Estándares de Codificación para Plataforma Tradicional

Estructura inicial

Configuración de la Pantalla

La configuración de la pantalla debe de estar a 800*600 píxeles, la cual es el estándar de resolución aceptable en los monitores.

Se debe de establecer la propiedad anchor para cada componente que se agregue; de tal forma que al momento de maximizar o minimizar la pantalla, los componentes se ajusten a ésta, esto cuando la forma permita modificar su tamaño.

Comentarios

Al principio del código de cada forma, clase o módulo se debe de especificar el nombre de éste, autor, fecha, número de versión, descripción e historia de cambios (dando por entendido que esta estructura de comentarios, será implementada desde la etapa de desarrollo; aún si no existe o existen cambios) mediante la utilización del siguiente formato:

Nombre: El nombre de la forma, clase o módulo
Autor(es): Toda aquella persona que contribuye en la creación de esta forma, clase o módulo
Descripción: Lo que la forma, clase o módulo hace (el qué, no el cómo)

Versión	Solicitud Servicio	Modificador	Fecha	Descripción del cambio	Versión del estándar
#.#.#	#Solicitud	Nombre	dd/mmm/yyyy	Descripción del cambio	Versión del documento que se empleó para la codificación.

Donde:

Versión: Versión de submódulo liberado.
Solicitud Servicio: Solicitud registrada en Sistema de Gestión para la Atención a Usuarios (SGAU's). Cuando no aplique debe de especificarse N/A.
Modificador: Persona que realizó el cambio.
Fecha: Fecha en que se realizó el cambio.
Descripción del cambio: Cambios / adecuaciones realizadas al proceso.
Versión del estándar: Versión del documento de codificación de plataforma tradicional empleado.

COPIA NO CONTROLADA

Declaraciones

Option Explicit

Se debe de verificar que la opción “Option Explicit” en la sección de herramientas – opciones, se encuentre activada.

Alcance de variables, constantes y objetos

Tanto una variable, constante u objeto, deben ser declaradas al inicio del bloque de código donde se van a emplear, lo cual nos representa su alcance.

Variables

Las variables deben ser siempre definidas con el más bajo alcance posible.
Las variables globales (Public) pueden crear enorme complejidad haciendo la lógica de la aplicación difícil de entender, además de también dificultar el reuso y mantenimiento del código. En visual Basic, variables globales deben usarse solamente cuando no hay otra conveniente forma de compartir datos entre formas. Cuando se haga uso de variables globales, es buena práctica declararlas en un simple módulo, agrupado por funciones. Dar un nombre manejable y entendible al módulo indicando su propósito.

El nombre estándar de variable consiste de un nombre normal para la variable precedida por dos identificadores. El primero debe ser uno de los siguientes:

Identificador Inicial	Descripción
g	Este denota que la variable tiene un alcance público o global.
m	Este denota que la variable tiene un alcance de Módulo (Forma).
l	Este denota que la variable tiene un alcance local.

La segunda letra ó prefijo codifica el tipo de variable acorde a la siguiente tabla, la cual nos denota algunos de los tipos de datos existentes:

Segundo Identificador	Tipo
sht	Short
int	Integer
byt	Byte
dml	Decimal
sng	Single

COPIA NO CONTROLADA

cls	Clase
dbl	Double
bln	Bolean
chr	Char
dte	Date
str	String
est	Estructuras de datos
obj	Objetos instanciados para clases creadas por usuario

El nombre de la variable debe ser sustantivo con la primera letra en capital. No use todas en capital para acrónimos. Los nombres deben ser designados para indicar su intención de uso fácilmente reconocido por un observador casual.

Ejemplos:

```
Dim gstrNombrePersona as string
Dim mstrRangoInteres as string
Dim intContadorCiclo as integer
Dim intContadorRenglones as integer
```

Constantes

Para la declaración de constantes en cualquier sección de código (según el alcance deseado), se debe de asignar el valor en el momento de su declaración y debe respetar el estándar de la declaración de variables además debe contener el sufijo “Constante” al final de la declaración de la misma.

Ejemplos:

```
Const mintIvaConstante as integer =15
```

COPIA NO CONTROLADA

Objetos

Para la declaración de objetos de base de datos, se debe de poner un prefijo seguido del nombre del mismo, con la primera letra en capital, como se muestra en la figura siguiente:

Objeto	Prefijo
Para objetos de SQL 2000 en adelante	
SqlConnection	sqlc
SqlDataReader	sqldr
SqlDataAdapter	sql
DataSet	sqlds
SqlCommand	sqlcmd
DataView	sqldv
SqlTransaction	sqltr
Para objetos de SQL 6.5 u otros Data Sources	
SqlConnection SqlDataReader	sqlcOLEDB sqldrOLEDB
SqlDataAdapter	sqldaOLEDB
DataSet	sqldsOLEDB
SqlCommand	sqlcmdOLEDB
DataView	sqldvOLEDB

Ejemplos:

```
Dim  sqlcIngresos as SqlConnection  'Concección a la base de datos Ingresos
Dim  sqldrFacultades as  SqlDataReader  'Recordset para almacenar temporalmente el catálogo de facultades
```

Inicialización

Todas las variables y/o constantes deben ser inicializadas inmediatamente después de ser declaradas.

Ejemplos:

```
Dim mintContadorRenglones as integer = 0
Const mintIVAConstante as integer = 15
```

Comentarios a las declaraciones de variables, constantes y Objetos

Agregue comentarios en seguida de las declaraciones de variables, constantes y objetos dando una descripción breve del uso que se le dará a ésta.

COPIA NO CONTROLADA

Ejemplos:

```
Dim mintContador As Integer  'Contador del  ciclo renglones del flex grid
Dim gsngTax As Single     'Tax mensual para los cálculos del pedimento
```

Bloques de código

Nota: Entiéndase por bloque de código, una clase, evento, procedimiento, función, ciclo, estructura de control, etc.

Comentarios a bloques de código

Cada bloque de código debe tener la descripción de lo que hace y debe ser precedido por una simple línea en blanco

Ejemplos:

```
Do while mblnEncontro =true

    'Este es un ciclo para algo
    <Bloque de código>
Loop

If condición then

    'Esta es una simple línea de comentario
    <Bloque de código>
End If
```

Procedimientos y funciones

Nombres de procedimientos y funciones deben ser verbos en combinación con la acción que realiza, siendo la primera letra minúscula y la primera letra de cada palabra interna en capital.

Ejemplos

ejecutarConciliación,
encontrarSiguienteCliente, etc.

COPIA NO CONTROLADA

Comentarios a Funciones y Procedimientos

Antes de cada procedimiento o función excepto eventos de los objetos generados por los componentes de la forma (dando por entendido que esta estructura de comentarios, será implementada desde la etapa de desarrollo; aún si no existe o existen cambios), el siguiente estilo de comentario debe ser agregado:

Autor(es): Toda aquella persona que contribuye en la creación de éste procedimiento o función.
Descripción: Lo que el procedimiento ó función hace (el qué, no el cómo).

Comentarios adicionales

Formato a comentarios cuando se agrega o modifica código

Para permitir una fácil determinación de quien ha alterado piezas de código, y deducir cuando los cambios fueron realizados, el siguiente formato debe ser adoptado para todos los comentarios que describan cambios al código:

```
'Nombre desarrollador
'Fecha
'Versión ligada a comentario inicial de la forma
'Descripción del cambio
```

Puntos adicionales

Declaraciones Select Case

Si existe un caso default, entonces se debe incluir en la sección “Case Else”, de lo contrario, se debe de utilizar para desplegar un error.

Ejemplos:

```
Select Case mintScore
        Case Is < 50
                mchrGrade = “F”
        Case Is < =100
                mchrGrade = “A”
        Case Else
                ' No se puede obtener mayor que 100%!!!!
End Select
```

COPIA NO CONTROLADA

Manejo de errores
El manejo de errores es esencial para que el usuario este informado de la problemática del sistema y principalmente para el control de la transacción, por lo que se debe de utilizar el manejo de errores estructurado (Try-Catch-Finally), teniendo en cuenta que si estamos manejando diferentes tipos de conexión a bases de datos, debemos administrar los errores por separado.

```
Try

Catch exSQLOLEDB as SQLOLEDB.SqlException

Catch exSQLClient as SQLClient.SqlException

Catch ex As Exception

Finally

End Try
```

Nota: En cada una de las secciones antes mencionadas dentro del catch, se debe poner dentro del messagebox, la fuente y la descripción de éste, así como los iconos representativos al mensaje de error, lo más claro posible para el usuario.

Operador &
Siempre usar el operador & cuando se concatenan caracteres y/o números.

Controles y Clases

Objetos en la ficha "Windows Forms"
Todos los controles en una forma deben ser renombrados inmediatamente después de que fueron colocados.

Los controles deben ser nombrados con un consistente prefijo que facilite la identificación del tipo de objeto, seguido por el nombre asignado con la primera letra en capital de cada palabra. Estos prefijos se enlistan a continuación:

Control	Prefijo
TextBox	txt
Button	but

COPIA NO CONTROLADA

Control	Prefijo
List Box	lst
Checked List Box	chkl
DataGrid	dgrd
Check Box	chk
Common Dialog	dlg
Form	frm
Combo Box	cbo
List View	lstv
Tree View	trev
Tab Control	tab
Label	lbl
LinkLabel	llbl
MainMenu o MenuStrip	mnu
Radio Button	rbut
Group Box	grb
Picture Box	pic
Timer	tmr
Date Time Picker	dtp
Month Calander	mcr
Rich Text Box	rtxt
Progress Bar	prb
HScrollBar	hsb
VScrollBar	vsb
Splitter	spr
DomainUpDown	dud
NumericUpDown	nud
TrackBar	tkb
ImageList	imgl
HelpProvider	hpr
ToolTip	ttp
ContextMenu	cmnu
ToolBar	tbr
StatusBar	stb
StatusStrip	sts
NotifyIcon	nty
OpenFileDialog	odlg
SaveFileDialog	sdlg
FolderBrowserDialog	fdlg
FontDialog	tdlg

COPIA NO CONTROLADA

Control	Prefijo
ColorDialog	cdlg
PrintDialog	pdlg
PrintPreviewDialog	ppdlg
PrintPreviewControl	ppctl
ErrorProvider	errp
PrintDocument	pd
AxMSFlexGrid	mfg
HierarchicalMSFlexGrid	hmfg
PageSetupDialog	psd
CrystalReportViewer	crv
DataGridViewer	dgv
WindowsMediaPlayer	wmp

Controles de menú

Las aplicaciones frecuentemente usan muchos controles de menú, por lo que se hace necesario tener una convención de nombramientos para estos controles. Los prefijos para los controles de menú deben extenderse más allá de la etiqueta inicial "mnu" mediante la asignación de un prefijo adicional por cada nivel de anidación. La siguiente tabla enlista algunos ejemplos:

Secuencia de leyendas en el menú	Nombre asignado al menú
Archivo Abrir	mnuArchivoAbrir
Archivo Enviar Email	mnuArchivoEnviarEmail
Archivo Enviar Fax	mnuArchivoEnviarFax
Formato Carácter	mnuFormatoCaracter
Ayuda Contenido	mnuAyudaContenido

Objetos

Cada objeto debe ser destruido cuando deje de utilizarse para ayudar a la administración de la memoria. Utilizar Objeto.Dispose si es que el objeto cuenta con este método, en caso contrario objeto = nothing

Conexión a base de datos

En la configuración de los parámetros de conexión a bases de datos, no deberá utilizarse el nombre de usuario ***sa*** en ninguno de los sistemas para su operación.

COPIA NO CONTROLADA

Instalador

El Instalador debe ser compilado en modo de operación Release.

- **En las carpetas del Instalador debe llevar lo siguiente:**
  - Carpeta de Aplicación:
    - Librerías del Sistema.
    - Icono del Sistema (deberá ser el mismo que se incluyo al MDI).
    - Resultado Primario (Ejecutable), cuyo nombre no debe tener espacios en blanco ni mayúsculas.
    - Manual de Ayuda

  - Carpeta Escritorio del Usuario:
    - Acceso directo al Resultado Primario del proyecto (Ejecutable) que se almacenará en la carpeta de la aplicación
    - Configurarle el icono de la carpeta de la aplicación y el nombre.

  - Carpeta Programas del Menú:
    - Acceso directo al Resultado Primario del proyecto (Ejecutable)que se almacenará en la carpeta de la aplicación
    - Configurarle el icono de la carpeta de la aplicación y el nombre.

- **Propiedades del instalador**
  - Las siguientes propiedades del instalador deben ser modificadas (en la ventana de propiedades):
    - Autor: UACH.
    - Manufactura: UACH.
    - Nombre del Producto: Sistema (espacio) Modulo (espacio) Submódulo.
    - Titulo: Sistema (espacio) Modulo (espacio) Submódulo.

- **Prerrequisitos del instalador**
  - Seleccionar la segunda opción donde dice que el instalador deberá ser descargado de la misma localización que la aplicación, luego seleccionar los siguientes prerrequisitos:
    - ".NET Framework 2.0".
    - Si utiliza reportes seleccionar "Crystal Reports for .NET Framework 2.0".

COPIA NO CONTROLADA