**Лекции**

**ЛК 14**. C#. Создание ASP.NET приложения.

*Создание простого веб-приложения*

1. Создаем в студии пустое ASP.NET приложение и добавляем в него следующие файлы

WebForm.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm.aspx.cs" Inherits="TestAspApplication.WebForm" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

</head>

<body>

<form id="form1" runat="server">

<div style="text-align:left; float: none; display: block; margin-left:auto ; margin-right: auto; width: 40%">

<asp:CheckBoxList ID="CheckBoxList" runat="server" Width="40%" />

</div>

<div style ="text-align: center">

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

</div>

<br/>

<div style="float: none; display: block; margin-left: auto; margin-right: auto; text-align: center; width: 40%">

<asp:Button ID="SaveBtn" runat="server" Text="Save" OnClick="SaveBtn\_Click" Width="20%" />

&nbsp;&nbsp;&nbsp;&nbsp;

<asp:Button ID="CancelBtn" runat="server" Text="Cancel" Width="20%" OnClick="CancelBtn\_Click" />

</div>

</form>

</body>

</html>

**WebForm.aspx.cs**

private readonly TestService \_service;

public WebForm()

{

\_service = new TestService();

}

protected void Page\_Load(object sender, EventArgs e)

{

if (CheckBoxList.Items.Count == 0)

{

var records = \_service.GetAllRecords();

var items = records.Select(t => new ListItem(t.Value, t.Key.ToString(), true)).ToList();

CheckBoxList.Items.AddRange(items.ToArray());

}

}

protected void SaveBtn\_Click(object sender, EventArgs e)

{

if (FileUpload.HasFile)

{

var ids = (from ListItem item

in CheckBoxList.Items

where item.Selected

select new Guid(item.Value)).ToArray();

\_service.Attach(ids, FileUpload.FileName, FileUpload.FileBytes);

}

}

protected void CancelBtn\_Click(object sender, EventArgs e)

{

// Пример того, как делать не нужно. Нажатие кнопки будет вызывать отравку данных на сервер.

// В данном случае достаточно JS кода

foreach (ListItem item in CheckBoxList.Items)

{

if (item.Selected)

{

item.Selected = false;

}

}

}

**TestService.cs**

private readonly EntityRepository \_rep;

public TestService()

{

var entityName = ConfigurationManager.AppSettings["entityName"];

// create new crm connection

ServicePointManager.ServerCertificateValidationCallback =

delegate(object s, X509Certificate certificate, X509Chain chain, SslPolicyErrors sslPolicyErrors)

{

return true;

};

var connection = new CrmConnection("Xrm")

{

ServiceConfigurationInstanceMode = ServiceConfigurationInstanceMode.PerName

};

var crmService = new OrganizationService(connection);

\_rep = new EntityRepository(crmService, entityName);

}

public Dictionary<Guid, string> GetAllRecords()

{

string nameField = ConfigurationManager.AppSettings["nameField"];

return \_rep.GetAllTests(nameField);

}

public void Attach(IEnumerable<Guid> recordIds, string fileName, byte[] fileBytes)

{

foreach (var id in recordIds)

{

\_rep.AttachFile(id, fileName, fileBytes);

}

}

**EntityRepository.cs**

private readonly IOrganizationService \_service;

private readonly string \_entityName;

public EntityRepository(IOrganizationService service, string entityName)

{

\_service = service;

\_entityName = entityName;

}

public Dictionary<Guid, string> GetAllTests(string fieldName)

{

// так выбирать все записи нельзя! для реальных задач берем записи постранично

var query = new QueryExpression(\_entityName)

{

ColumnSet = new ColumnSet(fieldName),

};

var res = \_service.RetrieveMultiple(query).Entities;

return res.ToDictionary(t => t.Id, t => t.GetAttributeValue<string>(fieldName));

}

public void AttachFile(Guid id, string fileName, byte[] fileBytes)

{

...

}

Web.config

<?xml version="1.0" encoding="utf-8"?>

<configuration>

<appSettings>

<add key="entityName" value="new\_test"/>

<add key="nameField" value="new\_name"/>

</appSettings>

<system.web>

<compilation debug="true" targetFramework="4.5.2" />

<httpRuntime targetFramework="4.5.2" />

</system.web>

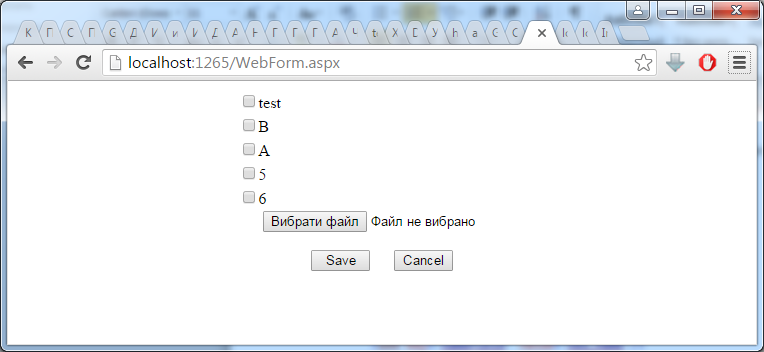
<connectionStrings>

<add name="Xrm" connectionString="Server=http://crm.uds.systems/Test; Username=UDS\s.ivantishen; Password=blabla"/>

</connectionStrings>

</configuration>

1. Публикуем веб-приложение и запускаем в браузере. Если всё сделано правильно, должна появиться страница следующего вида



1. Проверяем работу веб-приложения. Находим нужные записи и убеждаемся в том, что вложения созданы

