



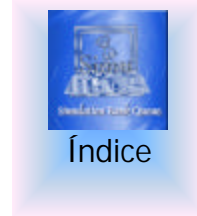
# Manual del Programador

## Simulation Bank Queue

---

Herramienta de Simulación para Analizar el comportamiento de línea de espera o Cola en una Institución Bancaria

# INDICE



Tema	Pág.
<b>INTRODUCCION .....</b>	<b>1</b>
<b>DEFINICIÓN DE LA BASE DE DATOS.....</b>	<b>2</b>
<b>FORMULARIOS .....</b>	<b>7</b>
<b>PROCEDIMIENTOS.....</b>	<b>10</b>
<b>FUNCIONES .....</b>	<b>12</b>
<b>CONSTANTES GLOBALES.....</b>	<b>13</b>
<b>VARIABLES GLOBALES.....</b>	<b>14</b>
<b>CODIGO FUENTE.....</b>	<b>15</b>

## **INTRODUCCION**



Introducción.

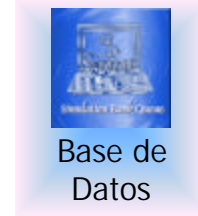
Simulation Bank Queue, conocido también como **SimuBQ** Ver.2.003 diseñado como trabajo de tesis bajo una plataforma de Windows 98, utilizando Microsoft Visual Basic 6, por su ambiente grafico, Microsoft Access 2000, para implementar la base de datos y Segate Crystal Reports como generador de los distintos reportes.

El manual del programador se encuentra descrito de la siguiente forma:

- ✓ Definición de la base de Datos.
- ✓ Formularios.
- ✓ Procedimiento.
- ✓ Funciones.
- ✓ Constantes Globales.
- ✓ Variables Globales.

A continuación se presenta una breve descripción de cada uno de los items antes mencionados:

## DEFINICIÓN DE LA BASE DE DATOS



**Tipo de base de datos:** Aplicación de Microsoft Access  
**Nombre de archivo:** VeVsSimuColas.mdb

### Descripción de tablas:

<b>Bancos</b>		
<b>Columnas</b>		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Bco_Nombre	Texto	255
<b>Índices de tabla</b>		
Nombre	Numero de campo	Campos
PK_Bancos	1	Ascendente

<b>Cajeros</b>		
<b>Columnas</b>		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero largo	4
NumCajero	Entero	2
<b>Índices de tabla</b>		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
Idx_Dia	1	Ascendente
Idx_NumCajero	1	Ascendente
Idx_Suc_Codigo	1	Ascendente
PrimaryKey	4	Ascendente Ascendente Ascendente Ascendente

<b>Cajeros Posición</b>		
<b>Columnas</b>		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero	2
NumCajero	Entero	2
Indice	Entero	2
<b>Índices de tabla</b>		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
Idx_Dia	1	Ascendente
Idx_NumCajero	1	Ascendente
Idx_Suc_Codigo	1	Ascendente
Suc_Codigo	1	Ascendente
SucursalesCajerosPosicion	2	Ascendente Ascendente

<b>CajerosTiempoE</b>		
<b>Columnas</b>		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
NumCliente	Entero largo	4
Dia	Entero largo	4
NumCajero	Entero	2
Hora	Texto	2
HoraEntradaCliente	Fecha/Hora	8
VEntCola	Doble	8
VSaCola	Doble	8
VEntServ	Doble	8
VSaServ	Doble	8
<b>Índices de tabla</b>		
Nombre	Numero de campo	Campos
Bco_Codigo	1	Ascendente
Idx_HoraEntradaCliente	1	Ascendente
Idx_NumCajero	1	Ascendente
PrimaryKey	4	Ascendente Ascendente Ascendente Ascendente
Idx_NumCliente	1	Ascendente
Suc_Codigo	1	Ascendente

<b>CajerosTransacciones</b>		
<b>Columnas</b>		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
NumCliente	Entero largo	4
Dia	Entero largo	4
NumCajero	Entero	2
NoTrans	Entero	2
<b>Índices de tabla</b>		
Nombre	Numero de campo	Campos
CajerosCajerosTransacciones	4	Ascendente Ascendente Ascendente Ascendente
Idx_Bco_Codigo	1	Ascendente
Idx_Dia	1	Ascendente
Idx_NoTrans	1	Ascendente
NumCliente	1	Ascendente
PromTransCajerosTransacciones	2	Ascendente Ascendente
Suc_Codigo	1	Ascendente

<b>ClienteAtendido</b>		
<b>Columnas</b>		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero largo	4 4
NumCliente	Entero largo	4
<b>Índices de tabla</b>		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
Idx_NumCliente	1	Ascendente
Idx_Suc_Codigo	1	Ascendente

### Coeficientes A de Entrada/Salida en Cola.

CoefAEntCola		
Columnas		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero	2
Hora	Entero	2
Entiqueta	Texto	5
CoeficienteA	Doble	8
Índices de tabla		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
Idx_Dia	1	Ascendente
Idx_Hora	1	Ascendente
PrimaryKey	4	Ascendente Ascendente Ascendente Ascendente
Idx_Suc_Codigo	1	Ascendente
SucursalesCoefA_EntCola	2	Ascendente Ascendente

CoefASalCola		
Columnas		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero	2
Hora	Entero	2
Entiqueta	Texto	5
CoeficienteA	Doble	8
Índices de tabla		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
Idx_Dia	1	Ascendente
Idx_Hora	1	Ascendente
PrimaryKey	4	Ascendente Ascendente Ascendente Ascendente
Idx_Suc_Codigo	1	Ascendente
SucursalesCoefA_SalCola	2	Ascendente Ascendente

### Tablas de Entrada/Salida en Cola.

EntCola		
Columnas		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero	2
Hora	Entero	2
Tiempo	Fecha/Hora	8
TiempoE	Doble	8
Índices de tabla		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
Idx_Dia	1	Ascendente
Idx_Hora	1	Ascendente
Idx_Suc_Codigo	1	Ascendente
SucursalesEntCola	2	Ascendente Ascendente

SalCola		
Columnas		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero	2
Hora	Entero	2
Tiempo	Fecha/Hora	8
TiempoS	Doble	8
Índices de tabla		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
Idx_Dia	1	Ascendente
Idx_Hora	1	Ascendente
Idx_Suc_Codigo	1	Ascendente
SucursaleSalCola	2	Ascendente Ascendente

**Tablas : Horas, Sucursales, PromTrans, Promedios Adicionales y PromediosVENTCola**

<b>Horas</b>		
Columnas		
Nombre	Tipo	Tamaño
Id	Fecha/Hora	8
Horas	Fecha/Hora	8
Índices de tabla		
Nombre	Numero de campo	Campos
Id	1	Ascendente

<b>Sucursales</b>		
Columnas		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Suc_Nombre	Texto	255
Índices de tabla		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
BancosSucursales	1	Ascendente
Idx_Suc_Codigo	1	Ascendente
PK_Sucursales	2	Ascendente Ascendente

<b>PromTrans</b>		
Columnas		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
No.Trans	Entero	2
Nombre_Transaccion	Texto	255
Promedios	Fecha/Hora	8
TpromSegs	Entero largo	4
Porcentaje	Doble	8
Índices de tabla		
Nombre	Numero de campo	Campos
Idx_Bco_Codigo	1	Ascendente
BancosPromTrans	1	Ascendente
PrimaryKey	2	Ascendente Ascendente

<b>PromediosAdicionales</b>		
Columnas		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero largo	4
Hora	Texto	50
NumAdiionales	Entero	2
Veco	Doble	8
Vsca	Doble	8
Índices de tabla		
Nombre	Numero de campo	Campos
Bco_Codigo	1	Ascendente
Suc_Codigo	1	Ascendente
SucursalesCajerosPosicion	2	Ascendente Ascendente

<b>PromediosVENTCola</b>		
Columnas		
Nombre	Tipo	Tamaño
Bco_Codigo	Entero largo	4
Suc_Codigo	Entero largo	4
Dia	Entero largo	4
Hora	Texto	50
NumPersonasSalPorCajero	Entero	2
Valor	Doble	8
Valor2	Doble	8
VentradaServ	Doble	8
VvalidaServ	Doble	8
Índices de tabla		
Nombre	Numero de campo	Campos
Bco_Codigo	1	Ascendente
Suc_Codigo	1	Ascendente

<b>GeneracionEncabezado</b>		
<b>Columnas</b>		
<b>Nombre</b>	<b>Tipo</b>	<b>Tamaño</b>
Bco_Codigo	Entero largo	4
Bco_Codigo	Entero largo	4
DiaCalendario	Fecha/Hora	8
Id	Entero largo	4
Dia	Entero	2
HoraInicio	Entero	2
HoraFin	Entero	2
NumServidores	Entero	2
CantPersonasEntran	Entero	2
Deserciones	Entero	2
CantTransacciones	Entero largo	4
CantTransCaja1	Entero largo	4
CantTransCaja2	Entero largo	4
CantTransCaja3	Entero largo	4
CantTransCaja4	Entero largo	4
CantTransCaja5	Entero largo	4
CantTransCaja6	Entero largo	4
CantTransCaja7	Entero largo	4
CantTransCaja8	Entero largo	4
CantTransCaja9	Entero largo	4
CantTransCaja10	Entero largo	4
CantTransCaja11	Entero largo	4
CantTransCaja12	Entero largo	4
CantTransCaja13	Entero largo	4
CantTransCaja14	Entero largo	4
CantTransCaja15	Entero largo	4
<b>Índices de tabla</b>		
Nombre	Número de campos	Campos:
Id	1	Ascendente
Idx_Bco_Codigo	1	Ascendente
Idx_Suc_Codigo	1	Ascendente
pkGeneracionEncabezado	4	Ascendente Ascendente Ascendente Ascendente
SucursalesGeneracionEncabezado	2	Ascendente Ascendente



## FORMULARIOS



**Lenguaje de programación:** Visual Basic 6.0  
**Nombre de archivo de proyecto:** VeVsSimuColas.vbp

---

<b>Nombre</b>	FrmGraficas
<b>Archivo</b>	frmGraficas.frm
<b>Descripción</b>	Muestra en diferentes formatos gráficos la representación de la cantidad de personas que entran a la cola, salen de la cola y la cantidad de personas atendidas por cajeros.

---

---

<b>Nombre</b>	FrmImportar
<b>Archivo</b>	frmImportar.frm
<b>Descripción</b>	En este formulario se selecciona un archivo en formato de Microsoft Excel el cual contendrá la siguiente información: <ul style="list-style-type: none"><li>- Entrada y salida de cola</li><li>- Transacciones del banco</li></ul> Existe la posibilidad de seleccionar de estas tres opciones cual es la que se desea importar, siendo completadas manualmente por el usuario la información no importada.

---

---

<b>Nombre</b>	frmInicioSimulacion
<b>Archivo</b>	frmInicioSimulacion.frm
<b>Descripción</b>	Formulario de confirmación de si se inicia o no una simulación

---

---

<b>Nombre</b>	frmParametros
<b>Archivo</b>	frmParametros.frm
<b>Descripción</b>	Formulario mediante el cual se obtienen los parámetros iniciales de una simulación los cuales son: la fecha, hora inicial, hora final y la cantidad de servidores

---

---

<b>Nombre</b>	FrmPress
<b>Archivo</b>	FrmPress.frm
<b>Descripción</b>	Formulario de presentación del sistema de simulación

---

---

<b>Nombre</b>	frmProgreso
---------------	-------------

---

---

<b>Archivo</b>	frmProgreso.frm
<b>Descripción</b>	Formulario que muestra el avance del proceso de adición de la información digitada por el usuario o importada mediante un archivo de Microsoft Excel relacionada a la entrada y salida de cola y servicio y a las transacciones del banco.

---

---

<b>Nombre</b>	frmRecepcionDeDatos
<b>Archivo</b>	frmRecepcionDeDatos.frm
<b>Descripción</b>	Formulario en el cual se introduce la información de campo referente a la entrada y salida de cola, y de transacciones. En esta forma se presenta también la información importada del archivo de Microsoft Excel.

---

---

<b>Nombre</b>	frmRecepcionDeDatos
<b>Archivo</b>	frmRecepcionDeDatos.frm
<b>Descripción</b>	Formulario en el cual se introduce la información de campo referente a la entrada y salida de cola y de transacciones. En esta forma se presenta también la información importada del archivo de Microsoft Excel. Esta información es completada ya sea que se esté creando un nuevo banco, una nueva sucursal ó si se desea redefinir la información de campo utilizada para la generación de los coeficientes utilizados en el proceso de simulación.

---

---

<b>Nombre</b>	frmReportes
<b>Archivo</b>	frmReportes.frm
<b>Descripción</b>	Contiene los diferentes reportes que genera el sistema, como por ejemplo: <ul style="list-style-type: none"><li>- Promedio de Entrada/Salida en cola</li><li>- Promedio de Entrada/Salida en servicio</li><li>- Cantidad de personas atendidas por cajero</li><li>- Cantidad de transacciones por cajero</li><li>- Tipos de transacciones</li><li>- Tiempo en sistema de personas seleccionadas</li></ul>

---

---

<b>Nombre</b>	FrmResultados
<b>Archivo</b>	frmResultados.frm
<b>Descripción</b>	Muestra los resultados generados de cada simulación, entre la información que presenta se tiene: Fecha, hora inicial, hora final, cantidad de personas que entran, cantidad de personas que salen, número de cajeros, número de deserciones y cantidad de transacciones.

---

---

<b>Nombre</b>	FrmSeleccionAnalisis
<b>Archivo</b>	frmSeleccionAnalisis.frm
<b>Descripción</b>	Permite elegir una nueva simulación (efectuando una nueva entrada de información de campo) o realizar la simulación con la información ya almacenada en la base de datos.

---

---

<b>Nombre</b>	frmSeleccionAnalisis
---------------	----------------------

---

---

<b>Archivo</b>	frmSeleccionAnalisis.frm
<b>Descripción</b>	Permite elegir una nueva simulación (efectuando una nueva entrada de información de campo) o realizar la simulación con la información ya almacenada en la base de datos.

---

<b>Nombre</b>	frmSeleccionBanco
<b>Archivo</b>	frmSeleccionBanco.frm
<b>Descripción</b>	Permite crear un banco y una sucursal, una sucursal ó ingresar al sistema seleccionando una existente.

---

<b>Nombre</b>	frmSimulacion
<b>Archivo</b>	frmSimulacion.frm
<b>Descripción</b>	Es el formulario principal del sistema, su función es presentar en forma gráfica la simulación de la cola y la atención por parte de los cajeros, este formulario integra además el acceso a las opciones de: <ul style="list-style-type: none"> <li>- Resultados de simulación</li> <li>- Gráficos representativos de la información generada</li> <li>- Generación de reportes</li> <li>- Almacenamiento en la base de la información de la simulación</li> <li>- Adición y retiro de cajeros</li> <li>- Toma del tiempo de una persona en el sistema</li> <li>- Introducción de casos especiales</li> </ul>

---

<b>Nombre</b>	frmSimulacionPersona
<b>Archivo</b>	frmSimulacionPersona.frm
<b>Descripción</b>	Muestra el número de la persona que ha entrado al sistema desde que inicio la simulación, y a la cual se le tomara el tiempo total que estará en el sistema.

---

<b>Nombre</b>	frmThem
<b>Archivo</b>	scroll1.frm
<b>Descripción</b>	Presenta el acerca de... del sistema

---

## PROCEDIMIENTOS



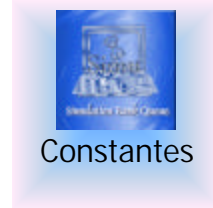
Nombre del procedimiento	Parámetros que recibe	Descripción de su función
Vinculación	Objeto de base de datos, nombre del vinculo a crear, cadena de conexión a Excel, nombre de la Hoja de Excel	Crea los vínculos a las hojas del libro de Excel según el archivo seleccionado.
ExisteTabla	Objeto de base de datos, nombre de la tabla	Elimina los vínculos anteriores a las hojas de un libro de Excel importado con anterioridad.
Main		Crea el DSN de usuario en computadora utilizando el DBEngine de Microsoft DAO y realiza la conexión a la base de datos VsSimuColas.mdb mediante ADO v2.5(Activex Data Object)
Espera	Tipo de puntero	Cambia el puntero del ratón de reloj de arena a flecha y viceversa
procCerrarRecordset	Objeto tipo recordset	Cierra un objeto recordset pasado como argumento
procConvMayusculas	Valor entero as cii de la tecla presionada	Convierte de mayúscula a minúsculas
CerrarSistema	Objeto tipo conexión ADO	Cierra la variable de conexión de la base de datos del sistema
GrabarInformacionDeNuevoBanco		Tiene como función la creación de un nuevo banco y sucursal, una nueva sucursal y una nueva información de campo de entrada y salida de cola y servicio y tipos de transacción
CoeficientesDeCola	Código de banco, código de sucursal, tipo de cola (entrada o salida), nombre del	Calcula los coeficientes A de entrada y salida de cola. Guarda además la

	coeficiente	información de coeficientes en la base de datos.
procCalcularPromediosVelocidad_E_S_ColaServicio	Dia, número de cajero	Calcula la velocidad promedio de entrada/salida de la cola y el servicio
procGenerarTransacciones	Dia de la simulación	Genera las transacciones por cada cajero basando se en el tiempo promedio de servicio y en el tiempo que dura cada transacción haciendo referencia a la tabla PromTrans
procCalcularPromediosVelocidad_S_Cola	Dia, número de cajeros	Calcula la velocidad promedio de salida de la cola la cual varia de pendiendo del número de cajeros en servicio
procCalcularPromediosTiempoEnServicio	Dia, número de cajeros	Calcula el tiempo promedio en servicio para un día específico
procCalculosAdicionales	Dia	Calcula los nuevos valores de velocidades en cola para los clientes que no son atendidos en su hora de ingreso
procCreacionDePromedios		Crea los promedios de entrada / salida de cola para la presentación en reporte al finalizar cada simulación

**FUNCIONES**

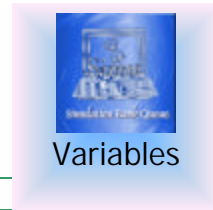
Nombre de la función	Parámetros que recibe	Parámetros que devuelve	Descripción de su función
fncHoraValida	Hora en formato dd/mm/yyyy	Valor booleano, FALSE la fecha no es válida, TRUE la fecha es válida	Indica si una fecha es válida o no
fncFORMATO_HORA	Valor de hora	Formato de hora	Devuelve la hora pasada como argumento en formato hh:mm:ss AM / PM
fncFORMATO_HORAMMSS	Valor de hora	Formato de hora	Devuelve la hora pasada como argumento en formato mm:ss
Lee_Campo	Valor de cadena o valor nulo	Valor de cadena pasada como argumento o carácter en blanco	Si el valor que recibe como argumento es nulo devuelve un carácter en blanco de lo contrario devuelve el mismo valor de cadena
fncNumeroTransacciones	Día	Número de transacciones	Devuelve el número de transacciones realizadas en total por todos los cajeros para una fecha en particular
procNuevaSimulacion	Día de simulación	Id de simulación	Devuelve el Id de Simulación para la fecha pasada como argumento en caso exista una simulación anterior en la misma fecha

## CONSTANTES GLOBALES



<b>Nombre de constante</b>	<b>Tipo de dato</b>	<b>Función</b>
NOMBRE_BASEDATOS	String	Representa el nombre físico del archivo de base de datos del sistema
MSGVALI	String	Representa la cadena de título 'Mensaje de validación' en los mensajes del sistema
MSGCONF	String	Representa la cadena de título 'Mensaje de confirmación' en los mensajes del sistema
MSGINFO	String	Representa la cadena de título 'Mensaje de información' en los mensajes del sistema

## **VARIABLES GLOBALES**



<b>Nombre de constante</b>	<b>Tipo de dato</b>	<b>Función</b>
gCodBanco	Long	Código del Banco seleccionado para efectuar la simulación
gCodSucur	Long	Código de la sucursal seleccionada para efectuar la simulación
gblnSimulacionAnterior	Boolean	Indica si se realiza la simulación con datos actuales o con nuevos
gstrNombreBanco	String	Nombre del banco seleccionado para efectuar la simulación
gstrNombreSucursal	String	Nombre de la sucursal seleccionada para efectuar la simulación
gldSimulacion	Long	Id de la simulación en proceso



## CODIGO FUENTE



Codigo

### FrmComparacion

Option Explicit

Sub procIdSimulacionesPasadas()

Dim rstSimus As New ADODB.Recordset

```
rstSimus.Open "SELECT Id " _
& "From GeneracionEncabezado " _
& "Where Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And DiaCalendario = #" &
frmSimulacion.lblFecha.Tag & "# And Id <> " & gIdSimulacion & " " _
& "ORDER BY Id", gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstSimus
While Not .EOF
cbxId.AddItem CStr.Fields(0)
.MoveNext
Wend
End With
If cbxId.ListCount > 0 Then cbxId.ListIndex = 0
procCerrarRecordset rstSimus
```

End Sub

Private Sub cbxId\_Click()

'Mostrar información de la Simulacion seleccionada

Dim rstInfo As New ADODB.Recordset

```
rstInfo.Open "SELECT HoraInicio, HoraFin, NumServidores, " _
& "CantPersonasEntran, CantPersonasSalen, " _
& "Deserciones, CantTransacciones, " _
& "CantCola, VEntCola, " _
& "VSalCola, VEntServ, VSalServ " _
& "From GeneracionEncabezado " _
& "WHERE Bco_Codigo=" & CStr(gCodBanco) & " AND " _
& "Suc_Codigo=" & CStr(gCodSucur) & " AND " _
& "DiaCalendario=#" & frmSimulacion.lblFecha.Tag & "# AND " _
& "Id=" & cbxId & ";", gCn, adOpenStatic, adLockOptimistic, adCmdText
```

With rstInfo

If Not .EOF Then

If .Fields(0) >= 12 Then

Select Case .Fields(0)

Case 12

lblHoraIni(1) = "12:00 PM"

Case 13

lblHoraIni(1) = "01:00 PM"

Case 14

lblHoraIni(1) = "02:00 PM"

Case 15

lblHoraIni(1) = "03:00 PM"

Case 16

lblHoraIni(1) = "04:00 PM"

Case 17

lblHoraIni(1) = "05:00 PM"

End Select

Else

lblHoraIni(1) = Format(.Fields(0), "0#") & ":00 AM"

End If

If .Fields(1) >= 12 Then

Select Case .Fields(1)

Case 12

lblHoraFin(1) = "12:00 PM"

Case 13

```

        lblHoraFin(1) = "01:00 PM"
    Case 14
        lblHoraFin(1) = "02:00 PM"
    Case 15
        lblHoraFin(1) = "03:00 PM"
    Case 16
        lblHoraFin(1) = "04:00 PM"
    Case 17
        lblHoraFin(1) = "05:00 PM"
    End Select
Else
    lblHoraFin(1) = Format(.Fields(1), "0#") & ":00 AM"
End If

lblCantEntrado(1) = .Fields(3)
lblCantSalida(1) = .Fields(4)
lblCantServ(1) = .Fields(2)
lblNumDeserciones(1) = .Fields(5)
lblNumTransacciones(1) = .Fields(6)
lblCantCola(1) = !CantCola
lblVPromEntCola(1) = Format(Int(!VEntCola / 60), "0#") & ":" & Format(!VEntCola - (Int(!VEntCola / 60) * 60), "0#") & "
min"
lblVPromSalCola(1) = Format(Int(!VSalCola / 60), "0#") & ":" & Format(!VSalCola - (Int(!VSalCola / 60) * 60), "0#") & "
min"
lblVPromEntServ(1) = Format(Int(!VEntServ / 60), "0#") & ":" & Format(!VEntServ - (Int(!VEntServ / 60) * 60), "0#") & "
min"
lblVPromSalServ(1) = Format(Int(!VSalServ / 60), "0#") & ":" & Format(!VSalServ - (Int(!VSalServ / 60) * 60), "0#") & "
min"
    End If
    End With
    procCerrarRecordset rstInfo
End Sub
Private Sub cbxId_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub cmbSimu_Click(Index As Integer)
    If Index = 0 Then
        With CrystalReport1
            .ReportFileName = App.Path & "\repComparacion.rpt"
            .Formulas(0) = "NUMSIMU1=" & lblId & ""
            .Formulas(1) = "HORAINI1=" & lblHoraIni(0) & ""
            .Formulas(2) = "HORAFIN1=" & lblHoraFin(0) & ""
            .Formulas(3) = "CANTENTRAN1=" & lblCantEntrado(0) & ""
            .Formulas(4) = "CANTSALEN1=" & lblCantSalida(0) & ""
            .Formulas(5) = "CANTCOLA1=" & lblCantCola(0) & ""
            .Formulas(6) = "CANTSERV1=" & lblCantServ(0) & ""
            .Formulas(7) = "DESERCIONES1=" & lblNumDeserciones(0) & ""
            .Formulas(8) = "TRANSACCIONES1=" & lblNumTransacciones(0) & ""
            .Formulas(9) = "VENTCOLA1=" & lblVPromEntCola(0) & ""
            .Formulas(10) = "VSALCOLA1=" & lblVPromSalCola(0) & ""
            .Formulas(11) = "VENTSERV1=" & lblVPromEntServ(0) & ""
            .Formulas(12) = "VSALSERV1=" & lblVPromSalServ(0) & ""

            .Formulas(13) = "NUMSIMU2=" & cbxId & ""
            .Formulas(14) = "HORAINI2=" & lblHoraIni(1) & ""
            .Formulas(15) = "HORAFIN2=" & lblHoraFin(1) & ""
            .Formulas(16) = "CANTENTRAN2=" & lblCantEntrado(1) & ""
            .Formulas(17) = "CANTSALEN2=" & lblCantSalida(1) & ""
            .Formulas(18) = "CANTCOLA2=" & lblCantCola(1) & ""
            .Formulas(19) = "CANTSERV2=" & lblCantServ(1) & ""
            .Formulas(20) = "DESERCIONES2=" & lblNumDeserciones(1) & ""
            .Formulas(21) = "TRANSACCIONES2=" & lblNumTransacciones(1) & ""
            .Formulas(22) = "VENTCOLA2=" & lblVPromEntCola(1) & ""
            .Formulas(23) = "VSALCOLA2=" & lblVPromSalCola(1) & ""
            .Formulas(24) = "VENTSERV2=" & lblVPromEntServ(1) & ""
            .Formulas(25) = "VSALSERV2=" & lblVPromSalServ(1) & ""

            .Formulas(26) = "BANCO=" & gstrNombreBanco & ""
            .Formulas(27) = "SUCURSAL=" & gstrNombreSucursal & ""
        End With
    End If
End Sub

```

```

        .Formulas(28) = "FECHA=" & lblFecha & ""
        .Action = 1
    End With
Else
    Unload frmComparacion
End If
End Sub
Private Sub Form_Load()
    'Presentar información de simulación actual
    lblNomBanco = gstrNombreBanco
    lblNomSucursal = gstrNombreSucursal
    lblFecha = frmSimulacion.lblFecha

    lblId = gIdSimulacion
    lblHoraIni(0) = frmSimulacion.lblHoraIni
    lblHoraFin(0) = frmSimulacion.lblHoraFin
    lblCantEntrado(0) = frmSimulacion.lblCantEntrado
    lblCantSalida(0) = frmSimulacion.lblCantSalida
    lblCantServ(0) = frmSimulacion.lblCantServ
    lblNumDeserciones(0) = frmSimulacion.lblDeserciones
    lblNumTransacciones(0) = fncNumeroTransacciones(Val(frmSimulacion.lblDia))
    lblCantCola(0) = frmSimulacion.lblCantCola
    lblVPromEntCola(0) = Format(Int(frmSimulacion.lng_VEntCola / 60), "0#") & ":" & Format(frmSimulacion.lng_VEntCola -
(Int(frmSimulacion.lng_VEntCola / 60) * 60), "0#") & " min"
    lblVPromSalCola(0) = Format(Int(frmSimulacion.lng_VSalCola / 60), "0#") & ":" & Format(frmSimulacion.lng_VSalCola -
(Int(frmSimulacion.lng_VSalCola / 60) * 60), "0#") & " min"
    lblVPromEntServ(0) = Format(Int(frmSimulacion.lng_VEntServ / 60), "0#") & ":" & Format(frmSimulacion.lng_VEntServ -
(Int(frmSimulacion.lng_VEntServ / 60) * 60), "0#") & " min"
    lblVPromSalServ(0) = Format(Int(frmSimulacion.lng_VSalServ / 60), "0#") & ":" & Format(frmSimulacion.lng_VSalServ -
(Int(frmSimulacion.lng_VSalServ / 60) * 60), "0#") & " min"
    'Obtener Id de las simulaciones pasadas
    proclIdSimulacionesPasadas
End Sub

```

### **FrmGraficosExcel**

```

Option Explicit

Public vntValorTipoGrafico
Dim Iniciando As Boolean
Sub procConfigurarGrafico()
    'Configurar gráfico
    On Error GoTo ErrHandler

    Kill App.Path & "\\Graficos.xls"
    CrearHojaExcel cbxTipoInfo.ListIndex, frmSimulacion.lblFecha.Tag

    Exit Sub
ErrHandler:
    If Err.Number = 53 Then
        Resume Next
    Else
        MsgBox Err.Description
    End If
End Sub
Private Sub cbxTipoGrafico_Click()
    Select Case cbxTipoGrafico.ListIndex
        Case 0
            vntValorTipoGrafico = xl3DColumnClustered
        Case 1
            vntValorTipoGrafico = xl3DBarClustered
        Case 2
            vntValorTipoGrafico = xlLine
        Case 3
            vntValorTipoGrafico = xl3DArea
    End Select
End Sub

```

```
Case 4
    vntValorTipoGrafico = xl3DPieExploded
End Select

procConfigurarGrafico
OLE1.Action = 1
End Sub
Private Sub cbxTipoGrafico_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub cbxTipoInfo_Click()
    If Not Iniciando Then
        Select Case cbxTipoGrafico.ListIndex
            Case 0
                vntValorTipoGrafico = xl3DColumnClustered
            Case 1
                vntValorTipoGrafico = xl3DBarClustered
            Case 2
                vntValorTipoGrafico = xlLine
            Case 3
                vntValorTipoGrafico = xl3DArea
            Case 4
                vntValorTipoGrafico = xl3DPieExploded
        End Select
    End Sub
Private Sub cbxTipoGrafico_Click()
    Select Case cbxTipoGrafico.ListIndex
        Case 0
            vntValorTipoGrafico = xl3DColumnClustered
        Case 1
            vntValorTipoGrafico = xl3DBarClustered
        Case 2
            vntValorTipoGrafico = xlLine
        Case 3
            vntValorTipoGrafico = xl3DArea
        Case 4
            vntValorTipoGrafico = xl3DPieExploded
    End Select

    procConfigurarGrafico
    OLE1.Action = 1
End Sub
Private Sub cbxTipoGrafico_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub cbxTipoInfo_Click()
    If Not Iniciando Then
        Select Case cbxTipoGrafico.ListIndex
            Case 0
                vntValorTipoGrafico = xl3DColumnClustered
            Case 1
                vntValorTipoGrafico = xl3DBarClustered
            Case 2
                vntValorTipoGrafico = xlLine
            Case 3
                vntValorTipoGrafico = xl3DArea
            Case 4
                vntValorTipoGrafico = xl3DPieExploded
        End Select

        If cbxTipoInfo.ListIndex = 2 And cbxTipoGrafico.ListCount = 4 Then
            cbxTipoGrafico.AddItem "Pastel"
            procConfigurarGrafico
            OLE1.Action = 1
        Else
            If cbxTipoGrafico.ListCount = 5 And cbxTipoInfo.ListIndex <> 2 Then
                cbxTipoGrafico.RemoveItem 4
                cbxTipoGrafico.ListIndex = 0
            End If
        End If
    End If
End Sub
```

```

Else
    procConfigurarGrafico
    OLE1.Action = 1
End If
End If

If cbxTipoInfo.ListIndex = 2 Then
    'Mostrar tabla de transacciones
    Dim db_local As Database
    Dim rstTrans As DAO.Recordset
    Dim itm As ListItem

    lsvTransacc.ListItems.Clear
    Set db_local = OpenDatabase(App.Path & "\\VeVsSimuColas.mdb")
    Set rstTrans = db_local.OpenRecordset("SELECT PromTrans.NoTrans, PromTrans.[Nombre Transacción] " _
        & "FROM PromTrans INNER JOIN CajerosTransacciones ON (PromTrans.NoTrans = CajerosTransacciones.NoTrans)
AND (PromTrans.Bco_Codigo = CajerosTransacciones.Bco_Codigo) " _
        & "Where (((PromTrans.Bco_Codigo) = " & gCodBanco & ") And ((CajerosTransacciones.NumCajero) <> -1)) " _
        & "GROUP BY PromTrans.NoTrans, PromTrans.[Nombre Transacción] ORDER BY PromTrans.NoTrans;",
dbOpenSnapshot)
    With rstTrans
        While Not .EOF
            Set itm = lsvTransacc.ListItems.Add(, , , 0)
            itm.SubItems(1) = .Fields(0)
            itm.SubItems(2) = .Fields(1)
            .MoveNext
        Wend
        .Close
    End With
    Set rstTrans = Nothing
    db_local.Close
    Set db_local = Nothing
    chkTrans.Visible = True
Else
    'Ocultar tabla de transacciones
    lsvTransacc.Visible = False
    chkTrans.Visible = False
    chkTrans.Value = 0
End If
End If
End Sub

Private Sub cbxTipoInfo_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub

Private Sub chkTrans_Click()
    If chkTrans.Value = 0 Then
        lsvTransacc.Visible = False
    Else
        lsvTransacc.Visible = True
    End If
End Sub

Private Sub cmbCerrar_Click()
    Unload frmGraficosExcel
    Set frmGraficosExcel = Nothing
    frmSimulacion.Show
End Sub

Private Sub Form_Load()
    Iniciando = True

    frmSimulacion.Hide
    With cbxTipoInfo
        .AddItem "Personas que entran/salen de la cola"
        .AddItem "Cantidad de personas atendidas por cajero"
        .AddItem "Transacciones realizadas"
        .AddItem "Promedio de entrada/salida en cola"
        .AddItem "Promedio de entrada/salida en servicio"
        .ListIndex = 0
    End With

```

```

Iniciando = False
With cbxTipoGrafico
    .AddItem "Columnas"
    .AddItem "Barras"
    .AddItem "Líneas"
    .AddItem "Áreas"
    .ListIndex = 0
End With
End Sub

```

### **FrmImportar**

Option Explicit

```
Sub CargarInformaciónDeExcel(By Val strTipo As Integer)
```

```
    Dim itm As ListItem
```

```
    Dim dblocal As DAO.Database
```

```
    Dim rstInfo As DAO.Recordset
```

```
    Set dblocal = OpenDatabase(App.Path & "\VeVsSimuColas.mdb", False, False)
```

```
    Select Case strTipo
```

```
        Case 0 'Colas (Entrada/Salida)
```

```
            Vinculación dblocal, "EntCola", "EXCEL 8.0;DATABASE=" & txtArchivo, "EntCola$"

```

```
            Set rstInfo = dblocal.OpenRecordset("EntCola", dbOpenSnapshot)
```

```
            With rstInfo
```

```
                frmRecepcionDeDatos.lsvTbl(0).ListItems.Clear
```

```
                While Not .EOF
```

```
                    Set itm = frmRecepcionDeDatos.lsvTbl(0).ListItems.Add(frmRecepcionDeDatos.lsvTbl(0).ListItems.Count + 1, , , 0)
```

```
                    itm.SubItems(1) = .Fields(0)
```

```
                    itm.SubItems(2) = .Fields(1)
```

```
                    itm.SubItems(3) = Mid(Format(.Fields(2), "hh:mm:ss"), 4, 5)
```

```
                    itm.SubItems(4) = Minute(.Fields(2)) * 60 + Second(.Fields(2))
```

```
                    DoEvents
```

```
                    .MoveNext
```

```
                Wend
```

```
                .Close
```

```
            End With
```

```
            Set rstInfo = Nothing
```

```
            Vinculación dblocal, "SalCola", "EXCEL 8.0;DATABASE=" & txtArchivo, "SalCola$"

```

```
            Set rstInfo = dblocal.OpenRecordset("SalCola", dbOpenSnapshot)
```

```
            With rstInfo
```

```
                frmRecepcionDeDatos.lsvTbl(1).ListItems.Clear
```

```
                While Not .EOF
```

```
                    Set itm = frmRecepcionDeDatos.lsvTbl(1).ListItems.Add(frmRecepcionDeDatos.lsvTbl(1).ListItems.Count + 1, , , 0)
```

```
                    itm.SubItems(1) = .Fields(0)
```

```
                    itm.SubItems(2) = .Fields(1)
```

```
                    itm.SubItems(3) = Mid(Format(.Fields(2), "hh:mm:ss"), 4, 5)
```

```
                    itm.SubItems(4) = Minute(.Fields(2)) * 60 + Second(.Fields(2))
```

```
                    DoEvents
```

```
                    .MoveNext
```

```
                Wend
```

```
                .Close
```

```
            End With
```

```
            Set rstInfo = Nothing
```

```
        Case 1 Transacciones
```

```
            Vinculación dblocal, "Transacciones", "EXCEL 8.0;DATABASE=" & txtArchivo, "Transacciones$"

```

```
            Set rstInfo = dblocal.OpenRecordset("Transacciones", dbOpenSnapshot)
```

```
            With rstInfo
```

```
                frmRecepcionDeDatos.lsvTbl(4).ListItems.Clear
```

```
                While Not .EOF
```

```
                    Set itm = frmRecepcionDeDatos.lsvTbl(4).ListItems.Add(frmRecepcionDeDatos.lsvTbl(4).ListItems.Count + 1, , , 0)
                    Lee_Campo(.Fields(0), , 0)
```

```
                    itm.SubItems(1) = Mid(Format(.Fields(1), "hh:mm:ss"), 4, 5)
```

```
                    itm.SubItems(2) = Minute(.Fields(1)) * 60 + Second(.Fields(1))
```

```
                    itm.SubItems(3) = .Fields(2)
```

```
                    DoEvents
```

```

        .MoveNext
    Wend
    .Close
End With
Set rstInfo = Nothing
End Select

dblocal.Close
Set dblocal = Nothing
End Sub
Private Sub chkSimu_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub cmbRuta_Click()
    CommonDialog1.DialogTitle = "Seleccionando archivo"
    CommonDialog1.Filter = "Archivo de Microsoft Excel (*.xls)*.xls"
    CommonDialog1.ShowOpen
    txtArchivo = CommonDialog1.FileName
    If Dir(txtArchivo, vbArchive) <> "" And txtArchivo <> "" And Right(txtArchivo, 4) = ".xls" Then
        chkSimu(0).Enabled = True
        chkSimu(1).Enabled = True
        cmbSimu(0).Enabled = True
    Else
        chkSimu(0).Enabled = False
        chkSimu(1).Enabled = False
        cmbSimu(0).Enabled = False
    End If
End Sub
Private Sub cmbSimu_Click(Index As Integer)
    If Index = 0 Then
        If Trim(txtArchivo) = "" Then
            MsgBox "Debe especificar la ruta del archivo de Excel", vbInformation, MSGVALI
            txtArchivo.SetFocus
        Else
            If Dir(txtArchivo, vbArchive) <> "" And txtArchivo <> "" And Right(txtArchivo, 4) = ".xls" Then
                If chkSimu(0).Value = 0 And chkSimu(1).Value = 0 Then
                    MsgBox "Debe seleccionar algun dato a importar", vbInformation, MSGVALI
                    chkSimu(0).SetFocus
                Else
                    Espera
                    Frame2.Visible = True
                    If chkSimu(0).Value = 1 Then
                        CargarInformaciónDeExcel 0
                    End If
                    If chkSimu(1).Value = 1 Then
                        CargarInformaciónDeExcel 1
                    End If

                    Frame2.Visible = False
                    Espera 1
                    frmRecepcionDeDatos.cmbSimu(3).Enabled = False
                    Unload frmImportar
                End If
            Else
                MsgBox "Archivo no existe", vbInformation, MSGVALI
                txtArchivo.SetFocus
            End If
        End If
    Else
        Unload frmImportar
    End If
End Sub
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode = 27 Then
        Unload frmImportar
    End If
End Sub
Private Sub Form_Load()
    Set cmbRuta.Picture = Image1.Picture
End Sub

```

```

Private Sub txtArchivo_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub txtArchivo_Validate(Cancel As Boolean)
    If Dir(txtArchivo, vbArchive) <> "" And txtArchivo <> "" And Right(txtArchivo, 4) = ".xls" Then
        chkSimu(0).Enabled = True
        chkSimu(1).Enabled = True
        cmbSimu(0).Enabled = True
        chkSimu(0).SetFocus
    Else
        chkSimu(0).Enabled = False
        chkSimu(1).Enabled = False
        cmbSimu(0).Enabled = False
    End If
End Sub

```

### **FrmParametros**

```

Option Explicit
Dim blnCambioFecha As Boolean

Private Sub cbxHora_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub cmbSimu_Click(Index As Integer)
    Dim Dia As Byte
    Dim dblocal As DAO.Database

    If Index = 0 Then
        If Val(cbxHora(0)) >= Val(cbxHora(1)) Then
            MsgBox "Hora inicial no puede ser mayor o igual que hora final", vbInformation, MSGVALI
            cbxHora(0).SetFocus
        Else
            If Val(txtSimu) = 0 Then
                MsgBox "La cantidad de servidores no puede ser cero", vbInformation, MSGVALI
                txtSimu.SetFocus
                Exit Sub
            End If
            If Val(txtSimu) > 15 Then
                MsgBox "La cantidad de servidores no puede ser mayor de 15", vbInformation, MSGVALI
                txtSimu.SetFocus
                Exit Sub
            End If
            Set dblocal = OpenDatabase(App.Path & "\VeVsSimuColas.mdb", False, False)

            'Estableciendo el Id de la Simulación a generar
            procNuevaSimulacion drpCalendario.Value

            'Grabando en Encabezado
            dblocal.Execute "INSERT INTO GeneracionEncabezado " _
                & "(Bco_Codigo, Suc_Codigo, Dia, HoraInicio, HoraFin, NumServidores, DiaCalendario, Id) " _
                & "VALUES(" & CStr(gCodBanco) & ", " & CStr(gCodSucur) & ", " _
                & Day(drpCalendario.Value) & ", " & cbxHora(0) & ", " & cbxHora(1) & ", " & txtSimu & ",#" &
                Format(drpCalendario.Value, "mm/dd/yyyy") & "#, " & gIdSimulacion & ");"

            'Configurando el día seleccionado en caso
            'sea mayor a 15
            Dia = Day(drpCalendario.Value)
            frmSimulacion.gDiaReal = Dia
            If Dia > 15 Then
                If Dia = 31 Then
                    Dia = 15
                Else
                    Dia = Dia - 15
                End If
            End If

            dblocal.Execute "DELETE * " _
                & "FROM ClienteAtendido " _

```



```

& "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " AND Dia = " & Dia

dblocal.Close
Set dblocal = Nothing

Screen.MousePointer = 11
frmSimulacion.gintDia = Dia
frmSimulacion.gintNumcajeros = Val(txtSimu)
frmSimulacion.gPerfilTransacc = cbxHora(2).ListIndex

Load frmSimulacion
frmSimulacion.Show

frmParametros.Hide
End If
Else
Unload frmParametros
frmSeleccionBanco.Show
End If
End Sub
Private Sub drpCalendario_Change()
    blnCambioFecha = True
End Sub
Private Sub drpCalendario_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub drpCalendario_LostFocus()
    Dim i%

    If blnCambioFecha Then
        If Weekday(drpCalendario.Value) = 7 Then
            cbxHora(0).Clear
            cbxHora(1).Clear
            For i = 8 To 12
                cbxHora(0).AddItem CStr(i)
            Next i
            cbxHora(0).ListIndex = 0

            For i = 8 To 12
                cbxHora(1).AddItem CStr(i)
            Next i
            cbxHora(1).ListIndex = 1
        Else
            cbxHora(0).Clear
            cbxHora(1).Clear
            For i = 8 To 17
                cbxHora(0).AddItem CStr(i)
            Next i
            cbxHora(0).ListIndex = 0

            For i = 8 To 17
                cbxHora(1).AddItem CStr(i)
            Next i
            cbxHora(1).ListIndex = 1
        End If

        frmParametros.Refresh
        cmbSimu(0).Refresh
        cmbSimu(1).Refresh
        DoEvents

        blnCambioFecha = False
    End If
End Sub
Private Sub drpCalendario_Validate(Cancel As Boolean)
    If Weekday(drpCalendario.Value) = 1 Then
        MsgBox "La simulación no se puede efectuar para un día Domingo", vbInformation, MSGVALI
        drpCalendario.SetFocus
        Cancel = True
    End If
End Sub

```

```

End Sub
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode = 27 Then
        Unload frmParametros
    End If
End Sub
Private Sub Form_Load()
    Dim i&
    Dim rstInfo As ADODB.Recordset

    blnCambioFecha = False
    drpCalendario.Value = Date
    If Weekday(drpCalendario.Value) = 1 Then
        drpCalendario.Value = drpCalendario.Value + 1

        cbxHora(0).Clear
        cbxHora(1).Clear
        For i = 8 To 17
            cbxHora(0).AddItem CStr(i)
        Next i
        cbxHora(0).ListIndex = 0

        For i = 8 To 17
            cbxHora(1).AddItem CStr(i)
        Next i
        cbxHora(1).ListIndex = 1
    Else
        If Weekday(drpCalendario.Value) = 7 Then
            cbxHora(0).Clear
            cbxHora(1).Clear
            For i = 8 To 12
                cbxHora(0).AddItem CStr(i)
            Next i
            cbxHora(0).ListIndex = 0

            For i = 8 To 12
                cbxHora(1).AddItem CStr(i)
            Next i
            cbxHora(1).ListIndex = 1
        Else
            cbxHora(0).Clear
            cbxHora(1).Clear
            For i = 8 To 17
                cbxHora(0).AddItem CStr(i)
            Next i
            cbxHora(0).ListIndex = 0

            For i = 8 To 17
                cbxHora(1).AddItem CStr(i)
            Next i
            cbxHora(1).ListIndex = 1
        End If
    End If

    With cbxHora(2)
        .AddItem "(Ninguna)"
        .AddItem "Por mayor frecuencia"
        .ListIndex = 0
    End With

    If gblnSimulacionAnterior Then
        'Mostrar datos de la última generación
        Set rstInfo = New ADODB.Recordset
        rstInfo.Open "SELECT * FROM GeneracionEncabezado " _
            & "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur), gCn, adOpenStatic,
            adLockOptimistic, adCmdText
        If Not rstInfo.EOF Then
            drpCalendario.Value = rstInfo!DiaCalendario

            If Weekday(drpCalendario.Value) = 1 Then

```

```

drpCalendario.Value = drpCalendario.Value + 1

cbxHora(0).Clear
cbxHora(1).Clear
For i = 8 To 17
  cbxHora(0).AddItem CStr(i)
Next i
cbxHora(0).ListIndex = 0

For i = 8 To 17
  cbxHora(1).AddItem CStr(i)
Next i
cbxHora(1).ListIndex = 1
Else
If Weekday(drpCalendario.Value) = 7 Then
  cbxHora(0).Clear
  cbxHora(1).Clear
  For i = 8 To 12
    cbxHora(0).AddItem CStr(i)
  Next i
  cbxHora(0).ListIndex = 0

  For i = 8 To 12
    cbxHora(1).AddItem CStr(i)
  Next i
  cbxHora(1).ListIndex = 1
Else
  cbxHora(0).Clear
  cbxHora(1).Clear
  For i = 8 To 17
    cbxHora(0).AddItem CStr(i)
  Next i
  cbxHora(0).ListIndex = 0

  For i = 8 To 17
    cbxHora(1).AddItem CStr(i)
  Next i
  cbxHora(1).ListIndex = 1
End If
End If

txtSimu = Trim(CStr(rstInfo!NumServidores))
End If
procCerrarRecordset rstInfo
End If
End Sub
Private Sub Form_Unload(Cancel As Integer)
  gblnSimulacionAnterior = False
End Sub
Private Sub txtSimu_GotFocus()
  If txtSimu = "0" Then
    txtSimu = ""
  End If
End Sub
Private Sub txtSimu_KeyDown(KeyCode As Integer, Shift As Integer)
  HacerTab KeyCode
End Sub
Private Sub txtSimu_KeyPress(KeyAscii As Integer)
  If (Chr(KeyAscii) >= "0" And Chr(KeyAscii) <= "9") Then
    'Ok
  Else
    If KeyAscii >= 32 Or KeyAscii = 13 Then KeyAscii = 0
  End If
End Sub
Private Sub txtSimu_LostFocus()
  If Trim(txtSimu) = "" Then
    txtSimu = "0"
  End If
  If Not IsNumeric(txtSimu) Then
    MsgBox "El campo no posee un valor numérico", vbInformation, MSGVALI
  End If
End Sub

```

```

    txtSimu.SetFocus
    Exit Sub
End If
txtSimu = Trim(CStr(Val(txtSimu)))
End Sub

```

### **FrmInicioSimulacion**

```

Option Explicit

Private Sub cmbSimu_Click(Index As Integer)
    If Index = 0 Then
        Load frmParametros
        frmParametros.Show
        Unload frmInicioSimulacion
    Else
        Unload frmInicioSimulacion
        gblnSimulacionAnterior = False
        frmSeleccionBanco.Show
    End If
End Sub

Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode = 27 Then
        Unload frmInicioSimulacion
        frmSeleccionBanco.Show
    End If
End Sub

```

### **FrmPress**

```

Option Explicit
'Pantalla de presentación
'inicial
Private Sub Form_Activate()
    Espera 1
End Sub

Private Sub Timer1_Timer()
    frmPress.Hide
    Load frmSeleccionBanco
    frmSeleccionBanco.Show
End Sub

```

### **FrmProgreso**

```

Option Explicit
Public gTipo As Byte
Private Sub Form_Activate()
    Select Case gTipo
        Case 0
            GrabarInformacionDeNuevoBanco
    End Select
    Unload frmProgreso
End Sub

Private Sub Form_Load()
    Espera
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Espera 1
End Sub

```

## **FrmRepcionDeDatos**

Option Explicit

Public gTipo As Byte  
Public gLngIdBanco&

```
Function HoraYaIntroducida(ByRef lsvLista As ListView, ByVal Dia As Integer, ByVal Hora As Integer) As Boolean
    Dim i&
    HoraYaIntroducida = False
    For i = 1 To lsvLista.ListItems.Count
        If CInt(lsvLista.ListItems(i).ListSubItems(1)) = Dia And CInt(lsvLista.ListItems(i).ListSubItems(2)) = Hora Then
            HoraYaIntroducida = True
            Exit For
        End If
    Next i
End Function
```

```
Function Validar(ByVal Tipo As Integer) As Boolean
```

```
    Validar = False
    Select Case Tipo
        'Colas
        Case 0 'Entrada
            If Trim(txtSimu(2)) = "" Then
                MsgBox "Debe digitar el día", vbInformation, MSGVALI
                txtSimu(2).SetFocus
                Exit Function
            End If
            If Trim(txtSimu(3)) = "" Then
                MsgBox "Debe digitar la hora", vbInformation, MSGVALI
                txtSimu(3).SetFocus
                Exit Function
            End If
            If mskSimu(3).Text = "__:_" Then
                MsgBox "Debe digitar el tiempo en minutos/segundos", vbInformation, MSGVALI
                mskSimu(3).SetFocus
                Exit Function
            End If

            If Val(txtSimu(2)) < 1 Or Val(txtSimu(2)) > 31 Then
                MsgBox "Día debe estar en el rango de 1 y 30 días", vbInformation, MSGVALI
                txtSimu(2).SetFocus
                Exit Function
            End If
            If Val(txtSimu(3)) < 0 Or Val(txtSimu(3)) > 24 Then
                MsgBox "Día debe estar en el rango de 0 y 24 horas", vbInformation, MSGVALI
                txtSimu(3).SetFocus
                Exit Function
            End If
        End If
```

```
        Case 1 'Salida
```

```
            If Trim(txtSimu(5)) = "" Then
                MsgBox "Debe digitar el día", vbInformation, MSGVALI
                txtSimu(5).SetFocus
                Exit Function
            End If
            If Trim(txtSimu(6)) = "" Then
                MsgBox "Debe digitar la hora", vbInformation, MSGVALI
                txtSimu(6).SetFocus
                Exit Function
            End If
            If mskSimu(4).Text = "__:_" Then
                MsgBox "Debe digitar el tiempo en minutos/segundos", vbInformation, MSGVALI
                mskSimu(4).SetFocus
                Exit Function
            End If

            If Val(txtSimu(5)) < 1 Or Val(txtSimu(5)) > 31 Then
                MsgBox "Día debe estar en el rango de 1 y 30 días", vbInformation, MSGVALI
                txtSimu(5).SetFocus
            End If
```

```

        Exit Function
    End If
    If Val(txtSimu(6)) < 0 Or Val(txtSimu(6)) > 24 Then
        MsgBox "Día debe estar en el rango de 0 y 24 horas", vbInformation, MSGVALI
        txtSimu(6).SetFocus
        Exit Function
    End If

    'Transacciones
    Case 4
        If Trim(txtSimu(14)) = "" Then
            MsgBox "Debe digitar el nombre de la transacción", vbInformation, MSGVALI
            txtSimu(14).SetFocus
            Exit Function
        End If
        If Trim(mskSimu(15)) = "__:_" Then
            MsgBox "Debe digitar el tiempo en minutos/segundos", vbInformation, MSGVALI
            mskSimu(15).SetFocus
            Exit Function
        End If

    End Select

    Validar = True
End Function
Sub MostrarFrame(ByVal Indice As Integer)
    Dim i%

    For i = Frame2.LBound To Frame2.UBound
        Frame2(i).Visible = False
    Next i

    Frame2(Indice).ZOrder
    Frame2(Indice).Top = 2640
    Frame2(Indice).Left = 2520
    Frame2(Indice).Visible = True
End Sub
Private Sub cmbAdd_Click(Index As Integer)
    Dim itm As ListItem

    Select Case Index
    Case 0
        If Validar(0) Then
            If HoraYaIntroducida(lsvTbl(0), CInt(txtSimu(2)), CInt(txtSimu(3))) Then
                MsgBox "No se puede introducir una hora más de una vez en un mismo día", vbCritical, MSGVALI
                txtSimu(2).SetFocus
                Exit Sub
            End If
            Set itm = lsvTbl(0).ListItems.Add(lsvTbl(0).ListItems.Count + 1, , , 0)
            itm.SubItems(1) = txtSimu(2)
            itm.SubItems(2) = txtSimu(3)
            itm.SubItems(3) = mskSimu(3)
            itm.SubItems(4) = txtSimu(4)
            txtSimu(2) = ""
            txtSimu(3) = ""
            mskSimu(3) = "__:_"
            txtSimu(4) = "0"
            txtSimu(2).SetFocus
        End If

    Case 1
        If Validar(1) Then
            If HoraYaIntroducida(lsvTbl(1), CInt(txtSimu(5)), CInt(txtSimu(6))) Then
                MsgBox "No se puede introducir una hora más de una vez en un mismo día", vbCritical, MSGVALI
                txtSimu(5).SetFocus
                Exit Sub
            End If
            Set itm = lsvTbl(1).ListItems.Add(lsvTbl(1).ListItems.Count + 1, , , 0)
            itm.SubItems(1) = txtSimu(5)
            itm.SubItems(2) = txtSimu(6)
        End If
    End Select
End Sub

```

```

        itm.SubItems(3) = mskSimu(4)
        itm.SubItems(4) = txtSimu(7)
        txtSimu(5) = ""
        txtSimu(6) = ""
        mskSimu(4) = "__:_"
        txtSimu(7) = "0"
        txtSimu(5).SetFocus
    End If

Case 4
    If Validar(4) Then
        Set itm = lsvTbl(4).ListItems.Add(lsvTbl(4).ListItems.Count + 1, , txtSimu(14), 0)
        itm.SubItems(1) = mskSimu(15)
        itm.SubItems(2) = txtSimu(15)
        itm.SubItems(3) = IIf(Trim(txtSimu(16)) = "", 0, txtSimu(16))
        txtSimu(14) = ""
        mskSimu(15) = "__:_"
        txtSimu(15) = "0"
        txtSimu(16) = ""
        txtSimu(14).SetFocus
    End If

End Select
End Sub
Private Sub cmbSimu_Click(Index As Integer)
    Select Case Index
        Case 0
            Select Case True
                Case Frame2(1).Visible
                    MostrarFrame 0
                    cmbSimu(0).Enabled = False
                    cmbSimu(1).Enabled = True
                    cmbSimu(4).Enabled = False

            End Select

        Case 1
            Select Case True
                Case Frame2(0).Visible
                    MostrarFrame 1
                    cmbSimu(0).Enabled = True

                Case Frame2(1).Visible
                    MostrarFrame 2
                    cmbSimu(1).Enabled = False
                    cmbSimu(4).Enabled = True

            End Select
            cmbSimu(4).Enabled = True
            cmbSimu(1).Enabled = False

        Case 2
            If MsgBox("Si cierra esta ventana perderá la información introducida. Desea continuar?", vbQuestion + vbYesNo,
                MSGCONF) = vbYes Then
                Unload frmRecepcionDeDatos
                frmSeleccionBanco.Show
            End If

        Case 3
            Load frmImportar
            frmImportar.Show 1

        Case 4
            If Frame2(1).Visible Then
                'Validar información y grabar
                txtSimu(0) = Trim(txtSimu(0))
                txtSimu(1) = Trim(txtSimu(1))

                If txtSimu(0) = "" Then
                    MsgBox "Debe introducir el nombre del banco", vbInformation, MSGVALI
                End If
            End If
    End Select
End Sub

```

```

        txtSimu(0).SetFocus
    Exit Sub
End If
If txtSimu(1) = "" Then
    MsgBox "Debe introducir el nombre de la sucursal", vbInformation, MSGVALI
    txtSimu(1).SetFocus
    Exit Sub
End If

If lsvTbl(0).ListItems.Count = 0 Then
    MsgBox "Debe introducir datos de entrada de cola", vbInformation, MSGVALI
    cmbSimu_Click 0
    cmbSimu_Click 0
    txtSimu(3).SetFocus
Else
    If lsvTbl(1).ListItems.Count = 0 Then
        MsgBox "Debe introducir datos de salida de cola", vbInformation, MSGVALI
        cmbSimu_Click 0
        cmbSimu_Click 0
        txtSimu(6).SetFocus
    Else
        If lsvTbl(4).ListItems.Count = 0 Then
            MsgBox "Debe introducir las transacciones del banco", vbInformation, MSGVALI
            txtSimu(14).SetFocus
        Else
            gstrNombreBanco = txtSimu(0)
            gstrNombreSucursal = txtSimu(1)
            frmProgreso.gTipo = 0
            Load frmProgreso
            frmProgreso.Show 1
            Unload frmRecepcionDeDatos
            Load frmInicioSimulacion
            frmInicioSimulacion.Show
        End If
    End If
End If
End If
End If
End Select
End Sub
Private Sub Form_Activate()
    Espera 1
End Sub

Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode = 27 Then
        Unload frmRecepcionDeDatos
        frmSeleccionBanco.Show
    End If
End Sub
Private Sub Form_Load()
    Espera
    Select Case gTipo
        Case 0
            frmRecepcionDeDatos.Caption = "Creando banco"
        Case 1
            frmRecepcionDeDatos.Caption = "Creando sucursal"
            txtSimu(0) = frmSeleccionBanco.cbxBanco
            txtSimu(0).Enabled = False
        Case 2
            txtSimu(0) = gstrNombreBanco
            txtSimu(0).Enabled = False
            txtSimu(1) = gstrNombreSucursal
            txtSimu(1).Enabled = False
            frmRecepcionDeDatos.Caption = "Capturando nuevos datos"
    End Select
    MostrarFrame 0
End Sub

```



```

Private Sub mskSimu_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub mskSimu_Validate(Index As Integer, Cancel As Boolean)
    Dim intTiempo As Long

    If mskSimu(Index).ClipText <> "" Then
        If Not fncHoraValida("00:" & mskSimu(Index)) Then
            MsgBox "Hora no válida", vbExclamation, MSGVALI
            Cancel = True
            Select Case Index
                Case 0
                    txtSimu(10) = "0"
                Case 1
                    txtSimu(13) = "0"
                Case 3
                    txtSimu(4) = "0"
                Case 4
                    txtSimu(7) = "0"
                Case 15
                    txtSimu(15) = "0"
            End Select
            mskSimu(Index).SetFocus
        Else
            mskSimu(Index) = fncFORMATO_HORAMMSS(dtpHora.Value)
            intTiempo = Minute(dtpHora.Value) * 60 + Second(dtpHora.Value)
            Select Case Index
                Case 0
                    txtSimu(10) = intTiempo
                Case 1
                    txtSimu(13) = intTiempo
                Case 3
                    txtSimu(4) = intTiempo
                Case 4
                    txtSimu(7) = intTiempo
                Case 15
                    txtSimu(15) = intTiempo
            End Select
        End If
    End If
End Sub
Private Sub txtSimu_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub txtSimu_KeyPress(Index As Integer, KeyAscii As Integer)
    Select Case Index
        Case 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13
            If (Chr(KeyAscii) >= "0" And Chr(KeyAscii) <= "9") Then
                'Ok
            Else
                If KeyAscii >= 32 Or KeyAscii = 13 Then KeyAscii = 0
            End If
        Case Else
            procConvMayusculas KeyAscii
    End Select
End Sub
Private Sub txtSimu_LostFocus(Index As Integer)
    Select Case Index
        Case 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13
            If Not IsNumeric(txtSimu(Index)) And Trim(txtSimu(Index)) <> "" Then
                MsgBox "El campo no posee un valor numérico", vbInformation, MSGVALI
                txtSimu(Index).SetFocus
                Exit Sub
            End If
    End Select
End Sub

```

### **FrmReportes**

## Option Explicit

```
Sub ProcPromedios(ByVal intTipo As Integer)
    Dim db_local As DAO.Database
    Dim rstInfo As DAO.Recordset
    Dim rstDatosGraficos As DAO.Recordset

    Set db_local = OpenDatabase(App.Path & "\\VeVsSimuColas.mdb", False, False)
    If intTipo = 0 Then
        db_local.Execute "DELETE * FROM _PromediosCola;"

        Set rstDatosGraficos = db_local.OpenRecordset("SELECT Hora, VPromEntCola, VPromSalCola FROM _PromediosCola",
        dbOpenDynaset)
        Set rstInfo = db_local.OpenRecordset("SELECT * " _
        & "FROM Qry_PromedioEntradaCola " _
        & "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" &
        frmSimulacion.lblFecha.Tag & "#", dbOpenSnapshot)
        While Not rstInfo.EOF
            rstDatosGraficos.FindFirst "Hora = " & rstInfo!Hora & ""
            If rstDatosGraficos.NoMatch Then
                rstDatosGraficos.AddNew
                rstDatosGraficos!Hora = rstInfo!Hora
                rstDatosGraficos!VPromEntCola = Round(rstInfo!PromedioDeVEntCola, 0)
                rstDatosGraficos!VPromSalCola = 0
            End If
            rstDatosGraficos.Update
            rstInfo.MoveNext
        Wend

        Set rstInfo = db_local.OpenRecordset("SELECT * " _
        & "FROM Qry_PromedioSalidaCola " _
        & "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" &
        frmSimulacion.lblFecha.Tag & "#", dbOpenSnapshot)
        While Not rstInfo.EOF
            rstDatosGraficos.FindFirst "Hora = " & rstInfo!Hora & ""
            If rstDatosGraficos.NoMatch Then
                rstDatosGraficos.AddNew
                rstDatosGraficos!Hora = rstInfo!Hora
                rstDatosGraficos!VPromEntCola = 0
                rstDatosGraficos!VPromSalCola = Round(rstInfo!PromedioDeVSalCola, 0)
            Else
                rstDatosGraficos.Edit
                rstDatosGraficos!VPromSalCola = rstDatosGraficos!VPromSalCola + Round(rstInfo!PromedioDeVSalCola, 0)
            End If
            rstDatosGraficos.Update
            rstInfo.MoveNext
        Wend
    Else
        db_local.Execute "DELETE * FROM _PromediosServicio;"

        Set rstDatosGraficos = db_local.OpenRecordset("SELECT Hora, VPromEntServ, VPromSalServ FROM _PromediosServicio",
        dbOpenDynaset)
        Set rstInfo = db_local.OpenRecordset("SELECT * " _
        & "FROM Qry_PromedioEntradaServ " _
        & "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" &
        frmSimulacion.lblFecha.Tag & "#", dbOpenSnapshot)
        While Not rstInfo.EOF
            rstDatosGraficos.FindFirst "Hora = " & rstInfo!Hora & ""
            If rstDatosGraficos.NoMatch Then
                rstDatosGraficos.AddNew
                rstDatosGraficos!Hora = rstInfo!Hora
                rstDatosGraficos!VPromEntServ = Round(rstInfo!PromedioDeVEntServ, 0)
                rstDatosGraficos!VPromSalServ = 0
            End If
            rstDatosGraficos.Update
            rstInfo.MoveNext
        Wend

        Set rstInfo = db_local.OpenRecordset("SELECT * " _
        & "FROM Qry_PromedioSalidaServ " _
```

```

& "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" &
frmSimulacion.lblFecha.Tag & "#", dbOpenSnapshot)
While Not rstInfo.EOF
rstDatosGraficos.FindFirst "Hora = " & rstInfo!Hora & ""
If rstDatosGraficos.NoMatch Then
rstDatosGraficos.AddNew
rstDatosGraficos!Hora = rstInfo!Hora
rstDatosGraficos!VPromEntServ = 0
rstDatosGraficos!VPromSalServ = Round(rstInfo!PromedioDeVSalServ, 0)
Else
rstDatosGraficos.Edit
rstDatosGraficos!VPromSalServ = rstDatosGraficos!VPromSalServ + Round(rstInfo!PromedioDeVSalServ, 0)
End If
rstDatosGraficos.Update
rstInfo.MoveNext
Wend
End If

db_local.Close
Set db_local = Nothing
End Sub
Private Sub cmbSimu_Click(Index As Integer)
If Index = 0 Then
Select Case List1.ListIndex
Case 0
ProcPromedios 0
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repPromedioEntSalCola.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""
.Formulas(1) = "SUCURSAL=" & gstrNombreSucursal & ""

.Action = 1
End With

Case 1
ProcPromedios 1
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repPromedioEntSalServ.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""
.Formulas(1) = "SUCURSAL=" & gstrNombreSucursal & ""

.Action = 1
End With

Case 2
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repEntradaSalidaPorHora.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""
.Formulas(1) = "SUCURSAL=" & gstrNombreSucursal & ""

.Action = 1
End With

Case 3
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repCantPersonasCajero.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""

```

```

.Formulas(1) = "SUCURSAL=" & gstrNombreSucursal & ""
.SelectionFormula = "{PersonasPorCajero.Bco_Codigo}=" & gCodBanco & " AND " & _
    "{PersonasPorCajero.Suc_Codigo}=" & gCodSucur

.Action = 1
End With

Case 4
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repCantPersonasCajeroPorHora.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""
.Formulas(1) = "SUCURSAL=" & gstrNombreSucursal & ""
.SelectionFormula = "{PersonasPorCajeroFto2.Bco_Codigo}=" & gCodBanco & " AND " & _
    "{PersonasPorCajeroFto2.Suc_Codigo}=" & gCodSucur

.Action = 1
End With

Case 5
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repTransaccionesPorCajero.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""
.Formulas(1) = "SUCURSAL=" & gstrNombreSucursal & ""
.SelectionFormula = "{QryTransaccionesPorCajero.Bco_Codigo}=" & gCodBanco & " AND " & _
    "{QryTransaccionesPorCajero.Suc_Codigo}=" & gCodSucur & " AND " & _
    "{QryTransaccionesPorCajero.Dia}=" & frmSimulacion.lblDia
.Action = 1
End With

Case 6
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repTransacciones.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""
.SelectionFormula = "{PromTrans.Bco_Codigo}=" & gCodBanco
.Action = 1
End With

Case 7
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repTiempoSistemaPersonasSelecc.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""
.Formulas(1) = "SUCURSAL=" & gstrNombreSucursal & ""
.Action = 1
End With

Case 8
With CrystalReport1
.Reset
.WindowState = crptMaximized
.ReportFileName = App.Path & "\repTransaccionesPorHora.rpt"
.WindowTitle = List1
.Formulas(0) = "BANCO=" & gstrNombreBanco & ""
.Formulas(1) = "SUCURSAL=" & gstrNombreSucursal & ""
.Action = 1
End With
End Select
Else
Unload frmReportes

```

```

    If Not frmResultados.Visible Then
        frmSimulacion.Show
    End If
End If
End Sub
Private Sub Form_Load()
    If frmSimulacion.Visible Then
        frmSimulacion.Hide
    End If
    With List1
        .AddItem "Promedio de Entrada/Salida en cola"
        .AddItem "Promedio de Entrada/Salida en servicio"
        .AddItem "Cantidad de personas que entran por hora y salen"
        .AddItem "Cantidad de personas atendidas por cajero"
        .AddItem "Cantidad de personas atendidas por cajero por hora"
        .AddItem "Cantidad de transacciones por cajero"
        .AddItem "Tipos de transacciones"
        .AddItem "Tiempo en sistema de personas seleccionadas"
        .AddItem "Transacciones por hora"
        .ListIndex = 0
    End With
End Sub

Private Sub List1_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub

```

## **FrmResultados**

Option Explicit

```

Private Sub cmbResultados_Click(Index As Integer)
    Select Case Index
        Case 0
            With CrystalReport1
                .ReportFileName = App.Path & "\repResultados.rpt"
                .Formulas(0) = "DIA=" & lblFecha & ""
                .Formulas(1) = "HORAINICIAL=" & lblHoraIni & ""
                .Formulas(2) = "HORAFINAL=" & lblHoraFin & ""
                .Formulas(3) = "CANTENTRAN=" & lblCantEntrado & ""
                .Formulas(4) = "CANTATENDIDAS=" & lblCantSalida & ""
                .Formulas(5) = "SERVIDORES=" & lblCantServ & ""
                .Formulas(6) = "DESERCIONES=" & lblNumDeserciones & ""
                .Formulas(7) = "TRANSACCIONES=" & lblNumTransacciones & ""
                .Formulas(8) = "BANCO=" & gstrNombreBanco & ""
                .Formulas(9) = "SUCURSAL=" & gstrNombreSucursal & ""
                .Formulas(10) = "CANTENCOLA=" & CInt(lblCantEntrado) - CInt(lblCantSalida) & ""
                .Action = 1
            End With

        Case 1
            Grabar resultados de la simulacion
            gCn.Execute "UPDATE GeneracionEncabezado " _
                & "SET NumServidores=" & lblCantServ & ", " _
                & "CantPersonasEntran=" & lblCantEntrado & ", " _
                & "CantPersonasSalen=" & lblCantSalida & ", " _
                & "Deserciones=" & lblNumDeserciones & ", " _
                & "CantTransacciones=" & lblNumTransacciones & " " _
                & "WHERE Bco_Codigo=" & CStr(gCodBanco) & " AND " _
                & "Suc_Codigo=" & CStr(gCodSucur) & " AND " _
                & "DiaCalendario=#" & frmSimulacion.lblFecha.Tag & "# AND " _
                & "Id=" & gIdSimulacion & ";"
            MsgBox "La informacion se grabo existosamente", vbInformation, MSGINFO
            frmSimulacion.mnuComparacion.Enabled = True
            frmSimulacion.gGrabadaSimulacion = True

        Case 2
            Load frmReportes
    End Select
End Sub

```

```

    frmReportes.Show 1

    Case 3
        Unload frmResultados
        frmSimulacion.Show
    End Select
End Sub
Private Sub Form_Load()
    On Error GoTo ErrHandler

    frmSimulacion.Hide
    lblDia = frmSimulacion.lblDia
    lblHoraIni = frmSimulacion.lblHoraIni
    lblHoraFin = frmSimulacion.lblHoraFin

    lblFecha = frmSimulacion.lblFecha
    lblCantEntrado = frmSimulacion.lblCantEntrado
    lblCantSalida = frmSimulacion.lblCantSalida
    lblCantServ = frmSimulacion.lblCantServ
    lblNumDeserciones = frmSimulacion.lblDeserciones
    lblNumTransacciones = Format(fncNumeroTransacciones(Val(frmSimulacion.lblDia)), "###,###,###,###,###")
    Exit Sub

ErrHandler:
    Resume Next
End Sub

```

### **FrmSeleccionAnalisis**

```

Option Explicit

Private Sub cmbSimu_Click(Index As Integer)
    Select Case Index
        Case 0
            Unload frmSeleccionAnalisis
            gblnSimulacionAnterior = True
            Load frmInicioSimulacion
            frmInicioSimulacion.Show

        Case 1
            Unload frmSeleccionAnalisis
            gblnSimulacionAnterior = False
            frmRecepcionDeDatos.gTipo = 2
            Load frmRecepcionDeDatos
            frmRecepcionDeDatos.Show

        Case 2
            Unload frmSeleccionAnalisis
            gblnSimulacionAnterior = False
            frmSeleccionBanco.Show
    End Select
End Sub
Private Sub Form_Activate()
    Espera 1
End Sub
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode = 27 Then
        Unload frmSeleccionAnalisis
        frmSeleccionBanco.Show
    End If
End Sub

```

## **FrmSeleccionBanco**

```

Option Explicit
'Procedimiento que muestra los nombres
'de los bancos en el DataCombo de bancos
Sub MostrarBancos()
    Dim rstBancos As New ADODB.Recordset
    rstBancos.Open "SELECT Bco_Nombre FROM Bancos ORDER BY Bco_Nombre;", gCn, adOpenStatic, adLockReadOnly,
adCmdText
    If Not rstBancos.EOF Then
        While Not rstBancos.EOF
            cbxBanco.AddItem rstBancos.Fields(0)
            rstBancos.MoveNext
        Wend
        cbxBanco.ListIndex = 0
    Else
        cmbSimu(1).Enabled = False
        cmbSimu(2).Enabled = False
        cmbSimu(0).Default = True
    End If
    procCerrarRecordset rstBancos
End Sub
'Procedimiento que muestra los nombres
'de las sucursales según el banco seleccionado
Sub MostrarSucursales()
    Dim rstSucursales As New ADODB.Recordset
    cbxSucursal.Clear

    rstSucursales.Open "SELECT Sucursales.Suc_Nombre " _
        & "FROM Bancos INNER JOIN Sucursales ON Bancos.Bco_Codigo = Sucursales.Bco_Codigo " _
        & "Where Bancos.Bco_Nombre = " & cbxBanco & " ORDER BY Sucursales.Suc_Nombre", gCn, adOpenStatic,
adLockReadOnly, adCmdText
    If Not rstSucursales.EOF Then
        cmbSimu(2).Enabled = True
        While Not rstSucursales.EOF
            cbxSucursal.AddItem rstSucursales.Fields(0)
            rstSucursales.MoveNext
        Wend
        cbxSucursal.ListIndex = 0
    Else
        cmbSimu(2).Enabled = False
    End If
    procCerrarRecordset rstSucursales
End Sub
Private Sub cbxBanco_Click()
    MostrarSucursales
End Sub
Private Sub cbxBanco_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub cbxSucursal_KeyDown(KeyCode As Integer, Shift As Integer)
    HacerTab KeyCode
End Sub
Private Sub cmbSimu_Click(Index As Integer)
    Dim rstCodigos As ADODB.Recordset

    Select Case Index
        Case 0, 1
            'Formulario para captura de nuevo banco, nueva sucursal
            'y tablas generales
            If Index = 0 Then
                frmRecepcionDeDatos.gTipo = 0
            Else
                Set rstCodigos = New ADODB.Recordset
                rstCodigos.Open "SELECT Bancos.Bco_Codigo " _
                    & "From Bancos " _
                    & "WHERE Bancos.Bco_Nombre=" & cbxBanco & "", gCn, adOpenStatic, adLockOptimistic, adCmdText
                If Not rstCodigos.EOF Then
                    gCodBanco = rstCodigos.Fields(0)
                End If
            End If
        End Select
    End Sub

```

```

    End If
    procCerrarRecordset rstCodigos
    frmRecepcionDeDatos.gTipo = 1
End If
Load frmRecepcionDeDatos
frmRecepcionDeDatos.Show
frmSeleccionBanco.Hide

Case 2
'Formulario para realizar el análisis
Set rstCodigos = New ADODB.Recordset
rstCodigos.Open "SELECT Bancos.Bco_Codigo " _
    & "From Bancos " _
    & "WHERE Bancos.Bco_Nombre=" & cbxBanco & """, gCn, adOpenStatic, adLockOptimistic, adCmdText
If Not rstCodigos.EOF Then
    gCodBanco = rstCodigos.Fields(0)
End If
procCerrarRecordset rstCodigos

Set rstCodigos = New ADODB.Recordset
rstCodigos.Open "SELECT Sucursales.Suc_Codigo " _
    & "FROM Bancos INNER JOIN Sucursales ON Bancos.Bco_Codigo = Sucursales.Bco_Codigo " _
    & "WHERE Sucursales.Bco_Codigo=" & gCodBanco & " AND Sucursales.Suc_Nombre=" & cbxSucursal & """, gCn,
adOpenStatic, adLockOptimistic, adCmdText
If Not rstCodigos.EOF Then
    gCodSucur = rstCodigos.Fields(0)
End If
procCerrarRecordset rstCodigos

gstrNombreBanco = cbxBanco
gstrNombreSucursal = cbxSucursal
Load frmSeleccionAnalisis
frmSeleccionAnalisis.Show
frmSeleccionBanco.Hide

Case 3
    CerrarSistema
End Select
End Sub
Private Sub Form_Load()
    Unload frmPress
    MostrarBancos
End Sub

```

### **FrmSimulacion**

Option Explicit

'Variables públicas

Public gintDia As Integer

Public gintNumcajeros As Integer

Public gdblVEntCola#

Public gdblVSaCola#

Public gGrabadaSimulacion As Boolean

Public gPerfilTransacc As Byte

Public gDiaReal%

'Variables locales

Dim mdteHoraSistema As Date

Dim mdteHoraSistemaFinal As Date

'Dim arrayCajero(0 To 14) As typeCajero

Dim mblnAgregarClientes As Boolean

Dim mIngCantIngresoUnaHora&

Dim mintPersonasEntranColaUnaHora%

Dim mPromedioIngresoSeisMin%

Dim mintPersonasEntranCola%

Dim mintPersonasSalenCola%

Dim mFinalizarSimulacion As Boolean

Dim mContinuarCasoEspecial As Boolean



```

Dim mmbteImagenAleatoria&
Dim mmlngUltimoVisible&
Dim mlngCantSalidaUnaHora&
Dim mintPersonasSalenColaUnaHora%
Dim mblnCambiarCajeros As Boolean
Dim mblnDesertar As Boolean
Dim mblnCasoEsp As Boolean
Dim mPromedioSalidaSeisMin%
Dim mtmpClientesPasarocajero%
Dim intHoraTemp As Date
Dim TiempoSimuDesercion As Double
Dim TiempoSimuCasoEspecial As Double
Dim dteHoraFiltro1 As Date
Dim dteHoraFiltro2 As Date
Dim db_local As DAO.Database

Public lng_VEntCola&
Public lng_VSalCola&
Public lng_VEntServ&
Public lng_VSalServ&

Sub ClientesCola(ByVal lngClientes As Long)
  Dim i&
  For i = imgCliente.LBound To imgCliente.UBound
    If lngClientes > 0 Then
      imgCliente(i).Visible = True
      lngClientes = lngClientes - 1
    Else
      imgCliente(i).Visible = False
    End If
  Next i
End Sub

Function SimulaciónFinalizada() As Boolean
  Dim rstComprob As DAO.Recordset
  Dim i%

  If lblHoraFin <> "05:00 PM" Then
    'Hora de finalización antes del final del día
    If Format(mdteHoraSistema, "hh:mm AM/PM") = lblHoraFin Then
      SimulaciónFinalizada = True
    End If
  Else
    If mdteHoraSistemaFinal >= #5:00:00 PM# Then
      'Hora de finalización al final del día
      Set rstComprob = db_local.OpenRecordset("SELECT Bco_Codigo, HanSalidoDeCola, Desercion " _
        & "From CajerosTiemposES " _
        & "WHERE HanSalidoDeCola=False AND Desercion=False AND " _
        & "Bco_Codigo=" & gCodBanco & " AND " _
        & "Suc_Codigo=" & gCodSucur & " AND Dia=" & lblDia & " AND " _
        & "Fecha=#" & lblFecha.Tag & "#;", dbOpenSnapshot)
      If Not rstComprob.EOF Then
        If IsNull(rstComprob!Bco_Codigo) Then
          For i = imgAtendCajero.LBound To imgAtendCajero.UBound
            imgAtendCajero(i).Visible = False
          Next i
          SimulaciónFinalizada = True
        End If
      Else
        For i = imgAtendCajero.LBound To imgAtendCajero.UBound
          imgAtendCajero(i).Visible = False
        Next i
        SimulaciónFinalizada = True
      End If
      rstComprob.Close
    End If
  End If
End Function

Sub CambiarHoraAuxiliar()
  db_local.Execute "UPDATE CajerosTiemposES " _
    & "SET HoraAuxiliar = " & Format(mdteHoraSistema, "hh:mm:ss") & " " _

```

```

    & "WHERE DebenSalirDeCola=False AND HanSalidoDeCola=FALSE AND Hora=" & Hour(mdteHoraSistema) - 1
End Sub
Function fncIdUltimaPersonaNoHaEntrado() As Long
    Dim i&
    For i = 0 To 250000
        If Not gArregloClientes(i).EntroACola Then
            fncIdUltimaPersonaNoHaEntrado = i
            Exit For
        End If
    Next i
End Function
Sub MarcarLasPersonasQueSaldranEnUnaHora()
    Dim lngCantPersonasSalenUnaHora&
    Dim rstMarcar As DAO.Recordset
    Dim strHoraIni$
    Dim strHoraFin$

    'Inicio de una hora, calcular la velocidad de salida de cola
    'promedio en base a la relación del número de cajeros
    lngCantPersonasSalenUnaHora = Round(360000 / fncPromedioSalCola(mdteHoraSistema, CInt(lblDia)) * 10, 0) *
    (CInt(lblCantServ) / 5)

    'Marcar el número de personas que saldrán en la hora actual
    strHoraIni = Left(Format(mdteHoraSistema, "HH:MM"), 2) & ":00:00"
    strHoraFin = Left(Format(mdteHoraSistema, "HH:MM"), 2) & ":59:59"

    Set rstMarcar = db_local.OpenRecordset("SELECT HanSalidoDeCola, DebenSalirDeCola, Desercion " _
    & "From CajerosTiemposES " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND Dia=" & lblDia & " AND " _
    & "Fecha=#" & lblFecha.Tag & "# AND " _
    & "HoraAuxiliar Between #12/30/1899 " & strHoraIni & "# And #12/30/1899 " & strHoraFin & "# AND Desercion = False " _
    & "ORDER BY NumCliente", dbOpenDynaset)

    With rstMarcar
        If Format(mdteHoraSistema, "hh:mm AM/PM") = lblHoraFin Then
            While Not .EOF
                If !Desercion = False Then
                    .Edit
                    !DebenSalirDeCola = True
                    .Update
                End If
                .MoveNext
            Wend
        Else
            While Not .EOF
                If lngCantPersonasSalenUnaHora > 0 Then
                    .Edit
                    !DebenSalirDeCola = True
                    .Update
                Else
                    If !HanSalidoDeCola = False Then
                        .Edit
                        !DebenSalirDeCola = False
                        .Update
                    End If
                End If
                lngCantPersonasSalenUnaHora = lngCantPersonasSalenUnaHora - 1

                .MoveNext
            Wend
        End If
        .Close
    End With
    Set rstMarcar = Nothing
End Sub
Function PrimeraPersonaNoVisible() As Variant
    Dim i&

    PrimeraPersonaNoVisible = Null

```

```

For i = imgCliente.LBound To imgCliente.UBound
  If Not imgCliente(i).Visible Then
    PrimeraPersonaNoVisible = i
  Exit For
End If
Next i
End Function
Function UltimaPersonaVisible() As Variant
  Dim i&

  UltimaPersonaVisible = Null
  For i = imgCliente.LBound To imgCliente.UBound
    If Not imgCliente(i).Visible Then
      If i - 1 >= 0 Then
        UltimaPersonaVisible = i - 1
      Else
        Exit For
      End If
    End If
  Next i

  UltimaPersonaVisible = Null
  For i = imgCliente.LBound To imgCliente.UBound
    If imgCliente(i).Visible Then
      UltimaPersonaVisible = i
    End If
  Next i

End Function
***** PROCEDIMIENTOS *****
Function fncSoloUnCajero() As Integer
  Dim i%

  fncSoloUnCajero = 0
  For i = imgCajeros.LBound To imgCajeros.UBound
    If imgCajeros(i).Visible Then
      fncSoloUnCajero = fncSoloUnCajero + 1
    End If
  Next i
End Function
Sub ProcAdi_QuitarCajero(ByVal blnAdicionar As Boolean)
  Dim i%

  If blnAdicionar Then
    For i = imgCajeros.LBound To imgCajeros.UBound
      If Not imgCajeros(i).Visible Then
        imgCajeros(i).Visible = True
        imgAtendCajero(i).Visible = False
        arrayCajero(i).Habilitado = True
        arrayCajero(i).Ocupado = False
        arrayCajero(i).NumCaja = i
      Exit For
    End If
  Next i
Else
    For i = imgCajeros.LBound To imgCajeros.UBound
      If imgCajeros(i).Visible Then
        imgCajeros(i).Visible = False
        imgAtendCajero(i).Visible = False
        arrayCajero(i).Habilitado = False
        arrayCajero(i).Ocupado = False
        arrayCajero(i).NumCaja = 0
      Exit For
    End If
  Next i
End If
DoEvents

```

```

End Sub
Sub procReiniciarCajeros()
    Dim i%
    For i = 0 To 14
        'Liberar cajeros
        arrayCajero(i).Ocupado = False
        arrayCajero(i).Habilitado = False
        arrayCajero(i).Limite = 0
        arrayCajero(i).Contador = 0

        'Liberar clientes
        imgAtendCajero(i).Visible = False
        imgAtendCajero(i).Picture = LoadResPicture(102, vbResBitmap)
        imgAtendCajero(i).Tag = "N"

    Next i
End Sub
Sub procLineasDeFrente()
    Dim i%
    For i = Line1.LBound To Line1.UBound
        Line1(i).ZOrder
    Next i
End Sub
Sub procOcultarClientesCola()
    Dim i%
    For i = imgCliente.LBound To imgCliente.UBound
        imgCliente(i).Visible = False
    Next i
End Sub
Sub procOcultarClientesEnServicio()
    Dim i%
    For i = imgAtendCajero.LBound To imgAtendCajero.UBound
        imgAtendCajero(i).Visible = False
    Next i
End Sub
Sub procAumentarMinutos(ByVal intNumMinutos As Integer)
    If Not lblHoraFin = Format(mdteHoraSistema, "hh:mm AM/PM") Then
        'Adicionar x minutos a la hora
        mdteHoraSistema = DateAdd("n", intNumMinutos, mdteHoraSistema)
    Else
        db_local.Execute "UPDATE CajerosTiemposES SET DebenSalirDeCola = TRUE;"
    End If
    mdteHoraSistemaFinal = DateAdd("n", intNumMinutos, mdteHoraSistemaFinal)
    lblHoraSimu = Format(mdteHoraSistemaFinal, "hh:mm AM/PM")
    DoEvents
End Sub
Sub procObtenerCoeficientes()
    Dim rstCoefA As ADODB.Recordset
    Dim i%

    Set rstCoefA = New ADODB.Recordset
    rstCoefA.Open "SELECT CoeficienteA " _
        & "FROM CoefA_EntCola " _
        & "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And Dia = " &
        CStr(gintDia) & " ORDER BY Hora;", gCn, adOpenStatic, adLockOptimistic, adCmdText
    With rstCoefA
        For i = 0 To 9
            mCoefAEntCola(i) = .Fields(0)
            .MoveNext
        Next i
    End With
    procCerrarRecordset rstCoefA

    Set rstCoefA = New ADODB.Recordset
    rstCoefA.Open "SELECT CoeficienteA " _
        & "FROM CoefA_SalCola " _
        & "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And Dia = " &
        CStr(gintDia) & " ORDER BY Hora;", gCn, adOpenStatic, adLockOptimistic, adCmdText
    With rstCoefA

```

```

    For i = 0 To 9
        mCoefASalCola(i) = .Fields(0)
        .MoveNext
    Next i
End With
procCerrarRecordset rstCoefA
End Sub
Sub procConfiguracionInicialDeCajeros()
    Dim rstCajerosPos As New ADODB.Recordset
    Dim CajeroNoHallado As Boolean
    Dim NumCajero%
    Dim i%

    'Mostrando aleatoriamente los cajeros activos
    'tomar posición respectiva de la base de datos
    'si existiese
    rstCajerosPos.Open "SELECT Indice " _
        & "From CajerosPosicion " _
        & "Where Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And Dia = " &
    Day(frmParametros.drpCalendario) & " And NumerosCajeros = " & frmParametros.txtSimu & " " _
        & "ORDER BY Indice;", gCn, adOpenStatic, adLockOptimistic, adCmdText
    If Not rstCajerosPos.EOF Then
        While Not rstCajerosPos.EOF
            imgCajeros(rstCajerosPos.Fields(0)).Visible = True
            arrayCajero(rstCajerosPos.Fields(0)).Habilitado = True
            arrayCajero(rstCajerosPos.Fields(0)).TiempoInicial = mdteHoraSistemaFinal
            rstCajerosPos.MoveNext
        Wend
    Else
        For i = 0 To Val(frmParametros.txtSimu) - 1
            CajeroNoHallado = True
            Randomize
            NumCajero = Int((15 * Rnd) + 0)
            While CajeroNoHallado
                If Not imgCajeros(NumCajero).Visible Then
                    'Mostrando imagen de cajero
                    imgCajeros(NumCajero).Visible = True
                    arrayCajero(NumCajero).Habilitado = True
                    arrayCajero(NumCajero).TiempoInicial = mdteHoraSistemaFinal
                    gCn.Execute "INSERT INTO CajerosPosicion (Bco_Codigo,Suc_Codigo,Dia,NumerosCajeros,Indice) " _
                        & "VALUES(" & CStr(gCodBanco) & "," & CStr(gCodSucur) & "," & Day(frmParametros.drpCalendario) & "," &
                    frmParametros.txtSimu & "," & CStr(NumCajero) & ");"

                    'Habilitando timer del cajero
                    'tmrCajero(NumCajero).Enabled = True
                    CajeroNoHallado = False
                Else
                    Randomize
                    NumCajero = Int((15 * Rnd) + 0)
                End If
            Wend
        Next i
    End If
    procCerrarRecordset rstCajerosPos
End Sub

Sub ProcSalidaCola()
    On Error GoTo ERRHANDLER
    Dim lngUltimaPersonaVisible As Variant
    Dim IdCajeroDispo As Variant
    Dim rstSacarCola As DAO.Recordset
    Dim lngTagClienteSalio&
    Dim i&
    Dim l_hombre_rojo As String

    'Obtener cajero disponible
    IdCajeroDispo = fncIdProxCajeroDisponible

    'Verificar si existe cajero disponible
    If Not IsNull(IdCajeroDispo) Then

```

```
If Mid(Format(mdteHoraSistema, "hh:mm AM/PM"), 4, 2) = "00" Then
  If lblHoraFin <> "05:00 PM" Then
    If Format(mdteHoraSistema, "hh:mm AM/PM") <> lblHoraFin Then
      CambiarHoraAuxiliar 'Personas que quedaron de la hora anterior
      MarcarLasPersonasQueSaldranEnUnaHora
    End If
  Else
    CambiarHoraAuxiliar 'Personas que quedaron de la hora anterior
    MarcarLasPersonasQueSaldranEnUnaHora
  End If
End If

If mblnDesertar Then
  DBEngine.BeginTrans
  db_local.Execute "UPDATE " _
    & "CajerosTiemposES " _
    & "SET Desercion = True " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND Dia=" & lblDia & " AND " _
    & "Fecha=#" & lblFecha.Tag & "# AND " _
    & "NumCliente = " & lblCantEntrado & " " _
    & ""
  DBEngine.CommitTrans
  mblnDesertar = False
  lblDeserciones = CInt(lblDeserciones) + 1
End If

If mblnCasoEsp Then
  If imgCliente(0).Visible = False Then
    lblMensaje = "¡No se pueden efectuar casos especiales ya que no hay personas en cola!"
    picMensaje.Visible = True
    tmrMensajes.Enabled = True
  Else
    Randomize
    mmlngUltimoVisible = PrimeraPersonaNoVisible - 1
    mmlngUltimoVisible = 0
    mmbteImagenAleatoria = Int((3 * Rnd) + 1)

    'Tipo de caso especial:
    '1 - Anciano
    '2 - Madre
    '3 - Embarazada
    Select Case mmbteImagenAleatoria
      Case 1
        Select Case mmlngUltimoVisible
          Case 0 To 24
            imgCliente(mmlngUltimoVisible).Picture = imgCasoE(0).Picture
          Case 25 To 48
            imgCliente(mmlngUltimoVisible).Picture = imgCasoE(1).Picture
          Case 49 To 71
            imgCliente(mmlngUltimoVisible).Picture = imgCasoE(0).Picture
        End Select
      Case 2
        Select Case mmlngUltimoVisible
          Case 0 To 24
            imgCliente(mmlngUltimoVisible).Picture = imgCasoE(2).Picture
          Case 25 To 48
            imgCliente(mmlngUltimoVisible).Picture = imgCasoE(3).Picture
          Case 49 To 71
            imgCliente(mmlngUltimoVisible).Picture = imgCasoE(2).Picture
        End Select
      Case 3
        Select Case mmlngUltimoVisible
          Case 0 To 24
            imgCliente(mmlngUltimoVisible).Picture = imgCasoE(4).Picture
          Case 25 To 48
            imgCliente(mmlngUltimoVisible).Picture = imgCasoE(5).Picture
          Case 49 To 71
```

```

        imgCliente(mmlngUltimoVisible).Picture = imgCasoE(4).Picture
    End Select
End Select

mContinuarCasoEspecial = True
DBEngine.BeginTrans
db_local.Execute "UPDATE " _
    & "CajerosTiemposES " _
    & "SET CasoEspecial = True " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND Dia=" & lblDia & " AND " _
    & "Fecha=#" & lblFecha.Tag & "# AND " _
    & "NumCliente = " & lblCantEntrado & " " _
    & ""
DBEngine.CommitTrans

tmrCasoEspecial.Enabled = True
DoEvents
While mContinuarCasoEspecial
    i = 0
    DoEvents
Wend
imgCliente(0).Picture = imgSust(0).Picture
End If
mblnCasoEsp = False
End If

'Verificar si el número de cajeros ha cambiado
If mblnCambiarCajeros Then
    Redefinir la cantidad de personas que saldrán
    MarcarLasPersonasQueSaldránEnUnaHora
    mblnCambiarCajeros = False
End If

'Sacar de cola a las personas con valor True en campo HanSalidoDeCola
Set rstSacarCola = db_local.OpenRecordset("SELECT NumCliente, NumCajero, HanSalidoDeCola, HoraSalidaCliente,
HoraEntradaServCliente, HoraAuxiliar, HoraSalidaServCliente, TiempoServicio, VSalCola, VEntServ, VServ _
    & "From CajerosTiemposES " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND Dia=" & lblDia & " AND " _
    & "Fecha=#" & lblFecha.Tag & "# AND " _
    & "HoraAuxiliar <= #12/30/1899 " & Format(dteHoraFiltro2, "HH:MM:SS") & "# AND " _
    & "DebenSalirDeCola = True AND HanSalidoDeCola = False AND Desercion = False " _
    & "ORDER BY NumCliente;", dbOpenDynaset)
With rstSacarCola
    While Not .EOF
        'Sacar a esta persona de cola
        'obtener el índice de la última imagen visible
        lngUltimaPersonaVisible = UltimaPersonaVisible
        lngUltimaPersonaVisible = 0

        If Not IsNull(lngUltimaPersonaVisible) Then
            imgCliente(lngUltimaPersonaVisible).Visible = False
            l_hombre_rojo = imgCliente(lngUltimaPersonaVisible).Tag

            'Quitar hombre rojo
            If imgCliente(lngUltimaPersonaVisible).Tag = "R" Then
                imgCliente(lngUltimaPersonaVisible).Picture = LoadResPicture(103, vbResBitmap)
                imgCliente(lngUltimaPersonaVisible).Tag = "N"
                g_hombre_rojo = True
            End If

            'Codigo de arrastre de personas
            ArrastreClientes
        End If

        '===== CAJERO ASIGNADO =====
        IdCajeroDispo = fncIdProxCajeroDisponible
        If IsNull(IdCajeroDispo) Then GoTo abajo
    End While
End With

```

```

Comprobar:
'   If IsNull(IdCajeroDispo) Then
'       procReiniciarCajeros
'       IdCajeroDispo = fnclIdProxCajeroDisponible
'       GoTo Comprobar
'   End If
'   arrayCajero(IdCajeroDispo).Ocupado = True
'   imgAtendCajero(IdCajeroDispo).Visible = True

'Configurar cajero
With arrayCajero(IdCajeroDispo)
    .Ocupado = True
    .Contador = 1
    .Limite = rstSacarCola!TiempoServicio / 4
End With

'Verificar si es hombre rojo
If l_hombre_rojo = "R" Then
imgAtendCajero(IdCajeroDispo).Visible = False
imgAtendCajero(IdCajeroDispo).Picture = LoadResPicture(102, vbResBitmap)
imgAtendCajero(IdCajeroDispo).Tag = "N"
'End If

Dim iii As Long, ttttiempo As Long
For iii = 0 To (10000 * IdCajeroDispo)
    ttttiempo = ttttiempo + 1
Next

.Edit
!NumCajero = IdCajeroDispo + 1
!HanSalidoDeCola = True
!HoraSalidaCliente = DateAdd("s", 5, !HoraAuxiliar)
!HoraEntradaServCliente = DateAdd("s", CInt(imgCajeros(IdCajeroDispo).Tag), !HoraSalidaCliente)
!HoraSalidaServCliente = DateAdd("s", !TiempoServicio, !HoraEntradaServCliente)
!VSalCola = Round(fncCalcVelocidad("ASalCola", !HoraSalidaCliente), 0)
!VEntServ = !VSalCola + CInt(imgCajeros(IdCajeroDispo).Tag)
!VSalServ = !TiempoServicio + !VEntServ
.Update

mintPersonasSalenCola = mintPersonasSalenCola + 1
If mintPersonasEntranCola - mintPersonasSalenCola < 0 Then mintPersonasEntranCola = mintPersonasSalenCola
lblCantSalida = mintPersonasSalenCola
lblCantCola = mintPersonasEntranCola - mintPersonasSalenCola
ClientesCola CInt(lblCantCola)
DoEvents

'Actualizar tabla CajerosTransacciones
db_local.Execute "UPDATE CajerosTransacciones SET NumCajero = " & CStr(IdCajeroDispo + 1) & " " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND " _
    & "Dia=" & lblDia & " AND Fecha=#" & lblFecha.Tag & "# AND NumCliente=" & .Fields(0)

'Insertar hombre rojo negro en cola

If l_hombre_rojo = "R" Then
    imgAtendCajero(IdCajeroDispo).Picture = LoadResPicture(102, vbResBitmap)
    imgAtendCajero(IdCajeroDispo).Tag = "R"
Else
    imgAtendCajero(IdCajeroDispo).Picture = LoadResPicture(101, vbResBitmap)
    imgAtendCajero(IdCajeroDispo).Tag = "N"
End If

imgAtendCajero(IdCajeroDispo).Visible = True
'msgbox clientedisponi
.MoveNext
Wend
End With
abajo:
rstSacarCola.Close
Set rstSacarCola = Nothing

```



```

Exit Sub

ERRHANDLER:
  MsgBox Err.Description
  Resume 0
End If
End Sub
***** FUNCIONES *****
Function fncIdProxCajeroDisponible() As Variant
  Dim i%

  fncIdProxCajeroDisponible = Null
  For i = 0 To 14
    If arrayCajero(i).Habilitado And Not arrayCajero(i).Ocupado Then
      'Cajero disponible
      fncIdProxCajeroDisponible = i
      Exit For
    End If
  Next i
End Function
Function fncPromedioSalColaAdd(ByVal fHora As Date) As Double
  Dim rstProm As New ADODB.Recordset

  fncPromedioSalColaAdd = 0
  rstProm.Open "SELECT VscA " _
    & "From PromediosAdicionales " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND " _
    & "Hora=" & Hour(fHora) & " AND " _
    & "Dia=" & lblDia & ";", gCn, adOpenStatic, adLockOptimistic, adCmdText

  With rstProm
    If Not .EOF Then
      fncPromedioSalColaAdd = .Fields(0) * 1000
    End If
  End With
  procCerrarRecordset rstProm
End Function
Function fncPromedioTiempoServ(ByVal fHora As Date) As Double
  Dim rstProm As New ADODB.Recordset

  fncPromedioTiempoServ = 0
  rstProm.Open "SELECT ABS(VvalidaServ-VentradaServ) AS T " _
    & "From PromediosVEntCola " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND " _
    & "Hora=" & Hour(fHora) & " AND " _
    & "Dia=" & lblDia & ";", gCn, adOpenStatic, adLockOptimistic, adCmdText

  With rstProm
    If Not .EOF Then
      fncPromedioTiempoServ = .Fields(0)
    End If
  End With
  rstProm.Close
  Set rstProm = Nothing
End Function

Private Sub chkMostrar_Click()
  If chkMostrar.Value = 0 Then
    Frame2.Visible = False
  Else
    Frame2.Visible = True
  End If
End Sub
Private Sub cmbPer_Click(Index As Integer)
  If Index = 0 Then
    gCn.Execute "UPDATES CajerosTiemposES " _
      & "SET Seleccion = True " _
      & "WHERE Bco_Codigo=" & gCodBanco & " AND " _

```

```

    & "Suc_Codigo=" & gCodSucur & " AND Dia=" & lblDia & " AND " _
    & "Fecha=#" & lblFecha.Tag & "# AND NumCliente=" & lblNumPersona & ";"
End If
imgFlecha.Visible = False
imgPersona.Visible = False
picTiempoPersona.Visible = False
cmbSimu(0).Enabled = True
End Sub
Private Sub cmbSimu_Click(Index As Integer)
    Dim lngIdProxCliente&
    Select Case Index
        Case 0
            lngIdProxCliente = mintPersonasEntranCola
            lblNumPersona = lngIdProxCliente
            picTiempoPersona.Visible = True
            cmbSimu(0).Enabled = False
            cmbPer(0).SetFocus

        Case 1
            Load frmGraficosExcel
            frmGraficosExcel.Show 1

        Case 2
            frmSimulacion.Hide
            Load frmResultados
            frmResultados.Show 1

        Case 3
            mnuArchivoCerrar_Click

        Case 4
            'Adicionar cajero
            If fncSoloUnCajero = 15 Or CInt(lblCantServ) = 15 Then
                MsgBox "¡No es posible tener más de 15 cajeros en el sistema!", vbCritical, MSGVALI
            Else
                lblCantServ = CStr(CInt(lblCantServ) + 1)
                'Recalcular velocidad de salida de cola
                procCalcularPromediosVelocidad_S_Cola CInt(lblDia), CInt(lblCantServ)
                mblnCambiarCajeros = True
                DoEvents
                ProcAdi_QuitarCajero True
                lblMensaje = "¡Se adicionó un cajero!"
                picMensaje.Visible = True
                tmrMensajes.Enabled = True
            End If

        Case 5
            'Quitar cajero
            If CInt(lblCantServ) = 0 Or fncSoloUnCajero = 1 Then
                MsgBox "¡No es posible dejar el sistema sin cajeros!", vbCritical, MSGVALI
            Else
                lblCantServ = CStr(CInt(lblCantServ) - 1)
                'Recalcular velocidad de salida de cola
                procCalcularPromediosVelocidad_S_Cola CInt(lblDia), CInt(lblCantServ)
                mblnCambiarCajeros = True
                DoEvents
                ProcAdi_QuitarCajero False
                lblMensaje = "¡Se retiró un cajero!"
                picMensaje.Visible = True
                tmrMensajes.Enabled = True
            End If

        Case 6
            mblnCasoEsp = True

        Case 7
            Unload frmSimulacion
            Unload frmParametros
            Set frmSimulacion = Nothing
            frmSeleccionBanco.Show
    End Select
End Sub

```

```
Case 8
    mnuOpcionesImprimir_Click

Case 9
    mnuOpcionesReportes_Click

Case 10
    mnuComparacion_Click

Case 11
    mnuArchivoOtraSimulacion_Click
End Select
'Reiniciar cajeros
'procReiniciarCajeros

End Sub
Private Sub Command1_Click()
    If imgCliente(0).Visible = False Then
        lblMensaje = "¡No se pueden efectuar deserciones ya que no hay personas en cola!"
        picMensaje.Visible = True
        tmrMensajes.Enabled = True
    Else
        lblMensaje = "¡Se efectuó una deserción!"
        picMensaje.Visible = True
        tmrMensajes.Enabled = True
        TiempoSimuDesercion = 500
        tmrDeserciones.Enabled = True
        'Marcar la persona que desertó
        mblnDesertar = True
    End If
End Sub
Private Sub Form_Activate()
    Espera 1
End Sub
Private Sub Form_Load()
    'Configurar archivo de ayuda
    App.HelpFile = App.Path & "/recursos/simubq.chm"

    procLineasDeFrente
    procOcultarClientesCola
    procOcultarClientesEnServicio

    'Registros de cajeros y clientes
    gCn.Execute "DELETE * FROM CajerosTiemposES;"
    gCn.Execute "DELETE * FROM CajerosTransacciones;"

    'Estableciendo hora inicial
    dtpHora.Value = Format(frmParametros.cbxHora(0), "00") & ":" & "00"
    mdteHoraSistema = dtpHora.Value
    mdteHoraSistemaFinal = dtpHora.Value

    'Configuracion inicial de cajeros, posición y número
    procConfiguracionInicialDeCajeros

    'Obteniedo valores de Coeficientes A
    procObtenerCoeficientes

    'Mostrando valores iniciales
    lblHoraSimu = Format(mdteHoraSistema, "hh:mm AM/PM")
    lblDia = gintDia
    lblFecha = Format(frmParametros.drpCalendario, "Long Date")
    lblFecha.Tag = Format(frmParametros.drpCalendario, "mm/dd/yyyy")
    dtpHora.Value = frmParametros.cbxHora(0) & ":00"
    dteHoraFiltro1 = dtpHora.Value
    lblHoraIni = Format(dtpHora.Value, "hh:mm AM/PM")
    dtpHora.Value = frmParametros.cbxHora(1) & ":00"
    dteHoraFiltro2 = DateAdd("s", 360, dteHoraFiltro1)
    lblHoraFin = Format(dtpHora.Value, "hh:mm AM/PM")
    lblCantServ = frmParametros.txtSimu
```

```

chkMostrar.Value = 1

'Calculando Promedios de entrada y salida de cola
procCalcularPromediosVelocidad_E_S_Cola CInt(lblDia), CInt(lblCantServ), frmSimulacion.gDiaReal
procGenerarDatosClientesE_S_Cola frmParametros.drpCalendario.Value, CInt(lblDia), CInt(Left(lblHoraIni, 2)),
CInt(Left(lblHoraFin, 2))

gGrabadaSimulacion = False
mblnAgregarClientes = True
DoEvents

'Habilitar timer de entrada en cola
Set db_local = OpenDatabase(App.Path & "\VeVsSimuColas.mdb", False, False)
MarcarLasPersonasQueSaldranEnUnaHora
tmrEntCola.Enabled = True

'Iniciliar entrada de hombre rojo
g_hombre_rojo = True
End Sub
Private Sub mnuArchivoCerrar_Click()
If Not gGrabadaSimulacion Then
If MsgBox("¿Desea grabar la información de la simulación antes de salir?", vbQuestion + vbYesNo, MSGCONF) = vbYes Then
mnuArchivoGuardar_Click
Else
gCn.Execute "DELETE * FROM GeneracionEncabezado " _
& "WHERE Bco_Codigo=" & CStr(gCodBanco) & " AND " _
& "Suc_Codigo=" & CStr(gCodSucur) & " AND " _
& "DiaCalendario=#" & frmSimulacion.lblFecha.Tag & "# AND " _
& "Id=" & gIdSimulacion & ";"
End If
End If
Unload frmSimulacion
Set frmSimulacion = Nothing
End
End Sub
Private Sub mnuArchivoGuardar_Click()
Dim rstVel As New ADODB.Recordset

rstVel.Open "SELECT Avg(CajerosTiemposES.VEntCola) AS PromedioDeVEntCola, Avg(CajerosTiemposES.VSalCola) AS
PromedioDeVSalCola, Avg(CajerosTiemposES.VEntServ) AS PromedioDeVEntServ, Avg(CajerosTiemposES.VSalServ) AS
PromedioDeVSalServ " _
& "From CajerosTiemposES WHERE (((CajerosTiemposES.Desercion)=False));", gCn, adOpenKeyset, adLockOptimistic,
adCmdText
If Not rstVel.EOF Then
lng_VEntCola = Round(rstVel.Fields(0), 0)
lng_VSalCola = Round(rstVel.Fields(1), 0)
lng_VEntServ = Round(rstVel.Fields(2), 0)
lng_VSalServ = Round(rstVel.Fields(3), 0)
End If
rstVel.Close
Set rstVel = Nothing
gCn.Execute "UPDATE GeneracionEncabezado " _
& "SET NumServidores=" & lblCantServ & ", " _
& "CantPersonasEntran=" & lblCantEntrado & ", " _
& "CantPersonasSalen=" & lblCantSalida & ", " _
& "Deserciones=" & lblDeserciones & ", " _
& "CantTransacciones=" & fncNumeroTransacciones(Val(frmSimulacion.lblDia)) & ", " _
& "VEntCola=" & lng_VEntCola & ", " _
& "VSalCola=" & lng_VSalCola & ", " _
& "VEntServ=" & lng_VEntServ & ", " _
& "VSalServ=" & lng_VSalServ & ", " _
& "CantCola=" & lblCantCola & " " _
& "WHERE Bco_Codigo=" & CStr(gCodBanco) & " AND " _
& "Suc_Codigo=" & CStr(gCodSucur) & " AND " _
& "DiaCalendario=#" & frmSimulacion.lblFecha.Tag & "# AND " _
& "Id=" & gIdSimulacion & ";"
MsgBox "La informacion se grabo existosamente", vbInformation, MSGINFO
mnuComparacion.Enabled = True
cmbSimu(10).Enabled = True
gGrabadaSimulacion = True

```

```

End Sub
Private Sub mnuArchivoNuevo_Click()
    If Not gGrabadaSimulacion Then
        If MsgBox("¿Desea grabar la información de la simulación antes de salir?", vbQuestion + vbYesNo, MSGCONF) = vbYes Then
            mnuArchivoGuardar_Click
        End If
    End If
    Unload frmSimulacion
    Unload frmParametros
    Set frmSimulacion = Nothing
    frmSeleccionBanco.Show
End Sub
Private Sub mnuArchivoOtraSimulacion_Click()
    If Not gGrabadaSimulacion Then
        If MsgBox("¿Desea grabar la información de la simulación antes de salir?", vbQuestion + vbYesNo, MSGCONF) = vbYes Then
            mnuArchivoGuardar_Click
        Else
            gCn.Execute "DELETE * FROM GeneracionEncabezado " _
                & "WHERE Bco_Codigo=" & CStr(gCodBanco) & " AND " _
                & "Suc_Codigo=" & CStr(gCodSucur) & " AND " _
                & "Dia_Calendario=#" & frmSimulacion.lblFecha.Tag & "# AND " _
                & "Id=" & gIdSimulacion & ";"
        End If
    End If
    frmParametros.Show
    Unload frmSimulacion
    Set frmSimulacion = Nothing
End Sub
Private Sub mnuAyudaAcerca_Click()
    Load frmThem
    frmThem.Show 1
End Sub
Private Sub mnuAyudaContenido_Click()
    SendKeys "{F1}"
End Sub
Private Sub mnuComparacion_Click()
    Load frmComparacion
    frmComparacion.Show 1
End Sub
Private Sub mnuGráficar_Click()
    Load frmGraficosExcel
    frmGraficosExcel.Show 1
End Sub
Private Sub mnuOpcionesImprimir_Click()
    With CrystalReport1
        .ReportFileName = App.Path & "repResultados.rpt"
        .Formulas(0) = "DIA=" & lblFecha & ""
        .Formulas(1) = "HORAINICIAL=" & lblHoraIni & ""
        .Formulas(2) = "HORAFINAL=" & lblHoraFin & ""
        .Formulas(3) = "CANTENTRAN=" & lblCantEntrado & ""
        .Formulas(4) = "CANTATENDIDAS=" & lblCantSalida & ""
        .Formulas(5) = "SERVIDORES=" & lblCantServ & ""
        .Formulas(6) = "DESERCIONES=" & lblDeserciones & ""
        .Formulas(7) = "TRANSACCIONES=" & fncNumeroTransacciones(Val(frmSimulacion.lblDia)) & ""
        .Formulas(8) = "BANCO=" & gstrNombreBanco & ""
        .Formulas(9) = "SUCURSAL=" & gstrNombreSucursal & ""
        .Formulas(10) = "CANTENCOLA=" & CInt(lblCantEntrado) - CInt(lblCantSalida) & ""
        .Action = 1
    End With
End Sub
Private Sub mnuOpcionesReportes_Click()
    Load frmReportes
    frmReportes.Show 1
End Sub
Private Sub mnuOpcionesResultados_Click()
    frmSimulacion.Hide
    Load frmResultados
    frmResultados.Show 1
End Sub

```

```
Private Sub tmrCajero_Timer()
    'Control de los tiempo de los cliente con los caejero
    VaciarCajeros (15) 'Indica que son 15 segundos
    DoEvents
End Sub

Private Sub tmrCasoEspecial_Timer()
    Select Case mmbtelImagenAleatoria
        Case 1
            imgCliente(0).Picture = imgCasoE(1).Picture
        Case 2
            imgCliente(0).Picture = imgCasoE(3).Picture
        Case 3
            imgCliente(0).Picture = imgCasoE(5).Picture
    End Select

    Select Case mmlngUltimoVisible
        Case 0 To 20
            imgCliente(mmlngUltimoVisible).Picture = imgSust(0).Picture
        Case 21 To 24
            imgCliente(mmlngUltimoVisible).Picture = imgSust(1).Picture
        Case 25 To 45
            imgCliente(mmlngUltimoVisible).Picture = imgSust(2).Picture
        Case 46 To 50
            imgCliente(mmlngUltimoVisible).Picture = imgSust(1).Picture
        Case 51 To 71
            imgCliente(mmlngUltimoVisible).Picture = imgSust(0).Picture
    End Select

    DoEvents
    tmrCasoEspecial.Enabled = False
    tmrCasoEspecialSecundario.Enabled = True
    DoEvents
End Sub

Private Sub tmrCasoEspecialSecundario_Timer()
    mContinuarCasoEspecial = False
    tmrCasoEspecialSecundario.Enabled = False
End Sub

Private Sub tmrDeserciones_Timer()
    Select Case TiempoSimuDesercion
        Case 500
            TiempoSimuDesercion = TiempoSimuDesercion + 500
            imgDesercion(0).Visible = True
        Case 1000
            TiempoSimuDesercion = TiempoSimuDesercion + 500
            imgDesercion(0).Visible = False
            imgDesercion(1).Visible = True
        Case 1500
            TiempoSimuDesercion = TiempoSimuDesercion + 500
            imgDesercion(1).Visible = False
            imgDesercion(2).Visible = True
        Case 2000
            TiempoSimuDesercion = TiempoSimuDesercion + 500
            imgDesercion(2).Visible = False
            imgDesercion(3).Visible = True
        Case 2500
            TiempoSimuDesercion = TiempoSimuDesercion + 500
            imgDesercion(3).Visible = False
            imgDesercion(4).Visible = True
        Case Else
            tmrDeserciones.Enabled = False
            imgDesercion(4).Visible = False
    End Select
End Sub

Private Sub tmrEntCola_Timer()
    Dim rstIntroClientes As DAO.Recordset
    Dim IndiceClienteNoVisible As Variant
    Dim i%
```

On Error GoTo ErrHandler

```

Set rstIntroClientes = db_local.OpenRecordset("SELECT NumCliente, HoraEntradaCliente " _
& "From CajerosTiemposES " _
& "WHERE Bco_Codigo=" & gCodBanco & " AND " _
& "Suc_Codigo=" & gCodSucur & " AND Dia=" & lblDia & " AND " _
& "Fecha=#" & lblFecha.Tag & "# AND " _
& "HoraEntradaCliente Between #12/30/1899 " & Format(dteHoraFiltro1, "HH:MM:SS") & "# And #12/30/1899 " &
Format(dteHoraFiltro2, "HH:MM:SS") & "# AND " _
& "HanSalidoDeCola=False " _
& "ORDER BY NumCliente", dbOpenSnapshot)
With rstIntroClientes
'Obtener el último índice del cliente no visible
IndiceClienteNoVisible = PrimeraPersonaNoVisible

While Not .EOF
'Solitar el ultimo espacio vacio
IndiceClienteNoVisible = PrimeraPersonaNoVisible
'Verificar si existe un espacio vacio en la cola
If Not IsNull(IndiceClienteNoVisible) Then
imgCliente(IndiceClienteNoVisible).Visible = True
IndiceClienteNoVisible = IndiceClienteNoVisible + 1

'Insertar hombre rojo
If g_hombre_rojo Then
imgCliente(IndiceClienteNoVisible).Picture = LoadResPicture(104, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "R"
g_hombre_rojo = False
Else
imgCliente(IndiceClienteNoVisible).Picture = LoadResPicture(103, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "N"
End If

'Imagen activa
i = IndiceClienteNoVisible
Select Case IndiceClienteNoVisible
Case 0 To 20
If g_hombre_rojo Then
frmSimulacion.imgCliente(i).Picture = LoadResPicture(104, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "R"
g_hombre_rojo = False
Else
frmSimulacion.imgCliente(i).Picture = LoadResPicture(103, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "N"
End If
Case 21 To 24
If g_hombre_rojo Then
frmSimulacion.imgCliente(i).Picture = LoadResPicture(108, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "R"
g_hombre_rojo = False
Else
frmSimulacion.imgCliente(i).Picture = LoadResPicture(105, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "N"
End If
Case 25 To 45
If g_hombre_rojo Then
frmSimulacion.imgCliente(i).Picture = LoadResPicture(109, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "R"
g_hombre_rojo = False
Else
frmSimulacion.imgCliente(i).Picture = LoadResPicture(106, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "N"
End If
Case 46 To 50
If g_hombre_rojo Then
frmSimulacion.imgCliente(i).Picture = LoadResPicture(110, vbResBitmap)
imgCliente(IndiceClienteNoVisible).Tag = "R"
g_hombre_rojo = False
Else
frmSimulacion.imgCliente(i).Picture = LoadResPicture(107, vbResBitmap)

```

```

        imgCliente(IndiceClienteNoVisible).Tag = "N"
    End If
    Case 51 To 71
        If g_hombre_rojo Then
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(104, vbResBitmap)
            imgCliente(IndiceClienteNoVisible).Tag = "R"
            g_hombre_rojo = False
        Else
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(103, vbResBitmap)
            imgCliente(IndiceClienteNoVisible).Tag = "N"
        End If
    End Select
    IndiceClienteNoVisible = IndiceClienteNoVisible + 1
End If

gArregloClientes(fncIdUltimaPersonaNoHaEntrado).EntroACola = True
mintPersonasEntranCola = mintPersonasEntranCola + 1

If mintPersonasEntranCola - mintPersonasSalenCola < 0 Then mintPersonasEntranCola = mintPersonasSalenCola
lblCantEntrado = mintPersonasEntranCola
lblCantCola = mintPersonasEntranCola - mintPersonasSalenCola
lblCantSalida = mintPersonasSalenCola
ClientesCola CInt(lblCantCola)

.MoveNext
Wend
End With
rstIntroClientes.Close
Set rstIntroClientes = Nothing

tmrEntCola.Enabled = False
tmrSalCola.Enabled = True
tmrCajero.Enabled = True
DoEvents

Exit Sub
ErrorHandler:
If Err.Number = 340 Then
    Resume Next
Else
    MsgBox Err.Description
End If
End Sub
Private Sub tmrHoraBanco_Timer()
    If SimulaciónFinalizada Then
        If CInt(lblDeserciones) > 0 Then
            If lblHoraFin = "05:00 PM" Then
                lblCantCola = "0"
                lblCantSalida = CInt(lblCantEntrado) - CInt(lblDeserciones)
                ClientesCola 0
            End If
        End If
    End If

    tmrEntCola.Enabled = False
    tmrSalCola.Enabled = False
    tmrHoraBanco.Enabled = False

    cmbSimu(8).Enabled = True
    cmbSimu(1).Enabled = True
    cmbSimu(2).Enabled = True
    cmbSimu(9).Enabled = True
    cmbSimu(7).Enabled = True
    cmbSimu(11).Enabled = True

    cmbSimu(0).Enabled = False
    cmbSimu(6).Enabled = False
    cmbSimu(4).Enabled = False
    cmbSimu(5).Enabled = False
    Command1.Enabled = False

```



```

mnuArchivoNuevo.Enabled = True
mnuArchivoOtraSimulacion.Enabled = True
mnuArchivoGuardar.Enabled = True
mnuOpciones.Enabled = True

db_local.Execute "UPDATE " _
& "CajerosTiemposES " _
& "SET VSalCola = VEntCola, HoraSalidaCliente = HoraEntradaCliente " _
& "WHERE Bco_Codigo=" & gCodBanco & " AND " _
& "Suc_Codigo=" & gCodSucur & " AND Dia=" & lblDia & " AND " _
& "Fecha=#" & lblFecha.Tag & "# AND CasoEspecial = True;"

MsgBox "¡La simulación ha finalizado!", vbExclamation, MSGINFO
Exit Sub
End If

procAumentarMinutos 6
tmrEntCola.Enabled = True

tmrHoraBanco.Enabled = False
End Sub
Private Sub tmrMensajes_Timer()
picMensaje.Visible = False
tmrMensajes.Enabled = False
End Sub
Private Sub tmrSalCola_Timer()
'=====
ProcSalidaCola
'=====
dteHoraFiltro1 = DateAdd("s", 1, dteHoraFiltro2)
dteHoraFiltro2 = DateAdd("s", 360, dteHoraFiltro2)
tmrSalCola.Enabled = False
tmrHoraBanco.Enabled = True
End Sub

```

### **FrmThem**

```

Private Sub Command1_Click()
Unload Me
End Sub
Private Sub Form_Activate()
Timer1.Enabled = True
End Sub
Private Sub Form_Load()
Dim tam As Integer
Timer1.Interval = 35
VScroll1.Max = Picture1.Height
VScroll1.Min = 0 - Text1.Height
VScroll1.Value = VScroll1.Max
End Sub
Private Sub Timer1_Timer()
If VScroll1.Value >= VScroll1.Min + 20 Then
VScroll1.Value = VScroll1.Value - 35
Else
VScroll1.Value = VScroll1.Max
DoEvents
End If
Text1.Top = VScroll1.Value
Text1.Visible = True
DoEvents
End Sub

```

### **ModPrincipal**

```

'-----
'Declaración de constantes públicas
'-----

```

```

Public Const NOMBRE_BASEDATOS = "VeVsSimuColas.mdb"
Public Const MSGVALI = "Mensaje de validación"
Public Const MSGCONF = "Mensaje de confirmación"
Public Const MSGINFO = "Mensaje de información"

'-----
'Declaración de variables públicas
'-----
'Variable con la cual se realiza la conexión
'a la base de datos VeVsSimuColas
Public gCn As ADODB.Connection
Public gArregloClientes(0 To 250000) As typeCliente

Public gCodBanco As Long
Public gCodSucur As Long
Public gblnSimulacionAnterior As Boolean
Public gstrNombreBanco$
Public gstrNombreSucursal$
Public gIdSimulacion As Long

Public Type typeCliente
    EntroACola As Boolean
    SalioDeCola As Boolean
End Type

Public Type typeCajero
    NumCaja As Integer
    Habilitado As Boolean
    Ocupado As Boolean
    TiempoInicial As Date
    TiempoFinal As Date
    Contador As Integer
    Limite As Integer
End Type

Sub Vinculación(dbsTemp As Database, _
    strTabla As String, strConectar As String, _
    strTablaOrigen As String)
    On Error GoTo ErrHandler

    Dim tdfVinculado As TableDef
    Dim intTemp As Integer
    Dim Intentos%
    Dim BorrandoTabla As Boolean

    Intentos = 0
    BorrandoTabla = True

    ExisteTabla dbsTemp, strTabla
    ' Crea un nuevo TableDef, establece las propiedades Connect
    ' y SourceTableName basadas en los argumentos transferidos
    ' y lo anexa a la colección TableDefs.
    Set tdfVinculado = dbsTemp.CreateTableDef(strTabla)
    tdfVinculado.Connect = strConectar
    tdfVinculado.SourceTableName = strTablaOrigen
    dbsTemp.TableDefs.Append tdfVinculado

    Exit Sub

ErrHandler:
    If BorrandoTabla Then
        Resume Next
    Else
        Espera 1
        MsgBox Err.Number & " " & Err.Description, vbCritical, "Ocurrió un error al tratar de vincular la tabla Menuemp"
    End If
End Sub

Sub ExisteTabla(ByRef DbEmpn As DAO.Database, ByRef nombreTabla As String)
    Dim i As Integer

```

```

For i = 0 To DbEmpn.TableDefs.Count - 1
  If UCase(DbEmpn.TableDefs(i).Name) = UCase(nombreTabla) Then
    DBEngine.BeginTrans
    DbEmpn.Execute "DROP TABLE " & UCase(nombreTabla)
    DBEngine.CommitTrans
    Exit For
  End If
Next i
End Sub
Sub Main()
  'Creando DSN de usuario en computadora
  'utilizando el DBEnginme de Microsoft DAO
  DBEngine.RegisterDatabase "Colas", _
    "Microsoft Access Driver (*.mdb)", True, _
    "DBQ=" & App.Path & "\ " & NOMBRE_BASEDATOS

  'Realizando la conexión a la
  'base de datos VeVsSimuColas.mdb
  'mediante ADO v2.5(Activex Data Object)

  'Inicializando variables de conexión
  Set gCn = New ADODB.Connection
  'Conentando se mediante DSN=Colas
  gCn.Open "Colas"
  'Iniciando/mostrando pantalla de presentación
  Espera
  Load frmPress
  frmPress.Show
End Sub

'Procedimiento que camia el puntero del mouse
'de flecha a reloj de arena y viceversa
Sub Espera(Optional modo)
  If IsMissing(modos) Then Screen.MousePointer = 11 Else Screen.MousePointer = 0
End Sub

'Procedimiento para cerrar Recordset y liberarlos de memoria
Sub procCerrarRecordset(ByRef rstRegistro As ADODB.Recordset)
  If rstRegistro.State = 1 Then
    rstRegistro.Close
    Set rstRegistro = Nothing
  End If
End Sub

'Convierte a letras mayúsculas recibiendo como argumento
'un valor Ascii
Sub procConvMayusculas(ByRef KeyAscii As Integer)
  KeyAscii = Asc(UCase(Chr(KeyAscii)))
  If KeyAscii = Asc(vbCr) Then KeyAscii = 0
End Sub

'Procedimiento para cerrar el programa y la conexión con la base de datos
'VeVsSimuColas.mdb y liberar de memoria la variable tipo conexión
Sub CerrarSistema()
  gCn.Close
  Set gCn = Nothing
End
End Sub

Sub GrabarInformacionDeNuevoBanco()
  Dim rstId As ADODB.Recordset
  Dim rstDatosTablas As ADODB.Recordset
  Dim strSql$
  Dim lngIdBanco&
  Dim lngIdSucursal&
  Dim itm As ListItem
  Dim i&

  'Iniciando la transacción
  'gCn.BeginTrans

```

```

Select Case frmRecepcionDeDatos.gTipo
Case 0 'Crear banco y sucursal
'Obtener un nuevo código de Banco
strSql = "SELECT IIF(ISNULL(Max(Bancos.Bco_Codigo)),1,Max(Bancos.Bco_Codigo)+1) AS MáxDeBco_Codigo " _
& "FROM Bancos;"
Set rstId = New ADODB.Recordset
rstId.Open strSql, gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstId
    lngIdBanco = .Fields(0)
End With
procCerrarRecordset rstId
'Crear Banco
gCn.Execute "INSERT INTO Bancos (Bco_Codigo,Bco_Nombre) VALUES(" & CStr(lngIdBanco) & "," &
frmRecepcionDeDatos.txtSimu(0) & ");"
gCodBanco = lngIdBanco

'Obtener un nuevo código de Sucursal
strSql = "SELECT IIF(ISNULL(Max(Sucursales.Suc_Codigo)),1,Max(Sucursales.Suc_Codigo)+1) AS MáxDeSuc_Codigo " _
& "From Sucursales WHERE Sucursales.Bco_Codigo=" & CStr(lngIdBanco) & ";"
Set rstId = New ADODB.Recordset
rstId.Open strSql, gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstId
    lngIdSucursal = .Fields(0)
End With
procCerrarRecordset rstId

'Crear Sucursal
gCn.Execute "INSERT INTO Sucursales " _
& "(Bco_Codigo,Suc_Codigo,Suc_Nombre) " _
& "VALUES(" & CStr(lngIdBanco) & "," & CStr(lngIdSucursal) & "," & frmRecepcionDeDatos.txtSimu(1) & ");"
gCodSucur = lngIdSucursal

Case 1 'Crear sucursal
lngIdBanco = CLng(gCodBanco)
'Obtener un nuevo código de Sucursal
strSql = "SELECT IIF(ISNULL(Max(Sucursales.Suc_Codigo)),1,Max(Sucursales.Suc_Codigo)+1) AS MáxDeSuc_Codigo " _
& "From Sucursales WHERE Sucursales.Bco_Codigo=" & CStr(lngIdBanco) & ";"
Set rstId = New ADODB.Recordset
rstId.Open strSql, gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstId
    lngIdSucursal = .Fields(0)
End With
procCerrarRecordset rstId

'Crear Sucursal
gCn.Execute "INSERT INTO Sucursales " _
& "(Bco_Codigo,Suc_Codigo,Suc_Nombre) " _
& "VALUES(" & CStr(lngIdBanco) & "," & CStr(lngIdSucursal) & "," & frmRecepcionDeDatos.txtSimu(1) & ");"
gCodSucur = lngIdSucursal

Case 2 'Nuevos datos
lngIdBanco = CLng(gCodBanco)
lngIdSucursal = CLng(gCodSucur)
gCn.Execute "DELETE * FROM EntCola " _
& "WHERE Bco_Codigo = " & CStr(lngIdBanco) & " AND Suc_Codigo = " & CStr(lngIdSucursal) & ";"
gCn.Execute "DELETE * FROM SalCola " _
& "WHERE Bco_Codigo = " & CStr(lngIdBanco) & " AND Suc_Codigo = " & CStr(lngIdSucursal) & ";"
gCn.Execute "DELETE * FROM CoefA_EntCola " _
& "WHERE Bco_Codigo = " & CStr(lngIdBanco) & " AND Suc_Codigo = " & CStr(lngIdSucursal) & ";"
gCn.Execute "DELETE * FROM CoefA_SalCola " _
& "WHERE Bco_Codigo = " & CStr(lngIdBanco) & " AND Suc_Codigo = " & CStr(lngIdSucursal) & ";"

gCn.Execute "DELETE * FROM PromTrans " _
& "WHERE Bco_Codigo = " & CStr(lngIdBanco) & ";"

End Select

frmProgreso.ProgressBar1.Value = 10
DoEvents

```

```

'Almacenar información en tabla EntCola (Entrada Cola)
Set rstDatosTablas = New ADODB.Recordset
With rstDatosTablas
    .Open "EntCola", gCn, adOpenKeyset, adLockOptimistic, adCmdTable
    For i = 1 To frmRecepcionDeDatos.lsvTbl(0).ListItems.Count
        .AddNew
        !Bco_Codigo = lngIdBanco
        !Suc_Codigo = lngIdSucursal
        !Dia = frmRecepcionDeDatos.lsvTbl(0).ListItems(i).ListSubItems(1)
        !Hora = frmRecepcionDeDatos.lsvTbl(0).ListItems(i).ListSubItems(2)
        !Tiempo = CDate("00:" & frmRecepcionDeDatos.lsvTbl(0).ListItems(i).ListSubItems(3))
        !TiempoE = frmRecepcionDeDatos.lsvTbl(0).ListItems(i).ListSubItems(4)
        .Update
    Next i
End With
procCerrarRecordset rstDatosTablas
frmProgreso.ProgressBar1.Value = 28
DoEvents

'Almacenar información en tabla SalCola (Salida Cola)
Set rstDatosTablas = New ADODB.Recordset
With rstDatosTablas
    .Open "SalCola", gCn, adOpenKeyset, adLockOptimistic, adCmdTable
    For i = 1 To frmRecepcionDeDatos.lsvTbl(1).ListItems.Count
        .AddNew
        !Bco_Codigo = lngIdBanco
        !Suc_Codigo = lngIdSucursal
        !Dia = frmRecepcionDeDatos.lsvTbl(1).ListItems(i).ListSubItems(1)
        !Hora = frmRecepcionDeDatos.lsvTbl(1).ListItems(i).ListSubItems(2)
        !Tiempo = CDate("00:" & frmRecepcionDeDatos.lsvTbl(1).ListItems(i).ListSubItems(3))
        !TiempoS = frmRecepcionDeDatos.lsvTbl(1).ListItems(i).ListSubItems(4)
        .Update
    Next i
End With
procCerrarRecordset rstDatosTablas
frmProgreso.ProgressBar1.Value = 46
DoEvents

If frmRecepcionDeDatos.gTipo <> 1 Then
'Almacenar información en tabla SalServ (Salida Servicio)
Set rstDatosTablas = New ADODB.Recordset
With rstDatosTablas
    .Open "PromTrans", gCn, adOpenKeyset, adLockOptimistic, adCmdTable
    For i = 1 To frmRecepcionDeDatos.lsvTbl(4).ListItems.Count
        .AddNew
        !Bco_Codigo = lngIdBanco
        !NoTrans = i
        .Fields("Nombre Transacción") = frmRecepcionDeDatos.lsvTbl(4).ListItems(i)
        !Promedios = CDate("00:" & frmRecepcionDeDatos.lsvTbl(4).ListItems(i).ListSubItems(1))
        !TPromSegs = Val(frmRecepcionDeDatos.lsvTbl(4).ListItems(i).ListSubItems(2))
        !Porcentaje = Val(frmRecepcionDeDatos.lsvTbl(4).ListItems(i).ListSubItems(3))
        .Update
    Next i
End With
procCerrarRecordset rstDatosTablas
End If

'CALCULO DE COEFICIENTES DE COLA (ENTRADA Y SALIDA)
CoeficientesDeCola lngIdBanco, lngIdSucursal, "Entrada", "A"
CoeficientesDeCola lngIdBanco, lngIdSucursal, "Salida", "A"

End Sub
'Calculo de los coeficientes de cola E/S
Sub CoeficientesDeCola(ByVal lngIdBanco As Long, _
    ByVal lngIdSucursal As Long, _
    ByVal strTipo As String, _
    ByVal strCoef As String)

Dim rstTablaBase As New ADODB.Recordset

```

```

Dim rstTablaFiltrada As ADODB.Recordset
Dim strNomTblBase$
Dim strNomTblCoef$
Dim strSQL_Ppal$
Dim strSQL_Filtrado$
'-----

Dim K() As Double
Dim a() As Double
Dim NumIndices As Byte
Dim intDia&

'Establece si la cola es de entrada o de salida
If strTipo = "Entrada" Then
'Establece si el coeficiente es A ó B
If strCoef = "A" Then
strNomTblBase = "EntCola"
strNomTblCoef = "CoefA_EntCola"
strSQL_Ppal = "SELECT Dia From EntCola WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " &
CStr(IngIdSucursal) & " GROUP BY Dia ORDER BY Dia;"
strSQL_Filtrado = "SELECT Hora, TiempoE FROM EntCola " _
& " WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " & CStr(IngIdSucursal) & " AND Dia = "

gCn.Execute "DELETE * FROM CoefA_EntCola " _
& " WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " & CStr(IngIdSucursal) & ";"
NumIndices = 10
Else
strNomTblBase = "CoefA_EntCola"
strNomTblCoef = "CoefB_EntCola"
strSQL_Ppal = "SELECT Etiqueta FROM CoefA_EntCola WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND
Suc_Codigo = " & CStr(IngIdSucursal) & " GROUP BY Etiqueta ORDER BY Etiqueta;"
strSQL_Filtrado = "SELECT Dia, CoeficienteA, Hora FROM CoefA_EntCola " _
& " WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " & CStr(IngIdSucursal) & " AND Etiqueta = "

gCn.Execute "DELETE * FROM CoefB_EntCola " _
& " WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " & CStr(IngIdSucursal) & ";"
NumIndices = 15
End If
Else
'Establece si el coeficiente es A ó B
If strCoef = "A" Then
strNomTblBase = "SalCola"
strNomTblCoef = "CoefA_SalCola"
strSQL_Ppal = "SELECT Dia FROM SalCola WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " &
CStr(IngIdSucursal) & " GROUP BY Dia ORDER BY Dia;"
strSQL_Filtrado = "SELECT Hora, TiempoS FROM SalCola " _
& " WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " & CStr(IngIdSucursal) & " AND Dia = "

gCn.Execute "DELETE * FROM CoefA_SalCola " _
& " WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " & CStr(IngIdSucursal) & ";"
NumIndices = 10
Else
strNomTblBase = "CoefA_SalCola"
strNomTblCoef = "CoefB_SalCola"
strSQL_Ppal = "SELECT Etiqueta FROM CoefA_SalCola WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND
Suc_Codigo = " & CStr(IngIdSucursal) & " GROUP BY Etiqueta ORDER BY Etiqueta;"
strSQL_Filtrado = "SELECT Dia, CoeficienteA, Hora FROM CoefA_SalCola " _
& " WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " & CStr(IngIdSucursal) & " AND Etiqueta = "

gCn.Execute "DELETE * FROM CoefB_SalCola " _
& " WHERE Bco_Codigo = " & CStr(IngIdBanco) & " AND Suc_Codigo = " & CStr(IngIdSucursal) & ";"
NumIndices = 15
End If
End If

ReDim K(1 To NumIndices)
ReDim a(1 To NumIndices, 1 To NumIndices)

'-----
'Recorriendo tabla principal
'-----

```

```

rstTablaBase.Open strSQL_Ppal, gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstTablaBase
  While Not .EOF

    '-----
    'Filtrando los registros a trabajar
    '-----
    Set rstTablaFiltrada = New ADODB.Recordset
    Debug.Print strSQL_Filtrado & IIf(strCofef = "A", .Fields(0) & " ORDER BY Hora;", "" & .Fields(0) & " ORDER BY Dia;")
    rstTablaFiltrada.Open strSQL_Filtrado & IIf(strCofef = "A", .Fields(0) & " ORDER BY Hora;", "" & .Fields(0) & " ORDER
BY Dia;"), gCn, adOpenStatic, adLockOptimistic, adCmdText
    For i = 1 To 10
      M = Val(rstTablaFiltrada.Fields(0))
      For j = 1 To 10
        If j = 1 Then
          a(i, j) = 1
        Else
          a(i, j) = a(i, j - 1) * M
        End If
      Next j
      K(i) = Val(rstTablaFiltrada.Fields(1))
      rstTablaFiltrada.MoveNext
    Next i

    For i = 1 To 10
      For j = 1 To 10
        If i <> j Then
          M = a(j, i) / a(i, i)
          For l = 1 To 10
            a(j, l) = a(j, l) - M * a(i, l)
          Next
          K(j) = K(j) - M * K(i)
        End If
      Next j
    Next i

    For i = 1 To 10
      K(i) = K(i) / a(i, i)
    Next

    rstTablaFiltrada.MoveFirst
    For i = 1 To 10
      If strCofef = "A" Then
        gCn.Execute "INSERT INTO " & strNomTblCofef _
          & " (Bco_Codigo, Suc_Codigo, Etiqueta, Dia, CoeficienteA, Hora) VALUES(" & CStr(IngIdBanco) & "," &
CStr(IngIdSucursal) & "," & "a" & CStr(i - 1) & "," & CStr(.Fields(0)) & "," & K(i) & "," & rstTablaFiltrada.Fields(0) & ");"
      Else
        gCn.Execute "INSERT INTO " & strNomTblCofef _
          & " (Bco_Codigo, Suc_Codigo, Etiqueta, CoeficienteB, Dia, EtiquetaA, Hora) VALUES(" & CStr(IngIdBanco) & "," &
& CStr(IngIdSucursal) & "," & "b" & CStr(i - 1) & "," & K(i) & "," & rstTablaFiltrada.Fields(0) & "," & .Fields(0) & "," &
rstTablaFiltrada.Fields(2) & ");"
      End If
      rstTablaFiltrada.MoveNext
    Next i

    procCerrarRecordset rstTablaFiltrada
    .MoveNext
  Wend
End With
procCerrarRecordset rstTablaBase
End Sub

Sub procCalcularPromediosVelocidad_E_S_Cola(ByVal intDia As Integer, ByVal numCaj As Integer, ByVal intDiaReal As Integer)
  Dim rstCofefA As ADODB.Recordset
  Dim j%
  Dim i%
  Dim e%
  Dim t As Date
  Dim arregloEntCola(0 To 9) As Double
  Dim arregloSalCola(0 To 9) As Double

```

```

Dim dblPromedioEntCola As Double
Dim dblPromedioSalCola As Double
Dim intConta%

dblPromedioEntCola = 0
dblPromedioSalCola = 0
intConta = 0

gCn.Execute "DELETE * FROM PromediosVEntCola " _
& "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & ";"

'ENTRADA DE COLA
Set rstCoefA = New ADODB.Recordset
rstCoefA.Open "SELECT CoeficienteA " _
& "FROM CoefA_EntCola " _
& "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And Dia = " & CStr(intDia)
& " ORDER BY Hora;", gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstCoefA
  For i = 0 To 9
    arregloEntCola(i) = .Fields(0)
    .MoveNext
  Next i
End With
procCerrarRecordset rstCoefA

'SALIDA DE COLA
Set rstCoefA = New ADODB.Recordset
rstCoefA.Open "SELECT CoeficienteA " _
& "FROM CoefA_SalCola " _
& "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And Dia = " & CStr(intDia)
& " ORDER BY Hora;", gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstCoefA
  For i = 0 To 9
    arregloSalCola(i) = .Fields(0)
    .MoveNext
  Next i
End With
procCerrarRecordset rstCoefA

t = #8:00:00 AM#
'De las 8 a las 17 horas
For i = 0 To 8
  'De intervalos de 6 minutos
  For e = 0 To 9
    'Incrementar cada hora en 6 minutos
    u = 24 * CDBl(t)

    For j = 0 To 9
      CalcVelocidad1 = CalcVelocidad1 + arregloEntCola(j) * (u ^ j)
      CalcVelocidad = CalcVelocidad + arregloSalCola(j) * (u ^ j)
    Next j
    'Debug.Print CStr(Round(Abs(CalcVelocidad1), 2)) & " " & CStr(t)
    dblPromedioEntCola = dblPromedioEntCola + Abs(CalcVelocidad1)
    dblPromedioSalCola = dblPromedioSalCola + Abs(CalcVelocidad)

    CalcVelocidad1 = 0
    CalcVelocidad = 0

    intConta = intConta + 1
    t = DateAdd("n", 6, t)

    If Mid(Format(t, "hh:mm AM/PM"), 4, 2) = "00" And CStr(Hour(t)) <> "8" Then
      'Grabar datos en tabla
      dblPromedioEntCola = Round(dblPromedioEntCola / intConta, 0)
      dblPromedioSalCola = Round(dblPromedioSalCola / intConta, 0)

      gCn.Execute "INSERT INTO PromediosVEntCola " _
& "(Bco_Codigo,Suc_Codigo,Dia,Hora,VPromEntCola,VPromSalCola) VALUES(" _

```



```

    & CStr(gCodBanco) & "," & gCodSucur & "," & intDia & "," & (CInt(Hour(t)) - 1) & "," & Round(dblPromedioEntCola
+ (dblPromedioEntCola * fncVariabilidadDias(intDiaReal)), 0) & "," & Round(5 * (dblPromedioSalCola + (dblPromedioSalCola *
fncVariabilidadDias(intDiaReal))) / numCaj, 0) & ");"

```

```

        dblPromedioEntCola = 0
        dblPromedioSalCola = 0
        intConta = 0
    End If
Next e
Next i
End Sub
'GENERADOR DE TRANSACCIONES POR CAJERO
Sub procGenerarTransacciones(ByVal Dia As Integer)
    Dim dblocal As DAO.Database
    Dim rstAtencion As DAO.Recordset
    Dim rstClientes As DAO.Recordset
    Dim rstTransac As DAO.Recordset
    Dim tmpTiempoTrans#
    Dim blnBuscar As Boolean

    Set dblocal = OpenDatabase(App.Path & "\VeVsSimuColas.mdb", False, False)

    Set rstTransac = dblocal.OpenRecordset("SELECT TPromSegs, NoTrans " _
    & "From PromTrans " _
    & "Where PromTrans.Bco_Codigo = " & gCodBanco & " " _
    & "ORDER BY PromTrans.Porcentaje DESC;", dbOpenSnapshot)

    dblocal.Execute "DELETE * FROM Cajeros " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND " _
    & "Dia=" & Dia & ";"

    dblocal.Execute "INSERT INTO Cajeros " _
    & "SELECT CajerosTiemposES.Bco_Codigo, CajerosTiemposES.Suc_Codigo, CajerosTiemposES.Dia,
CajerosTiemposES.NumCajero " _
    & "From CajerosTiemposES " _
    & "GROUP BY CajerosTiemposES.Bco_Codigo, CajerosTiemposES.Suc_Codigo, CajerosTiemposES.Dia,
CajerosTiemposES.NumCajero " _
    & "Having CajerosTiemposES.Bco_Codigo = " & gCodBanco & " And " _
    & "CajerosTiemposES.Suc_Codigo = " & gCodSucur & " And " _
    & "CajerosTiemposES.Dia = " & Dia & " " _
    & "ORDER BY CajerosTiemposES.NumCajero;"

    dblocal.Execute "DELETE * FROM CajerosTransacciones " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND " _
    & "Dia=" & Dia & ";"

    Set rstClientes = dblocal.OpenRecordset("SELECT CajerosTiemposES.NumCajero, [CajerosTiemposES]![VSalServ]-
[CajerosTiemposES]![VEntServ] AS Expr1, CajerosTiemposES.NumCliente " _
    & "From CajerosTiemposES " _
    & "Where CajerosTiemposES.Bco_Codigo = " & gCodBanco & " And " _
    & "CajerosTiemposES.Suc_Codigo = " & gCodSucur & " And " _
    & "CajerosTiemposES.Dia = " & Dia & " " _
    & "ORDER BY CajerosTiemposES.NumCajero;", dbOpenSnapshot)
    With rstClientes
        While Not .EOF
            'Teniendo el tiempo seleccionar
            'las posibles transacciones
            If .Fields(1) > 0 Then
                tmpTiempoTrans = .Fields(1)
                blnBuscar = True
                rstTransac.MoveFirst

                For i = 1 To rstTransac.RecordCount
                    If tmpTiempoTrans > 0 Then
                        'Veamos si la cantidad de tiempo de la transaccion actual
                        'es menor o igual a la cantidad de tiempo en atención
                        If tmpTiempoTrans - rstTransac.Fields(0) >= 0 Then

```

```

    dblocal.Execute "INSERT INTO CajerosTransacciones " _
      & "(Bco_Codigo,Suc_Codigo,Dia,NumCliente,NumCajero,NoTrans) " _
      & "VALUES(" _
      & gCodBanco & "," _
      & gCodSucur & "," _
      & Dia & "," _
      & .Fields(2) & "," _
      & .Fields(0) & "," _
      & rstTransac.Fields(1) & ");"

    tmpTiempoTrans = tmpTiempoTrans - rstTransac.Fields(0)
  End If
Else
  Exit For
End If
rstTransac.MoveNext
Next i
End If

.MoveNext
Wend
.Close
End With
Set rstClientes = Nothing

dblocal.Close
Set dblocal = Nothing
End Sub
Sub procCalcularPromediosVelocidad_S_Cola(ByVal intDia As Integer, ByVal numCaj As Integer)
  Dim rstCoefA As ADODB.Recordset
  Dim j%
  Dim i%
  Dim e%
  Dim t As Date
  Dim arregloSalServ(0 To 9) As Double
  Dim dblPromedioServ As Double
  Dim intConta%

  dblPromedioServ = 0
  intConta = 0

  'SALIDA DE COLA
  Set rstCoefA = New ADODB.Recordset
  rstCoefA.Open "SELECT CoeficienteA " _
    & "FROM CoefA_SalCola " _
    & "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And Dia = " & CStr(intDia)
  & " ORDER BY Hora;", gCn, adOpenStatic, adLockOptimistic, adCmdText
  With rstCoefA
    For i = 0 To 9
      arregloSalServ(i) = .Fields(0)
      .MoveNext
    Next i
  End With
  procCerrarRecordset rstCoefA

  t = #8:00:00 AM#
  'De las 8 a las 17 horas
  For i = 0 To 8
    'De intervalos de 6 minutos
    For e = 0 To 9
      'Incrementar cada hora en 6 minutos
      u = 24 * CDbl(t)

      For j = 0 To 9
        CalcVelocidad = CalcVelocidad + arregloSalServ(j) * (u ^ j)
      Next j
      dblPromedioServ = dblPromedioServ + Abs(CalcVelocidad)

      CalcVelocidad = 0
      intConta = intConta + 1
    Next e
  Next i
End Sub

```

```

t = DateAdd("n", 6, t)

If Mid(Format(t, "hh:mm AM/PM"), 4, 2) = "00" And CStr(Hour(t)) <> "8" Then
'Grabar datos en tabla
dblPromedioServ = Round(dblPromedioServ / intConta, 0)

gCn.Execute "UPDATE PromediosVEntCola SET VPromEntCola=" & 5 * dblPromedioServ / numCaj & " " _
& "WHERE Bco_Codigo = " & CStr(gCodBanco) & " AND " _
& "Suc_Codigo = " & CStr(gCodSucur) & " AND " _
& "Dia = " & CStr(intDia) & " AND " _
& "Hora = " & (CInt(Hour(t)) - 1) & ";"

dblPromedioServ = 0
intConta = 0
End If
Next e
Next i
End Sub
Sub procCalcularPromediosTiempoEnServicio(ByVal intDia As Integer, ByVal numCaj As Integer)
Dim rstCoefA As ADODB.Recordset
Dim j%
Dim i%
Dim e%
Dim t As Date
Dim arregloEntCola(0 To 9) As Double
Dim arregloSalServ(0 To 9) As Double
Dim dblPromedioCola As Double
Dim dblPromedioServ As Double
Dim intConta%

dblPromedioCola = 0
dblPromedioServ = 0
intConta = 0

'ENTRADA EN SERVICIO
Set rstCoefA = New ADODB.Recordset
rstCoefA.Open "SELECT CoeficienteA " _
& "FROM CoefA_EntServ " _
& "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And Dia = " & CStr(intDia)
& " ORDER BY Hora;", gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstCoefA
For i = 0 To 9
arregloEntCola(i) = .Fields(0)
.MoveNext
Next i
End With
procCerrarRecordset rstCoefA

'SALIDA DE SERVICIO
Set rstCoefA = New ADODB.Recordset
rstCoefA.Open "SELECT CoeficienteA " _
& "FROM CoefA_SalServ " _
& "WHERE Bco_Codigo = " & CStr(gCodBanco) & " And Suc_Codigo = " & CStr(gCodSucur) & " And Dia = " & CStr(intDia)
& " ORDER BY Hora;", gCn, adOpenStatic, adLockOptimistic, adCmdText
With rstCoefA
For i = 0 To 9
arregloSalServ(i) = .Fields(0)
.MoveNext
Next i
End With
procCerrarRecordset rstCoefA

t = #8:00:00 AM#
'De las 8 a las 17 horas
For i = 0 To 8
'De intervalos de 6 minutos
For e = 0 To 9
Incrementar cada hora en 6 minutos
u = 24 * CDBl(t)

```

```

For j = 0 To 9

    CalcVelocidad1 = CalcVelocidad1 + arregloEntCola(j) * (u ^ j)
    CalcVelocidad = CalcVelocidad + arregloSalServ(j) * (u ^ j)
Next j
dblPromedioCola = dblPromedioCola + Abs(CalcVelocidad1)
dblPromedioServ = dblPromedioServ + Abs(CalcVelocidad)

CalcVelocidad1 = 0
CalcVelocidad = 0

intConta = intConta + 1
t = DateAdd("n", 6, t)

If Mid(Format(t, "hh:mm AM/PM"), 4, 2) = "00" And CStr(Hour(t)) <> "8" Then
    'Grabar datos en tabla
    dblPromedioCola = Round(dblPromedioCola / intConta, 0)
    dblPromedioServ = Round(dblPromedioServ / intConta, 0)

    gCn.Execute "UPDATE PromediosVEntCola " _
        & "SET VentradaServ = " & CStr(dblPromedioCola) & ", VsalidaServ = " & CStr(dblPromedioServ) & ",
NumPersonasSalPorCajero = " & CInt(360000 / (Abs(dblPromedioServ - dblPromedioCola) * 1000) * 10) & " " _
        & "WHERE Bco_Codigo=" & CStr(gCodBanco) & " AND " _
        & "Suc_Codigo=" & CStr(gCodSucur) & " AND " _
        & "Dia=" & intDia & " AND Hora=" & (CInt(Hour(t)) - 1) & " ";

    dblPromedioCola = 0
    dblPromedioServ = 0
    intConta = 0
End If
Next e
Next i
End Sub
Sub procCalculosAdicionales(ByVal intDia As Integer)
    Dim dblocal As DAO.Database
    Dim rst1 As DAO.Recordset
    Dim rst2 As DAO.Recordset
    Dim intPec As Integer
    Dim intPsc As Integer
    Dim intVec As Integer
    Dim intVsc As Integer
    Dim intVzca As Integer
    Dim intTc As Integer
    Dim intPa As Integer

    Set dblocal = OpenDatabase(App.Path & "\VeVsSimuColas.mdb", False, False)
    dblocal.Execute "DELETE * " _
        & "FROM PromediosAdicionales " _
        & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
        & "Suc_Codigo=" & gCodSucur & " AND " _
        & "Dia=" & CStr(intDia) & " ";

    Set rst2 = dblocal.OpenRecordset("PromediosAdicionales", dbOpenTable)
    Set rst1 = dblocal.OpenRecordset("SELECT * " _
        & "FROM PromediosVEntCola " _
        & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
        & "Suc_Codigo=" & gCodSucur & " AND " _
        & "Dia=" & CStr(intDia) & " ", dbOpenSnapshot)
    With rst1
        While Not .EOF
            If !Hora <> 8 Then
                If intPec - intPsc > 0 Then
                    'Existen adicionales
                    intPa = intPec - intPsc
                    intVzca = intTc + !Valor2

                    dblocal.Execute "INSERT INTO PromediosAdicionales " _
                        & "(Bco_Codigo,Suc_Codigo,Dia,Hora,NumAdicionales,Vzca,Vsca) VALUES(" _
                        & gCodBanco & ", " & gCodSucur & ", " & intDia & ", " & !Hora & ", " _

```

```

        & intPa & "," & intVec & "," & intVsca & ");"
    End If
End If

'Almacenando valores
intVec = !Valor
intVsc = !Valor2
intTc = !Valor2 - !Valor
intPec = Round(360000 / (!Valor * 1000) * 10, 0)
intPsc = Round(360000 / (!Valor2 * 1000) * 10, 0)
.MoveNext
Wend
.Close
End With
Set rst1 = Nothing
End Sub
Sub procCreacionDePromedios()
    Dim rstPromCola As New ADODB.Recordset
    Dim rstPromServ As New ADODB.Recordset
    Dim rstCal As ADODB.Recordset
    Dim strSql$
    Dim strCondHora$
    Dim i%

    rstPromCola.Open "_PromediosCola", gCn, adOpenKeyset, adLockOptimistic, adCmdTable
    rstPromServ.Open "_PromediosServicio", gCn, adOpenKeyset, adLockOptimistic, adCmdTable

    For i = 8 To 17
        Select Case i
            Case 8
                strCondHora = "#12/30/1899 8:0:0# And #12/30/1899 9:0:0#"
            Case 9
                strCondHora = "#12/30/1899 9:6:0# And #12/30/1899 10:0:0#"
            Case 10
                strCondHora = "#12/30/1899 10:6:0# And #12/30/1899 11:0:0#"
            Case 11
                strCondHora = "#12/30/1899 11:6:0# And #12/30/1899 12:0:0#"
            Case 12
                strCondHora = "#12/30/1899 12:6:0# And #12/30/1899 13:0:0#"
            Case 13
                strCondHora = "#12/30/1899 13:6:0# And #12/30/1899 14:0:0#"
            Case 14
                strCondHora = "#12/30/1899 14:6:0# And #12/30/1899 15:0:0#"
            Case 15
                strCondHora = "#12/30/1899 15:6:0# And #12/30/1899 16:0:0#"
            Case 16
                strCondHora = "#12/30/1899 16:6:0# And #12/30/1899 17:0:0#"
            Case 17
                strCondHora = "#12/30/1899 17:6:0# And #12/30/1899 18:0:0#"
        End Select

        strSql = "SELECT Count(CajerosTiemposES.NumCliente) AS CuentaDeNumCliente, Avg(CajerosTiemposES.VEntCola) AS
PromedioDeVEntCola, Avg(CajerosTiemposES.VSalCola) AS PromedioDeVSalCola " _
        & "From CajerosTiemposES " _
        & "WHERE CajerosTiemposES.HoraEntradaCliente Between " & strCondHora & " " _
        & "GROUP BY CajerosTiemposES.Bco_Codigo, CajerosTiemposES.Suc_Codigo, CajerosTiemposES.Dia " _
        & "HAVING CajerosTiemposES.Bco_Codigo=" & CStr(gCodBanco) & " AND CajerosTiemposES.Suc_Codigo=" &
CStr(gCodSucur) & " AND CajerosTiemposES.Dia=" & frmSimulacion.lblDia
        Set rstCal = New ADODB.Recordset
        rstCal.Open strSql, gCn, adOpenStatic, adLockOptimistic, adCmdText
        With rstCal
            If Not .EOF Then
                gCn.Execute "INSERT INTO
_PromediosCola(Bco_Codigo,Suc_Codigo,DiaCalendario,Id,Hora,CantEntraron,CantSalieron,VEntCola,VSalCola)"
            End If
        End With
        procCerrarRecordset rstCal
    Next i
End Sub
'Valida valores de hora

```

```

Public Function fncHoraValida(ByVal strHora As String) As Boolean
    On Error GoTo ErrHandler
    frmRecepcionDeDatos.dtpHora.Value = strHora
    fncHoraValida = True
    Exit Function

ErrHandler:
    fncHoraValida = False
End Function
'Devuelve el formato de hora
Public Function fncFORMATO_HORA(ByVal dtmHora As Variant) As String
    If IsNull(dtmHora) Then fncFORMATO_HORA = "_:_____" Else fncFORMATO_HORA = Format(dtmHora, "hh:mm:ss
AM/PM")
End Function

Public Function fncFORMATO_HORAMMSS(ByVal dtmHora As Variant) As String
    If IsNull(dtmHora) Then fncFORMATO_HORAMMSS = "_:_" Else fncFORMATO_HORAMMSS = Mid(Format(dtmHora,
"hh:mm:ss"), 4, 5)
End Function
Public Function Lee_Campo(ByVal S_parCadena As Variant) As String
    Lee_Campo = IIf(IsNull(S_parCadena), "", Trim(S_parCadena))
End Function
Function fncNumeroTransacciones(ByVal Dia As Integer) As Long
    Dim rstNumTrans As New ADODB.Recordset

    fncNumeroTransacciones = 0
    rstNumTrans.Open "SELECT Count(Dia) AS CuentaDeDia " _
        & "From CajerosTransacciones " _
        & "WHERE Bco_Codigo = " & CStr(gCodBanco) & " AND " _
        & "Suc_Codigo = " & CStr(gCodSucur) & " AND " _
        & "Dia = " & CStr(Dia) & " AND NumCajero<>1;", gCn, adOpenStatic, adLockOptimistic, adCmdText
    If Not rstNumTrans.EOF Then
        If Not IsNull(rstNumTrans.Fields(0)) Then
            fncNumeroTransacciones = rstNumTrans.Fields(0)
        End If
    End If
    procCerrarRecordset rstNumTrans
End Function
Function procNuevaSimulacion(ByVal fec As Date) As Long
    Dim rstId As New ADODB.Recordset

    gIdSimulacion = 1
    rstId.Open "SELECT Max(GeneracionEncabezado.Id) AS MáxDeId " _
        & "From GeneracionEncabezado " _
        & "WHERE Bco_Codigo=" & CStr(gCodBanco) & " AND " _
        & "Suc_Codigo=" & CStr(gCodSucur) & " AND " _
        & "DiaCalendario=#" & Format(fec, "mm/dd/yyyy") & "#;", gCn, adOpenStatic, adLockOptimistic, adCmdText
    With rstId
        If Not .EOF Then
            If Not IsNull(.Fields(0)) Then gIdSimulacion = .Fields(0) + 1
        End If
    End With
    procCerrarRecordset rstId
End Function
Function fncPromedioEntCola(ByVal fHora As Date, ByVal intDia, Optional c) As Double
    Dim rstProm As New ADODB.Recordset

    fncPromedioEntCola = 0
    rstProm.Open "SELECT VPromEntCola " _
        & "From PromediosVEntCola " _
        & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
        & "Suc_Codigo=" & gCodSucur & " AND " _
        & "Hora=" & IIf(IsMissing(c), Hour(fHora), Hour(fHora) - 1) & " AND " _
        & "Dia=" & intDia & ";", gCn, adOpenStatic, adLockOptimistic, adCmdText

    With rstProm
        If Not .EOF Then
            fncPromedioEntCola = .Fields(0) * 1000
        End If
    End With
End Function

```

```

rstProm.Close
Set rstProm = Nothing
End Function
Function fncPromedioSalCola(ByVal fHora As Date, ByVal intDia, Optional c) As Double
    Dim rstProm As New ADODB.Recordset

    fncPromedioSalCola = 0
    rstProm.Open "SELECT VPromSalCola " _
        & "From PromediosVEntCola " _
        & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
        & "Suc_Codigo=" & gCodSucur & " AND " _
        & "Hora=" & IIf(IsMissing(c), IIf(Hour(fHora) = 17, 16, Hour(fHora)), Hour(fHora) - 1) & " AND " _
        & "Dia=" & intDia & ";", gCn, adOpenStatic, adLockOptimistic, adCmdText

    With rstProm
        If Not .EOF Then
            fncPromedioSalCola = .Fields(0) * 1000
        End If
    End With
    rstProm.Close
    Set rstProm = Nothing
End Function
Function fncCalcVelocidad(ByVal strTipo As String, _
    ByVal t As Date) As Double
    Dim i%
    Dim u#

    fncCalcVelocidad = 0
    u = 24 * CDbl(t)

    Select Case strTipo
        Case "AEntCola"
            For i = 0 To 9
                fncCalcVelocidad = fncCalcVelocidad + mCoefAEntCola(i) * (u ^ i)
            Next
        Case "ASalCola"
            For i = 0 To 9
                fncCalcVelocidad = fncCalcVelocidad + mCoefASalCola(i) * (u ^ i)
            Next
    End Select

    fncCalcVelocidad = Abs(fncCalcVelocidad)
End Function
Function fncVariabilidadDias(ByVal intDia) As Double
    fncVariabilidadDias = 0

    If intDia > 15 Then
        Select Case intDia
            Case 16
                fncVariabilidadDias = 1 / 5
            Case 17
                fncVariabilidadDias = 1 / 10
            Case 18
                fncVariabilidadDias = 1 / 10
            Case 19
                fncVariabilidadDias = 1 / 5
            Case 20
                fncVariabilidadDias = 1 / 15
            Case 21
                fncVariabilidadDias = 1 / 6
            Case 22
                fncVariabilidadDias = 1 / 10
            Case 23
                fncVariabilidadDias = 1 / 18
            Case 24
                fncVariabilidadDias = 1 / 20
            Case 25
                fncVariabilidadDias = 1 / 25
            Case 26

```

```
        fncVariabilidadDias = -1 / 10
    Case 27
        fncVariabilidadDias = -1 / 15
    Case 28
        fncVariabilidadDias = -1 / 20
    Case 29
        fncVariabilidadDias = -1 / 20
    Case 30
        fncVariabilidadDias = -1 / 25
    Case 31
        fncVariabilidadDias = -1 / 30
    End Select
End If
End Function
*****
```

## **ModSimulacion**

Option Explicit

```
Dim arraySegundos(0 To 3600) As Boolean
Public mCoefAEntCola(9) As Double
Public mCoefASalCola(9) As Double
Public mCoefAEntServ(9) As Double
Public mCoefASalServ(9) As Double
```

```
'Variables de gráficos
Public g_oGráficoExcel As Excel.Chart
Dim m_oExcel As New Excel.Application
Dim m_oHojaExcel As Excel.Worksheet
Dim m_oLibroExcel As Excel.Workbook
Public g_sCircuito$
Public g_hombre_rojo As Boolean
```

```
Public Type typeTrans
    CodigoTransaccion As Integer
    Cantidad As Integer
End Type
```

```
Sub procInicializarCtrlSegundos()
    Dim i%
    For i = 0 To 3600
        arraySegundos(i) = False
    Next i
End Sub
```

```
Sub procGenerarDatosClientesE_S_Cola(ByVal dteFecha As Date, _
    ByVal intDia As Integer, _
    ByVal intHoraIni As Integer, _
    ByVal intHoraFin As Integer)
```

```
    Dim lngCantPersonasEntranUnaHora%
    Dim rstCajerosTiemposES As New ADODB.Recordset
```

```
    Dim db_local As DAO.Database
    Dim rstTransacciones As DAO.Recordset
    Dim rstInfo As DAO.Recordset
    Dim dteHora As Date
    Dim SegundosNoHallados As Boolean
    Dim NumSegundos%
    Dim ContaPersonas%
    Dim intNumTotalTrans&
    Dim numTransPorFrecuencia&
    Dim pVez As Boolean
    Dim i%
    Dim j%
```

```
    pVez = True
    Select Case intHoraIni
```



```
Case 1
  intHoraIni = 13
Case 2
  intHoraIni = 14
Case 3
  intHoraIni = 15
Case 4
  intHoraIni = 16
Case 5
  intHoraIni = 17
End Select
Select Case intHoraFin
Case 1
  intHoraFin = 13
Case 2
  intHoraFin = 14
Case 3
  intHoraFin = 15
Case 4
  intHoraFin = 16
Case 5
  intHoraFin = 17
End Select
rstCajerosTiemposES.Open "CajerosTiemposES", gCn, adOpenKeyset, adLockOptimistic, adCmdTable
For i = intHoraIni To intHoraFin - 1
  Select Case i
  Case 8
    dteHora = #8:00:00 AM#
  Case 9
    dteHora = #9:00:00 AM#
  Case 10
    dteHora = #10:00:00 AM#
  Case 11
    dteHora = #11:00:00 AM#
  Case 12
    dteHora = #12:00:00 PM#
  Case 13
    dteHora = #1:00:00 PM#
  Case 14
    dteHora = #2:00:00 PM#
  Case 15
    dteHora = #3:00:00 PM#
  Case 16
    dteHora = #4:00:00 PM#
  Case 17
    dteHora = #5:00:00 PM#
  End Select

  'Obtener el número de personas que entraran en esta hora
  'y la hora en la que deberan entrar
  lngCantPersonasEntranUnaHora = Round(360000 / fncPromedioEntCola(dteHora, intDia) * 10, 0)
  procInicializarCtrlSegundos

  While lngCantPersonasEntranUnaHora > 0
    'Definir velocidad de entrada en cola instantanea
    'y hora de entrada
    SegundosNoHallados = True
    Randomize
    NumSegundos = Int((3600 * Rnd) + 0)
    While SegundosNoHallados
      If arraySegundos(NumSegundos) = False Then
        arraySegundos(NumSegundos) = True
        SegundosNoHallados = False
        lngCantPersonasEntranUnaHora = lngCantPersonasEntranUnaHora - 1
      Else
        Randomize
        NumSegundos = Int((3600 * Rnd) + 0)
      End If
    End While
  Wend
Wend
```

```

'Grabar resultados en la base de datos
If pVez Then
  ContaPersonas = 0
  pVez = False
End If
For j = 0 To 3600
  If arraySegundos(j) Then
    With rstCajerosTiemposES
      .AddNew
      !Bco_Codigo = gCodBanco
      !Suc_Codigo = gCodSucur
      !NumCliente = ContaPersonas
      !Dia = intDia
      !Fecha = dteFecha
      !NumCajero = -1
      !Hora = Hour(dteHora)
      !HoraEntradaCliente = DateAdd("s", j, dteHora)
      !HoraAuxiliar = !HoraEntradaCliente
      'Por cada cliente se debe generar sus transacciones
      !TiempoServicio = fncGenerarTransacciones(ContaPersonas, intDia, Format(dteFecha, "mm/dd/yyyy"), -1,
Hour(dteHora))
      !VEntCola = Round(fncCalcVelocidad("AEntCola", !HoraEntradaCliente), 0)
      .Update
    End With

    ContaPersonas = ContaPersonas + 1
  End If
Next j
Next i

*****
intNumTotalTrans = 0
If frmSimulacion.gPerfilTransacc = 1 Then
  Set db_local = OpenDatabase(App.Path & "\\VeVsSimuColas.mdb", False, False)
  Set rstInfo = db_local.OpenRecordset("SELECT Count(NoTrans) AS CuentaDeNoTrans " _
    & "From CajerosTransacciones " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
    & "Suc_Codigo=" & gCodSucur & " AND " _
    & "Dia=" & intDia & " AND " _
    & "Fecha=#" & Format(dteFecha, "mm/dd/yyyy") & "#;", dbOpenSnapshot)

  If Not rstInfo.EOF Then
    If Not IsNull(rstInfo.Fields(0)) Then
      intNumTotalTrans = rstInfo.Fields(0)
    End If
  End If
  rstInfo.Close
  Set rstInfo = Nothing

  If intNumTotalTrans > 0 Then
    Set rstTransacciones = db_local.OpenRecordset("SELECT NoTrans " _
      & "From CajerosTransacciones " _
      & "Where Bco_Codigo=" & gCodBanco & " And Suc_Codigo=" & gCodSucur & " And Dia=" & intDia & " And Fecha=#"
    & Format(dteFecha, "mm/dd/yyyy") & "# " _
      & "ORDER BY NoTrans", dbOpenDynaset)

    'Distribución de transacciones en base a porcentajes de frecuencia
    Set rstInfo = db_local.OpenRecordset("SELECT Porcentaje, NoTrans " _
      & "From PromTrans Where Bco_Codigo = " & gCodBanco & " AND Porcentaje>0 ORDER BY Porcentaje DESC",
dbOpenSnapshot)
    With rstInfo
      While Not .EOF
        numTransPorFrecuencia = Int(intNumTotalTrans / 100 * .Fields(0))

        'Obtener el número de transacciones a definir
        While Not rstTransacciones.EOF And numTransPorFrecuencia > 0
          rstTransacciones.Edit
          rstTransacciones!NoTrans = .Fields(1)
          rstTransacciones.Update
        End While
      End While
    End With
  End If
End If

```

```

        rstTransacciones.MoveNext
        numTransPorFrecuencia = numTransPorFrecuencia - 1
    Wend
    .MoveNext
Wend

rstInfo.MoveFirst
While Not rstTransacciones.EOF
    rstTransacciones.Edit
    rstTransacciones!NoTrans = rstInfo.Fields(1)
    rstTransacciones.Update
    rstTransacciones.MoveNext
Wend
.Close
End With
Set rstInfo = Nothing
End If
rstTransacciones.Close
Set rstTransacciones = Nothing
db_local.Close
Set db_local = Nothing
End If
End Sub

Sub CrearHojaExcel(ByVal TipoInfo As Integer, ByVal fecSimu As String)
    Set m_oLibroExcel = m_oExcel.Workbooks.Add
    Set m_oHojaExcel = m_oLibroExcel.Worksheets(1)

    Dim Indice&
    Dim numColumnas&
    Dim db_local As DAO.Database
    Dim rstInfo As DAO.Recordset
    Dim rstDatosGraficos As DAO.Recordset

    Set db_local = OpenDatabase(App.Path & "\VeVsSimuColas.mdb", False, False)

    Select Case TipoInfo
    Case 0
        db_local.Execute "DELETE * FROM _GraficoEntSalPersonas;"

        Set rstDatosGraficos = db_local.OpenRecordset("SELECT HoraEntrada, Entraron, Salieron FROM _GraficoEntSalPersonas",
dbOpenDynaset)
        Set rstInfo = db_local.OpenRecordset("SELECT * " _
& "FROM Qry_NumPersonasEntraron " _
& "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" & fecSimu & "#",
dbOpenSnapshot)
        While Not rstInfo.EOF
            rstDatosGraficos.FindFirst "HoraEntrada = " & rstInfo!HoraEntrada & ""
            If rstDatosGraficos.NoMatch Then
                rstDatosGraficos.AddNew
                rstDatosGraficos!HoraEntrada = rstInfo!HoraEntrada
                rstDatosGraficos!Entraron = rstInfo!CuentaDeNumCliente
                rstDatosGraficos!Salieron = 0
            End If
            rstDatosGraficos.Update
            rstInfo.MoveNext
        Wend

        Set rstInfo = db_local.OpenRecordset("SELECT * " _
& "FROM Qry_NumPersonasSalieron " _
& "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" & fecSimu & "#",
dbOpenSnapshot)
        While Not rstInfo.EOF
            rstDatosGraficos.FindFirst "HoraEntrada = " & rstInfo!HoraSalida & ""
            If rstDatosGraficos.NoMatch Then
                rstDatosGraficos.AddNew
                rstDatosGraficos!HoraEntrada = rstInfo!HoraSalida
                rstDatosGraficos!Entraron = 0
                rstDatosGraficos!Salieron = rstInfo!CuentaDeNumCliente
            Else

```

```

    rstDatosGraficos.Edit
    rstDatosGraficos!Salieron = rstDatosGraficos!Salieron + rstInfo!CuentaDeNumCliente
End If
rstDatosGraficos.Update
rstInfo.MoveNext
Wend

'Horas
Indice = 1
numColumns = 0
If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
While Not rstDatosGraficos.EOF
    With m_oHojaExcel.Rows(1)
        .Font.Bold = True
        .Cells(, Indice).Value = CStr(rstDatosGraficos.Fields(0))
    End With
    Indice = Indice + 1
    numColumns = numColumns + 1
    rstDatosGraficos.MoveNext
Wend

'Entrada en cola
If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
Indice = 1
While Not rstDatosGraficos.EOF
    With m_oHojaExcel.Rows(2)
        .Cells(, Indice).Value = CStr(If(IsNull(rstDatosGraficos.Fields(1)), 0, rstDatosGraficos.Fields(1)))
    End With
    Indice = Indice + 1
    rstDatosGraficos.MoveNext
Wend

'Salida en cola
If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
Indice = 1
While Not rstDatosGraficos.EOF
    With m_oHojaExcel.Rows(3)
        .Cells(, Indice).Value = CStr(If(IsNull(rstDatosGraficos.Fields(2)), 0, rstDatosGraficos.Fields(2)))
    End With
    Indice = Indice + 1
    rstDatosGraficos.MoveNext
Wend

rstDatosGraficos.Close
Set rstDatosGraficos = Nothing
rstInfo.Close
Set rstInfo = Nothing
db_local.Close
Set db_local = Nothing
CrearGráfico frmGraficosExcel.vntValorTipoGrafico, TipoInfo, numColumns

Case 1
Set rstInfo = db_local.OpenRecordset("SELECT CajerosTiemposES.NumCajero, " _
& "Count(CajerosTiemposES.NumCliente) AS CuentaDeNumCliente " _
& "From CajerosTiemposES " _
& "Where (CajerosTiemposES.Bco_Codigo = " & gCodBanco & ") And (CajerosTiemposES.Suc_Codigo = " & gCodSucur
& ") And (CajerosTiemposES.Fecha = #" & fecSimu & "#) And (CajerosTiemposES.Desercion = False) " _
& "GROUP BY CajerosTiemposES.NumCajero " _
& "Having CajerosTiemposES.NumCajero <> -1 " _
& "ORDER BY CajerosTiemposES.NumCajero", dbOpenSnapshot)

'Cajeros
Indice = 1
numColumns = 0
If rstInfo.RecordCount > 0 Then rstInfo.MoveFirst
While Not rstInfo.EOF
    With m_oHojaExcel.Rows(1)
        .Font.Bold = True
        .Cells(, Indice).Value = CStr(rstInfo.Fields(0))
    End With

```

```

    Indice = Indice + 1
    numColumnas = numColumnas + 1
    rstInfo.MoveNext
Wend

'Cantidad
If rstInfo.RecordCount > 0 Then rstInfo.MoveFirst
Indice = 1
While Not rstInfo.EOF
    With m_oHojaExcel.Rows(2)
        .Cells(, Indice).Value = CStr(IIf(IsNull(rstInfo.Fields(1)), 0, rstInfo.Fields(1)))
    End With
    Indice = Indice + 1
    rstInfo.MoveNext
Wend

rstInfo.Close
Set rstInfo = Nothing
db_local.Close
Set db_local = Nothing
CrearGráfico frmGraficosExcel.vntValorTipoGrafico, TipoInfo, numColumnas

Case 2
If frmGraficosExcel.vntValorTipoGrafico <> xl3DPieExploded Then
    Set rstInfo = db_local.OpenRecordset("SELECT PromTrans.NoTrans, Count(PromTrans.NoTrans) AS CuentaDeNoTrans "
    -
        & "FROM PromTrans INNER JOIN CajerosTransacciones ON (PromTrans.NoTrans = CajerosTransacciones.NoTrans)
AND (PromTrans.Bco_Codigo = CajerosTransacciones.Bco_Codigo) " _
        & "Where PromTrans.Bco_Codigo = " & gCodBanco & " AND CajerosTransacciones.NumCajero<>-1 " _
        & "GROUP BY PromTrans.NoTrans " _
        & "ORDER BY PromTrans.NoTrans", dbOpenSnapshot)
    Else
        Set rstInfo = db_local.OpenRecordset("SELECT PromTrans.[Nombre Transacción], Count(PromTrans.NoTrans) AS
CuentaDeNoTrans " _
        & "FROM PromTrans INNER JOIN CajerosTransacciones ON (PromTrans.NoTrans = CajerosTransacciones.NoTrans)
AND (PromTrans.Bco_Codigo = CajerosTransacciones.Bco_Codigo) " _
        & "Where PromTrans.Bco_Codigo = " & gCodBanco & " AND CajerosTransacciones.NumCajero<>-1 " _
        & "GROUP BY PromTrans.[Nombre Transacción];", dbOpenSnapshot)
    End If

    'Transacción
    Indice = 1
    numColumnas = 0
    If rstInfo.RecordCount > 0 Then rstInfo.MoveFirst
    While Not rstInfo.EOF
        With m_oHojaExcel.Rows(1)
            .Font.Bold = True
            .Cells(, Indice).Value = CStr(rstInfo.Fields(0))
        End With
        Indice = Indice + 1
        numColumnas = numColumnas + 1
        rstInfo.MoveNext
    Wend

    'Cantidad
    If rstInfo.RecordCount > 0 Then rstInfo.MoveFirst
    Indice = 1
    While Not rstInfo.EOF
        With m_oHojaExcel.Rows(2)
            .Cells(, Indice).Value = CStr(IIf(IsNull(rstInfo.Fields(1)), 0, rstInfo.Fields(1)))
        End With
        Indice = Indice + 1
        rstInfo.MoveNext
    Wend

    rstInfo.Close
    Set rstInfo = Nothing
    db_local.Close
    Set db_local = Nothing
    CrearGráfico frmGraficosExcel.vntValorTipoGrafico, TipoInfo, numColumnas

```

```

Case 3
db_local.Execute "DELETE * FROM _PromediosCola;"

Set rstDatosGraficos = db_local.OpenRecordset("SELECT Hora, VPromEntCola, VPromSalCola FROM _PromediosCola",
dbOpenDynaset)
Set rstInfo = db_local.OpenRecordset("SELECT * " _
& "FROM Qry_PromedioEntradaCola " _
& "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" & fecSimu & "#",
dbOpenSnapshot)
While Not rstInfo.EOF
rstDatosGraficos.FindFirst "Hora = " & rstInfo!Hora & ""
If rstDatosGraficos.NoMatch Then
rstDatosGraficos.AddNew
rstDatosGraficos!Hora = rstInfo!Hora
rstDatosGraficos!VPromEntCola = Round(rstInfo!PromedioDeVEntCola, 0)
rstDatosGraficos!VPromSalCola = 0
End If
rstDatosGraficos.Update
rstInfo.MoveNext
Wend

Set rstInfo = db_bcal.OpenRecordset("SELECT * " _
& "FROM Qry_PromedioSalidaCola " _
& "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" & fecSimu & "#",
dbOpenSnapshot)
While Not rstInfo.EOF
rstDatosGraficos.FindFirst "Hora = " & rstInfo!Hora & ""
If rstDatosGraficos.NoMatch Then
rstDatosGraficos.AddNew
rstDatosGraficos!Hora = rstInfo!Hora
rstDatosGraficos!VPromEntCola = 0
rstDatosGraficos!VPromSalCola = Round(rstInfo!PromedioDeVSalCola, 0)
Else
rstDatosGraficos.Edit
rstDatosGraficos!VPromSalCola = rstDatosGraficos!VPromSalCola + Round(rstInfo!PromedioDeVSalCola, 0)
End If
rstDatosGraficos.Update
rstInfo.MoveNext
Wend

'Horas
Indice = 1
numColumns = 0
If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
While Not rstDatosGraficos.EOF
With m_oHojaExcel.Rows(1)
.Font.Bold = True
.Cells(, Indice).Value = CStr(rstDatosGraficos.Fields(0))
End With
Indice = Indice + 1
numColumns = numColumns + 1
rstDatosGraficos.MoveNext
Wend

'Entrada en cola
If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
Indice = 1
While Not rstDatosGraficos.EOF
With m_oHojaExcel.Rows(2)
.Cells(, Indice).Value = CStr(If(IsNull(rstDatosGraficos.Fields(1)), 0, rstDatosGraficos.Fields(1)))
End With
Indice = Indice + 1
rstDatosGraficos.MoveNext
Wend

'Salida en cola
If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
Indice = 1
While Not rstDatosGraficos.EOF

```

```

    With m_oHojaExcel.Rows(3)
        .Cells(, Indice).Value = CStr(If(IsNull(rstDatosGraficos.Fields(2)), 0, rstDatosGraficos.Fields(2)))
    End With
    Indice = Indice + 1
    rstDatosGraficos.MoveNext
Wend

rstDatosGraficos.Close
Set rstDatosGraficos = Nothing
rstInfo.Close
Set rstInfo = Nothing
db_local.Close
Set db_local = Nothing
CrearGráfico frmGraficosExcel.vntValorTipoGrafico, TipoInfo, numColumnas

Case 4
    db_local.Execute "DELETE * FROM _PromediosServicio;"

    Set rstDatosGraficos = db_local.OpenRecordset("SELECT Hora, VPromEntServ, VPromSalServ FROM
    _PromediosServicio", dbOpenDynaset)
    Set rstInfo = db_local.OpenRecordset("SELECT * " _
    & "FROM Qry_PromedioEntradaServ " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" & fecSimu & "#",
dbOpenSnapshot)
    While Not rstInfo.EOF
        rstDatosGraficos.FindFirst "Hora = " & rstInfo!Hora & ""
        If rstDatosGraficos.NoMatch Then
            rstDatosGraficos.AddNew
            rstDatosGraficos!Hora = rstInfo!Hora
            rstDatosGraficos!VPromEntServ = Round(rstInfo!PromedioDeVEntServ, 0)
            rstDatosGraficos!VPromSalServ = 0
        End If
        rstDatosGraficos.Update
        rstInfo.MoveNext
    Wend

    Set rstInfo = db_local.OpenRecordset("SELECT * " _
    & "FROM Qry_PromedioSalidaServ " _
    & "WHERE Bco_Codigo=" & gCodBanco & " AND Suc_Codigo=" & gCodSucur & " AND Fecha=#" & fecSimu & "#",
dbOpenSnapshot)
    While Not rstInfo.EOF
        rstDatosGraficos.FindFirst "Hora = " & rstInfo!Hora & ""
        If rstDatosGraficos.NoMatch Then
            rstDatosGraficos.AddNew
            rstDatosGraficos!Hora = rstInfo!Hora
            rstDatosGraficos!VPromEntServ = 0
            rstDatosGraficos!VPromSalServ = Round(rstInfo!PromedioDeVSalServ, 0)
        Else
            rstDatosGraficos.Edit
            rstDatosGraficos!VPromSalServ = rstDatosGraficos!VPromSalServ + Round(rstInfo!PromedioDeVSalServ, 0)
        End If
        rstDatosGraficos.Update
        rstInfo.MoveNext
    Wend

    'Horas
    Indice = 1
    numColumnas = 0
    If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
    While Not rstDatosGraficos.EOF
        With m_oHojaExcel.Rows(1)
            .Font.Bold = True
            .Cells(, Indice).Value = CStr(rstDatosGraficos.Fields(0))
        End With
        Indice = Indice + 1
        numColumnas = numColumnas + 1
        rstDatosGraficos.MoveNext
    Wend

    'Promedio entrada en servicio

```

```

If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
Indice = 1
While Not rstDatosGraficos.EOF
  With m_oHojaExcel.Rows(2)
    .Cells(, Indice).Value = CStr(If(IsNull(rstDatosGraficos.Fields(1)), 0, rstDatosGraficos.Fields(1)))
  End With
  Indice = Indice + 1
  rstDatosGraficos.MoveNext
Wend

'Promedio salida en servicio
If rstDatosGraficos.RecordCount > 0 Then rstDatosGraficos.MoveFirst
Indice = 1
While Not rstDatosGraficos.EOF
  With m_oHojaExcel.Rows(3)
    .Cells(, Indice).Value = CStr(If(IsNull(rstDatosGraficos.Fields(2)), 0, rstDatosGraficos.Fields(2)))
  End With
  Indice = Indice + 1
  rstDatosGraficos.MoveNext
Wend

rstDatosGraficos.Close
Set rstDatosGraficos = Nothing
rstInfo.Close
Set rstInfo = Nothing
db_local.Close
Set db_local = Nothing
CrearGráfico frmGraficosExcel.vntValorTipoGrafico, TipoInfo, numColumnas
End Select
End Sub
Sub CrearGráfico(ByVal vntTipoGraf, ByVal TipoInfo As Integer, ByVal columnaFinal As Long)
  Set g_oGráficoExcel = m_oLibroExcel.Charts.Add

  Select Case TipoInfo
    Case 0
      With g_oGráficoExcel
        .ChartType = vntTipoGraf
        .SizeWithWindow = True
        .SetSourceData Source:=m_oHojaExcel.Range("A2", Chr(columnaFinal + 64) & "3"), PlotBy:=xlRows

        .Location Where:=xlLocationAsNewSheet, Name:="ESCola"
        .HasTitle = True
        .ChartTitle.Characters.Text = "PERSONAS QUE ENTRAN Y SALEN DE LA COLA"

        With .Axes(xlCategory, xlPrimary)
          .HasTitle = True
          .AxisTitle.Characters.Text = "Horas"
        End With
        With .Axes(xlValue, xlPrimary)
          .HasTitle = True
          .AxisTitle.Characters.Text = "Cantidad de personas"
          .CategoryNames = m_oHojaExcel.Range("A1:" & Chr(columnaFinal + 64) & "1")
        End With

        .SeriesCollection(1).Name = "Entraron"
        .SeriesCollection(2).Name = "Salieron"

        .PlotArea.Interior.Color = RGB(192, 192, 255)
        .PlotArea.Border.Color = RGB(192, 192, 255)

      End With

    Case 1
      With g_oGráficoExcel
        .ChartType = vntTipoGraf
        .SizeWithWindow = True
        .SetSourceData Source:=m_oHojaExcel.Range("A2", Chr(columnaFinal + 64) & "2"), PlotBy:=xlRows

        .Location Where:=xlLocationAsNewSheet, Name:="PersCaj"
        .HasTitle = True

```



```
.ChartTitle.Characters.Text = "CANTIDAD DE PERSONAS ATENDIDAS POR CAJERO"

With .Axes(xlCategory, xlPrimary)
    .HasTitle = True
    .AxisTitle.Characters.Text = "Cajeros"
End With
With .Axes(xlValue, xlPrimary)
    .HasTitle = True
    .AxisTitle.Characters.Text = "Cantidad de personas"
    .CategoryNames = m_oHojaExcel.Range("A1:" & Chr(columnaFinal + 64) & "1")
End With

.SeriesCollection(1).Name = "Cajeros"

.PlotArea.Interior.Color = RGB(192, 192, 255)
.PlotArea.Border.Color = RGB(192, 192, 255)

End With

Case 2
With g_oGráficoExcel
    .ChartType = vntTipoGraf
    .SizeWithWindow = True
    .SetSourceData Source:=m_oHojaExcel.Range("A2", Chr(columnaFinal + 64) & "2"), PlotBy:=xlRows

    .Location Where:=xlLocationAsNewSheet, Name:="PersCaj"

If vntTipoGraf = xl3DPieExploded Then
    'Gráfico pastel
    .SeriesCollection(1).Name = "Transacciones"
    With .Axes(xlValue, xlPrimary)
        .CategoryNames = m_oHojaExcel.Range("A1:" & Chr(columnaFinal + 64) & "1")
    End With
    .SeriesCollection(1).ApplyDataLabels Type:=xlDataLabelsShowPercent
    .PlotArea.Interior.Color = RGB(192, 192, 255)
    .PlotArea.Border.Color = RGB(192, 192, 255)
Else
    .HasTitle = True
    .ChartTitle.Characters.Text = "TRANSACCIONES REALIZADAS"

    With .Axes(xlCategory, xlPrimary)
        .HasTitle = True
        .AxisTitle.Characters.Text = "Transacciones"
    End With
    With .Axes(xlValue, xlPrimary)
        .HasTitle = True
        .AxisTitle.Characters.Text = "Cantidad de transacciones"
        .CategoryNames = m_oHojaExcel.Range("A1:" & Chr(columnaFinal + 64) & "1")
    End With

    .SeriesCollection(1).Name = "Transacciones"

    .PlotArea.Interior.Color = RGB(192, 192, 255)
    .PlotArea.Border.Color = RGB(192, 192, 255)
End If

End With

Case 3
With g_oGráficoExcel
    .ChartType = vntTipoGraf
    .SizeWithWindow = True
    .SetSourceData Source:=m_oHojaExcel.Range("A2", Chr(columnaFinal + 64) & "3"), PlotBy:=xlRows

    .Location Where:=xlLocationAsNewSheet, Name:="ESCola"
    .HasTitle = True
    .ChartTitle.Characters.Text = "PROMEDIO DE ENTRADA/SALIDA EN COLA"

    With .Axes(xlCategory, xlPrimary)
```

```

        .HasTitle = True
        .AxisTitle.Characters.Text = "Horas"
    End With
    With .Axes(xlValue, xlPrimary)
        .HasTitle = True
        .AxisTitle.Characters.Text = "Tiempo (seg)"
        .CategoryNames = m_oHojaExcel.Range("A1:" & Chr(columnaFinal + 64) & "1")
    End With

    .SeriesCollection(1).Name = "Velocidad promedio de entrada"
    .SeriesCollection(2).Name = "Velocidad promedio de salida"

    .PlotArea.Interior.Color = RGB(192, 192, 255)
    .PlotArea.Border.Color = RGB(192, 192, 255)

End With

Case 4
With g_oGráficoExcel
    .ChartType = vntTipoGraf
    .SizeWithWindow = True
    .SetSourceData Source:=m_oHojaExcel.Range("A2", Chr(columnaFinal + 64) & "3"), PlotBy:=xlRows

    .Location Where:=xlLocationAsNewSheet, Name:="ESCola"
    .HasTitle = True
    .ChartTitle.Characters.Text = "PROMEDIO DE ENTRADA/SALIDA EN SERVICIO"

    With .Axes(xlCategory, xlPrimary)
        .HasTitle = True
        .AxisTitle.Characters.Text = "Horas"
    End With
    With .Axes(xlValue, xlPrimary)
        .HasTitle = True
        .AxisTitle.Characters.Text = "Tiempo (seg)"
        .CategoryNames = m_oHojaExcel.Range("A1:" & Chr(columnaFinal + 64) & "1")
    End With

    .SeriesCollection(1).Name = "Velocidad promedio de entrada"
    .SeriesCollection(2).Name = "Velocidad promedio de salida"

    .PlotArea.Interior.Color = RGB(192, 192, 255)
    .PlotArea.Border.Color = RGB(192, 192, 255)

End With

End Select

With m_oExcel
    .DisplayAlerts = False
    m_oLibroExcel.SaveAs _
        App.Path & "\Graficos", xlNormal
    m_oLibroExcel.Close
    Set m_oLibroExcel = Nothing
End With
End Sub
Public Sub HacerTab(ByVal Tecla As Integer)
    If Tecla = 13 Then
        SendKeys "{TAB}"
    End If
End Sub
Function fncGenerarTransacciones(ByVal NumCliente As Integer, _
    ByVal intDia As Integer, _
    ByVal strFecha As String, _
    ByVal intNumCaj As Integer, _
    ByVal intHora As Integer) As Double

Dim db_local As DAO.Database
Dim rsTransac As DAO.Recordset
Dim rstFrecuencia As DAO.Recordset
Dim strSql$

```

```

Dim i&
Dim j&
Dim db_ListaTrans% 'Indica el número de transacciones en la base de datos
Dim arrayTransacciones() As typeTrans
Dim intNumTrans%
Dim intCodTrans%
Dim intCantCadaTrans%
Dim blnCodigoRepetido As Boolean
Dim intAcumTiempo&

intAcumTiempo = 0
fncGenerarTransacciones = 0
db_ListaTrans = 0
Set db_local = OpenDatabase(App.Path & "\VeVsSimuColas.mdb", False, False)
Set rstTransac = db_local.OpenRecordset("SELECT Count(Bco_Codigo) AS CuentaDeBco_Codigo " _
& "From PromTrans " _
& "WHERE Bco_Codigo = " & gCodBanco, dbOpenSnapshot)
With rstTransac
    If Not .EOF Then
        If Not IsNull(.Fields(0)) Then
            db_ListaTrans = .Fields(0)
        End If
    End If
    .Close
End With
Set rstTransac = Nothing

'Definir número de transacciones
Randomize
intNumTrans = Int((3 * Rnd) + 1)
ReDim arrayTransacciones(0 To intNumTrans - 1)

'Definir código de cada transaccion
blnCodigoRepetido = False
For i = 0 To intNumTrans - 1
NvoCodigoTransaccion:
    Randomize
    intCodTrans = Int((db_ListaTrans * Rnd) + 1)

    For j = 0 To intNumTrans - 1
        If arrayTransacciones(i).CodigoTransaccion = intCodTrans Then
            blnCodigoRepetido = True
        End If
    Next j

    If blnCodigoRepetido Then
        blnCodigoRepetido = False
        GoTo NvoCodigoTransaccion
    End If
    arrayTransacciones(i).CodigoTransaccion = intCodTrans
Next i

'Definir cantidad de veces a realizar cada transacción
For i = 0 To intNumTrans - 1
    Randomize
    intCantCadaTrans = Int((3 * Rnd) + 1)
    arrayTransacciones(i).Cantidad = intCantCadaTrans
Next i

'Crear transacciones en la base de datos
For i = 0 To intNumTrans - 1
    intCantCadaTrans = arrayTransacciones(i).Cantidad
    While intCantCadaTrans > 0
        db_local.Execute "INSERT INTO CajerosTransacciones " _
& "(Bco_Codigo,Suc_Codigo,Dia,Fecha,NumCliente,NumCajero,NoTrans,Hora) " _
& "VALUES(" & gCodBanco & "," & gCodSucur & "," & intDia & "," & strFecha & "," _
& NumCliente & "," & intNumCaj & "," & arrayTransacciones(i).CodigoTransaccion & "," & intHora & ")"
    End While
Next i

```

```

    intCantCadaTrans = intCantCadaTrans - 1
Wend
Next i

Definir el tiempo en servicio basándose en todas las transacciones
For i = 0 To intNumTrans - 1
    Set rstTransac = db_local.OpenRecordset("SELECT TPromSegs " _
        & "From PromTrans " _
        & "WHERE Bco_Codigo=" & gCodBanco & " AND " _
        & "NoTrans=" & arrayTransacciones(i).CodigoTransaccion & ";", dbOpenSnapshot)
    If Not rstTransac.EOF Then
        If Not IsNull(rstTransac.Fields(0)) Then intAcumTiempo = intAcumTiempo + (rstTransac.Fields(0) *
arrayTransacciones(i).Cantidad)
        End If
        rstTransac.Close
    Next i

    fncGenerarTransacciones = intAcumTiempo
    Set rstTransac = Nothing
    db_local.Close
    Set db_local = Nothing
End Function

Funcion para generar el arrastre de los clientes
Sub ArrastreClientes()
    Dim i%
    Dim iii As Long, ttttiempo As Long

'Recorrer vector de imagenes
For i = frmSimulacion.imgCliente.LBound To (frmSimulacion.imgCliente.UBound - 2)
'Apagar imagenes
If frmSimulacion.imgCliente(i + 1).Visible Then
    frmSimulacion.imgCliente(i).Tag = frmSimulacion.imgCliente(i + 1).Tag

'Imagen activa
Select Case i
    Case 0 To 20
        If frmSimulacion.imgCliente(i).Tag = "R" Then
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(104, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(103, vbResBitmap)
        End If
    Case 21 To 24
        If frmSimulacion.imgCliente(i).Tag = "R" Then
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(108, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(105, vbResBitmap)
        End If
    Case 25 To 45
        If frmSimulacion.imgCliente(i).Tag = "R" Then
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(109, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(106, vbResBitmap)
        End If
    Case 46 To 50
        If frmSimulacion.imgCliente(i).Tag = "R" Then
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(110, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(107, vbResBitmap)
        End If
    Case 51 To 71
        If frmSimulacion.imgCliente(i).Tag = "R" Then
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(104, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i).Picture = LoadResPicture(103, vbResBitmap)
        End If
    End Select

    frmSimulacion.imgCliente(i).Visible = frmSimulacion.imgCliente(i + 1).Visible

```

```
'Retardo de traslado
For iii = 0 To (500000)
    ttttiempo = ttttiempo + 1
Next

frmSimulacion.imgCliente(i + 1).Visible = False

'Imagen anterior
Select Case (i + 1)
    Case 0 To 20
        If frmSimulacion.imgCliente(i + 1).Tag = "R" Then
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(104, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(103, vbResBitmap)
        End If
    Case 21 To 24
        If frmSimulacion.imgCliente(i + 1).Tag = "R" Then
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(108, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(105, vbResBitmap)
        End If
    Case 25 To 45
        If frmSimulacion.imgCliente(i + 1).Tag = "R" Then
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(109, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(106, vbResBitmap)
        End If
    Case 46 To 50
        If frmSimulacion.imgCliente(i + 1).Tag = "R" Then
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(110, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(107, vbResBitmap)
        End If
    Case 51 To 71
        If frmSimulacion.imgCliente(i + 1).Tag = "R" Then
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(104, vbResBitmap)
        Else
            frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(103, vbResBitmap)
        End If
    End Select

End If
Next i
'Apagar ultima imagen
If frmSimulacion.imgCliente(i + 1).Visible Then
    frmSimulacion.imgCliente(i + 1).Tag = "N"
    frmSimulacion.imgCliente(i + 1).Visible = False
    frmSimulacion.imgCliente(i + 1).Picture = LoadResPicture(103, vbResBitmap)
End If
End Sub
```

## **ModTiempo**

```
*****
*libreria de funciones para control de simulacion
*****
```

```
'Declaracion de variables
Public arrayCajero(0 To 14) As typeCajero
```

Funciones

```
'Verificar si hay cajero disponible
Function CajeroDisponible()
    Dim i%, Indice%
```

```
'Inicializacion
Indice = -1

'Recorrer vector
For i = LBound(arrayCajero, 1) To UBound(arrayCajero, 1)
  If arrayCajero(i).Habilitado Then
    If Not arrayCajero(i).Ocupado Then
      Indice = i: Exit For
    End If
  End If
Next

'Retornar valor
CajeroDisponible = Indice
End Function

'Vaciar cajero
Public Sub VaciarCajeros(ai_limitetiempo As Integer)
  Dim i%, ii%
  Dim Cadena, cadena2 As String

  Cadena = ""
  'Aumentar contadores
  For ii = LBound(arrayCajero, 1) To UBound(arrayCajero, 1)
    If arrayCajero(ii).Habilitado And arrayCajero(ii).Ocupado Then
      arrayCajero(ii).Contador = arrayCajero(ii).Contador + 1
      Cadena = Cadena & Str(arrayCajero(ii).Contador) & ","
      cadena2 = cadena2 & Str(arrayCajero(ii).Limite) & ","
    End If
  Next
  frmSimulacion.Text2 = Cadena
  frmSimulacion.Text3 = cadena2

  'Recorrer cajeros
  For i = LBound(arrayCajero, 1) To UBound(arrayCajero, 1)
    'Analizar que cajero se salio de su tiempo limite
    If arrayCajero(i).Contador >= arrayCajero(i).Limite Then
      'Liberar cajero
      With arrayCajero(i)
        .Contador = 0
        .Limite = 0
        .Ocupado = False
      End With

      'Desaparecer cliente
      frmSimulacion.imgAtendCajero(i).Visible = False
    End If
  Next
End Sub
```