolution Center Profile

Welcome, test facilitator

Business Name: test facilitator's Test Store | Account Type: Business | Status: **Verified** | Account Limits: [View Limits](#)

PayPal balance: **\$485.20 USD** Tiền tăng từ 0->485.20

My recent activity | [Payments received](#) | [Payments sent](#) [View all of my transactions](#)

My recent activity - Last 7 days (Jul 17, 2015-Jul 24, 2015)

Archive	What's this	Date	Type	Name/Email	Payment status	Details	Order status/Actions	Gross
<input type="checkbox"/>	<input checked="" type="checkbox"/> Jul 24, 2015	Payment From	test buyer	Completed	<a href="#">Details</a>	<a href="#">Issue refund</a>	\$500.00 USD	

Notifications

- [Add a mobile phone number](#)
- [Add a home phone number](#)
- [Policy Updates](#)

My account tools

- [My Business Setup](#)
- [Manage Users](#)

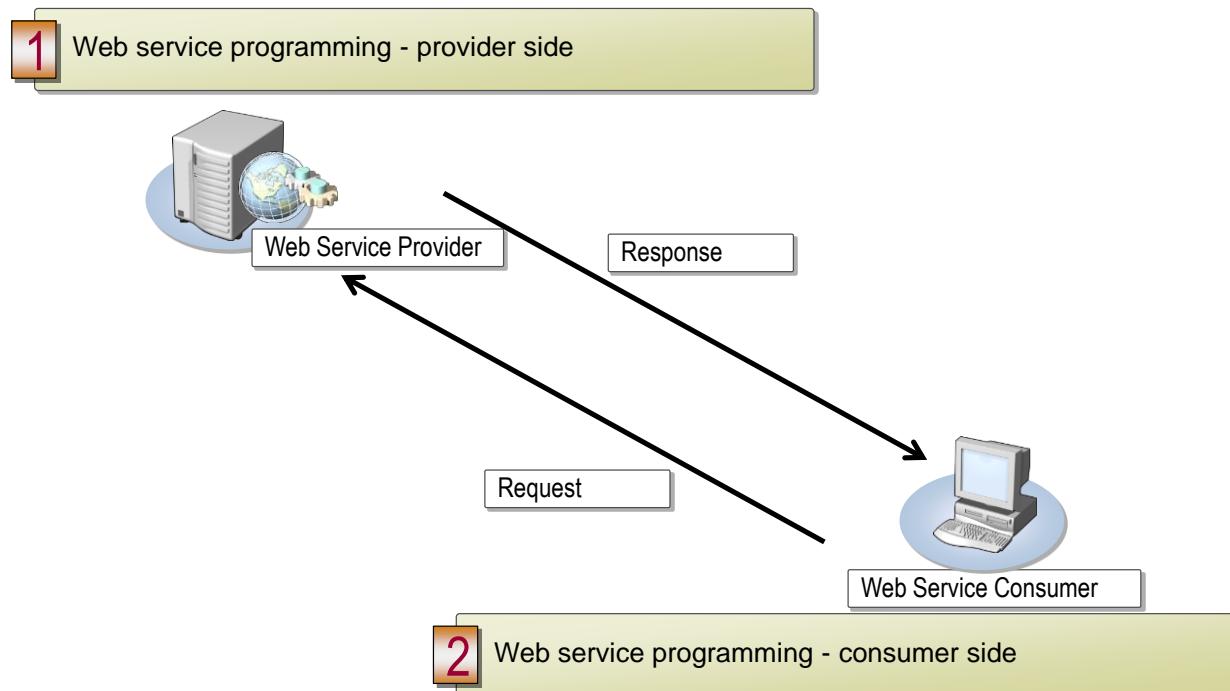
**Bước 8:** Tương tự login vào account buyer để xem thông tin về transaction phía người mua (buyer):

The screenshot shows a PayPal account dashboard. At the top, there's a navigation bar with links for My Account, Send Money, Request Money, Merchant Services, and Products & Services. Below the navigation bar, there are links for Overview, Add Money, Withdraw, History, Statements, Resolution Center, and Profile. A message "Welcome, test buyer" is displayed. The account type is Personal, status is Verified, and account limits are shown. The balance is \$9,499.00 USD, with a red box highlighting it. The word "Tiền giảm" is written next to the balance. Below the balance, there are links for My recent activity, Payments received, and Payments sent, along with a link to View all of my transactions. A table titled "My recent activity - Last 7 days (Jul 17, 2015-Jul 24, 2015)" shows two transactions: an authorization to test facilitator's Test Store completed on Jul 24, 2015, and a transfer from PayPal completed on Jul 24, 2015. The table includes columns for Date, Type, Name/Email, Payment status, Details, Order status/Actions, and Gross. On the right side, there's a "Notifications" sidebar with links to Add a mobile phone number, Add a home phone number, and Policy Updates. A note indicates the last log in was on July 24, 2015, at 7:28 AM PDT.

Date	Type	Name/Email	Payment status	Details	Order status/Actions	Gross
Jul 24, 2015	Authorization To	test facilitator's Test Store	Completed	<a href="#">Details</a>		-\$500.00 USD
Jul 24, 2015	Transfer From	PayPal	Completed	<a href="#">Details</a>		\$9,999.00 USD

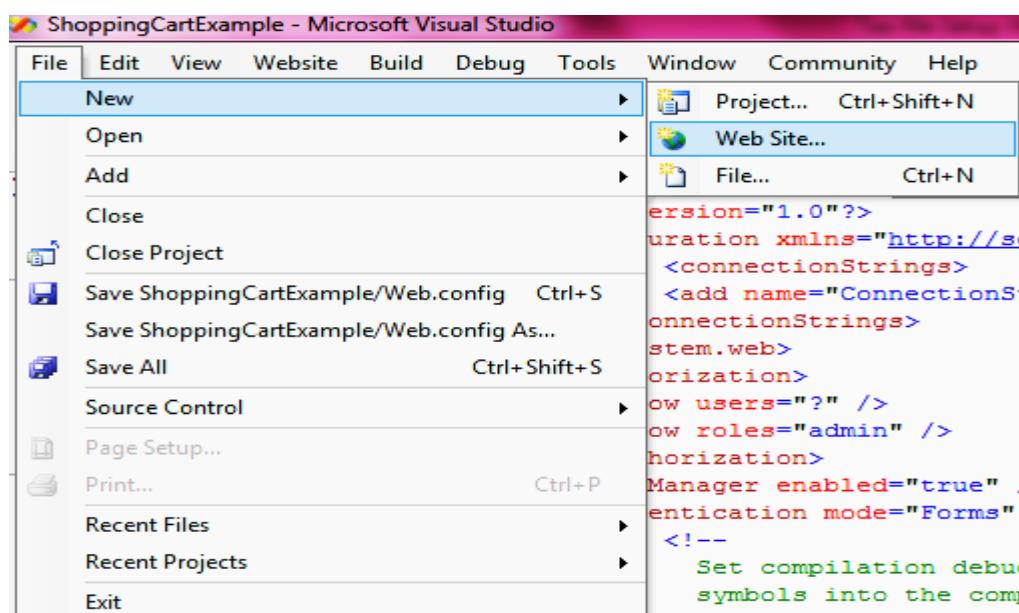
## HƯỚNG DẪN 3: WEBSITE SERVICES

# Xây dựng trang web cung cấp Web Service

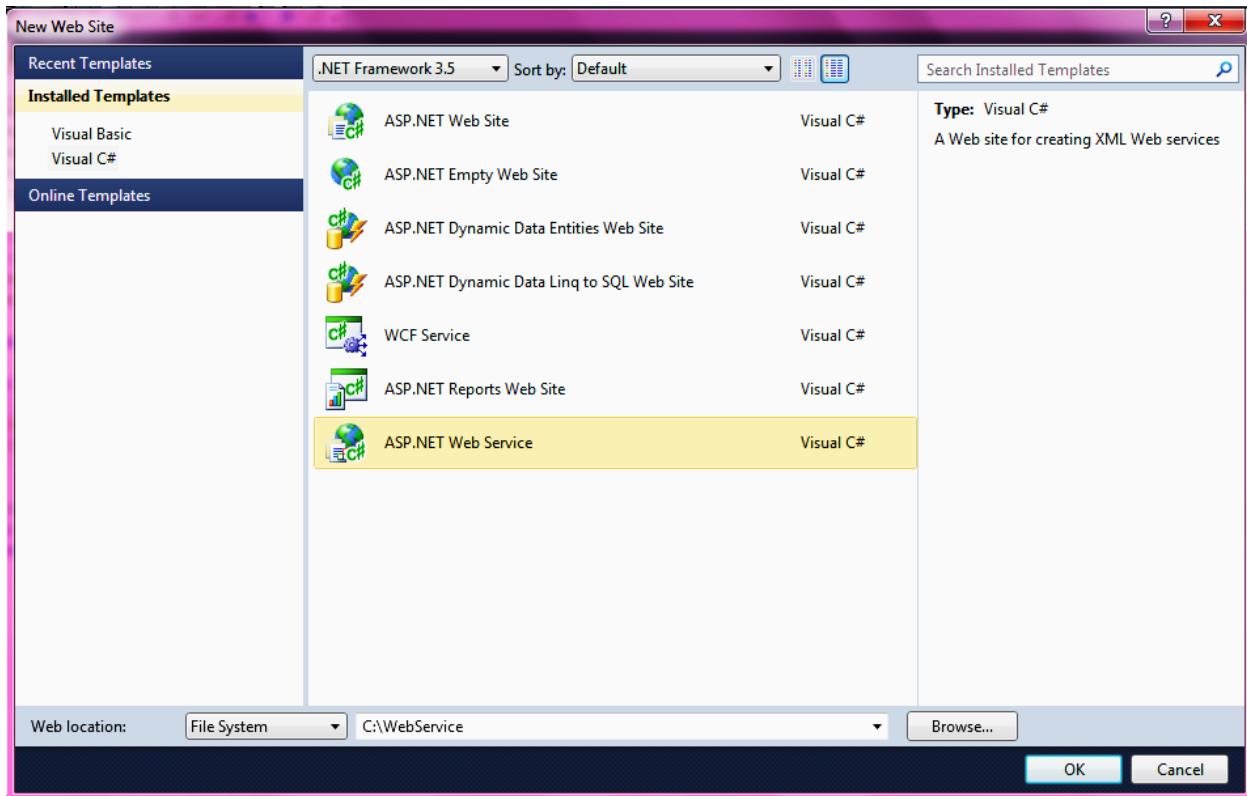


**Bước 1:** Đầu tiên mở chương trình Microsoft Visual Studio lên.

**Bước 2:** Tạo một project dạng website File → New → Web Site.



Rồi làm như sau



Sau khi bấm OK. Ta được project như sau

```

App_Code/Service.cs X
=====
Service.cs

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Services;
6
7  [WebService(Namespace = "http://tempuri.org/")]
8  [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
9  // To allow this Web Service to be called from script, using ASP.NET AJAX, uncomment the following line.
10 // [System.Web.Script.Services.ScriptService]
11
12 public class Service : System.Web.Services.WebService
13 {
14     public Service () {
15
16         //Uncomment the following line if using designed components
17         //InitializeComponent();
18     }
19
20     [WebMethod]
21     public string HelloWorld() {
22         return "Hello World";
23     }
24 }

```

Khai báo phương thức Web Service:

Bổ sung chỉ thị **[WebMethod]** trước khai báo hàm

Các thuộc tính của **[WebMethod]**:

- BufferResponse: Sử dụng bộ nhớ đệm để response.
- CacheDuration: thời gian cache.
- Description: mô tả.
- EnableSession: bật session.
- TransactionOption: chọn lựa transaction.

Ví dụ ở đây viết thêm phương thức WhatIsYourName:

```
12  public class Service : System.Web.Services.WebService
13  {
14      public Service () {
15
16          //Uncomment the following line if using designed components
17          //InitializeComponent();
18      }
19
20      [WebMethod]
21      public string HelloWorld() {
22          return "Hello World";
23      }
24
25
26      [WebMethod]
27      public string WhatIsYourName(string firstname, string lastname)
28      {
29          return firstname + " " + lastname;
30      }
31 }
```

Bạn bấm F5, chạy để xem kết quả như sau

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [HelloWorld](#)
- [WhatIsYourName](#)

This web service is using <http://tempuri.org/> as its default namespace.

**Recommendation: Change the default namespace before the XML Web service is made public.**

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. <http://tempuri.org/> is available for XML Web services that are under development, but published XML Web services should use a more permanent namespace.

Your XML Web service should be identified by a namespace that you control. For example, you can use your company's Internet domain name as part of the namespace. Although many XML Web service namespaces look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces are URIs.)

For XML Web services created using ASP.NET, the default namespace can be changed using the WebService attribute's Namespace property. The WebService attribute is an attribute applied to the class that contains the XML Web service methods. Below is a code example that sets the namespace to "http://microsoft.com/webservices/":

```
C#
[WebService(Namespace="http://microsoft.com/webservices/")]
public class MyWebService {
    // implementation
}

Visual Basic
<WebService(Namespace:="http://microsoft.com/webservices/")> Public Class MyWebService
    ' implementation
End Class

C++
[WebService(Namespace="http://microsoft.com/webservices/")]
public ref class MyWebService {
    // implementation
};

For more details on XML namespaces, see the W3C recommendation on Namespaces in XML.
For more details on WSDL, see the WSDL Specification.
For more details on URIs, see RFC 2396.
```

Như hình trên, ta thấy service có địa chỉ:

<http://localhost:50976/WebService/Service.asmx>

Click vào WhatYourIsName

Click [here](#) for a complete list of operations.

## WhatIsYourName

### Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
firstname:	Jacky
lastname:	Hung

[Invoke](#)

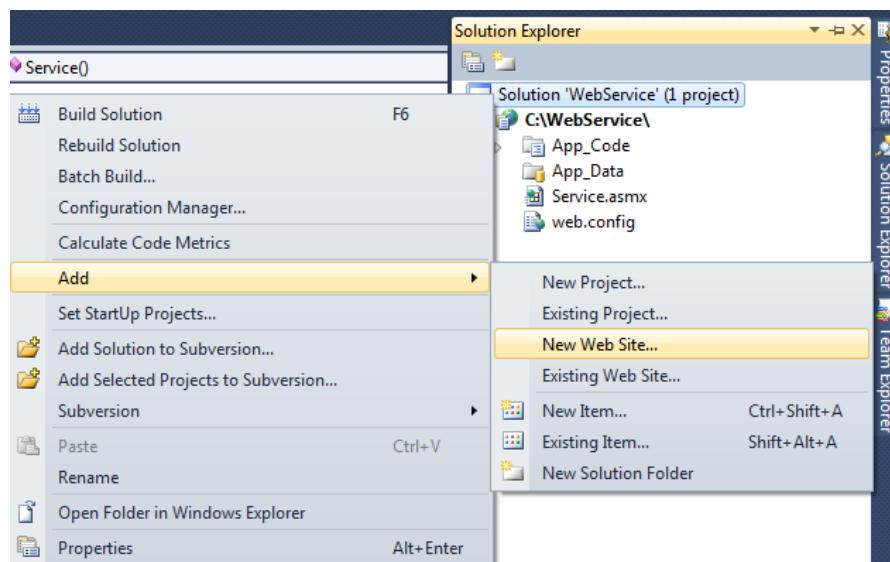
Bạn điền các thông số cần thiết cho hàm WhatIsYourName. Bấm vào nút Invoke để triệu gọi phương thức đó với các đối số đó. Và kết quả trả về như sau



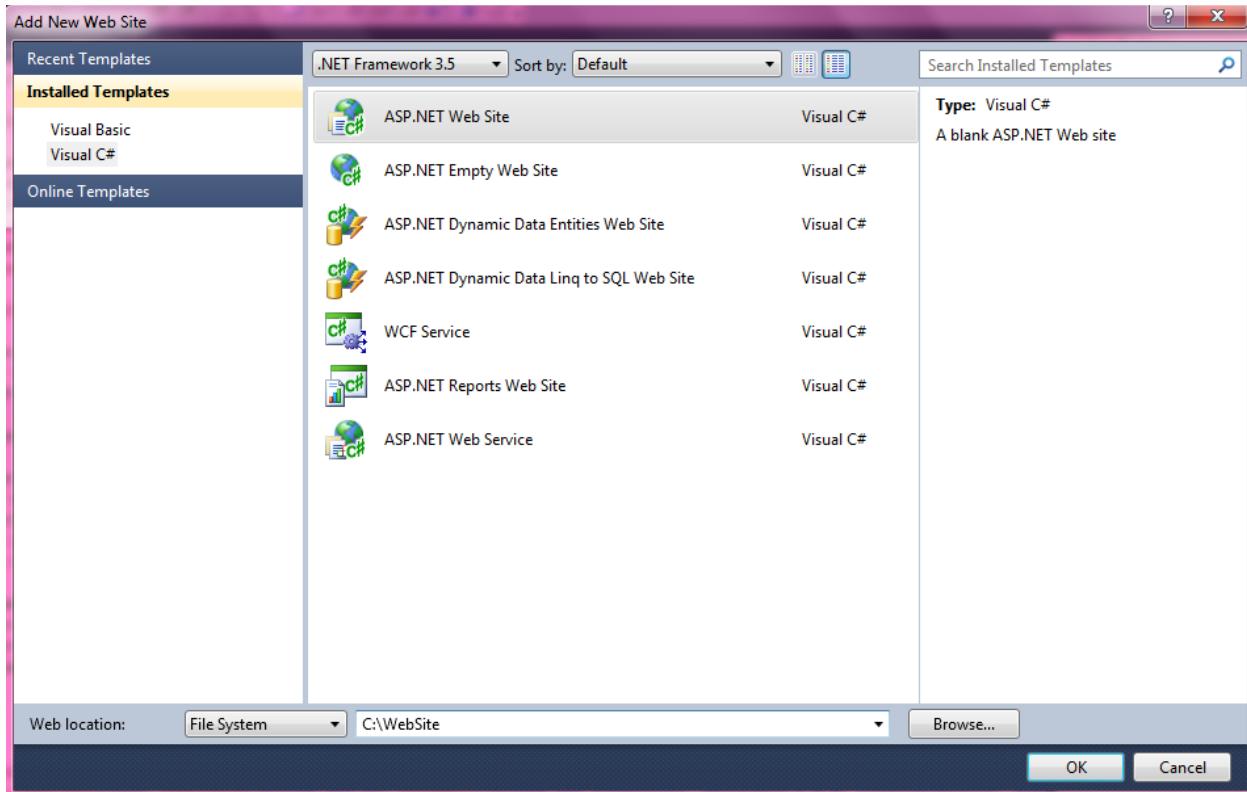
The screenshot shows a browser window with the address bar set to "localhost:50976/WebService/Service.asmx/WhatIsYourName". The page content displays the XML response: "This XML file does not appear to have any style information associated with it. The document tree is shown below." followed by the string "<string xmlns="http://tempuri.org/">Jacky Hung</string>".

## Xây dựng trang web sử dụng Website Service

### Bước 1: Tạo thêm một project website

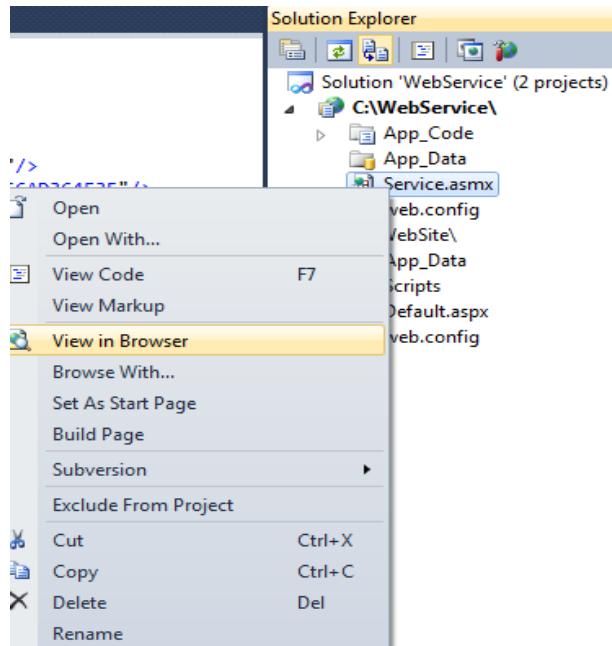


Click chuột phải vào Solution → Add → New Web Site.



## Bước 2: Add Web Reference.

Click chuột phải vào Service.asmx rồi chọn View in Browser.



Copy link sau để làm bước

sau

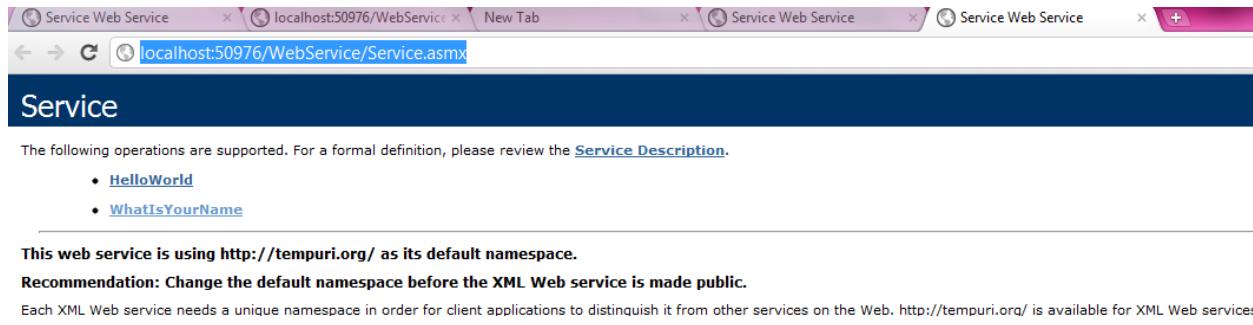
để

làm

bước

sau:

<http://localhost:50976/WebService/Service.asmx>



The following operations are supported. For a formal definition, please review the [Service Description](#).

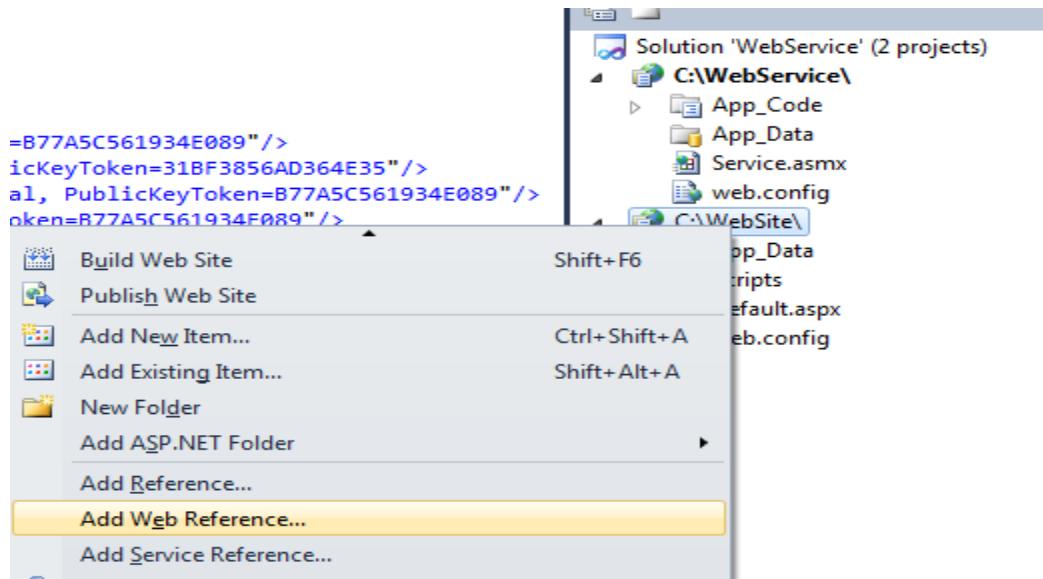
- [HelloWorld](#)
- [WhatIsYourName](#)

This web service is using <http://tempuri.org/> as its default namespace.

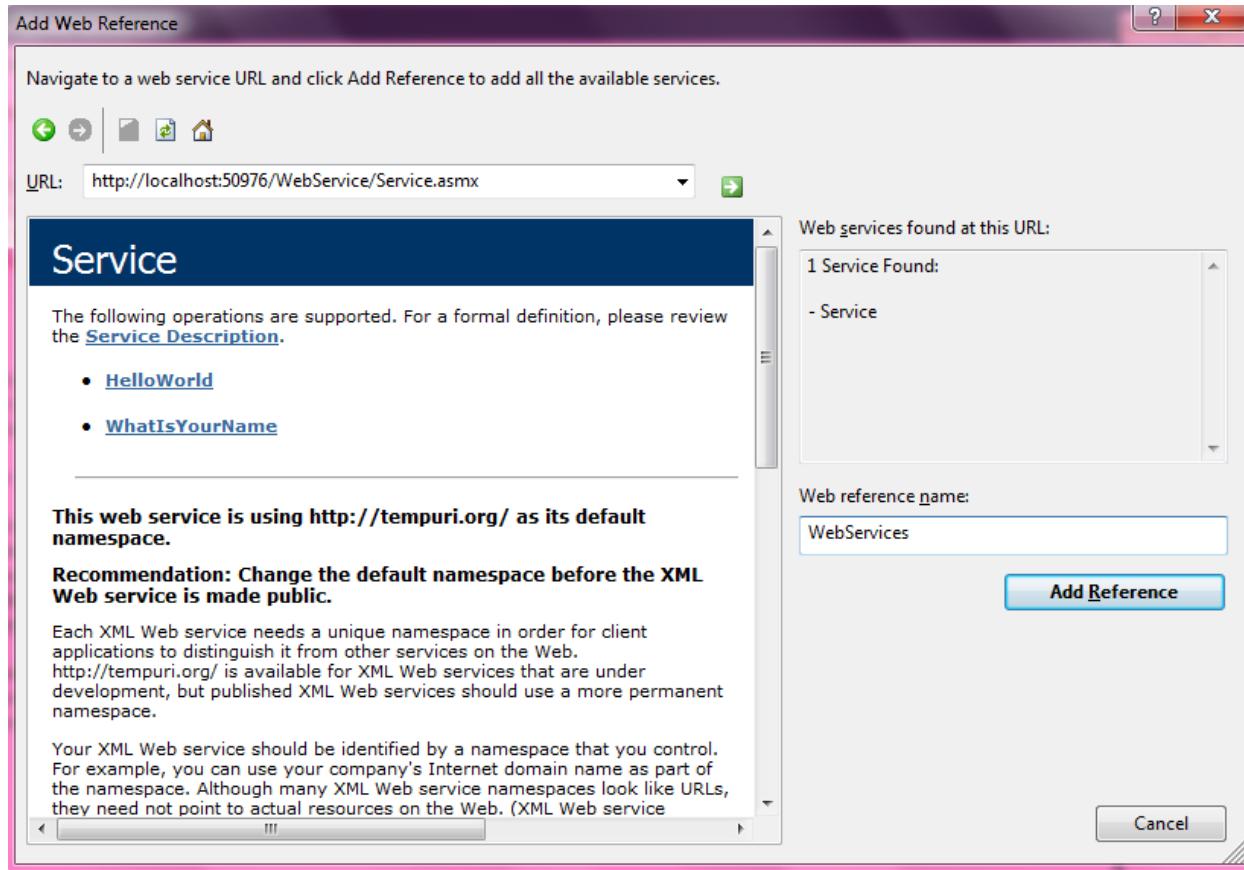
Recommendation: Change the default namespace before the XML Web service is made public.

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. <http://tempuri.org/> is available for XML Web services

Click chuột phải vào C:\ Website\ → Add Web Reference...

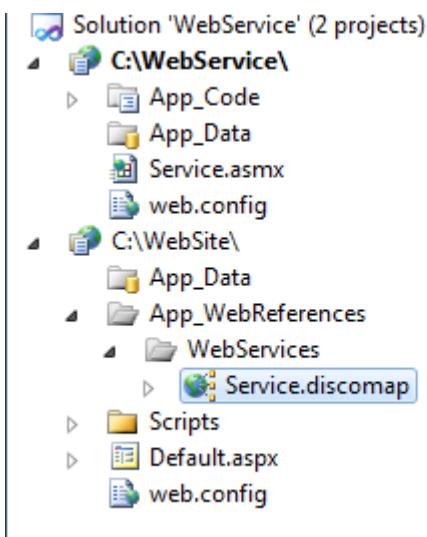


Past link trên vào hộp thoại sau và bấm Enter



Đặt tên cho server reference là WebServices rồi nhập nút add reference.

Và bạn được kết quả sau



Trong file Default.aspx bạn design lại như sau

The screenshot shows the Microsoft Visual Studio IDE. At the top, there are tabs for Default.aspx.cs, web.config, Default.aspx (which is the active tab), and App\_Code/Service.cs. Below the tabs, a title bar says "Client Objects & Events" and "(No Events)". The main area displays the ASPX page's source code:

```

1  <%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" 
2
3  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"
4
5  <html xmlns="http://www.w3.org/1999/xhtml">
6  <head runat="server">
7      <title></title>
8  </head>
9  <body>
10 <form id="form1" runat="server">
11     <div>
12         <asp:Button ID="Button1" runat="server" Text="Hello" onclick="Button1_Click" />
13         <asp:Label ID="Label1" runat="server" Text="Kết quả"></asp:Label>
14         <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
15     </div>
16     <div>
17         <asp:Button ID="Button2" runat="server" Text="What is your name" onclick="Button2_Click" />
18         <asp:Label ID="Label2" runat="server" Text="Kết quả"></asp:Label>
19         <asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>
20     </div>
21     </form>
22 </body>
23 </html>
24
25

```

Below the code editor, a preview window shows the ASPX page with two buttons and two text boxes. The first button is labeled "Hello" and the second is labeled "What is your name". The first text box contains "Kết quả" and the second contains "Kết quả".

Trong file Default.aspx.cs bạn thêm 2 hàm xử lý nút button như sau

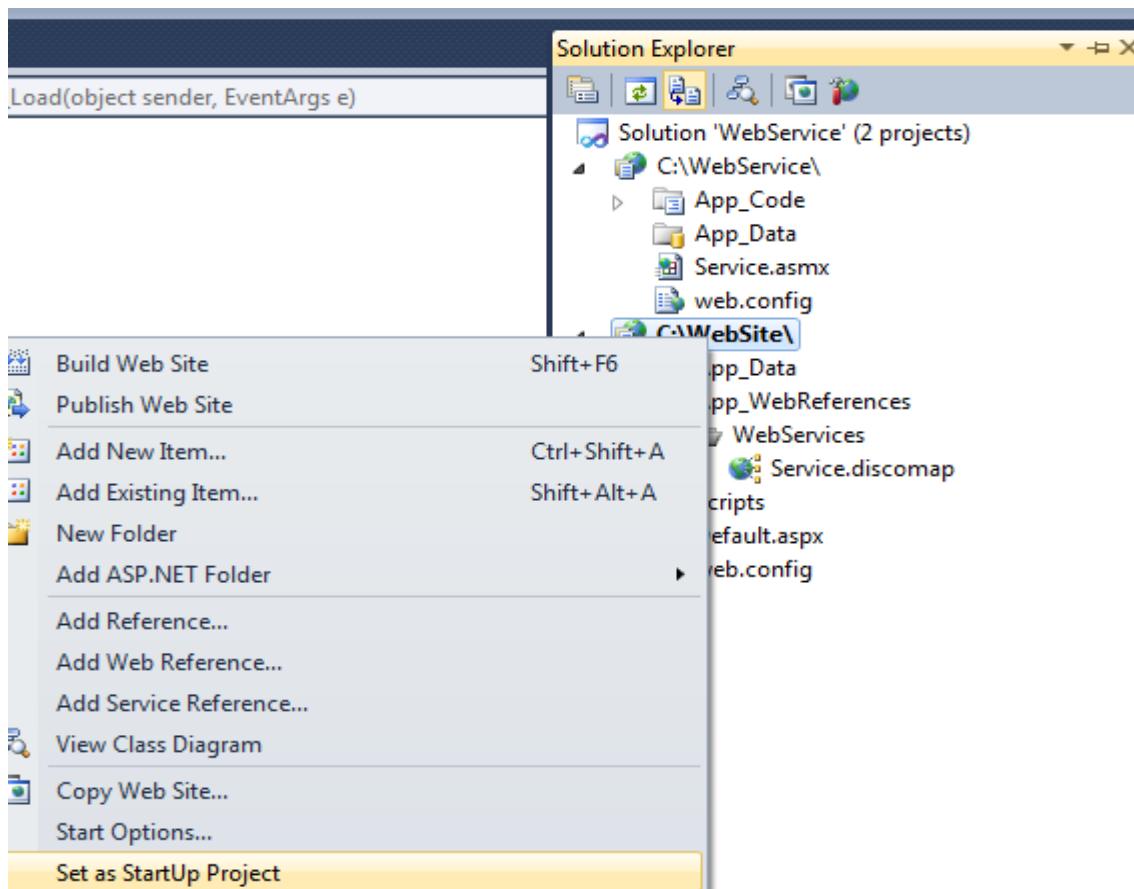
```

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        WebServices.Service service = new WebServices.Service();
        TextBox1.Text = service.HelloWorld();
    }
    protected void Button2_Click(object sender, EventArgs e)
    {
        WebServices.Service service = new WebServices.Service();
        TextBox2.Text = service.WhatIsYourName("Jacky", "Hưng");
    }
}

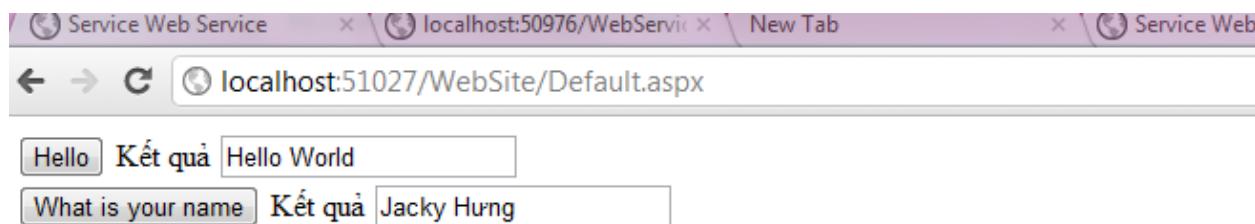
```

**Bước 3:** Chạy chương trình.

Set Project chính là trang web site của bạn.



Bấm F5 để chạy chương trình và nhấn vào 2 button để xem kết quả

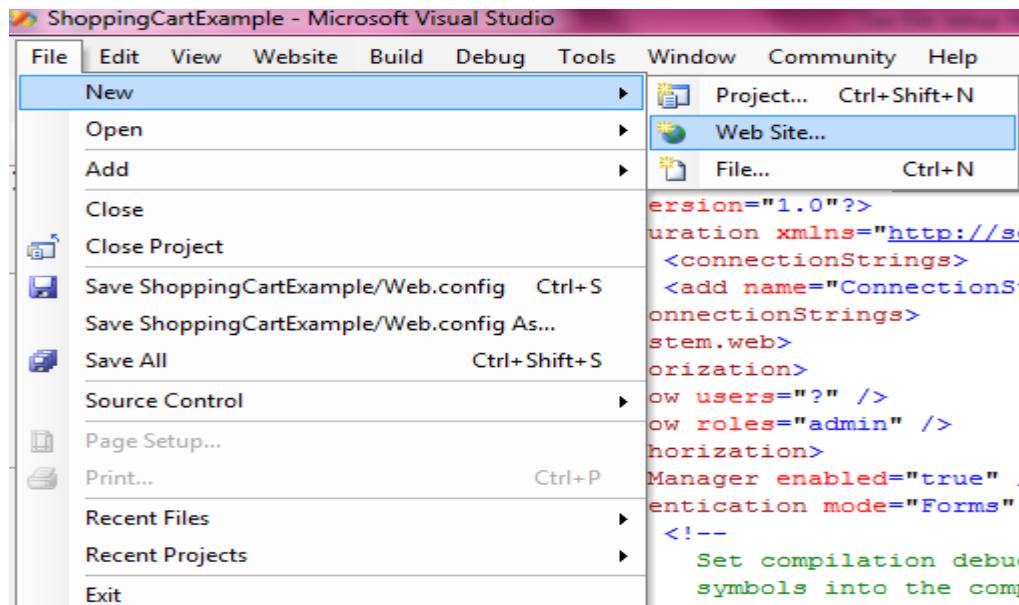


# HƯỚNG DẪN 4: GOOGLE MAP

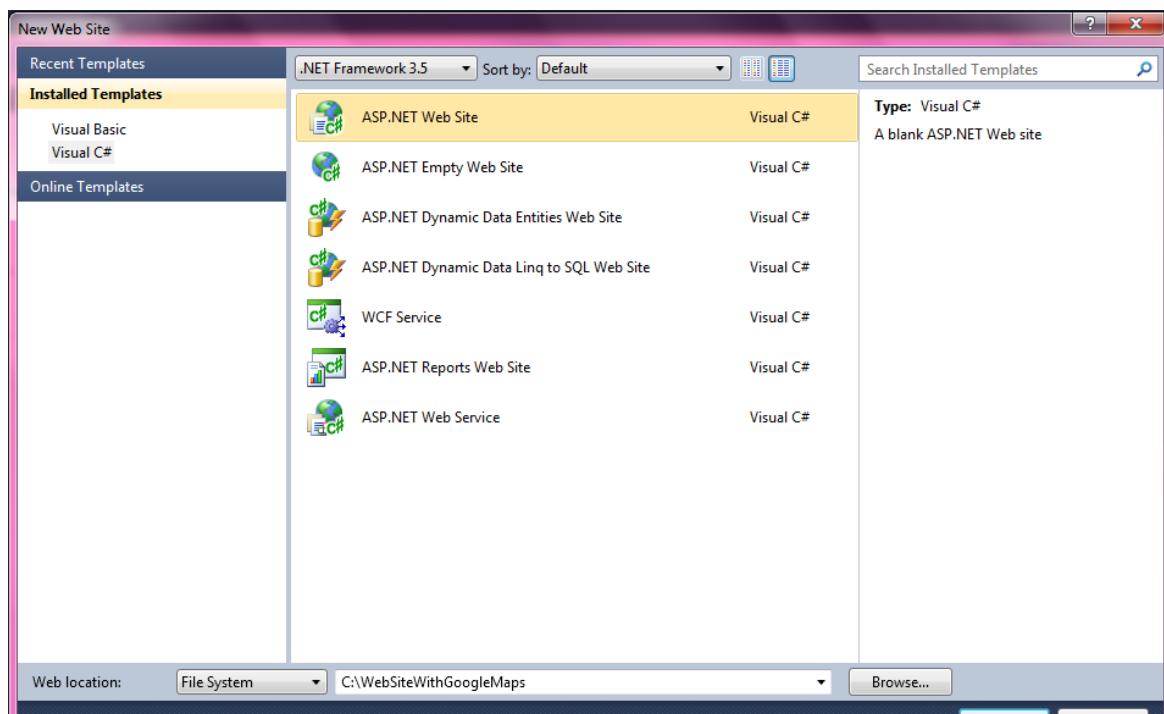
## Xây dựng trang web đánh dấu điểm có sử dụng google maps

**Bước 1:** Đầu tiên mở chương trình Microsoft Visual Studio lên.

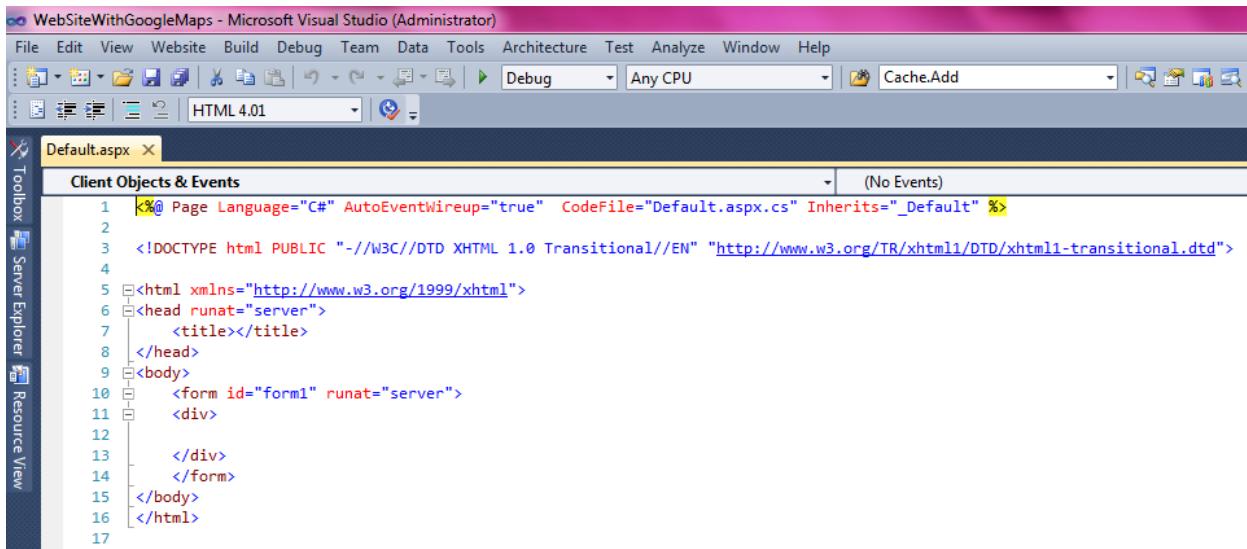
**Bước 2:** Tạo một project dạng website File → New → Web Site.



Rồi làm như sau.



Sau khi bấm OK. Ta được project như sau



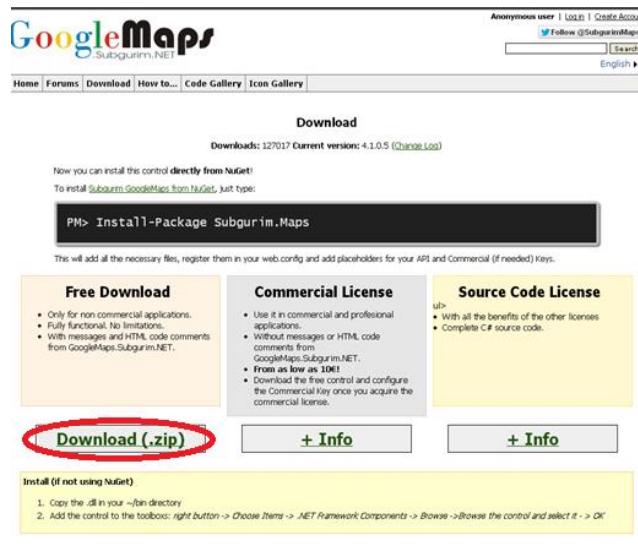
```

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>
</div>
</form>
</body>
</html>

```

### Bước 3: Dowload thư viện GMaps.dll tại

<http://en.googlemaps.subgurim.net/descargar.aspx>.



Downloads: 127017 Current version: 4.1.0.5 ([Change Log](#))

Now you can install the control [directly from NuGet!](#)

To install Subgurim.GoogleMaps from NuGet, just type:

PM> Install-Package Subgurim.Maps

This will add all the necessary files, register them in your web.config and add placeholders for your API and Commercial (if needed) keys.

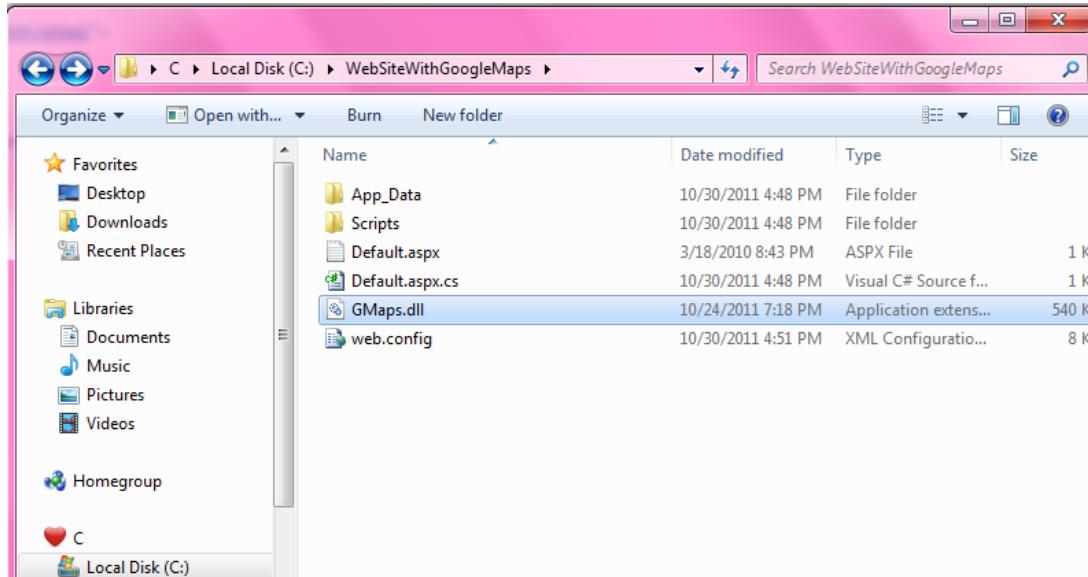
<b>Free Download</b>	<b>Commercial License</b>	<b>Source Code License</b>
<ul style="list-style-type: none"> <li>Only for non commercial applications.</li> <li>Fully functional. No limitations.</li> <li>With messages and HTML code comments from GoogleMaps.Subgurim.NET.</li> </ul>	<ul style="list-style-type: none"> <li>Use it in commercial and professional applications.</li> <li>Without messages or HTML code comments from GoogleMaps.Subgurim.NET.</li> <li>From as little as 10K.</li> <li>Download the free control and configure the Commercial Key once you acquire the commercial license.</li> </ul>	<ul style="list-style-type: none"> <li>With all the benefits of the other licenses.</li> <li>Complete C# source code.</li> </ul>

**Download (.zip)**      + Info      + Info

Install (if not using NuGet)

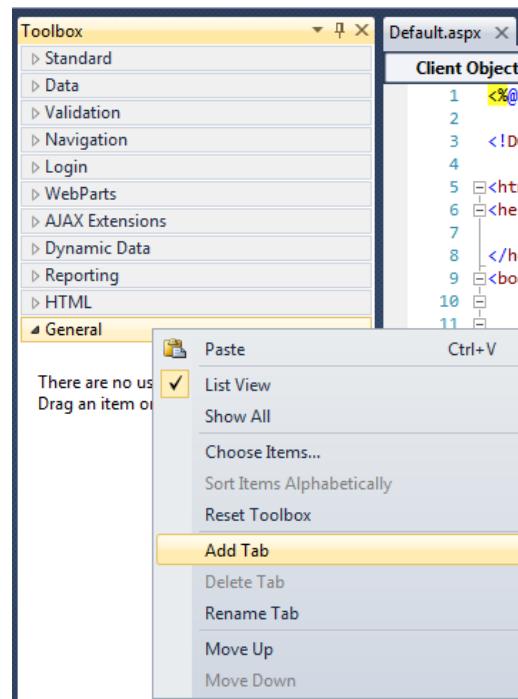
- Copy the .dll in your ~\bin directory
- Add the control to the toolbox: right button -> Choose Items -> .NET Framework Components -> Browse ->Browse the control and select it -> OK'

Giải nén ra, bạn được file Gmaps.dll. Chép file này vào project website của bạn

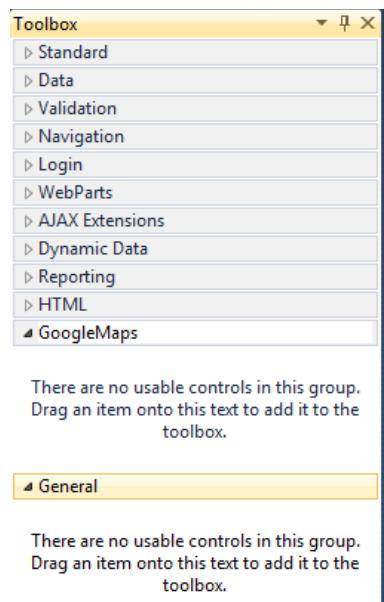


**Bước 4:** Add GMaps.dll vào Toolbox trong VS của bạn (thực hiện bước này để bạn kéo thả khi lập trình cho dễ) như sau:

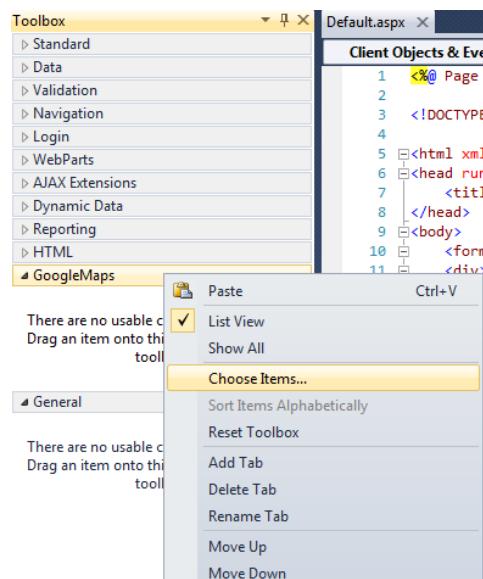
Chọn Tool Box -> Phải chuột vào General Tab -> chọn Add New Tab



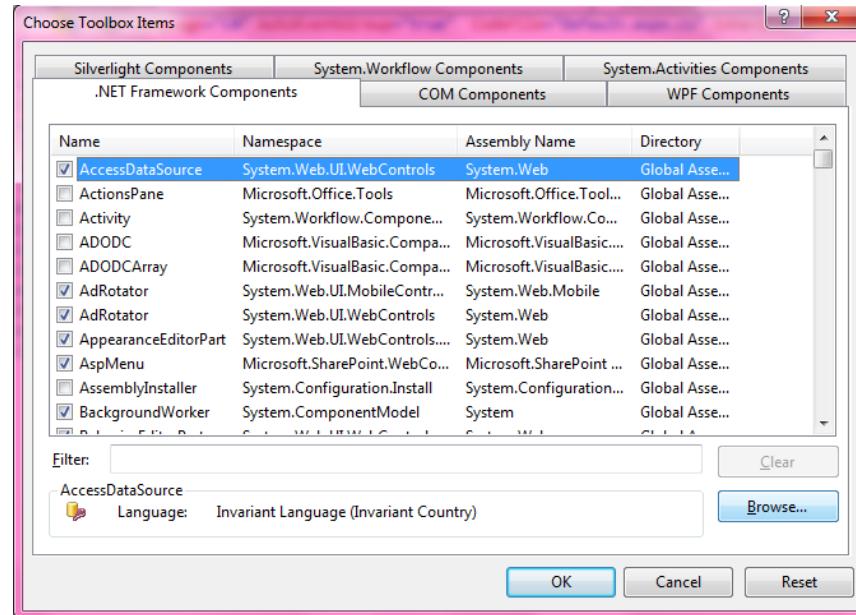
Đặt tên Tab là GoogleMaps



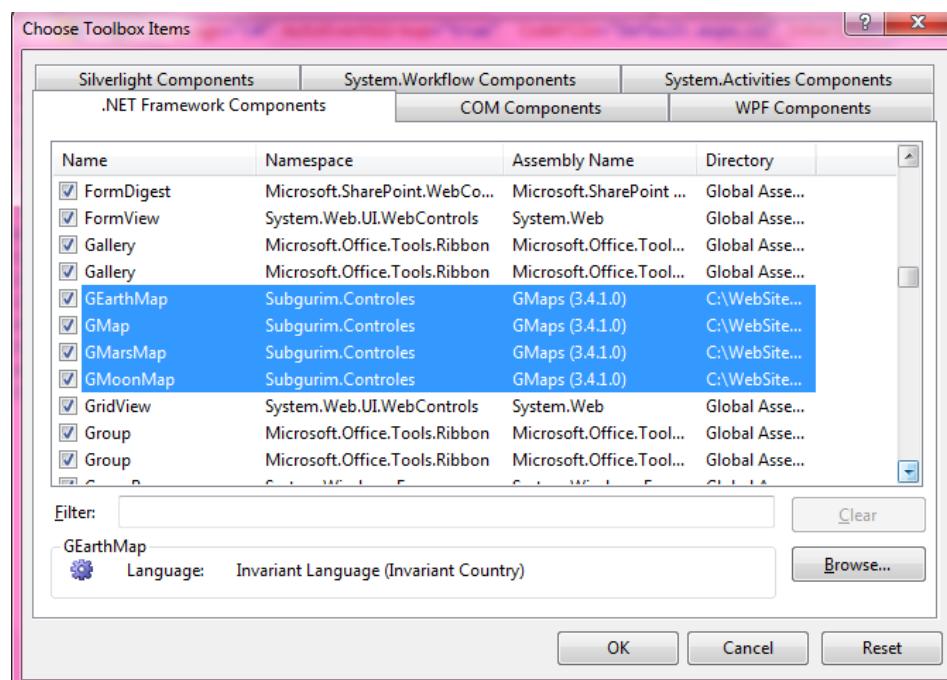
Nhấp phải chuột vào GoogleMaps chọn Choose Item..



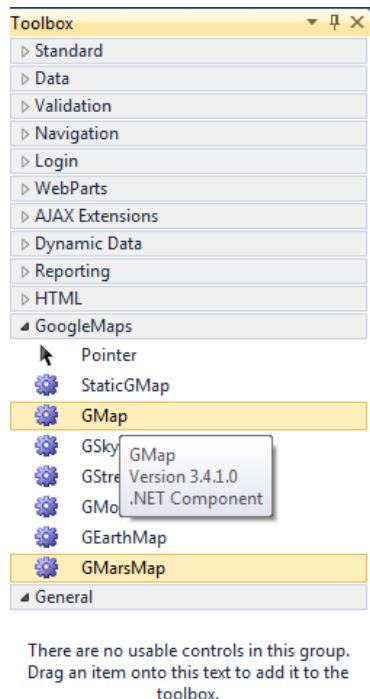
Click nút Browse



Chọn đến nơi lưu GMaps.dll -> OK



Bấm OK. Rồi nhìn lại bên toolbox bạn thấy có các control của google maps



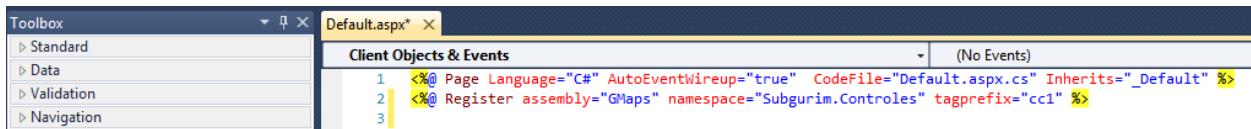
### **Bước 5:** Thiết kế trang default.aspx

Bước này minh họa thiết kế 1 Form đơn giản để người dùng nhập các thông tin cần search.

A screenshot of the default.aspx design view in Visual Studio. The form consists of four text input fields labeled 'Street', 'District', 'City', and 'Country', each preceded by a descriptive label ('Street:', 'District:', 'City:', 'Country:'). Below these fields is a 'Show' button. The entire form is contained within a single table cell, indicated by the 'td' label at the top left of the form area.

Đầu tiên đăng ký dùng assembly của Gmaps.dll bạn cần đăng ký dòng code sau trong trang aspx (Nếu bạn kéo thả từ Toolbox như hướng dẫn bước 2 thì dòng code này sẽ tự sinh cho bạn)

```
<%@ Register assembly="GMaps" namespace="Subgurim.Controles" tagprefix="cc1"
%>
```



Trong thân thẻ div của body trang bạn thêm dòng code này vào để thiết kế

```
<form id="form1" runat="server">
<table cellpadding="0" cellspacing="0" class="style2">
<tr>
    <td colspan="2">&nbsp;</td>
    <td width="70%" rowspan="6"><cc1:GMap ID="GMap1" runat="server" Width="700px" Height="500px" /></td>
</tr>
<tr>
    <td valign="top"> Street:</td>
    <td valign="top"> <asp:TextBox ID="txtStreet" runat="server"></asp:TextBox> </td>
</tr>
<tr>
    <td valign="top"> District:</td>
    <td valign="top"> <asp:TextBox ID="txtDistrict" runat="server"></asp:TextBox> </td>
</tr>
<tr>
    <td valign="top"> City:</td>
    <td valign="top"> <asp:TextBox ID="txtCity" runat="server"></asp:TextBox> </td>
</tr>
<tr>
    <td valign="top"> Country:</td>
    <td valign="top"> <asp:TextBox ID="txtCountry" runat="server"></asp:TextBox></td>
</tr>
<tr>
    <td valign="top" colspan="2"><asp:Button ID="btnShow" runat="server" Text="Show" /></td>
</tr>
</table>
</form>
```

**Bước 6:** Viết code xử lý khi nhấp vào nút ShowMap thì sẽ hiển thị vị trí tên đường, thành phố và đất nước lên.

Trong file defualt.aspx thêm đoạn code sau vào

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using Subgurim.Controles;
using System.Configuration;

public partial class Default2 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void btnShow_Click(object sender, EventArgs e)
    {
        GMap1.reset();
        //Lấy địa chỉ
        string Dia_chi = string.Format("{0},{1},{2},{3}", txtStreet.Text
                                         , txtDistrict.Text
                                         , txtCity.Text
                                         , txtCountry.Text);
        //Lấy API Key
        string skey = ConfigurationManager.AppSettings["googlemap.subgurim.net"];
        //Yêu cầu google map trả về tọa độ ứng với địa chỉ đưa cho google
        GeoCode geocode;
        geocode = GMap1.getGeoCodeRequest(Dia_chi);
        //lấy kinh độ và vĩ độ mà google trả về
        Subgurim.Controles.GLatLn gLatLn =
            new Subgurim.Controles.GLatLn(geocode.Placemark.coordinates.lat, geocode.Placemark.coordinates.lng);
        //tạo marker(đánh dấu) tại kinh độ vĩ độ
        Subgurim.Controles.GMarker gMarker = new Subgurim.Controles.GMarker(gLatLn);
        GMap1.resetMarkers();
        //đưa đánh dấu lên bản đồ
        GMap1.Add(gMarker);
        //chú thích tại điểm đánh dấu
        GInfoWindow window = new GInfoWindow(gMarker, @"<b>" +
                                                "</b> <br /> <b>Địa chỉ</b>:" + txtStreet.Text +
                                                "<br /><b>Quận</b>:" + txtDistrict.Text +
                                                "<br /><b>Thành Phố</b>:" + txtCity.Text + " ", true);
        GMap1.Add(window);
        GMap1.resetCenter();
        GMap1.setCenter(gLatLn, 16, Subgurim.Controles.GMapType.GTypes.Normal);
        //đưa điều khiển phóng to thu nhỏ vào bản đồ
        GMap1.Add(new GControl(GControl.preBuilt.GOvewMapControl));
        GMap1.Add(new GControl(GControl.preBuilt.LargeMapControl));
    }
}

```

Chú ý phải using thư viện.

### Bước 7: Đăng ký google API Keys

Vào

link

[https://developers.google.com/maps/documentation/javascript/tutorial#api\\_key](https://developers.google.com/maps/documentation/javascript/tutorial#api_key) để đăng ký và lấy googleAPI Key.

The screenshot shows a browser window with the URL [https://developers.google.com/maps/documentation/javascript/tutorial#api\\_key](https://developers.google.com/maps/documentation/javascript/tutorial#api_key). The page title is "Obtaining an API Key". A note states: "\* Google Maps API for Business developers must *not* include a key in their requests. Please refer to [Loading the Google Maps JavaScript API](#) for Business-specific instructions." Below this, a section titled "To create your API key:" contains four steps:

- Visit the APIs Console at <https://code.google.com/apis/console> and log in with your Google Account.
- Click the Services link from the left-hand menu.
- Activate the Google Maps API v3 service.
- Click the API Access link from the left-hand menu. Your API key is available from the API Access page, in the Simple API Access section. Maps API applications use the Key for browser apps.

A screenshot of the "API Access" section in the Google APIs console is shown, highlighting the "API Access" link in the sidebar.

## Sau khi đăng nhập rồi chọn API Access

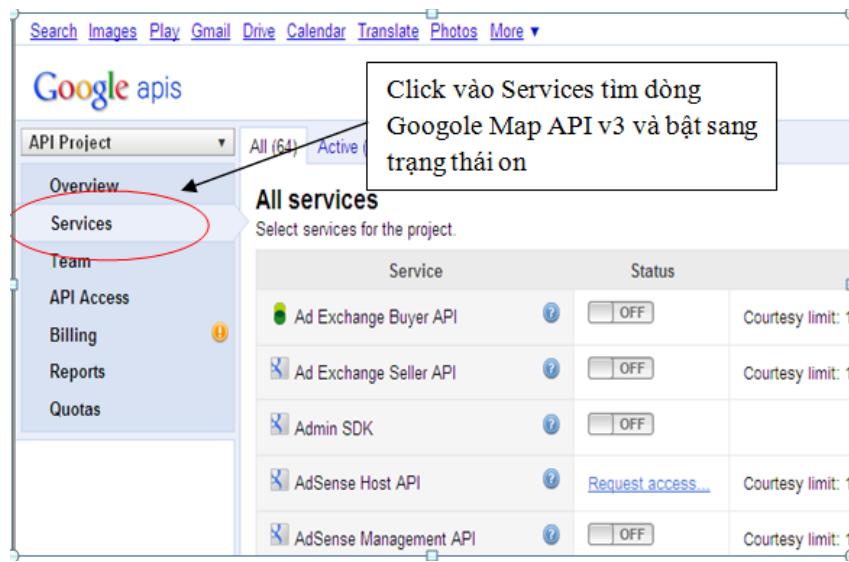
The screenshot shows the "API Project" sidebar with "API Access" selected. The main area is titled "API Access" with the sub-section "Authorized API Access". It explains OAuth 2.0 and provides a "Create an OAuth 2.0 client ID..." button. Below this is the "Simple API Access" section, which uses API keys for non-user data access. It shows a table for a "Key for browser apps (with referrers)" with the following details:

API key:	AIzaSyC9fo4zE6qBaTaG-6hV29ddwhd1OnYKPwM	Generate new key...
Referrers:	Any referrer allowed	Edit allowed referrers...
Activated on:	Aug 9, 2012 2:42 AM	Delete key...
Activated by:	jackyhung12345@gmail.com - <b>you</b>	

At the bottom are buttons for "Create new Server key..." and "Create new Browser key...".

Nếu như ko có browser apps hì bạn bấm nút create new browser key để lấy key.

Tuy nhiên để sử dụng được thì cần phải bật tính năng Google Map API như sau:



Google Cloud Storage JSON API	<a href="#">?</a>	<input type="button" value="OFF"/>	
Google Compute Engine	<a href="#">?</a>	<input type="button" value="OFF"/>	<a href="#">Pricing</a>
Google Contacts CardDAV API	<a href="#">?</a>	<input type="button" value="OFF"/>	Courtesy limit: 10,000 requests/day
Google Maps Android API v2	<a href="#">?</a>	<input type="button" value="OFF"/>	
Google Maps API v2	<a href="#">?</a>	<input type="button" value="OFF"/>	Courtesy limit: 25,000 requests/day • <a href="#">Pricing</a>
Google Maps API v3	<a href="#">?</a>	<input checked="" type="button" value="ON"/>	Courtesy limit: 25,000 requests/day • <a href="#">Pricing</a>
Google Maps Coordinate API	<a href="#">?</a>	<input type="button" value="OFF"/>	Courtesy limit: 1,000 requests/day
Google Maps Engine API	<a href="#">?</a>	<input type="button" value="OFF"/>	Courtesy limit: 10,000 requests/day
Google Maps Geolocation API	<a href="#">?</a>	<input type="button" value="OFF"/>	Courtesy limit: 0 requests/day • <a href="#">Pricing</a>
Google Maps SDK for iOS	<a href="#">?</a>	<input type="button" value="OFF"/>	

### Bước 8: Sửa web.config

Ở bước 6 khi viết code bạn dùng string skey = ConfigurationManager.AppSettings["googlemaps.subgurim.net"]; do đó bạn cần sửa trong web.config bằng cách thêm key="googlemaps.Subgurim.net" bên trong thẻ appSettings như sau:

```

<configuration>
  <system.web>
    <compilation debug="true" targetFramework="4.0">
      <assemblies>
        <add assembly="System.Design, Version=4.0.0.0, Culture=neutral, PublicKey
          <add assembly="System.Web.Extensions.Design, Version=4.0.0.0, Culture=neu
          <add assembly="System.Windows.Forms, Version=4.0.0.0, Culture=neutral, Pu
        </assemblies>
      <appSettings>
        <add key="googlemap.subgurim.net" value="Chép API key ở bước trên vào đây"/>
      </appSettings>
      <connectionStrings>
        <add name="Chuoi_ket_noi" connectionString="Data Source=.\SQLEXPRESS;Initial
      </connectionStrings>
    </system.web>
  </configuration>

```

Thay dòng YourGoogleMapsAPIKeyHere bằng API Key mà bạn đã có được ở bước 7.

**Bước 9:** Chạy và xem kết quả

## Ứng dụng tìm đường đi

**Bước 1:** Tạo một project mới dạng Website.

**Bước 2:** Thiết kế cho trang default.aspx như sau

```

<body>
  <form id="form1" runat="server">
    <table cellpadding="0" cellspacing="0" class="style2">
      <tr>
        <td>
          Từ: <asp:TextBox ID="tb_fromPoint" runat="server" meta:resourcekey="tb_fromPointResource1"></asp:TextBox>
          Đến: <asp:TextBox ID="tb_endPoint" runat="server" meta:resourcekey="tb_endPointResource1"></asp:TextBox>

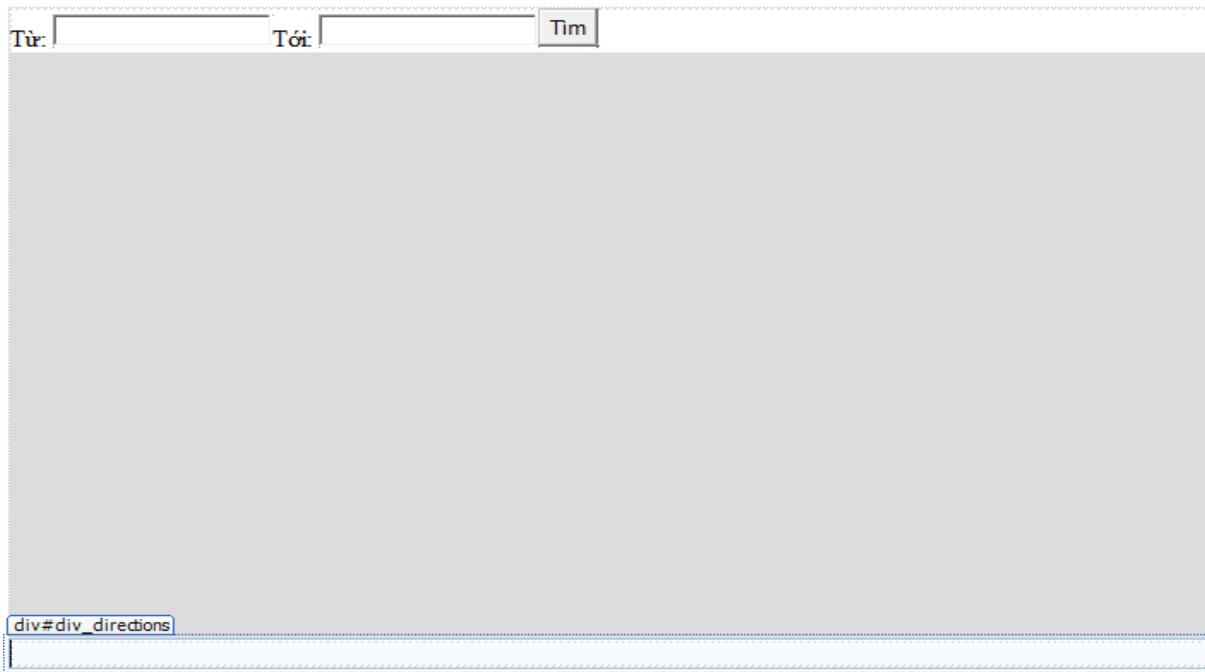
          <input type="button" id="bt_Go" value="Tìm" />

          <br />

          <cc1:GMap ID="GMap1" runat="server" Width="1000px" Height="800px" />

          <div id="div_directions"></div>
        </td>
      </tr>
      </table>
    </form>
  </body>

```



**Bước 3:** Xử lý khi tìm đường đi. Vào default.aspx.cs và code như sau:

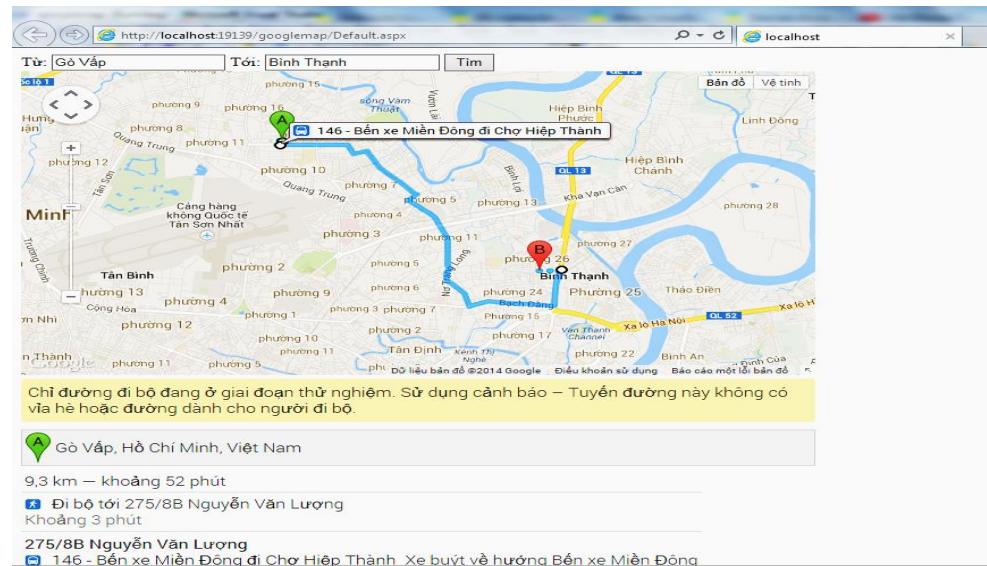
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using Subgurim.Controles;

public partial class Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            GDirection direction = new GDIRECTION();
            direction.autoGenerate = false;
            direction.buttonElementId = "bt_Go";
            //Lấy địa chỉ xuất phát
            direction.fromElementId = tb_fromPoint.ClientID;
            //lấy địa chỉ đích
            direction.toElementId = tb_endPoint.ClientID;
            //hiển thị chỉ dẫn
            direction.divElementId = "div_directions";
            direction.clearMap = true;
            //chỉ dẫn tránh đường cao tốc (mặc định là false)
            direction.avoidHighways = true;
            //lựa chọn phương tiện di chuyển(mặc định là Driving - xe hơi)
            // chọn GDIRECTION.GTravelModeEnum.TRANSIT (sử dụng xe bus)
            // GDIRECTION.GTravelModeEnum.WALKING (đi bộ)
            direction.travelMode = GDIRECTION.GTravelModeEnum.TRANSIT;
            //đưa chỉ dẫn lên bản đồ
        }
    }
}
```

```
        GMap1.Add(direction);
        GMap1.Add(new GControl(GControl.preBuilt.GOOverviewMapControl));
        GMap1.Add(new GControl(GControl.preBuilt.MapTypeControl));
        GMap1.Add(new GControl(GControl.preBuilt.LargeMapControl));

    }
}
```

#### **Bước 4:** Xem Kết quả



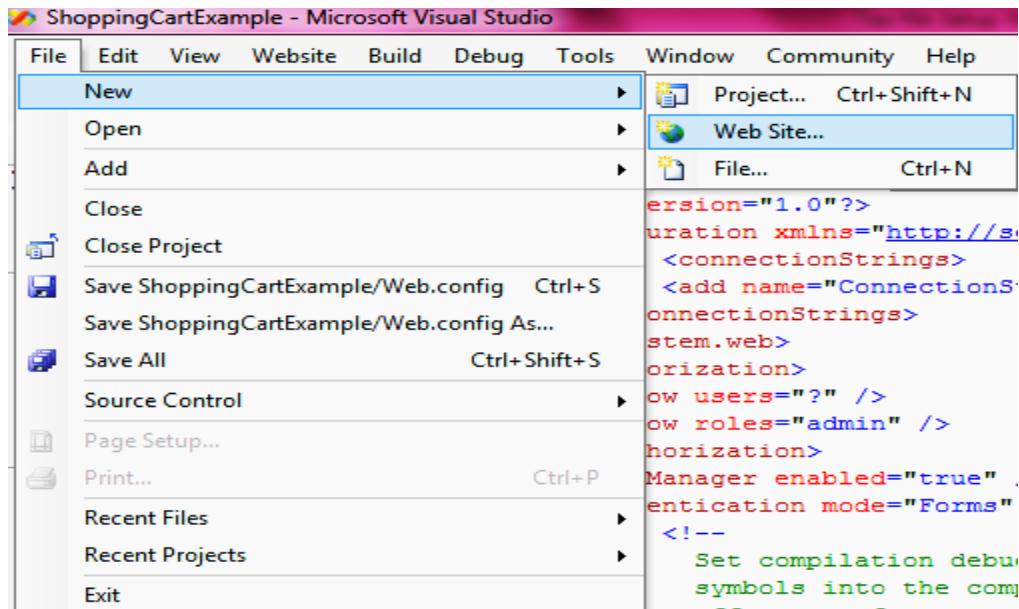
Muốn đổi ngôn ngữ của bản đồ ta vào properties của control Gmap tìm thuộc tính Language và chọn mã quốc gia vào. Ví dụ: Việt Nam – vi, Anh – en, Tây Ban Nha – Es-ES, Pháp – fr, ...

# HƯỚNG DẪN 5: TẠO VÀ LẤY TIN TỨC RSS

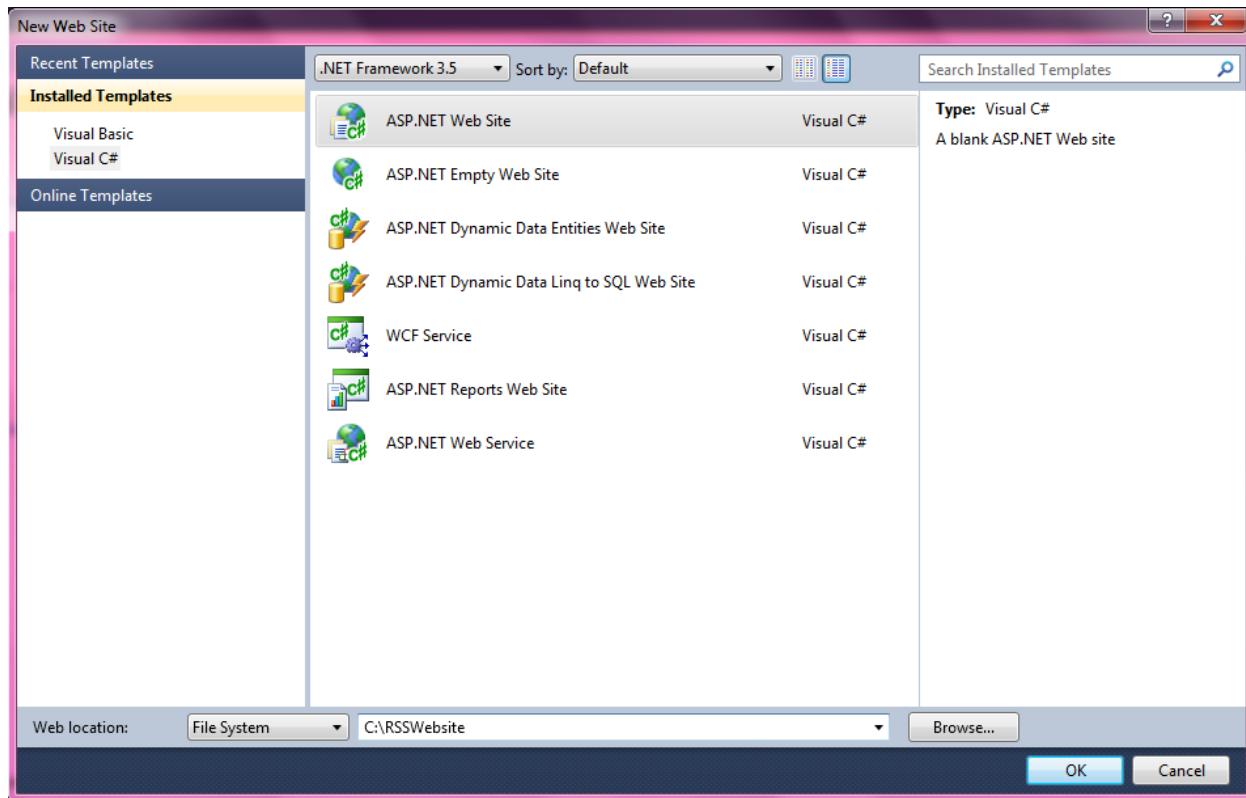
## Hướng dẫn tạo RSS trong asp.net

**Bước 1:** Đầu tiên mở chương trình Microsoft Visual Studio lên.

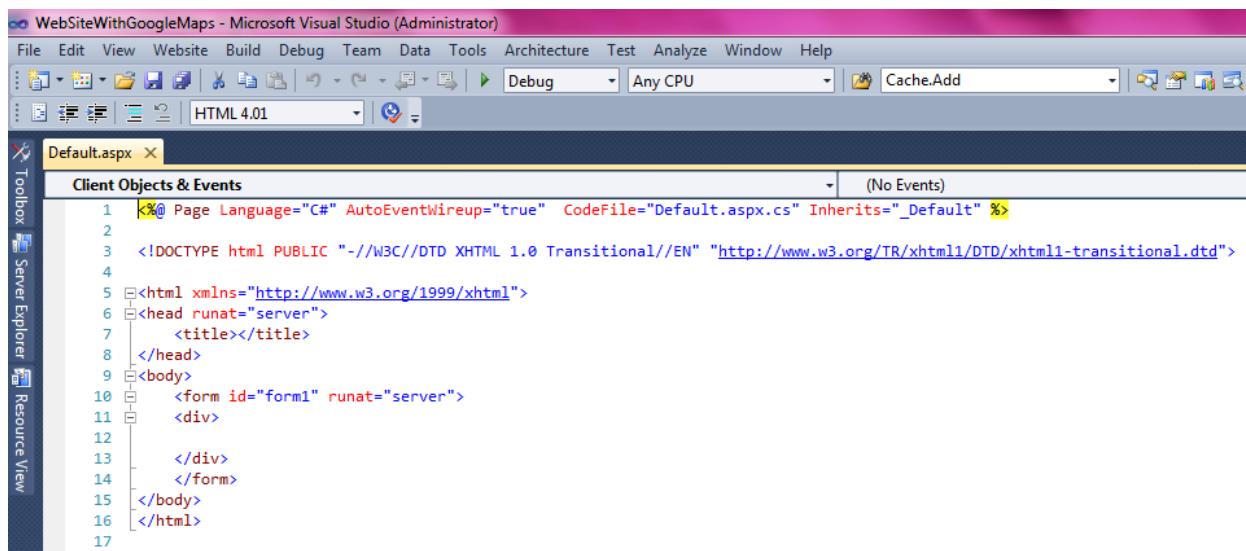
**Bước 2:** Tạo một project dạng website File → New → Web Site.



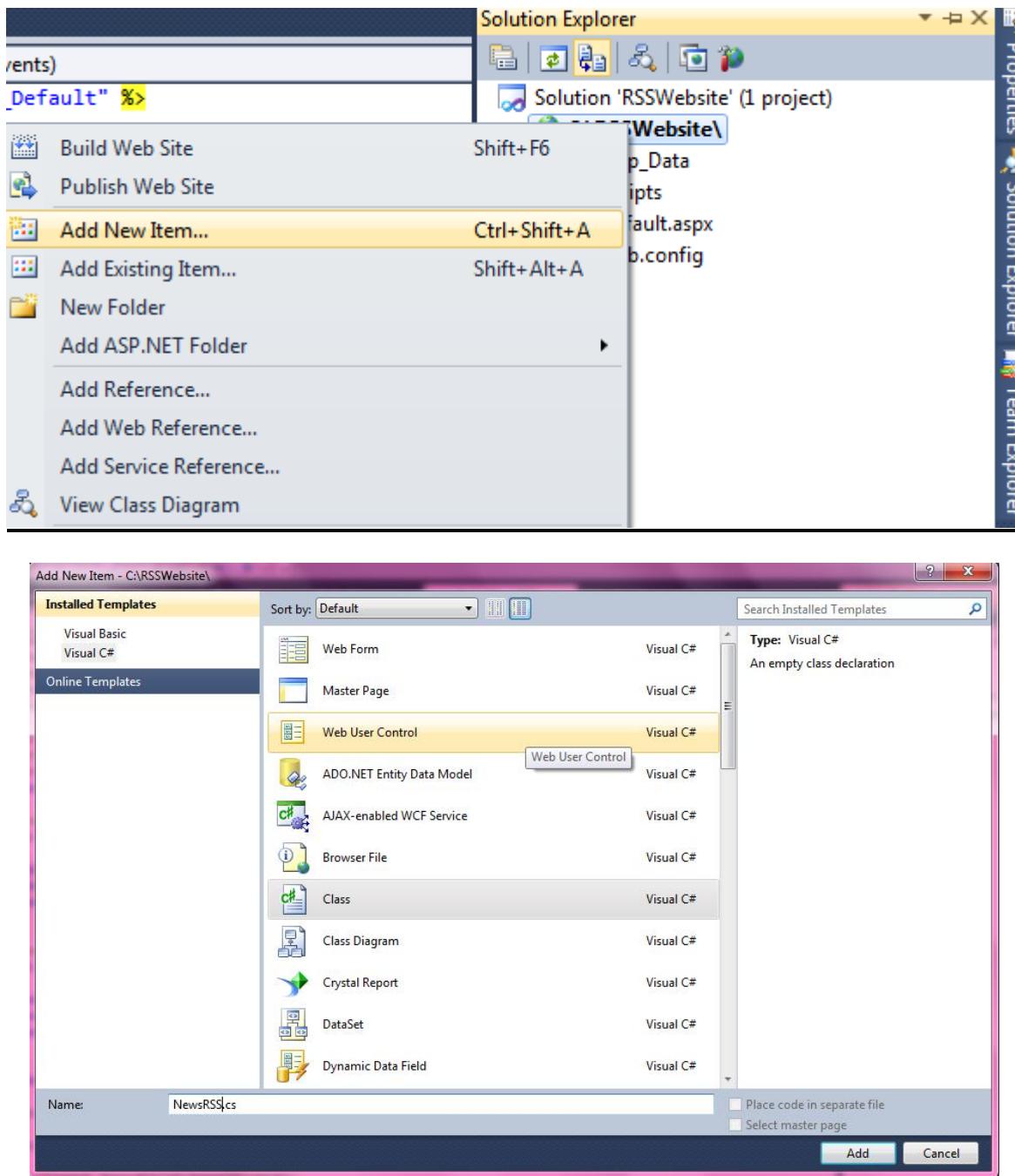
Rồi làm như sau



Sau khi bấm OK. Ta được project như sau



Tạo một file NewsRSS.cs như sau để tiến hành code.



**Bước 3:** Trong file NewsRSS.cs tiến hành thêm các đoạn code sau

Trong hàm Contructor của lớp NewsRSS thay đổi như sau

```

10  public class NewsRSS
11  {
12      XmlDocument rss; // tên biến lưu tài liệu xml
13      public NewsRSS()
14      {
15          // tạo một tài liệu dạng xml như sau
16          rss = new XmlDocument();
17          XmlDeclaration xmlDeclaration = rss.CreateXmlDeclaration("1.0", "utf-8", null);
18          rss.InsertBefore(xmlDeclaration, rss.DocumentElement);
19
20         XmlElement rssElement = rss.CreateElement("rss");
21          XmlAttribute rssVersionAttribute = rss.CreateAttribute("version");
22
23          rssVersionAttribute.InnerText = "2.0";
24          rssElement.Attributes.Append(rssVersionAttribute);
25
26          rss.AppendChild(rssElement);
27
28      }
29

```

### Tạo một struct cho một RSS Channel

```

//struct cho một RSS Channel
public struct RssChannel
{
    public string Title; // tên tựa đề của kênh/chủ đề ví dụ có thể đặt là "JACKYHUNG BLOG Website";
    public string Link; // link chính thức đến kênh hay chủ đề ví dụ "http://vn.360plus.yahoo.com/jackyhung12345";
    public string Description; // mô tả kênh hay chủ đề "Blog của Jacky Hưng";
}

```

### Tạo cấu trúc cho một RSS Item

```

//struct cho một RSS Item
public struct RssItem
{
    public string Title;// tên tựa đề của item/bài viết ví dụ có thể đặt là "Tổng hợp tài liệu môn học";
    public string Link;// link chính thức đến item/bài viết ví dụ "http://vn.360plus.yahoo.com/jackyhung12345/article?mid=5";
    public string Description;// mô tả item/bài viết "Tổng hợp tài liệu các môn học của Jacky Hưng";
}

```

### Viết Method tạo ra một RSS channel như sau

```

public void addRssChannel(RssChannel channel)
{
    // tạo ra cấu trúc XML gồm các tag như cấu trúc của RSS Channel
    XmlElement channelElement = rss.CreateElement("channel");
    XmlNode rssElement = rss.SelectSingleNode("rss");
    rssElement.AppendChild(channelElement);

    XmlElement titleElement = rss.CreateElement("title");
    titleElement.InnerText = channel.Title;
    channelElement.AppendChild(titleElement);

    XmlElement linkElement = rss.CreateElement("link");
    linkElement.InnerText = channel.Link;
    channelElement.AppendChild(linkElement);

    XmlElement descriptionElement = rss.CreateElement("description");
    descriptionElement.InnerText = channel.Description;
    channelElement.AppendChild(descriptionElement);

    // Tạo một thẻ tag XML để ghi thông tin chung
    XmlElement generatorElement = rss.CreateElement("generator");
    generatorElement.InnerText = "Your RSS Generator name and version. JACKYHUNG :)";
    channelElement.AppendChild(generatorElement);
}

```

Viết method tạo ra một RSS Item như sau

```

71 public void addRssItem(RssItem item)
72 {
73     // tạo ra cấu trúc XML gồm các tag như cấu trúc của RSS ITEM
74     XmlElement itemElement = rss.CreateElement("item");
75     XmlNode channelElement = rss.SelectSingleNode("rss/channel");
76     XmlElement titleElement = rss.CreateElement("title");
77     titleElement.InnerText = item.Title;
78     itemElement.AppendChild(titleElement);

79     XmlElement linkElement = rss.CreateElement("link");
80     linkElement.InnerText = item.Link;
81     itemElement.AppendChild(linkElement);

82     XmlElement descriptionElement = rss.CreateElement("description");
83     descriptionElement.InnerText = item.Description;
84     itemElement.AppendChild(descriptionElement);

85     // Thêm item vào tài liệu xml
86     channelElement.AppendChild(itemElement);
87 }

```

Và thêm một hàm sau để lấy về dạng xml

```

public string RssDocument
{
    get { return rss.OuterXml; }
}

```

**Bước 4:** Trong file Default.aspx tiến hành thêm các đoạn code sau vào hàm pageload như sau

```

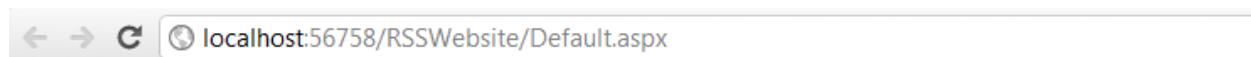
public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        // tạo RSS
        NewsRSS rss = new NewsRSS();
        // Tạo RSS Channel
        NewsRSS.RssChannel channel = new NewsRSS.RssChannel();
        channel.Title = "JACKYHUNG BLOG Website";
        channel.Link = "http://vn.360plus.yahoo.com/jackyhung12345";
        channel.Description = "Blog của Jacky Hưng";
        rss.addRssChannel(channel); // thêm RSS channel vào RSS

        // Tạo RSS Item
        NewsRSS.RssItem item = new NewsRSS.RssItem();
        item.Title = "Tổng hợp tài liệu môn học";
        item.Link = "http://vn.360plus.yahoo.com/jackyhung12345/article?mid=5";
        item.Description = "Tổng hợp tài liệu các môn học của Jacky Hưng";
        rss.addRssItem(item); // thêm RSS Item vào RSS

        // Hiển thị tài liệu RSS lên page defualt này
        Response.Clear();
        Response.ContentType = "text/xml";
        Response.Write(rss.RssDocument);
        Response.End();
    }
}

```

Rồi bấm F5 chạy trang web



This XML file does not appear to have any style information associated with it. The document tree is shown below.

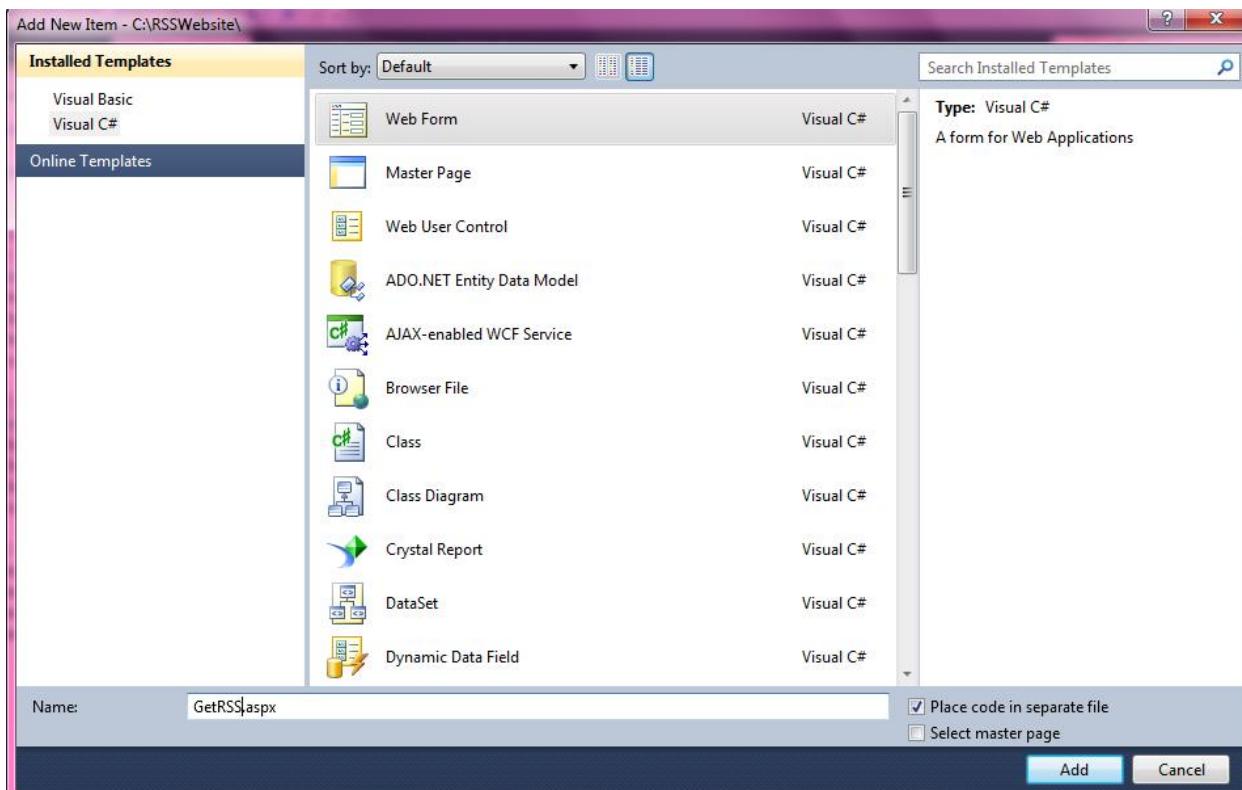
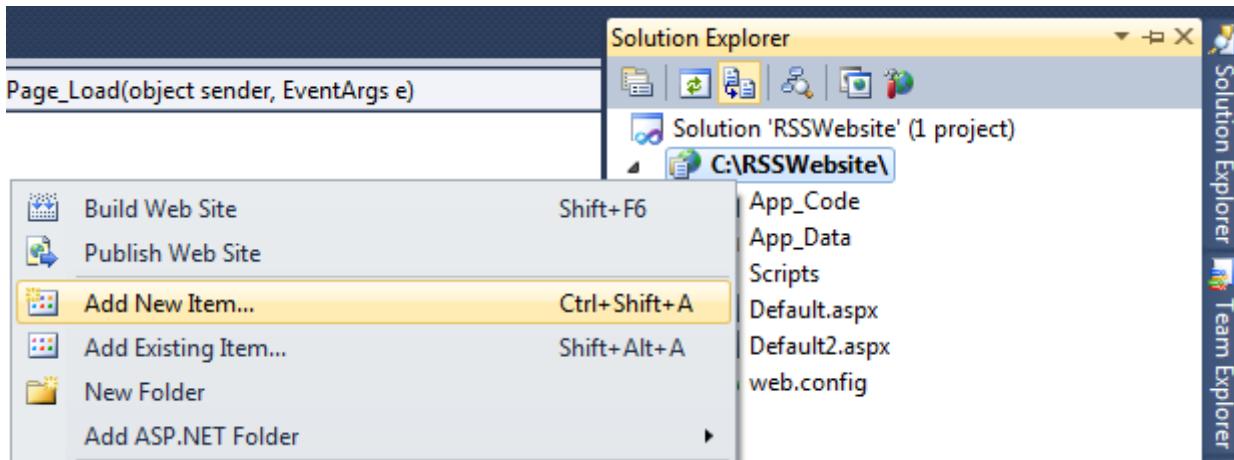
```

<rss version="2.0">
  <channel>
    <title>JACKYHUNG BLOG Website</title>
    <link>http://vn.360plus.yahoo.com/jackyhung12345</link>
    <description>Blog của Jacky Hưng</description>
    <generator>Your RSS Generator name and version. JACKYHUNG :)</generator>
    <item>
      <title>Tổng hợp tài liệu môn học</title>
      <link>http://vn.360plus.yahoo.com/jackyhung12345/article?mid=5</link>
      <description>Tổng hợp tài liệu các môn học của Jacky Hưng</description>
    </item>
  </channel>
</rss>

```

## Hướng dẫn lấy RSS từ các trang

**Bước 1:** Đầu tiên thêm trang aspx mới để lấy và hiển thị nội dung RSS



**Bước 2:** Trong trang design page GetRSS này bạn tạo ra một DataList để lưu các RSS lấy về được từ một link đến page RSS nào đó. Chẳng hạn như <http://vnexpress.net/rss/gl/xa-hoi.rss>

```

<body>
    <form id="form1" runat="server">
        <div>
            <asp:DataList ID="dlRSS" runat="server" Width="100%">
                <ItemTemplate>
                    <div class="RSSTitle">
                        <asp:HyperLink ID="TitleLink" runat="server" Text='<%# Eval("title") %>' NavigateUrl='<%# Eval("link") %>' />
                    </div>
                    <div class="RSSSubtitle">
                        <asp:Label ID="SubtitleLabel" runat="server" Text='<%# Eval("description") %>' />
                    </div>
                    <div class="RSSInfo">
                        posted on <asp:Label ID="DateRSSedLabel" runat="server" Text='<%# Eval("pubDate", "{0:d} @ {0:t}") %>' />
                    </div>
                </ItemTemplate>
            </asp:DataList>
        </div>
    </form>
</body>

```

DataList này bao gồm các thông tin: title, link, description và ngày post.

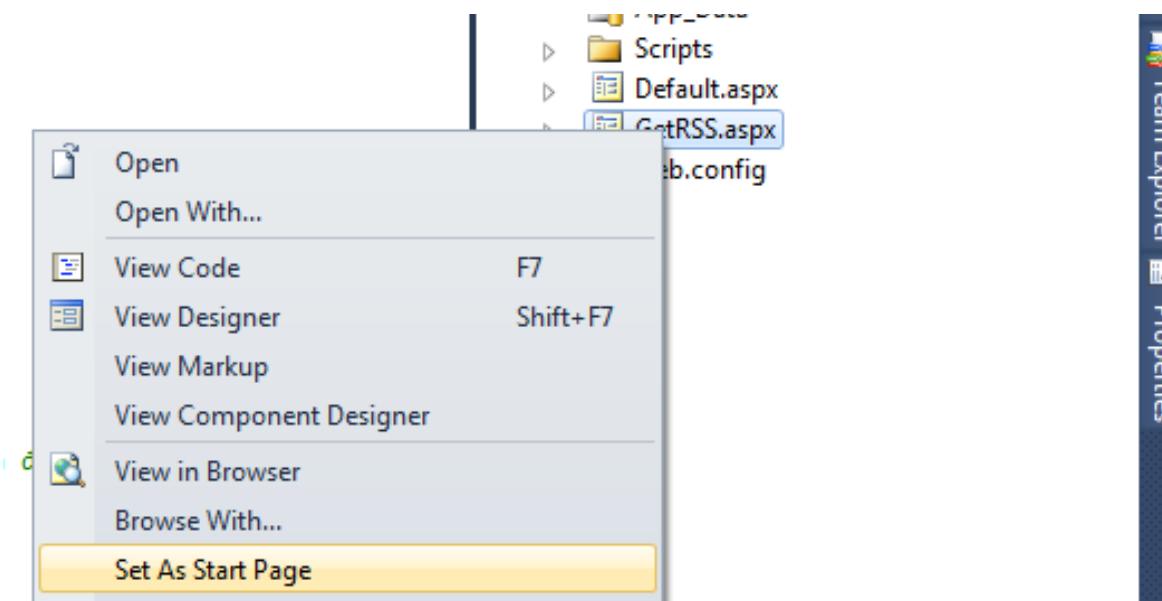
**Bước 3:** Trong trang GetRSS.aspx.cs thêm dòng code sau vào

```

6  using System.Web.UI.WebControls;
7  using System.Xml;
8  using System.Data;
9
10 public partial class GetRSS : System.Web.UI.Page
11 {
12     protected void Page_Load(object sender, EventArgs e)
13     {
14         XmlTextReader reader = null;
15         try
16         {
17             reader = new XmlTextReader("http://vnexpress.net/rss/gl/xa-hoi.rss");// đường dẫn đến nơi lấy rss
18             DataSet ds = new DataSet();
19             ds.ReadXml(reader);
20             dlRSS.DataSource = ds.Tables["item"];
21             dlRSS.DataBind();
22         }
23         catch (Exception ex)
24         {
25             Response.Write(ex.Message);
26         }
27         finally
28         {
29             reader.Close();
30         }
31     }
32 }

```

**Bước 3:** Click chuột phải vào GetRSS.aspx rồi chọn Set As Start Page.



Rồi bấm chạy trang web.

The screenshot shows a web browser window with the URL 'localhost:5678/RSSWebsite/GetRSS.aspx'. The page displays a list of news items:

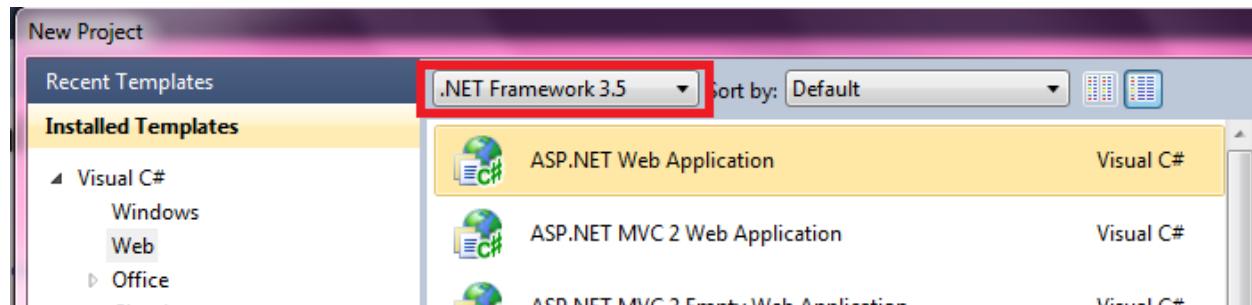
- Dêm 12/11, sau đại thắng 8-0 của U23 Việt Nam trước Brunei tại SEA Games 26, hàng nghìn người dân TP HCM mang theo cờ hoa dỗ về các tuyến đường trung tâm hò reo mừng chiến thắng.  
-> [Đại thắng 8-0, U23 Việt Nam giành lại ngôi đầu bảng B](#)  
posted on Sun, 13 Nov 2011 03:50:30 GMT @ Sun, 13 Nov 2011 03:50:30 GMT  
['Cái giá cho danh hiệu kỳ quan thế giới không hề cao'](#)
- Trao đổi với VnExpress chiều 12/11, ông Nguyễn Văn Tinh, Cục trưởng Hợp tác quốc tế (Bộ Văn hóa Thể thao và Du lịch) khẳng định, nếu so với các chương trình xúc tiến du lịch thì đợt quảng bá vịnh Hạ Long không tốn kém.  
-> [Vịnh Hạ Long vào top 7 kỳ quan thế giới mới](#)  
posted on Sun, 13 Nov 2011 02:52:29 GMT @ Sun, 13 Nov 2011 02:52:29 GMT  
[Thảo My giành ngôi vị Miss teen 2011](#)
- Tối 12/11, 20 thiếu nữ đã khoe vẻ duyên dáng, xinh đẹp cùng khả năng múa, hát, nhảy... trong đêm chung kết Miss teen tổ chức tại Tuần Châu (Quảng Ninh). Thảo My đã xuất sắc đoạt vương miện Miss teen 2001.  
-> [Nữ sinh Miss Teen tạo dáng trước biển Hạ Long](#)  
posted on Sun, 13 Nov 2011 07:10:21 GMT @ Sun, 13 Nov 2011 07:10:21 GMT  
[Người dân đỗ xô đăng ký dự thi tiếng Hàn](#)
- Hai ngày qua, người dân Nghệ An, Hà Tĩnh đổ đến Trung tâm giới thiệu việc làm tỉnh Nghệ An đăng ký dự thi tiếng Hàn với mong muốn được sang Hàn Quốc làm việc. Do lượng người quá đông nên đã xảy ra cảnh hỗn loạn.

At the bottom, there is a taskbar with icons for various applications and a system tray showing the date and time (11/13/2011, 11:06 PM).

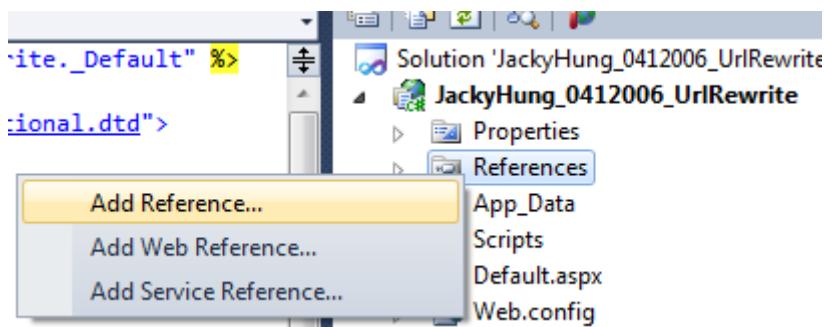
# HƯỚNG DẪN 6: URL REWRITE

Sử dụng asp.net routing trong NET 3.5.

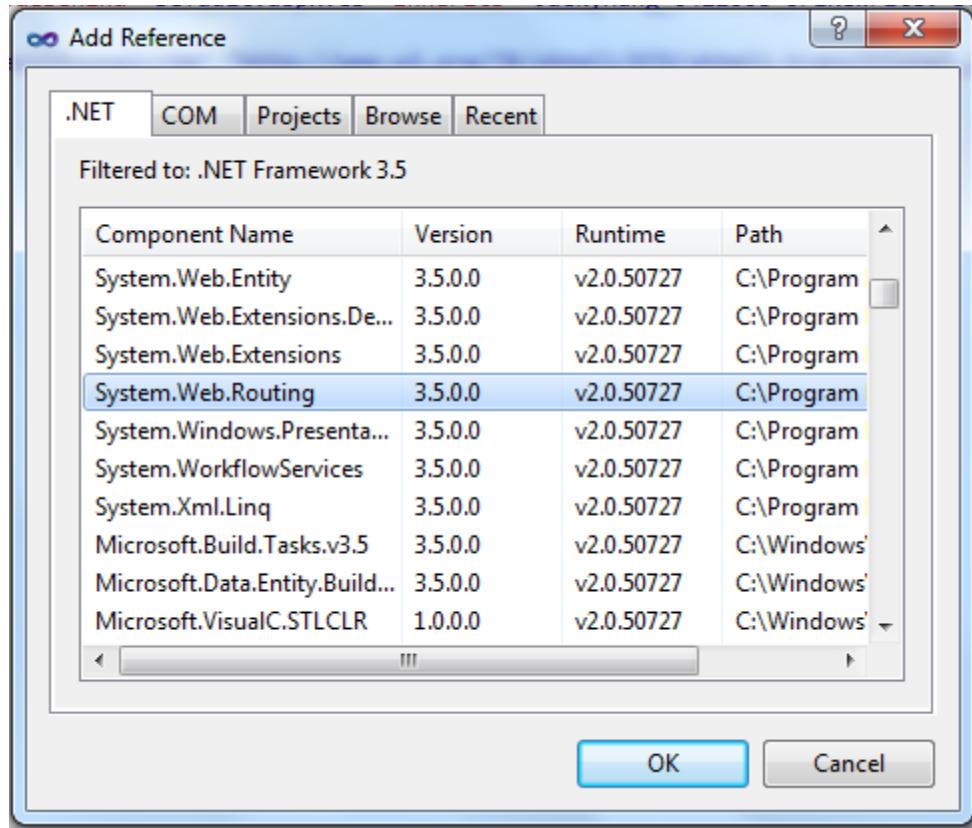
Bước 1: Tạo 1 website với framework 3.5.



Bước 2: Phải chuột vào Project chọn Add References:



Chọn thẻ .Net -> chọn System.web.routing->OK



**Bước 3:** Mở web.config tìm thẻ `<httpModules>` `</httpModules>` và thêm dòng code sau vào trong thẻ

```

<add
      name="RoutingModule"
      type="System.Web.Routing.UrlRoutingModule, System.Web.Routing, Version=3.5.0.0,
      Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>
  
```

Kết quả:

```

<httpHandlers>
  <remove verb="*" path="*.asmx"/>
  <add verb="*" path="*.asmx" validate="false" type="System.Web.Script.Ser...
  <add verb="*" path="*_AppService.axd" validate="false" type="System.Web.!
  <add verb="GET,HEAD" path="ScriptResource.axd" type="System.Web.Handlers...
</httpHandlers>
<httpModules>
  <add name="ScriptModule" type="System.Web.Handlers.ScriptModule, System.W...
  <add name="RoutingModule" type="System.Web.Routing.UrlRoutingModule, Syst...
</httpModules>
</system.web>
  
```

Hoặc

```

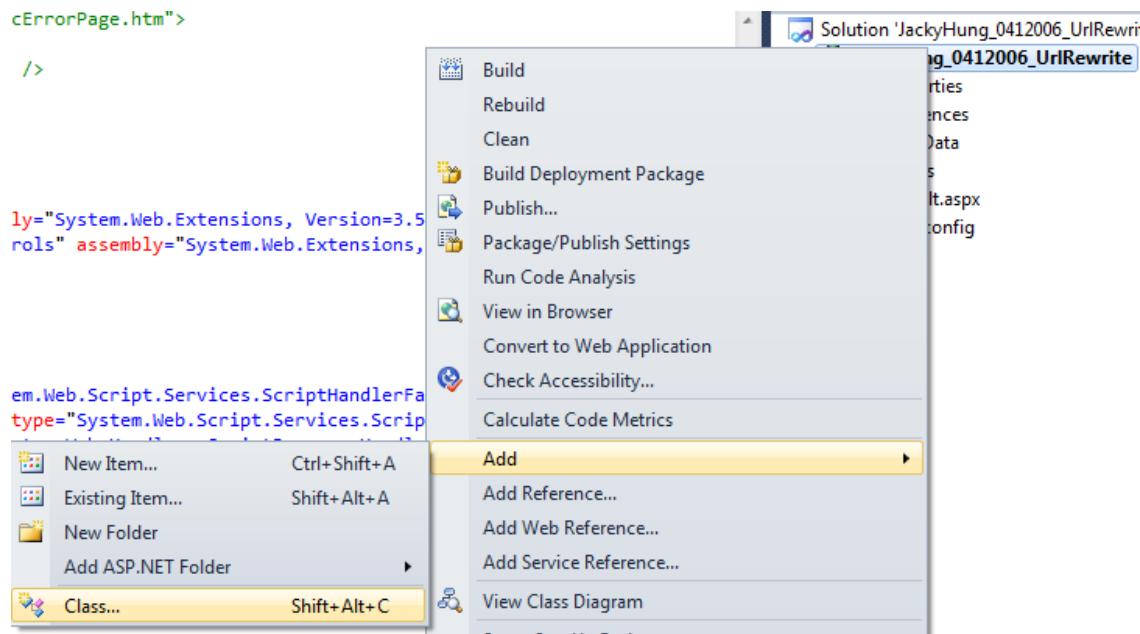
<system.web>
  <compilation debug="true" targetFramework="4.0" />
  <httpModules>
    <add name="RoutingModule" type="System.Web.Routing.UrlRoutingModule,
      System.Web.Routing, Version=3.5.0.0,
      Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>
  </httpModules>
</system.web>

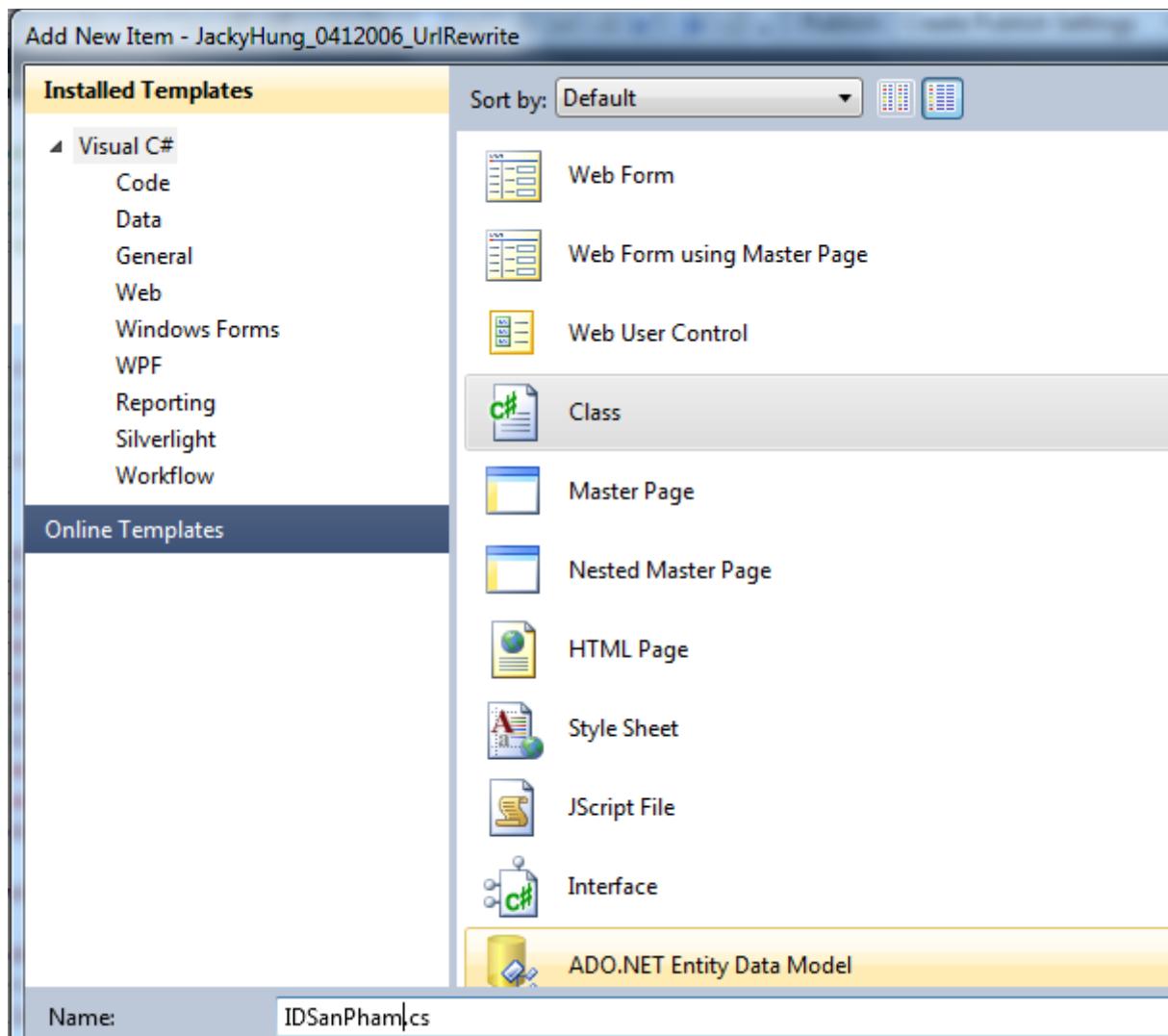
```

**Bước 4:** Rewrite page Default.aspx (không có tham số(querystring) )

Minh họa từ Default.aspx -> SanPham.html

Tạo lớp có tên là IDSanPham.cs





Chuyển thông tin lớp IDSanpham này thành lớp interface và kế thừa interface IHttpHandler như sau:

IDSanPham.cs\* X Web.config\* Default.aspx

JackyHung\_0412006.UrlRewrite.IDSanPham

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace JackyHung_0412006.UrlRewrite
{
    public interface IDSanPham : IHttpHandler
    {
    }
}

```

**Bước 5:** Tạo class SanPhamRouteHandler và code như sau

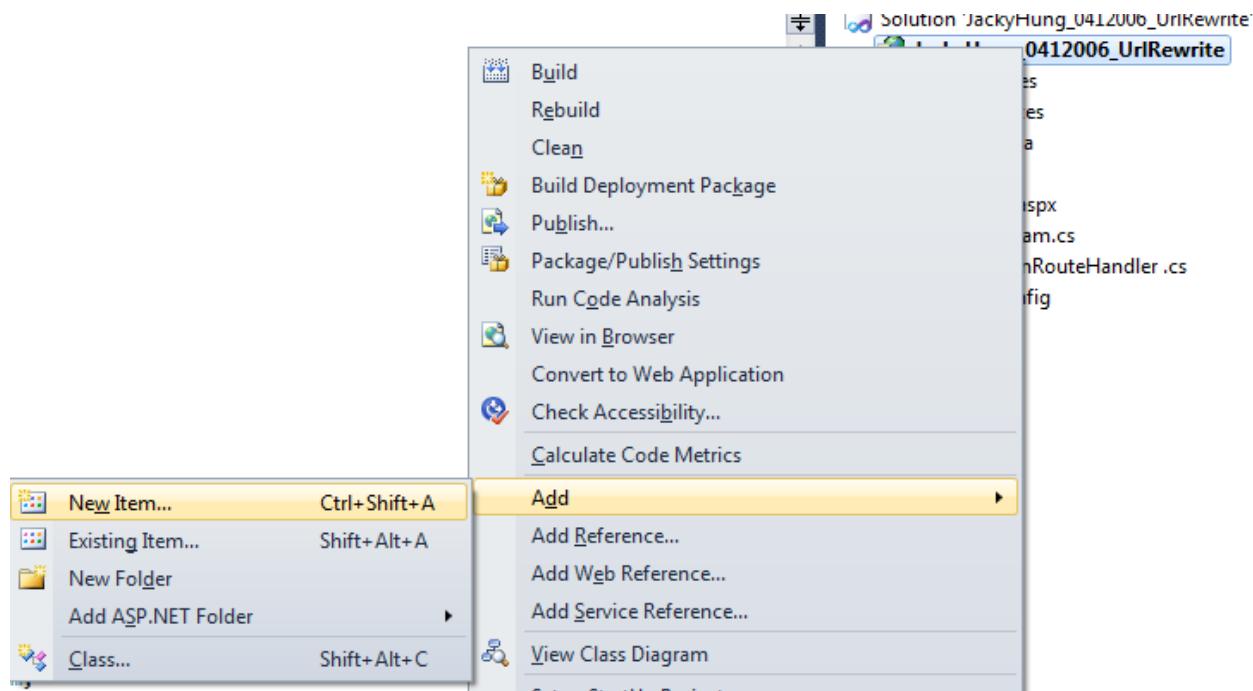
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Routing;
using System.Web.Compilation;
using System.Web.UI;

public class SanPhamRouteHandler: IRouteHandler //chú ý kế thừa lớp này
{
    public SanPhamRouteHandler()
    {
    }

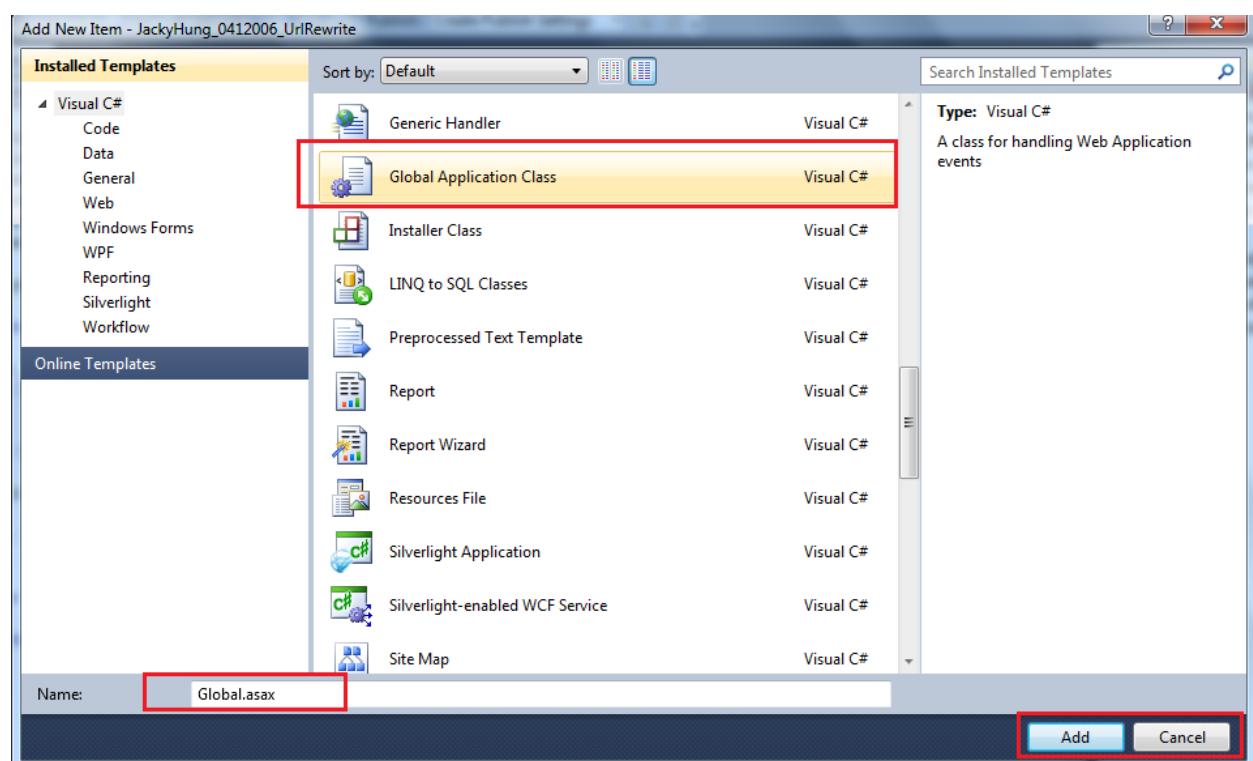
    string _virtualPath;
    public SanPhamRouteHandler(string virtualPath)
    {
        _virtualPath = virtualPath;
    }

    public IHttpHandler GetHttpHandler(RequestContext requestContext)
    {
        var display = BuildManager.CreateInstanceFromVirtualPath(
            _virtualPath, typeof(Page)) as IDSanPham;
        return display;
    }
}
```

**Bước 6:** Tạo thêm file Global.asax. Như sau click chuột phải project vào add new item.



Rồi sau đó chọn như hình:



Trong file Global.asax viết hàm RegisterRoutes và được gọi trong hàm Application\_Start như sau:

```

void Application_Start(object sender, EventArgs e)
{
    // Code that runs on application startup
    RegisterRoutes();
}

private static void RegisterRoutes()
{
    System.Web.Routing.RouteTable.Routes.Add(
        "SanPham", new System.Web.Routing.Route("SanPham.html",
            new SanPhamRouteHandler("~/Default.aspx")));
}

```

Tên routing(không  
được trùng khi đang  
ký cho những page)

Tên muốn hiển thị  
ra trình duyệt

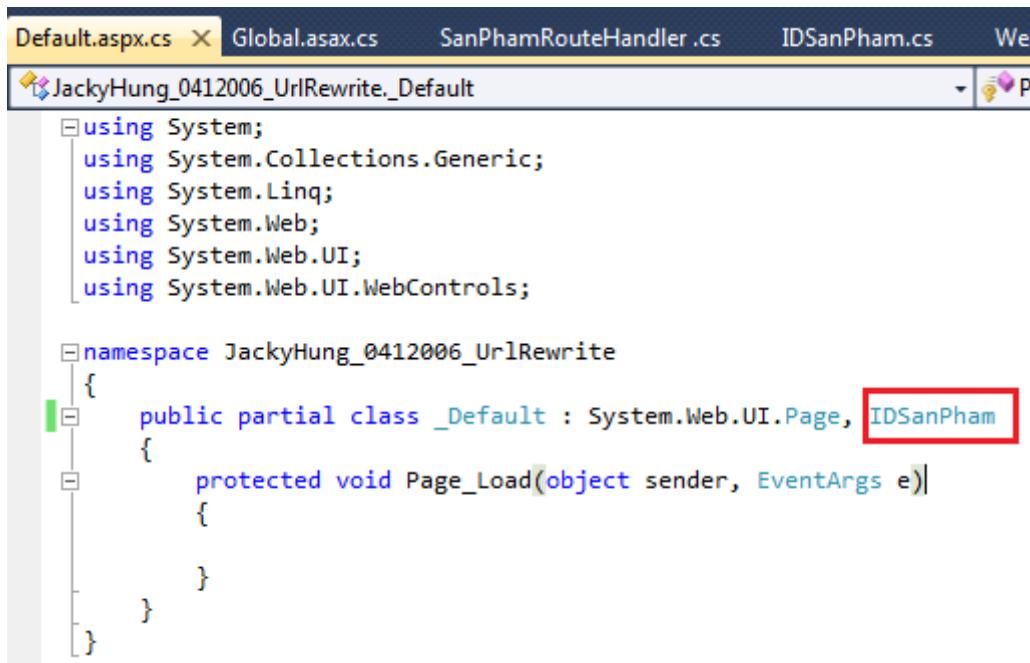
tên page cần  
rewrite

```

private static void RegisterRoutes()
{
    System.Web.Routing.RouteTable.Routes.Add(
        "SanPham", new System.Web.Routing.Route("SanPham.html",
            new SanPhamRouteHandler("~/Default.aspx")));
}

```

**Bước 7:** Mở trang default như sau:



```

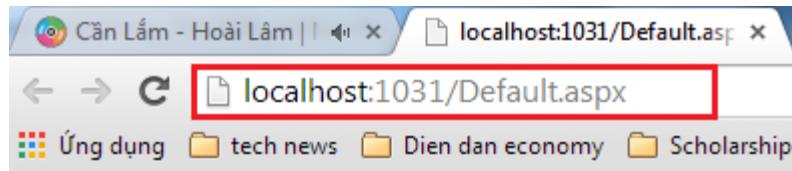
Default.aspx.cs X Global.asax.cs SanPhamRouteHandler.cs IDSanPham.cs Web
JackyHung_0412006.UrlRewrite.Default

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

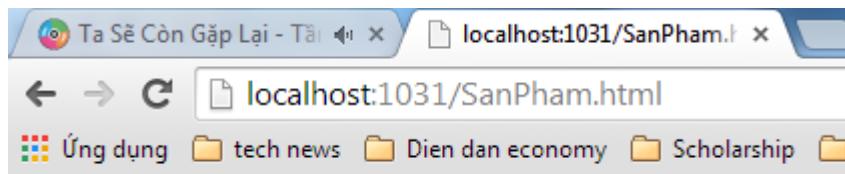
namespace JackyHung_0412006.UrlRewrite
{
    public partial class _Default : System.Web.UI.Page, IDSanPham
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }
    }
}

```

**Bước 8:** Xong bước này ta có thể chạy thử trang Default.aspx.



Bây giờ thay chữ default.aspx bằng SanPham.html.



Như vậy ta đã thực hiện xong quá trình rewrite url. Bây giờ ta tạo nội dung và làm thêm trang url rewrite cho các trang sản phẩm.

**Bước 9:** Tạo 1 database mẫu với 1 bảng là SanPham với 1 thuộc tính là MaSP (tự tăng kiểu int1) và TenSP (kiểu chuỗi nvarchar). Tự tạo dữ liệu mẫu trong bảng này.

**Bước 11:** Thiết kế lại trang Default.aspx như sau:

```

<body>
<form id="form1" runat="server">

```

```

<div>

<asp:Repeater ID="rptDemo" runat="server">

<HeaderTemplate>
</HeaderTemplate>
<ItemTemplate>
    <asp:HyperLink Text='<%#Eval("tensp") %>' ID="HyperLink1" runat="server"
        NavigateUrl='<%# "~/Chi-tiet-san-pham"+Eval("MaSP")+".html" %>'>
    </asp:HyperLink> //chú ý phần tó màu vàng
</ItemTemplate>
<FooterTemplate>
</FooterTemplate>
</asp:Repeater>
</div>
</form>
</body>

```

**Bước 12:** Hoàn thành code. Sửa hàm load của file Default.aspx.cs lại như sau:

```

public partial class _Default : System.Web.UI.Page, IDSanPham //chú ý
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (IsPostBack == false)
        {
            string connectionString =
                @"Server=.\sqlexpress;Database=Demo;Trusted_Connection=True";
            SqlConnection conn = new SqlConnection(connectionString);

```

```
SqlCommand command = new SqlCommand("select * from SanPham",  
conn);  
  
DataTable dt = new DataTable();  
  
try  
{  
    conn.Open();  
  
    dt.Load(command.ExecuteReader());  
  
    rptDemo.DataSource = dt;  
  
    rptDemo.DataBind();  
  
}  
  
catch (Exception ex)  
{  
    throw ex;  
}  
  
finally  
{  
    conn.Close();  
}  
}  
}  
}
```

**Bước 13:** Tao trang ChiTietSP.aspx và thiết kế như sau

```
<body>  
  
<form id="form1" runat="server">  
  
<div>
```

```
<asp:Repeater ID="rptDemoCT" runat="server">  
    <HeaderTemplate>  
    </HeaderTemplate>  
    <ItemTemplate>  
        Mã Sản phẩm : <a><%# Eval("MaSP") %></a><br />Tên Sản phẩm :  
        <a><%# Eval("tensp") %></a>  
    </ItemTemplate>  
    <FooterTemplate>  
    </FooterTemplate>  
</asp:Repeater>  
</div>  
</form>  
</body>
```

**Bước 14:** Rewrite Page ChiTietSP.aspx. Trang này nhận QuerString từ trang Default.aspx

Minh Họa ChiTietSP.aspx -> Chi-tiet-san-pham[MaSP].html(sau phần minh họa này chúng ta có thể reander theo luật riêng có cá biến của chúng ta chẳng hạn như [Tên sản phẩm]/[id] )

Tạo lớp interface có tên là IDChiTietSP.cs và code như sau

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

/// <summary>
/// Summary description for IDChiTietSP
/// </summary>
public interface IDChiTietSP : IHttpHandler
{
    string MaSP { get; set; }
```

Có nhiều biến thì tạo  
nhiều property tương ứng

**Bước 15:** Tạo lớp ChiTietSPRouteHandler.cs và code như sau

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Routing;
using System.Web.Compilation;
using System.Web.UI;
public class ChiTietSPRouteHandler : IRouteHandler
{
    public ChiTietSPRouteHandler()
    {
    }
    string _virtualPath;
    public ChiTietSPRouteHandler(string virtualPath)
    {
        _virtualPath = virtualPath;
    }
}
```

```
public IHttpHandler GetHttpHandler(RequestContext requestContext)
{
    if (requestContext.RouteData.Values["MaSP"] != null)
    {
        var display = BuildManager.CreateInstanceFromVirtualPath(
            _virtualPath, typeof(Page)) as IDChiTietSP;
        if (display != null)
        {
            display.MaSP = requestContext.RouteData.Values["MaSP"] as string;
            return display;
        }
        else
        {
            return null;
        }
    }
    else
    {
        return null;
    }
}
```

**Bước 16:** Giống như trang Default.aspx ta cũng phải đăng ký routing để rewrite

```

void Application_Start(object sender, EventArgs e)
{
    // Code that runs on application startup
    RegisterRoutes();
}

private static void RegisterRoutes()
{
    System.Web.Routing.RouteTable.Routes.Add(
        "SanPham", new System.Web.Routing.Route("SanPham.html",
        new SanPhamRouteHandler("~/Default.aspx")));
    Phải khác nhau không
    // được trùng
    System.Web.Routing.RouteTable.Routes.Add("Chi-tiet-san-pham",
        new System.Web.Routing.Route("Chi-tiet-san-pham{MaSP}.html",
        new ChiTietSPRouteHandler("~/ChiTietSP.aspx")))
    );
}

```

**Bước 17:** Hoàn thành code. Sửa hàm load của trang ChiTietSP.aspx.cs lại như sau:

```

public partial class ChiTietSP : System.Web.UI.Page, IDChiTietSP //chú ý

{
    public string MaSP { set; get; }

    protected void Page_Load(object sender, EventArgs e)
    {
        if (IsPostBack == false)
        {
            string connectionString =
                @"Server=.\sqlexpress;Database=Demo;Trusted_Connection=True";
            SqlConnection conn = new SqlConnection(connectionString);
            string query = "select * from SanPham where MaSP=" + MaSP + "";
            SqlCommand command = new SqlCommand(query, conn);

```

```
DataTable dt = new DataTable();

try
{
    conn.Open();
    dt.Load(command.ExecuteReader());
    rptDemoCT.DataSource = dt;
    rptDemoCT.DataBind();
}

catch (Exception ex)
{
    throw ex;
}

finally
{
    conn.Close();
}

}
```

**Bước 18:** Chạy và xem kết quả.

## Sử dụng thư viện routing trong .NET

**Bước 1:** Tạo 1 website mới.

Vào đường dẫn <http://www.urlrewriting.net/159/en/downloads.html> down thư viện UrlRewritingNet.UrlRewriter.dll và đặt ở trong thư mục Bin của project trên.

Tạo 1 trang Default.aspx và thiết kế như sau:

```
<body>

    <form id="form1" runat="server">

        <div>

            <asp:Repeater ID="rptDemo" runat="server">

                <HeaderTemplate>

                    </HeaderTemplate>

                <ItemTemplate>

                    <asp:HyperLink Text='<%#Eval("tensp") %>' ID="HyperLink1" runat="server"
NavigateUrl=""> </asp:HyperLink>

                </ItemTemplate>

                <FooterTemplate>

                    </FooterTemplate>

                </asp:Repeater>

            </div>

        </form>

    </body>
```

**Bước 2:** Tao trang ChiTietSP.aspx và thiết kế như sau

```
<body>

    <form id="form1" runat="server">

        <div>

            <asp:Repeater ID="rptDemoCT" runat="server">

                <HeaderTemplate>

                    </HeaderTemplate>

                <ItemTemplate>
```

```

Mã Sản phẩm : <a><%# Eval("MaSP") %></a><br />Tên Sản phẩm :
<a><%# Eval("tensp") %></a>

</ItemTemplate>

<FooterTemplate>

</FooterTemplate>

</asp:Repeater>

</div>

</form>

</body>

```

**Bước 3:** Cấu hình cho web.config. Tìm lần lượt các thẻ sau và chép đoạn code tương ứng sau và trong thẻ nếu chưa có thì thêm thẻ đó vào.

Code sau nằm trong thẻ này <configuration></configuration>

```

<urlrewritingnet rewriteOnlyVirtualUrls="true" contextItemsPrefix="QueryString"
defaultProvider="RegEx"
xmlns="http://www.urlrewriting.net/schemas/config/2006/07">

<rewrites>

</rewrites>

</urlrewritingnet>

```

Code sau nằm trong thẻ này <configSections> </configSections>

```

<section name="urlrewritingnet" requirePermission="false"
type="UrlRewritingNet.Configuration.UrlRewriteSection,UrlRewritingNet.UrlRewriter"/>

```

Code sau nằm trong thẻ này <httpModules></httpModules>

```

<add name="UrlRewriteModule" type="UrlRewritingNet.Web.UrlRewriteModule,
UrlRewritingNet.UrlRewriter" />

```

Code sau nằm trong thẻ này <modules></modules>

```
<add name="UrlRewriteModule" type="UrlRewritingNet.Web.UrlRewriteModule,
UrlRewritingNet.UrlRewriter"/>
```

Kết quả được như sau:

```

<configuration>
  <urlrewritingnet rewriteOnlyVirtualUrls="true" contextItemsPrefix="QueryString"
    defaultProvider="RegEx"
    xmlns="http://www.urlrewriting.net/schemas/config/2006/07">
    <rewrites>
    </rewrites>
  </urlrewritingnet>

  <configSections>
    <section name="urlrewritingnet" requirePermission="false"
      type="UrlRewritingNet.Configuration.UrlRewriteSection,
      UrlRewritingNet.UrlRewriter"/>
  </configSections>
  <system.web>
    <compilation debug="true" targetFramework="4.0" />
    <httpModules>
      <add name="UrlRewriteModule" type="UrlRewritingNet.Web.UrlRewriteModule,
        UrlRewritingNet.UrlRewriter" />
    </httpModules>
  </system.web>

  <system.webServer>
    <modules>
      <add name="UrlRewriteModule" type="UrlRewritingNet.Web.UrlRewriteModule,
        UrlRewritingNet.UrlRewriter"/>
    </modules>
  </system.webServer>
</configuration>

```

#### Bước 4: Thực hiện url Rewrite:

Ta có 2 page:

- Page Default.aspx hiển thị tên tất cả sản phẩm có trong database.
- Trang ChiTietSP.aspx để xem thông tin chi tiết từ trang Default.aspx sang.



Vào dòng code chú ý ở bước thứ 2 ở thẻ `<rewrites> </rewrites>` (Web.config) thiết các luật cần Rewrite như sau:

```
<add name="product" virtualUrl="~/Chi-tiet-san-pham(.*).html"
      rewriteUrlParameter="ExcludeFromClientQueryString"
      destinationUrl="~/ChiTietSP.aspx?MaSP=$1" ignoreCase="true"/>
```

(.\*) : Đại diện cho 1 biến truyền vào ( nếu truyền 2 biến thì làm 2 lần vd (.\*)(.\*) )

\$1: Đại diện cho QueryString (Nếu có 2 biến thì tương ứng SanPham.aspx?Ma1=\$1& Ma2=\$2

name: là duy nhất nếu làm cho trang khác thì phải đặt tên khác

**Bước 5:** Vào trang Default.aspx tìm và sửa lại dòng code sau

```
<asp:HyperLink Text='<%#Eval("tensp") %>' ID="HyperLink1" runat="server"
      NavigateUrl='<%# "~/Chi-tiet-san-pham"+Eval("MaSP")+".html" %>'>
</asp:HyperLink>
```

**Bước 6:** Hoàn thành code. Sửa hàm load ở file Default.aspx.cs như sau:

```
public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (IsPostBack == false)
        {
            string connectionString =
                @"Server=.\sqlexpress;Database=Demo;Trusted_Connection=True";
            SqlConnection conn = new SqlConnection(connectionString);
            SqlCommand command = new SqlCommand("select * from SanPham",
                conn);
            DataTable dt = new DataTable();
            try
            {
```

```
conn.Open();

dt.Load(command.ExecuteReader());

rptDemo.DataSource = dt;

rptDemo.DataBind();

}

catch (Exception ex)

{

    throw ex;

}

finally

{

    conn.Close();

}

}

}

}
```

**Bước 7:** Hoàn thành code. Sửa hàm load ở file ChiTietSP.aspx.cs như sau

```
public partial class ChiTietSP : System.Web.UI.Page

{

    protected void Page_Load(object sender, EventArgs e)

    {

        if (IsPostBack == false)

        {

            string connectionString = 

                @"Server=.\sqlexpress;Database=Demo;Trusted_Connection=True";
```

```
SqlConnection conn = new SqlConnection(connectionString);

    string query      = "select * from SanPham where
MaSP="+int.Parse(Request.QueryString["MaSP"])+"";

    SqlCommand command = new SqlCommand(query, conn);

    DataTable dt = new DataTable();

    try

    {

        conn.Open();

        dt.Load(command.ExecuteReader());

        rptDemoCT.DataSource = dt;

        rptDemoCT.DataBind();

    }

    catch (Exception ex)

    {

        throw ex;

    }

    finally

    {

        conn.Close();

    }

}

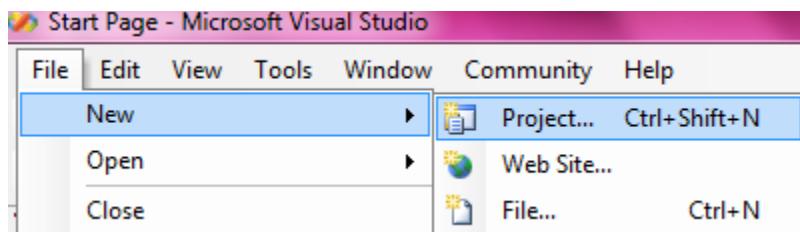
}
```

# HƯỚNG DẪN 7: TẠO WINDOWS SERVICE VÀ CÀI ĐẶT BẰNG CMD

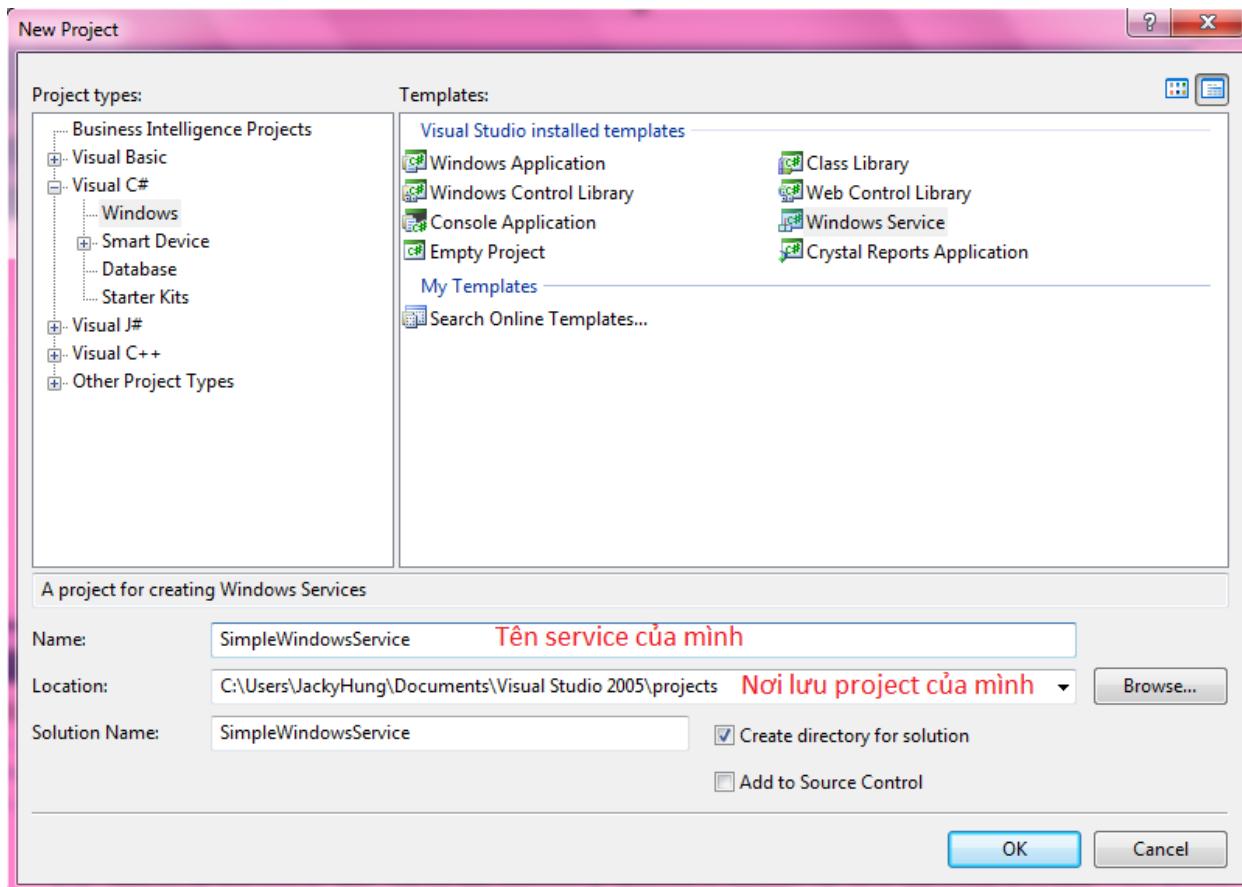
## Tạo windows service

**Bước 1:** Đầu tiên mở chương trình Microsoft Visual Studio lên.

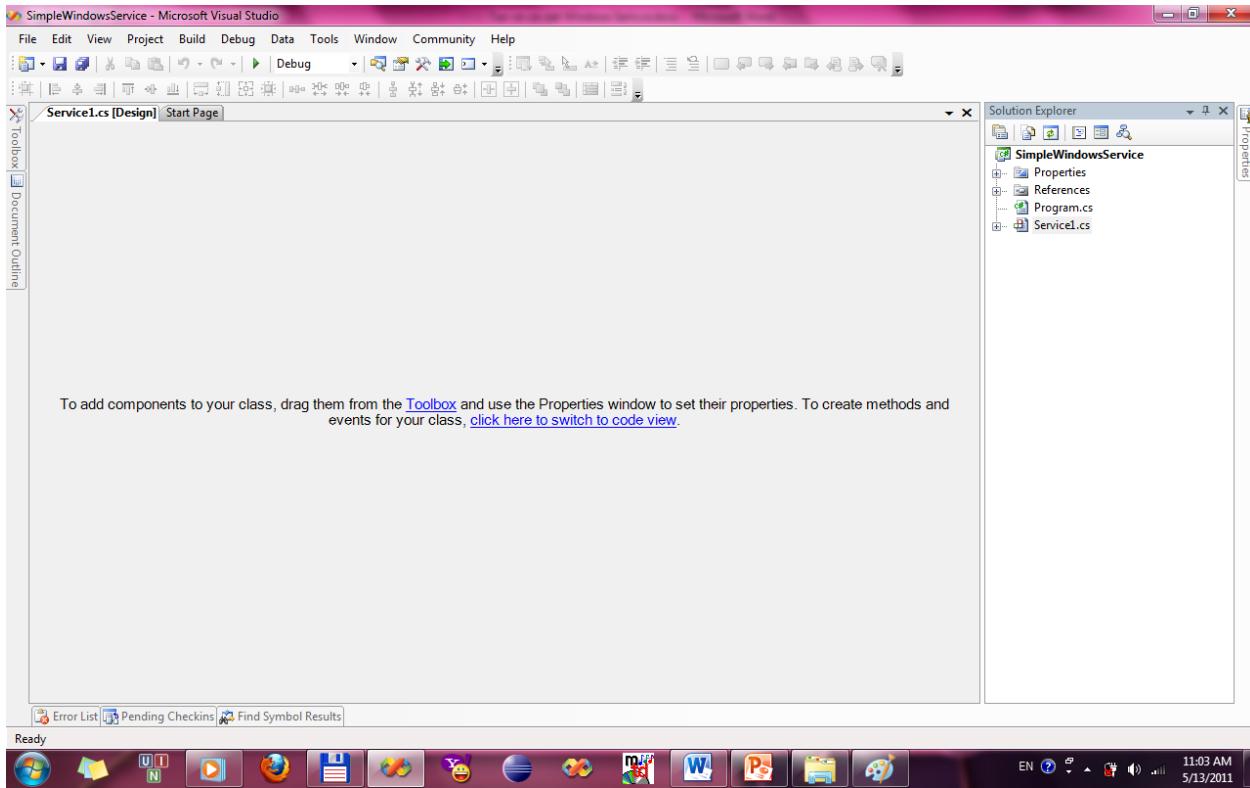
**Bước 2:** Tạo một project mới.



**Bước 3:** Trong Project Types chọn Visual C# → Windows. Bên khung Template chọn Windows Service.

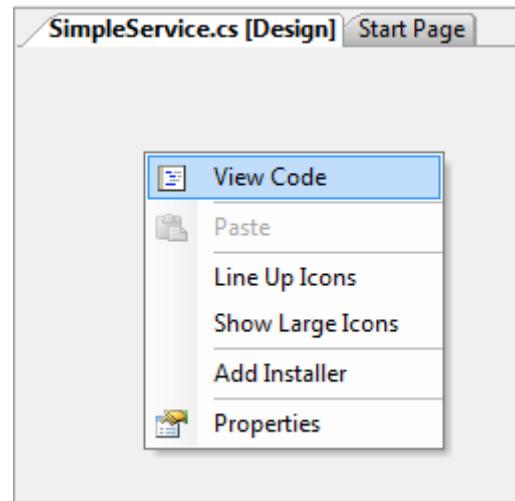


Bấm Ok. Microsoft Visual Studio sẽ tạo ra một project dạng như sau



Ở đây chương trình sẽ tạo ra một service có tên là service1. Nếu bạn muốn tên service khác thì click phải chuột vào Service1.cs trong khung Solution Explorer rồi chọn rename. Sau đó gõ tên service của bạn vào và bấm enter. Ở đây đổi sang là SimpleService.

**Bước 4:** Trong khung SimpleService bạn click chuột phải và chọn view code



Thì sẽ hiển thị ra source code của file form design này

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.ServiceProcess; Khai báo thư viện này để sử dụng dịch vụ windows service
using System.Text;

namespace SimpleWindowsService
{
    public partial class SimpleService : ServiceBase Các service thường kế thừa từ lớp Service Base
    {
        public SimpleService()
        {
            InitializeComponent();
        }

        protected override void OnStart(string[] args) Hàm OnStart này được gọi thực thi
        {
            // TODO: Add code here to start your service.
        }

        protected override void OnStop() Hàm OnStop này sẽ được gọi thực thi
        {
            // TODO: Add code here to perform any tear-down necessary to stop your service.
        }
    }
}

```

**Bước 5:** Như phân tích ở trên. Bây giờ ta sẽ làm một vài điều khi service chúng ta Start hay Stop nhé. Chẳng hạn như:

Khi service này start thì bạn mở một cái file C:\SimpleService.txt và ghi vào thông tin "SimpleService: Service Started". Nếu file C:\SimpleService.txt chưa tồn tại thì tạo ra và ghi vào.

```

protected override void OnStart(string[] args)

{
    // TODO: Add code here to start your service.

    FileStream fs = new
    FileStream(@"C:\SimpleService.txt", FileMode.OpenOrCreate, FileAccess.Write);

    StreamWriter m_streamWriter = new StreamWriter(fs);

    m_streamWriter.BaseStream.Seek(0, SeekOrigin.End);

    m_streamWriter.WriteLine("SimpleService: Service Started\n");

    m_streamWriter.Flush();

    m_streamWriter.Close();
}

```

Khi service này stop thì bạn mở một cái file C:\SimpleService.txt và ghi vào thông tin "SimpleService: Service Stopped". Nếu file C:\SimpleService.txt chưa tồn tại thì tạo ra và ghi vào.

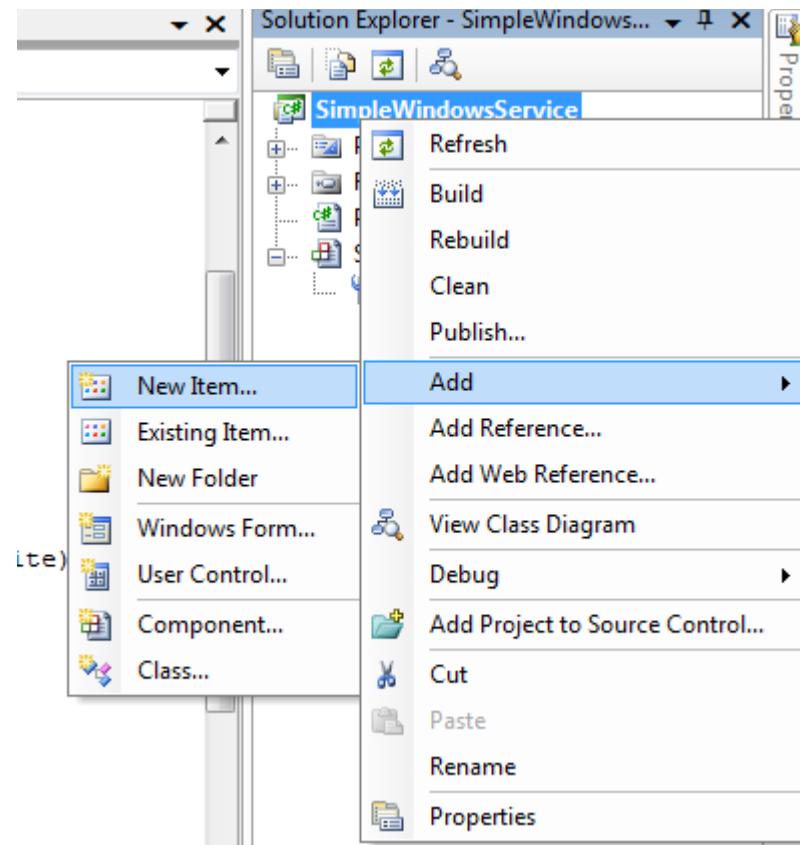
```
protected override void OnStop()
{
    // TODO: Add code here to perform any tear-down necessary to stop your
    // service.

    FileStream fs = new FileStream(@"C:\SimpleService.txt",
        FileMode.OpenOrCreate, FileAccess.Write);

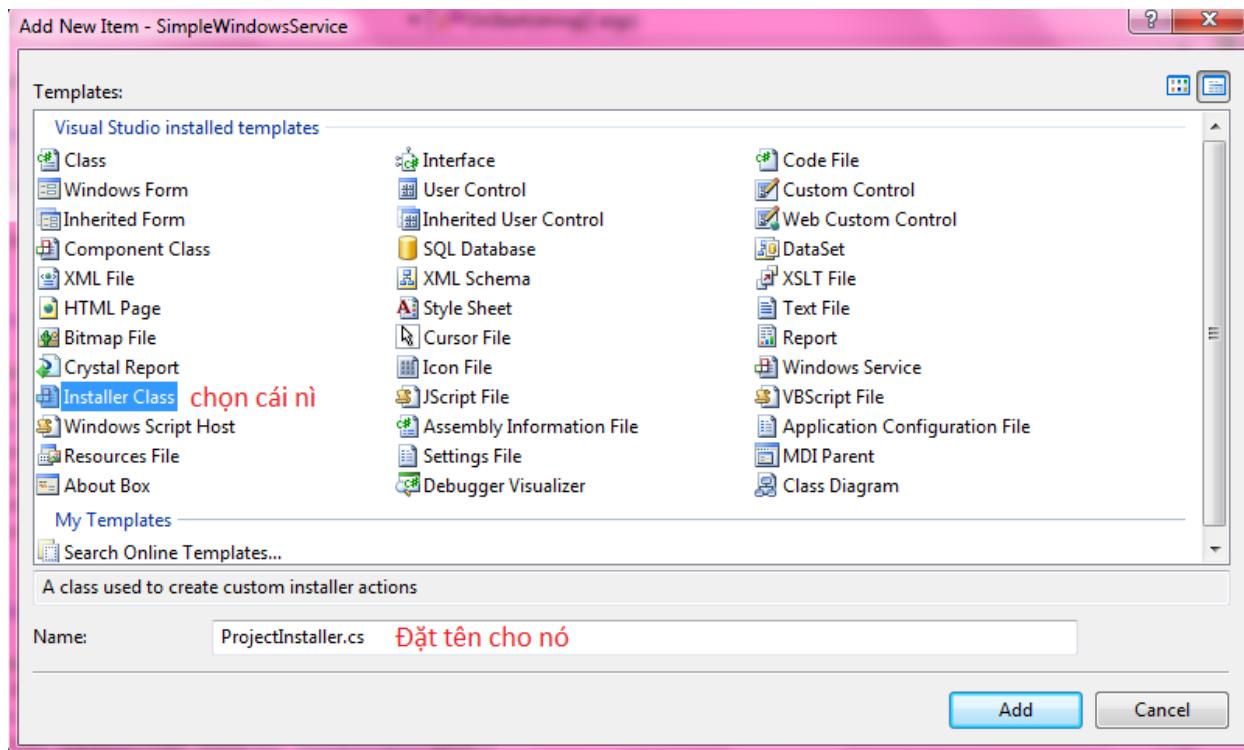
    StreamWriter m_streamWriter = new StreamWriter(fs);
    m_streamWriter.BaseStream.Seek(0, SeekOrigin.End);
    m_streamWriter.WriteLine("SimpleService: Service Stopped\n");
    m_streamWriter.Flush();
    m_streamWriter.Close();
}
```

**Bước 6:** Bạn tạo ra một installer để thực hiện việc cài đặt service của bạn vào hệ điều hành windows.

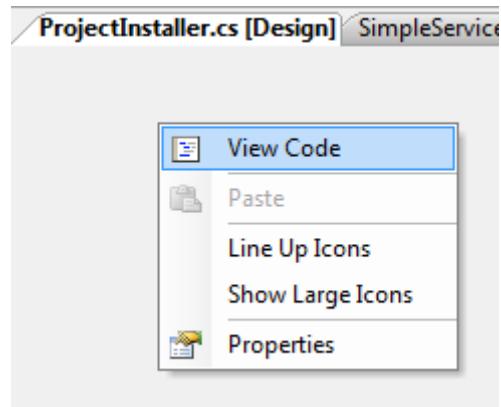
Click chuột phải vào SimpleWindowsService bên mục Solution Explorer. Chọn Add → New Item.



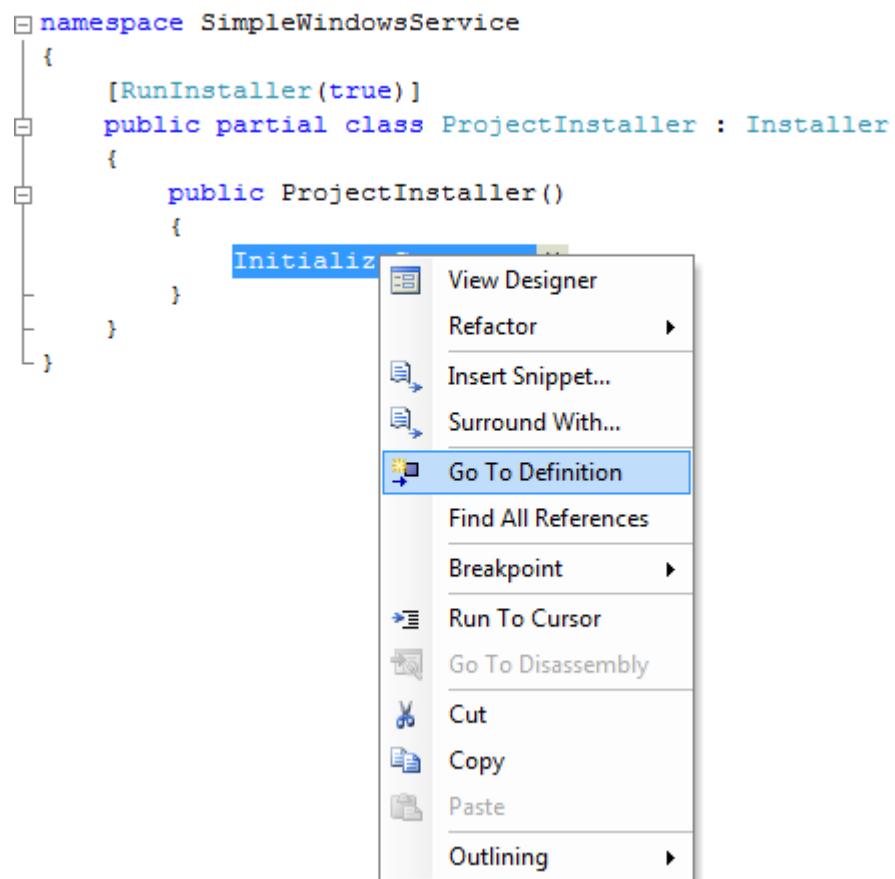
Sau đó một dialog hiện ra làm như sau



Rồi bấm Add. Microsoft sẽ tạo ra một form để design, bạn click chuột phải vào form đó rồi chọn View Code



Sau đó bạn click đúp chuột trái vào chữ InitializeComponent. Rồi click chuột phải và chọn Go To Definition.



Tại khung code bạn tiến hành thêm như sau

```

    #region Component Designer generated code

    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        components = new System.ComponentModel.Container();
    }
    private System.ServiceProcess.ServiceProcessInstaller serviceProcessInstallerSimple;
    private System.ServiceProcess.ServiceInstaller serviceInstallerSimple;

    #endregion
}

```

thêm vào

```

private                                         System.ServiceProcess.ServiceProcessInstaller
serviceProcessInstallerSimple;

private System.ServiceProcess.ServiceInstaller serviceInstallerSimple;

```

Trong hàm InitializeComponent thêm đoạn code sau vào

```

private void InitializeComponent()
{
    components = new System.ComponentModel.Container();

    this.serviceProcessInstallerSimple = new System.ServiceProcess.ServiceProcessInstaller();
    this.serviceInstallerSimple = new System.ServiceProcess.ServiceInstaller();
    //
    // serviceProcessInstallerSimple
    //
    this.serviceProcessInstallerSimple.Account = System.ServiceProcess.ServiceAccount.LocalSystem;
    this.serviceProcessInstallerSimple.Password = null;
    this.serviceProcessInstallerSimple.Username = null;
    //
    // serviceInstallerSimple
    //
    this.serviceInstallerSimple.ServiceName = "SimpleService";
    this.serviceInstallerSimple.StartType = System.ServiceProcess.ServiceStartMode.Automatic;
    //
    // ProjectInstaller
    //
    this.Installers.AddRange(new System.Configuration.Install.Installer[] {
        this.serviceInstallerSimple,
        this.serviceProcessInstallerSimple});
}

private System.ServiceProcess.ServiceProcessInstaller serviceProcessInstallerSimple;
private System.ServiceProcess.ServiceInstaller serviceInstallerSimple;

```

<pre> this.serviceProcessInstallerSimple System.ServiceProcess.ServiceProcessInstaller(); </pre>	= new
<pre> this.serviceInstallerSimple = new System.ServiceProcess.ServiceInstaller(); // // serviceProcessInstallerSimple </pre>	

```

// =  

this.serviceProcessInstallerSimple.Account  

System.ServiceProcess.ServiceAccount.LocalSystem;  

this.serviceProcessInstallerSimple.Password = null;  

this.serviceProcessInstallerSimple.Username = null;  

//  

// serviceInstallerSimple  

//  

this.serviceInstallerSimple.ServiceName = "SimpleService";  

this.serviceInstallerSimple.StartType =  

System.ServiceProcess.ServiceStartMode.Automatic;  

//  

// ProjectInstaller  

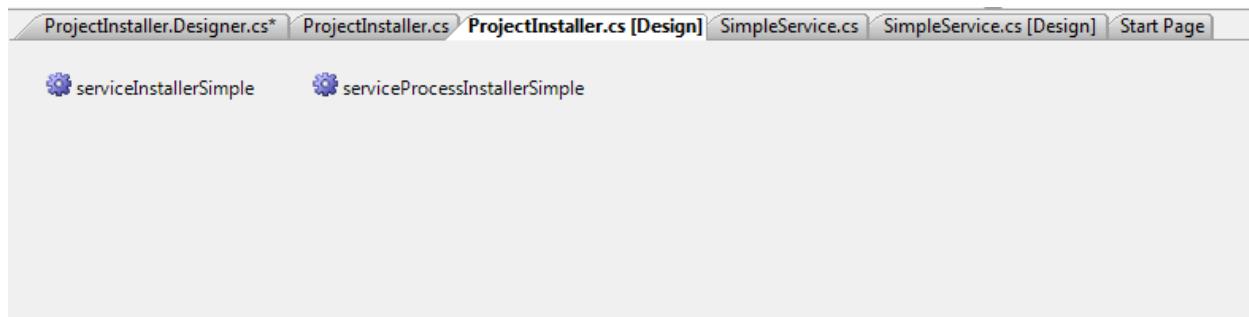
//  

this.Installers.AddRange(new System.Configuration.Install.Installer[] {  

this.serviceInstallerSimple,  

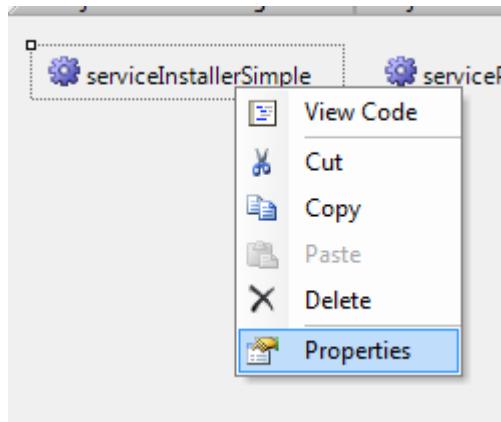
this.serviceProcessInstallerSimple});
```

Sau khi làm xong bạn qua cái form ProjectInstaller.cs[Design] bạn sẽ thấy 2 cái control là simpleInstallerSimple và serviceProcessInstallerSimpler

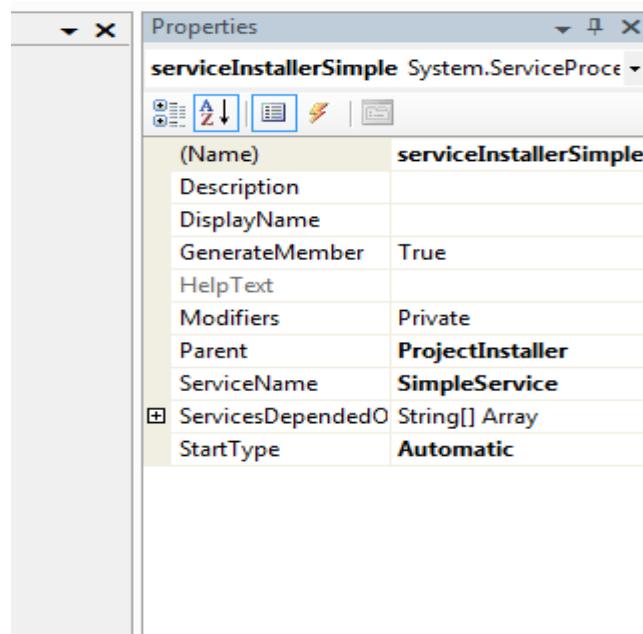


### Bước 7: cấu hình

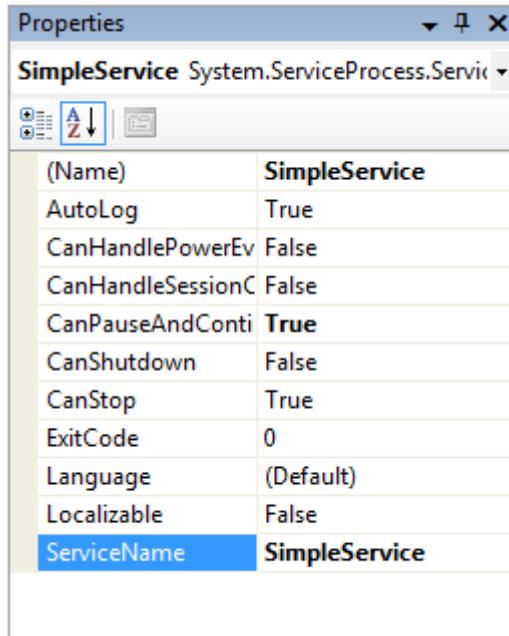
Tại control serviceInstallerSimple bạn click chuột phải chọn Properties.



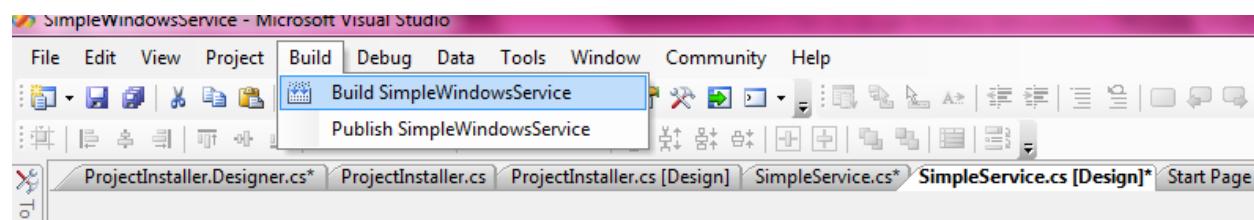
Tại khung Properties bạn chỉnh lại như sau:



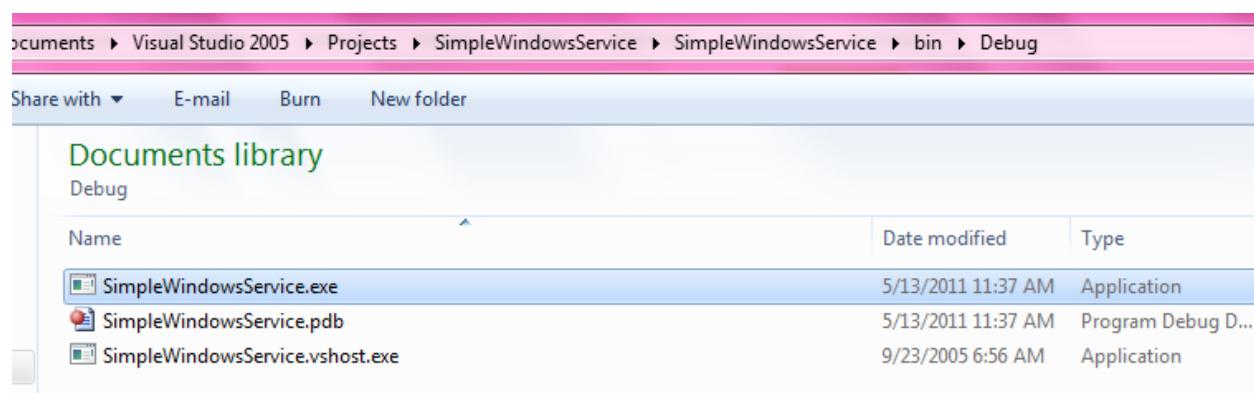
Bạn qua SimpleService.cs[Design], cũng chọn properties rồi chỉnh lại như sau



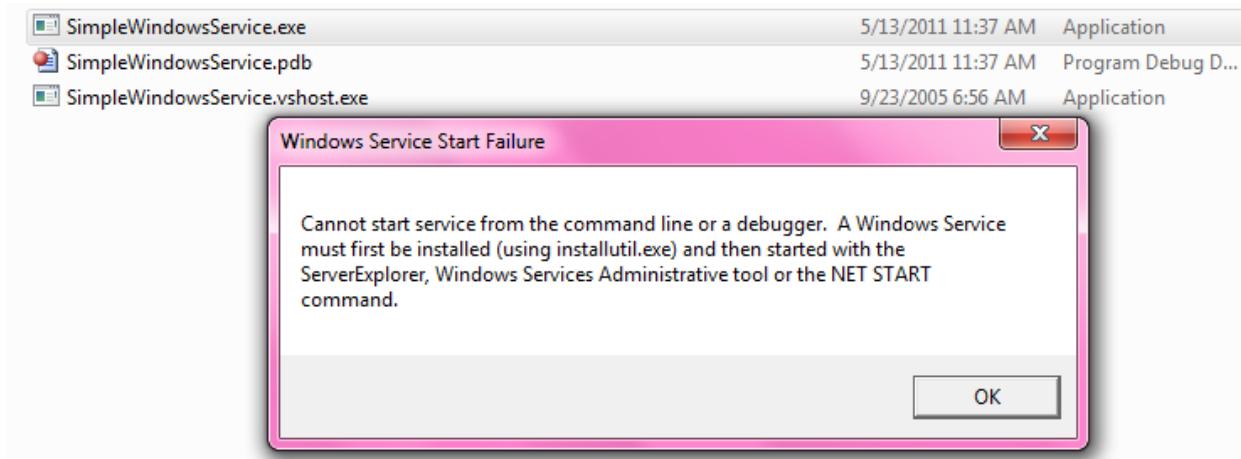
Tới đây bạn vào Build → chọn Build SimpleWindowsService



Sau khi build xong bạn sẽ được một file SimpleWindowsService.exe ở thư mục debug nơi lưu project của các bạn



Bạn click đúp chạy file SimpleWindowsService.exe đó thì sẽ bị báo lỗi như sau:



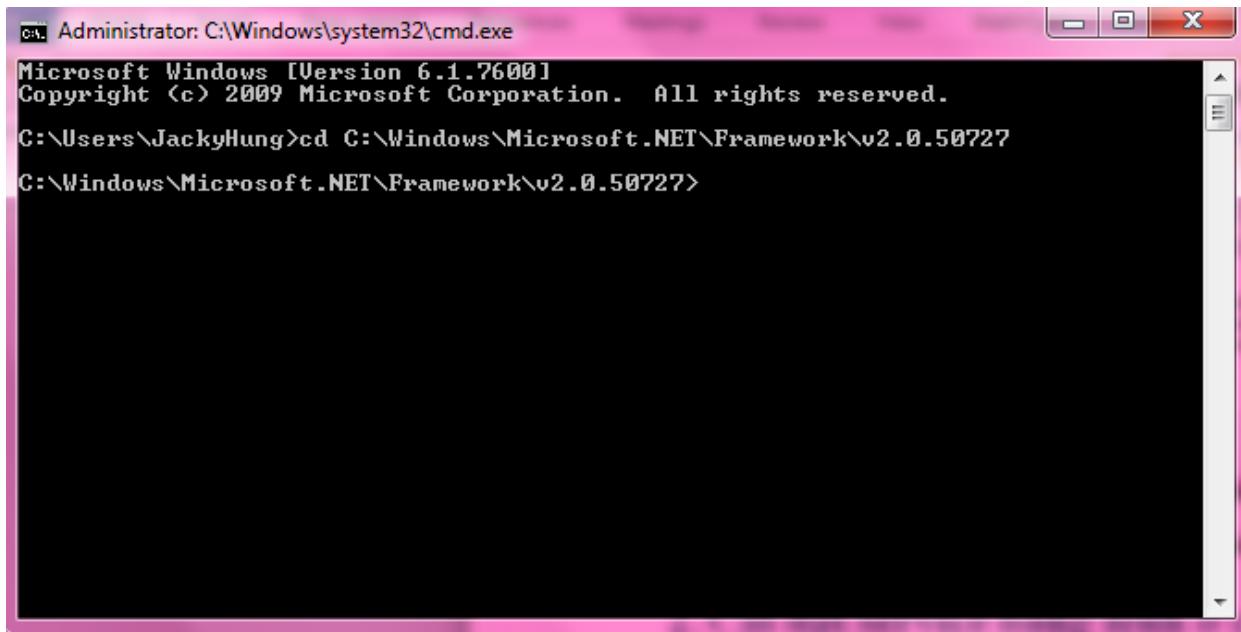
Lý do nó không tìm thấy file installutil.exe để tiến hành cài đặt

## Cài đặt service bằng lệnh ở cmd

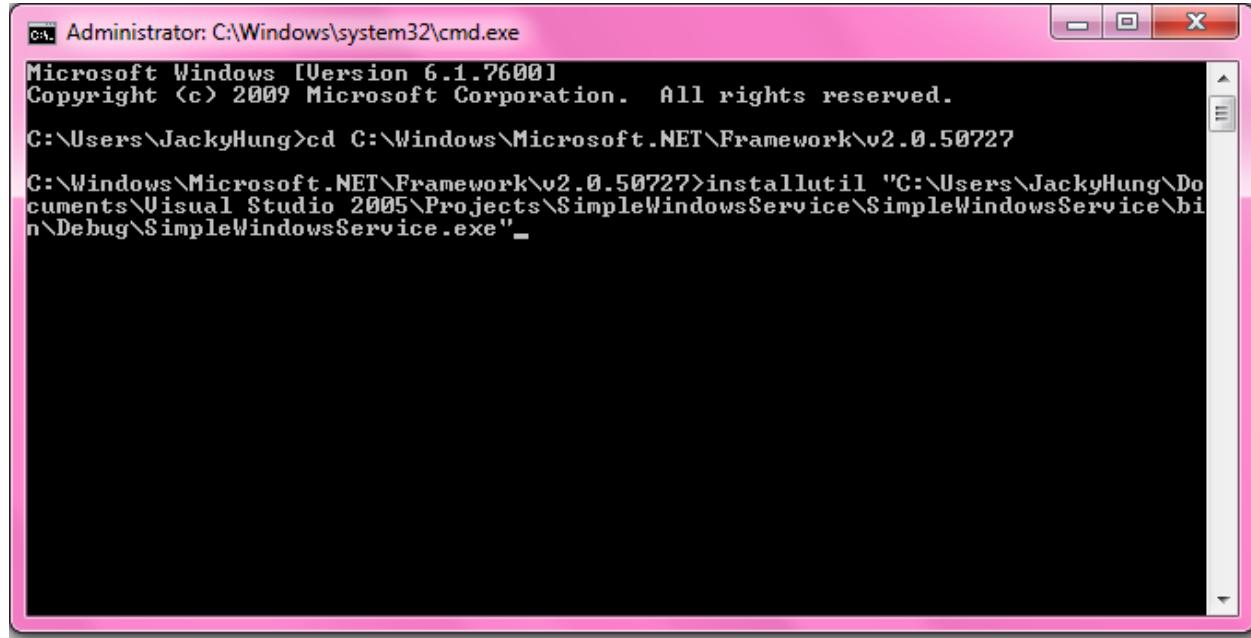
Trên máy của bạn file installutil.exe nó nằm ở thư mục này  
C:\Windows\Microsoft.NET\Framework\v2.0.50727

Do đó ta tiến hành cài đặt như sau

**Bước 1:** mở cmd lên rồi cd chuyển đến thư mục  
C:\Windows\Microsoft.NET\Framework\v2.0.50727



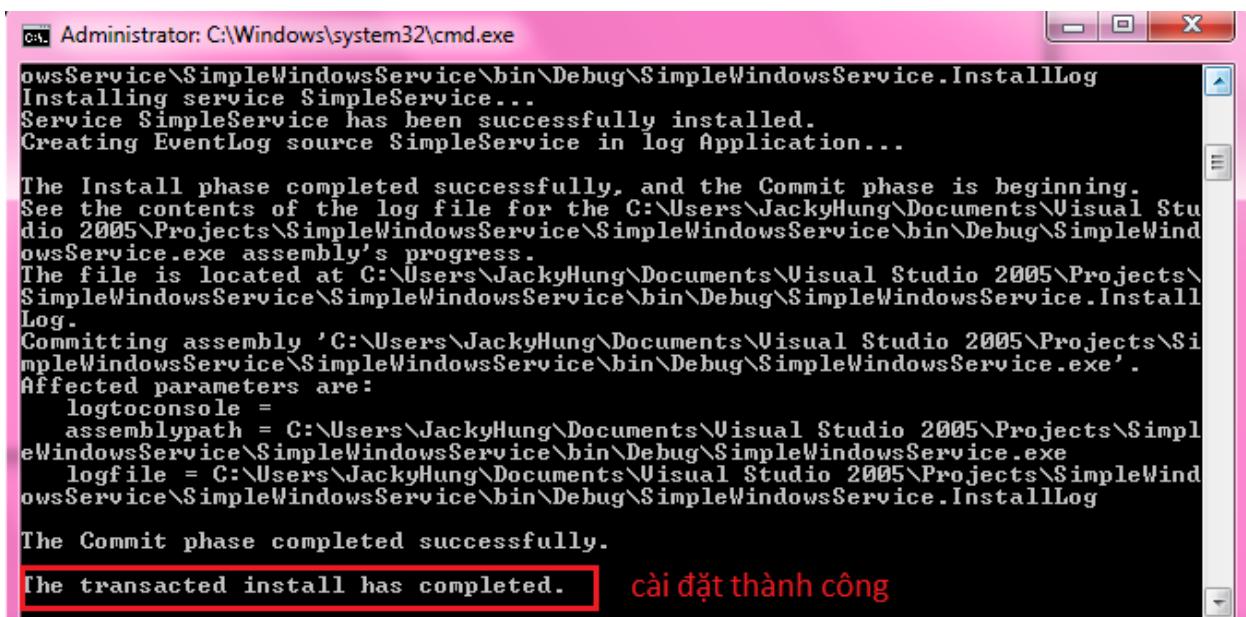
**Bước 2:** gõ câu lệnh installutil “đường dẫn đến file SimpleWindowsService.exe”.



```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright <c> 2009 Microsoft Corporation. All rights reserved.

C:\Users\JackyHung>cd C:\Windows\Microsoft.NET\Framework\v2.0.50727
C:\Windows\Microsoft.NET\Framework\v2.0.50727>installutil "C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe"
```

Lưu ý có dấu "". Rồi bấm enter



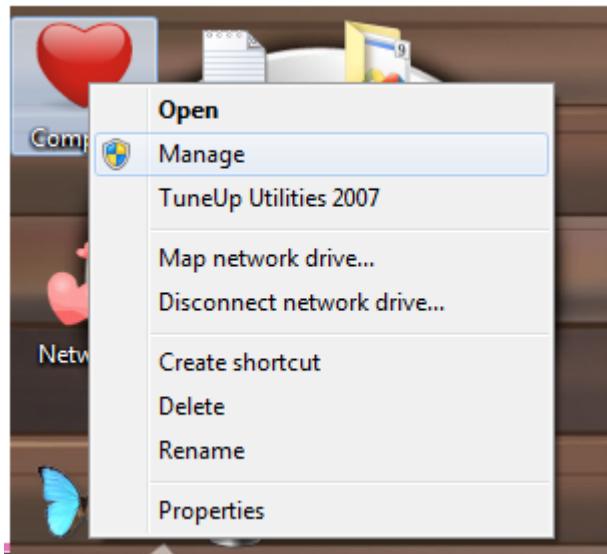
```
Administrator: C:\Windows\system32\cmd.exe
SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.InstallLog
Installing service SimpleService...
Service SimpleService has been successfully installed.
Creating EventLog source SimpleService in log Application...

The Install phase completed successfully, and the Commit phase is beginning.
See the contents of the log file for the C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe assembly's progress.
The file is located at C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.InstallLog.
Committing assembly 'C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe'.
Affected parameters are:
    logtoconsole =
        assemblypath = C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe
        logfile = C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.InstallLog

The Commit phase completed successfully.
The transacted install has completed.          cài đặt thành công
```

### Bước 3: Xem kết quả cài đặt

Click chuột phải vào biểu tượng my computer → manage.

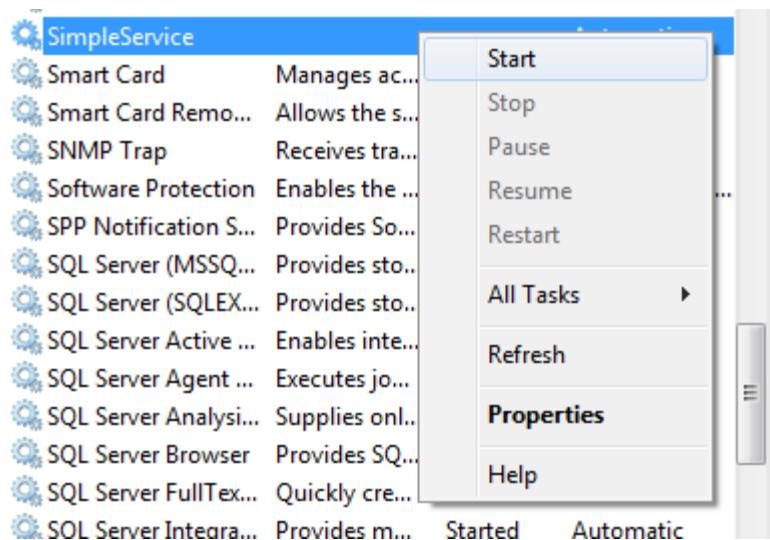


Hộp thoại hiện ra, bên trái chọn Service như hình, sau đó kéo xuống bạn sẽ thấy service của bạn là SimpleService.

The screenshot shows the 'Computer Management' window. On the left, under 'Services and Applications', the 'Services' icon is selected and highlighted with a red box. The right pane displays a table of services:

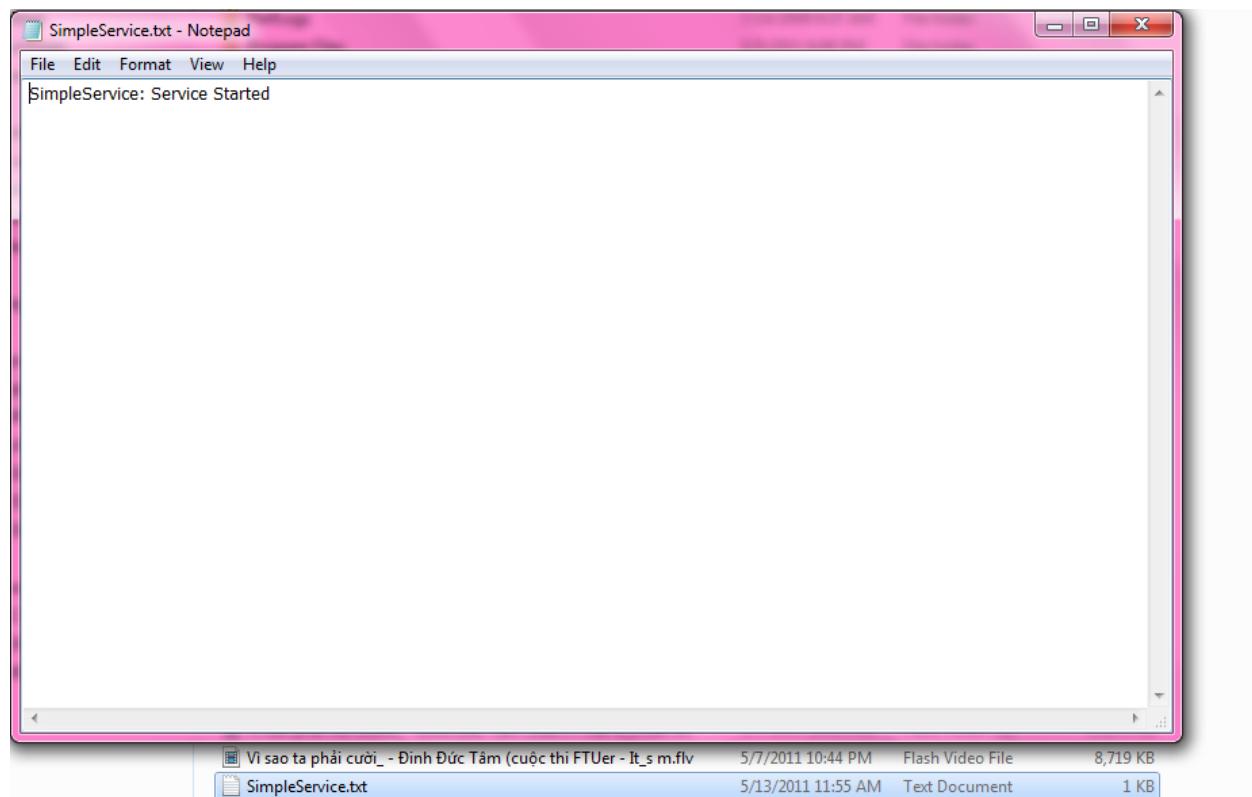
Name	Description	Status	Startup Type
Secondary Logon	Enables star...	Manual	Manual
Secure Socket Tun...	Provides su...	Started	Manual
Security Accounts...	The startup ...	Started	Automatic
Security Center	The WSCSV...	Started	Automatic (D...
Server	Supports fil...	Started	Automatic
Shell Hardware De...	Provides no...	Started	Automatic
<b>SimpleService</b>			<b>Automatic</b>
Smart Card	Manages ac...	Manual	Manual
Smart Card Remo...	Allows the s...	Manual	Manual
SNMP Trap	Receives tra...	Manual	Manual
Software Protection	Enables the ...	Automatic (D...	Automatic
SPP Notification S...	Provides So...	Manual	Manual
SQL Server (MSSQ...	Provides sto...	Started	Automatic
SQL Server (SQLEX...	Provides sto...	Started	Automatic
SQL Server Active ...	Enables inte...	Disabled	Disabled
SQL Server Agent ...	Executes jo...	Manual	Manual
SQL Server Analysi...	Supplies onl...	Started	Automatic
SQL Server Browser	Provides SO...	Started	Automatic

Bạn click chuột phải vô SimpleService → Start.



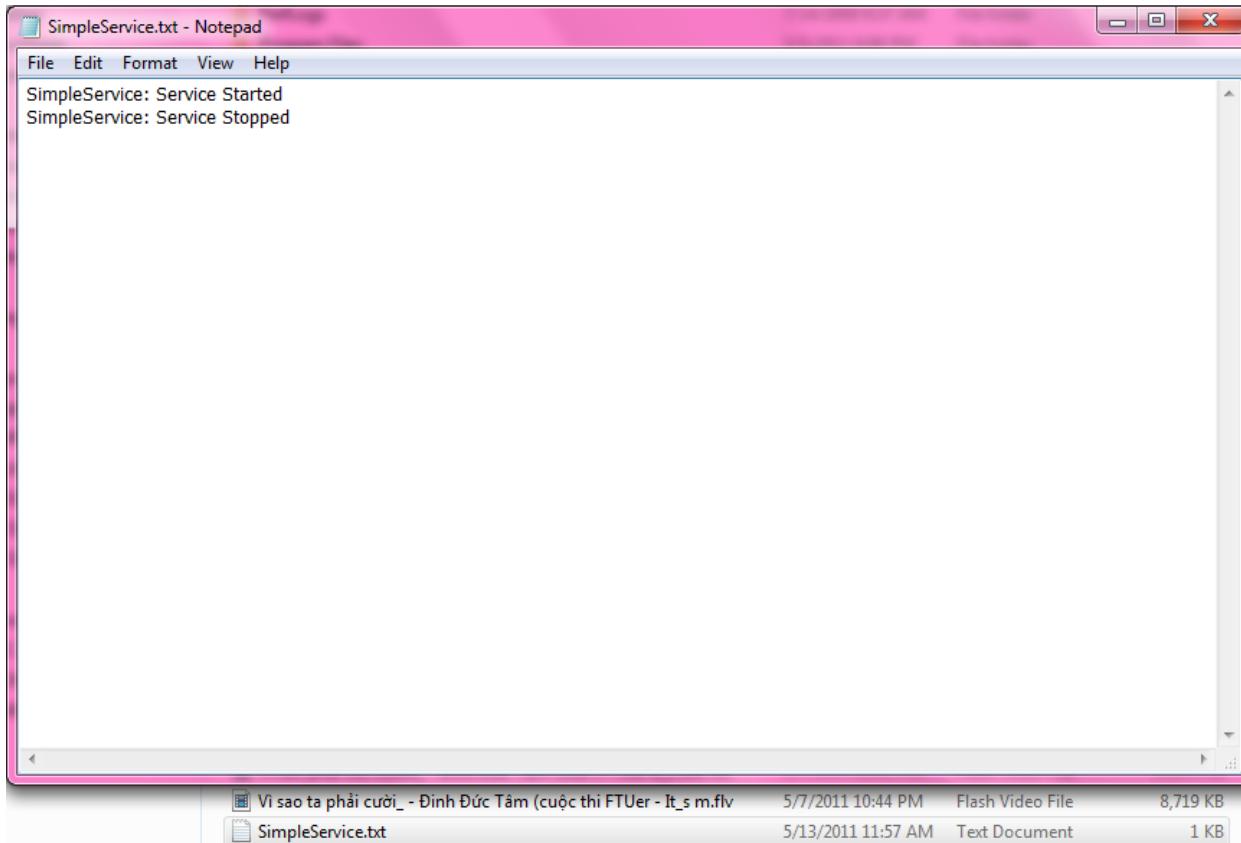
SimpleService của bạn đã bắt đầu hoạt động và gọi hàm OnStart trong source code.

Bạn vào ổ C sẽ có 1 file SimpleService.txt và bạn mở lên



Rồi bạn lại vào service và bạn click chuột phải lên SimpleService và chọn Stop. Nó sẽ gọi hàm OnStop của bạn và thực thi code trong đó.

Sau đó lại mở file SimpleService.txt sẽ có thông tin.



Để uninstall một service thì bạn làm tương tự như quá trình cài đặt mà chỉ thêm /u vào thôi như hình

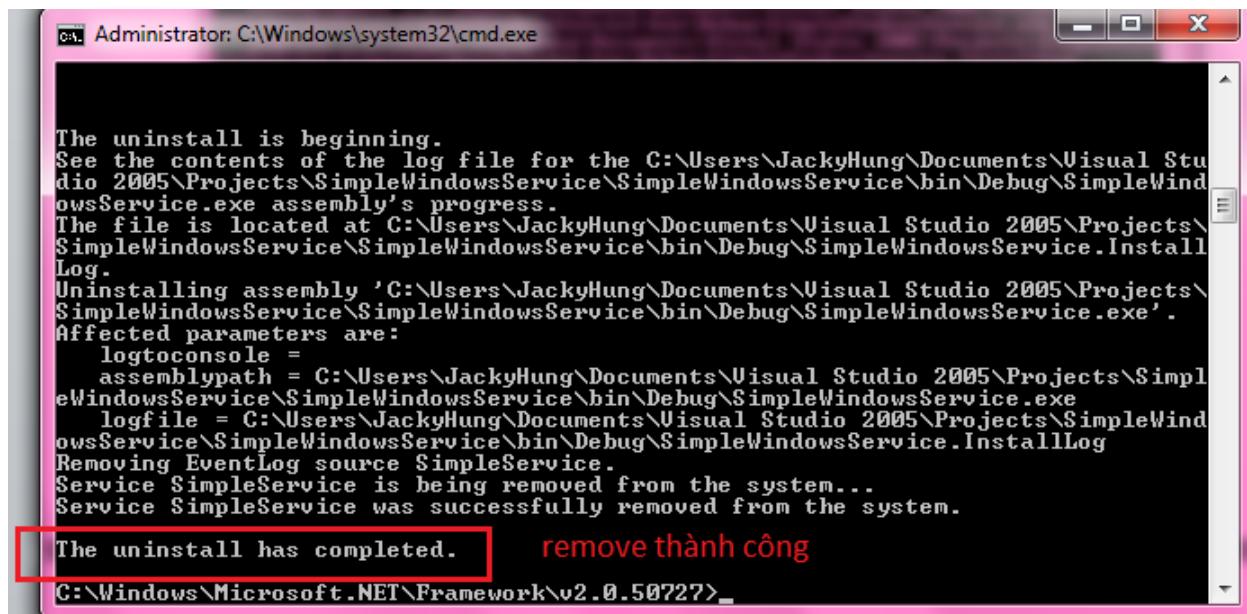
```
dio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWind
owsService.exe assembly's progress.
The file is located at C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\
SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.Install
Log.
Committing assembly 'C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\Si
mpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe'.
Affected parameters are:
    logtoconsole =
        assemblypath = C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\Simpl
eWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe
        logfile = C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWind
owsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.InstallLog

The Commit phase completed successfully.

The transacted install has completed.

C:\Windows\Microsoft.NET\Framework\v2.0.50727>installutil /u "C:\Users\JackyHung
\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService
\bin\Debug\SimpleWindowsService.exe"
```

Bấm Enter



```

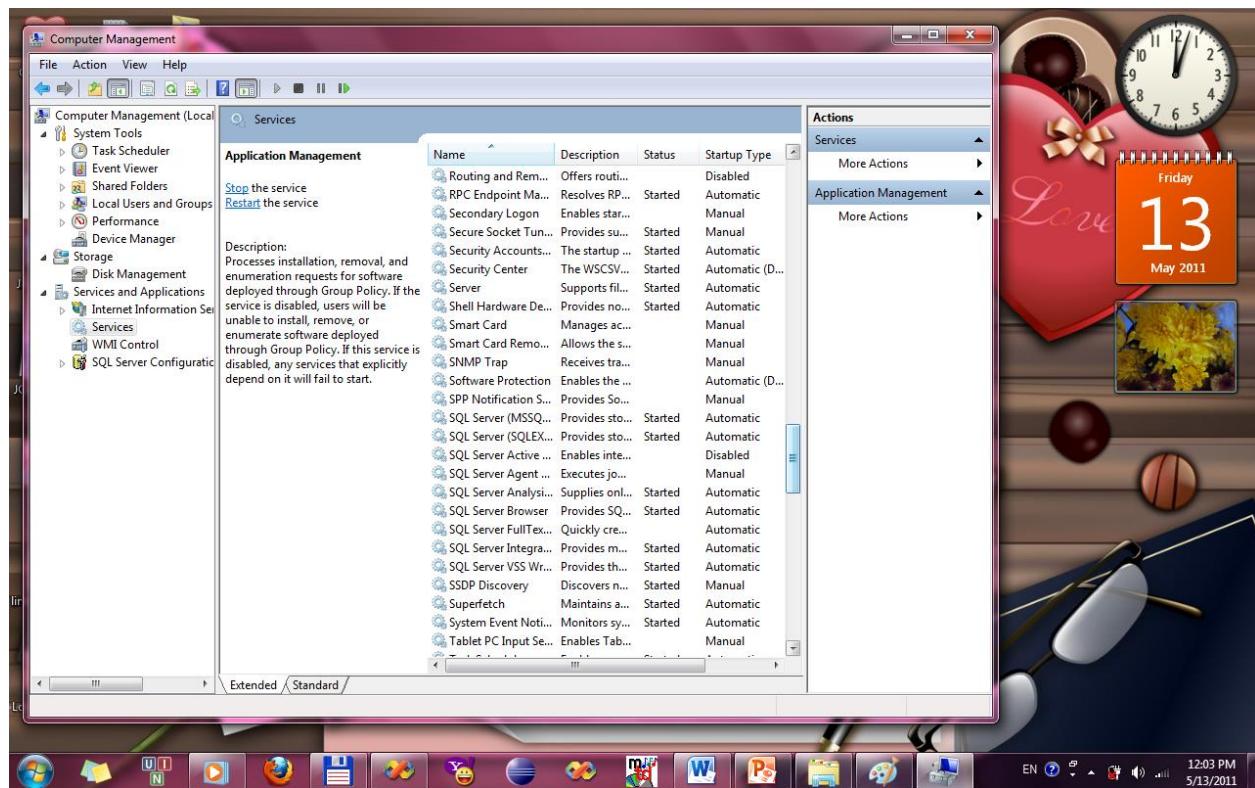
Administrator: C:\Windows\system32\cmd.exe

The uninstall is beginning.
See the contents of the log file for the C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe assembly's progress.
The file is located at C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.InstallLog.
Uninstalling assembly 'C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe'.
Affected parameters are:
    logtoconsole =
    assemblypath = C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.exe
    logfile = C:\Users\JackyHung\Documents\Visual Studio 2005\Projects\SimpleWindowsService\SimpleWindowsService\bin\Debug\SimpleWindowsService.InstallLog
Removing EventLog source SimpleService.
Service SimpleService is being removed from the system...
Service SimpleService was successfully removed from the system.

The uninstall has completed.      remove thành công
C:\Windows\Microsoft.NET\Framework\v2.0.50727>_

```

Ta vào lại danh sách service trên windows bạn sẽ không tìm thấy service của bạn nữa



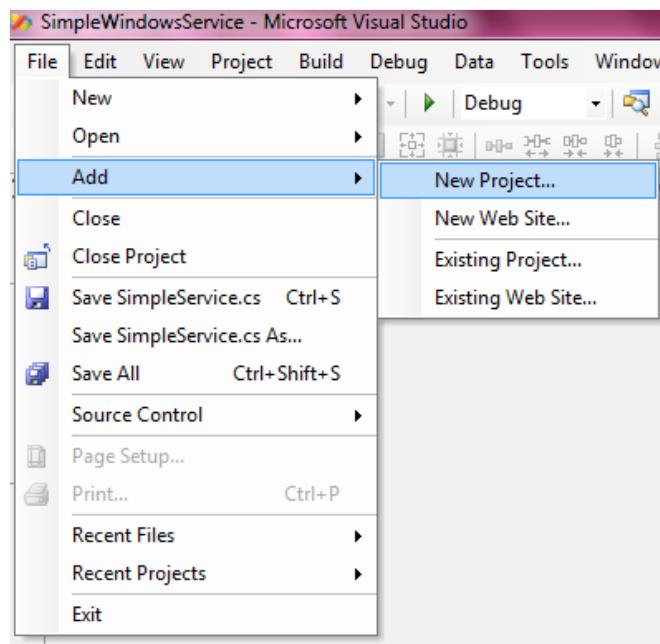
# HƯỚNG DẪN 8: CÀI ĐẶT WINDOWS SERVICE BẰNG FILE VÀ THIẾT KẾ GIAO DIỆN QUẢN LÝ

## Tạo file cài đặt windows service

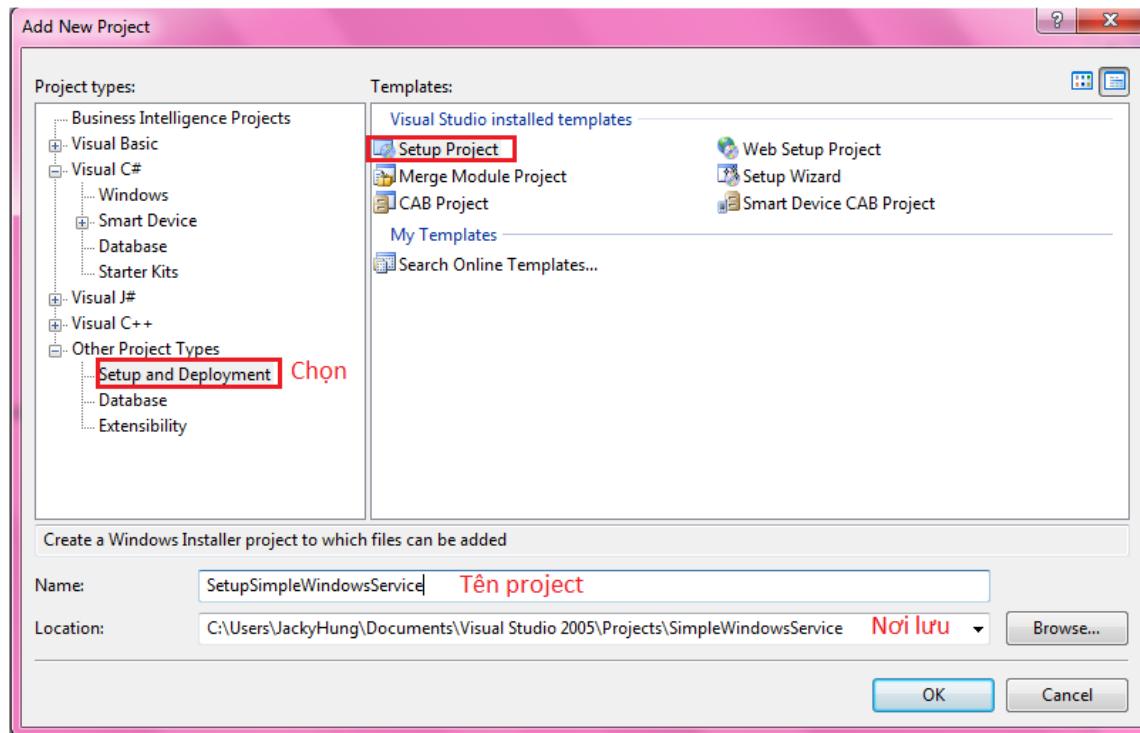
**Bước 1:** Đầu tiên mở chương trình Microsoft Visual Studio lên.

**Bước 2:** Mở project tạo service mà bạn đã tạo trước đó. Ở đây là project SimpleWindowsService.

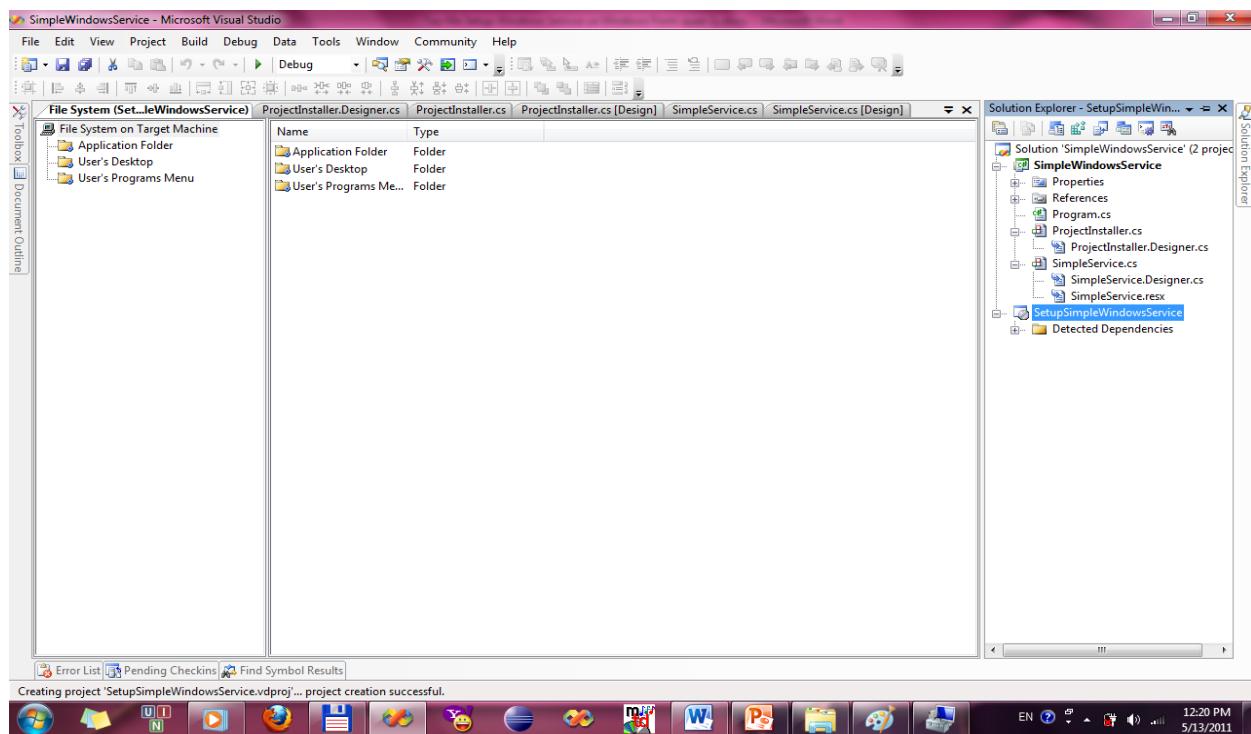
**Bước 3:** Sau khi bạn mở xong vào File → Add → New Project.



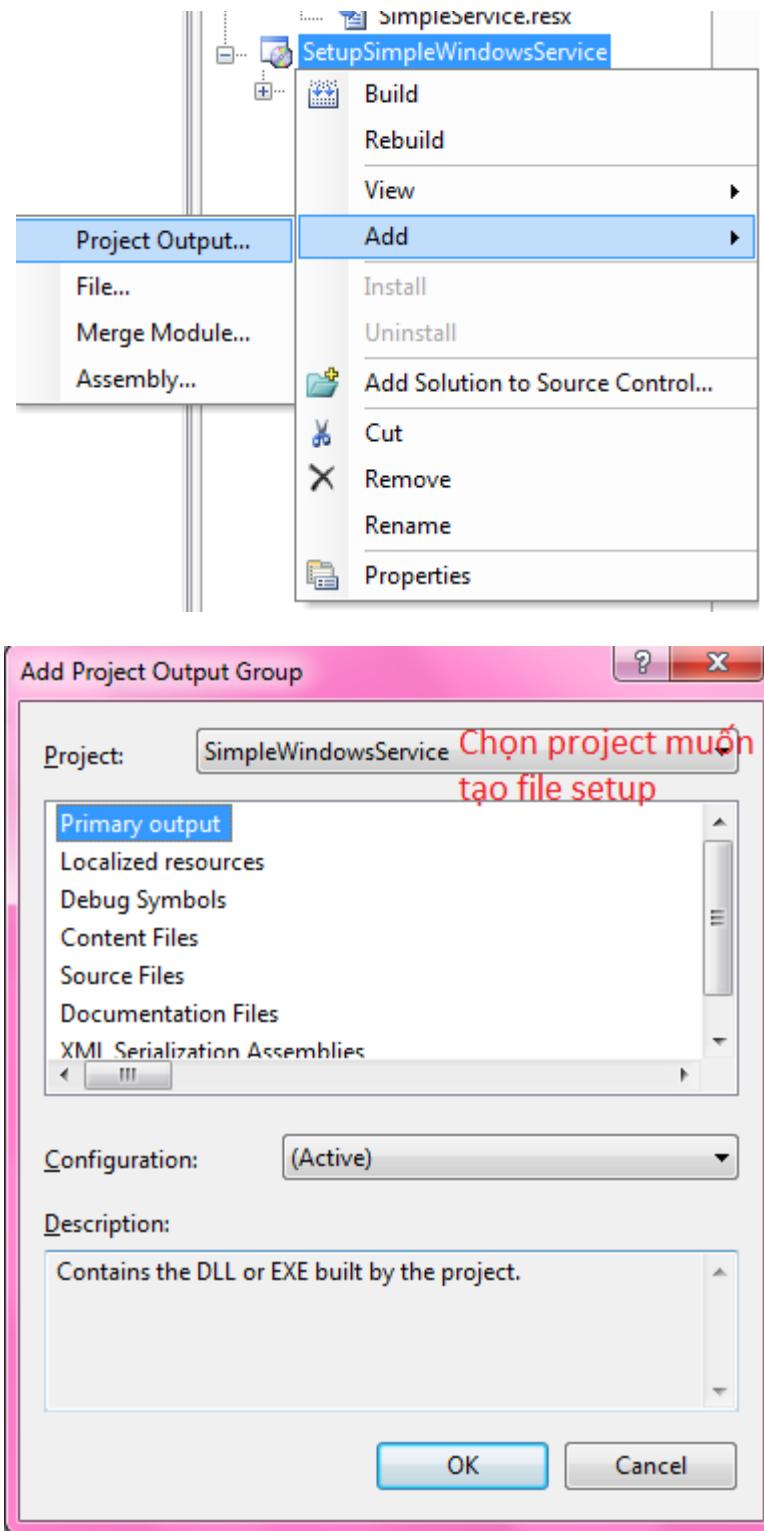
Rồi làm như sau



Sau khi bấm OK. Bạn đã có thêm project : SetupSimpleWindowsService



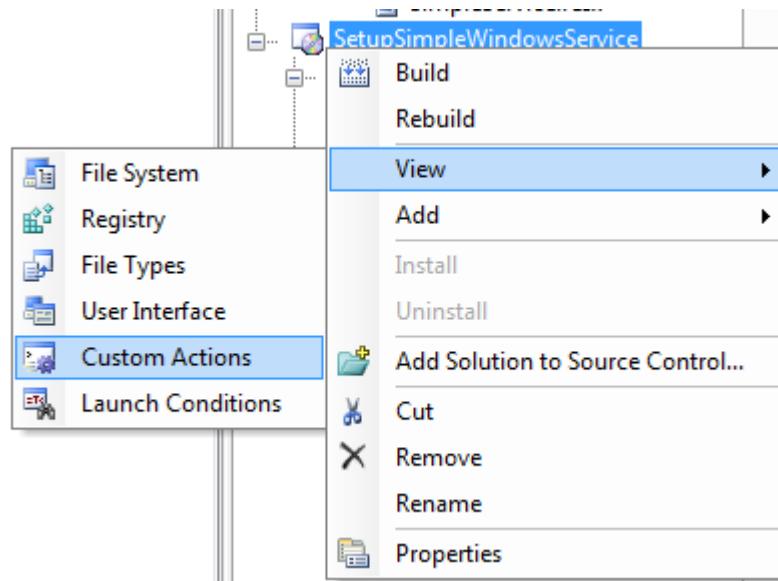
**Bước 4:** Click chuột phải vào SetupSimpleWindowsService → Add → Project Output.



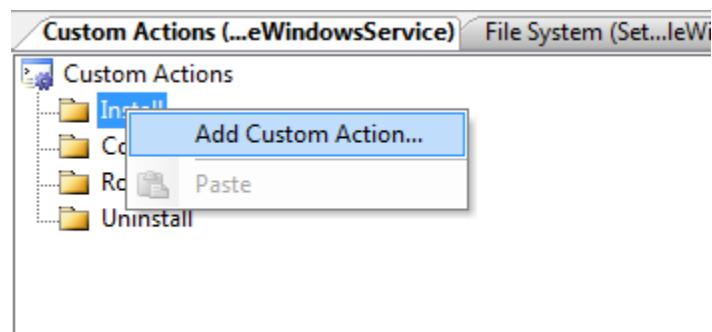
Bấm Ok.

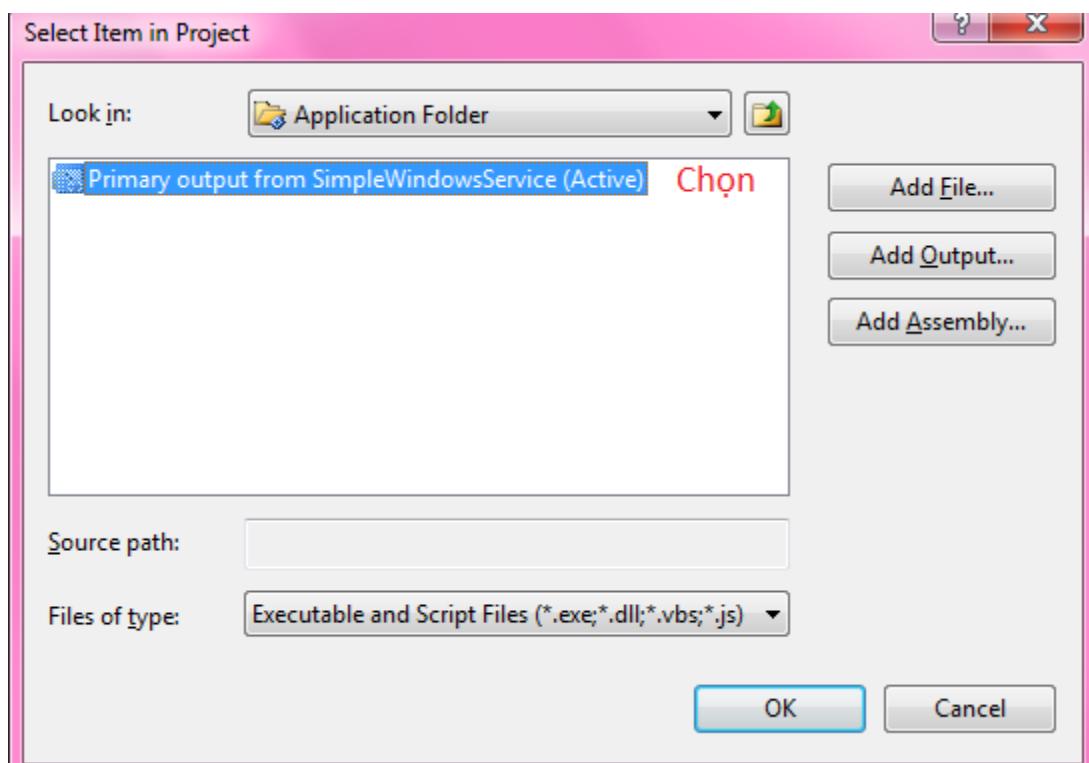
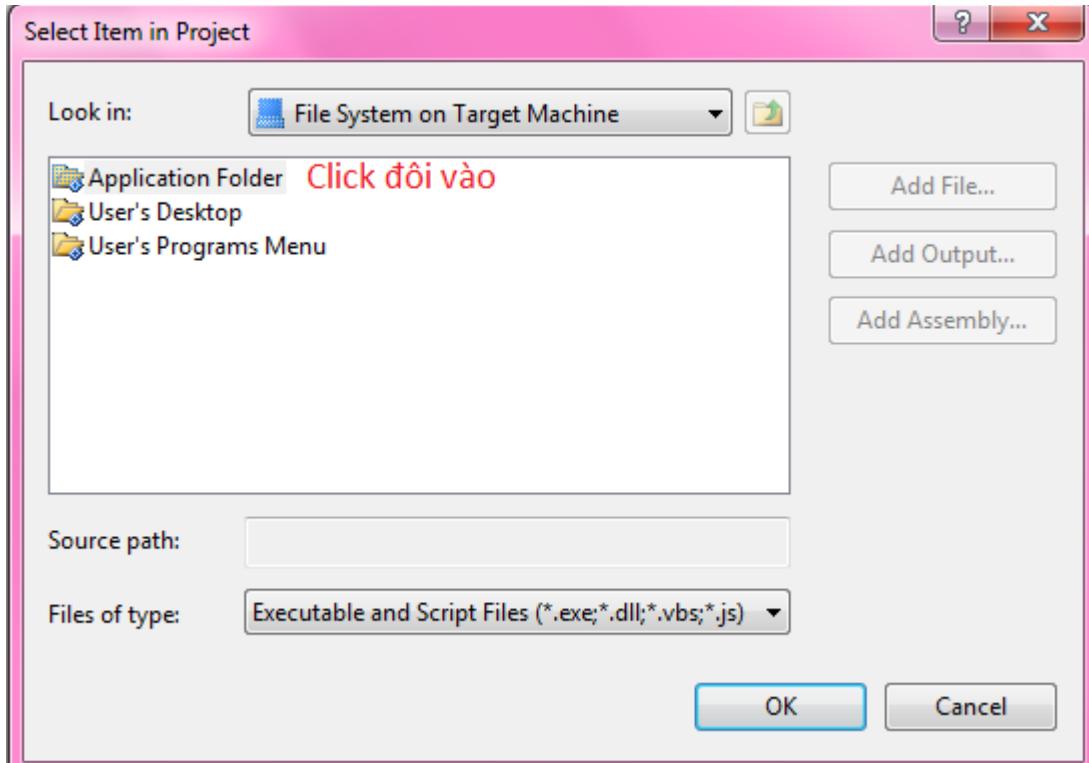
#### **Bước 5:** Cấu hình

Click chuột phải vào SetupSimpleWindowsService → View → Custom Actions

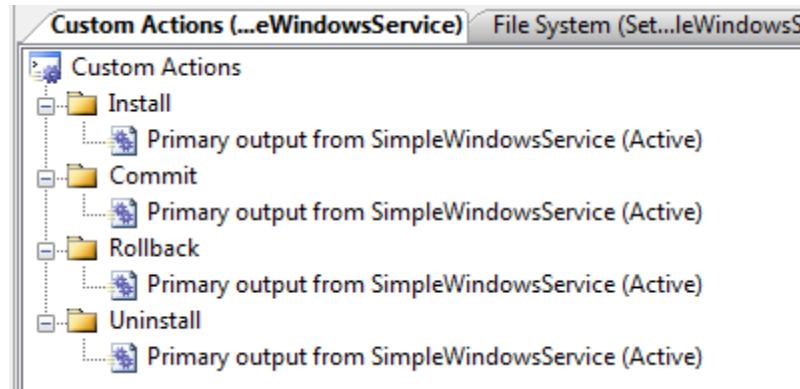


Click phải chuột vào từng folder rồi chọn Add Custom Action

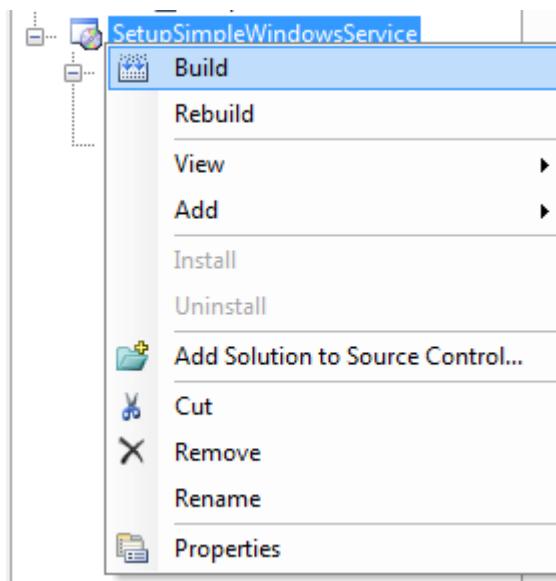




Tương tự kết quả cuối cùng được

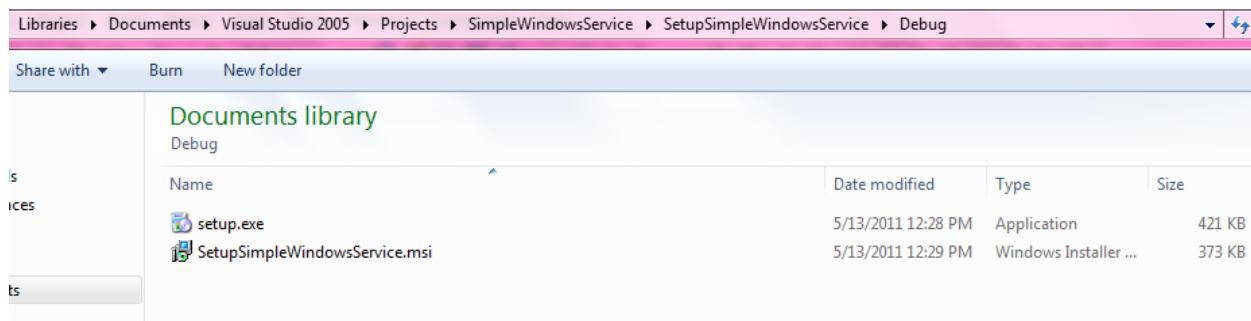


**Bước 6:** Click chuột phải vào SetupSimpleWindowsService → build

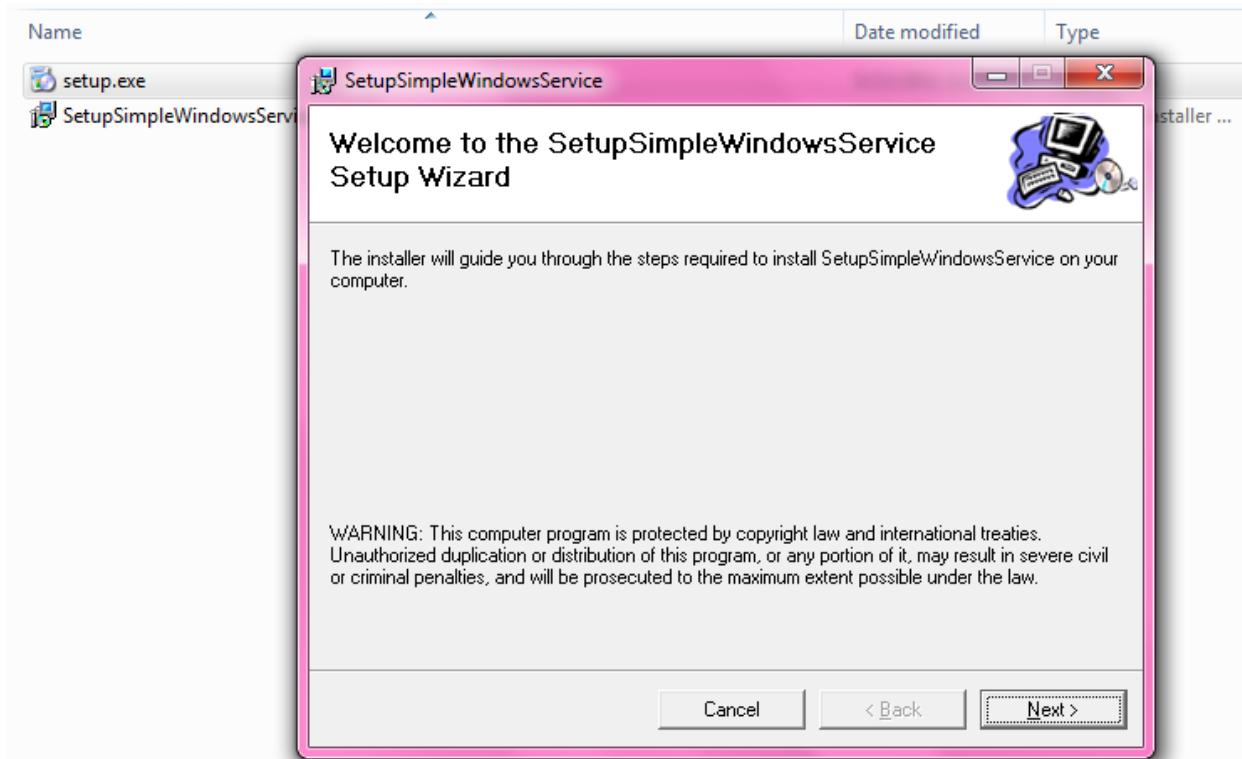


**Bước 7:** Cài đặt.

Sau khi build xong bạn sẽ được một file cài đặt tạo ra trong thư mục debug nơi chứa project SetupSimpleWindowsService này.

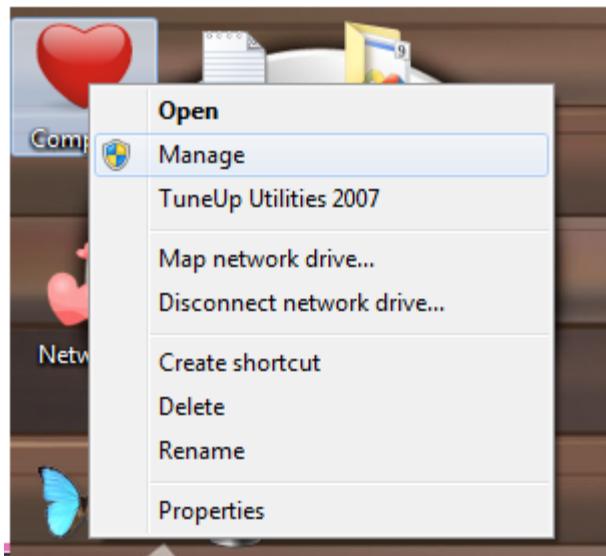


Bạn click đúp vào file setup để tiến hành cài đặt service của bạn vào máy tính.

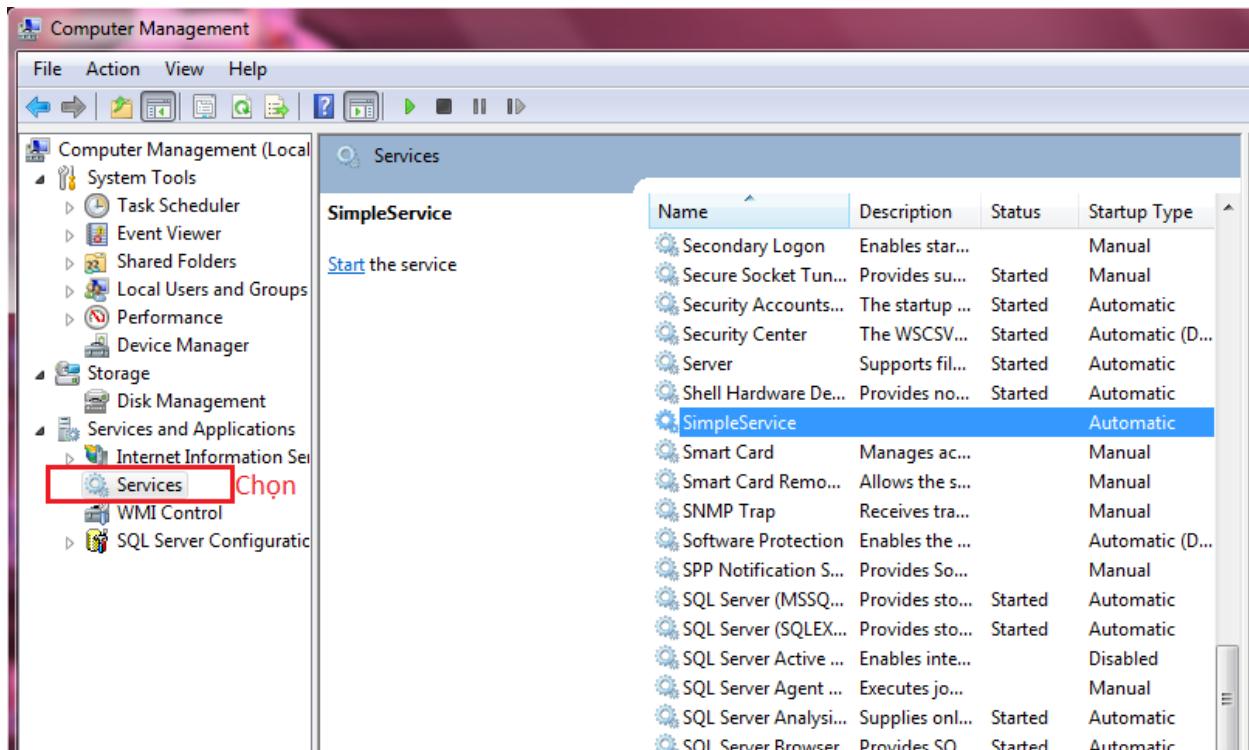


### **Bước 8:** Xem kết quả cài đặt

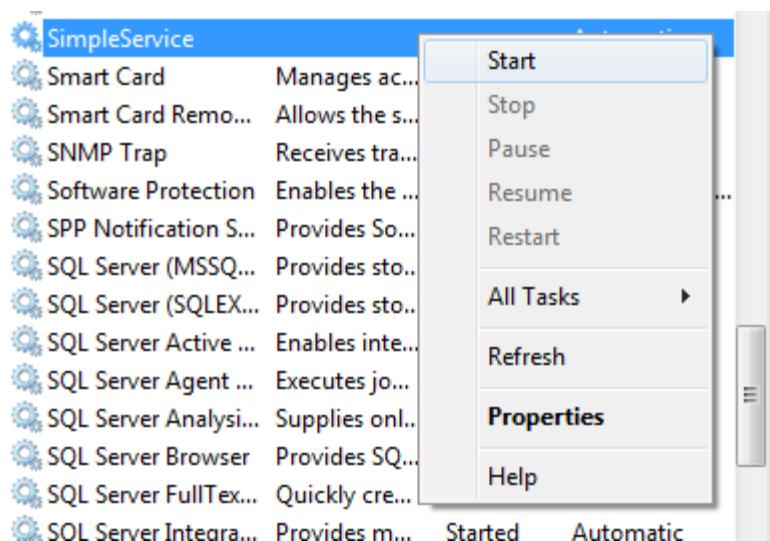
Click chuột phải vào biểu tượng my computer → manage.



Hộp thoại hiện ra, bên trái chọn Service như hình, sau đó kéo xuống bạn sẽ thấy service của bạn là SimpleService.

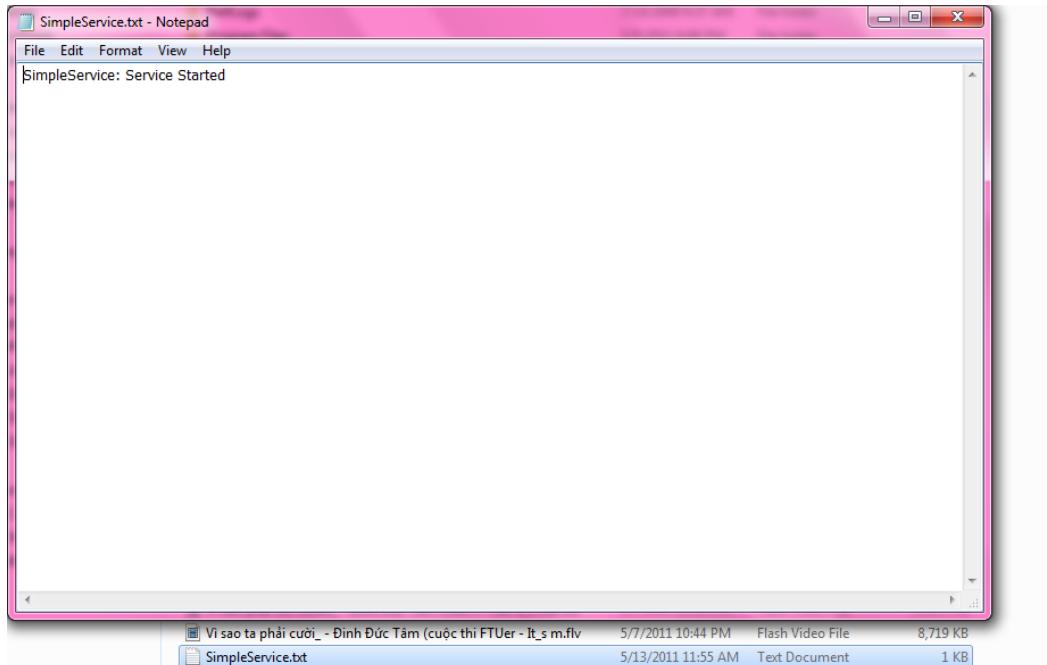


Bạn click chuột phải vô SimpleService → Start.



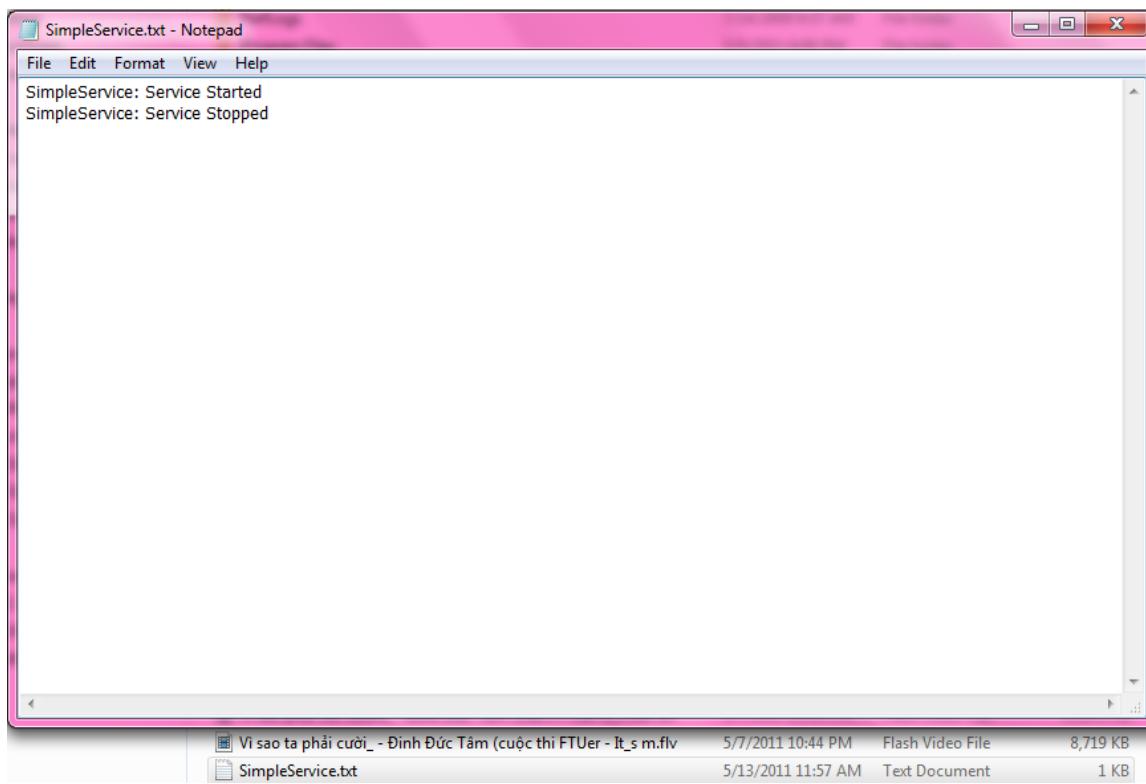
SimpleService của bạn đã bắt đầu hoạt động và gọi hàm OnStart trong source code.

Bạn vào ổ C sẽ có 1 file SimpleService.txt và bạn mở lên



Rồi bạn lại vào service và bạn click chuột phải lên SimpleService và chọn Stop. Nó sẽ gọi hàm OnStop của bạn và thực thi code trong đó.

Sau đó lại mở file SimpleService.txt sẽ có thông tin.

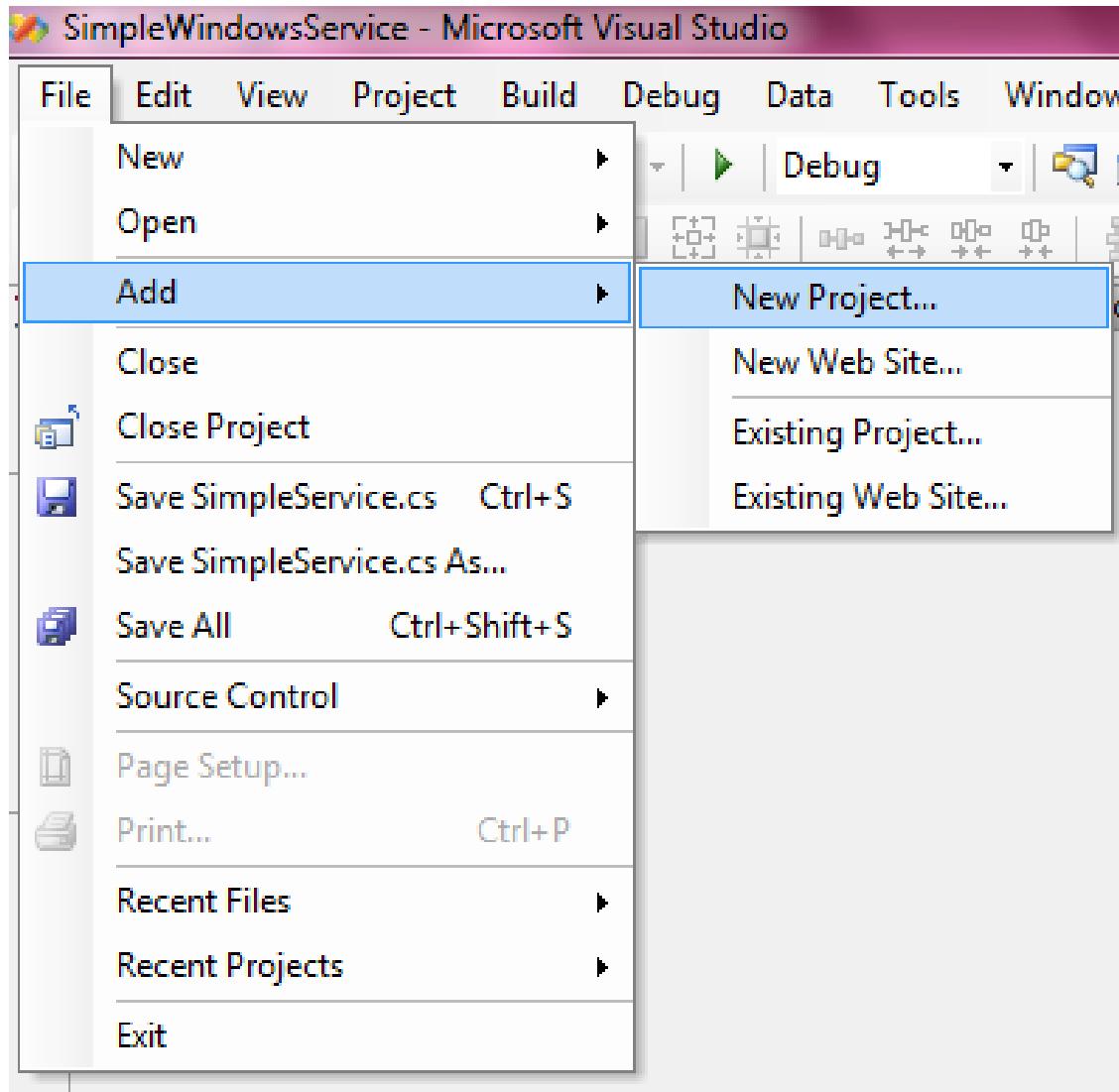


## Chương trình quản lý WindowService

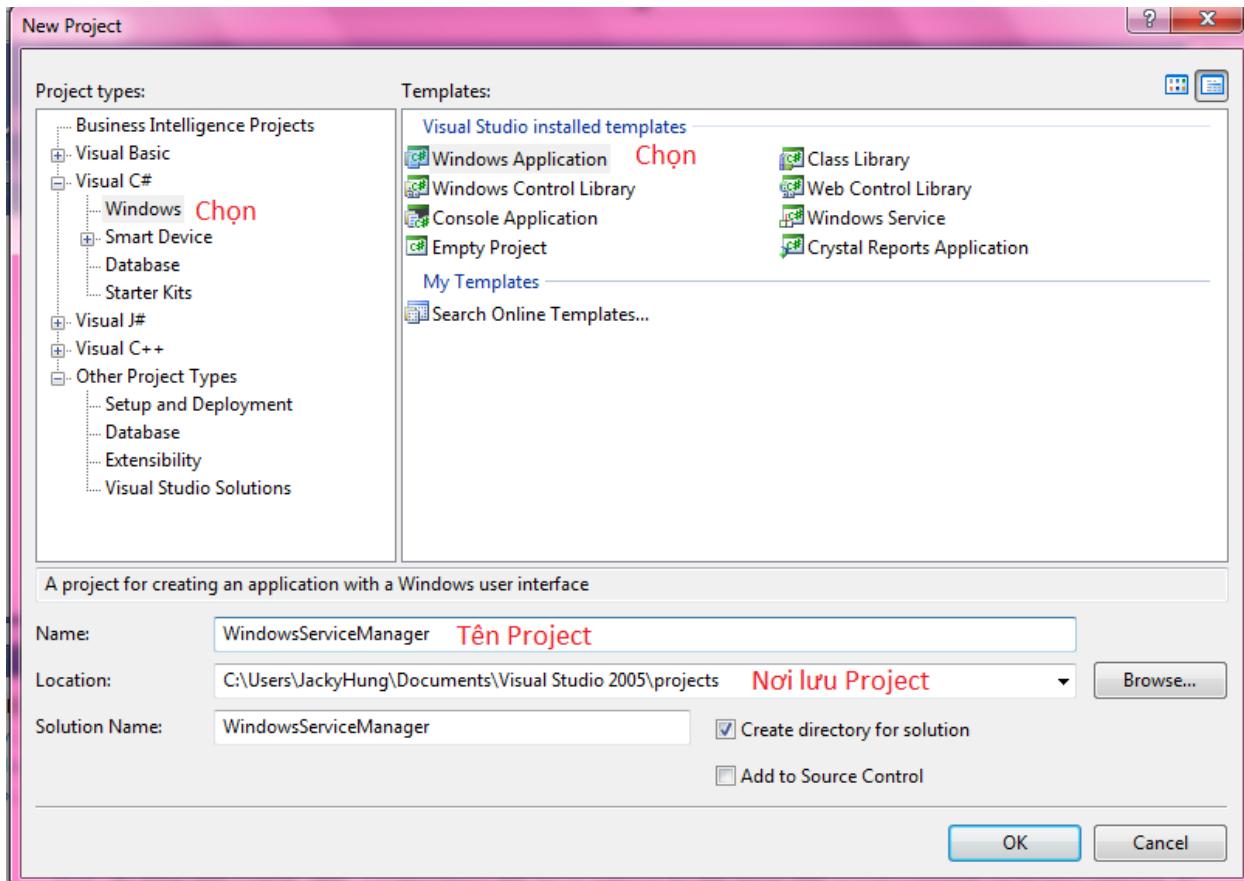
**Bước 1:** Đầu tiên mở chương trình Microsoft Visual Studio lên.

**Bước 2:** Mở project tạo service mà bạn đã tạo trước đó. Ở đây là project SimpleWindowsService.

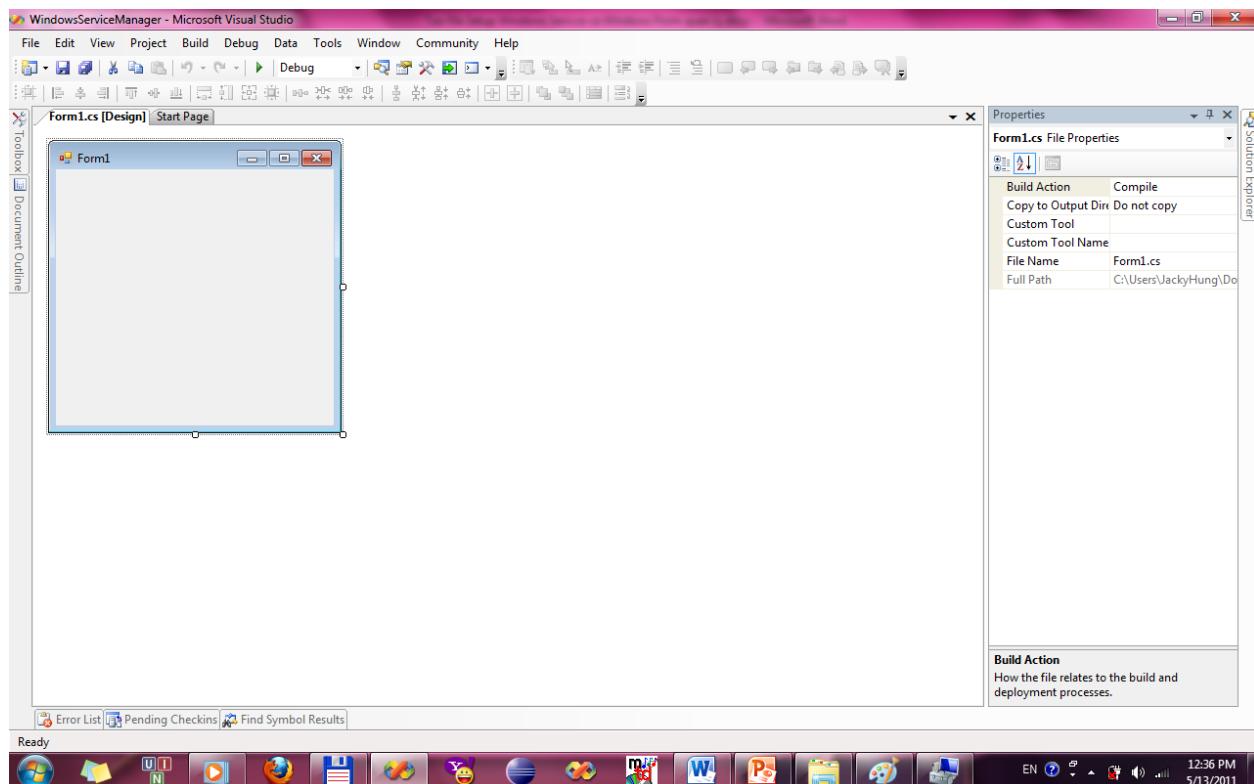
**Bước 3:** Sau khi bạn mở xong vào File → Add → New Project.



Rồi làm như sau:



Sau khi bấm ok.



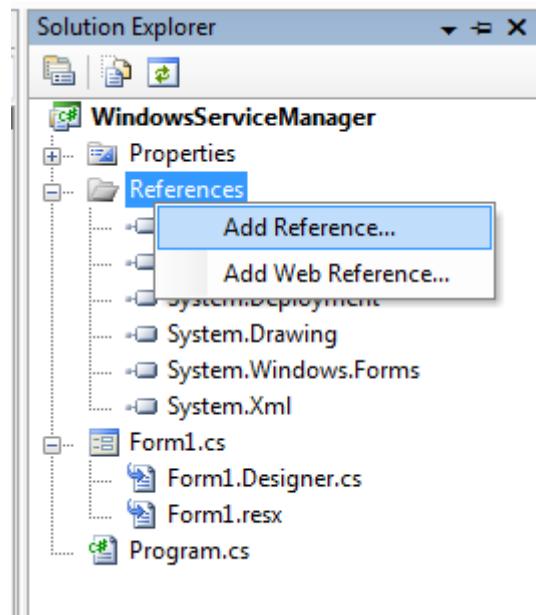
**Bước 4:** Thiết kế form như sau

Các tên btnStart, btnStop, btnPause, btnResume, btnExit là các cái name của button.

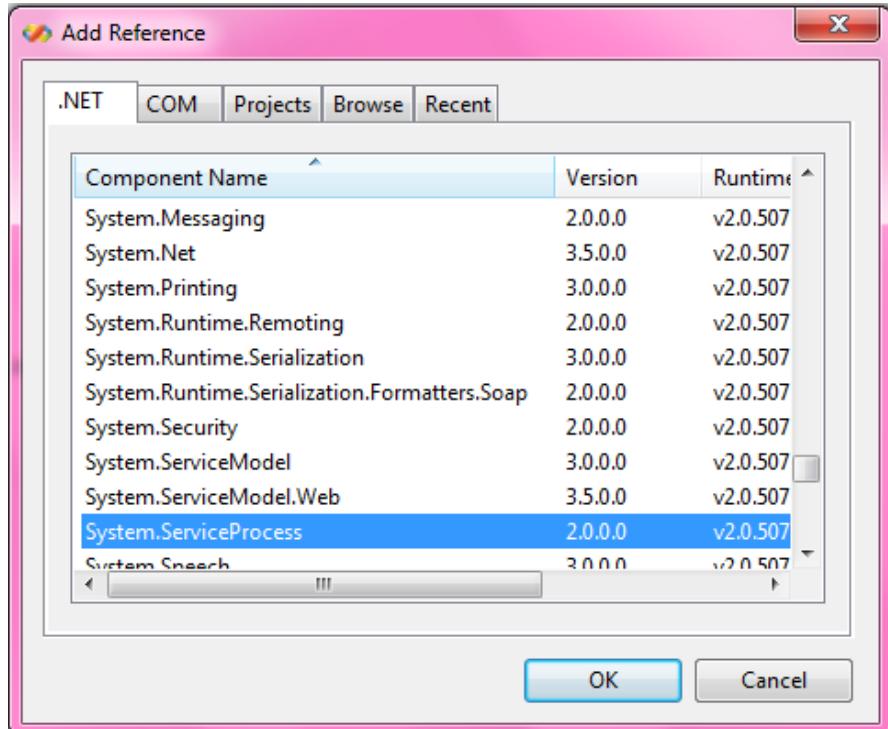
Tên txtWindowsService là cái name của textbox.

**Bước 5:** Import thư viện cần để quản lý windows service

Click chuột phải vào References → add Reference....



Kéo xuống rồi chọn System.ServiceProcess.



Bấm OK.

### Bước 6: Code xử lý các sự kiện click chuột vào các button

Click đôi chuột vào nút Start.

```

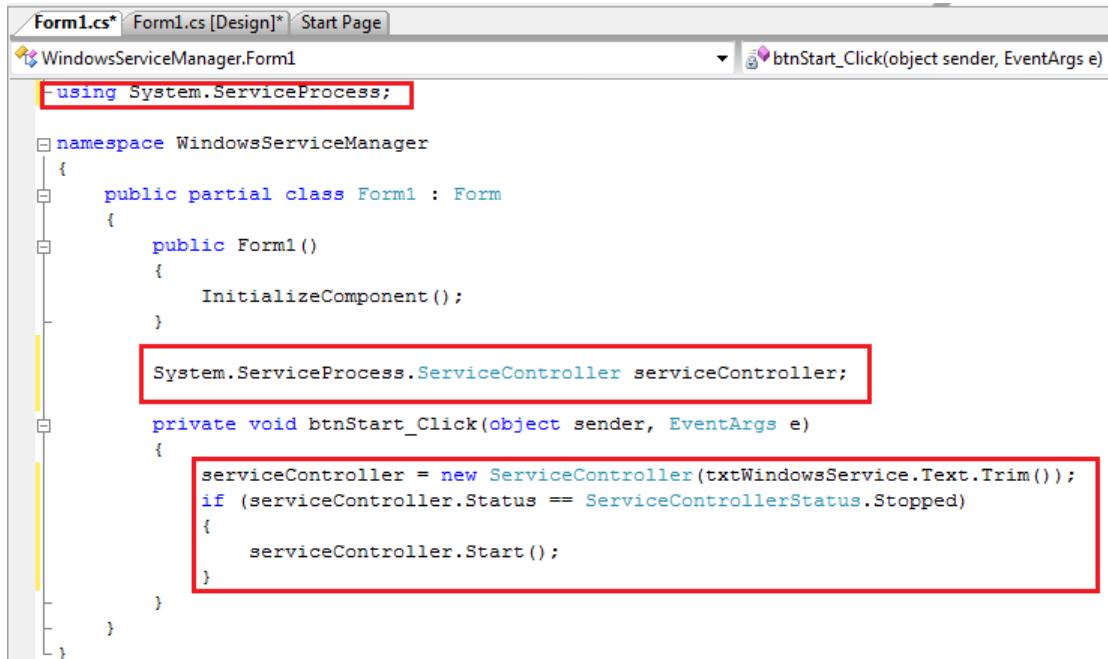
Form1.cs* Form1.cs [Design]* Start Page
WindowsServiceManager.Form1
using System.ServiceProcess;

namespace WindowsServiceManager
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnStart_Click(object sender, EventArgs e)
        {
        }
    }
}

```

Tại đây ta tiến hành code như sau:



```

Form1.cs* Form1.cs [Design]* Start Page
WindowsServiceManager.Form1
▼ btnStart_Click(object sender, EventArgs e)

using System.ServiceProcess;

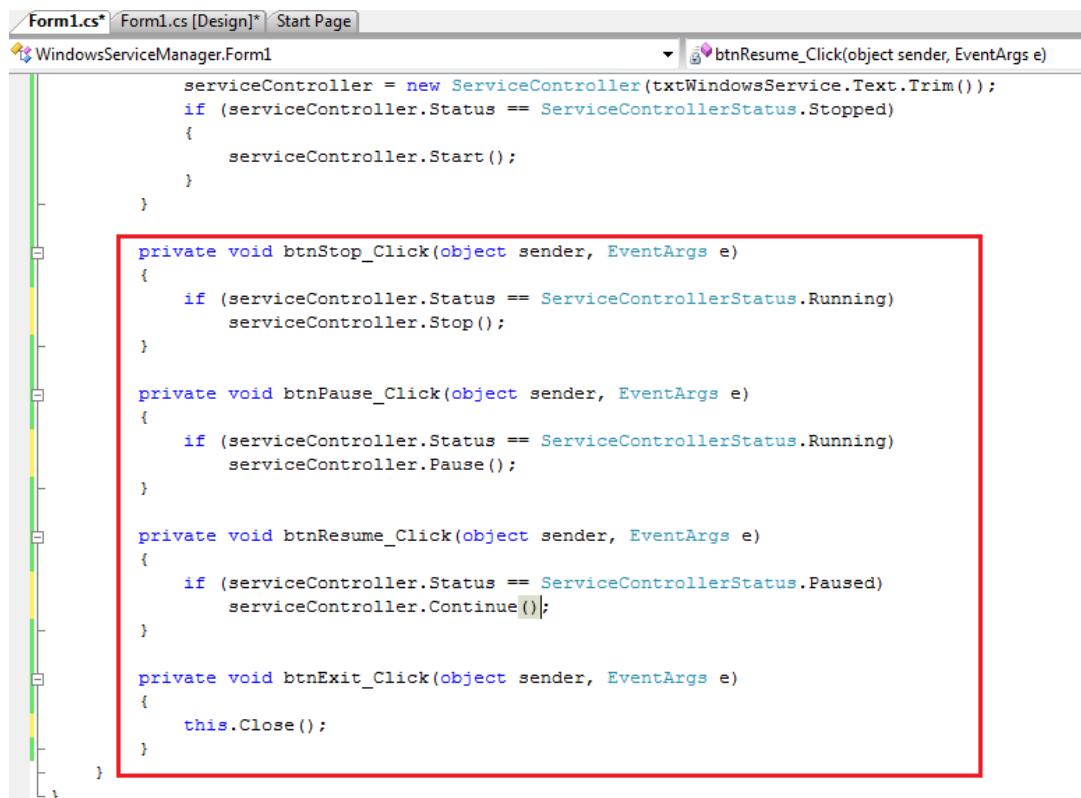
namespace WindowsServiceManager
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        System.ServiceProcess.ServiceController serviceController;

        private void btnStart_Click(object sender, EventArgs e)
        {
            serviceController = new ServiceController(txtWindowsService.Text.Trim());
            if (serviceController.Status == ServiceControllerStatus.Stopped)
            {
                serviceController.Start();
            }
        }
    }
}

```

Tương tự click đôi chuột vào các button Stop, Pause, Resume, Exit và viết code tương ứng.



```

Form1.cs* Form1.cs [Design]* Start Page
WindowsServiceManager.Form1
▼ btnResume_Click(object sender, EventArgs e)

        serviceController = new ServiceController(txtWindowsService.Text.Trim());
        if (serviceController.Status == ServiceControllerStatus.Stopped)
        {
            serviceController.Start();
        }

private void btnStop_Click(object sender, EventArgs e)
{
    if (serviceController.Status == ServiceControllerStatus.Running)
        serviceController.Stop();
}

private void btnPause_Click(object sender, EventArgs e)
{
    if (serviceController.Status == ServiceControllerStatus.Running)
        serviceController.Pause();
}

private void btnResume_Click(object sender, EventArgs e)
{
    if (serviceController.Status == ServiceControllerStatus.Paused)
        serviceController.Continue();
}

private void btnExit_Click(object sender, EventArgs e)
{
    this.Close();
}

```

Bước 7: Chạy và xem kết quả ở