Lab 4 – WebServices

Web Services can convert your application into a Web-application, which can publish its function or message to the rest of the world.

Web Services are published, found, and used through the Web.

What are Web Services?

- Web services are application components.
- Web services communicate using open protocols.
- Web services are self-contained and self-describing.
- Web services can be discovered using UDDI.
- Web services can be used by other applications.
- XML is the basis for Web services.

How Does it Work?

The basic Web services platform is XML + HTTP.

XML provides a language which can be used between different platforms and programming languages and still express complex messages and functions.

The HTTP protocol is the most used Internet protocol.

Web services platform elements:

- SOAP (Simple Object Access Protocol) - is an XML-based protocol to let applications exchange information over HTTP.
- WSDL (Web Services Description Language) - is an XML-based language for locating and describing Web services.
- UDDI (Universal Description, Discovery and Integration) - is a directory service where companies can register and search for Web services.

By using Web services, your application can publish its function or message to the rest of the world.

Web services use XML to code and to decode data, and SOAP to transport it (using open protocols).

Web Services have Two Types of Uses

Reusable application-components.

There are things applications need very often. So why make these over and over again?

Web services can offer application-components like: currency conversion, weather reports, or even language translation as services.

Connect existing software.

Web services can help to solve the interoperability problem by giving different applications a way to link their data.

With Web services you can exchange data between different applications and different platforms.

Създаване на Уеб услуга

Follow the steps to make a new project:

1. File -> New Project -> Visual C# / Web -> ASP.NET Web Service Application and name it **myService**
2. Delete the file Service1.asmx.cs, and then open Service1.asmx (You can see them in Solution Explorer)
3. From the first line delete CodeBehind="Service1.asmx.cs"
4. On new line paste the code:

```
using System;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;

namespace myService
{

    [WebService(Namespace = "http://localhost/")]

    public class Service1 : System.Web.Services.WebService
    {
        [WebMethod]
        public string HelloWorld()
        {
            return "Hello World";
        }

        [WebMethod]
        public int Add(int a, int b)
        {
            return a + b;
        }

        [WebMethod]
        public string FahrenheitToCelsius(string Fahrenheit)
        {
            double fahr = double.Parse(Fahrenheit);
            return (((fahr - 32) / 9.0) * 5.0).ToString();
        }

        [WebMethod]
        public string CelsiusToFahrenheit(string Celsius)
        {
            double cel = double.Parse(Celsius);
            return (((cel * 9.0) / 5.0) +32).ToString();
        }
    }
}
```

Click the Save button. Open your project files and copy Servise1.asmx file to the folder C:\inetpub\wwwroot.
Your Web Service is ready and is published on the localhost server. In order to see it and test it open this address in your browser http://localhost/Service1.asmx.

Използване на уеб услугите в ASP.NET страница.

Make a new ASP.NET web site by following the steps:

1. File -> New Web Site -> ASP.NET Web Site
2. Delete the file Default.aspx.cs
3. Open Default.aspx and between <div> ….. </div> paste the code:

```
    <p>
        <em>First Number to Add </em>:
        <asp:TextBox ID="txtNum1" runat="server" 
Width="43px">4</asp:TextBox>
    </p>
    <p>
        <em>Second Number To Add </em>:
        <asp:TextBox ID="txtNum2" runat="server" 
Width="44px">5</asp:TextBox>
    </p>
    <p>
        <strong><u>Web Service Result -</u></strong>
    </p>
    <p>
        <em>Hello world Service</em> :
        <asp:Label ID="Label1" runat="server" Font-
Underline="True">Label</asp:Label>
    </p>
    <p>
        <em>Add Service</em> :  <asp:Label ID="Label2" 
runat="server" Font-Underline="True">Label</asp:Label>
    </p>
    <p align="left">
        <asp:Button ID="runSrvice" OnClick="runSrvice_Click" 
runat="server" Text="Execute">
        </asp:Button>
    </p>
```

To use our web service we need to add a reference to it:

1. Right click in Solution Explorer on your project -> **Add Service Referrence**...
2. In the Address bar we enter the address of our Service (which this time is **http://localhost/Service1.asmx**) and click the **Go** button. Ater it is successfully found

click OK. The referrence we made now is called ServiceReference1, and we will use it to control the service.

Add this script to your web page:

```
<script runat="server">
    void runSrvice_Click(Object sender, EventArgs e)
    {
        ServiceReference1.Service1SoapClient mySvc = new
ServiceReference1.Service1SoapClient();
        Label1.Text = mySvc.HelloWorld();
        Label2.Text = mySvc.Add(Int32.Parse(txtNum1.Text),
Int32.Parse(txtNum2.Text)).ToString();
    }
</script>
```

Start the application with CTRL+F5.