

Övning 2 - Webb & Multimediadesign.

Uppgift: Hantera gästbok genom flash och asp.net (del 1/2)

I denna övning kommer flash anropa en Generic Handler på en webbplats som kommer hantera XML genom c#. Flash hämtar sedan informationen från den uppdaterade XML-filen.

Steg 1:

Skapa en ny tom webbsida i Visual Studio 2010 och lägg till följande tre filer genom *add* i solution explorer; Default.aspx, Handler.ashx (generic handler) och XMLFile1.xml.

En generic handler fungerar som en webservice som kan anropas för att utföra olika .net relaterade uppgifter, som att läsa från databaser eller xml.

Nedan listas den kod som ska läggas in i respektive fil. Läs kommentarerna för att skapa uppfattning om vad som händer.

XMLFile1.xml

```
<Guestbook>
</Guestbook>
```

Handler.ashx

```
<%@ WebHandler Language="C#" Class="Handler" %>

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
//Importera denna för att hantera xml.
using System.Xml;
//Importera denna för att hantera filer.
using System.IO;

public class Handler : IHttpHandler
{
    HttpContext context;
    public void ProcessRequest(HttpContext context)
    {
        //Spara referens till context(anroparen av webservice).
        this.context = context;

        //Kolla om en namnvariabel finns. (name=Klas)
        if (context.Request.Params["name"] != null)
        {
            //Hämta variabler.
            String name = context.Request.Params["name"];
            String message = context.Request.Params["message"];
            //Skapa variabel som håller aktuellt datum.
            String date = DateTime.Now.ToShortDateString();

            AddXML(name, message, date);
        }
    }

    void AddXML(String name, String message, String date)
    {
        XmlDocument myXmlDocument = new XmlDocument();
```

```

//Ladda den skapade XML-filen.
myXmlDocument.Load(context.Server.MapPath("~/XMLFile1.xml"));
XmlNode myXmlNode = myXmlDocument.DocumentElement.FirstChild;

//Skapa ett nytt xml element och generera dess 3 attribut.
XmlElement myXmlElement = myXmlDocument.CreateElement("entry");
myXmlElement.SetAttribute("name", name);
myXmlElement.SetAttribute("message", message);
myXmlElement.SetAttribute("date", date);

//Lägg till rad i XML-fil och spara den.
myXmlDocument.DocumentElement.InsertBefore(myXmlElement, myXmlNode);
myXmlDocument.Save(context.Server.MapPath("~/XMLFile1.xml"));

//Skicka svar till anropare (flash eller browser)
context.Response.Write("XML data tillagd.");
}

public bool IsReusable
{
    get
    {
        return false;
    }
}
}

```

Steg 2:

Testkör nu genom att starta debug eller högerklicka på Handler.ashx och välj view in browser. Det händer inget eftersom det inte skickat med några variabelpar, lägg till följande i adressfältet: /Handler.ashx?name=Klas&Message=Hej!

XMLFile1.xml ska nu innehålla följande data:

```

<?xml version="1.0" encoding="utf-8"?>
<Guestbook>
  <entry name="Klas" message="Hej!" date="3/9/2012" />
</Guestbook>

```

Den nya datan i filen består av attribut istället för noder. Dessa hanteras på ett annat sett men fyller samma funktion.

Steg 3:

Prova lägga till fler attribut, som till exempel epostadress eller efternamn i XML-filen.