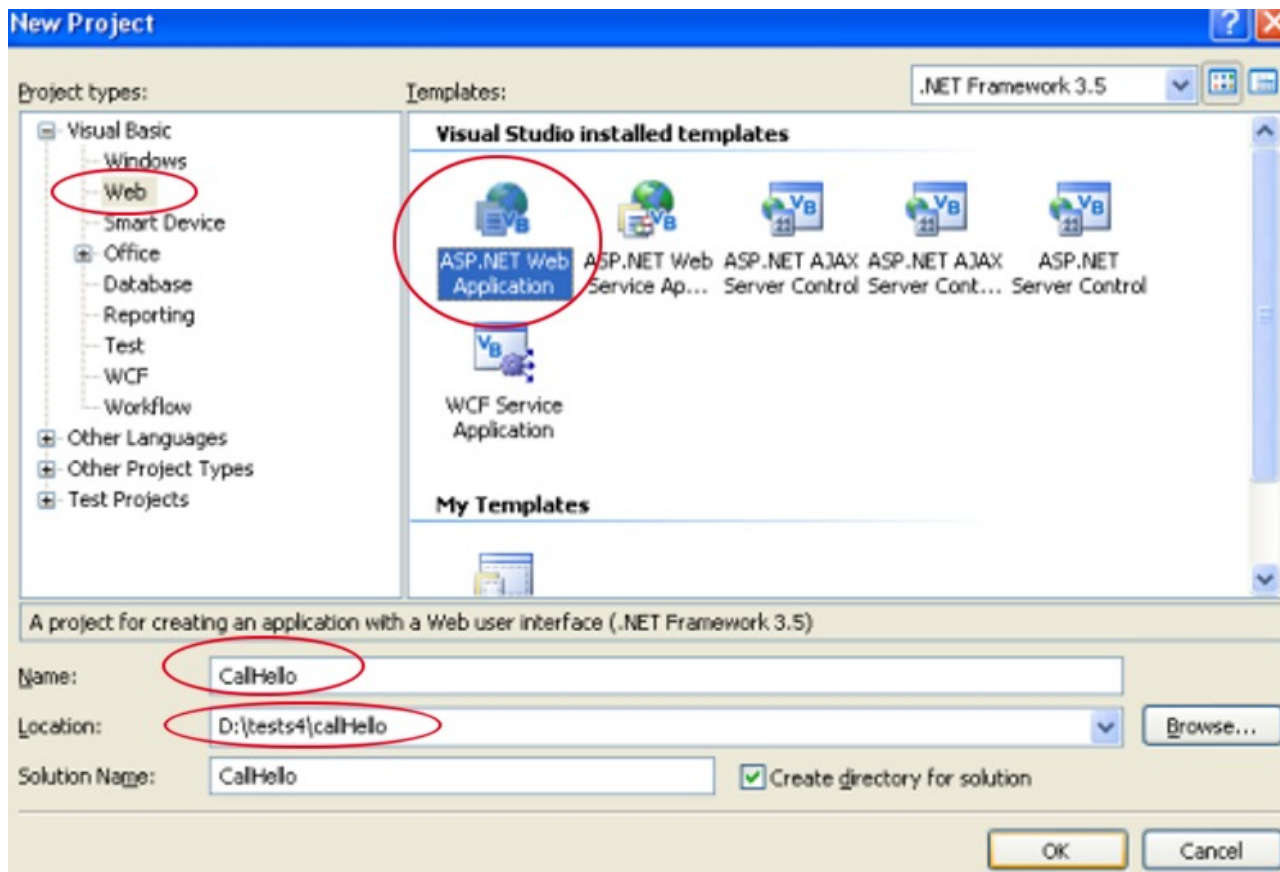


WebService naudojimas Asp.Net aplinkoje

Anksčiau aprašėme, [kaip sukurti paprastą WebService](#), apie [papildomas WebService galimybes](#), o taip pat kaip jį [panaudoti ASP puslapiuose](#). Dabar aprašysime, kaip iškviešti *WebService* metoda iš .Net aplinkos (atskiru, ir čia aptariamu atveju, iš **Asp.Net** puslapio, numatančio *Web* formą, *WebService* parametrų įvedimui).

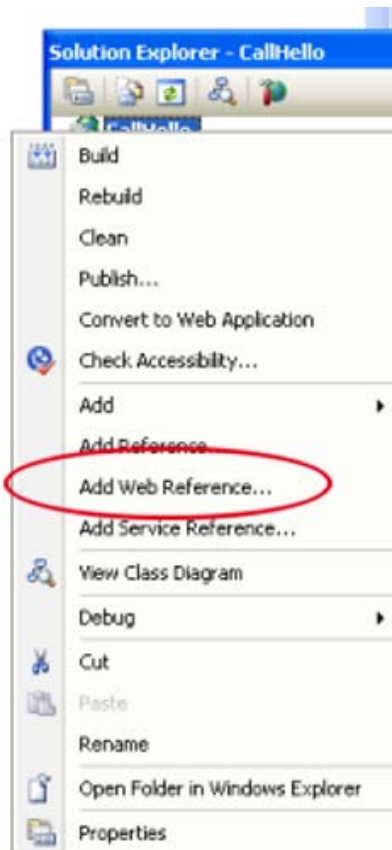
Pirmiausia susikurkime naują projektą mūsų mažam pavyzdėliui. „Visual Studio“ meniu pasirenkame *File -> New Project...* Tada projektų tipų srityje išskleidžiame *Visual Basic* grupę ir pasirenkame *Web* tipą. Iš *Templates* pasirenkame **ASP.NET Web Application**. Lauke *Name*: nurodome mūsų sprendimo pavadinimą (tebūnie *CallHello*), o taip pat katalogą, kuriame bus išsaugotas mūsų sprendimas (*Location*: laukas). Paspaudžiame OK mygtuką (žr. iliustraciją):



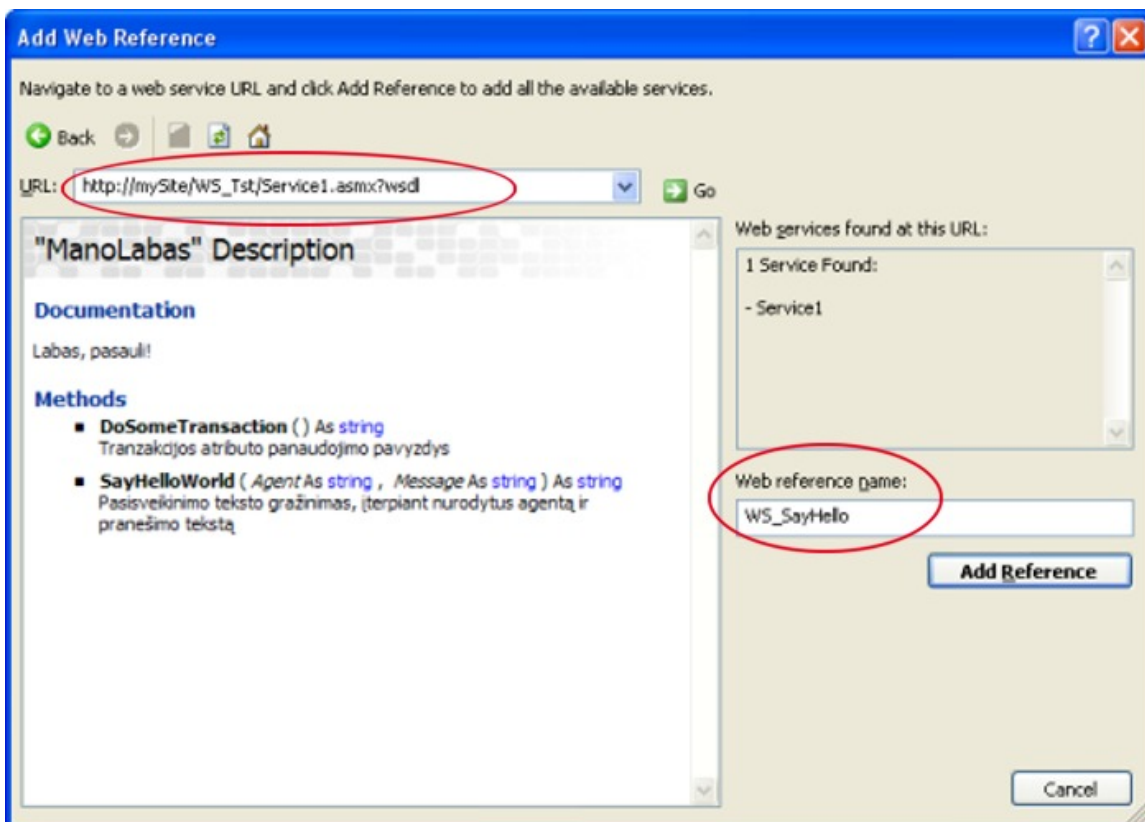
Pagal nutylėjimą sukūrė ASP.Net sprendimą, turintį **Default.aspx** puslapį. Kodo rodinyje (įjungiamas ekrano apačioje esančioje veiksmų juostoje paspaudus *Source*; o greta esantis *Split* mygtukas padalija ekraną ir leidžia vienu metu matyti ir kodą, ir puslapio vizualų vaizdą, t.y. *Design*) pirmojoje eilutėje pašalinkite atributą **CodeBehind="Default.aspx.vb"** – tai nurodys, kad savo sprendime nenaudojame jokio papildomo failo kodui. Taip pat mūsų puslapį pervardinkime jį į **CallHello.aspx**. Žr. iliustraciją šiems veiksams:



Tada *Solution Explorer* srityje (jei jo nėra, jis įjungiamas meniu *View -> Solution Explorer*) pasirenkame *WebService* nuorodos įtraukimo į projektą priemonę (meniu atsiveria dešinio pelės klavišo spragtelėjimu virš projekto **CallHello** pavadinimo):



Dialogo lange **URL:** lauke surenkame nuorodą į *WebService*, kurį ketiname naudoti. Mes naudosime [ankstesniame straipsnyje aprašytą](#) *WebService*, turintį *SayHelloWorld* metodą. Lauke **Web reference:** įvedame nuorodos į *WebService* pavadinimą (ją naudosime savo kode, kai reikės kreiptis į *WebService*) – mūsų pavyzdyje tai **WS_SayHello**. Paspaudžiame **Add Reference** mygtuką.



Solution Explorer srityje atsiranda nuoroda į *WebService*, tad galime užprogramuoti jos naudojimą. Puslapyje sukuriame formą, kurioje įvesime kreipinio į *WebService* parametrus, o taip pat *WebService* iškvietimą, kai bus paspaustas mygtukas **Kviesti**. Formą apdoros tas pats **CallHello.aspx** puslapis. Puslapio sukūrimui galite panaudoti ir vizualaus kūrimo priemones, tačiau galutinis jo kodas turėtų atrodyti maždaug taip:

```
<%@ Page Language="vb" AutoEventWireup="false" Inherits="CallHello._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>WebService naudojimo pavyzdys</title>
</head>
<body>
<%
    Dim sResponse As String = ""
    Dim sAgent As String, sText As String

    sText = Request.Form("CallWS") & ""
    If sText.Length > 0 Then ' Paspaustas submit mygtukas
        sAgent = (Request.Form("fAgent") & "").Trim
        sText = (Request.Form("fText") & "").Trim
        If sAgent.Length + sText.Length > 0 Then ' ar bent vienas laukas užpildytas
            ' Kviečiamas Webservice
            Dim WS As CallHello.WS_SayHello.ManoLabas = New CallHello.WS_SayHello.ManoLabas
            sResponse = WS.SayHelloWorld(sAgent, sText)
            WS = Nothing
        Else
            sResponse = "Nepateikti duomenys!"
        End If
    End If

    %>
<h2>WebService kvietimas</h2>
<p><%=sResponse%></p>
<form id="WS_call_Form" method='post' action='CallHello.aspx'>
<table border='0'><tr><td> Agentas:</td><td><input name='fAgent' size='66' /></td></tr>
<tr><td>Tekstas:</td><td><input name='fText' size='66' /></td></tr>
<tr><td>&nbsp;</td><td><input type='Submit' name='CallWS' value='Kviesti' /></td></tr>
</table>
</form>
</body>
</html>
```

Trumpi paaiškinimai:

Kodą sudaro dvi pagrindinės dalys: apatinėje dalyje yra formos aprašas (<form ..> ... </form>) – realizuojantis formą, kurios išvaizda pateikta žemiau. Viršutinėje dalyje, šios formos apdorojimo dalis.

`Request.Form("CallWS")` paima „submit“ mygtuko reikšmę, kuri lygi „Kviesti“ (ir tuo pačiu jos ilgis nelygus 0) tuo atveju, kai buvo paspaustas „Kviesti“ mygtukas. Šiuo atveju patikriname, ar bent vienas formos laukų nėra tuščias (čia mūsų papildomas reikalavimas ©), ir jei bent vienas turi reikšmę, su tais parametrais iškviečiamas *WebService* metodas *SayHelloWorld*.

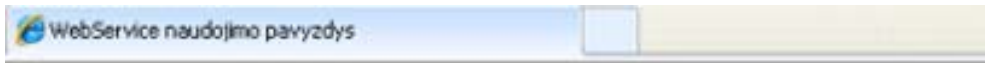
WebService objektas aprašomas ir sukuriamas eilutėje

```
Dim WS As CallHello.WS_SayHello.ManoLabas = New CallHello.WS_SayHello.ManoLabas
```

Čia **CallHello** – mūsų projekto pavadinimas, **WS_SayHello** – nuorodos į *Webservice* pavadinimas (aprašyta prieš tai), o **ManoLabas** – *WebService* suteiktas pavadinimas (žr. jo [kodą su paaiškinimais](#)). Panaudojant aprašytą **WS** objektą, kviečiame metodą, nurodymdami parametrų reikšmes:

```
sResponse = WS.SayHelloWorld(sAgent, sText)
```

Viskas labai paprasta. Belieka išbandyti, kaip veikia mūsų sprendimas. Tai galime padaryti, net nepublikuodami jo. **Ctrl-F5** (vykdyti) iškvies langą su naršykle, pateikiančia mūsų formą, kurioje įveskime laukų reikšmes:



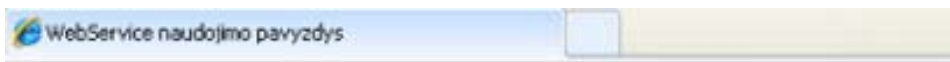
WebService kvietimas

Hello Betmanas aa Kreipinys:1

Agentas:

Tekstas:

Paspaudus **Kviesti** mygtuką, iškviečiamas *WebService* metodas ir jo atsakymas pateikiamas tarp puslapio pavadinimo ir formos:



WebService kvietimas

Hello Betmanas [kovą su blogiu! Kreipinys:1

Agentas:

Tekstas:

Pastaba: Kreipinio skaitliuko reikšmė visada lygi **1**, nes naršyklė kaskart pradeda naują seansą su *Webservice*.

Sėkmės eksperimentuojant toliau!