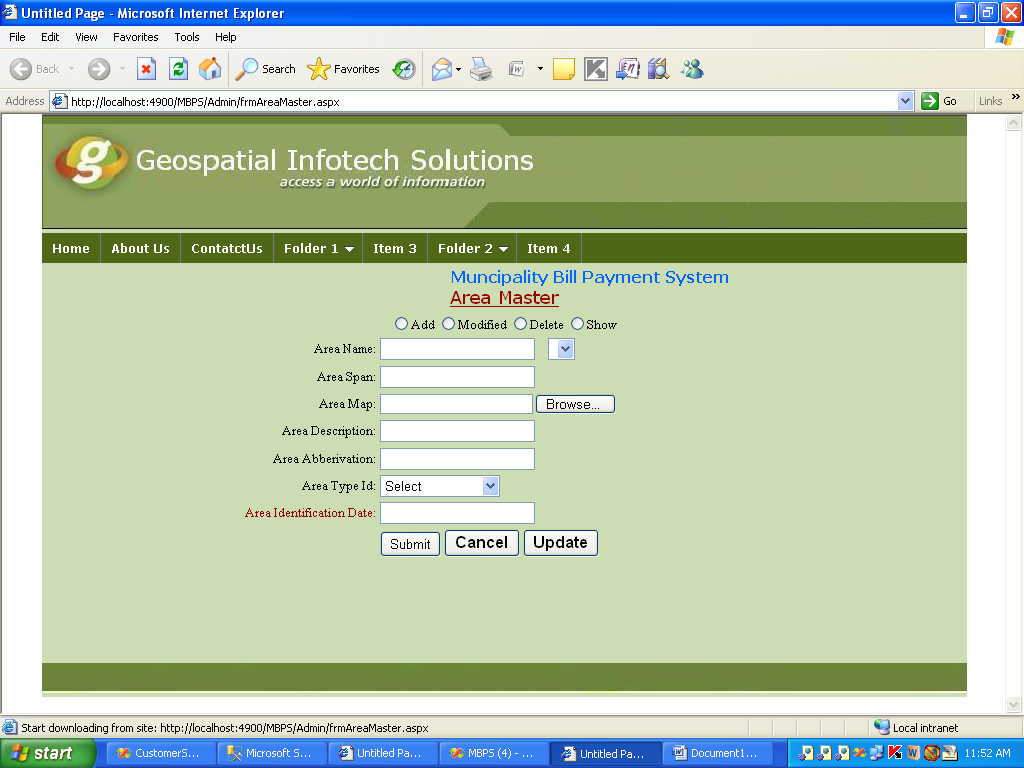
**Geaspacial Info Tech Solutions”**

**AreaMaster.aspx:**

****

**Html:**

<%@ Page Language="C#" MasterPageFile="~/Admin/MasterPage.master" AutoEventWireup="true" CodeFile="frmAreaMaster.aspx.cs" Inherits="Admin\_frmAreaMaster" Title="Untitled Page" %>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">

<table width="100%" border="0" cellpadding="0" cellpadding="0" >

<tr>

<td align="center" >

<marquee behavior="alternate"

style="font-weight: 700; font-size: medium; color: #0066FF; font-family: Verdana;">Muncipality Bill Payment System</marquee>

<span class="style2">Area Master</span>

</td>

</tr>

<tr>

<td>

<table width="100%" >

<tr>

<td colspan ="5" align="center">

<asp:RadioButton ID="RadioButton1" runat="server" GroupName="a" Text="Add"

AutoPostBack="True" oncheckedchanged="RadioButton1\_CheckedChanged" />

<asp:RadioButton ID="RadioButton2" runat="server" GroupName="a"

Text="Modified" AutoPostBack="True"

oncheckedchanged="RadioButton2\_CheckedChanged" />

<asp:RadioButton ID="RadioButton3" runat="server" GroupName="a" Text="Delete"

oncheckedchanged="RadioButton3\_CheckedChanged" />

<asp:RadioButton ID="RadioButton4" runat="server" GroupName="a" Text="Show"

oncheckedchanged="RadioButton3\_CheckedChanged" AutoPostBack="True" />

</td>

</tr>

<tr>

<td style="text-align: right" >

Area Name:

</td>

<td style="text-align: left">

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

<asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"

ErrorMessage="\*" ControlToValidate="txtAreaName"></asp:RequiredFieldValidator>

<asp:DropDownList ID="ddlAreaId" runat="server" AutoPostBack="True"

onselectedindexchanged="ddlAreaId\_SelectedIndexChanged">

</asp:DropDownList>

<asp:RequiredFieldValidator ID="RequiredFieldValidator17" runat="server"

ControlToValidate="ddlAreaId" ErrorMessage="RequiredFieldValidator">\*</asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td style="text-align: right">

Area Span:

</td>

<td style="text-align: left">

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

<asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server"

ErrorMessage="\*" ControlToValidate="TxtAreaSpan"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td style="text-align: right">

Area Map:

</td>

<td style="text-align: left">

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

<asp:Image ID="Image1" runat="server" Height ="20" Width ="50"

Visible="False"/>

</td>

</tr>

<tr>

<td style="text-align: right">

Area Description:

</td>

<td style="text-align: left">

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

<asp:RequiredFieldValidator ID="RequiredFieldValidator13" runat="server"

ErrorMessage="\*" ControlToValidate="txtAreaDescription"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td style="text-align: right">

Area Abberivation:

</td>

<td style="text-align: left">

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

<asp:RequiredFieldValidator ID="RequiredFieldValidator14" runat="server"

ErrorMessage="\*" ControlToValidate="txtAreaAbbervation"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td style="text-align: right">

Area Type Id:

</td>

<td style="text-align: left">

<asp:DropDownList ID="ddlAreaTypeId" runat="server" Height="16px" Width="120px">

</asp:DropDownList>

</td>

</tr>

<tr>

<td style="text-align: right" class="style1">

Area Identification Date:

</td>

<td style="text-align: left" class="style1">

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

<asp:RequiredFieldValidator ID="RequiredFieldValidator16" runat="server"

ErrorMessage="\*" ControlToValidate="txtAreaIdentificationDate"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td style="text-align: right">

&nbsp;</td>

<td style="text-align: left">

<asp:Button ID="Button1" runat="server" onclick="Submit\_Click" Text="Submit"

style="height: 26px" />

<asp:Button ID="BtnCacel" runat="server" Text="Cancel" OnClick="BtnCacel\_Click" Width="76px"

CausesValidation="False" style="font-weight: 700; font-size: medium" />

<asp:Button ID="BtnUpdate" runat="server" Text="Update" OnClick="BtnUpdate\_Click" Width="76px"

CausesValidation="False" style="font-weight: 700; font-size: medium" />

<asp:Label ID="lblmessage" runat="server" BorderStyle="None" Text="Label"

Visible="False"></asp:Label>

</td>

</tr>

<tr>

<td style="text-align: right" align="center" colspan="2">

<div width="100%">

<center>

<asp:GridView ID ="grd" Width="900px" runat ="server" CellPadding="4" ForeColor="#333333"

GridLines="None" onselectedindexchanged="grd\_SelectedIndexChanged"

Visible="False">

<FooterStyle BackColor="#1C5E55" Font-Bold="True" ForeColor="White" />

<RowStyle BackColor="#E3EAEB" />

<PagerStyle BackColor="#666666" ForeColor="White" HorizontalAlign="Center" />

<SelectedRowStyle BackColor="#C5BBAF" Font-Bold="True" ForeColor="#333333" />

<HeaderStyle BackColor="#1C5E55" Font-Bold="True" ForeColor="White" />

<EditRowStyle BackColor="#7C6F57" />

<AlternatingRowStyle BackColor="White" />

</asp:GridView>

</center>

</div>

</td>

</tr>

</table>

</td>

</tr>

</table>

</asp:Content>

<asp:Content ID="Content3" runat="server" contentplaceholderid="head">

<link href="../jqueryslidemenu.css" rel="stylesheet" type="text/css" />

<script src="../CS/jquery.min.js" type="text/javascript"></script>

<script src="../CS/jqueryslidemenu.js" type="text/javascript"></script>

<style type="text/css">

.style1

{

color: #990000;

}

.style2

{

color: #990000;

font-size: large;

font-family: Verdana;

text-decoration: underline;

}

</style>

</asp:Content>

**Areamaster.cs**

using System;

using System.Collections;

using System.Configuration;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using System.Data.SqlClient;

public partial class Admin\_frmAreaMaster : System.Web.UI.Page

{

ClsAreaMaster AreaMaster = new ClsAreaMaster();

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

{

if (!IsPostBack)

{

AreaTypeId();

GETDATA();

}

}

public void AreaTypeId()

{

try

{

DataSet ds = new DataSet();

ds = AreaMaster.SelectAreaTYpeId ();

ddlAreaTypeId.DataSource = ds;

ddlAreaTypeId.DataTextField = "AreaTYpeName";

ddlAreaTypeId.DataValueField = "AreaTypeId";

ddlAreaTypeId.DataBind();

ddlAreaTypeId.Items.Insert(0, "Select");

}

catch (Exception ex)

{

lblmessage.Text = ex.Message;

}

}

public void AreaId()

{

try

{

DataSet ds = new DataSet();

ds = AreaMaster.SelectAreaId();

ddlAreaId.DataSource = ds;

ddlAreaId.DataTextField = "AreaName";

ddlAreaId.DataValueField = "AreaId";

ddlAreaId.DataBind();

ddlAreaId.Items.Insert(0, "Select");

}

catch (Exception ex)

{

lblmessage.Text = ex.Message;

}

}

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

{

try

{

if (RadioButton1.Checked == true)

{

string s = "~/fileipload/" + fupAreaMap.FileName;

fupAreaMap.SaveAs(Server.MapPath(s));

AreaMaster.AreaName = txtAreaName.Text;

AreaMaster.AreaSpan = Convert.ToDecimal(TxtAreaSpan.Text);

AreaMaster.AreaMap = s.ToString();

AreaMaster.AreaDescription = txtAreaDescription.Text;

AreaMaster.AreaAbbervation = txtAreaAbbervation.Text;

AreaMaster.AreaTypeId = Convert.ToInt16(ddlAreaTypeId.SelectedValue );

AreaMaster.AreaIdentificationDate = Convert.ToDateTime(txtAreaIdentificationDate.Text);

AreaMaster.InsertAreaMaster();

lblmessage.Visible = true;

GETDATA();

lblmessage.Text = "inserted";

}

else if (RadioButton2.Checked == true)

{

}

}

catch (Exception ex)

{

lblmessage.Text = ex.Message.ToString();

}

}

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

{

CancelDate();

}

void CancelDate()

{

Control ctl = this.Master.FindControl("ContentPlaceHolder1");

foreach (Control c in ctl.Controls)

{

//Clear Textbox

if (c is TextBox)

{

((TextBox)c).Text = "";

}

//Reset DropDownList

if (c is DropDownList)

{

for (int i = 0; i < ((DropDownList)c).Items.Count; i++)

((DropDownList)c).Items[i].Selected = false;

((DropDownList)c).Items[0].Selected = true;

}

//Uncheck Selected CheckBox

if (c is CheckBox)

{

((CheckBox)c).Checked = false;

}

}

}

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

{

string s = "~fileupload" + fupAreaMap.FileName;

fupAreaMap.SaveAs(s);

AreaMaster.AreaName = txtAreaName.Text;

AreaMaster.AreaSpan = Convert.ToInt16(TxtAreaSpan.Text);

AreaMaster.AreaMap = s.ToString();

AreaMaster.AreaTypeId = Convert.ToInt16(ddlAreaTypeId.Text);

AreaMaster.AreaIdentificationDate = Convert.ToDateTime(txtAreaIdentificationDate.Text);

AreaMaster.UpdateAreaMaster();

GETDATA();

}

protected void RadioButton1\_CheckedChanged(object sender, EventArgs e)

{

ddlAreaId.Visible = false;

RequiredFieldValidator17.Visible = false;

txtAreaName.Visible = true;

}

protected void RadioButton2\_CheckedChanged(object sender, EventArgs e)

{

txtAreaName.Visible = false;

ddlAreaId.Visible = true;

RequiredFieldValidator17.Visible = true;

AreaId();

}

protected void RadioButton3\_CheckedChanged(object sender, EventArgs e)

{

grd.Visible = true;

}

protected void grd\_SelectedIndexChanged(object sender, EventArgs e)

{

}

protected void ddlAreaId\_SelectedIndexChanged(object sender, EventArgs e)

{

try

{

int AreaId = Convert.ToInt32(ddlAreaId.SelectedValue);

DataSet ds = AreaMaster.GetAreaIdSelect(AreaId );

if (ds.Tables[0].Rows.Count != 0)

{

DataTable dt = ds.Tables[0];

txtAreaName .Text=dt.Rows[0][1].ToString ();

TxtAreaSpan.Text = dt.Rows[0][2].ToString();

Image1.ImageUrl = dt.Rows[0][3].ToString();

txtAreaDescription.Text = dt.Rows[0][4].ToString();

txtAreaAbbervation.Text = dt.Rows[0][5].ToString();

ddlAreaTypeId.SelectedValue = dt.Rows[0][6].ToString();

txtAreaIdentificationDate.Text = dt.Rows[0][7].ToString();

}

}

catch (Exception)

{

throw;

}

}

void GETDATA()

{

try

{

DataSet ds= AreaMaster.SelectAreaId();

grd.DataSource = ds;

grd.DataBind();

}

catch(Exception ex)

{

lblmessage.Text =ex.Message ;

}

}

}

**ClassFile:**

using System;

using System.Data;

using System.Configuration;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using System.Data.SqlClient;

using EmployeeFunction.DAL;

/// <summary>

/// Summary description for ClsAreaMaster

/// </summary>

public class ClsAreaMaster

{

public ClsAreaMaster()

{

//

// TODO: Add constructor logic here

//

}

public static DataSet ds;

public int AreaId { get; set; }

public string AreaName { get; set; }

public decimal AreaSpan{ get; set; }

public string AreaDescription { get; set; }

public string AreaMap { get; set; }

public int AreaTypeId{ get; set; }

public string AreaAbbervation { get; set; }

public DateTime AreaIdentificationDate{ get; set; }

public void InsertAreaMaster()

{

SqlParameter[] p=new SqlParameter[7];

try

{

p[0]=new SqlParameter("@AreaName",this.AreaName);

p[1]=new SqlParameter("@AreaSpan",this.AreaSpan);

p[2]=new SqlParameter("@Areamap",this.AreaMap);

p[3]=new SqlParameter("@AreaTypeId",this.AreaTypeId);

p[4] = new SqlParameter("@AreaDescription", this.AreaDescription);

p[5] = new SqlParameter("@AreaAbbervation", this.AreaAbbervation);

p[6] = new SqlParameter("@AreaIdentifiedDate", this.AreaIdentificationDate);

SqlHelper.ExecuteNonQuery(ConnectionString.Conn, CommandType.StoredProcedure, "Sp\_Insert\_AreaMaster", p);

}

catch (Exception ex)

{

throw new ArgumentException(ex.Message);

}

}

public DataSet SelectAreaTYpeId()

{

try

{

string s = "sp\_selectAreaTypeMaster";

return SqlHelper.ExecuteDataset(ConnectionString.Conn, CommandType.StoredProcedure, s);

}

catch (Exception)

{

throw;

}

}

public DataSet SelectAreaId()

{

try

{

string s = "sp\_SelctAreaMaster";

return SqlHelper.ExecuteDataset(ConnectionString.Conn, CommandType.StoredProcedure, s);

}

catch (Exception)

{

throw;

}

}

public void UpdateAreaMaster()

{

SqlParameter[] p = new SqlParameter[5];

try

{

p[0] = new SqlParameter("@AreaName", this.AreaName);

p[1] = new SqlParameter("@AreaSpan", this.AreaSpan);

p[2] = new SqlParameter("@Areamap", this.AreaMap);

p[3] = new SqlParameter("@AreaTypeId", this.AreaTypeId);

p[4] = new SqlParameter("@AreaIdentificationDate", this.AreaIdentificationDate);

SqlHelper.ExecuteNonQuery(ConnectionString.Conn, CommandType.StoredProcedure, "tbl\_AreaMaster\_Update", p);

}

catch (Exception ex)

{

throw new ArgumentException(ex.Message);

}

}

public DataSet GetAreaIdSelect(int AreaId)

{

try

{

SqlParameter p = new SqlParameter("@AreaId", AreaId);

ds = new DataSet();

ds = SqlHelper.ExecuteDataset(ConnectionString .Conn , CommandType.StoredProcedure, "sp\_SelectAreaMaster", p);

return ds;

}

catch (Exception ex)

{

throw new ArgumentException(ex.Message);

}

}

}