

GUIA PRACTICA

Los objetivos de esta guía práctica son:

- a) Descargar a manera de reportes formatos Excel empleando la librería EppPlus
- b) Manera el cierre de sesión automática tras un tiempo de inacción de la aplicación

PREREQUISITO: Haber terminado la GuiaPractica_MapadelSitio, dado que en el presente documento se emplearán elementos que se acoplaron en dicha guía.

PARTE 1: DESCARGA DEL ARCHIVO EXCEL CON EL LISTADO DE PROVEEDORES

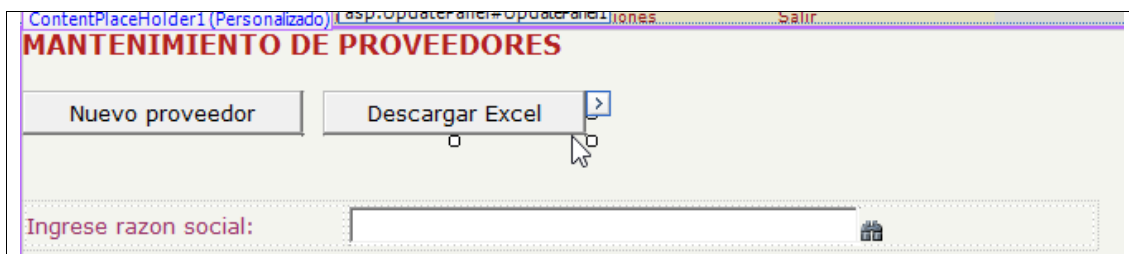
Paso 1:

Descargue la plantilla de la semana 15

Paso 2:

Abra su solución DemoVentas_3Capas. Desde el sitio SitioWEBVentas GUI agregue una referencia (así como lo hizo en su momento desde su aplicación Windows cuando se hizo la descarga de un reporte Excel) a la librería Eppplus.dll que está dentro de la carpeta de su solución.

Luego de ello diríjase al formulario “WebMantProveedor” hecho en la semana 13. Agregue el botón llamado “btnDescargar” donde su texto sea “Descargar Excel” tal como se aprecia en la siguiente imagen:



Paso 3:

En la carpeta “Mantenimientos” de su sitio Web copie el archivo “ListadoProveedores.xlsx” desde su plantilla de la semana 15. La idea es que haciendo clic en el botón “btnDescargar” se descargue el archivo Excel “ListadoProveedores.xlsx” con el listado de todos los proveedores. Para ello vaya a la ventana de código del formulario “WebMantProveedores.aspx” y en la parte superior agregue los 2 siguientes using (comentados como Final):

```
using System.Data;  
// FINAL: Agregue los siguientes using:  
using OfficeOpenXml;  
using System.IO;
```

Paso 4:

En el evento Load del mismo formulario agregue las líneas enmarcadas en rojo:

```
protected void Page_Load(object sender, EventArgs e)  
{  
    try  
    {  
        // Agregamos estas 2 líneas para permitir la descarga del archivo Excel en el boton btnDescargar  
        ScriptManager scriptManager = ScriptManager.GetCurrent(this.Page);  
        scriptManager.RegisterPostBackControl(this.btnDescargar);  
  
        if (Page.IsPostBack == false)  
        {  
            // Creamos la vista en memoria y cargamos los datos  
            dtv = new DataView(objProveedorBL.ListarProveedor());  
            Session["Vista"] = dtv;  
            CargarDatos("");  
        }  
    }  
}
```

Estas líneas se hacen para que el botón btnDescargar pueda, sin crear conflictos con el UpdatePanel, descargar el archivo Excel.

Paso 5:

En el evento clic del botón “btnDescargar” ingrese el siguiente código:

```
protected void btnDescargar_Click(object sender, EventArgs e)
{
    try
    {
        // Ruta del archivo plantilla del reporte
        String rutaarchivo = Server.MapPath("~/") + @"Mantenimientos\ListadoProveedores.xlsx";
        // Obtenemos los proveedores
        DataTable dtProveedores = new DataTable();
        dtProveedores = objProveedorBL.ListarProveedor();
        // fila de inicio del reporte
        Int16 fila1 = 5;
        using (var pck = new OfficeOpenXml.ExcelPackage(new FileInfo(rutaarchivo)))
        {
            //Nombre de archivo para descargar
            String filename = "ListadoProveedores-" + DateTime.Today.ToShortDateString();
            ExcelWorksheet ws = pck.Workbook.Worksheets["Hoja1"];

            //llenamos el Excel con los proveedores
            foreach (DataRow drProveedor in dtProveedores.Rows)
            {
                ws.Cells[fila1, 1].Value = drProveedor["Cod_prv"].ToString();
                ws.Cells[fila1, 2].Value = drProveedor["Raz_Soc_prv"].ToString();
                ws.Cells[fila1, 3].Value = drProveedor["Dir_prv"].ToString();
                ws.Cells[fila1, 4].Value = drProveedor["Tel_prv"].ToString();
                ws.Cells[fila1, 5].Value = drProveedor["Departamento"].ToString() + "-" + drProveedor["Provincia"].ToString() + "-" +
                    drProveedor["Distrito"].ToString();
                ws.Cells[fila1, 6].Value = drProveedor["Rep_ven"].ToString();
                fila1 += 1;
            }

            //modificando el ancho de las columnas
            ws.Column(1).Width = 30;
            ws.Column(2).Width = 50;
            ws.Column(3).Width = 50;
            ws.Column(4).Width = 40;
            ws.Column(5).Width = 45;
            ws.Column(6).Width = 45;

            //Escribir de nuevo al cliente y descargar el archivo desde el navegador
            Response.Clear();
            Response.ContentType = "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet";
            Response.AddHeader("content-disposition", "attachment; filename=" + filename + ".xlsx");
            using (var memoryStream = new MemoryStream())
            {
                pck.SaveAs(memoryStream);
                memoryStream.WriteTo(Response.OutputStream);
            }
            Response.End();
        }
    }
    catch (Exception ex)
    {
        lblMensaje.Text = ex.Message;
        mpeMensaje.Show();
    }
}
```

Paso 6:

Muestre el formulario “WebMantProveedor.aspx” desde su navegador y compruebe la descarga del archivo Excel con el listado de proveedores.

PARTE 2: MANEJO DE LA CADUCIDAD DE LA SESION

Paso 1:

Ubíquese en la vista de código HTML de su página maestra "DemoMaster". Dentro de la etiqueta

`<asp:ContentPlaceHolder id="ContentPlaceHolder1" runat="server">`

copie y pegue el código del archivo de texto llamado "ScriptCaducidad" entregado en la plantilla. El código en el formulario "DemoMaster" debería quedar así:

```

<asp:ContentPlaceHolder id="ContentPlaceHolder1" runat="server">
<!--AGREGAR FINAL: Se agrega este script java con el manejo de la expiracion de la sesion....-->
<!--Impide emplear el boton Regresar en el navegador-->
<script language="javascript" type="text/javascript">
    javascript:window.history.forward(1);
</script>-->
<script language="javascript" type="text/javascript">
    var sessionTimeoutWarning =
    "<%= System.Configuration.ConfigurationManager.AppSettings["AdvertenciaSession"].ToString() %>"
    var sessionTimeout = "<%= Session.Timeout %>";
    var timeOnPageLoad = new Date();

    //Para la advertencia ....
    setTimeout('SessionWarning()', parseInt(sessionTimeoutWarning) * 60 * 1000);
    //Para redireccionar a otra pagina
    setTimeout('RedirectToWelcomePage()', parseInt(sessionTimeout) * 60 * 1000);

    //Funcion de advertencia....
    function SessionWarning() {
        //Tiempo antes de expirar
        var minutesForExpiry = (parseInt(sessionTimeout) -
            parseInt(sessionTimeoutWarning));
        var message = "Tu sesion va a expirar en... " + minutesForExpiry +
            " minutos. Por favor graba o realiza alguna accion.";
        alert(message);
        var currentTime = new Date();
        //Tiempo de expiracion
        var timeForExpiry = timeOnPageLoad.setMinutes(timeOnPageLoad.getMinutes()
            + parseInt(sessionTimeout));

        //Si en caso la hora actual ha superado el tiempo de expiracion
        if (Date.parse(currentTime) > timeForExpiry) {
            alert("La sesion ha expirado. Seras redireccionado a la pagina de bienvenida");
            window.location = "../WebLogin.aspx";
        }
    }

    //Funcion para la redireccion a la pagina WebLoginText.aspx....
    function RedirectToWelcomePage() {
        alert("La sesion ha expirado. Seras redireccionado a la pagina de bienvenida");
        window.location = "../WebLogin.aspx";
    }
</script>
<!--FINAL : Fin del script de la validacion de la sesion.-->
</asp:ContentPlaceHolder>

```

Nota: A partir de aquí viene todo el código pre existente en el formulario, el cual no ha sido modificado.

Paso 2:

En el Web.config de su sitio WEB, adicionar las líneas enmarcadas en rojo:

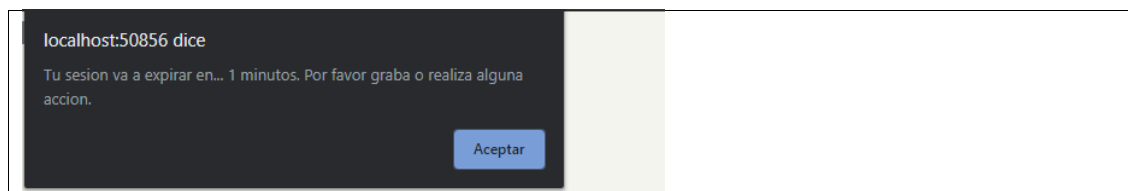
```
<?xml version="1.0"?>
<!-- Para obtener más información sobre cómo configurar la aplicación ASP.NET, visite
https://go.microsoft.com/fwlink/?LinkId=169433 -->
<configuration>
  <!-- FINAL: Se agrega un parametro en la configuracion que determina la cantidad de minutos
  de inactividad para que muestre la advertencia. En este caso sera solo un minuto -->
  <appSettings>
    <add key="AdvertenciaSession" value="1"/>
  </appSettings>

  <connectionStrings>
    <add name="Ventas" connectionString="server=LAPTOP-ORH1E3IB\JLEON;Database=VentasLeon;Integrated Security=true" providerName=
  </connectionStrings>
  <system.web>
    <siteMap>
      <providers>
        <remove name="MySqlSiteMapProvider"/>
      </providers>
    </siteMap>
    <!-- FINAL: Agregar para establecer el tiempo de inactividad de la sesion -->
    <sessionState timeout="2">
    </sessionState>
    <compilation debug="true" targetFramework="4.6.1">
    </compilation>
    <assemblies>
      <add assembly="System.Design, Version=4.0.0.0, Culture=neutral, PublicKeyToken=B03F5F7F11D50A3A"/>
      <add assembly="System.Web.Extensions.Design, Version=4.0.0.0, Culture=neutral, PublicKeyToken=31BF3856AD364E35"/>
    </assemblies>
    <httpRuntime targetFramework="4.6.1">
    </httpRuntime>
  </system.web>
```

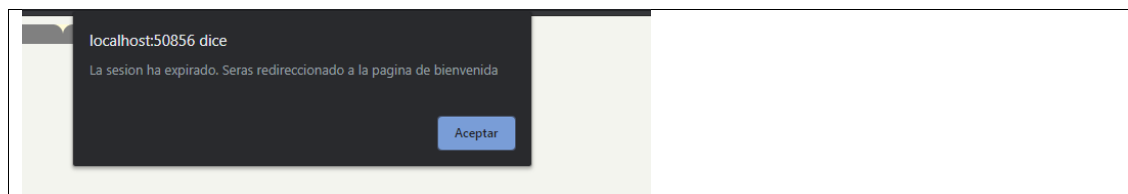
Esto se hace para manejar un parámetro de advertencia de inactividad de 1 minuto ("AdvertenciaSession") y lo segundo para indicar que el tiempo de inactividad (timeout) de la aplicación será de 2 minutos

Paso 3:

Muestre el formulario "WebMantProveedor.aspx" (o cualquier otro) en su navegador y deje de interactuar con la aplicación durante un minuto. Vera este mensaje

**Paso 4:**

Haga clic en "Aceptar" y siga sin interactuar con el aplicativo durante un minuto más y vera como sale este mensaje:



Haga clic en "Aceptar" y lo llevara al formulario "WebLogin.aspx" para que reinicie sesión. Incluso el botón retroceder del navegador queda sin efecto.