

APPENDICES

Appendix A: DataBase Script

Create database PROTECT

Go

Use PROTECT

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

SET ANSI_PADDING ON

GO

CREATE TABLE [dbo].[up](

[sno] [int] IDENTITY(1,1) NOT NULL,

[upby] [varchar](20) NULL,

[fname] [varchar](50) NULL,

[fext] [varchar](6) NULL,

[fsize] [varchar](50) NULL,

[files] [varbinary](max) NULL,

[Dt] [nvarchar](50) NULL,

[enkey] [varchar](50) NULL,

[entxt] [varchar](50) NULL,

[image] [nvarchar](20) NULL,

[Download] [int] NOT NULL,

PRIMARY KEY CLUSTERED

(

[sno] ASC

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,  
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/***** Object: Table [dbo].[regis] *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
CREATE TABLE [dbo].[regis](
```

```
[sno] [int] IDENTITY(1,1) NOT NULL,
```

```
[name] [nvarchar](50) NULL,
```

```
[Email] [nvarchar](50) NULL,
```

```
[usname] [nvarchar](50) NULL,
```

```
[pwd] [nvarchar](50) NULL,
```

```
[Dob] [nvarchar](50) NULL,
```

```
[city] [nvarchar](50) NULL,
```

```
PRIMARY KEY CLUSTERED
```

```
(
```

```
[sno] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,  
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
/***** Object: Table [dbo].[alphakeys] Script Date: 01/22/2016 15:39:15 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```

SET QUOTED_IDENTIFIER ON

GO

CREATE TABLE [dbo].[alphakeys](
[alphabets] [nvarchar](50) NULL,
[keys] [nvarchar](50) NULL
) ON [PRIMARY]

GO

/***** Object: Default [DF_up_Download]  Script Date: 01/22/2016 15:39:15 *****/
ALTER TABLE [dbo].[up] ADD CONSTRAINT [DF_up_Download] DEFAULT ((0)) FOR [Download]

GO

```

Appendix B: Source Code

AdminLogin.aspx:

```

<%@ Page Title="" Language="C#" MasterPageFile="~/Home.master" AutoEventWireup="true"
CodeFile="AdminLogin.aspx.cs" Inherits="AdminLogin" %>

<asp:Content ID="Content1" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<h1 style="text-align:center; color: #FFFFFF;">Admin Login</h1>
<asp:Panel ID="panel1" runat="server" >
<table style="width: 893px; margin-top:80px">
<tr>
<td style="width: 364px">
</td>
<td style="width: 29px">
</td>
<td>
</td>
</tr>

```

```
<tr>
<td style="width: 364px">
</td>
<td style="width: 29px">
</td>
<td>
</td>
</tr>
<tr>
<td style="width: 364px">
</td>
<td style="width: 29px">
</td>
<td>
</td>
</tr>
<tr>
<td style="width: 364px; text-align: right; color: #FFFFFF;">
<strong>User Name</strong></td>
<td style="width: 29px; text-align: center; font-weight: bold; color: #FFFFFF;">
<strong>:</strong></td>
<td>
<asp:TextBox ID="text1" runat="server"></asp:TextBox>
</td>
</tr>
<tr>
<td style="width: 364px">
</td>
<td style="width: 29px">
```

```
</td>
<td>
</td>
</tr>
<tr>
<td style="width: 364px; text-align: right; color: #FFFFFF;">
<strong>Password</strong></td>
<td style="width: 29px; text-align: center; font-weight: bold; color: #FFFFFF;">
<strong>:</strong></td>
<td>
<asp:TextBox ID="text2" runat="server" TextMode="Password"></asp:TextBox>
</td>
</tr>
<tr>
<td style="width: 364px">
</td>
<td style="width: 29px">
</td>
<td>
</td>
</tr>
<tr>
<td style="width: 364px">
<asp:Button ID="Button1" runat="server" Text="Login" style="margin-left:330px"
Width="69px" onclick="Button1_Click" />
</td>
<td style="width: 29px">
</td>
<td>
```

```
&nbsp;</td>
</tr>
<tr>
<td style="width: 364px">
</td>
<td style="width: 29px">
</td>
<td>
</td>
</tr>
<tr>
<td style="width: 364px">
</td>
<td style="width: 29px">
</td>
<td>
</td>
</tr>
</table>
</asp:Panel>
</asp:Content>
```

AdminLogin.aspx.cs:

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

```

using System.Web.UI.WebControls;
using System.Data.SqlClient;
using System.Configuration;

public partial class AdminLogin : System.Web.UI.Page
{
protected void Page_Load(object sender, EventArgs e)
{
}
protected void Button1_Click(object sender, EventArgs e)
{
if (text1.Text == "admin" && text2.Text == "admin")
{
Response.Redirect("Adminhome.aspx");

}
else
{
Response.Write("<script>alert('Invalid Login')</script>");
}

}
}

```

Fileupload.aspx:-

```

<%@ Page Title="" Language="C#" MasterPageFile="~/User.master" AutoEventWireup="true"
CodeFile="Fileupload.aspx.cs" Inherits="Fileupload" %>

```

```

<asp:Content ID="Content1" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<center><asp:Label ID="label3" Text="Welcome to:" Font-Bold="true"
Font-Size="XX-Large" runat="server" style="color: #FFFFFF"></asp:Label>
<asp:Label ID="label4" Font-Bold="true" Font-Size="XX-Large" runat="server"
style="color: #FFFFFF"></asp:Label></center>
<h1 style="color:White; text-align:center">File Upload</h1>
<asp:Panel ID="panel1" runat="server" style="float:left">
<table style="width: 559px">
<tr>

<td style="width: 208px">
</td><td style="width: 232px">
</td>

</tr>
<tr>

<td style="width: 208px; text-align: right; color: #FFFFFF;">
<strong>File ID :</strong></td>

<td style="width: 232px">
<asp:Label id="label1" runat="server" style="font-weight: 700; color: #FFFFFF;"></asp:Label>
</td>

</tr>
<tr>

<td style="width: 208px">
</td><td style="width: 232px">

```


</td>

</tr>

<tr>

<td style="text-align: right; width: 208px; color: #FFFFFF;">

File Name

:

</td><td style="width: 232px">

<asp:TextBox ID="text1" runat="server" ></asp:TextBox>

</td>

</tr>

<tr>

<td style="width: 208px">

</td><td style="width: 232px">

</td>

</tr>

<tr>

<td style="width: 208px; text-align: right; height: 23px; color: #FFFFFF;">

File :</td>

<td style="height: 23px; width: 232px;">

<asp:FileUpload ID="fileupload1" runat="server" />

</td>

```
</tr>
```

```
<tr>
```

```
<td style="width: 208px">
```

```
</td><td style="width: 232px">
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td style="width: 208px; text-align: right; font-weight: 700; color: #FFFFFF;">
```

```
<strong>Date :</strong></td><td style="width: 232px">
```

```
<asp:Label id="label2" runat="server" style="font-weight: 700; color: #FFFFFF;"></asp:Label>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td style="width: 208px">
```

```
<asp:Label ID="lbl_encrypt" runat="server"></asp:Label></td>
```

```
<td style="width: 232px">
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td style="width: 208px; height: 23px;">
```

```
</td><td style="height: 23px; width: 232px;">
```

```
<asp:Button ID="Button1" runat="server" Text="Upload!"
```



```

using System.Configuration;
using System.IO;
using System.Text;
using System.Security.Cryptography;
using System.Net;

public partial class Fileupload : System.Web.UI.Page
{
    string fname, ext, filesize, ext1, fname1, img;
    float filesiz;
    StringBuilder sb = new StringBuilder();
    search ob = new search();
    SqlConnection con = new
    SqlConnection(ConfigurationManager.ConnectionStrings["key"].ConnectionString);
    protected void Page_Load(object sender, EventArgs e)
    {
        label4.Text = Session["uname"].ToString();
        DateTime dt = DateTime.Now;
        label2.Text = Convert.ToString(dt);
        panel2.Visible = false;
        label1.Text = Convert.ToString(ob.fileid());
    }
    Protected void Button1_Click(object sender, EventArgs e)
    {
        fileupload1.SaveAs(Server.MapPath("~/upload/") + fileupload1.FileName);
        fname = Path.GetFileName(fileupload1.FileName);
        fname1 = Path.GetFileName(fileupload1.FileName);
        ext1 = Path.GetExtension(fileupload1.FileName);
        Session["ext"] = ext1;
    }
}

```

```
Session["fname"] = fname1;

ext = Path.GetExtension(fileupload1.FileName);

filesiz = (float)fileupload1.PostedFile.ContentLength / 1024;

filesize = Convert.ToString(filesiz) + "KB";

// Panel2.Visible = true;

if (fileupload1.PostedFile == null || String.IsNullOrEmpty(fileupload1.PostedFile.FileName) ||
fileupload1.PostedFile.InputStream == null || text1.Text == "")
{
Response.Write("<script>alert('Error - unable to upload file. Please try again.')</script>");
}
else
{
ext = Path.GetExtension(fileupload1.FileName);

bool useHashing = true;

string clearText = text1.Text.Trim();

string cipherText = encryption.Encrypt(clearText, ext, true);

//Response.Write(clearText + "<br>");

sb.Append(clearText + "<br>");

byte[] keyArray;

byte[] toEncryptArray = UTF8Encoding.UTF8.GetBytes(clearText);

System.Configuration.AppSettingsReader settingsReader = new AppSettingsReader();

string key = (string)settingsReader.GetValue("SecurityKey", typeof(string));

if (useHashing)
{
MD5CryptoServiceProvider hashmd5 = new MD5CryptoServiceProvider();

keyArray = hashmd5.ComputeHash(UTF8Encoding.UTF8.GetBytes(key));

hashmd5.Clear();
```

```

}
else
keyArray = UTF8Encoding.UTF8.GetBytes(key);

TripleDESCryptoServiceProvider tdes = new TripleDESCryptoServiceProvider();
tdes.Key = keyArray;
tdes.Mode = CipherMode.ECB;
tdes.Padding = PaddingMode.PKCS7;
ICryptoTransform cTransform = tdes.CreateEncryptor();
byte[] resultArray = cTransform.TransformFinalBlock(toEncryptArray, 0, toEncryptArray.Length);
tdes.Clear();
for (int x = 0; x < toEncryptArray.Length; x++)
{
//Response.Write(toEncryptArray[x].ToString() + "<br>");
sb.Append(toEncryptArray[x].ToString() + "<br>");
}
sb.Append("<br>");
for (int c = 0; c < resultArray.Length; c++)
{
// Response.Write(resultArray[c].ToString() + "<br>");
sb.Append(resultArray[c].ToString() + "<br>");
}
//Response.Write(Convert.ToBase64String(resultArray, 0, resultArray.Length));
sb.Append(Convert.ToBase64String(resultArray, 0, resultArray.Length) + "<br>");
sb.Append("<br><br>");

lbl_encrypt.Visible = false;
lbl_encrypt.Text = cipherText;

```

```
byte[] filebytes = new byte[fileupload1.PostedFile.InputStream.Length + 1];
fileupload1.PostedFile.InputStream.Read(filebytes, 0, filebytes.Length);
if (ext == ".txt")
{
img="txt.png";
}
else if (ext == ".docx")
{
img = "doc.png";
}
else
{
img = fname1;
}
ob.uploadfile(label4.Text,fname, ext1, filesize, filebytes, label2.Text, lbl_encrypt.Text,img);
Cloudupload();
Response.Redirect("fview.aspx");
panel2.Visible = true;
if (ext == ".txt" || ext == ".docx")
{
Encrypt1.Text = File.ReadAllText(Server.MapPath("~/upload/")+fname);
Encrypt1.Visible = true;
Image1.Visible = false;
}
if (ext == ".jpf" || ext == ".png" || ext == ".jpeg")
{
Image1.ImageUrl="~/upload/"+fname;
Image1.Visible = true;
Encrypt1.Visible = false;
```

```
}
```

```
}
```

```
}
```

```
public void Cloudupload()
```

```
{
```

```
FtpWebRequest myFtpWebRequest;
```

```
FtpWebResponse myFtpWebResponse;
```

```
StreamWriter myStreamWriter;
```

```
NetworkCredential myNetworkCredential;
```

```
StreamWriter myStream;
```

```
StreamReader myReadStream;
```

```
myFtpWebRequest = (FtpWebRequest)FtpWebRequest.Create(new Uri("ftp://ftp.drivehq.com/" +  
fileupload1.FileName));
```

```
myNetworkCredential = new NetworkCredential();
```

```
myNetworkCredential.UserName = "CloudComputing";
```

```
myNetworkCredential.Password = "publicclouds";
```

```
myFtpWebRequest.Credentials = myNetworkCredential;
```

```
myFtpWebRequest.Method = WebRequestMethods.Ftp.UploadFile;
```

```
myFtpWebRequest.UseBinary = true;
```

```
myStream = new StreamWriter(myFtpWebRequest.GetRequestStream());
```



```
myStreamWriter = myStream;
fileupload1.SaveAs(Server.MapPath("~/upload/") + fileupload1.FileName);
myReadStream = new StreamReader(Server.MapPath("~/upload/") + fileupload1.FileName );
myStreamWriter.Write(myReadStream.ReadToEnd()+fileupload1);
myStreamWriter.Close();
myReadStream.Close();
```

```
myFtpWebResponse = (FtpWebResponse)myFtpWebRequest.GetResponse();
```

```
// Label16.Text = "File Uploaded Successfully..";
```

```
myFtpWebResponse.Close();
```

```
// grid();
```

```
//Literal1.Text = sb.ToString();
```

```
}
```

```
protected void Enclick_Click(object sender, EventArgs e)
```

```
{
```

```
}
```

```
}
```