



UNIVERSIDAD MARIANO GALVEZ DE GUATEMALA
Diseño de Sistemas
Auditoria de Sistemas
Ing. Eduardo del Águila



EBRI CENTER
Encryption, BackUp and Recovery Information

Manual de Usuario Modulo de Seguridad

Nombre: Jorge Leonel Lam Pazos
Carné: 0901-11-6753

Guatemala, Octubre de 2014



Contenido

MANUAL DE USUARIO DE SEGURIDAD	2
FUNCIONES Y FORMULARIOS.....	2
INICIO DE SESION.....	2
RECUPERACION DE CONTRASEÑA.....	2
CUENTA PERSONAL	3
FUNCIONES.....	4
VARIABLES ENCAPSULADAS UTILIZABLES	5
REFERENCIA	6
EJEMPLOS DE USO DE LAS FUNCIONES.....	10



MANUAL DE USUARIO DE SEGURIDAD

FUNCIONES Y FORMULARIOS

La librería de seguridad contiene una serie de funciones y formularios que se encargaran de verificar los credenciales y actividades de los usuarios, además de agregar otros controles.

INICIO DE SESION

Este formulario esta contenido dentro de la librería dll de seguridad como “wfInicioSesion” para poder usarlo se encuentra en la dll con la referencia “Modulo de Seguridad_3._0.Presentacion”.

RECUPERACION DE CONTRASEÑA

Cuando la contraseña se ha olvidado, se puede usar acceder a este form sin necesidad de referencias en el Inicio de Sesión, si contesta correctamente se enviara a un correo registrado en la devolución de la contraseña.



CUENTA PERSONAL

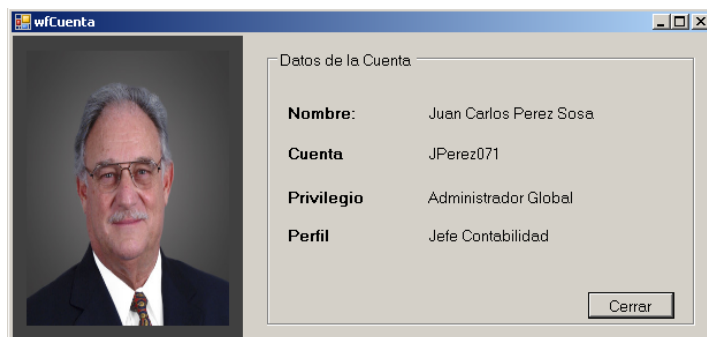
Este formulario muestra los datos personales de la cuenta del usuario para poder utilizarlo se tiene que hacer una referencia en donde usar con las siguientes instrucciones:

- 1) Agregar en los “using” :

```
using Modulo_de_Seguridad_3._0.Negocio;  
using Modulo_de_Seguridad_3._0.Presentacion;
```

- 2) En donde se desee usar agregar el form, ya sea un boton o Menu poner las instrucciones :

```
N_ControlesUsuario Controles = new N_ControlesUsuario();  
Controles.vIniciar();  
new wfCuenta().Show();
```





FUNCIONES

El programa posee un rango de funciones para el programa los cuales permiten diferentes operaciones, los cuales son:

- `vIniciar();`
Permite cargar los controles necesarios para el Formulario de Cuenta de Usuario.
- `vCerrarSesion();`
Permite cerrar la sesión y regresar nuevamente al Formulario de Inicio de Sesión.
- `vSalirSesion();`
Permite cerrar la sesión y salir de todo el programa.
- `bmpRevisarConexion();`
(Opcional) Hace una revisión de la conexión con la base de datos. Devuelve un valor como imagen el cual puede ser verde (Sincronizado) o rojo (Desconectado). Usar en conjunto con un objeto que utilice una imagen, además de usar un objeto "Timer" para hacer los periodos de revisión.

** Para acceder al rango de funciones es necesario agregar la instrucción en los using de:

```
using Modulo_de_Seguridad_3._0.Negocio;
```

Además de agregar la instrucción como variable global, si se desea de:

```
private N_ControlesUsuario Controles = new N_ControlesUsuario();
```



VARIABLES ENCAPSULADAS UTILIZABLES

Variables utilizables las cuales almacenan información estática que pueden ser utilizadas por otros módulos.

- alAplicaciones

Almacena un arraylist el cual contiene tres partes importantes:

- 1) Número de una aplicación, con formato "int".
- 2) Nombre de la aplicación, con formato "string".
- 3) Privilegios de la aplicación, con formato "Dictionary".

Para la lectura de estos se puede hacer de la siguiente manera:

```
String derechos = string.Empty;
ArrayList alAplicaciones = new E_DatosPublicos().alAplicaciones;

foreach (E_Aplicaciones Dato in alAplicaciones)
{
    int codigoApp = Dato.CodigoAplicacion;
    string nombreApp = Dato.NombreAplicacion;
    Dictionary <string,int> DerechosApp = Dato.PrivilegiosAplicacion;
    foreach (KeyValuePair<string, int> Valor in DerechosApp)
    {
        derechos += Valor.Key + ": " + Valor.Value + Environment.NewLine;
    }
    MessageBox.Show(codigoApp + ", " + nombreApp + Environment.NewLine +
    derechos);
    derechos = string.Empty;
}
```

- iCodigoUsuario

Almacena el código del Usuario que ha registrado el usuario.

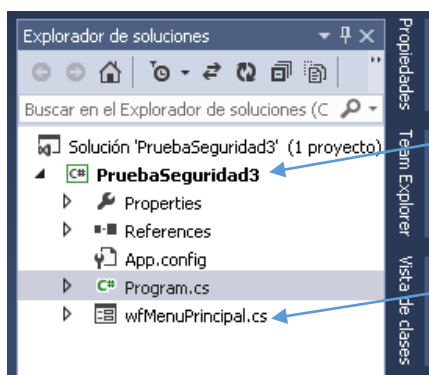
****Para poder acceder a estas variables es necesario agregar a los "using" la instrucción:**

```
using Modulo_de_Seguridad_3._0.Entidades;
```



REFERENCIA

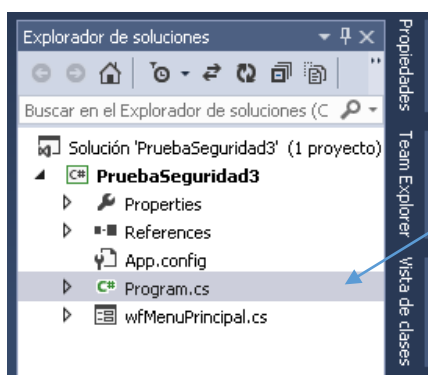
Antes de comenzar se necesita crear un proyecto nuevo de tipo Windows Form Application, para comenzar el usuario tendría que crear un nuevo Windows Form y el cual deberá estar como MDI, este Windows Form nos servirá como nuestro Menu Principal para cargar



Nuevo Proyecto Windows Form Application (Ej. "PruebaSeguridad3")

Nuevo Windows Form, este será nuestro Menu Principal (Ej. "wfMenuPrincipal")

Para poder comenzar con nuestro Formulario de Inicio de Sesión, en nuestro proyecto necesitamos entrar en "Program.cs" (Esa clase contiene las instrucciones de inicio del proyecto, está la genera el proyecto que nosotros creamos).

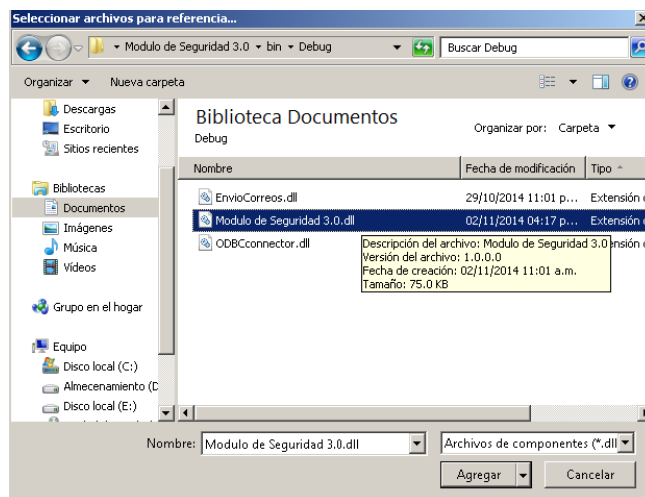
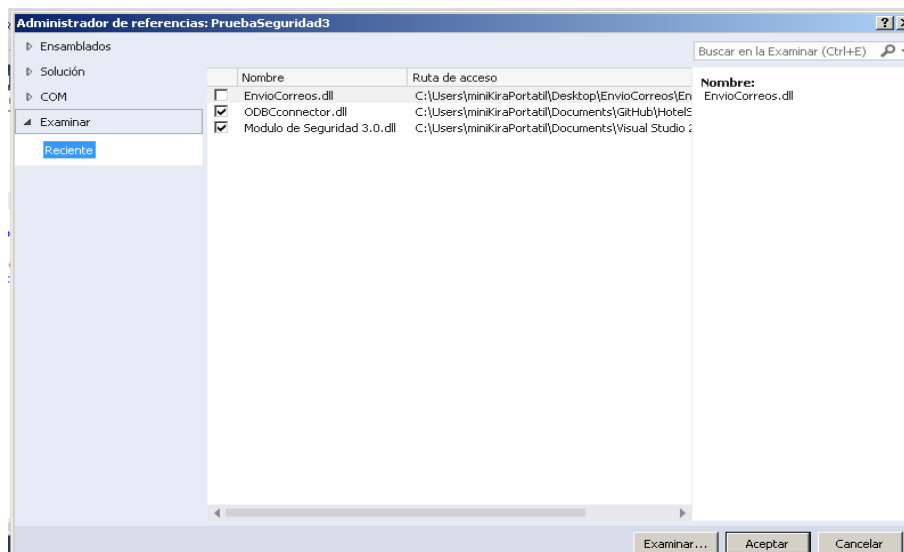


Clase Program, contenida dentro de nuestro proyecto de Ejemplo: PruebaSeguridad3



Para comenzar a programar y agregar nuestro inicio de sesión, necesitamos agregar la librería a nuestro proyecto nuevo.

- 1) En referencias, damos click derecho y buscamos la opción “Agregar referencia...”.
- 2) Luego agregamos nuestra dll “Modulo de Seguridad 3.0” por medio del botón examinar y luego localizamos nuestra librería, y le damos en aceptar.



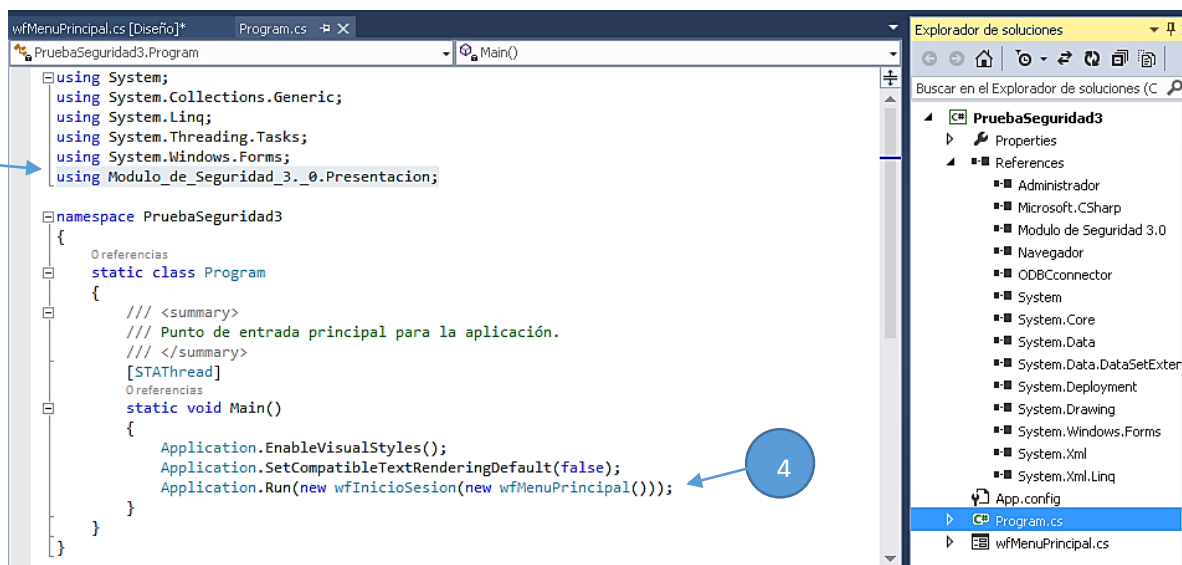
- 3) Una vez agregada la librería de “Modulo de Seguridad 3.0” (Para estar seguro de la referencia, verificar su explorador de soluciones en “Reference”, si existe el nombre) necesitamos agregar la referencia en la parte de programación de la clase “Program”. En los using para poder usar el Inicios de Sesión, se debe agregar la Instrucción:

```
using Modulo_de_Seguridad_3._0.Presentacion;
```

Una vez agregada la referencia se podrán acceder a todos lo forms disponibles de la librería.

- 4) Para poder usar el form de Inicio de Sesión, en la instrucción “Application.Run()”, se agrega entre los paréntesis “new wfInicioSesion()”, además de colocar dentro de la instrucción de “new wfInicioSesion()” se agrega el formulario que será nuestro Menu Principal, Ejemplo “new wfMenuPrincipal()”. La instrucción debería quedar de la siguiente manera:

```
Application.Run(new wfInicioSesion(new wfMenuPrincipal()));
```





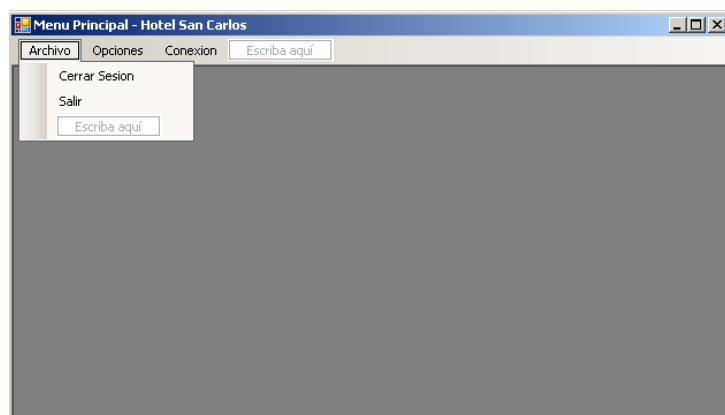
- 5) Ahora podremos probar si nuestro Inicio de Sesión funciona de manera correcta



**** Nota cada grupo deberá construir su Menú Strip y su MDI según los estándares , el usuario será responsable de como use los forms, funciones y variables.**



EJEMPLOS DE USO DE LAS FUNCIONES



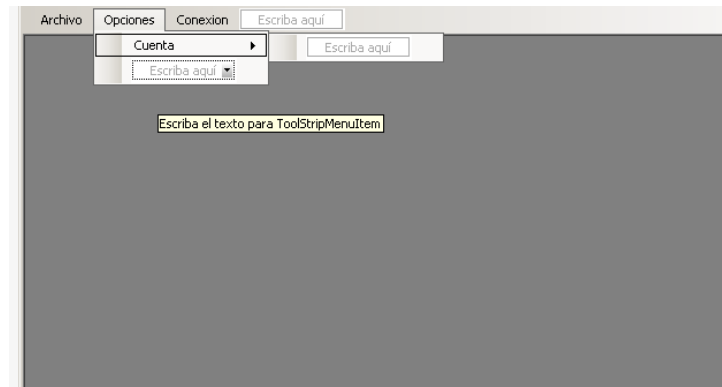
Instrucciones para cada MenuItem y Load Form

```
private N_ConrolesUsuario Controles = new N_ConrolesUsuario();

private void wfMenuPrincipal_Load(object sender, EventArgs e)
{
    Controles.vIniciar();
}

private void toolStripMenuItem2_Click(object sender, EventArgs e)
{
    Controles.vCerrarSesion();
}

private void salirToolStripMenuItem_Click(object sender, EventArgs e)
{
    Controles.vSalirSesion();
}
```



```
private void cuentaToolStripMenuItem_Click(object sender, EventArgs e)
{
    new wfCuenta().Show();
}
```

** Nota para poder usarlo se necesita agregar la referencia

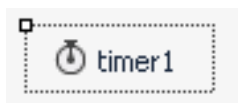
```
using Modulo_de_Seguridad_3._0.Presentacion;
```



Para poder utilizar la revisión de conexión se necesita agregar un MenuItem y agregar en la propiedad "DisplayStyle", la opción "ImageAndText".



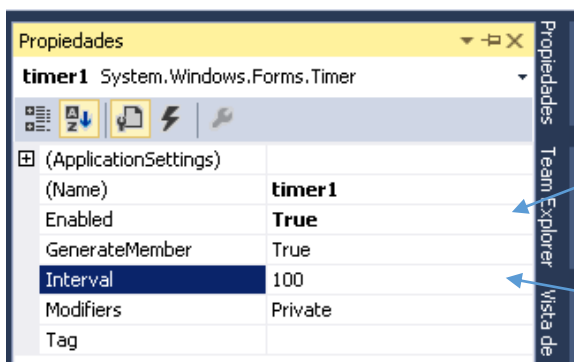
Además para estar haciendo las revisiones de parte del programa se necesita agregar un Timer con las siguientes instrucciones:



```
private void timer1_Tick(object sender, EventArgs e)
{
    conexionToolStripMenuItem.Image = Controles.bmpRevisarConexion();
}
```

**** Nota el StripMenuItem puede variar su nombre, no necesariamente es el indicado en el ejemplo**

Una vez agregada la instrucción se habilita el Timer en las propiedades



Habilitar el timer
con "True"

Agregar Intervalo de Tiempo
(Predeterminado son 100 ms),
se recomienda 1 seg (1000 ms).