1. Aplicar Diseño a Pantalla:

Dentro de los divs que contienen controles de formularios, ***todos los elementos*** deben estar dentro de una sola etiqueta <ul> y separados por la etiqueta <li> para que tomen la estructura basada en listas en listas, de la siguiente forma:

<li>

<label for="PaginaDeContenido\_txtNomProveedor" runat="server">

<asp:Label ID="lblNomProveedor" runat="server" Text="Nombre Proveedor\*"> </asp:Label></label>

<asp:TextBox ID="txtNomProveedor" runat="server" ToolTip="Nombre del Proveedor" CssClass="itext"></asp:TextBox>

</li>

<li>

<label for="PaginaDeContenido\_txtRazonSocial">

<asp:Label ID="lblRazonSocial" runat="server" Text="Razón Social\*" ></asp:Label></label>

<asp:TextBox ID="txtRazonSocial" runat="server" ToolTip="Razón Social del proveedor" CssClass="itext" ></asp:TextBox>

</li>

<li>

<label for="PaginaDeContenido\_txtNomContacto">

<asp:Label ID="lblNomContacto" runat="server" Text="Nombre de Contacto" ></asp:Label></label>

<asp:TextBox ID="txtNomContacto" runat="server" ToolTip="Nombre de Contacto del proveedor" CssClass="itext"></asp:TextBox>

</li>

A excepción de cuando se desea usar un listradiobutton, se sigue la siguiente estructura:

<li>

<label for="lblStatus"><asp:Label runat="server" ID="lblStatus">Estatus\*</asp:Label></label>

<ul>

<li>

<asp:RadioButton ID="rdbActivo" runat="server" CssClass="iradio" Checked="false" ToolTip="Activo" GroupName="Estatus" /><label for="lblActivo" class="ilabel"><asp:Label runat="server" ID="lblActivo">Activo</asp:Label></label></li>

<li>

<asp:RadioButton ID="rdbInactivo" runat="server" CssClass="iradio" Checked="false" ToolTip="Inactivo" GroupName="Estatus" /> <label for="lblInactivo" class="ilabel"><asp:Label runat="server"

ID="lblInactivo">Inactivo</asp:Label></label></li>

</ul>

</li>

NOTA: Para más información al respecto revisar otros formularios ya hechos.

1. Estándares de Programación

* En el código html no usar: &nbsp; ó <br> ya que el estilo que se aplica al ser un lista automáticamente le da los espacios requeridos.
* Revisar que en ninguna parte diga: Modulo, sino mas bien: Proceso.
* Verificar que todos los controles usados en html tengan un ID correcto, que describa el papel que realizan: Que no diga: Label4 Sino mas bien: lblAlgo Siguiendo el estándar del nombrado de controles.
* En los DropDownList hasta el momento en otras pantallas hemos usado que el primer ítem diga: Seleccione. Para que todos queden igual quitar:<>
* Cambiar nombrado del control ReportViewer1, se debe llamar algo asi como: rpvRoles.
* No usar números mágicos en el código, por ejemplo:

claveEstadoEntidad = 0; Mas bien cambiar el numero por un enumerador que describa lo que significa el numero 0.

* No usar cadenas en el código, cambiarlas por constantes. La clase donde se encuentran las constantes esta en la capa de Datos, carpeta de Constantes, en la clase Mensajes. Encontraras todas las constantes hasta el momento agregadas. Puede que algunas de las palabras que tienes como cadenas ya estén agregadas como constantes. Solo tendras que agregar las que no estén.
* Comentar todos los métodos, para ello usar algo como esto:

/// <summary>

/// Evento que ocurre al cargarse la página.

/// </summary>

/// <param name="sender">Objeto que produce el evento</param>

/// <param name="e">Parámetros del evento</param>

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

{

QVC.Negocios.Session.UserSession usuarioLogueado = (QVC.Negocios.Session.UserSession)Session[ConstanteCampo.C\_UsuarioLogueado];

if (usuarioLogueado != null)

{

if (!Page.IsPostBack)

{

Seguridad s = new Seguridad();

if (s.BuscarFuncion(usuarioLogueado, Seguridad.Procesos.AdministradorSoftware))

{

ClaveEmpresa = usuarioLogueado.CveEntEmpresa;

ClaveUsuario = usuarioLogueado.CveEntUsuario;

IP = usuarioLogueado.IP;

ViewCatalogs.ActiveViewIndex = Convert.ToInt32(Datos.Numeradores.VistasMultiView.VistaGrid);

CargarProcesos();

}

else

{

Response.Redirect(Mensajes.C\_UrlGR\_002A);

}

}

CargarScripts();

}

else

Response.Redirect(Mensajes.C\_UrlGR\_002);

}

Para describir lo que hace el método y sus parámetros que recibe.

* En los Dacs no usar nombres de propiedades como: ‘roldes’, mas bien rolDescripcion.

No usar abreviaturas de palabras y para identificar una palbra de otra empezar partir de 2da. Palabra con mayúsculas.

* Verificar que los comentarios en las propiedades correspondan a la propiedad, ejemplo:

private String funcdes;

/// <summary>

/// Obtiene o establece la propiedad de Estatus

/// </summary>

/// <value>ClaveEntidad tipo de dato String.</value>

public String Funcdes

{

get { return this.funcdes; }

set { this.funcdes = value; }

}

No corresponde Funcdes a lo que dice su comentario.

* Falta comentar la clase RolReporteFactoria. Revisar como están otras factorías.
* En los dacs, falta el encabezado:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Develop By: MAI-HummingBird \*/

/\* Software Developer: María Elena Cervantes Santos

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/