

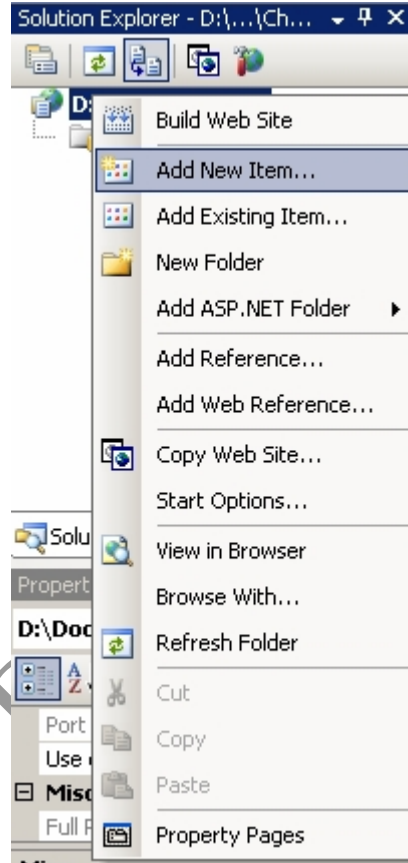
الفصل الثانى : الوراثة من الصفحة الأم و أدوات الملاحة.

## Master Page and Navigation tools.

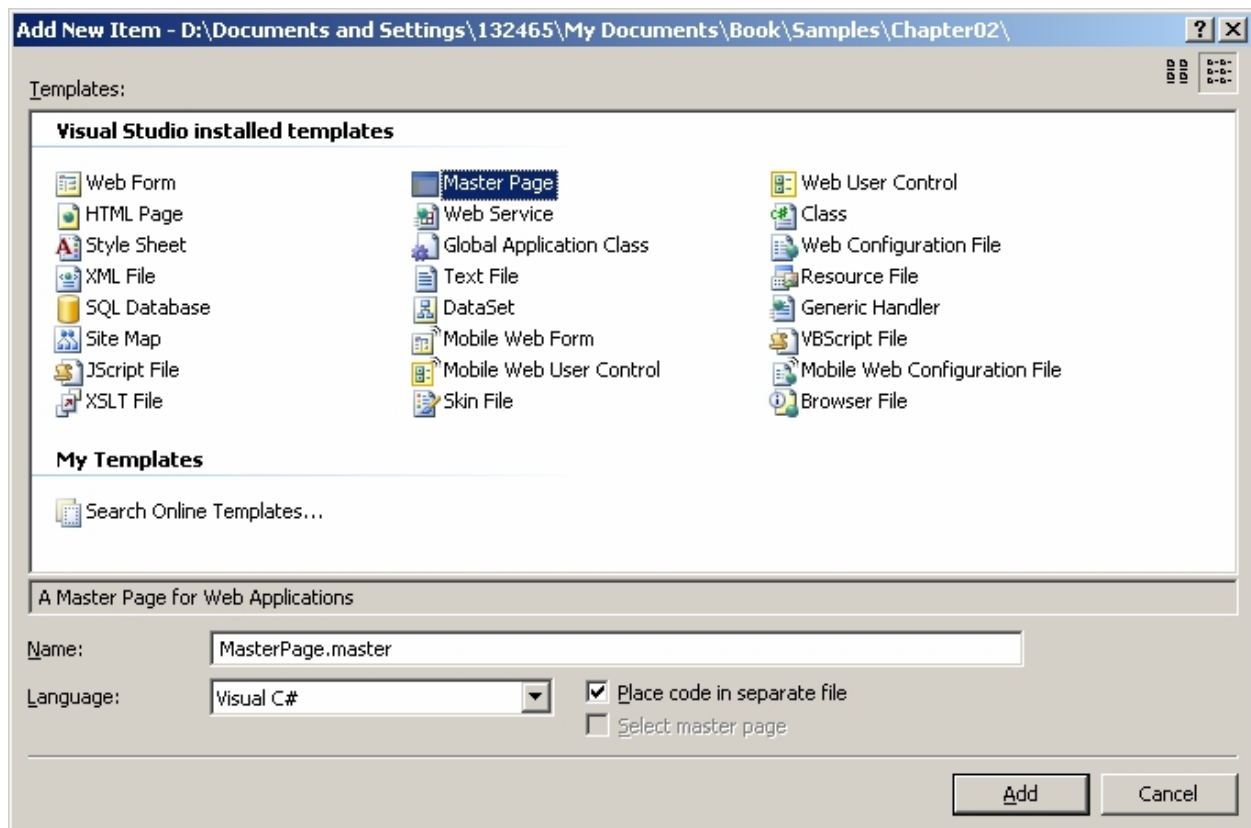
---

من إحدى المميزات الجديدة و التي سوف تحدث فارقا كبيرا من حيث توفير الوقت و المجهود المبذول في ضبط و أعداد تصميم الصفحات على ما أظن أن معظم المطورين يعانون من التعامل مع التصميم **User interface design** و هذا الموضوع كان يسبب الكثير من الأزعاج للمطورين حيث كان يستهلك الكثير من الوقت الآن حل هذه المشكلة في الميزة الجديدة التي يوفرها **Visual Studio.net 2005** هذه الميزة هي توريث واجه المستخدم. هذه الميزة من أهم المميزات التي تساعد و توفر الكثير من الوقت و المجهود، الآن سوف نضع مثال بسيط للتعامل مع الـ **Master Page** أو ما يسمى بتوريث واجه التعامل.

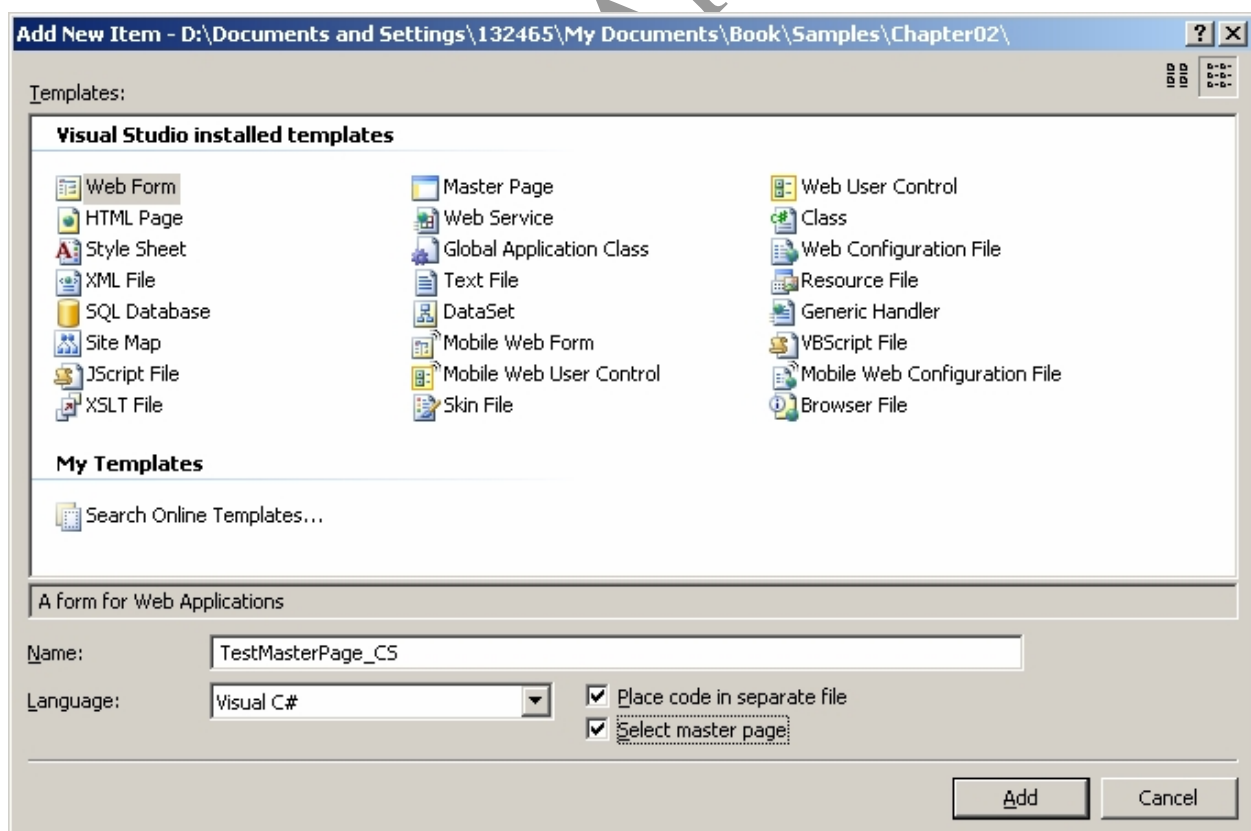
و الآن لنبدأ بمثل صغير لكيفية استخدام الـ **MasterPage** و كيف يمكن تفعيل هذه الميزة الجديدة.



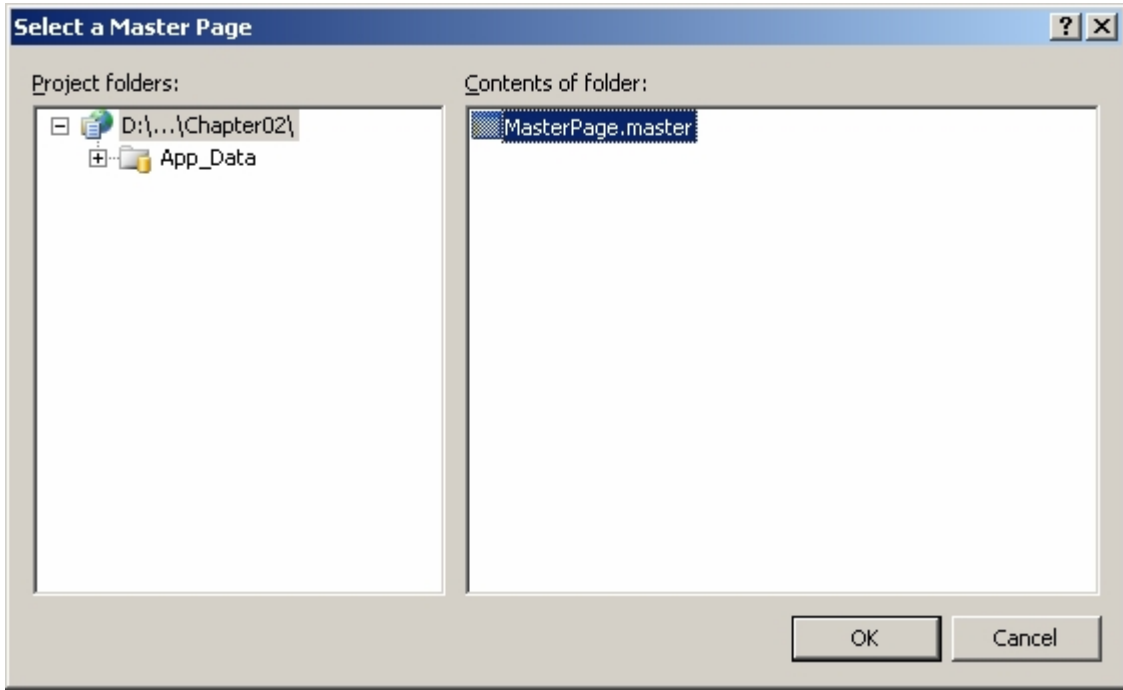
الشكل ١ نقوم أولاً بإضافة عنصر جديد للمشروع.



الشكل ٢ نختار Master Page من قائمة القوالب Templates

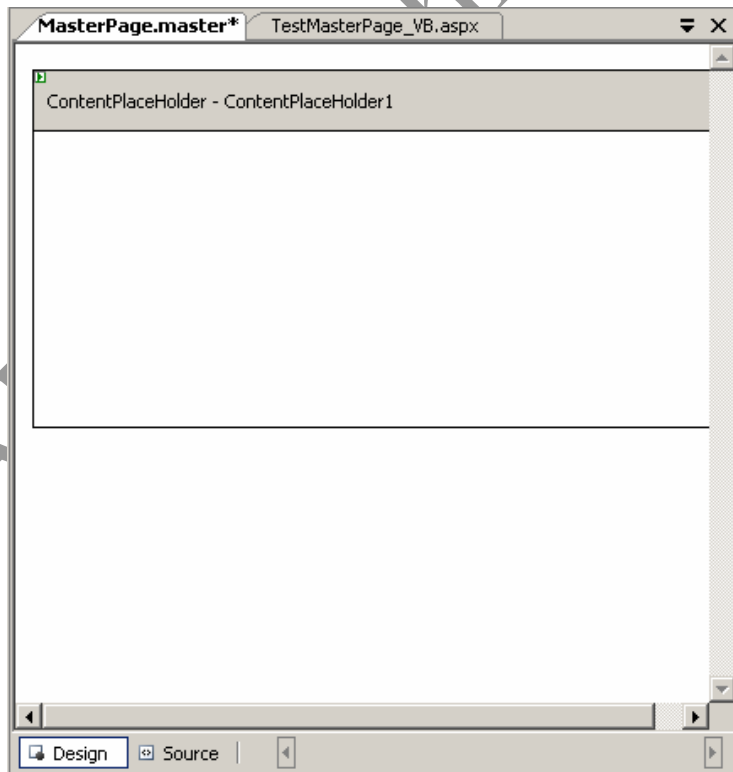


الشكل ٣ ثم نقوم بعد ذلك بإضافة عنصر جديد هذه المرة سوف نختار Web Form من قائمة القوالب و سوف نقوم بتفعيل الخيار Select Master Page بعد ذلك نضغط على الزر Add



الشكل ٤ : سوف تظهر هذه النافذة و التي سوف تختار منها الصفحة الأم التي تريد وراثتها الشكل منها، أختار الصفحة الأم التي تريدها ثم أضغط

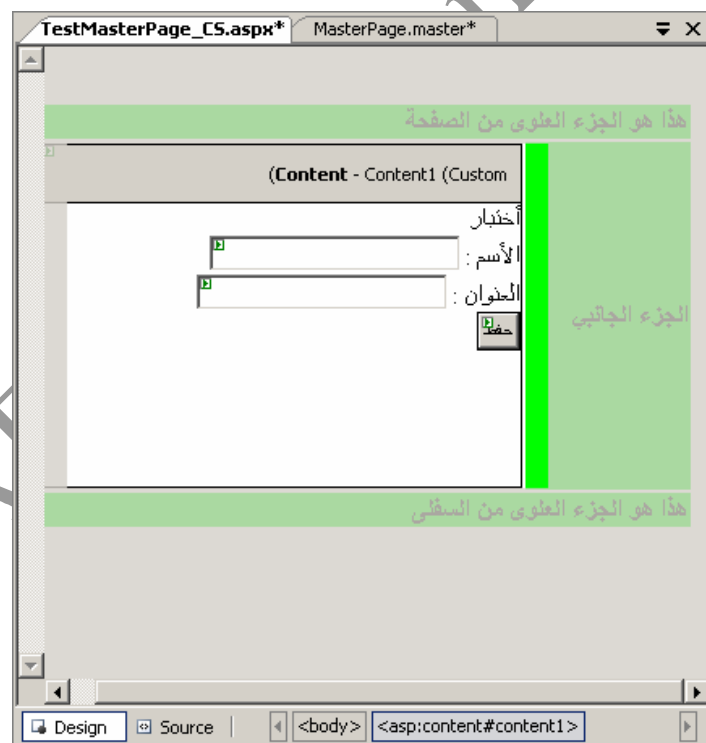
OK



الشكل 5 هذا هو شكل الـ Master Page في حالة التصميم Design mode و سوف نلاحظ وجود ما يسمى بالـ ContentPlaceHolder و هذا العنصر مهم جدا حيث أن هذا العنصر هو الذي سوف يتحوى على أى صفحة ترث من الصفحة الأم، من هنا يتضح لنا كيفية عمل الـ MasterPage و هي أن الصفحة الأم صفحة عادية معرفة لدى المترجم بشكل خاص بها ContentPlaceHolder و الذى سوف يكون همزة الوصل بين الـ Master Page و الصفحة التي ترث منها، للعلم طريقة عمل الـ Master Page مشابهة لحد كبير خاصية الـ Master Slide الموجودة فى الـ Power Point

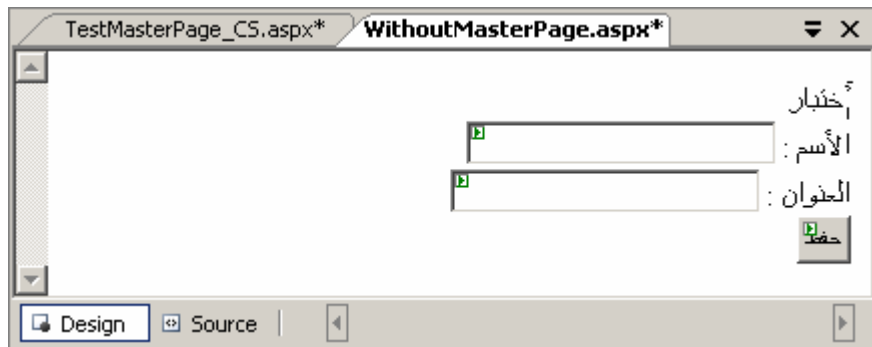


الشكل ٦ قم برسم جدول وحافظ على وجود العنصر ContentPlaceholder



الشكل 7 قم بفتح الصفحة وسوف يظهر لك الشكل كما هو موضح كل ما عليك الآن هو وضع الأدوات سوف تلاحظ أن الأجزاء الخاصة بالMasterPage قد تحول لونة الى درجة داكنه و هذا يعني أنه لا يمكن تعديل هذا الجزء من هذه الصفحة .

و لكن ماذا لو أردنا أن نرث شكل صفحة بدون استخدام معالج الMasterPage لاحظ المثال الآتي



الشكل ٨ هذه الصفحة لا تراث من أى صفحة أم قم بفتح الـ Source الخاص بها

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

<!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" dir="rtl">
<head runat="server">
<title>Untitled Page</title>
</head>
<body>
<form id="form1" runat="server">
<div>
أختبار <br />
الاسم :
<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox><br />
العنوان :
<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox><br />
<asp:Button ID="Button1" runat="server" Text="حفظ" /></div>
</form>
</body>
</html>
```

قم بمسح الكود المظلل كما هو موضح فى المقطع السابق

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="WithoutMasterPage.aspx.cs" Inherits="WithoutMasterPage"
MasterPageFile="~/MasterPage.master" %>

<asp:Content ContentPlaceHolderID=ContentPlaceHolder1 ID=ContentHolder
runat=server><div>
أختبار <br />
```

: الاسم

```
<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox><br />
```

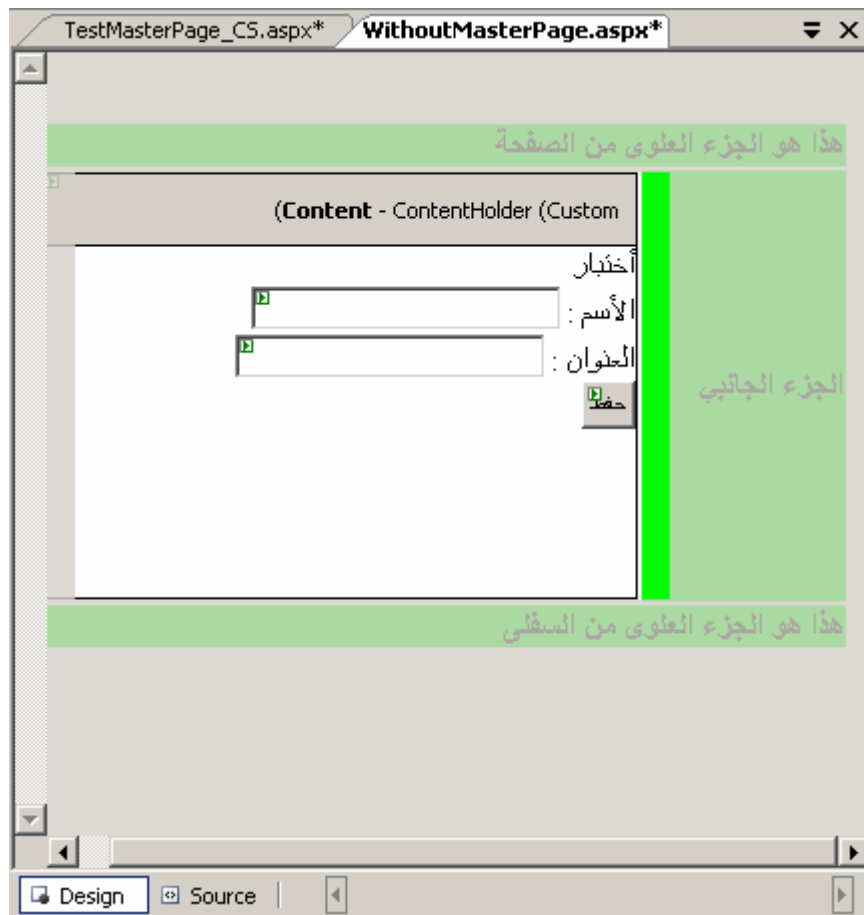
: العنوان

```
<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox><br />
```

```
<asp:Button ID="Button1" runat="server" Text="حفظ" /></div>
```

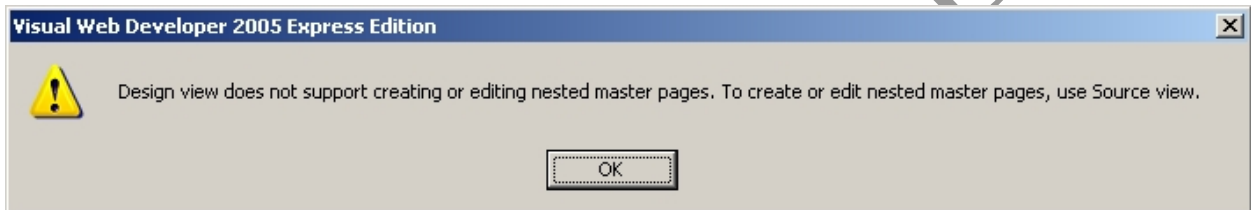
```
</asp:Content>
```

قم بإضافة الكود المظلل و لاحظ الفرق في الشكل التالي



الشكل ٩ هذا هو شكل الصفحة بعد تعديل كود الـ HTML ، الآن أصبحت هذه الصفحة ترث من MasterPage بدون استخدام المعالج،

و الآن لنعرض ميزة أخرى مميزة جدا في الـ **MasterPage** و هي إمكانية وراثة شكل من صفحة أم الى صفحة أخرى بمعنى **Master Page inherits Master Page** وهذا مفيد في بعض الأحيان و لكن اذا استخدم هذه الخاصية سوف تفقد إمكانية تعامل الـ **VS.net** مع الصفحات التي ترث من الصفحة الأم الوارثة من صفحة أخرى و سوف تظهر لك هذه الرسالة،



الشكل 10

بالإضافة الى أنه عند محاولتك لوراثة صفحة أم الى صفحة أخرى لن تتمكن من استخدام المعالج الخاص بالـ **MasterPage** و سوف تضطر الى تعديل الكود يدويا كالآتي

```
<%@ Master Language="C#" AutoEventWireup="true"
CodeFile="MasterPage2.master.cs"
Inherits="MasterPage2"
MasterPageFile="~/MasterPage.master" %>

<asp:Content ContentPlaceHolderID=ContentPlaceHolder1 ID=test runat=server >
    <div>
        <asp:contentplaceholder id="ContentPlaceHolder1" runat="server">
        </asp:contentplaceholder>
    </div>
</asp:Content>
```

كما أننا في سابقا قمنا بأيضاح كيفية عمل وراثة للصفحات يدويا بدون استخدام المعالج.

هناك أيضا إضافة مهمة جدا و هي أنه يمكن عمل **Master Page** افتراضية و التي يتم اختيارها في حالة عدم اختيارك

لـ **Master Page** و هي أنك تقوم في ملف الـ **Web.Config** بإضافة السطر التالي تحت **system.web**

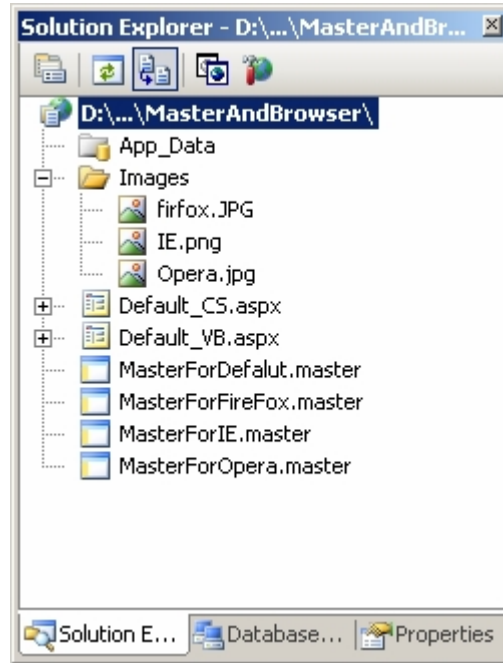
```
<pages masterPageFile="~/WholeSite.master" />
```

هذا السطر سوف يوفر عليك عناء اختيار الصفحة و تحديدها و ما الى ذلك حيث أن هذا الخيار سوف يتيح لك ترك

**MasterPageFile** فارغة بدون قيمة في موجة الصفحة **Page Directive** كما هو في السطر التالي

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

و الآن لنعرض كيفية تغيير الصفحة الأم في لحظة عرض الصفحة الأساسية،  
قم ببدأ مشروع جديد و أضف اليه الملفات كما هو موضح في الشكل



الشكل ١١ ( في الملفات Default قم بأختيار اللغة حسب ما تريد VB أو C# ) لابد و أن تكون الصفحات Default ترث الشكل من  
MasterPage و هي موجودة بأسم MasterForDefault

والان نقوم بعمل ضبط للصفحة الأم كما سبق بطريقة بسيطة جدا و الآن سوف نضيف الكود التالي ليقوم بتغيير الصفحة الأم  
حسب نوع المستعرض

```
protected void Page_PreInit(object sender, EventArgs e)
{
    switch(Request.Browser.Browser.ToLower())
    {
        case "ie":
            MasterPageFile = "MasterForIE.master";
            break;
        case "opera":
            MasterPageFile = "MasterForOpera.master";
            break;
        case "firefox":
            MasterPageFile = "MasterForFireFox.master";
            break;
    }
}
```

```
default :  
    MasterPageFile = "MasterForDefault.master";  
    break;  
}  
}
```

#### الكود بصيغة الـ C#

```
Protected Sub Page_PreInit(ByVal sender As Object, ByVal e As  
System.EventArgs) Handles Me.PreInit  
    Select Case Request.Browser.Browser.ToLower()  
        Case "ie"  
            MasterPageFile = "MasterForIE.master"  
        Case "opera"  
            MasterPageFile = "MasterForOpera.master"  
        Case "firefox"  
            MasterPageFile = "MasterForFireFox.master"  
        Case Else  
            MasterPageFile = "MasterForDefault.master"  
    End Select  
End Sub
```

#### الكود بصيغة الـ Visual Basic

هذا الكود يقوم بتغيير الصفحة الأم قبل تحديد أي صفحة أم سيتم اختيارها وهذا طبعا بعد معرفة نوع المستعرض.



الشكل 12 في حالة المستعرض Opera

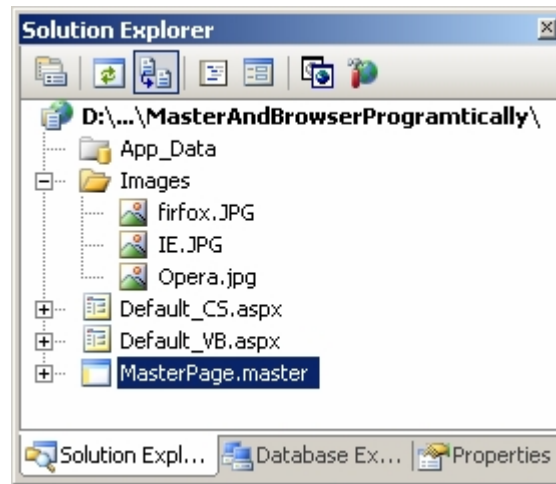


الشكل 13 في حالة المستعرض Firefox

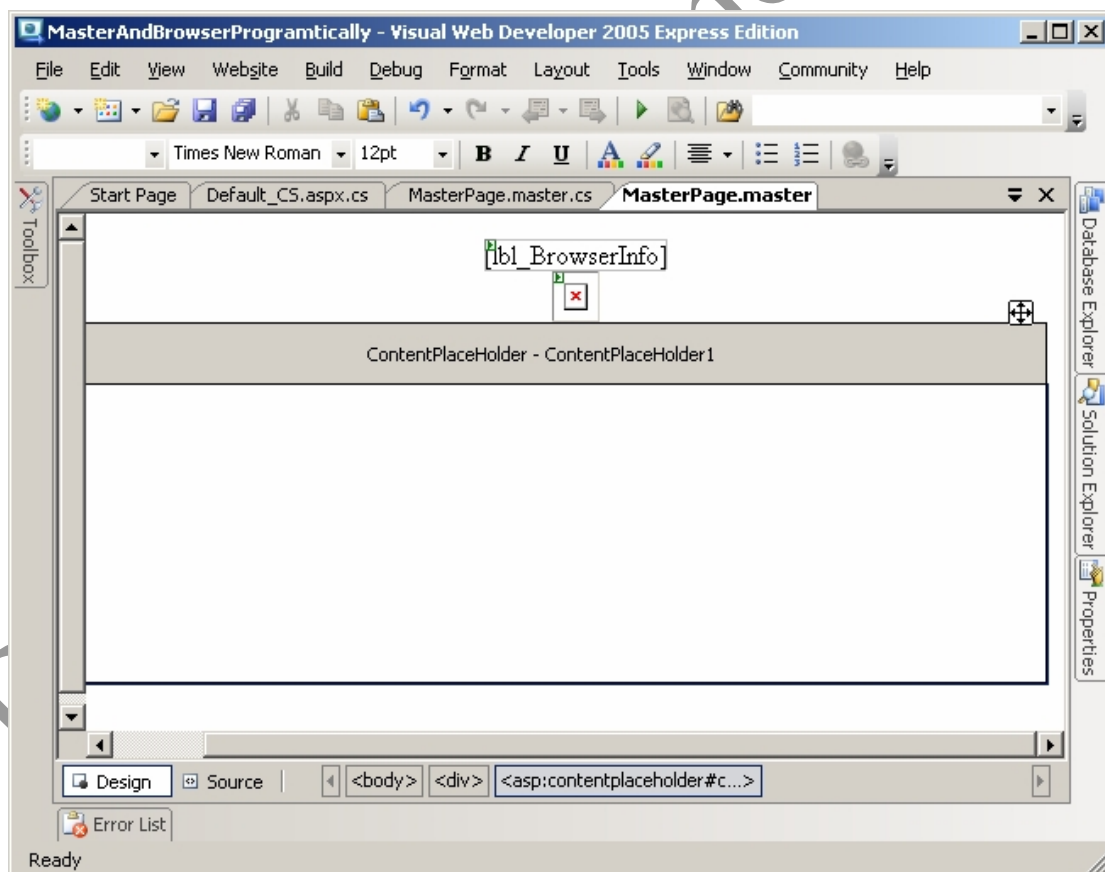


الشكل 14 في حالة استخدام Internet Explorer

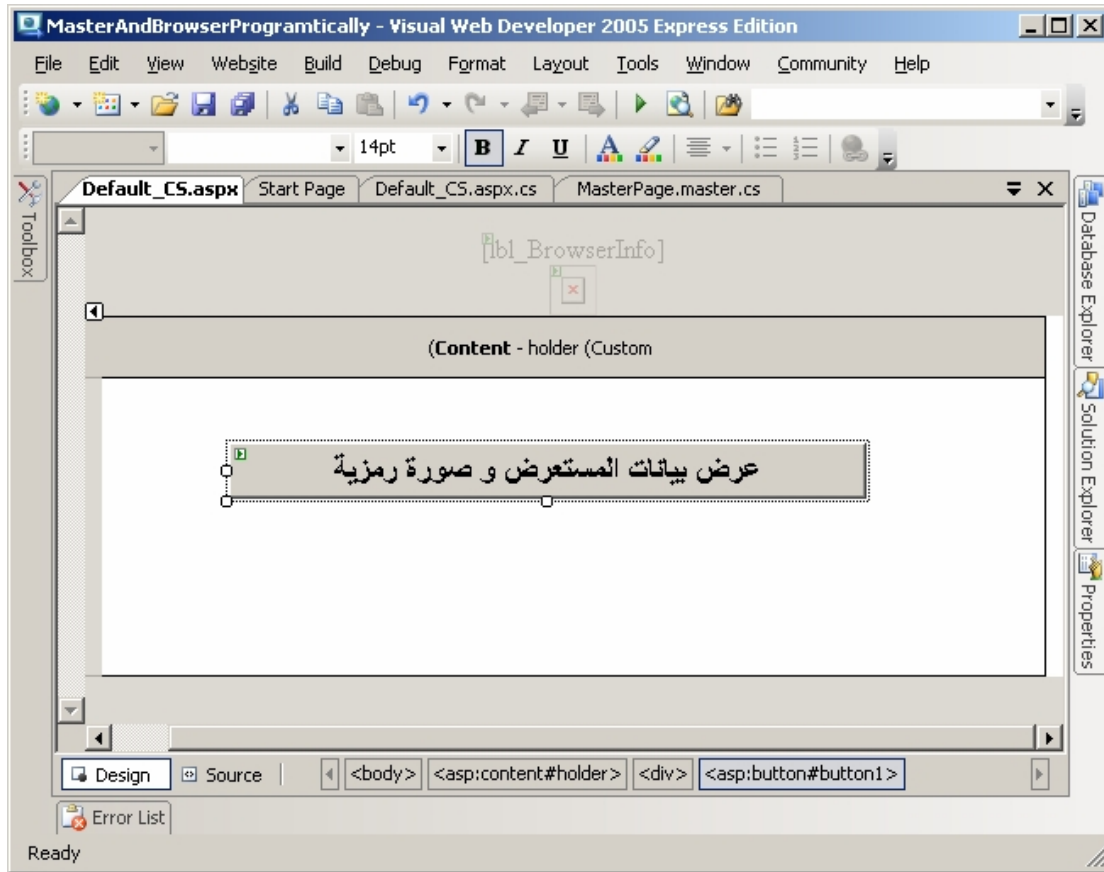
أرى أننا في المثال السابق صعبنا الأمور على أنفسنا بعض الشيء فما رأيكم أن نقوم بتعديل الـ **MasterPage** برمجياً عوضاً عن التعامل بهذا الشكل وهو أن نقوم بعمل صفحة أم لكل مستعرض و الآن نعرض مثالاً، قم ببدأ مشروع جديد و أضف اليه الملفات كما هو موضح في الشكل



الشكل ١٥ أبدأ مشروع جديد و ضع به الملفات الموجودة في الشكل



الشكل 16 في الصفحة الأم قم بإضافة Label و Image لمنطقة العرض و بعد ذلك قم بتغيير أسم الـ Label الى 'lbl\_BrowserInfo' و قم بتغيير أسم الـ Image الى 'img\_browserImage'



الشكل ١٧ قم بالدخول على الصفحة Default\_xx (xx تعني حسب اللغة التي تستخدم) قم بإضافة زر و غير Text الخاص به كما هو موضح

```
protected void Button1_Click(object sender, EventArgs e)
{
    Image imgHolder = (Image)Master.FindControl("img_BrowserImage");
    هنا نقوم بالحصول على العنصر Image في الصفحة الأم و وضعة في متغير من نفس النوع حتى
    يتسنى لنا التعامل معه مثل باقى العناصر.
    Label lblHolder = (Label)Master.FindControl("lbl_BrowserInfo");
    هنا نقوم بالحصول على العنصر Label في الصفحة الأم و وضعة في متغير من نفس النوع حتى
    يتسنى لنا التعامل معه مثل باقى العناصر.
    imgHolder.ImageUrl = "~/\\Images\\" + Request.Browser.Browser + ".jpg";
    نقوم بتغيير مسار الصورة الى الصورة حسب نوع المستعرض و ذلك بدمج مسار مجلد الصور مع
    نوع المستعرض بالإضافة الى امتداد الصورة
    lblHolder.Text = string.Format("<pre>{0}</pre>", Request.Browser.Browser);
    عرض أسم المستعرض
}
```

### الكود بصيغة C#

```
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim imgHolder As Image = CType(Master.FindControl("img_BrowserImage"), Image)
```

هنا نقوم بالحصول على العنصر Image في الصفحة الأم و وضعة في متغير من نفس النوع حتى يتسنى لنا التعامل معه مثل باقى العناصر.

```
Dim lblHolder As Label = CType(Master.FindControl("lbl_BrowserInfo"), Label)
```

هنا نقوم بالحصول على العنصر Label في الصفحة الأم و وضعة في متغير من نفس النوع حتى يتسنى لنا التعامل معه مثل باقى العناصر.

```
imgHolder.ImageUrl = "~\Images\" + Request.Browser.Browser + ".jpg"
```

نقوم بتغيير مسار الصورة الى الصورة حسب نوع المستعرض و ذلك بدمج مسار مجلد الصور مع نوع المستعرض بالإضافة الى امتداد الصورة

```
lblHolder.Text = String.Format("<pre>{0}</pre>", Request.Browser.Browser)
```

عرض اسم المستعرض

End Sub

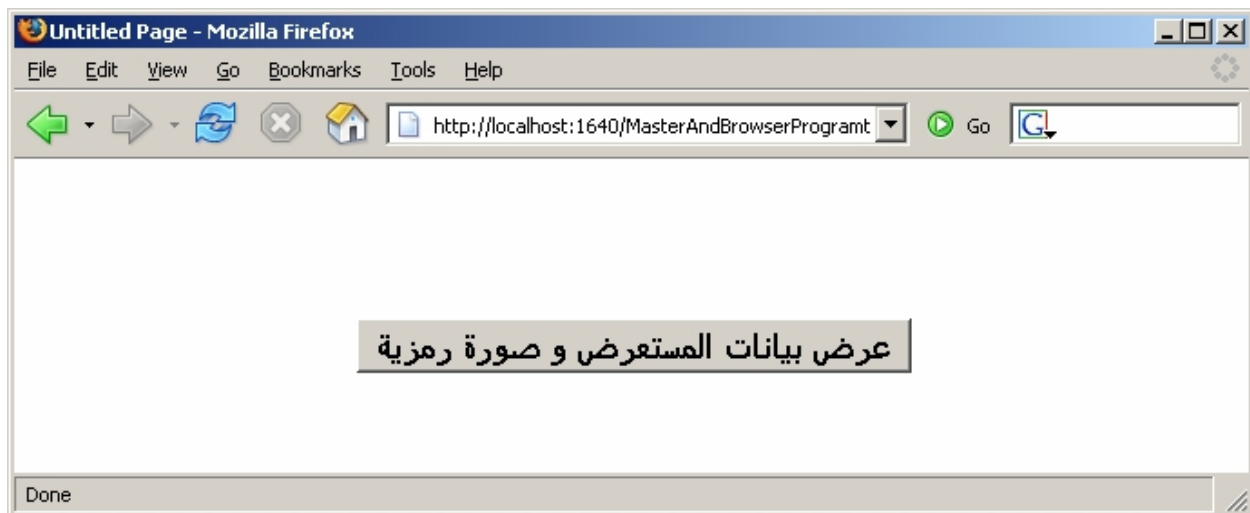
### الكود بصيغة VB.net



الشكل 18 شكل ناتج المثال على مستعرض انترنت اكسبلورر و ذلك قبل الضغط على "عرض بيانات المستعرض"



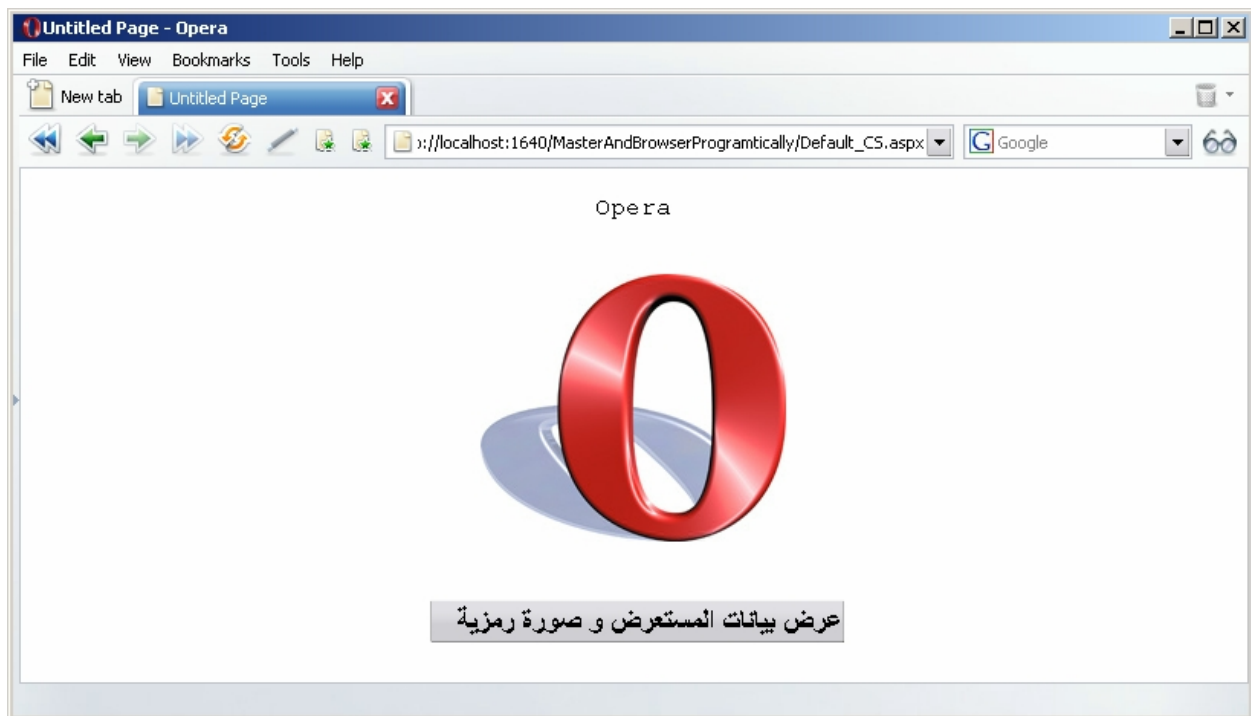
الشكل 19 شكل ناتج المثال على مستعرض أوبرا و ذلك قبل الضغط على "عرض بيانات المستعرض"



الشكل 20 شكل ناتج المثال على مستعرض فايرفوكس و ذلك قبل الضغط على "عرض بيانات المستعرض"



الشكل 21 شكل ناتج المثال على مستعرض إنترنت إكسبلورار و ذلك بعد الضغط على "عرض بيانات المستعرض"



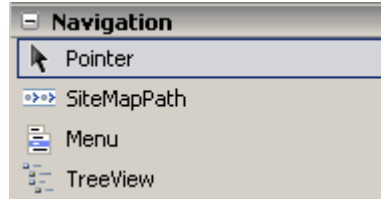
الشكل 22 شكل ناتج المثال على مستعرض أوبرا و ذلك بعد الضغط على "عرض بيانات المستعرض"



الشكل 23 شكل ناتج المثال على مستعرض فايرفوكس و ذلك بعد الضغط على "عرض بيانات المستعرض"

## أدوات الملاحة Navigation Tools

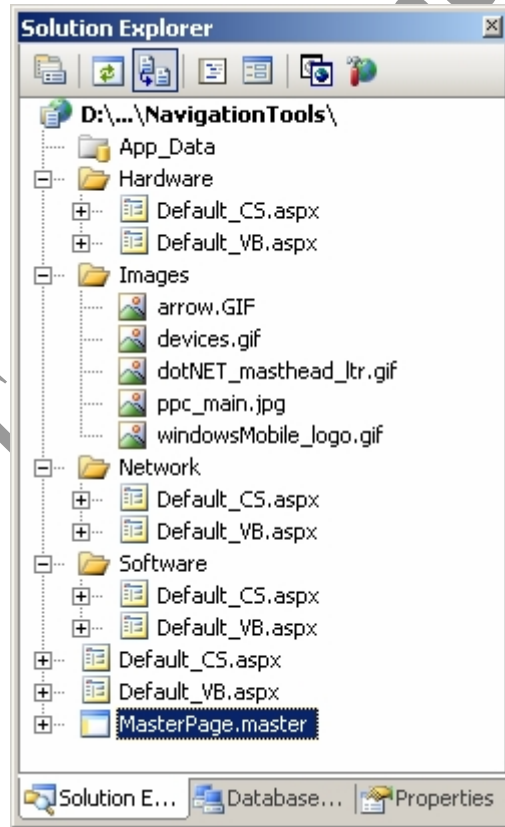
و الآن و بعد الانتهاء من سرد كيفية التعامل مع الـ **Master Page** ننتقل الى كيفية الربط و التعامل مع الـ **Navigation tools** و هي من أحد المزايا التي توفر الكثير من الوقت و المجهود و الآن لنعرض أدوات الملاحة الجديدة في **VS.net 2005** و هي موضحة في الشكل التالي



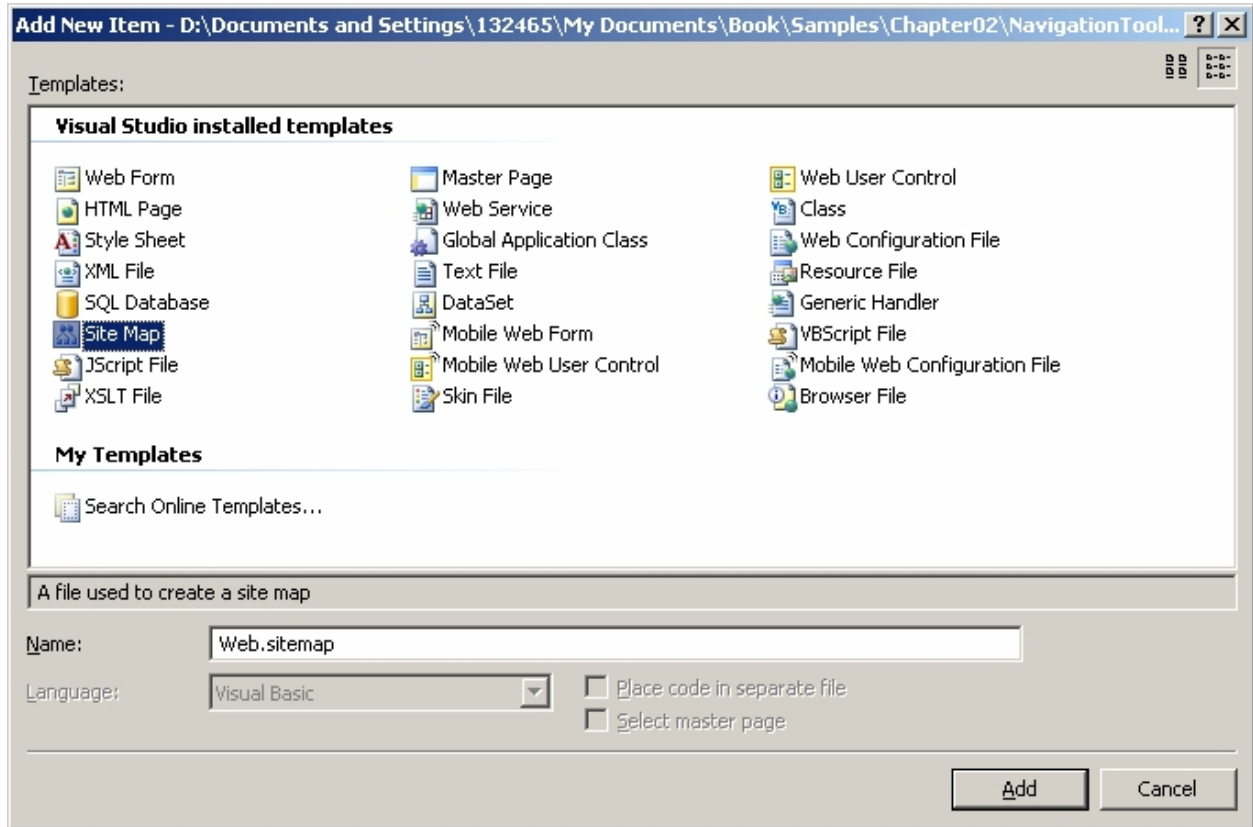
الشكل 24

والآن سوف نشرح هذه الأدوات واحدة تلو الأخر فلا تقلق وسوف يكون كالعادة أمثلة توضح كيفية التعامل مع هذه الأدوات و الآن لنبدأ بالمثال الأول مع الأداة **SiteMapPath** و **Menu** حيث كلا منهما يكمل الآخر، هما ليس من الواجب أن يكون الأداة مستعمتان في نفس الوقت و لكن كشكل يفضل وجودهما معا و هذا من وجه نظري الشخصية، و الآن لنعرض المثال.

أولا نقوم بأنشاء مشروع جديد و إضافة ملفات له كما هو موضح في الشكل التالي



الشكل ٢٥

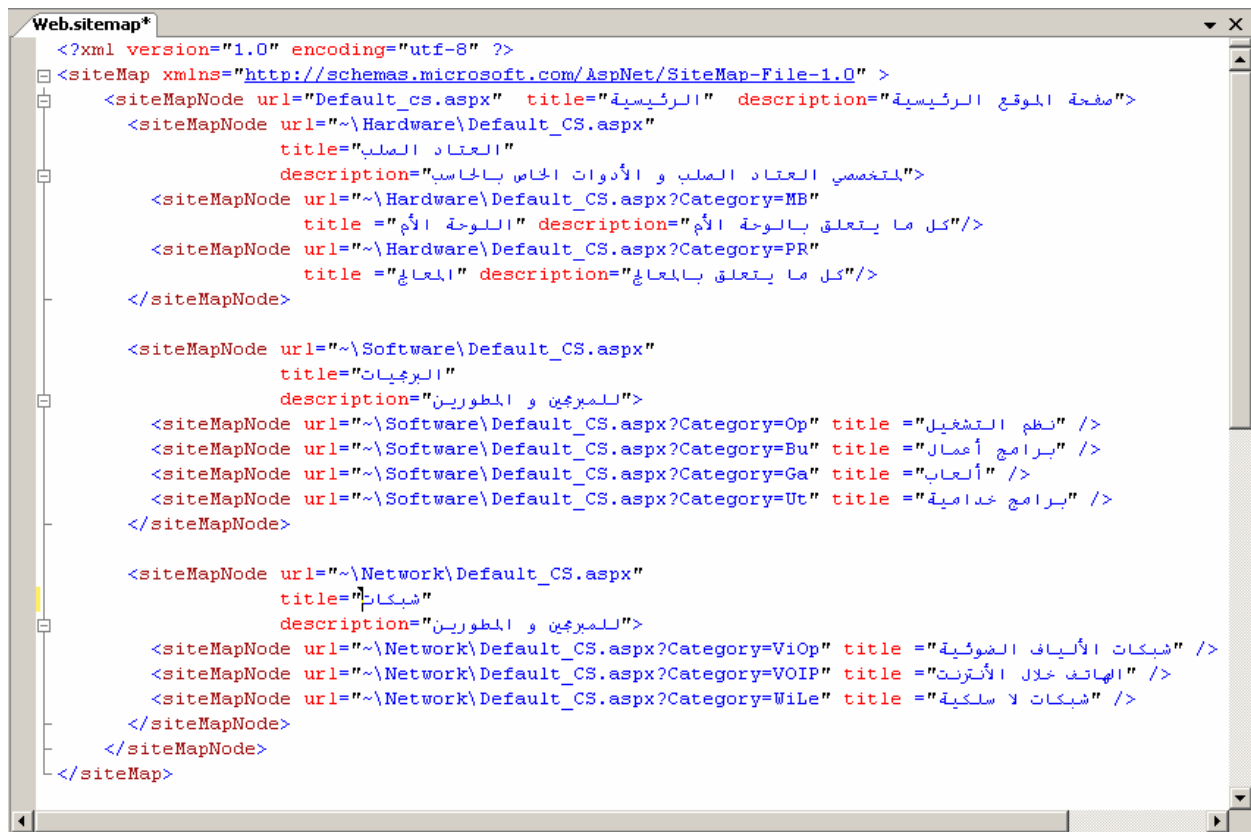


الشكل ٢٦ نقوم بأضافة عنصر جديد و هو Site Map من القوالب الموجودة Templates

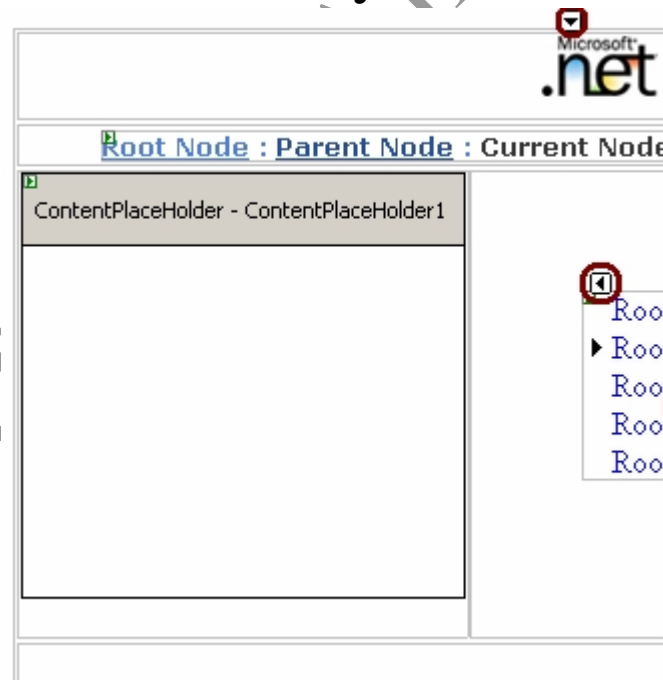
فائدة الـ Site Map :

هذا الملف هو الدليل الذي يكون به جميع وصلات المشروع أو كما يسمى خارطة الموقع بعد اختيارك للملف Site Map سوف تحتاج الى إدخال بيانات كل وصلة في الموقع الخاص بك و هي كالآتي URL و فيها عنوان الصفحة Title العنوان الذي سوف يظهر

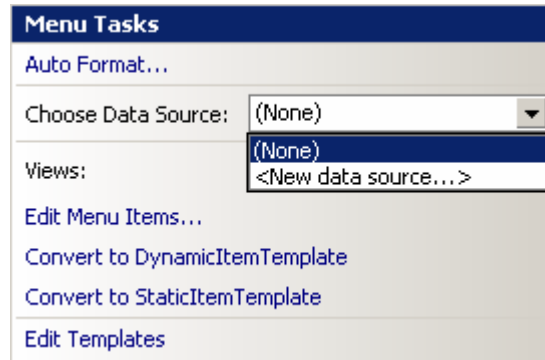
Description وصف مبسط عن الوصلة و ما تحتوية و الوصف يظهر في شكل Tip Roles و فيها يحدد من له الحق في أن يرى هذه الوصلة (سيتم شرح هذه النقطة في الفصل الخاص بالأمن). بعد أن قمت بأضافة الملف قم بأضافة الوصلات الخاصة بموقعك كما هو الحال في الشكل التالي .



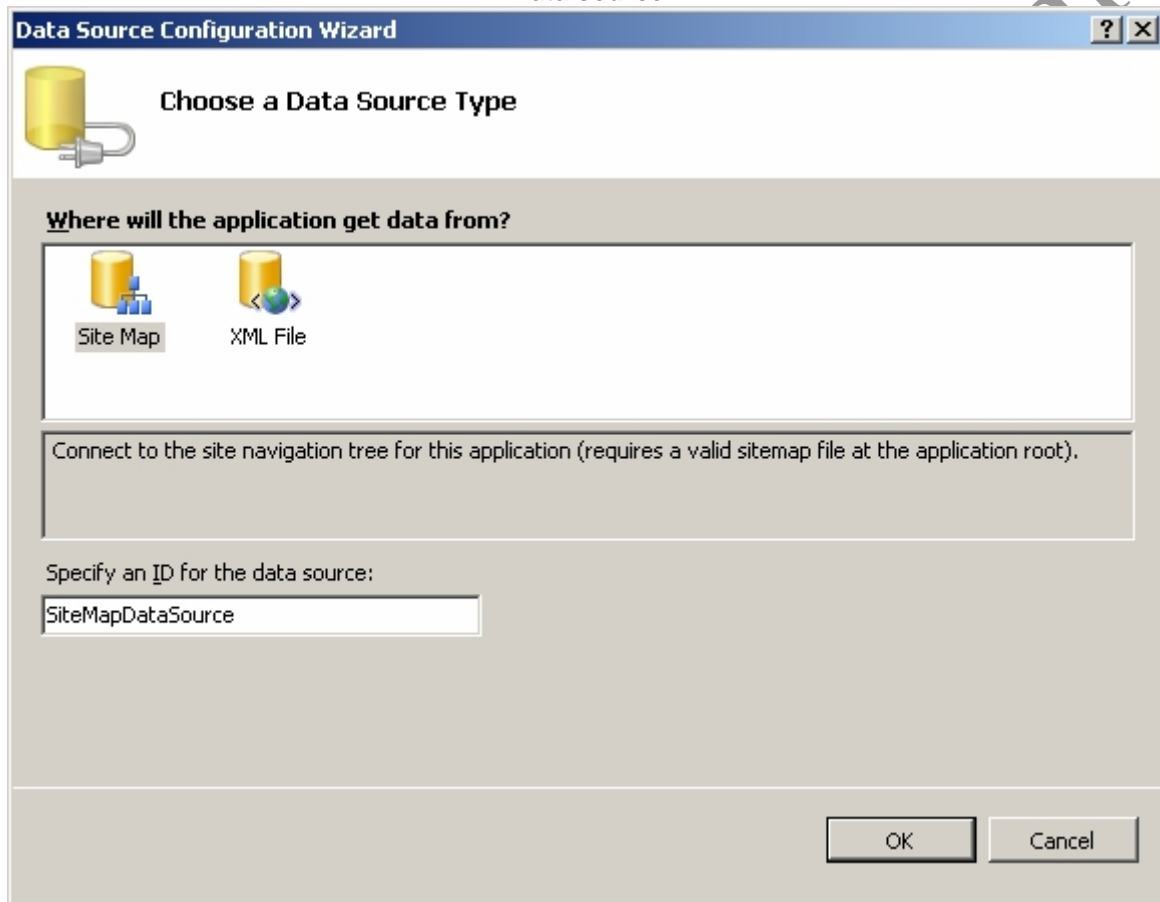
الشكل ٢٧



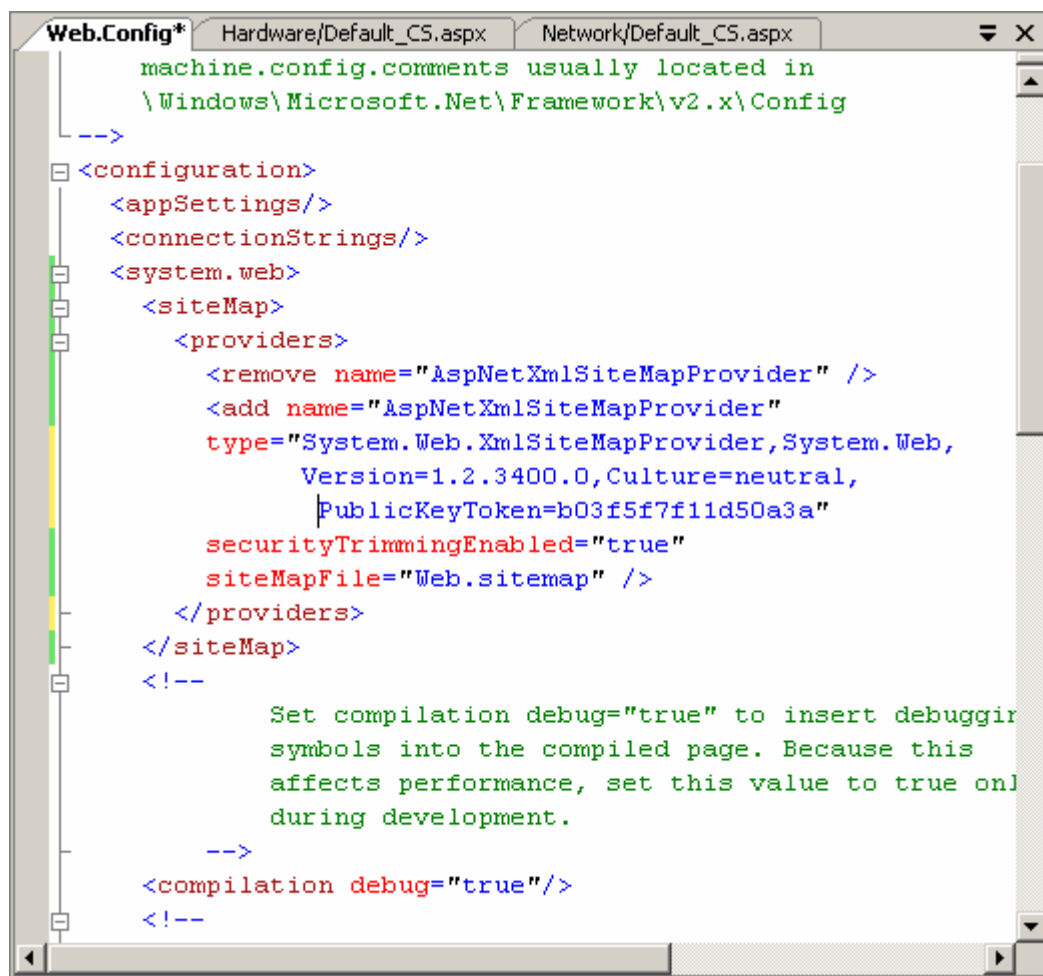
الشكل ٢٨ يعد ذلك نقوم بإضافة بعض العناصر للصفحة الأم Master Page و هي كما نرى جدول و به Site Map Path و Menu و الـ Content Place Holder، الدوائر الحمراء تشير الى ما يسمى بالمساعد الذكي Smart Tag حيث عن الضغط عليها تظهر لك قائمة بأهم الخواص للعنصر التابع له.



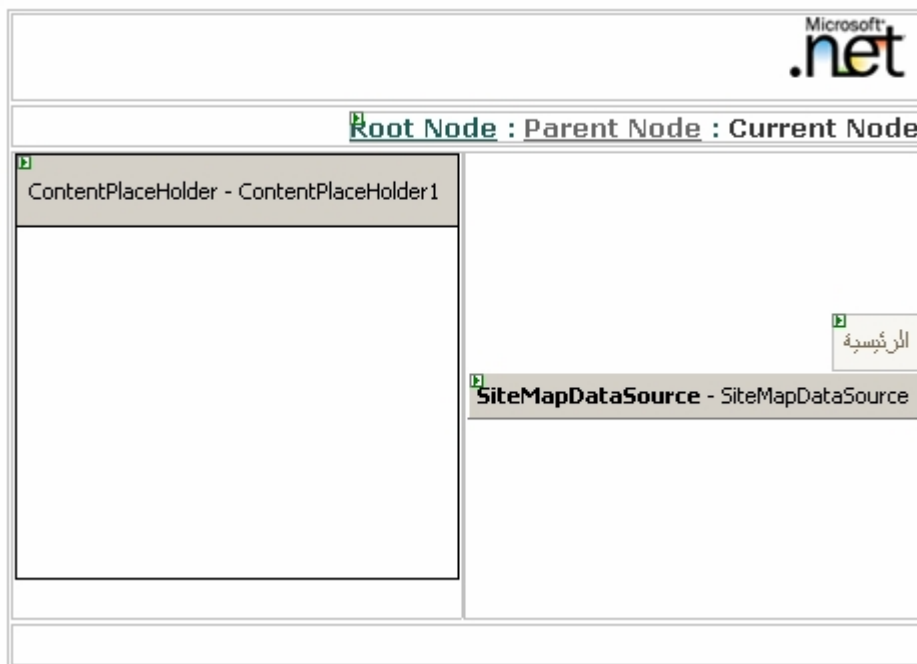
الشكل ٢٩ عند الضغط على الـ Smart Tag الخاص بالـ Menu نرى هذا الشكل نقوم الآن بتحديد مصدر البيانات للقائمة و سوف نختار **New Data source**



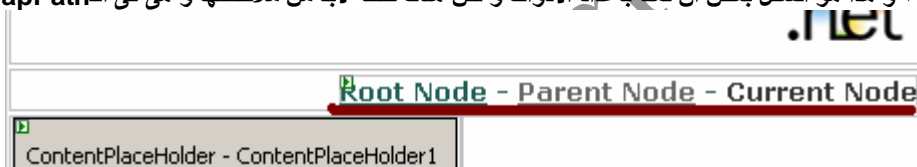
الشكل ٣٠ عند الاختيار سوف تظهر هذه النافذة و التي تختار منها نوع مصدر البيانات التي سوف تستخدمها القائمة و سوف نختار الـ **Site Map**



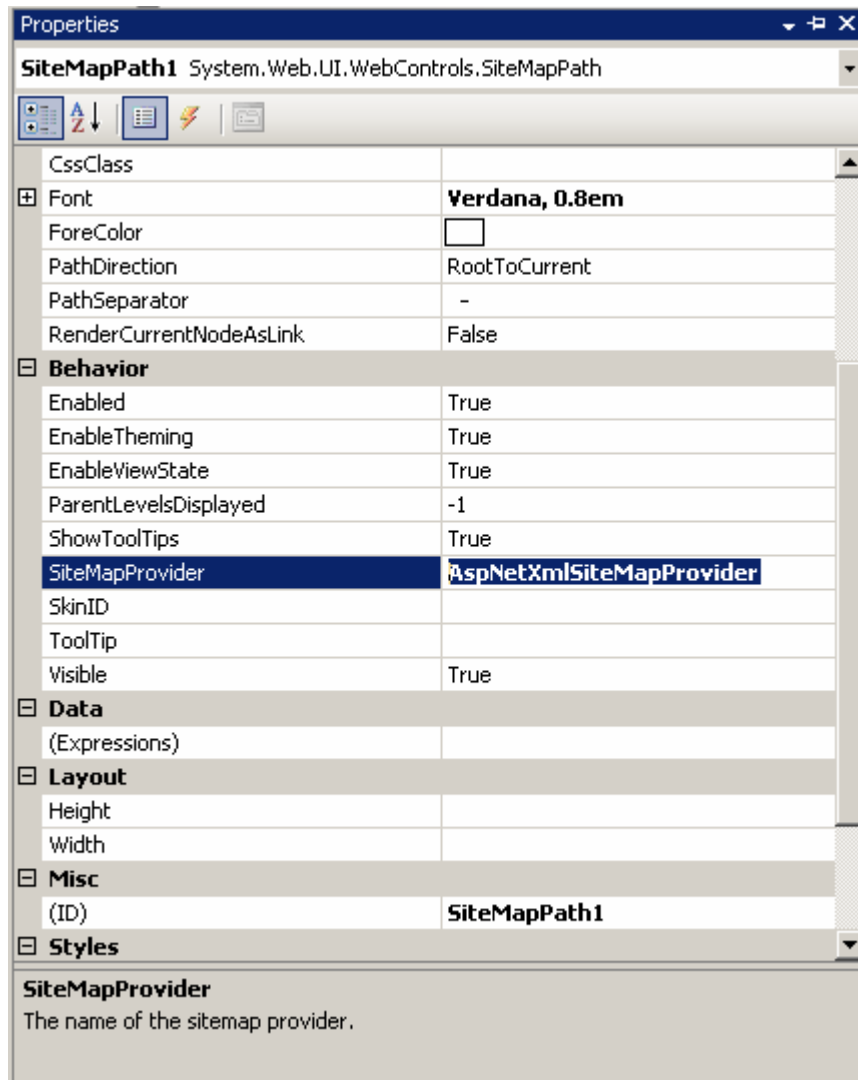
الشكل ٣١ بعد ذلك نقوم بالدخول على الملف Web.Config و سوف نضيف بعض السطور فيه و هذه السطور لتعريف ما يسمى بالمخصص Provider و هنا يتم تعريف بعض المكتبات بالإضافة الى معلومات أخرى الشئ المتغير الوحي فيها هو الاسم Name و الملف الخارطة sitemapfile كما نرى في الشكل.



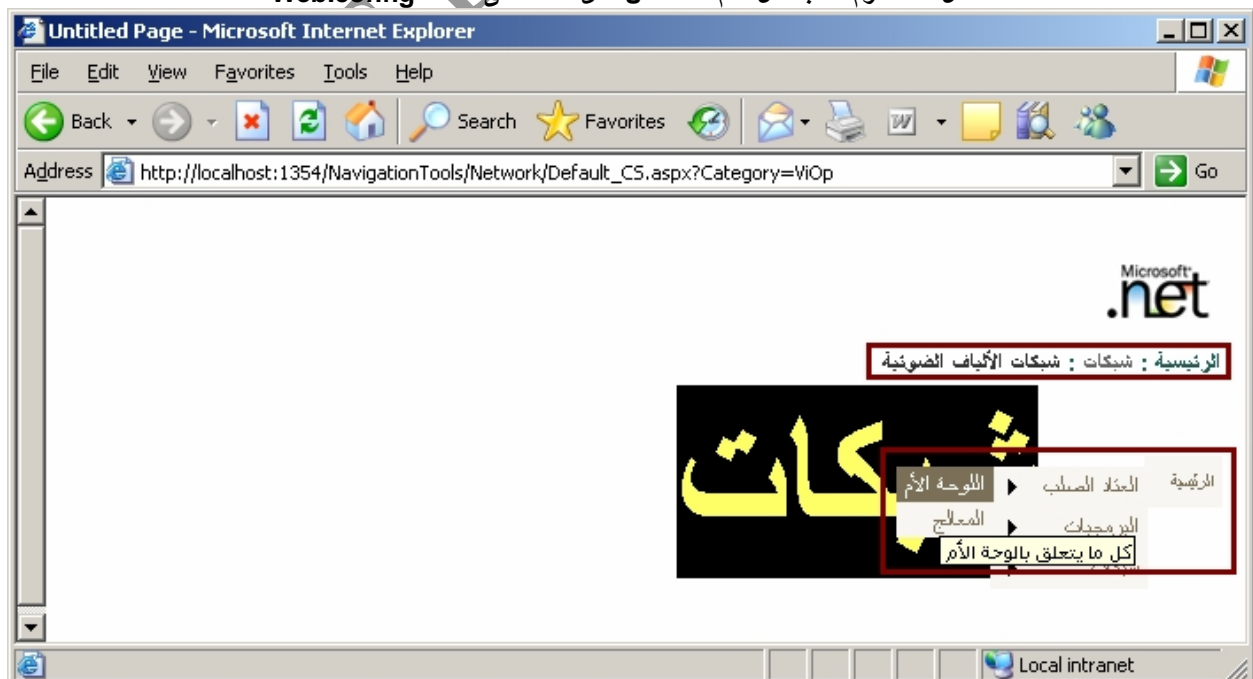
الشكل ٣٢ و هذا هو الشكل بعض أن قمنا بأعداد الأدوات و لكن هناك نقطة لابد من ملاحظتها و هي في الـ `siteMapPath`



الشكل ٣٣ كما نرى نقوم بتفعيل العنصر `siteMapPath`

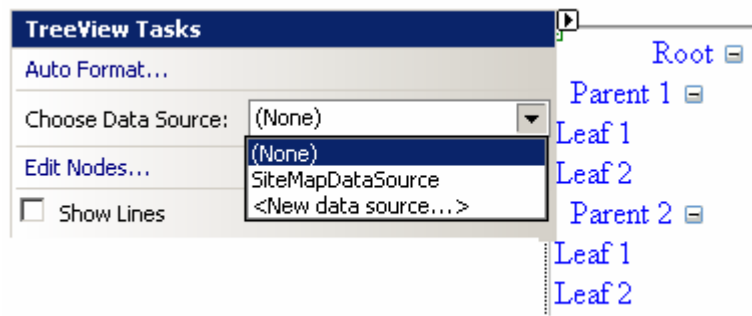


الشكل ٣٤ نقوم هنا بأدخال أسم المخصص الذي أدخلناه في ملف Web.config

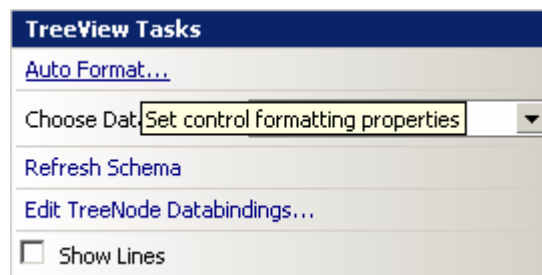


الشكل 35 هذا هو الشكل يعد أن قمنا بضبط جميع الإعدادات الخاصة بهذه الأدوات.

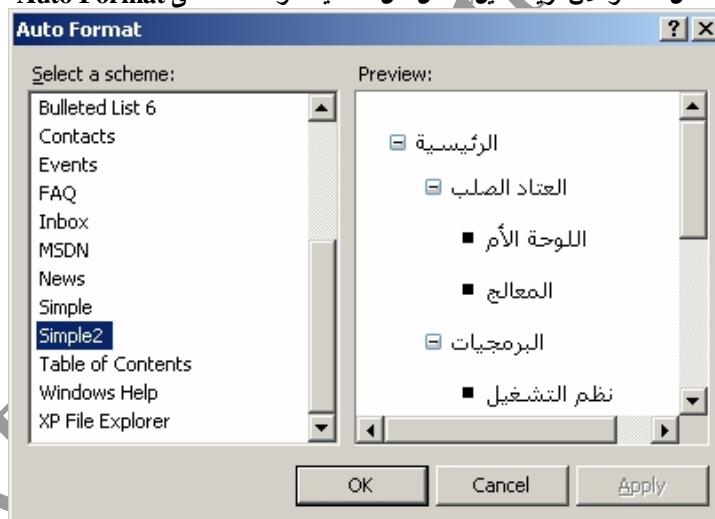
## و الآن لنر المثال مع الأداة TreeView



الشكل ٣٦ نقوم بوضع العنصر TreeView و بعد ذلك نقوم بالضغط على الـ Smart Tag الخاص بها و نختار SiteMapDataSource و نحن الآن نكمل على المثال السابق حيث أننا سوف نستغل عنصر جلب البيانات المستخدم في العنصر Menu



الشكل ٣٧ و الآن نريد تعديل الشكل، كل ما عليك هو الضغط على Auto Format



الشكل ٣٨ أختار الشكل الذي تريده و من ثم OK



الشكل 39 الشكل النهائي للمثال.

و في النهاية هذا الفصل أود أن نذكر سوياً ملخصاً لما تعلمناه في هذا الفصل:

١. التعامل مع الصفحة الأم بالمعالج و يدوياً و برمجياً
٢. التعامل مع أدوات الملاحة

a. SiteMapPath

b. Menu

c. TreeView

d. SiteMapDataSource