

# Developing a Survey Application (Part 2)

By Bipin Joshi

LANGUAGES: VB.NET  
ASP.NET VERSIONS: 2.0

In Part 1 of this series we designed the database to store survey information. We developed a master page for all the administrative web forms. We also developed a web form for managing survey definitions. Any survey consists of a set of questions and possible choices. The administrator needs to define questions and their choices for a particular survey. Additionally, participants of a survey need to be specified so that invitations can be sent to them. This is precisely the theme of this part.

## Managing Survey Questions and Choices

Now we will proceed to develop two more web forms – ManageQuestions.aspx and ManageChoices.aspx. The former allows you to manage questions belonging to a survey and the later allows you to manage choices of objective type questions. To start, add these two web forms to the Admin folder of the web site.

The general layout of the ManageQuestions.aspx is shown in Figure 15.

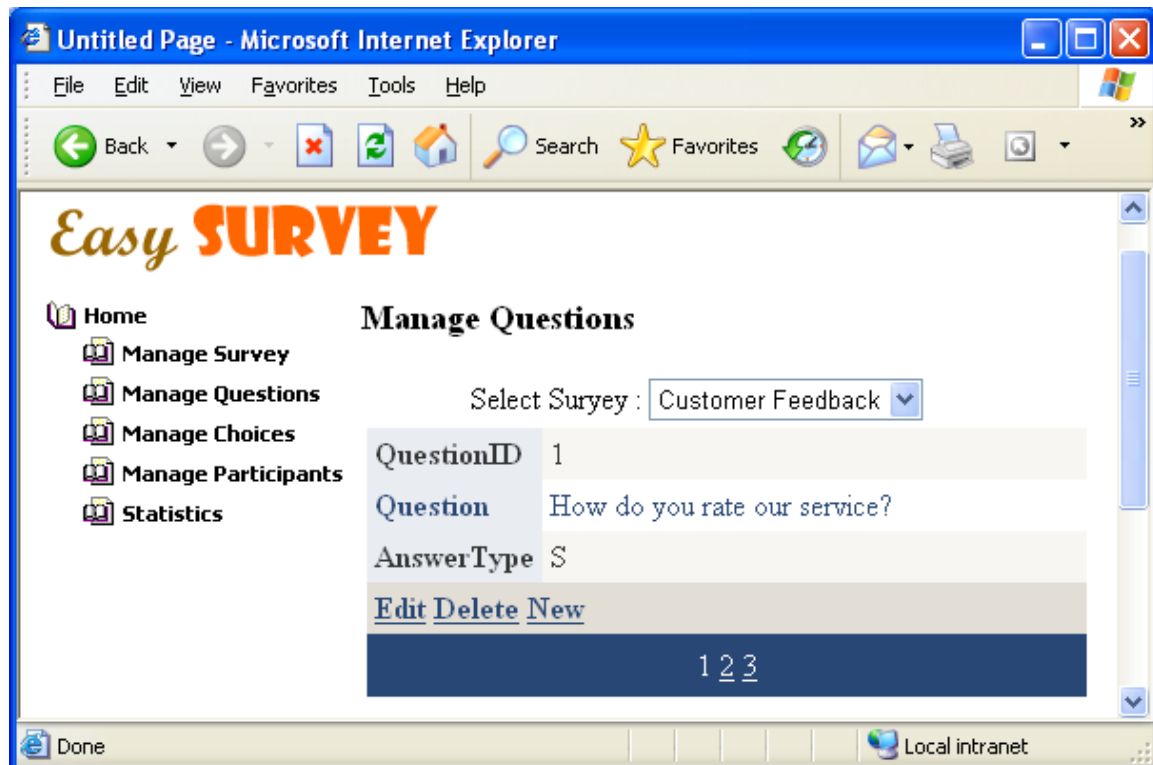


Figure 15: Layout of ManageQuestions.aspx

There is a DropDownList at the top of the web form that lists all the available surveys. Once a survey is selected the questions belonging to that survey are displayed in a DetailsView control. The administrator can then add, edit or delete the questions.

To design this web form, drag and drop two SQL data source control on the web form. Configure the first (SqlDataSource1) SQL data source control to select SurveyID and Title from Survey table. Similarly, configure the other SQL data source control (SqlDataSource2) to select all the rows from SurveyQuestions table matching the selected SurveyID. In order to add this WHERE condition you need to use the “Add WHERE Clause” dialog of the SQL data source configuration wizard. Figure 16 shows this dialog with the WHERE condition specified.

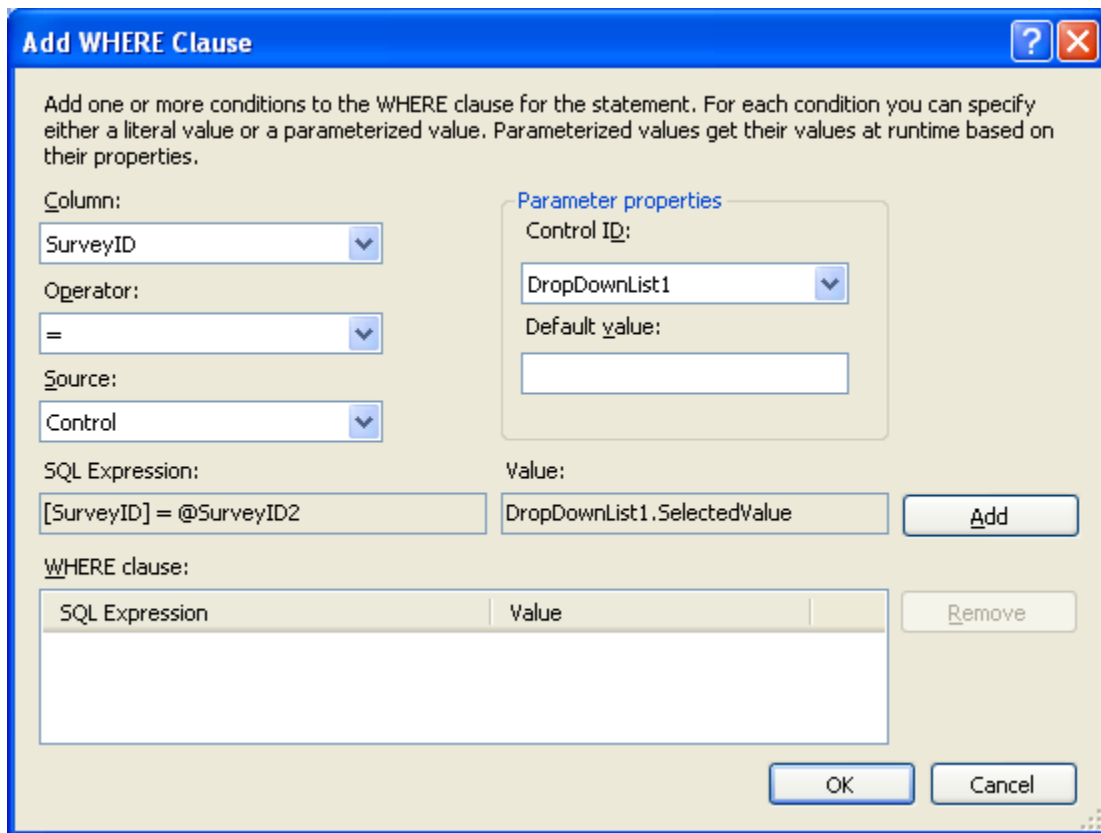


Figure 16: Add WHERE Clause dialog

Also, ensure that “Generate INSERT, UPDATE and DELETE statements” checkbox is checked in the “Advanced SQL Generation Options” dialog (refer Figure 12).

Now drag and drop a DropDownList control and set its DataSourceID property to SqlDataSource1. Set its DataTextField and DatavalueField properties to Title and SurveyID respectively. Also, set its AutoPostBack property to True.

Next, drag and drop a DetailsView control on the web form and set its DataSourceID property to SqlDataSource2. From the smart tag of the DetailsView enable paging, inserting, editing and deleting. The Question and AnswerType fields of the DetailsView control are template fields. The Question template contains a multi line Textbox whereas AnswerType field contains a DropDownList with required options. Convert these two fields into TemplateFields using the "Fields" dialog. Creating the Question template field is fairly simple. Simply set the TextMode property of the Textbox from EditItemTemplate and InsertItemTemplate to MultiLine. To create the AnswerType template field, drag and drop a DropDownList controls in EditItemTemplate and InsertItemTemplate and add three items to both the DropDownLists as shown in Figure 17.

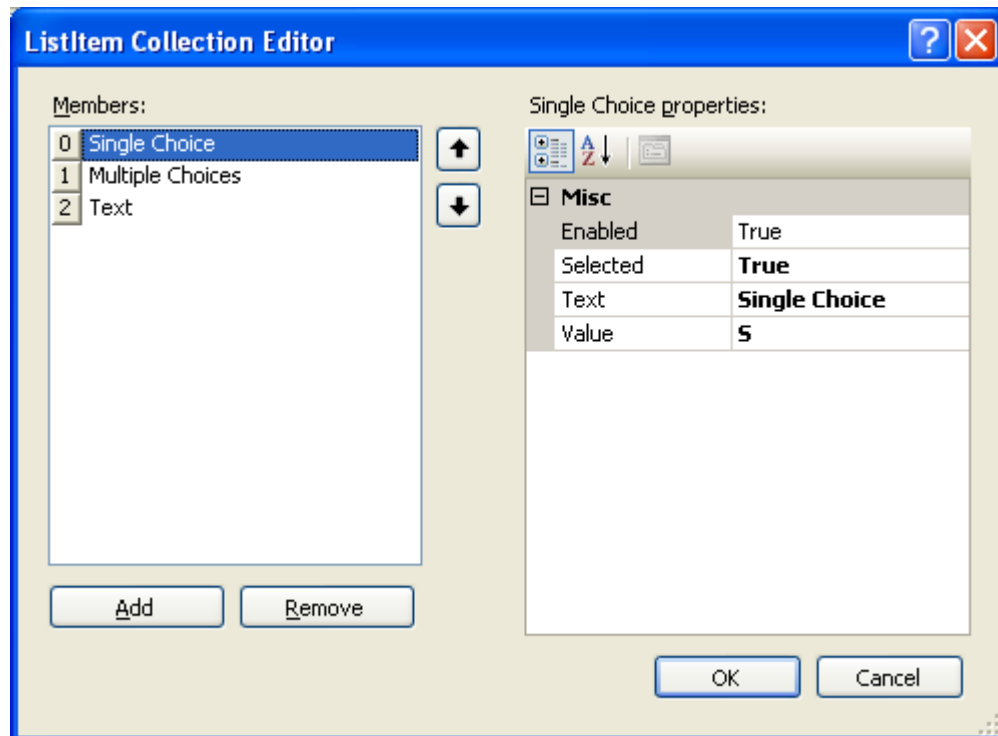


Figure 17: Adding items to the DropDownList

Note that the Value property of Single Choice ListItem is set to S, that of Multiple Choices is set to M and that of Text is set to T. Figure 18 shows these DropDownLists after adding the items.

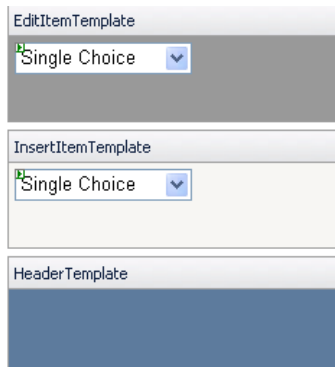


Figure 18: EditItemTemplate and InsertItemTemplate of DetailsView

Now, select “Edit DataBindings...” option from the smart tag of both the DropDownLists and bind SelectedValue property to AnswerType column (Figure 19).

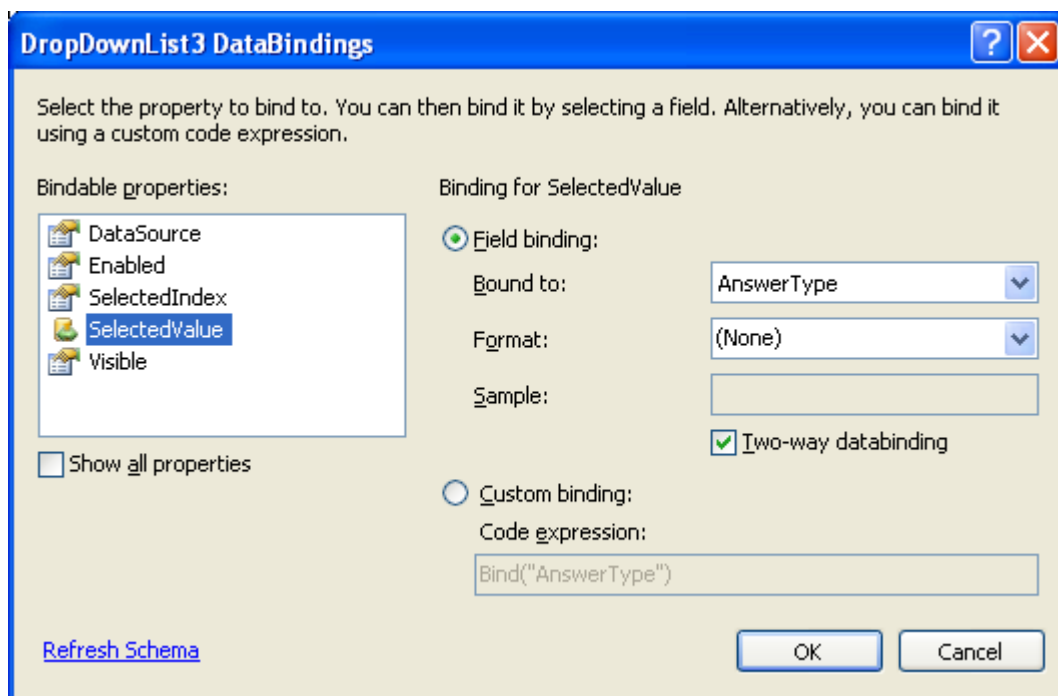


Figure 19: Data binding SelectedValue property of DropDownList

The DetailsView does not include SurveyID column. However, we need to set this column during INSERT or UPDATE operations because each question must belong to some survey. This is done by handling ItemInserting and ItemUpdating events. Figure 20 shows these event handlers.

```
Protected Sub DetailsView1_ItemInserting(ByVal sender As
Object, ByVal e As
System.Web.UI.WebControls.DetailsViewInsertEventArgs)
Handles DetailsView1.ItemInserting
e.Values("SurveyID") = DropDownList1.SelectedValue
```

```
End Sub
```

```
Protected Sub DetailsView1_ItemUpdating(ByVal sender As  
Object, ByVal e As  
System.Web.UI.WebControls.DetailsViewUpdateEventArgs)  
Handles DetailsView1.ItemUpdating  
e.NewValues("SurveyID") = DropDownList1.SelectedValue  
End Sub
```

Figure 20: Handling ItemInserting and ItemUpdating event handlers in ManageQuestions.aspx

The event handlers set the SurveyID parameter from Values and NewValues collections with the one selected in the DropDownList.

Figure 21 shows the complete markup of ManageQuestions.aspx.

```
<%@ Page Language="VB"  
MasterPageFile="~/Admin/AdminMasterPage.master"  
AutoEventWireup="false" CodeFile="ManageQuestions.aspx.vb"  
Inherits="Admin_ManageQuestions" title="Untitled Page" %>  
<asp:Content ID="Content1"  
ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">  
<asp:Label ID="Label2" runat="server" Text="<h3>Manage  
Questions</h3>"></asp:Label>  
<table style="width: 100%">  
<tr>  
<td align="right">  
<asp:Label ID="Label1" runat="server" Text="Select Surveye  
:"></asp:Label></td>  
<td>  
<asp:DropDownList ID="DropDownList1" runat="server"  
AutoPostBack="True" DataSourceID="SqlDataSource1"  
DataTextField="Title" DataValueField="SurveyID">  
</asp:DropDownList></td>  
</tr>  
<tr>  
<td colspan="2">  
<asp:DetailsView ID="DetailsView1" runat="server"  
AllowPaging="True" AutoGenerateRows="False"  
CellPadding="4" DataKeyNames="QuestionID"  
DataSourceID="SqlDataSource2" ForeColor="#333333"  
GridLines="None" Height="50px" Width="100%">  
<FooterStyle BackColor="#5D7B9D" Font-Bold="True"  
ForeColor="White" />  
<CommandRowStyle BackColor="#E2DED6" Font-Bold="True" />  
<EditRowStyle BackColor="#999999" />
```

```

<RowStyle BackColor="#F7F6F3" ForeColor="#333333" />
<PagerStyle BackColor="#284775" ForeColor="White"
HorizontalAlign="Center" />
<Fields>
<asp:BoundField DataField="QuestionID"
HeaderText="QuestionID" InsertVisible="False"
ReadOnly="True" SortExpression="QuestionID" />
<asp:TemplateField HeaderText="Question"
SortExpression="Question">
<EditItemTemplate>
<asp:TextBox ID="TextBox1" runat="server" Text='<%#
Bind("Question") %>' TextMode="MultiLine"></asp:TextBox>
</EditItemTemplate>
<InsertItemTemplate>
<asp:TextBox ID="TextBox1" runat="server" Text='<%#
Bind("Question") %>' TextMode="MultiLine"></asp:TextBox>
</InsertItemTemplate>
<ItemTemplate>
<asp:Label ID="Label2" runat="server" Text='<%#
Bind("Question") %>'></asp:Label>
</ItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="AnswerType"
SortExpression="AnswerType">
<EditItemTemplate>
<asp:DropDownList ID="DropDownList3" runat="server"
SelectedValue='<%# Bind("AnswerType") %>'>
<asp:ListItem Selected="True" Value="S">Single
Choice</asp:ListItem>
<asp:ListItem Value="M">Multiple Choices</asp:ListItem>
<asp:ListItem Value="T">Text</asp:ListItem>
</asp:DropDownList>
</EditItemTemplate>
<InsertItemTemplate>
<asp:DropDownList ID="DropDownList2" runat="server"
SelectedValue='<%# Bind("AnswerType") %>'>
<asp:ListItem Value="S">Single Choice</asp:ListItem>
<asp:ListItem Value="M">Multiple Choices</asp:ListItem>
<asp:ListItem Value="T">Text</asp:ListItem>
</asp:DropDownList>
</InsertItemTemplate>
<ItemTemplate>
<asp:Label ID="Label1" runat="server" Text='<%#
Bind("AnswerType") %>'></asp:Label>
</ItemTemplate>
</asp:TemplateField>

```

```

<asp:CommandField ShowDeleteButton="True"
ShowEditButton="True" ShowInsertButton="True" />
</Fields>
<FieldHeaderStyle BackColor="#E9ECF1" Font-Bold="True"
Width="10%" />
<HeaderStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
<AlternatingRowStyle BackColor="White" ForeColor="#284775"
/>
</asp:DetailsView>
</td>
</tr>
</table>
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ ConnectionStrings:ConnectionString
%>"
SelectCommand="SELECT [SurveyID], [Title] FROM [Survey]
ORDER BY [Title]"></asp:SqlDataSource>
<asp:SqlDataSource ID="SqlDataSource2" runat="server"
ConnectionString="<%= $ ConnectionStrings:ConnectionString
%>"
DeleteCommand="DELETE FROM [SurveyQuestions] WHERE
[QuestionID] = @QuestionID"
InsertCommand="INSERT INTO [SurveyQuestions] ([SurveyID],
[Question], [AnswerType]) VALUES (@SurveyID, @Question,
@AnswerType) "
SelectCommand="SELECT * FROM [SurveyQuestions] WHERE
([SurveyID] = @SurveyID) "
UpdateCommand="UPDATE [SurveyQuestions] SET [SurveyID] =
@SurveyID, [Question] = @Question, [AnswerType] =
@AnswerType WHERE [QuestionID] = @QuestionID">
<DeleteParameters>
<asp:Parameter Name="QuestionID" Type="Int32" />
</DeleteParameters>
<UpdateParameters>
<asp:Parameter Name="SurveyID" Type="Int32" />
<asp:Parameter Name="Question" Type="String" />
<asp:Parameter Name="AnswerType" Type="String" />
<asp:Parameter Name="QuestionID" Type="Int32" />
</UpdateParameters>
<SelectParameters>
<asp:ControlParameter ControlID="DropDownList1"
Name="SurveyID" PropertyName="SelectedValue"
Type="Int32" />
</SelectParameters>
<InsertParameters>
<asp:Parameter Name="SurveyID" Type="Int32" />

```

```

<asp:Parameter Name="Question" Type="String" />
<asp:Parameter Name="AnswerType" Type="String" />
</InsertParameters>
</asp:SqlDataSource>
</asp:Content>

```

Figure 21: Markup of ManageQuestions.aspx

Now, let's develop the ManageChoices.aspx. Figure 22 shows the general layout of this web form.

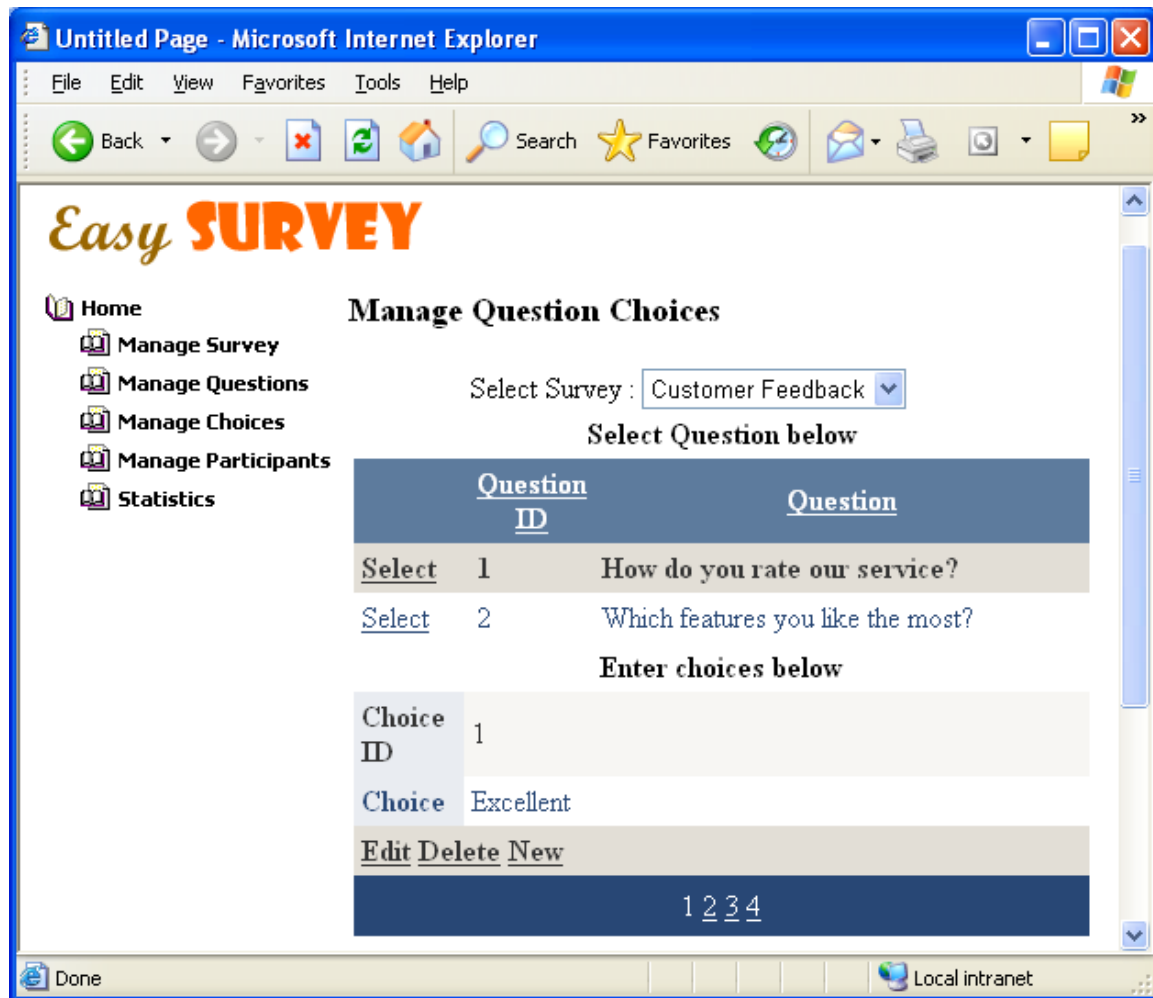


Figure 22: Layout of ManageChoices.aspx

The web form consists of a DropDownList presenting a list of surveys. Below the DropDownList there is a GridView that displays all the objective type questions from the selected survey. Selecting a question displays its choices in a DetailsView wherein you can add, edit or delete them.



To design this web form, add three SQL data source controls on the web form. Configure the first one to select all the records from Survey table. Configure the second one to fetch all the questions matching the selected SurveyID. Finally, configure the third one to select all the choices from the SurveyChoices table for the selected QuestionID from the GridView. Add the WHERE conditions for the second and the third SQL data source control as discussed before. Also, ensure that “Generate INSERT, UPDATE and DELETE statements” checkbox is checked in the “Advanced SQL Options” dialog.

Now, drag and drop a DropDownList and bind it with SqlDataSource1. Similarly, drag and drop a GridView and a DetailsView and bind them with SqlDataSource2 and SqlDataSource3 respectively.

When any of the question is selected in the Grid that is not having any choices added yet we display the DetailsView in insert mode otherwise we display it in read only mode. This is achieved by handling the DataBound event of DetailsView as shown in Figure 23.

```
Protected Sub DetailsView1_DataBound(ByVal sender As
Object, ByVal e As System.EventArgs) Handles
DetailsView1.DataBound
If DetailsView1.Rows.Count = 0 Then
    DetailsView1.ChangeMode(DetailsViewMode.Insert)
End If
End Sub
```

Figure 23: Handling DataBound event of DetailsView

The event handler checks the row count of the DetailsView control. If it is zero then the mode of the DetailsView is changed to Insert using ChangeMode() method.

The DetailsView doesn't include QuestionID column nevertheless it is required while inserting or updating a record. The ItemInserting and ItemDeleting event handlers do the trick as before. Figure 24 shows these event handlers.

```
Protected Sub DetailsView1_ItemInserting(ByVal sender As
Object, ByVal e As
System.Web.UI.WebControls.DetailsViewInsertEventArgs)
Handles DetailsView1.ItemInserting
e.Values("QuestionID") = GridView1.SelectedValue
End Sub
```

```
Protected Sub DetailsView1_ItemUpdating(ByVal sender As
Object, ByVal e As
System.Web.UI.WebControls.DetailsViewUpdateEventArgs)
Handles DetailsView1.ItemUpdating
```

```
e.NewValues("QuestionID") = GridView1.SelectedValue
End Sub
```

Figure 24: Handling ItemInserting and ItemUpdating event handlers in ManageChoices.aspx

The event handlers set the QuestionID parameter from Values and NewValues collections with the one selected in the GridView. Figure 25 shows the complete markup of ManageChoices.aspx.

```
<%@ Page Language="VB"
MasterPageFile="~/Admin/AdminMasterPage.master"
AutoEventWireup="false" CodeFile="ManageChoices.aspx.vb"
Inherits="Admin_ManageChoices" title="Untitled Page" %>
<asp:Content ID="Content1"
ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<asp:Label ID="Label2" runat="server" Text="<h3>Manage
Question Choices</h3>"></asp:Label>
<table width="100%">
<tr>
<td align="right">
<asp:Label ID="Label1" runat="server"
Text="Select Survey :"></asp:Label>
</td>
<td>
<asp:DropDownList ID="DropDownList1" runat="server"
AutoPostBack="True" DataSourceID="SqlDataSource1"
DataTextField="Title" DataValueField="SurveyID">
</asp:DropDownList></td>
</tr>
<tr>
<td align="center" colspan="2">
<asp:Label ID="Label3" runat="server" Font-Bold="True"
Text="Select Question below"></asp:Label></td>
</tr>
<tr>
<td colspan="2">
<asp:GridView ID="GridView1" runat="server"
AllowPaging="True" AllowSorting="True"
AutoGenerateColumns="False" CellPadding="4"
DataKeyNames="QuestionID" DataSourceID="SqlDataSource2"
ForeColor="#333333" GridLines="None" PageSize="5"
Width="100%">
<FooterStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
<Columns>
<asp:CommandField ShowSelectButton="True" />
```

```

<asp:BoundField DataField="QuestionID" HeaderText="Question
ID" InsertVisible="False"
ReadOnly="True" SortExpression="QuestionID">
<ItemStyle Width="10%" />
</asp:BoundField>
<asp:BoundField DataField="Question" HeaderText="Question"
SortExpression="Question" />
</Columns>
<RowStyle BackColor="#F7F6F3" ForeColor="#333333" />
<EditRowStyle BackColor="#999999" />
<SelectedRowStyle BackColor="#E2DED6" Font-Bold="True"
ForeColor="#333333" />
<PagerStyle BackColor="#284775" ForeColor="White"
HorizontalAlign="Center" />
<HeaderStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
<AlternatingRowStyle BackColor="White" ForeColor="#284775"
/>
</asp:GridView>
</td>
</tr>
<tr>
<td align="center" colspan="2">
<asp:Label ID="Label4" runat="server" Font-Bold="True"
Text="Enter choices below"></asp:Label></td>
</tr>
<tr>
<td colspan="2">
<asp:DetailsView ID="DetailsView1" runat="server"
AllowPaging="True" AutoGenerateRows="False"
CellPadding="4" DataKeyNames="ChoiceID"
DataSourceID="SqlDataSource3" ForeColor="#333333"
GridLines="None" Height="50px" Width="100%">
<FooterStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
<CommandRowStyle BackColor="#E2DED6" Font-Bold="True" />
<EditRowStyle BackColor="#999999" />
<RowStyle BackColor="#F7F6F3" ForeColor="#333333" />
<PagerStyle BackColor="#284775" ForeColor="White"
HorizontalAlign="Center" />
<Fields>
<asp:BoundField DataField="ChoiceID" HeaderText="Choice ID"
InsertVisible="False"
ReadOnly="True" SortExpression="ChoiceID" />
<asp:BoundField DataField="Choice" HeaderText="Choice"
SortExpression="Choice" />

```

```

<asp:CommandField ShowDeleteButton="True"
ShowEditButton="True" ShowInsertButton="True" />
</Fields>
<FieldHeaderStyle BackColor="#E9ECF1" Font-Bold="True"
Width="15%" />
<HeaderStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
<AlternatingRowStyle BackColor="White" ForeColor="#284775"
/>
</asp:DetailsView>
</td>
</tr>
</table>
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ ConnectionStrings:ConnectionString
%>"
SelectCommand="SELECT [SurveyID], [Title] FROM
[Survey]"></asp:SqlDataSource>
<asp:SqlDataSource ID="SqlDataSource2" runat="server"
ConnectionString="<%= $ ConnectionStrings:ConnectionString
%>"
SelectCommand="SELECT * FROM [SurveyQuestions] WHERE
([SurveyID] = @SurveyID) AND ([AnswerType] <>
@AnswerType))">
<SelectParameters>
<asp:ControlParameter ControlID="DropDownList1"
Name="SurveyID"PropertyName="SelectedValue"
Type="Int32" />
<asp:Parameter DefaultValue="T" Name="AnswerType"
Type="String" />
</SelectParameters>
</asp:SqlDataSource>
<asp:SqlDataSource ID="SqlDataSource3" runat="server"
ConnectionString="<%= $ ConnectionStrings:ConnectionString
%>"
DeleteCommand="DELETE FROM [SurveyChoices] WHERE [ChoiceID]
= @ChoiceID" InsertCommand="INSERT INTO [SurveyChoices]
([QuestionID], [Choice]) VALUES (@QuestionID, @Choice)"
SelectCommand="SELECT * FROM [SurveyChoices] WHERE
([QuestionID] = @QuestionID)"
UpdateCommand="UPDATE [SurveyChoices] SET [QuestionID] =
@QuestionID, [Choice] = @Choice WHERE [ChoiceID] =
@ChoiceID">
<DeleteParameters>
<asp:Parameter Name="ChoiceID" Type="Int32" />
</DeleteParameters>
<UpdateParameters>

```

```

<asp:Parameter Name="QuestionID" Type="Int32" />
<asp:Parameter Name="Choice" Type="String" />
<asp:Parameter Name="ChoiceID" Type="Int32" />
</UpdateParameters>
<SelectParameters>
<asp:ControlParameter ControlID="GridView1"
Name="QuestionID" PropertyName="SelectedValue"
Type="Int32" />
</SelectParameters>
<InsertParameters>
<asp:Parameter Name="QuestionID" Type="Int32" />
<asp:Parameter Name="Choice" Type="String" />
</InsertParameters>
</asp:SqlDataSource>
</asp:Content>

```

Figure 25: Markup of ManageChoices.aspx

## Managing Survey Participants

Managing survey participants is fairly easy task. Figure 26 shows how this web form looks. The web form consists of a DropDownList that presents list of all surveys. Upon selecting a survey all the participants of that survey are displayed in a DetailsView. Using the DetailsView you can add, edit or delete participants.

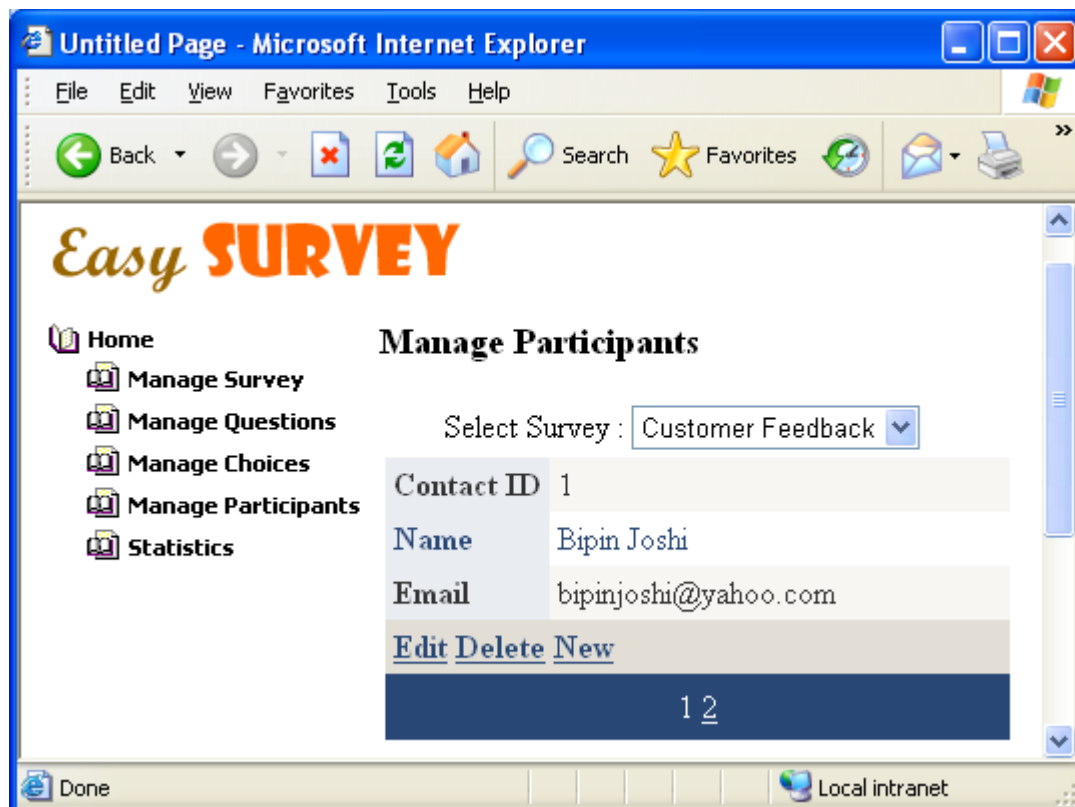


Figure 26: layout of ManageParticipants.aspx

In order to design the web form, drag and drop two SQL data source controls on it. Configure the first one to select all the records from the Survey table and configure the other one to select all the participants from SurveyParticipants table matching the selected survey.

Then drag and drop a DropDownList control on the web form and set its properties as in the previous case. Drag and drop a DetailsView control on the web form and set its DataSourceID property to SqlDataSource2. When you add a new survey it will not have any participants. In that case the DetailsView should be shown in insert mode otherwise it should be shown in read only mode. This is achieved by handling the DataBound event of DetailsView (Figure 27).

```
Protected Sub DetailsView1_DataBound(ByVal sender As
Object, ByVal e As System.EventArgs) Handles
DetailsView1.DataBound
If DetailsView1.Rows.Count = 0 Then
    DetailsView1.ChangeMode(DetailsViewMode.Insert)
End If
End Sub
```

Figure 27: Handling DataBound event of DetailsView

The code should be familiar as we used similar code in ManagedChoices.aspx. The SurveyID column of SurveyParticipants table is not displayed in the DetailsView but we need to supply it during inserts and updates. This is achieved by handling ItemInserting and ItemUpdating events of the DetailsView. The relevant code is shown in Figure 28.

```
Protected Sub DetailsView1_ItemInserting(ByVal sender As
Object, ByVal e As
System.Web.UI.WebControls.DetailsViewInsertEventArgs)
Handles DetailsView1.ItemInserting
    e.Values("SurveyID") = DropDownList1.SelectedValue
End Sub

Protected Sub DetailsView1_ItemUpdating(ByVal sender As
Object, ByVal e As
System.Web.UI.WebControls.DetailsViewUpdateEventArgs)
Handles DetailsView1.ItemUpdating
    e.NewValues("SurveyID") = DropDownList1.SelectedValue
End Sub
```

Figure 28: Handling ItemInserting and ItemUpdating events of DetailsView

The event handlers set the SurveyID parameter from Values and NewValues collections with the one selected in the DropDownList. Figure 29 shows the complete markup of ManageParticipants.aspx

```
<%@ Page Language="VB"
MasterPageFile="~/Admin/AdminMasterPage.master"
AutoEventWireup="false"
CodeFile="ManageParticipants.aspx.vb"
Inherits="ManageParticipants" title="Untitled Page" %>
<asp:Content ID="Content1"
ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<asp:Label ID="Label2" runat="server" Text="<h3>Manage
Participants</h3>"></asp:Label>
<table width="100%">
<tr>
<td align="right">
<asp:Label ID="Label1" runat="server" Text="Select Survey
:"></asp:Label></td>
<td align="left">
<asp:DropDownList ID="DropDownList1" runat="server"
DataSourceID="SqlDataSource1"
DataTextField="Title" DataValueField="SurveyID">
</asp:DropDownList></td>
</tr>
<tr>
<td colspan="2">
<asp:DetailsView ID="DetailsView1" runat="server"
AllowPaging="True" AutoGenerateRows="False"
CellPadding="4" DataKeyNames="ContactID"
DataSourceID="SqlDataSource2" ForeColor="#333333"
GridLines="None" Height="50px" Width="100%">
<FooterStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
<CommandRowStyle BackColor="#E2DED6" Font-Bold="True" />
<EditRowStyle BackColor="#999999" />
<RowStyle BackColor="#F7F6F3" ForeColor="#333333" />
<PagerStyle BackColor="#284775" ForeColor="White"
HorizontalAlign="Center" />
<Fields>
<asp:BoundField DataField="ContactID" HeaderText="Contact
ID" InsertVisible="False"
ReadOnly="True" SortExpression="ContactID" />
<asp:BoundField DataField="Name" HeaderText="Name"
SortExpression="Name" />
<asp:BoundField DataField="Email" HeaderText="Email"
SortExpression="Email" />
```

```

<asp:CommandField ShowDeleteButton="True"
ShowEditButton="True" ShowInsertButton="True" />
</Fields>
<FieldHeaderStyle BackColor="#E9ECF1" Font-Bold="True"
Width="10%" Wrap="False" />
<HeaderStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
<AlternatingRowStyle BackColor="White" ForeColor="#284775"
/>
</asp:DetailsView>
</td>
</tr>
</table>
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%= $ConnectionStrings:ConnectionString
%>"
SelectCommand="SELECT [SurveyID], [Title] FROM
[Survey]"></asp:SqlDataSource>
<asp:SqlDataSource ID="SqlDataSource2" runat="server"
ConnectionString="<%= $ConnectionStrings:ConnectionString
%>"
DeleteCommand="DELETE FROM [SurveyParticipants] WHERE
[ContactID] = @ContactID"
InsertCommand="INSERT INTO [SurveyParticipants]
([SurveyID], [Name], [Email]) VALUES (@SurveyID, @Name,
@email)"
SelectCommand="SELECT * FROM [SurveyParticipants] WHERE
([SurveyID] = @SurveyID)"
UpdateCommand="UPDATE [SurveyParticipants] SET [SurveyID] =
@SurveyID, [Name] = @Name, [Email] = @Email WHERE
[ContactID] = @ContactID">
<DeleteParameters>
<asp:Parameter Name="ContactID" Type="Int32" />
</DeleteParameters>
<UpdateParameters>
<asp:Parameter Name="SurveyID" Type="Int32" />
<asp:Parameter Name="Name" Type="String" />
<asp:Parameter Name="Email" Type="String" />
<asp:Parameter Name="ContactID" Type="Int32" />
</UpdateParameters>
<SelectParameters>
<asp:ControlParameter ControlID="DropDownList1"
Name="SurveyID" PropertyName="SelectedValue"
Type="Int32" />
</SelectParameters>
<InsertParameters>
<asp:Parameter Name="SurveyID" Type="Int32" />

```



```
<asp:Parameter Name="Name" Type="String" />
<asp:Parameter Name="Email" Type="String" />
</InsertParameters>
</asp:SqlDataSource>
</asp:Content>
```

Figure 29: Markup of ManageParticipants.aspx

This completes the major part of our administrative pages.

## **Conclusion**

In this part we developed web forms for managing survey questions and their choices. A question can have single choice, multiple choice or free text answers. Depending on the answer type we need to render RadioButtons, CheckBoxes or TextBoxes when the user takes the survey. Each survey has participants which are notified about the survey by sending an email. We developed a web form for managing these participants. In the next part we will see how to send invitations to the participants, generate the survey form and generate survey statistics.

*Bipin Joshi is the founder and owner of BinaryIntellect Consulting ([www.binaryintellect.com](http://www.binaryintellect.com)) where he conducts professional training programs on .NET technologies. He is the author and publisher of Developer's Guide to ASP.NET 2.0 and co-author of three WROX press books on .NET 1.x. He runs two community web sites dedicated to .NET articles and code samples - [www.dotnetbips.com](http://www.dotnetbips.com), and [www.binaryintellect.net](http://www.binaryintellect.net). He is a Microsoft MVP, member of ASPInsiders, MCAD and MCT. When away from computers he spends time practicing Yoga and meditation. You can contact him via his blog at [www.bipinjoshi.com](http://www.bipinjoshi.com) where he writes about life, Yoga and spirituality*