Agregar Búsquedas Modales al Formulario de Compras con: JQuery, AJAX & JSon

1. Agregue un botón para buscar el producto, junto antes del productoTextBox

<input id="buscarProductoButton" type="button" value="..." />

1. Antes del último asp:content creemos la división que servirá de búsqueda para los productos

<div id="dialog-search-producto" title="Búsqueda de productos">

Críterio:

<asp:TextBox ID="criterioProductoTextBox" CssClass="criterioProductoTextBox" runat="server"></asp:TextBox>

<input id="buscarProducto2Button" type="button" value="Buscar" />

<div id="dataProducto"></div>

</div>

1. Agregamos el script que hace el llamado del dialogo, después del efecto de date picker:

$("#dialog-search-proveedor").hide();

$("#buscarProductoButton").click(function () {

$("#dialog-search-producto").dialog({

resizable: false,

height: 500,

width: 700,

modal: true,

buttons: {

"Cerrar": function () {

$(this).dialog("close");

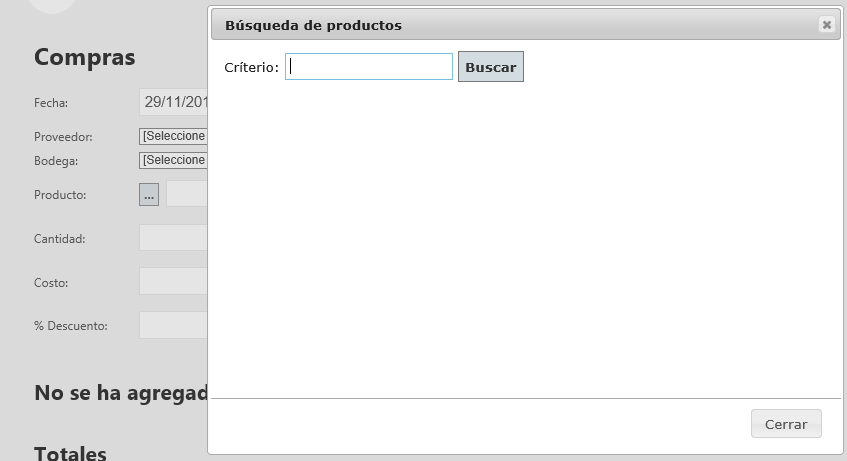
}

}

});

});

1. Probemos lo que llevamos hasta el momento:



1. Agreguemos el código que llena la tabla de búsqueda dependiendo del criterio ingresado, antes de cerrar el $(function adicione estas líneas al script

$("#buscarProducto2Button").click(function () {

var pageUrl = '<%= ResolveUrl("~/Movimientos/Compras.aspx/GetRecordsProducto") %>';

var criterio = $("#<%= criterioProductoTextBox.ClientID %>").val();

var parameter = { "criterio": criterio }

$.ajax({

url: pageUrl,

type: "POST",

data: JSON.stringify(parameter),

contentType: "application/json; charser=utf-8",

datatype: "json",

success: function (result) {

$("#dataProducto").empty();

var data = JSON.parse(result.d);

var array = data.Table;

var temp = "<table><tr><th>ID Producto</th><th>Descripción</th><th>Selección</th></tr>";

for (var i = 0; i < array.length; i++) {

temp += "<tr>";

temp += "<td><label>" + array[i].IDProducto + "</label></td>";

temp += "<td><label>" + array[i].Descripcion + "</label></td>";

temp += "<td><input itag='Select' type='button' value='Seleccionar' onclick='return SelectRecordProducto(this)'/></td>";

temp += "</tr>";

}

temp += "</table>";

$("#dataProducto").append(temp);

},

error: function (err) {

alert("Fail");

}

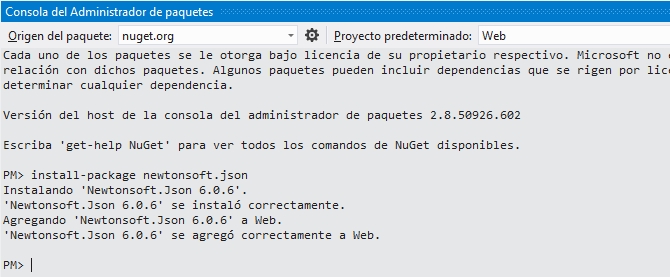
});

});

1. Para poder probar, debemos crear el Web Method GetRecordsProducto , pero primero ebemos de agregar un paquete JSon a nuestro proyecto. En herramientas -> Administrador de paquetes NuGet -> Consola del administrador de paquetes, ejecute el siguiente comando:

install-package newtonsoft.json

Debe aparecer una ventana como esta:



1. Ahora si, procedamos a crear el Web Method GetRecordsProducto:

[WebMethod]

public static string GetRecordsProducto(string criterio)

{

SqlConnection cn = new SqlConnection(ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString);

SqlDataAdapter da = new SqlDataAdapter("SELECT IDProducto, Descripcion FROM Producto WHERE Descripcion LIKE '" + criterio + "%' ORDER BY Descripcion", cn);

DataSet ds = new DataSet();

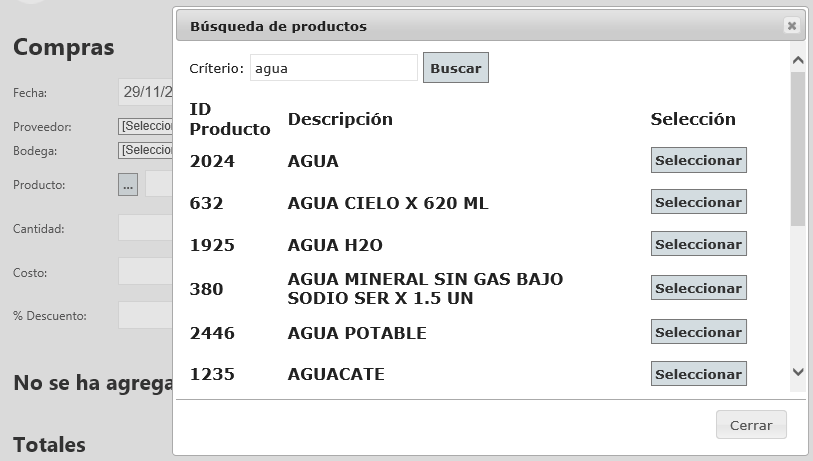
da.Fill(ds);

string data = JsonConvert.SerializeObject(ds, Formatting.Indented);

return data;

}

1. Probemos lo que llevamos hasta el momento:



1. Agregue la propiedad CssClass al productoTextBox y al consultarButton, por orden se recomienda que sea el mismo id del control

<asp:TextBox ID="productoTextBox" CssClass="productoTextBox" runat="server" Width="135px"></asp:TextBox>

<asp:Button ID="consultarButton" CssClass="consultarButton" runat="server" Text="Consultar" OnClick="consultarButton\_Click" />

1. Ahora agreguemos el código para seleccionar el producto buscado. Agregue estas líneas al empezar el script:

function SelectRecordProducto(control) {

var ID = $(control).closest("tr").find("label").html();

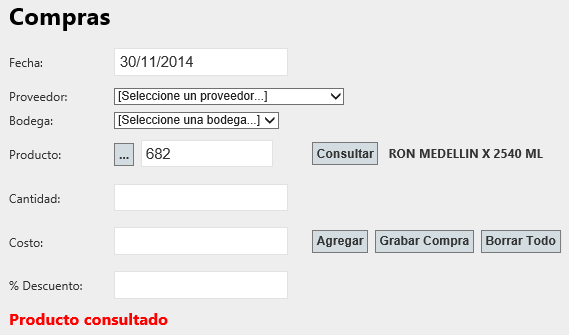
$(".productoTextBox").val(ID);

$(".consultarButton").click();

$("#dialog-search-producto").dialog("close");

}

1. Probemos la búsqueda:



Taran!!!!!

1. Ya que implementamos la búsqueda de productos, ahora implementemos la búsqueda de proveedores. Iniciamos creando el botón de búsqueda de proveedores, justo antes del drop down list de proveedores.

<input id="buscarProveedorButton" type="button" value="..." />

1. Creemos la división que contendrá la búsqueda, agregue estas líneas antes del último asp:content

<div id="dialog-search-proveedor" title="Búsqueda de proveedor">

Críterio:

<asp:TextBox ID="criterioProveedorTextBox" CssClass="criterioProveedorTextBox" runat="server"></asp:TextBox>

<input id="buscarProveedor2Button" type="button" value="Buscar" />

<div id="dataProveedor"></div>

</div>

1. Agreguemos estas líneas, luego del hide de la división de búsqueda de producto:

$("#dialog-search-proveedor").hide();

$("#buscarProveedorButton").click(function () {

$("#dialog-search-proveedor").dialog({

resizable: false,

height: 500,

width: 700,

modal: true,

buttons: {

"Cerrar": function () {

$(this).dialog("close");

}

}

});

});

1. Probemos lo que llevamos hasta el momento para la búsqueda de proveedor:



1. Implementemos el código que hace la búsqueda de registros en la base de datos. Agregue estas líneas antes de cerrar el $function(

$("#buscarProveedor2Button").click(function () {

var pageUrl = '<%= ResolveUrl("~/Movimientos/Compras.aspx/GetRecordsProveedor") %>';

var criterio = $("#<%= criterioProveedorTextBox.ClientID %>").val();

var parameter = { "criterio": criterio }

$.ajax({

url: pageUrl,

type: "POST",

data: JSON.stringify(parameter),

contentType: "application/json; charser=utf-8",

datatype: "json",

success: function (result) {

$("#dataProveedor").empty();

var data = JSON.parse(result.d);

var array = data.Table;

var temp = "<table><tr><th>ID Proveedor</th><th>Nombre Comercial</th><th>Selección</th></tr>";

for (var i = 0; i < array.length; i++) {

temp += "<tr>";

temp += "<td><label>" + array[i].IDProveedor + "</label></td>";

temp += "<td><label>" + array[i].Nombre + "</label></td>";

temp += "<td><input itag='Select' type='button' value='Seleccionar' onclick='return SelectRecordProveedor(this)'/></td>";

temp += "</tr>";

}

temp += "</table>";

$("#dataProveedor").append(temp);

},

error: function (err) {

alert("Fail");

}

});

});

1. Agreguemos el Web Method de búsqueda de registros de proveedor:

[WebMethod]

public static string GetRecordsProveedor(string criterio)

{

SqlConnection cn = new SqlConnection(ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString);

SqlDataAdapter da = new SqlDataAdapter("SELECT IDProveedor, Nombre FROM Proveedor WHERE Nombre LIKE '" + criterio + "%' ORDER BY Nombre", cn);

DataSet ds = new DataSet();

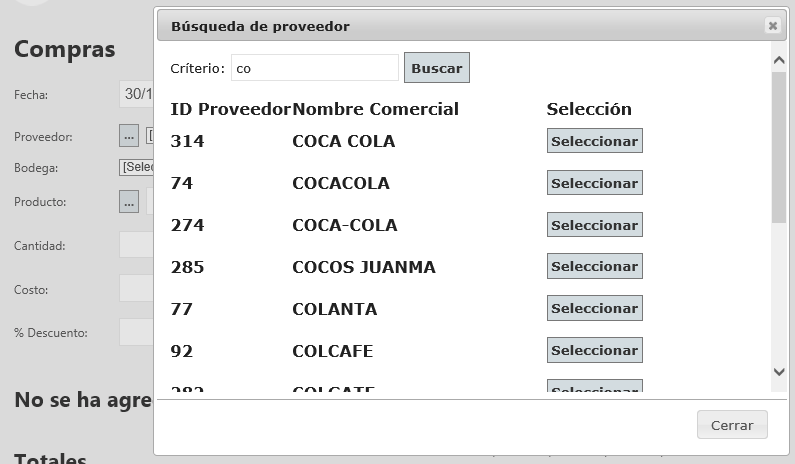
da.Fill(ds);

string data = JsonConvert.SerializeObject(ds, Formatting.Indented);

return data;

}

1. Probemos lo que llevamos hasta el momento:



1. Coloquemos la propiedad CssClass al Drop Down List de proveedor:

<asp:DropDownList ID="proveedorDropDownList" CssClass="proveedorDropDownList" runat="server" DataSourceID="provedoresSqlDataSource" DataTextField="Nombre" DataValueField="IDProveedor">

1. E implementemos las líneas del botón Seleccionar. Al iniciar el script, coloque las siguientes líneas:

function SelectRecordProveedor(control) {

var ID = $(control).closest("tr").find("label").html();

$(".proveedorDropDownList").val(ID);

$("#dialog-search-proveedor").dialog("close");

}

1. Ya podemos probar la funcionalidad completa:

