WEB SERVICES DE AD http://10.57.254.25/WSAD/Authenticationservice.asmx

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.Services.Description;

using System.Data;

using System.Configuration;

namespace genera\_identificador

{

public partial class Login : System.Web.UI.Page

{

string Autentica;

public static String ConnString2 = Cryptografia.Decrypt(ConfigurationManager.ConnectionStrings["ConnString2"].ConnectionString.ToString());

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

{

}

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

{

try

{

// Se conecta al WS

WebReference.AuthenticationService ws = new WebReference.AuthenticationService();

ws.Credentials = System.Net.CredentialCache.DefaultNetworkCredentials;

Boolean b = ws.ValidarUsuario(TxtUsuario.Text, TxtContrasena.Text, "BAC\_NT");

DataTable dtValidar = new DataTable();

String error = string.Empty;

UsuarioClass objuser = new UsuarioClass();

objuser.idproyecto = "053";

objuser.username = TxtUsuario.Text;

DataSet ds = new DataSet();

DataSet ObtenerRol = new DataSet();

if (ws.ValidarUsuario(TxtUsuario.Text, TxtContrasena.Text, "BAC\_NT"))

{

ds = Negocio.Obtenerautentica(objuser, ConnString2);

Autentica = ds.Tables["Table"].Rows[0][0].ToString();

if (ds.Tables["Table"].Rows[0][0].ToString() == "1")

{

ObtenerRol = Negocio.ObtieneRol(objuser, ConnString2);

//ds = objuser.ObtieneRol(objuser, ref error);

objuser.rol = ObtenerRol.Tables[0].Rows[0][4].ToString();

// OpcionesClass objop = new OpcionesClass();

//objop = Negocio.ObtenerOpciones(objuser, ConnString);

Session["UsuarioNombre"] = objuser.username;

Session["UsuarioRol"] = objuser.rol;

Session["Acceso"] = Autentica;

Response.Redirect("~/Default.aspx");

}

else

{

LblMensaje.Text = "Usuario o Contraseña Incorrecta.";

}

}

else

{

LblMensaje.Text = "Usuario o Contraseña Incorrecta.";

}

}

catch (System.Net.WebException exc)

{

LblMensaje.Text = "Error al conectar con el Servidor Remoto";

}

}

}

}

public static DataSet Obtenerautentica(UsuarioClass objuser, string conn)

{

return AccesoDatos.autentica(objuser, conn);

}

//Autentica el usuario logeado

public static DataSet autentica(UsuarioClass user, string conn)

{

DataSet resultados = new DataSet();

SqlConnection con = new SqlConnection(conn);

try

{

SqlCommand cmd = new SqlCommand();

cmd.CommandType = CommandType.StoredProcedure;

cmd.CommandText = "SECP\_AutorizadoAlModulo";

cmd.Parameters.Add("@CodigoModulo", SqlDbType.VarChar).Value = user.idproyecto;

cmd.Parameters.Add("@CodigoUsuario", SqlDbType.VarChar).Value = user.username;

cmd.Connection = con;

SqlDataAdapter da = new SqlDataAdapter(cmd);

da.Fill(resultados);

da.Dispose();

return resultados;

}

catch (Exception ex)

{

return resultados;

}

}

public static DataSet ObtieneRol(UsuarioClass objuser, string conn)

{

return AccesoDatos.ObtieneRol(objuser, conn);

}

//Obtiene que tipo de rolo tiene el usurio para el acceso a las pantallas

public static DataSet ObtieneRol(UsuarioClass user, string conn)

{

SqlConnection con = new SqlConnection(conn);

DataSet resultados = new DataSet();

try

{

SqlCommand cmd = new SqlCommand();

cmd.CommandType = CommandType.StoredProcedure;

cmd.CommandText = "SECP\_RolModulo";

cmd.Parameters.Add("@CodigoModulo", SqlDbType.VarChar).Value = user.idproyecto;

cmd.Parameters.Add("@CodigoUsuario", SqlDbType.VarChar).Value = user.username;

cmd.Connection = con;

SqlDataAdapter da = new SqlDataAdapter(cmd);

da.Fill(resultados);

da.Dispose();

return resultados;

}

catch (Exception ex)

{

return resultados;

}

}