

UNIVERSIDAD MARIANO GALVEZ DE GUATEMALA

INGENIERÍA EN SISTEMAS DE INFORMACIÓN Y CIENCIAS DE LA
COMPUTACIÓN

INGENIERIA DEL SOFTWARE

SECCIÓN A CAMPUS, MATUTINO

MANUAL DE USUARIO NAVEGADOR

Cristian Estrada 0901-13-1085

Otto Hernández 0901-13-290

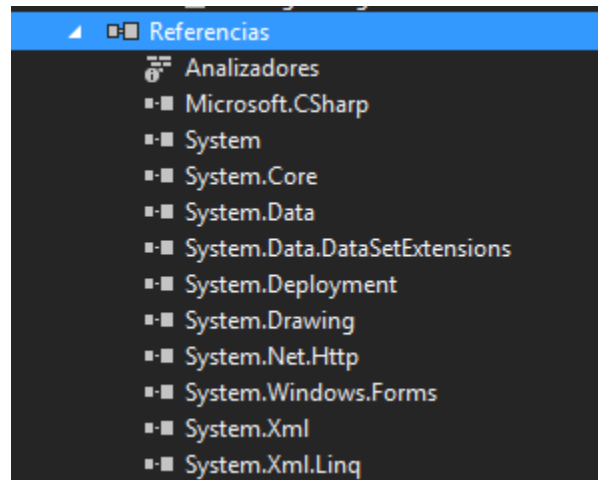
Javier Figueroa 0901-13-2923

Índice

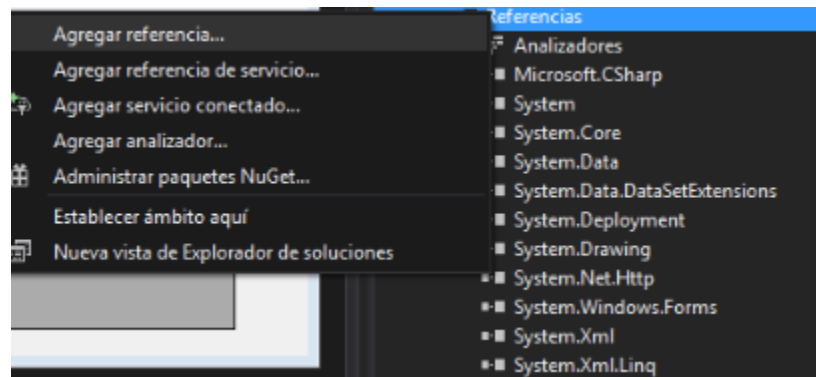
MANUAL DE USUARIO.....	3
Guardar	5
Editar	6
Eliminar	7
Nuevo	7
Cancelar.....	8
Navegación.....	9

MANUAL DE USUARIO

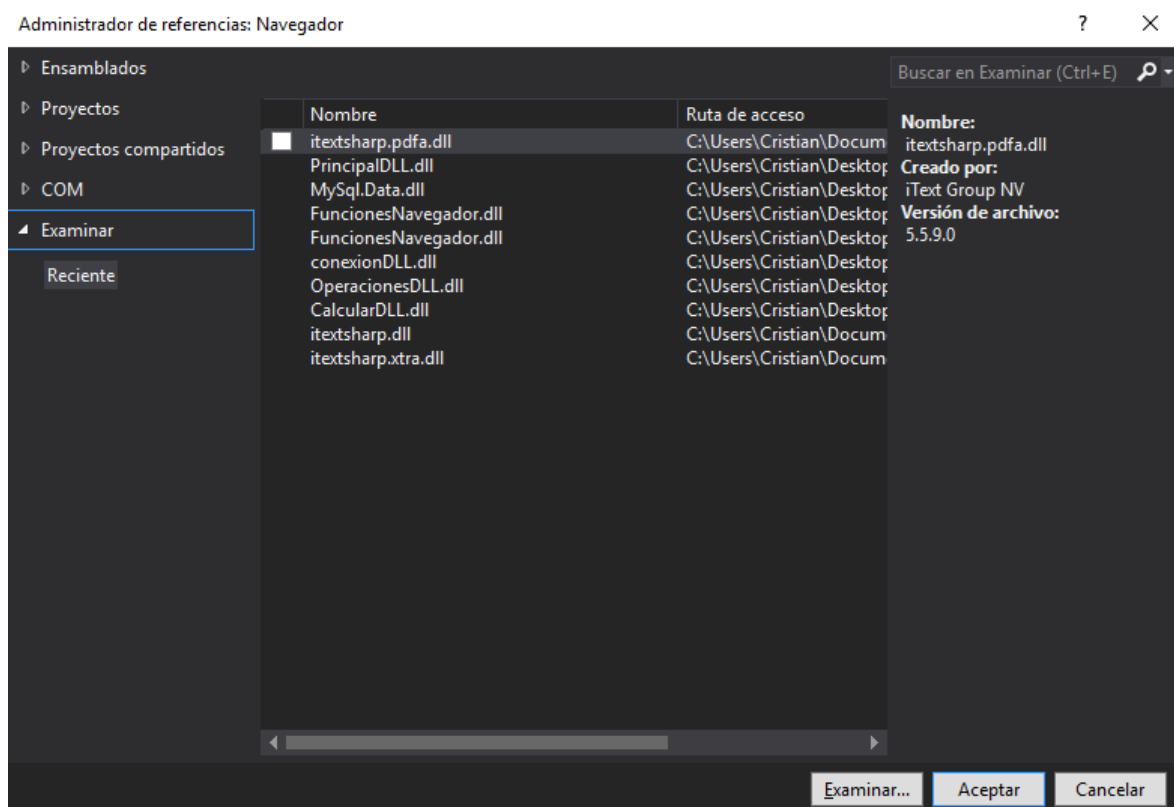
Para agregar el DLL para las funciones del navegador



Se tendrá que ir a la opción de Referencias en el Explorador de Soluciones



Al darle clic derecho en Referencias saldrán las opciones en la cual se tendrá que darle clic a la opción Agregar referencia.

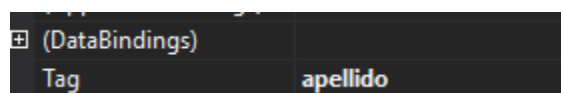


Para agregar la DLL se tendrá que ir al botón de Examinar, en la cual tendrá que ir a buscar donde está el DLL y seleccionar y agregarlo y así mismo Aceptar los cambios.

TAG de los componentes

Para agregar los Tags de los diferentes componentes como textox, comboBox, etc.

Se tendrá que ir a las propiedades del componente y buscar la siguiente sección:



En la cual se tendrá que ingresar el atributo de la tabla de la base de datos que se va ingresar en la cual va servir para el ingreso de los datos y reconocimiento de los atributos de cada tabla al momento de ingresar, modificar, etc.



Deberá agregar los botones teniendo en cuenta el siguiente orden y así mismo la estandarización de los botones:

- btn_nuevo
- btn_guardar
- btn_editar
- btn_eliminar
- btn_buscar
- btn_cancelar
- btn_actualizar
- btn_anterior
- btn_primer
- btn_siguiente
- btn_ultimo

Hacer referencia a la librería para utilizar las funciones de la DLL

- Agregar la siguiente librería
 - `using` FuncionesNavegador;

Guardar

El siguiente código es para el botón guardar

```

Línea 1 textBox4.Text = comboBox1.SelectedValue.ToString();
Línea 2 textBox5.Text = dateTimePicker1.Value.ToString("yyyy-MM-dd");
Línea 3 CapaNegocio fn = new CapaNegocio();
Línea 4 TextBox[] textbox = { textBox2, textBox3, textBox4, textBox5 };
Línea 5 DataTable datos = fn.construirDataTable(textbox);
  
```

Explicación:

- En la línea 1 se muestra como debe ser enviado el valor de un combobox, se debe colocar el texto del combobox dentro de un textbox no visible que se encuentra en el form y luego el textbox es el que debe ser incluido en el arreglo de textbox a ser enviado.
- En la línea 2 se muestra como debe ser enviado el valor de un datetimepicker, se debe convertir el valor seleccionado en este y colocarlo en un textbox no visible que se encuentra en el form y luego el textbox en el que se coloca el valor del datetimepicker debe ser incluido en el arreglo de textbox a ser enviado.
- Para la línea 3 lo que se tiene que realizar es crear un objeto de una de las clases del DLL brindado por el grupo 1.

- En la línea 4 se debe crear un arreglo con los textbox que se enviarán para realizar el query de inserción.
- En la línea 5 se hace referencia a una de las funciones de la clase de la cual se creó el objeto, se debe enviar el arreglo de textbox creado para realizar la inserción en la tabla correspondiente

```
if (datos.Rows.Count==0)
{
    MessageBox.Show("Hay campos vacios", "Favor Verificar",MessageBoxButtons.OK,
    MessageBoxIcon.Error);
}
```

En este código realiza una verificación para saber si el datatable recibido está vacío, si está vacío significa que no todos los textbox contenían texto por lo que se envía un mensaje pidiendo que se ingresen todos los datos.

```
string tabla = "empleado";
```

La variable tabla, dentro de “” es donde debe colocar el nombre de la tabla a la que se irán a ingresar los datos

```
if (Editar)
{
    fn.modificar(datos, tabla, atributo, Codigo);
}
else
{
    fn.insertar(datos, tabla);
}
```

Este If es para el manejo tanto del INSERT como del UPDATE, si la variable booleana Editar es = true entonces se realizará la función de editar mientras que si está en false entonces se realizará la función de insertar

Editar

Deberá agregar el siguiente código en el botón de Editar:

Agregar las variables globales

```
String Codigo;
Boolean Editar;
```

El siguiente código es del botón que hay que agregar

```
Linea 1 Editar = true;
Linea 2 atributo = "codigo";
Linea 3 Codigo = this.dataGridView1.CurrentRow.Cells[0].Value.ToString();
Linea 4 TextBox[] textbox = { textBox4, textBox3, textBox2, textBox5, textbox6 };
Linea 5 CapaNegocio fn = new CapaNegocio();
Linea 6 fn.llenartextbox(textbox, dataGridView1);
```

- En la línea 1 la variable Boolean nos va a servir en la función del void de insertar del guardar para identificar si es editar en el momento que sea = TRUE
- En la línea 2 el atributo es un parámetro que nos servirá cual es atributo de la tabla que van a manejar para actualizar.
- En la línea 3 se debe colocar el Código que seleccionado por la celda del identificador que van a cambiar en este caso si es un id, la variable va a tomar el dato que este en el datagrid del registro a modificar.
- En las líneas 4,5 y 6 se arma un arreglo de los textbox que se quiere que tomen el valor de ese registro para colocar cada campo del registro seleccionado del datagridview en los textbox que corresponden.

Eliminar

El siguiente código se agrega al botón de eliminar donde se realizara la inactividad de un registro en el momento de eliminar

```
private void btn_eliminar_Click(object sender, EventArgs e)
{
    String codigo2 = this.dataGridView1.CurrentRow.Cells[0].Value.ToString();
    String atributo2 = "codigo";
```

- La variable Codigo2 es para almacenar el dato a eliminar.
- La variable atributo2 es el atributo de la tabla para ir a buscar en la tabla para ser eliminar.

```
var resultado = MessageBox.Show("DESEA BORRAR EL REGISTRO SELECCIONADO",
"CONFIRME SU ACCION", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
```

Esta variable manda un mensaje para ver si se desea o no eliminar el registro.

```
if (resultado == DialogResult.Yes)
{
    CapaNegocio fn = new CapaNegocio();
    string tabla = "empleado";
    fn.eliminar(tabla, atributo2, codigo2);
}
```

Se agrega para llamar la función de eliminar de DLL y la variable tabla se almacena y se envía con el nombre de la tabla que se van hacer los cambios.

Nuevo

Se debe seleccionar el botón de Nuevo y darle doble clic para editar las funciones y agregar el siguiente código dependiendo de los controles que se deseen habilitar:

```
CapaNegocio fn = new CapaNegocio();
```

- Esta función permitirá llamar a los procedimientos de desactivación de controles en los formularios, enviando como parámetros las siguientes variables:
 - Nombre del control que desea habilitar/inhabilitar

- Colocar “true” si desea activar un control o colocar “false” si lo desea inhabilitar al presionar el botón Nuevo

Por ejemplo:

```
CapaNegocio fn = new CapaNegocio();
fn.ControlTextbox(textBox1, true);
```

- Funciones permitidas:
 - fn.ControlTextbox()
 - fn.ControlButton()
 - fn.ControlCheckBox()
 - fn.ControlCheckedListBox()
 - fn.ControlComboBox()
 - fn.ControlDateTimePicker()
 - fn.ControlListBox()
 - fn.ControlListView()
 - fn.ControlNumericUpDown()
 - fn.ControlPictureBox()
 - fn.ControlRadioButton()
 - fn.ControlDataGridView()

Cancelar

Se debe seleccionar el botón de Cancelar y darle doble clic para editar las funciones y agregar el siguiente código dependiendo de los controles que se deseen inhabilitar:

```
CapaNegocio fn = new CapaNegocio();
```

- Esta función permitirá llamar a los procedimientos de desactivación de controles en los formularios, enviando como parámetros las siguientes variables:
 - Nombre del control que desea habilitar/inhabilitar
 - Colocar “true” si desea activar un control o colocar “false” si lo desea inhabilitar al presionar el botón Nuevo

Por ejemplo:

```
CapaNegocio fn = new CapaNegocio();
fn.ControlTextbox(textBox1, false);
```

- Funciones permitidas:
 - fn.ControlTextbox()
 - fn.ControlButton()
 - fn.ControlCheckBox()
 - fn.ControlCheckedListBox()
 - fn.ControlComboBox()
 - fn.ControlDateTimePicker()
 - fn.ControlListBox()
 - fn.ControlListView()
 - fn.ControlNumericUpDown()
 - fn.ControlPictureBox()
 - fn.ControlRadioButton()
 - fn.ControlDataGridView()

Navegación

El navegador de registros consta en cuatro botones los cuales nos ayudaran a movernos por los registros ingresados.

Nota Importante

Lo importante que se debe tener en cuenta es que los alias de los encabezados de las columnas de los datagridviews no deben cambiarse pues según la función realizada para la navegación se comparara el tag de los textbox con los nombres de las columnas de los datagridview para colocar los datos en sus respectivos espacios.



Boton primero: Este boton tiene la funcion de colocarnos en el primer registro del datagridview y al mismo tiempo colocar todos los datos del registro en los textbox correspondientes.

En el siguiente codigo se describe los parametros a enviar a la funcion para que nos devuelva todos los datos disponibles en la tabla.

```
Linea 1 CapaNegocio fn = new CapaNegocio();  
Linea 2 fn.PrimerO(dataGridView1);  
Linea 3 TextBox[] textbox = { textBox4, textBox3, textBox2, textBox5, textbox6};  
Linea 4 fn.llenartextbox(textbox, dataGridView1);  
Linea 5 dateTimePicker1.Text = textBox5.Text;  
Linea 6 comboBox1.Text = textBox4.Text;
```

- En la linea 1 se crea el objeto para poder utilizar las funciones de la dll del navegador.
- En la linea 2 se envia el datagrid a la funcion primero para que nos coloque en el primer registro.
- En la linea 3 se crea un arreglo de textbox donde se colocaran los datos del primer registro.
- En la linea 4 se envia el arreglo de textbox y el datagrid para que este realice su respectiva funcion para colocar los datos del primer registro en los textbox correspondientes.
- En la linea 5 se muestra un caso especial en el cual el texto del datetimepicker contendra lo que este en el textbox invisible donde se coloco la fecha del primer registro.
- En la linea 6 se muestra un caso especial en el cual el texto del combobox contendra lo que este en el textbox invisible donde se coloco el dato correspondiente al primer registro.

Boton ultimo: Este boton tiene la funcion de colocarnos en el ultimo registro del datagridview y al mismo tiempo colocar todos los datos del registro en los textbox correspondientes.

En el siguiente codigo se describe los parametros a enviar a la funcion para que nos devuelva todos los datos disponibles en la tabla.

```
Linea 1 CapaNegocio fn = new CapaNegocio();
Linea 2 fn.Ultimo(dataGridView1);
Linea 3 TextBox[] textbox = { textBox4, textBox3, textBox2, textBox5, textbox6};
Linea 4 fn.llenartextbox(textbox, dataGridView1);
Linea 5 dateTimePicker1.Text = textBox5.Text;
Linea 6 comboBox1.Text = textBox4.Text;
```

- En la linea 1 se crea el objeto para poder utilizar las funciones de la dll del navegador.
- En la linea 2 se envia el datagrid a la funcion ultimo para que nos coloque en el ultimo registro.
- En la linea 3 se crea un arreglo de textbox donde se colocaran los datos del ultimo registro.
- En la linea 4 se envia el arreglo de textbox y el datagrid para que este realice su respectiva funcion para colocar los datos del ultimo registro en los textbox correspondientes.
- En la linea 5 se muestra un caso especial en el cual el texto del dateTimePicker contendra lo que este en el textbox invisible donde se coloco la fecha del ultimo registro.
- En la linea 6 se muestra un caso especial en el cual el texto del comboBox contendra lo que este en el textbox invisible donde se coloco el dato correspondiente al ultimo registro.

Botones de anterior: Este boton tiene la funcion de colocarnos en el registro anterior del datagridview y al mismo tiempo colocar todos los datos del registro en los textbox correspondientes.

En el siguiente codigo se describe los parametros a enviar a la funcion para que nos devuelva todos los datos disponibles en la tabla.

```
Linea 1 CapaNegocio fn = new CapaNegocio();
Linea 2 fn.Anterior(dataGridView1);
Linea 3 TextBox[] textbox = { textBox4, textBox3, textBox2, textBox5, textbox6};
Linea 4 fn.llenartextbox(textbox, dataGridView1);
Linea 5 dateTimePicker1.Text = textBox5.Text;
Linea 6 comboBox1.Text = textBox4.Text;
```

- En la linea 1 se crea el objeto para poder utilizar las funciones de la dll del navegador.
- En la linea 2 se envia el datagrid a la funcion anterior para que nos coloque en el registro anterior.
- En la linea 3 se crea un arreglo de textbox donde se colocaran los datos del registro anterior.
- En la linea 4 se envia el arreglo de textbox y el datagrid para que este realice su respectiva funcion para colocar los datos del registro anterior en los textbox correspondientes.

- En la línea 5 se muestra un caso especial en el cual el texto del datetimepicker contendrá lo que este en el textbox invisible donde se colocó la fecha del registro anterior.
- En la línea 6 se muestra un caso especial en el cual el texto del combobox contendrá lo que este en el textbox invisible donde se colocó el dato correspondiente al registro anterior.

Botón siguiente registro: Este botón tiene la función de colocarnos en el registro siguiente del datagridview y al mismo tiempo colocar todos los datos del registro en los textbox correspondientes.

En el siguiente código se describe los parámetros a enviar a la función para que nos devuelva todos los datos disponibles en la tabla.

```

Línea 1 CapaNegocio fn = new CapaNegocio();
Línea 2 fn.Siguiente(dataGridView1);
Línea 3 TextBox[] textbox = { textBox4, textBox3, textBox2, textBox5, textbox6};
Línea 4 fn.llenartextbox(textbox, dataGridView1);
Línea 5 dateTimePicker1.Text = textBox5.Text;
Línea 6 comboBox1.Text = textBox4.Text;

```

- En la línea 1 se crea el objeto para poder utilizar las funciones de la dll del navegador.
- En la línea 2 se envía el datagrid a la función anterior para que nos coloque en el siguiente registro.
- En la línea 3 se crea un arreglo de textbox donde se colocarán los datos del siguiente registro.
- En la línea 4 se envía el arreglo de textbox y el datagrid para que este realice su respectiva función para colocar los datos del siguiente registro en los textbox correspondientes.
- En la línea 5 se muestra un caso especial en el cual el texto del datetimepicker contendrá lo que este en el textbox invisible donde se colocó la fecha del siguiente registro.
- En la línea 6 se muestra un caso especial en el cual el texto del combobox contendrá lo que este en el textbox invisible donde se colocó el dato correspondiente al siguiente registro.