



Manual de Usuario

Navegador

Grupo No.3



El presente documento contiene una serie de pasos los cuales deben ser tomados en cuenta para la utilización de la dll del navegador. Anexo al mismo se les entregará una dll la cual podrá ser probada , y un proyecto de pruebas en el cual pueden verificar su funcionamiento.

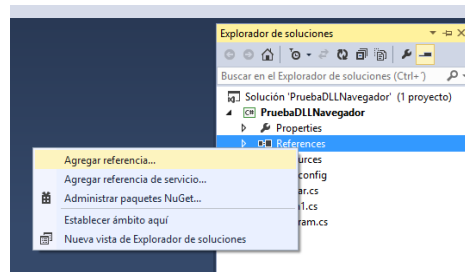


Contenido

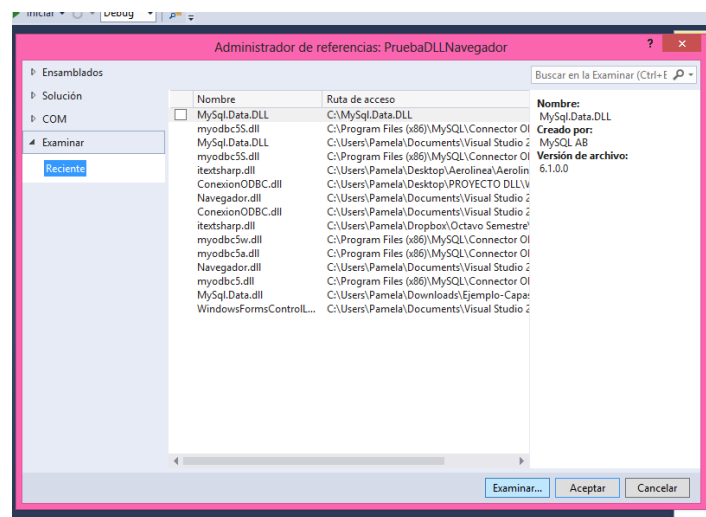
1. Deberá agregar dll del navegador y odbc al proyecto como referencia.....	3
2. Pasos a Seguir para programar funciones del navegador:.....	5
Nuevo	5
Editar	5
Eliminar	6
Guardar	6
Cancelar.....	7
Refrescar	8
Imprimir.....	8
Buscar	9
Flechas (Ir Primero, Anterior, Siguiente, Ir Último).....	11



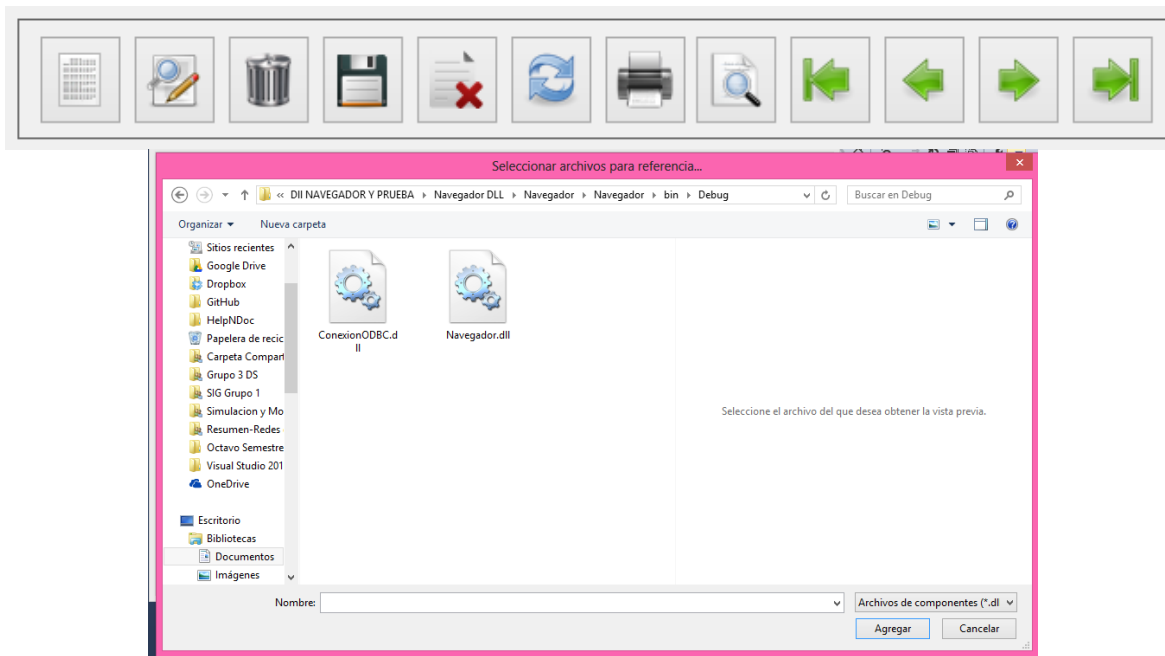
1. Deberá agregar dll del navegador y odbc al proyecto como referencia.



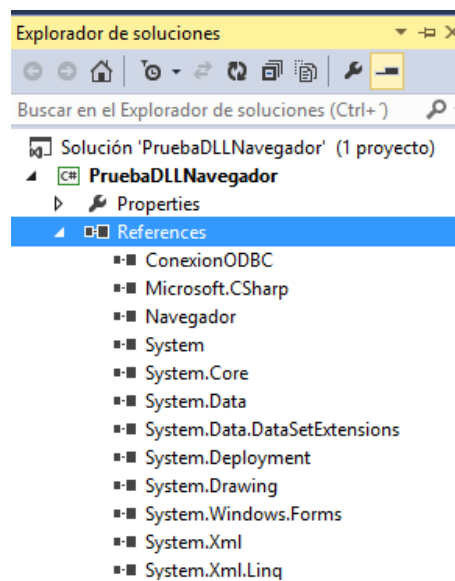
Aparecerá la siguiente pantalla y deberá dar clic en Examinar, y buscar la dll en la ubicación donde la posee, para ser agregada al proyecto.



Selecciona las dll a agregar y da clic en "ACEPTAR"



Notará que en las referencias le aparecerá las dll de conexión ODBC y la dll del navegador.



Deberá agregar los botones en el siguiente orden: tomando en cuenta el nombre de los mismos, el cual es el siguiente:



- btnnuevo
- btncancelar



- btneliminar
- btnguardar
- btncancelar
- btnrefrescar
- btnimprimir
- btnbuscar
- btnirPrimero
- btnanterior
- btnsiguiente
- btnirUltimo

2. Pasos a Seguir para programar funciones del navegador:

- Deberá agregar la siguiente librería en la parte superior del proyecto.
 - o `using Navegador;`

Nuevo

- Deberá dar doble clic en el botón de nuevo y agregar el siguiente código:

```
clasnegocio cnegocio = new clasnegocio();
```

- La presente función le permitirá activar los textbox contenidos en el formulario. enviando como parámetro (1,2)
 - o (1) nombre del textbox que desea activar.
 - o (2) deberá colocar "true" como parámetro lo cual le permitirá activar el textbox.

```
cnegocio.funactivarDesactivarTextbox(textBox1, true);  
cnegocio.funactivarDesactivarTextbox(textBox2, true);  
cnegocio.funactivarDesactivarTextbox(textBox3, true);
```

- La presente función le permitirá activar los combobox contenidos en el formulario. Enviando como parámetro (1,2)
 - o (1) nombre del combobox que desea activar.
 - o (2) deberá colocar "true" como parámetro lo cual le permitirá activar el combobox.

```
cnegocio.funactivarDesactivarCombobox(comboBox1, true);
```

Editar

Deberá dar clic en el botón de editar y colocar el siguiente código.

- Arreglo de textbox con tags agregados, los mismos utilizados por el insertar donde escribirá los datos que desea modificar.



```
TextBox[] arraytexto = { textBox1, textBox2, textBox3, textBox5 };
```

```
clasnegocio cn = new clasnegocio();
string tabla = "Cliente";
Boolean permiso = true;
string codigo = comboBox1.SelectedValue.ToString();
```

- En la siguiente función le permitirá enviar los siguientes parámetros:
 - o (1) Nombre de la **tabla** que desea editar
 - o (2) boolean de **permiso**, con valor true para que pueda editar.
 - o (3) **array** de texto con información de textbox agrupada.
 - o (4) **código** que desea editar (llave primaria), puede auxiliarse de un combobox con los códigos como value, y visualizando la descripción de la información a modificar.
 - o (5) Nombre de campo de llave primaria de la tabla.
- ```
cn.EditarObjetos("CLIENTE", permiso, arraytexto, codigo, "IdCliente");
// cn.EditarObjetos("CLIENTE", true, arr, "15", "IdCliente");
// , "IdCliente");
textBox1.Text = "";
textBox2.Text = "";
textBox3.Text = "";
textBox5.Text = "";
```

## Eliminar

- Deberá dar clic en el botón de eliminar y agregar el siguiente código:

```
/* Las consideraciones a tomar en cuenta son las siguientes:
 funeliminarRegistro(1,2,3,4);
 * 1) Nombre tabla a eliminar registro
 * 2) Código de llave primaria a eliminar
 * 3) Campo de llave primaria
 * 4) Nombre del campo estado de la tabla
*/
clasnegocio cnegocio = new clasnegocio();
string tabla, codEliminar, campoLlavePrimaria, campoEstado;
tabla = "cliente";
codEliminar = comboBox1.SelectedValue.ToString();
campoLlavePrimaria = "IdCliente";
campoEstado = "Estado";

cnegocio.funeliminarRegistro(tabla,codEliminar,campoLlavePrimaria,campoEstado);
```

## Guardar

- Deberá dar doble clic en el botón de guardar.
  - El código a agregar es el siguiente:
1. Se declara un array de textbox en el cual entre las llaves colocará el nombre de todos los textbox con datos que desea insertar.
    - a. `TextBox[] datos = { textBox1, textBox2, textBox3, textBox5};`
  2. Deberá declarar dos variables con las siguientes propiedades:
    - a. `string tabla = "Cliente";` //colocar nombre de la tabla donde desea insertar.
    - b. `Boolean permiso = true;` // permiso debe encontrarse en true, para poder insertar.



3. Instancia la clase negocio que se encuentra en la dll y llama a la función AsignarObjetos la cual realiza la función de insertar.

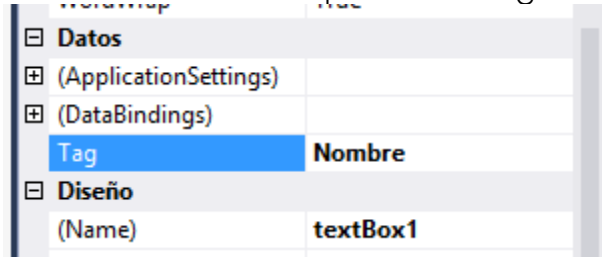
- a. `clasnegocio cn = new clasnegocio();`
- b. `cn.AsignarObjetos(tabla, permiso, datos);` // parámetros (1,2,3)
  - i. (1) string de tabla, donde colocó el nombre de la tabla donde se desea insertar.
  - ii. (2) boolean de permiso, el cual debe tener valor true para poder insertar.
  - iii. (3) array de textbox, declarado con nombre de textbox agregados.

4. En el diseño del sistema debe tomar en cuenta los siguientes pasos:

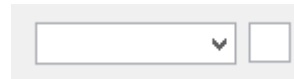
- a. Agregar textbox



- b. Ir a propiedades, buscar la propiedad TAG donde deberá colocar el nombre del campo en la base de datos al cual enviará el contenido del textbox que está configurando.



- c. Repetirá el proceso con cada uno de los textbox que utiliza. Nota: si su sistema utiliza combobox para selección de información, deberá colocar un textbox invisible a la par el cual recibirá la información del combobox y ese textbox enviarlo al array. Tomando en cuenta que por el tipo de array que debe manejar únicamente recibirá textbox.



5. Para limpiar los textbox:

```
textBox1.Text = "";
textBox2.Text = "";
textBox3.Text = "";
textBox5.Text = "";
```

## Cancelar

```
clasnegocio cnegocio = new clasnegocio();
```

- La presente función le permitirá desactivar los textbox contenidos en el formulario. enviando como parámetro (1,2)
  - o (1) nombre del textbox que desea desactivar.
  - o (2) deberá colocar "false" como parámetro lo cual le permitirá desactivar el textbox.

```
cnegocio.funactivarDesactivarTextbox(textBox1, false);
cnegocio.funactivarDesactivarTextbox(textBox2, false);
```



```
cnegocio.funactivarDesactivarTextbox(textBox3, false);
```

- La presente función le permitirá desactivar los combobox contenidos en el formulario. Enviando como parámetro (1,2)
  - (1) nombre del combobox que desea desactivar.
  - (2) deberá colocar "false" como parámetro lo cual le permitirá desactivar el combobox.

```
cnegocio.funactivarDesactivarCombobox(comboBox1, false);
```

## Refrescar

Para refrescar datos en combobox y gridview deberá tomar en cuenta el siguiente código a agregar:

- Dar doble clic en botón de refrescar y colocar el siguiente código:

```
clasnegocio cnegocio = new clasnegocio();
/*
 * Función: funconsultarRegistrosCombo
 Esta funcion envía como parámetros (1,2,3,4) :
 * 1) Nombre Campo Value combobox
 * 2) Query select a la BD
 * 3) Nombre Campo DisplayMember combobox
 * 4) Nombre del combobox
 */
cnegocio.funconsultarRegistrosCombo("IdCliente", "SELECT IdCliente,nombre from
cliente WHERE ESTADO='ACTIVO'", "nombre", comboBox1);

/*
 * Función: funconsultarRegistros
 Esta funcion envía como parámetros (1,2,3,4) :
 * 1) Nombre Tabla
 * 2) Query select a la BD
 * 3) Procedimiento que realiza (consulta)
 * 4) Nombre del datagridview
 */
cnegocio.funconsultarRegistros("clientes", "SELECT nombre,apellido,direccion from
cliente WHERE ESTADO='ACTIVO'", "consulta", dataGridView1);
```

- Nota: podrá colocar estas mismas funciones en el load del formulario si desea que se llenen en el inicio, ya que las presentes funciones , consultan en base de datos y despliegan según sea la necesidad la información requerida en combobox o en gridview.

## Imprimir

- Deberá agregar al form, un textbox adicional el cual servirá de auxiliar para visualizar la ruta donde se encuentra el archivo. En este caso utilizamos el "textBox4"





- Deberá dar doble clic en el botón de imprimir y agregar las siguientes líneas:

```
clasnegocio cnegocio = new clasnegocio();
cnegocio.funimprimir(textBox4); // envía como parámetro a la función de
imprimir contenida en la dll,el nombre del textbox (textBox4).
```

- Al dar clic en el presente botón le aparecerá una ventana para seleccionar el archivo que desea imprimir, el cual se le ha autorizado según el usuario y rol que posee el mismo. Deberá repetir el proceso dos veces y luego le imprimirá el documento.

## Buscar

- Para este proceso primeramente deberá agregar el formulario llamado buscar.cs el cual se le proporciona en la carpeta Adjunta. Y deberá instanciarlo con el siguiente código.

```
buscar busc = new buscar();
busc.Show();
```

Nota: no borrar ningún elemento ni código en el presente formulario.



- Deberá dar doble clic en el botón que dice buscar y podrá visualizar el siguiente código:

```
//---- DEBE ENVIAR LOS SIGUIENTES PARAMETROS (1,2,3)
/*
 * (1) tabla
 * (2) base de datos
 * (3) campos que deseo visualizar
 * sfuncrearBusqueda("cliente", "EjemploCapas",
"idcliente,nombre,apellido,direccion");
 */
```

```
sfuncrearBusqueda("cliente", "EjemploCapas", "idcliente,nombre,apellido,direccion");
```

- La función anteriormente descrita instancia a una función programada en la parte superior del presente formulario. La cual le muestra las condiciones de funcionamiento de los radio button agregados en el mismo.:

```
string condicion;
string squerybuscar;
String ssigno;
public void sfuncrearBusqueda(String tabla,String basedatos,String
camposvisualizar) {

 clasnegocio cnegocio = new clasnegocio();
 if (opcion == 1) { ssigno = ">"; condicion = comboBox2.Text + ssigno
+ "" + textBox1.Text + ""; }
 if (opcion == 2) { ssigno = "<"; condicion = comboBox2.Text + ssigno
+ "" + textBox1.Text + ""; }
 if (opcion == 3) { ssigno = "="; condicion = comboBox2.Text + ssigno
+ "" + textBox1.Text + ""; }
 if (opcion == 4) { condicion = comboBox2.Text + " BETWEEN " + ""
+textBox1.Text + "" + " AND " + "" + textBox2.Text + "";}

 squerybuscar = "SELECT "+camposvisualizar+" FROM " + tabla + " WHERE
" + condicion + "";
 //MessageBox.Show(squerybuscar);
 cnegocio.funconsultarRegistros(tabla,squerybuscar, "consulta",
griddatosBusqueda);
 // cnegocio.funconsultarRegistros("clientes", "SELECT * from
cliente", "consulta", griddatosBusqueda);
 limpiar();

}
private void limpiar() {
 textBox1.Text = "";
 textBox2.Text = "";
}

public int opcion;
private void radioButton1_CheckedChanged(object sender, EventArgs e)
{
 textBox1.Visible = true;
 textBox2.Visible = false;
 opcion = 1;
}
```



```

}

private void radioButton3_CheckedChanged(object sender, EventArgs e)
{
 textBox1.Visible = true;
 textBox2.Visible = false;
 opcion = 3;
}

private void radioButton2_CheckedChanged(object sender, EventArgs e)
{
 textBox1.Visible = true;
 textBox2.Visible = false;
 opcion = 2;
}

private void radioButton4_CheckedChanged(object sender, EventArgs e)
{
 textBox1.Visible = true;
 textBox2.Visible = true;
 opcion = 4;
}

```

- Deberá buscar la siguiente función y modificar los siguientes parámetros:

```

private void frmbuscar_Load(object sender, EventArgs e)
{
 string tabla = "cliente"; //agregar nombre de la tabla a la cual
 desea buscarle los datos.
 string basedatos = "colchoneria"; // deberá agregar el nombre de la
 base de datos donde se encuentra realizando las búsquedas

// PARÁMETROS : (1)(2)(3)
/*
 * (1) NOMBRE DE LA TABLA COLOCADO EN STRING tabla
 * (2) NOMBRE DE LA BASE DE DATOS COLOCADO EN STRING basedatos
 * (3) NOMBRE DEL COMBOBOX QUE SE ESTÁ UTILIZANDO.
 */
 funllenarCombo(tabla,basedatos, comboBox2);
}

```

## Flechas (Ir Primero, Anterior, Siguiente, Ir Último)

- Para la programación de las presentes flechas únicamente deberá tomar en cuenta el siguiente código:

```

private void btnirPrimero_Click(object sender, EventArgs e)
{
 clasnegocio cnegocio = new clasnegocio();
 cnegocio.funPrimero(dataGridView1);
// parámetro que envía es nombre del GridView el cual desea recorrer (dataGridView1)
// en este caso.
}

```



```
}

private void btnanterior_Click(object sender, EventArgs e)
{
 clasnegocio cnegocio = new clasnegocio();
 cnegocio.funAnterior(dataGridView1);
 // parámetro que envía es nombre del GridView el cual desea recorrer (dataGridView1)
 // en este caso.
}

private void btnsiguiente_Click(object sender, EventArgs e)
{
 clasnegocio cnegocio = new clasnegocio();
 cnegocio.funSiguiente(dataGridView1);
 // parámetro que envía es nombre del GridView el cual desea recorrer (dataGridView1)
 // en este caso.
}

private void btnirUltimo_Click(object sender, EventArgs e)
{
 clasnegocio cnegocio = new clasnegocio();
 cnegocio.funUltimo(dataGridView1);
 // parámetro que envía es nombre del GridView el cual desea recorrer (dataGridView1)
 // en este caso.
}

- }
```