

## INFORME TÉCNICO

Para ingresar al sistema el usuario debe estar registrado. Este usuario debe tener nombre completo y contraseña para verificar su identificación y permitir el acceso al sistema.

### 1.1 Ingreso al Sistema



**Figura 1.1** Ingreso al sistema.

**Fuente:** La Autora

**Ingresar:** Para ello digitar el nombre del usuario y la clave. Si el usuario no es el correcto sale un mensaje de error “usuario no correcto”, si la contraseña no es correcta muestra un mensaje de “contraseña inválida”. Presionar el botón Aceptar para ingresar al sistema.

Presionar el botón Salir para cerrar el sistema.

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmAutenticacion
    Public user As String
    Dim pass As String
    Dim cod As String

    Private Sub FrmAutenticacion_Load(sender As Object, e As EventArgs) Handles
Me.Load
```

```

        abrir()
    End Sub

    Private Sub btningreso_Click(sender As Object, e As EventArgs) Handles
btningreso.Click
        Try
            If usuarioRegistrado(txtnombre.Text) = True Then
                user = txtnombre.Text
                pass = contrasena(txtnombre.Text)
                cod = codigo(txtnombre.Text)
                lblcodigo.Text = cod
                If pass.Equals(txtclave.Text) = True Then
                    Me.Hide()
                    If ConsultarTipoUsuario(txtnombre.Text) = "A" Then
                        FrmEstandar.ShowDialog()
                    ElseIf ConsultarTipoUsuario(txtnombre.Text) = "C" Then
                        FrmInvitado.ShowDialog()
                    Else
                        MsgBox("Tipo de usuario no encontrado")
                    End If
                Else
                    MsgBox("Contraseña Invalida", MsgBoxStyle.Critical)
                End If
            Else
                MsgBox("El Usuario: " + txtnombre.Text + " no se encuentra
registrado")
            End If
        Catch ex As Exception
            MsgBox(ex.ToString)
        End Try
    End Sub

    Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
        End
    End Sub
End Class

```

## 1.2 Pantalla Inicio Administrador

Esta pantalla sólo se habilita el momento que un usuario de tipo administrador ha ingresado al sistema.



**Figura 1.2** Ventana Inicio Administrador.

**Fuente:** La Autora

Presenta la hora actual del sistema. Posee los siguientes botones:

- **Usuarios:** Abre la ventana de Control y Supervisión de los usuarios del sistema.
- **Empleados:** Abre la ventana de Control y Supervisión de los trabajadores que ingresarán por la puerta que tiene la chapa electrónica.
- **Alarma:** Abre la ventana de Control y Supervisión de la alarma.
- **Cortina:** Abre la ventana de Control y Supervisión de la cortina.
- **Puerta:** Abre la ventana de Control y Supervisión de la puerta.
- **Luces:** Abre la ventana de Control y Supervisión de las luces.
- **Salir:** Cierra el sistema.

La luz roja aparece en la pantalla de forma intermitente muestra que la puerta se encuentra abierta.



**Figura 1.3** Ventana Puerta Abierta.

**Fuente:** La Autora

La luz verde está encendida indica que la alarma se encuentra activada.



**Figura 1.4** Ventana Alarma Activada.

**Fuente:** La Autora

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmEstandar
    Public codiguito As String
    Public Luces, J, BandS, BA1, BandL, BandLA, L1, L2, L3, BL1, BL2, BL3, P1,
    PE, I, C1, C2, alarma, PA, CPA As Byte
    Public au11, au12, au13 As Byte
```

```

Public Fechabúsqueda As Date
Public horaabuscara, horadbuscar As String

Private Sub FrmEstandar_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    alarma = 0

    'TODO: esta línea de código carga datos en la tabla
    'BDDdomoDataSet.prog_alarma' Puede moverla o quitarla según sea necesario.
    Me.Prog_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.prog_alarma)
    'TODO: esta línea de código carga datos en la tabla
    'BDDdomoDataSet.alarma_evento' Puede moverla o quitarla según sea necesario.
    Me.Alarma_eventoTableAdapter.Fill(Me.BDDdomoDataSet.alarma_evento)
    'TODO: esta línea de código carga datos en la tabla
    'BDDdomoDataSet.ingreso_trabajador' Puede moverla o quitarla según sea necesario.
    Me.Ingreso_trabajadorTableAdapter.Fill(Me.BDDdomoDataSet.ingreso_trabajador)
    AxWinsock1.Protocol = MSWinsockLib.ProtocolConstants.sckTCPProtocol
    AxWinsock1.RemoteHost = "10.10.100.254"
    AxWinsock1.RemotePort = 8899
    AxWinsock1.Connect()
    Luces = 0
    I = 0
    J = 0
    BA1 = 0
    BandS = 0
    BandL = 0
    BandLA = 0
    PA = 0
    BL1 = 0
    BL2 = 0
    BL3 = 0
    L1 = 0
    L2 = 0
    L3 = 0
    P1 = 0
    PE = 0
    C1 = 0
    C2 = 0
    aul1 = 0
    aul2 = 0
    aul3 = 0
End Sub

Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles btnsalir.Click
    AxWinsock1.Close()
End Sub

Private Sub btnluces_Click(sender As Object, e As EventArgs) Handles btnluces.Click
    FrmLuces.Show()
End Sub

Private Sub btnpuerta_Click(sender As Object, e As EventArgs) Handles btnpuerta.Click
    AxWinsock1.SendData("0")
    FrmPuerta.Show()
End Sub

```

```

Private Sub btnusuarios_Click(sender As Object, e As EventArgs) Handles
btnusuarios.Click
    FrmUsuarios.Show()

End Sub

Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
    lblhora.Text = TimeOfDay

End Sub

Private Sub BtnAlarma_Click(sender As Object, e As EventArgs) Handles
BtnAlarma.Click
    FrmProgAlarma.Show()

End Sub

Private Sub btnEmpleados_Click(sender As Object, e As EventArgs) Handles
btnEmpleados.Click
    FrmTrabajadores.Show()

End Sub

Private Sub btncortinas_Click(sender As Object, e As EventArgs) Handles
btncortinas.Click
    FrmCortina.Show()

End Sub

Private Sub Timer2_Tick(sender As Object, e As EventArgs) Handles Timer2.Tick
    If BandS = 2 Then
        BandS = 3
    End If
    If BandS = 1 Then
        AxWinsock1.SendData("S")
        BandS = 2
    End If
    If BandS = 0 Then
        AxWinsock1.SendData("S")
        BandS = 1
    End If
    '***** Abre la Puerta
    If P1 = 2 Then
        AxWinsock1.SendData("O")

        P1 = 0
    End If
    '***** Si la clave es incorrecta
    If PE = 2 Then
        AxWinsock1.SendData("E")
        PE = 0
    End If
    '***** Si la puerta esta abierta
    If PA = 1 Then
        If CPA = 0 Then
            lblLedPuerta.Text = "Puerta Abierta"
            LedPuerta.Image =
Image.FromFile("C:\Users\Lupita\Documents\Visual Studio
2012\Projects\WindowsApplication1\imagenes\Led Rectangular Rojo Encendido 1.bmp")
        End If
        If CPA = 1 Then
            lblLedPuerta.Text = ""
            LedPuerta.Image =
Image.FromFile("C:\Users\Lupita\Documents\Visual Studio
2012\Projects\WindowsApplication1\imagenes\Led Rectangular Rojo.bmp")

```

```

    End If
    CPA = CPA + 1
    If CPA > 1 Then
        CPA = 0
    End If
End If
'***** Activa la Alarma
If BA1 = 2 Then
    AxWinsock1.SendData("a")
    BA1 = 0
End If

If BA1 = 4 Then
    AxWinsock1.SendData("d")
    BA1 = 0
End If

'***** Enciende Todas Las Luces
If BandL = 2 Then
    J = J + 1
    If J = 1 Then
        AxWinsock1.SendData("B")
        FrmLuces.btnluz1.Text = "Luz 1 On"
        L1 = 1
    End If
    If J = 2 Then
        AxWinsock1.SendData("D")
        FrmLuces.btnluz2.Text = "Luz 2 On"
        L2 = 1
    End If
    If J = 3 Then
        AxWinsock1.SendData("G")
        FrmLuces.btnluz3.Text = "Luz 3 On"
        L3 = 1
        BandL = 0
        J = 0
    End If
End If

'***** Apaga Todas Las Luces
If BandLA = 2 Then
    J = J + 1
    If J = 1 Then
        AxWinsock1.SendData("C")
        FrmLuces.btnluz1.Text = "Luz 1 Off"
        L1 = 0
    End If
    If J = 2 Then
        AxWinsock1.SendData("F")
        FrmLuces.btnluz2.Text = "Luz 2 Off"
        L2 = 0
    End If
    If J = 3 Then
        AxWinsock1.SendData("H")
        FrmLuces.btnluz3.Text = "Luz 3 Off"
        L3 = 0
        BandLA = 0
        J = 0
    End If
End If
'***** Enciende La Cortina
If C1 = 2 Then

```

```

        AxWinsock1.SendData("K")
        C1 = 0
    End If
    '***** Apaga La Cortina
    If C1 = 5 Then
        AxWinsock1.SendData("L")
        C1 = 0
    End If
    If C2 = 2 Then
        AxWinsock1.SendData("M")
        C2 = 0
    End If
    If C2 = 5 Then
        AxWinsock1.SendData("N")
        C2 = 0
    End If
    '***** Enciende Luz 1
    If BL1 = 2 Then
        AxWinsock1.SendData("B")

        BL1 = 0
    End If
    '***** Enciende Luz 2
    If BL2 = 2 Then
        AxWinsock1.SendData("D")
        BL2 = 0
    End If
    '***** Enciende Luz 3
    If BL3 = 2 Then
        AxWinsock1.SendData("G")
        BL3 = 0
    End If

    If (L1 = 1) Or (L2 = 1) Or (L3 = 1) Then
        FrmLuces.btntodas.Text = "Apagar Todas"
        Luces = 1
    Else
        FrmLuces.btntodas.Text = "Encender Todas"
        Luces = 0
    End If

    '***** Apaga Luz 1
    If BL1 = 5 Then
        AxWinsock1.SendData("C")
        BL1 = 0
    End If
    '***** Apaga Luz 2
    If BL2 = 5 Then
        AxWinsock1.SendData("F")
        BL2 = 0
    End If
    '***** Apaga Luz 3
    If BL3 = 5 Then
        AxWinsock1.SendData("H")
        BL3 = 0
    End If

    If BandL = 1 Then
        BandL = 2
    End If
    If BandLA = 1 Then
        BandLA = 2
    End If

```



```

End If
If BL1 = 1 Then
    BL1 = 2
End If
If BL2 = 1 Then
    BL2 = 2
End If
If BL3 = 1 Then
    BL3 = 2
End If
If BL1 = 4 Then
    BL1 = 5
End If
If BL2 = 4 Then
    BL2 = 5
End If
If BL3 = 4 Then
    BL3 = 5
End If
If BA1 = 1 Then
    BA1 = 2
End If
If P1 = 1 Then
    P1 = 2
End If
If PE = 1 Then
    PE = 2
End If
If C1 = 1 Then
    C1 = 2
End If
If C1 = 4 Then
    C1 = 5
End If
If C1 = 1 Then
    C1 = 2
End If
If C2 = 4 Then
    C2 = 5
End If
If C2 = 1 Then
    C2 = 2
End If
If (C1 = 0) And (C2 = 0) And (PE = 0) And (P1 = 0) And (BA1 = 0) And
(BandS = 3) And (BandL = 0) And (BandLA = 0) And (BL1 = 0) And (BL2 = 0) And (BL3
= 0) Then
    AxWinsock1.SendData(".")
End If
End Sub

```

```

Private Sub AxWinsock1_DataArrival(sender As Object, e As
AxMSWinsockLib.DMSWinsockControlEvents_DataArrivalEvent) Handles
AxWinsock1.DataArrival
    Dim StrData As String
    StrData = ""
    AxWinsock1.GetData(StrData)
    TxtWsk.Text = StrData
    'Procedimiento de ingreso de la clave de la Puerta
    If Len(TxtWsk.Text) > 3 Then
        TxtPassword.Text = ""
        I = 2
    End If
End Sub

```

```

Do
    If Len(TxtPassword.Text) = 0 Then
        TxtPassword.Text = Mid(TxtWsk.Text, I, 1)
    Else
        TxtPassword.Text = TxtPassword.Text & Mid(TxtWsk.Text, I, 1)
    End If
    I = I + 1

    Loop Until (Mid(TxtWsk.Text, I, 1) = "#")
End If
If I > 2 Then ' graba el acceso por la puerta en la bdd
    If clavepuertaregistrado(TxtPassword.Text) Then ' verifica la clave
de acceso a la puerta
        P1 = 1
        Dim horaa As TimeSpan
        Dim dato As Integer
        Dim codigoingresopuerta As String
        dato = CInt(Int((10000 * Rnd()) + 1))
        codigoingresopuerta = Str(dato)
        Dim codigotrabaj As String =
obtcodigotrabajador(TxtPassword.Text) ' obtiene el codigo del trabajador
        TxtPassword.Text = ""
        horaa = TimeSpan.Parse(DateAndTime.TimeString())
        While codigoRegistradoIngPuerta(codigoingresopuerta) = True
            dato = CInt(Int((10000 * Rnd()) + 1))
            codigoingresopuerta = Str(dato)
        End While
        Me.Ingreso_trabajadorTableAdapter.Insert(codigoingresopuerta,
codigotrabaj, DateAndTime.Today, horaa)
    Else
        PE = 1
        TxtPassword.Text = ""
    End If
End If
If TxtWsk.Text = "4" Then 'Si la puierta se abrio
    PA = 1
End If
If TxtWsk.Text = "5" Then 'Si la puierta se cerro
    LblLedPuerta.Text = "Puerta Cerrada"
    LedPuerta.Image = Image.FromFile("C:\Users\Lupita\Documents\Visual
Studio 2012\Projects\WindowsApplication1\imagenes\Led Rectangular Rojo.bmp")
    PA = 0
    CPA = 0
End If
If TxtWsk.Text = "c" Then 'Si la alarma se activo

    Dim horaa, horad As TimeSpan
    Dim codigoalmacenar As String

    Dim dato As Integer
    LblLedAlarma.Text = "Alarma Activada"
    LedAlarma.Image = Image.FromFile("C:\Users\Lupita\Documents\Visual
Studio 2012\Projects\WindowsApplication1\imagenes\Led Rectangular Verde
Encendido.bmp")
    dato = CInt(Int((10000 * Rnd()) + 1))
    horaa = TimeSpan.Parse(DateAndTime.TimeString())
    horad = TimeSpan.Parse("00:00")
    codigoalmacenar = "A" + Str(dato)
    codiguito = FrmAutenticacion.Lblcodigo.Text

    Try
        While codigoRegistradoAlarma(codigoalmacenar) = True

```

```

        dato = CInt(Int((10000 * Rnd()) + 1))
        codigoalmacenar = "A" + Str(dato)
    End While

    Me.Prog_alarmaTableAdapter.Insert(codigoalmacenar, codiguito,
DateAndTime.Today, horaa, horad)
    Me.Prog_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.prog_alarma)

    Catch ex As Exception
        MsgBox(ex.ToString)
    End Try
End If
If TxtWsk.Text = "b" Then 'Si la alarma se desactivo
    Dim horaa, horad As TimeSpan
    Dim codigoalmacenar As String

    Dim dato As Integer
    LblLedAlarma.Text = "Alarma Desactivada"
    LedAlarma.Image = Image.FromFile("C:\Users\Lupita\Documents\Visual
Studio 2012\Projects\WindowsApplication1\imagenes\Led Rectangular Verde.bmp")
    dato = CInt(Int((10000 * Rnd()) + 1))
    horad = TimeSpan.Parse(DateAndTime.TimeString())
    horaa = TimeSpan.Parse("00:00")
    codigoalmacenar = "A" + Str(dato)
    codiguito = FrmAutenticacion.Lblcodigo.Text
    Try
        While codigoRegistradoAlarmar(codigoalmacenar) = True
            dato = CInt(Int((10000 * Rnd()) + 1))
            codigoalmacenar = "A" + Str(dato)
        End While
        Me.Prog_alarmaTableAdapter.Insert(codigoalmacenar, codiguito,
DateAndTime.Today, horaa, horad)
        Me.Prog_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.prog_alarma)
    Catch ex As Exception
        MsgBox(ex.ToString)
    End Try
End If
If TxtWsk.Text = "1" Then 'Si se activo la Zona1
    Dim horaactual As TimeSpan
    Dim fechactual As Date
    horaactual = TimeSpan.Parse(DateAndTime.TimeString())
    fechactual = DateAndTime.Today()
    Dim codigoalar As String
    Dim dato As Integer

    dato = CInt(Int((10000 * Rnd()) + 1))
    codigoalar = "A1" + Str(dato)
    While codigoRegistradoAlarmaEvento(codigoalar) = True
        dato = CInt(Int((10000 * Rnd()) + 1))
        codigoalar = "A1" + Str(dato)
    End While

    Me.Alarma_eventoTableAdapter.Insert(codigoalar, fechactual,
horaactual, "activado", "on", "off", "off")
    MsgBox("Alerta de Intrusión Zona 1", MsgBoxStyle.Critical)
End If
If TxtWsk.Text = "2" Then 'Si se activo la Zona2
    Dim horaactual As TimeSpan
    Dim fechactual As Date

```

```

horaactual = TimeSpan.Parse(DateAndTime.TimeString())
fechactual = DateAndTime.Today()
Dim codigoalar As String
Dim dato As Integer

dato = CInt(Int((10000 * Rnd()) + 1))
codigoalar = "A1" + Str(dato)

While codigoRegistradoAlarmaEvento(codigoalar) = True
    dato = CInt(Int((10000 * Rnd()) + 1))
    codigoalar = "A1" + Str(dato)
End While
Me.Alarma_eventoTableAdapter.Insert(codigoalar, fechactual,
horaactual, "activado", "off", "on", "off")
MsgBox("Alerta de Intrusión Zona 2", MsgBoxStyle.Critical)
End If
If TxtWsk.Text = "3" Then 'Si se activo la Zona3
    Dim horaactual As TimeSpan
    Dim fechactual As Date
    horaactual = TimeSpan.Parse(DateAndTime.TimeString())
    fechactual = DateAndTime.Today()
    Dim codigoalar As String
    Dim dato As Integer

    dato = CInt(Int((10000 * Rnd()) + 1))
    codigoalar = "A1" + Str(dato)

    While codigoRegistradoAlarmaEvento(codigoalar) = True
        dato = CInt(Int((10000 * Rnd()) + 1))
        codigoalar = "A1" + Str(dato)
    End While
    Me.Alarma_eventoTableAdapter.Insert(codigoalar, fechactual,
horaactual, "activado", "off", "off", "on")
    MsgBox("Alerta de Intrusión Zona 3", MsgBoxStyle.Critical)
End If
'codigo para controlar las luces del interruptor
If TxtWsk.Text = "e" Then
    L1 = 1
    FrmLuces.btnluz1.Text = "Luz 1 On"
End If
If TxtWsk.Text = "i" Then
    L1 = 0
    FrmLuces.btnluz1.Text = "Luz 1 Off"
End If
If TxtWsk.Text = "g" Then
    L2 = 1
    FrmLuces.btnluz2.Text = "Luz 2 On"
End If
If TxtWsk.Text = "j" Then
    L2 = 0
    FrmLuces.btnluz2.Text = "Luz 2 Off"
End If
If TxtWsk.Text = "h" Then
    L3 = 1
    FrmLuces.btnluz3.Text = "Luz 3 On"
End If
If TxtWsk.Text = "k" Then
    L3 = 0
    FrmLuces.btnluz3.Text = "Luz 3 Off"
End If
If TxtWsk.Text = "?" Then
    AxWinsock1.SendData("?")

```

```

    End If
End Sub

Private Sub Ingreso_trabajadorBindingNavigatorSaveItem_Click(sender As
Object, e As EventArgs)
    Me.Validate()
    Me.Ingreso_trabajadorBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.BDDdomoDataSet)

End Sub

Private Sub Timer3_Tick(sender As Object, e As EventArgs) Handles Timer3.Tick
    Dim fec As String
    Fechabusqueda = DateTime.Today
    Dim hhha, hhhd As Date
    fec = Fechabusqueda.ToString("MM-dd-yyyy")
    hhha = DateTime.TimeOfDay()
    hhhd = DateTime.TimeOfDay()
    horaabuscar = hhha.ToString("HH:mm")
    horabuscar = hhhd.ToString("HH:mm")
    'codigo para horario de la alarma

    While Fechabusa(fec) = True
        While horaactivado(horaabuscar) = True
            BA1 = 2
            Exit While
        End While
        Exit While
    End While

    While Fechabusa(fec) = True
        While horadesactivado(horabuscar) = True
            BA1 = 4
            Exit While
        End While
        Exit While
    End While
    'codigo para horario de las luces
    'codigo para encendido
    While Fechabus1(fec) = True
        While horaluza(horaabuscar) = True
            If luz1() = True Then
                BL1 = 1
            End If
            Exit While
        End While
        Exit While
    End While

    While Fechabus1(fec) = True
        While horaluza(horaabuscar) = True
            If luz2() = True Then
                BL2 = 1
            End If
            Exit While
        End While
        Exit While
    End While

    While Fechabus1(fec) = True
        While horaluza(horaabuscar) = True
            If luz3() = True Then
                BL3 = 1
            End If
        End While
    End While

```

```

        End If
        Exit While
    End While
    Exit While
End While
'codigo para apagado se apagan todas las luces
While horalf(horadbuscar) = True
    BL1 = 4
    BL2 = 4
    BL3 = 4

    Exit While
End While
'codigo para horario de la cortina

While Fechabuscortina(fec) = True
    While horacortinaabre(horaabuscar) = True
        C1 = 2
        Exit While
    End While
    Exit While
End While

While Fechabuscortina(fec) = True
    While horacortinacierre(horadbuscar) = True
        C1 = 5
        Exit While
    End While
    Exit While
End While
End Sub
End Class

```

### 1.3 Ventana Usuarios

Administra los usuarios del sistema para su ingreso al mismo.



Figura 1.5 Ventana gestión de usuarios.

Fuente: La Autora

Esta ventana tiene los siguientes botones:

- **Guardar:** Para ingresar un nuevo usuario al sistema.
  - A) Asignarle un código, este debe iniciar con la letra "U" para que se identifique que el registro pertenece a un usuario del sistema.
  - B) Escribir el nombre completo permite hasta 50 caracteres.
  - C) Escribir una clave de acceso permite hasta 10 caracteres alfanuméricos.
  - D) Escribir el tipo de usuario ya sea administrador la letra "A" o general la letra "C".
- **Editar:** Un usuario existente, escribir el código de un usuario existente en la casilla correspondiente. Puede visualizar los códigos de usuario en una tabla que contiene los usuarios registrados en la base de datos, esta se encuentra en la parte inferior de la ventana. Cada campo a modificarse es verificado para que la información a cambiarse sea correcta, es necesario volver a escribir todos los campos para una adecuada modificación.
- **Eliminar:** Un usuario al ingresar el código del usuario, no es necesario ingresar ningún otro dato.
- **Buscar:** Al ingresar un código de usuario se muestra toda la información del mismo permitiendo verificar si existe o no.
- **Salir:** Regresa al formulario estándar para escoger otra opción si es necesario.

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmUsuarios

    Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
        Close()
    End Sub

    Private Sub btnclick_Click(sender As Object, e As EventArgs) Handles
btnclick.Click
        If TxtNombre.Text = "" Then
            MsgBox("Ingrese un nombre válido")
        ElseIf TxtClave.Text = "" Then
            MsgBox("Ingrese clave válida")
        ElseIf TxtTipo.Text = "" Then
```

```

        MsgBox("Ingrese un tipo de usuario válido")
    Else
        Me.Usuario_sistemaTableAdapter1.EditarUsuarioSistema(TxtCodigo.Text,
        TxtNombre.Text, TxtClave.Text, TxtTipo.Text)

Me.Usuario_sistemaTableAdapter1.Fill(Me.BDDdomoDataSet.usuario_sistema)
        TxtCodigo.Text = ""
        TxtNombre.Text = ""
        TxtClave.Text = ""
        TxtTipo.Text = ""
    End If
End Sub

Private Sub FrmUsuarios_Load(sender As Object, e As EventArgs) Handles
 MyBase.Load
    Me.Usuario_sistemaTableAdapter1.Fill(Me.BDDdomoDataSet.usuario_sistema)

End Sub

Private Sub btnguardar_Click(sender As Object, e As EventArgs) Handles
btnguardar.Click
    Try
        If codigoRegistrado(TxtCodigo.Text) = True Then
            MsgBox("Usuario ya esta registrado")

            Else
                Me.Usuario_sistemaTableAdapter1.Insert(TxtCodigo.Text,
                TxtNombre.Text, TxtClave.Text, TxtTipo.Text)

Me.Usuario_sistemaTableAdapter1.Fill(Me.BDDdomoDataSet.usuario_sistema) 'carga
datos en la tabla
                TxtCodigo.Text = ""
                TxtNombre.Text = ""
                TxtClave.Text = ""
                TxtTipo.Text = ""
            End If
        Catch ex As Exception
            MsgBox(ex.ToString)
        End Try
    End Sub

Private Sub Usuario_sistemaBindingNavigatorSaveItem_Click(sender As Object, e
As EventArgs)
    Me.Validate()
    Me.Usuario_sistemaBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.BDDdomoDataSet)

End Sub

Private Sub btneliminar_Click(sender As Object, e As EventArgs) Handles
btneliminar.Click
    If TxtCodigo.Text = "" Then
        MsgBox("Ingrese codigo por favor")
    Else
        Me.Usuario_sistemaTableAdapter1.Eliminarprogalarma(TxtCodigo.Text)
        Me.Usuario_sistemaTableAdapter1.Eliminarprogcortina(TxtCodigo.Text)
        ' Me.Usuario_sistemaTableAdapter1.Eliminarusralarma(TxtCodigo.Text)
        ' Me.Usuario_sistemaTableAdapter1.Eliminarusrcortina(TxtCodigo.Text)
        Me.Usuario_sistemaTableAdapter1.Eliminarusuerdeluces(TxtCodigo.Text)
        'Me.Usuario_sistemaTableAdapter1.Eliminarusrлуces(TxtCodigo.Text)
        ' Me.Usuario_sistemaTableAdapter1.Eliminarusrac1prog(TxtCodigo.Text)
    End If
End Sub

```



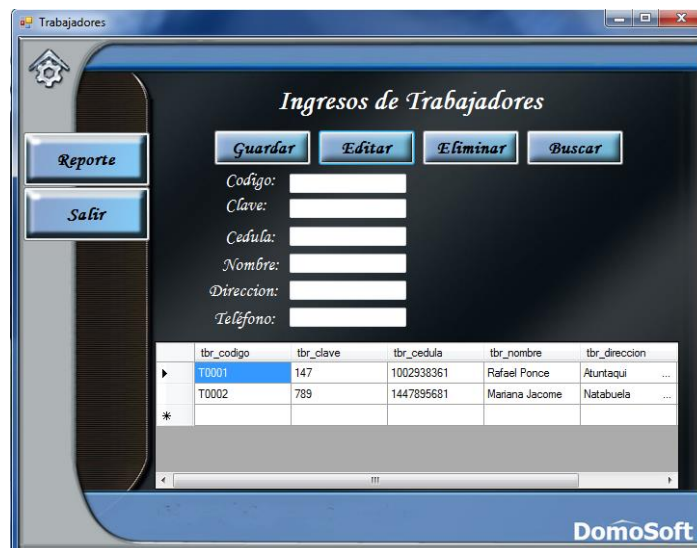
```

        Me.Usuario_sistemaTableAdapter1.Eliminarusuarsistema(TxtCodigo.Text)
Me.Usuario_sistemaTableAdapter1.Fill(Me.BDDdomoDataSet.usuario_sistema)
    TxtCodigo.Text = ""
    End If
End Sub

Private Sub btnbuscar_Click(sender As Object, e As EventArgs) Handles
btnbuscar.Click
    If TxtCodigo.Text = "" Then
        MsgBox("ingrese código por favor")
    Else
Me.Usuario_sistemaTableAdapter1.FillBy(Me.BDDdomoDataSet.usuario_sistema,
TxtCodigo.Text)
    End If
End Sub
End Class

```

#### 1.4. Ventana Trabajadores



**Figura 1.6** Ventana gestión de Trabajadores.

**Fuente:** La Autora

Esta ventana tiene los siguientes botones:

- **Guardar:** Para ingresar un nuevo trabajador al sistema.
  - A) Asignarle un código, este debe iniciar con la letra “T” para que se identifique que el registro pertenece a un trabajador.
  - B) Escriba la cédula del empleado con un máximo de 15 caracteres.
  - C) Escriba el nombre completo del trabajador, hasta 50 caracteres.

- D) Escriba la dirección del empleado hasta 70 caracteres.
- E) Escriba el teléfono del trabajador hasta 15 caracteres.

Todos estos campos son obligatorios ya que si algún momento pasa un seceso inesperado se puede comunicar y localizar al trabajador.

- **Editar:** Un trabajador existente, cada campo a modificarse es verificado para que la información a cambiarse sea correcta, es necesario volver a escribir todos los campos para una adecuada modificación.
- **Eliminar:** Un empleado al ingresar el código del trabajador, no es necesario ingresar ningún otro dato.
- **Buscar:** Al ingresar un código de trabajador se muestra toda la información del mismo permitiendo verificar si existe o no.
- **Salir:** Regresa al formulario estándar para escoger otra opción si es necesario.
- **Reporte:** Muestra los ingresos que se han hecho por la puerta electrónica.
  - A) Seleccionar la fecha a detallar los ingresos de los trabajadores por la puerta
  - B) Clic en el botón Fecha Ok.
  - C) Otra opción es dar clic en el botón reporte Ok para visualizar todo el reporte de ingresos.

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmTrabajadores

    Private Sub FrmTrabajadores_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        'TODO: esta línea de código carga datos en la tabla
        'BDDdomoDataSet.Trabajador' Puede moverla o quitarla según sea necesario.
        Me.TrabajadorTableAdapter.Fill(Me.BDDdomoDataSet.Trabajador)

    End Sub

    Private Sub btnguardar_Click(sender As Object, e As EventArgs) Handles btnguardar.Click
        Try
            If codigoRegistradoTrabajador(TxtCodigo.Text) = True Then
                MsgBox("Trabajador ya esta registrado")
            Else
```

```

        Me.TrabajadorTableAdapter.Insert(TxtCodigo.Text, txtclave.Text,
txtcedula2.Text, TxtNombre.Text, TxtDireccion.Text, TxtTelefono.Text)
        Me.TrabajadorTableAdapter.Fill(Me.BDDdomoDataSet.Trabajador)
'carga datos en la tabla
        TxtCodigo.Text = ""
        txtclave.Text = ""
        txtcedula2.Text = ""
        TxtNombre.Text = ""
        TxtDireccion.Text = ""
        TxtTelefono.Text = ""
    End If
    Catch ex As Exception
        MsgBox(ex.ToString)
    End Try

End Sub
Private Sub btneditar_Click(sender As Object, e As EventArgs) Handles
btneditar.Click
    If txtcedula2.Text = "" Then
        MsgBox("Ingrese una cédula válida")
    ElseIf txtclave.Text = "" Then
        MsgBox("Ingrese clave válido")
    ElseIf TxtNombre.Text = "" Then
        MsgBox("Ingrese nombre válido")
    ElseIf TxtDireccion.Text = "" Then
        MsgBox("Ingrese una dirección válida")
    Else
        Me.TrabajadorTableAdapter.Editartrabajador(TxtCodigo.Text,
txtclave.Text, txtcedula2.Text, TxtNombre.Text, TxtDireccion.Text,
TxtTelefono.Text)
        Me.TrabajadorTableAdapter.Fill(Me.BDDdomoDataSet.Trabajador)
        TxtCodigo.Text = ""
        TxtCedula.Text = ""
        TxtNombre.Text = ""
        TxtDireccion.Text = ""
        TxtTelefono.Text = ""
    End If
End Sub

Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
    Close()
End Sub

Private Sub btneliminar_Click(sender As Object, e As EventArgs) Handles
btneliminar.Click
    Me.TrabajadorTableAdapter.Eliminaringresopuerta(TxtNombre.Text)
    Me.TrabajadorTableAdapter.Eliminaringresotrabajador(TxtCodigo.Text)
    Me.TrabajadorTableAdapter.EliminarTrabajador(TxtCodigo.Text)
    Me.TrabajadorTableAdapter.Fill(Me.BDDdomoDataSet.Trabajador)
    TxtCodigo.Text = ""
End Sub

Private Sub TrabajadorBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs)
    Me.Validate()
    Me.TrabajadorBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.BDDdomoDataSet)

End Sub

```

```

Private Sub btnbuscar_Click(sender As Object, e As EventArgs) Handles
btnbuscar.Click
    Me.TrabajadorTableAdapter.FillBy(Me.BDDdomoDataSet.Trabajador,
TxtCodigo.Text)
End Sub

Private Sub btnreporte_Click(sender As Object, e As EventArgs) Handles
btnreporte.Click
    FrmReporteIngreso.Show()
End Sub

Private Sub TxtCodigo_TextChanged(sender As Object, e As EventArgs) Handles
TxtCodigo.TextChanged

End Sub

Private Sub TxtNombre_TextChanged(sender As Object, e As EventArgs) Handles
TxtNombre.TextChanged

End Sub

Private Sub TxtDireccion_TextChanged(sender As Object, e As EventArgs)
Handles TxtDireccion.TextChanged

End Sub

Private Sub TxtTelefono_TextChanged(sender As Object, e As EventArgs) Handles
TxtTelefono.TextChanged

End Sub

Private Sub LblCodigo_Click(sender As Object, e As EventArgs) Handles
LblCodigo.Click

End Sub

Private Sub LlblNombre_Click(sender As Object, e As EventArgs) Handles
LlblNombre.Click

End Sub

Private Sub LblDireccion_Click(sender As Object, e As EventArgs) Handles
LblDireccion.Click

End Sub

Private Sub LblTelefono_Click(sender As Object, e As EventArgs) Handles
LblTelefono.Click

End Sub

Private Sub lblcedula_Click(sender As Object, e As EventArgs) Handles
lblcedula.Click

End Sub

Private Sub TxtCedula_TextChanged(sender As Object, e As EventArgs) Handles
TxtCedula.TextChanged

End Sub
End Class

```



**Figura 1.7** Ventana para visualizar los ingresos de empleados por la puerta.

**Fuente:** La Autora

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmReporteIngreso

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
btncompleta.Click
        Me.Ingresos_puertaTableAdapter.Fill(Me.BDDdomoDataSet.ingresos_puerta)
    End Sub

    Private Sub FrmReporteIngreso_Load(sender As Object, e As EventArgs) Handles
 MyBase.Load
        'TODO: esta línea de código carga datos en la tabla
'BDDdomoDataSet.ingresos_puerta' Puede moverla o quitarla según sea necesario.
        Me.Ingresos_puertaTableAdapter.Fill(Me.BDDdomoDataSet.ingresos_puerta)

    End Sub

    Private Sub btnfecha_Click(sender As Object, e As EventArgs) Handles
btnfecha.Click
        Dim ordfecha As Date
        ordfecha = dtpfecha.Value
        Me.Ingresos_puertaTableAdapter.FillBy(Me.BDDdomoDataSet.ingresos_puerta,
ordfecha)
    End Sub

    Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
        Close()
    End Sub
End Class
```

## 1.5 Ventana Programación Alarma

El Administrador puede configurar horarios para la activación de la alarma. Contiene también las opciones para controlar la activación de la alarma y visualizar si hubo intrusión por alguna zona protegida por la alarma.

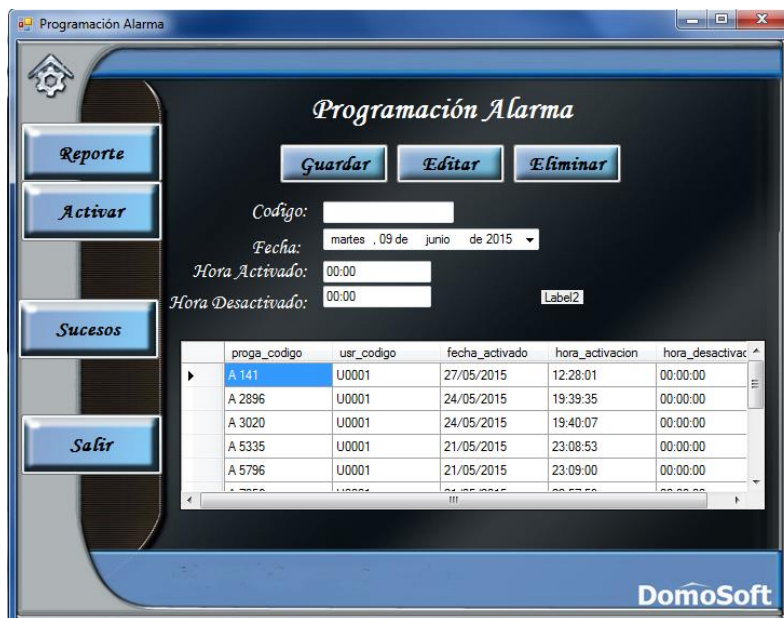


Figura 1.8 Ventana programación de la alarma.

Fuente: La Autora

En esta ventana se puede observar los siguientes botones:

- **Activar:** Si se lo presiona la alarma se activará inmediatamente.
- **Guardar:** Para guardar un nuevo horario de control de la alarma se debe:
  - A) Asignarle un código, este debe iniciar con la letra "A" para que se identifique que el registro pertenece a un horario de la alarma.
  - B) Escoger la fecha en la cual se va a activar el dispositivo.
  - C) Seleccionar la hora para activar la alarma, su formato debe ser 00:00 donde los dos primeros son la hora representada en veinticuatro horas y los dos siguientes son los minutos. Ejemplo, si desea que se active a las dos de la tarde deberá escribir 14:00, si desea que se active a las nueve de la mañana se debe digitar 09:00.
  - D) Seleccionar la hora para la desactivación de la alarma.

- **Editar:** Un horario existente, cada campo a modificarse es verificado para que la información a cambiarse sea correcta, es necesario volver a escribir todos los campos para una adecuada modificación.
- **Eliminar:** Un horario al ingresar el código del horario, no es necesario ingresar ningún otro dato.
- **Sucesos:** Abre una ventana donde se puede visualizar si ha pasado algo fuera de lo programado con la intrusión del sistema de alarma. Este indica la zona donde se rompió la seguridad, la hora y fecha del suceso.

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmProgAlarma
    Public codiguito As String
    Private Sub Prog_alarmaBindingNavigatorSaveItem_Click(sender As Object, e As EventArgs)
        Me.Validate()
        Me.Prog_alarmaBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.BDDdomoDataSet)

    End Sub

    Private Sub FrmProgAlarma_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        'TODO: esta línea de código carga datos en la tabla
        'BDDdomoDataSet.horario_alarma' Puede moverla o quitarla según sea necesario.
        Me.Horario_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.horario_alarma)
        'TODO: esta línea de código carga datos en la tabla
        'BDDdomoDataSet.prog_alarma' Puede moverla o quitarla según sea necesario.
        Me.Prog_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.prog_alarma)
        codiguito = FrmAutenticacion.Lblcodigo.Text
    End Sub

    Private Sub btnguardar_Click(sender As Object, e As EventArgs) Handles btnguardar.Click

        Try
            If codigoRegistradoAlarma(TxtCodigo.Text) = True Then
                MsgBox("Horario Alarma ya esta registrado")

            Else
                Dim horaa, horad As TimeSpan

                horaa = TimeSpan.Parse(txthorac.Text)
                horad = TimeSpan.Parse(txthorad.Text)
                txtcodigouser.Text = FrmAutenticacion.txtnombre.Text
                Me.Horario_alarmaTableAdapter.Insert(TxtCodigo.Text, codiguito, Dtfecha.Value, horaa, horad)

            End If

            Me.Horario_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.horario_alarma)
            TxtCodigo.Text = ""
            txthorac.Text = "00:00"
            txthorad.Text = "00:00"

        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
End Class
```

```

        End If
    Catch ex As Exception
        MsgBox(ex.ToString)
    End Try

End Sub

Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
    Close()

End Sub

Private Sub btnreporte_Click(sender As Object, e As EventArgs) Handles
btnreporte.Click
    FrmAlarma.Show()

End Sub

Private Sub btncodificar_Click(sender As Object, e As EventArgs) Handles
btncodificar.Click
    If TxtCodigo.Text = "" Then
        MsgBox("Ingrese un código válido")
    ElseIf txthorac.Text = "" Then
        MsgBox("Ingrese hora válida")
    ElseIf txthorad.Text = "" Then
        MsgBox("Ingrese hora válida")
    Else
        Me.Horario_alarmaTableAdapter.Editarhorarioalarma(TxtCodigo.Text,
codiguito, Dtfecha.Value, txthorac.Text, txthorad.Text)

        Me.Horario_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.horario_alarma)
        TxtCodigo.Text = ""
        txthorac.Text = "00:00"
        txthorad.Text = "00:00"
    End If
End Sub

Private Sub btneliminar_Click(sender As Object, e As EventArgs) Handles
btneliminar.Click
    Me.Horario_alarmaTableAdapter.Eliminaarhorario(TxtCodigo.Text)
    Me.Horario_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.horario_alarma)
End Sub

Private Sub btnsucesos_Click(sender As Object, e As EventArgs) Handles
btnsucesos.Click
    Frmseguridad.Show()
End Sub

Private Sub btnactivar_Click(sender As Object, e As EventArgs) Handles
btnactivar.Click

    TxtCodigo.Text = ""
    txthorac.Text = "00:00"
    txthorad.Text = "00:00"

    FrmEstandar.BA1 = 1
End Sub

Private Sub btndesactivar_Click(sender As Object, e As EventArgs) Handles
btndesactivar.Click

```



```

FrmEstandar.BA1 = 4
End Sub
End Class

```



**Figura 1.9** Formulario de reporte de intrusión activado por la alarma.

**Fuente:** La Autora

- **Salir:** Regresa al formulario estándar para escoger otra opción si es necesario.
- **Reporte:** Muestra los horarios que se han registrado en la BDD.
  - A) Seleccionar la fecha para detallar los horarios registrados en una fecha específica. Clic en el botón Fecha Ok.
  - B) Otra opción es dar clic en el botón reporte Ok para visualizar todos los registros.

El código fuente correspondiente a este formulario detallo a continuación:

```

Public Class Frmseguridad

    Private Sub Alarma_eventoBindingNavigatorSaveItem_Click(sender As Object, e
As EventArgs) Handles Alarma_eventoBindingNavigatorSaveItem.Click
        Me.Validate()
        Me.Alarma_eventoBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.BDDdomoDataSet)

    End Sub

```

```

Private Sub Frmseguridad_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    'TODO: esta línea de código carga datos en la tabla
    'BDDdomoDataSet.alarma_evento' Puede moverla o quitarla según sea necesario.
    Me.Alarma_eventoTableAdapter.Fill(Me.BDDdomoDataSet.alarma_evento)

End Sub

Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles btnsalir.Click
    Close()
End Sub

Private Sub btncompleta_Click(sender As Object, e As EventArgs) Handles btncompleta.Click
    Me.Alarma_eventoTableAdapter.Fill(Me.BDDdomoDataSet.alarma_evento)
End Sub

Private Sub btnfecha_Click(sender As Object, e As EventArgs) Handles btnfecha.Click
    Me.Alarma_eventoTableAdapter.FillBy(Me.BDDdomoDataSet.alarma_evento,
    dtpfechaact.Value)
End Sub
End Class

```

usr_nombre	fecha_activado	hora_activacion	usr_codigo	hora_desactiva
Saul Bolanos	27/05/2015	12:28:01	U0001	00:00:00
Saul Bolanos	24/05/2015	19:39:35	U0001	00:00:00
Saul Bolanos	24/05/2015	19:40:07	U0001	00:00:00
Saul Bolanos	21/05/2015	23:08:53	U0001	00:00:00
Saul Bolanos	21/05/2015	23:09:00	U0001	00:00:00
Saul Bolanos	21/05/2015	22:57:59	U0001	00:00:00
Saul Bolanos	29/05/2015	13:45:04	U0001	00:00:00
Saul Bolanos	27/05/2015	12:26:37	U0001	00:00:00

**Figura 1.10** Formulario de reporte de los eventos de la alarma.

**Fuente:** La Autora

El código fuente correspondiente a este formulario detallo a continuación:

```

Public Class FrmAlarma

    Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles btnsalir.Click

```

```

Close()
End Sub

Private Sub FrmAlarma_Load(sender As Object, e As EventArgs) Handles MyBase.Load
'TODO: esta línea de código carga datos en la tabla
'BDDdomoDataSet.usuario_alarma' Puede moverla o quitarla según sea necesario.
Me.Usuario_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.usuario_alarma)

End Sub

Private Sub btncompleta_Click(sender As Object, e As EventArgs) Handles btncompleta.Click
Me.Usuario_alarmaTableAdapter.Fill(Me.BDDdomoDataSet.usuario_alarma)
End Sub

Private Sub btnfecha_Click(sender As Object, e As EventArgs) Handles btnfecha.Click
Me.Usuario_alarmaTableAdapter.FillBy(Me.BDDdomoDataSet.usuario_alarma,
dtpfechaact.Value) 'muestra los horarios de activacion de la la alarma por fechas
End Sub
End Class

```

## 1.6 Ventana Programación de la Cortina

El administrador del sistema puede controlar los horarios de apertura y cierre de la cortina. Además puede directamente abrir o cerrar la cortina.

**Figura 1.11** Formulario para la programación de cortina.

**Fuente:** La Autora

Este formulario tiene los siguientes botones:

- **Abrir:** Al presionar esta opción se abre la cortina.

- **Cerrar:** Al presionar esta opción se cierra la cortina.
- **Salir:** Regresa al formulario anterior.
- **Guardar:** Para guardar un nuevo horario de control de la cortina se debe:
  - A) Asignarle un código, este debe iniciar con la letra “C” para que se identifique que el registro pertenece a un horario de la cortina.
  - B) Escoger la fecha en la cual se va a activar el dispositivo.
  - C) Seleccionar la hora para abrir la cortina, su formato debe ser 00:00 donde los dos primeros son la hora representada en veinticuatro horas y los dos siguientes son los minutos. Ejemplo, si desea que se abra a las dos de la tarde deberá escribir 14:00, si desea que se abra a las nueve de la mañana se debe digitar 09:00.
  - D) Seleccionar la hora para la cerrar la cortina.
- **Editar:** Un horario existente, cada campo a modificarse es verificado para que la información a cambiarse sea correcta, es necesario volver a escribir todos los campos para una adecuada modificación.
- **Eliminar:** Un horario al ingresar el código del horario, no es necesario ingresar ningún otro dato.
- **Reporte:** Muestra en otra ventana los horarios que se han registrado en la BDD.
  - A) Seleccionar la fecha para detallar los horarios registrados en una fecha específica.
  - B) Clic en el botón Fecha Ok.
  - C) Otra opción es dar clic en el botón reporte Ok para visualizar todos los registros.

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmCortina
    Public codiguito As String
    Public horaa, horad As TimeSpan

    Private Sub Prog_cortinaBindingNavigatorSaveItem_Click(sender As Object, e As EventArgs)
        Me.Validate()
        Me.Prog_cortinaBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.BDDdomoDataSet)
    End Sub
```

```

Private Sub FrmCortina_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
    'TODO: esta línea de código carga datos en la tabla
    'BDDdomoDataSet.prog_cortina' Puede moverla o quitarla según sea necesario.
    Me.Prog_cortinaTableAdapter.Fill(Me.BDDdomoDataSet.prog_cortina)
    codiguito = FrmAutenticacion.Lblcodigo.Text
End Sub

Private Sub btnguardar_Click(sender As Object, e As EventArgs) Handles
btnguardar.Click
    Try
        If codigoRegistradoCortina(TxtCodigo.Text) = True Then
            MsgBox("Horario Cortina ya esta registrado")

        Else
            horaa = TimeSpan.Parse(txthorab.Text)
            horad = TimeSpan.Parse(txthoracierre.Text)

            txtcodigouser.Text = FrmAutenticacion.txtnombre.Text
            Me.Prog_cortinaTableAdapter.Insert(TxtCodigo.Text, codiguito,
Dtfecha.Value, horaa, horad)
            Me.Prog_cortinaTableAdapter.Fill(Me.BDDdomoDataSet.prog_cortina)
            TxtCodigo.Text = ""
            txthorab.Text = "00:00"
            txthoracierre.Text = "00:00"
        End If
    Catch ex As Exception
        MsgBox(ex.ToString)
    End Try

End Sub

Private Sub btneditar_Click(sender As Object, e As EventArgs) Handles
btneditar.Click
    If TxtCodigo.Text = "" Then
        MsgBox("Ingrese un codigo válido")
    ElseIf txthorab.Text = "" Then
        MsgBox("Ingrese hora válida")
    ElseIf txthoracierre.Text = "" Then
        MsgBox("Ingrese hora válida")
    Else
        horaa = TimeSpan.Parse(txthorab.Text)
        horad = TimeSpan.Parse(txthoracierre.Text)
        Me.Prog_cortinaTableAdapter.Editarcortina(TxtCodigo.Text, codiguito,
Dtfecha.Value, txthorab.Text, txthoracierre.Text)
        Me.Prog_cortinaTableAdapter.Fill(Me.BDDdomoDataSet.prog_cortina)
        TxtCodigo.Text = ""
        txthorab.Text = "00:00"
        txthoracierre.Text = "00:00"
    End If
End Sub

Private Sub btneliminar_Click(sender As Object, e As EventArgs) Handles
btneliminar.Click
    Me.Prog_cortinaTableAdapter.EliminarCortina(TxtCodigo.Text)
    Me.Prog_cortinaTableAdapter.Fill(Me.BDDdomoDataSet.prog_cortina)
    TxtCodigo.Text = ""
    txthorab.Text = "00:00"
    txthoracierre.Text = "00:00"
End Sub

```

```

Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
    Close()
End Sub

Private Sub btnreporte_Click(sender As Object, e As EventArgs) Handles
btnreporte.Click
    FrmReporCortina.Show()
End Sub

Private Sub btnabrir_Click(sender As Object, e As EventArgs) Handles
btnabrir.Click
    FrmEstandar.C1 = 1
End Sub

Private Sub btncerrar_Click(sender As Object, e As EventArgs) Handles
btncerrar.Click
    FrmEstandar.C1 = 4
End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    FrmEstandar.C2 = 1
End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
    FrmEstandar.C2 = 4
End Sub
End Class

```



**Figura 1.12** Ventana reporte eventos de la cortina.

**Fuente:** La Autora

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmReporCortina
```

```
Private Sub FrmReporCortina_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
'TODO: esta línea de código carga datos en la tabla  
'BDDdomoDataSet.usuario_cortina' Puede moverla o quitarla según sea necesario.  
Me.Usuario_cortinaTableAdapter.Fill(Me.BDDdomoDataSet.usuario_cortina)
```

```
End Sub
```

```
Private Sub btnfecha_Click(sender As Object, e As EventArgs) Handles btnfecha.Click
```

```
Me.Usuario_cortinaTableAdapter.FillBy(Me.BDDdomoDataSet.usuario_cortina, dtpfechaact.Value)
```

```
' Me.Usuario_cortinaTableAdapter.Fill(Me.BDDdomoDataSet.usuario_cortina)
```

```
End Sub
```

```
Private Sub btncompleta_Click(sender As Object, e As EventArgs) Handles btncompleta.Click
```

```
Me.Usuario_cortinaTableAdapter.Fill(Me.BDDdomoDataSet.usuario_cortina)
```

```
End Sub
```

```
Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles btnsalir.Click
```

```
Close()
```

```
End Sub
```

```
End Class
```

## 1.7 Ventana Puerta

Al seleccionar el botón Puerta del menú de la Ventana inicial del sistema se abre la ventana para su respectivo control.

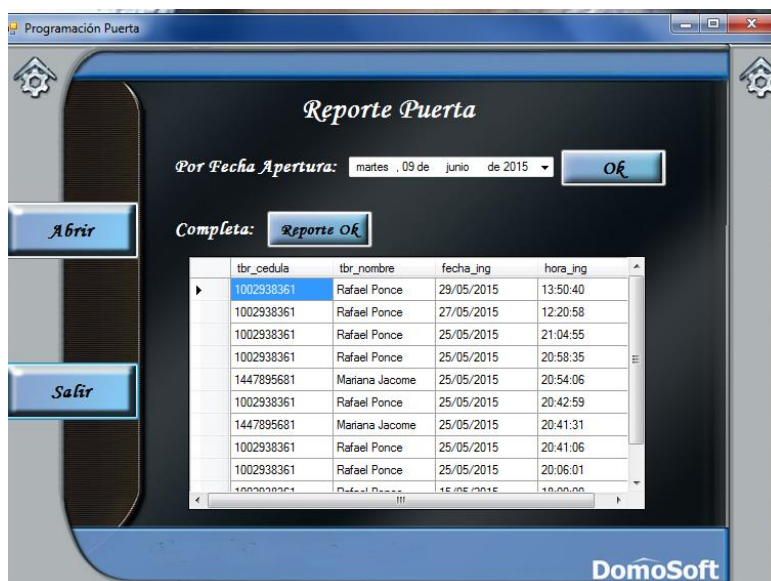


Figura 1.13 Ventana gestión de la puerta.

Fuente: La Autora

Posee el botón Abrir al presionarlo se abre la chapa electrónica.

Muestra los ingresos de la puerta que se han registrado en la BDD. Tiene do opciones:

- Seleccionar la fecha para detallar los datos registrados en una fecha específica. Clic en el botón Fecha Ok.
- Clic en el botón reporte Ok para visualizar todos los registros.

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmPuerta

    Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
        Close()
    End Sub

    Private Sub FrmPuerta_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        'TODO: esta línea de código carga datos en la tabla
'BDDdomoDataSet.ingreso_trabajador' Puede moverla o quitarla según sea necesario.
Me.Ingreso_trabajadorTableAdapter.Fill(Me.BDDdomoDataSet.ingreso_trabajador)
        'TODO: esta línea de código carga datos en la tabla
'BDDdomoDataSet.ingresos_puerta' Puede moverla o quitarla según sea necesario.
        Me.Ingresos_puertaTableAdapter.Fill(Me.BDDdomoDataSet.ingresos_puerta)

    End Sub

    Private Sub btnfecha_Click(sender As Object, e As EventArgs) Handles
btnfecha.Click

        Me.Ingresos_puertaTableAdapter.FillBy(Me.BDDdomoDataSet.ingresos_puerta,
dtpfechaact.Value)
    End Sub

    Private Sub btncompleta_Click(sender As Object, e As EventArgs) Handles
btncompleta.Click
        Me.Ingresos_puertaTableAdapter.Fill(Me.BDDdomoDataSet.ingresos_puerta)
    End Sub

    Private Sub btnabrir_Click(sender As Object, e As EventArgs) Handles
btnabrir.Click
        FrmEstandar.P1 = 1

    End Sub
End Class
```



## 1.8 Ventana Programación de la Iluminación

Permite controlar el encendido y apagado de las luces sea directamente al presionar en los botones o al ser registrado en un horario para su ejecución.

progl_codigo	usr_codigo	fecha_activacion	hora_encendido	hora_apagado	luz1
L00001	U0001	18/05/2015	17:00:00	22:00:00	on
L00002	U0001	19/05/2015	06:00:00	08:00:00	on
L00003	U0001	26/05/2015	20:00:00	00:00:00	off

Figura 1.14 Ventana programación de la iluminación.

Fuente: La Autora

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmLuces
    Public codiguito As String
    Public horaa, horad As TimeSpan

    Private Sub btnreporte_Click(sender As Object, e As EventArgs) Handles
btnreporte.Click
        FrmReporLuces.Show()
    End Sub

    Private Sub Prog_lucesBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs)
        Me.Validate()
        Me.Prog_lucesBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.BDDdomoDataSet)

    End Sub

    Private Sub FrmLuces_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        'TODO: esta línea de código carga datos en la tabla
        'BDDdomoDataSet.prog_luces' Puede moverla o quitarla según sea necesario.
```

```

Me.Prog_lucesTableAdapter.Fill(Me.BDDdomoDataSet.prog_luces)
codiguito = FrmAutenticacion.Lblcodigo.Text

'Carga a los botones el estado de las luces
If FrmEstandar.Luces = 1 Then
    btntodas.Text = "Apagar Todas"
End If
If FrmEstandar.Luces = 2 Then
    btntodas.Text = "Encender Todas"
End If

If FrmEstandar.L1 = 1 Then
    btnluz1.Text = "Luz 1 On"
End If
If FrmEstandar.L1 = 2 Then
    btnluz1.Text = "Luz 1 Off"
End If

If FrmEstandar.L2 = 1 Then
    btnluz2.Text = "Luz 2 On"
End If
If FrmEstandar.L2 = 2 Then
    btnluz2.Text = "Luz 2 Off"
End If

If FrmEstandar.L3 = 1 Then
    btnluz3.Text = "Luz 3 On"
End If
If FrmEstandar.L3 = 2 Then
    btnluz3.Text = "Luz 3 Off"
End If
End Sub

Private Sub btnguardar_Click(sender As Object, e As EventArgs) Handles
btnguardar.Click

    Try
        If codigoRegistradoLuces(TxtCodigo.Text) = True Then
            MsgBox("Horario iluminación ya esta registrado")

        Else
            horaa = TimeSpan.Parse(txthorab.Text)
            horad = TimeSpan.Parse(txthoracierre.Text)
            Me.Prog_lucesTableAdapter.Insert(TxtCodigo.Text, codiguito,
Dtfecha.Value, horaa, horad, txtluz1.Text, txtluz2.Text, txtluz3.Text)
            Me.Prog_lucesTableAdapter.Fill(Me.BDDdomoDataSet.prog_luces)
            TxtCodigo.Text = ""
            txthorab.Text = "00:00"
            txthoracierre.Text = "00:00"
            txtluz1.Text = ""
            txtluz2.Text = ""
            txtluz3.Text = ""

        End If
    Catch ex As Exception
        MsgBox(ex.ToString)
    End Try

End Sub

Private Sub btneliminar_Click(sender As Object, e As EventArgs) Handles
btneliminar.Click

```

```

If TxtCodigo.Text = "" Then
    MsgBox("Ingreso código válido")
Else

    Me.Prog_lucesTableAdapter.EliminarLuces(TxtCodigo.Text)
    Me.Prog_lucesTableAdapter.Fill(Me.BDDdomoDataSet.prog_luces)
    TxtCodigo.Text = ""
    txthorab.Text = "00:00"
    txthoracierre.Text = "00:00"
    txtluz1.Text = ""
    txtluz2.Text = ""
    txtluz3.Text = ""
End If
End Sub

Private Sub btneditar_Click(sender As Object, e As EventArgs) Handles
btneditar.Click
    If TxtCodigo.Text = "" Then
        MsgBox("Ingreso código válido")
    ElseIf txthorab.Text = "" Then
        MsgBox("Ingreso una hora válida")
    ElseIf txthoracierre.Text = "" Then
        MsgBox("Ingreso un hora válida")
    ElseIf txtluz1.Text = "" Then
        MsgBox("Ingreso un ON oara encendido y OFF para apagado")
    ElseIf txtluz2.Text = "" Then
        MsgBox("Ingreso un ON oara encendido y OFF para apagado")
    ElseIf txtluz3.Text = "" Then
        MsgBox("Ingreso un ON oara encendido y OFF para apagado")
    Else
        Me.Prog_lucesTableAdapter.EditarLuces(TxtCodigo.Text, codiguito,
Dtfecha.Value, txthorab.Text, txthoracierre.Text, txtluz1.Text, txtluz2.Text,
txtluz3.Text)
        Me.Prog_lucesTableAdapter.Fill(Me.BDDdomoDataSet.prog_luces)
        TxtCodigo.Text = ""
        txthorab.Text = "00:00"
        txthoracierre.Text = "00:00"
        txtluz1.Text = ""
        txtluz2.Text = ""
        txtluz3.Text = ""
    End If
End Sub

Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
    Close()
End Sub

Private Sub btntodas_Click(sender As Object, e As EventArgs) Handles
btntodas.Click
    FrmEstandar.Luces = FrmEstandar.Luces + 1
    If FrmEstandar.Luces = 1 Then
        btntodas.Text = "Apagar Todas"
        FrmEstandar.J = 0
        FrmEstandar.BandL = 1
    End If
    If FrmEstandar.Luces = 2 Then
        btntodas.Text = "Encender Todas"
        FrmEstandar.J = 0
        FrmEstandar.BandLA = 1
        FrmEstandar.Luces = 0
    End If
End Sub

```

```

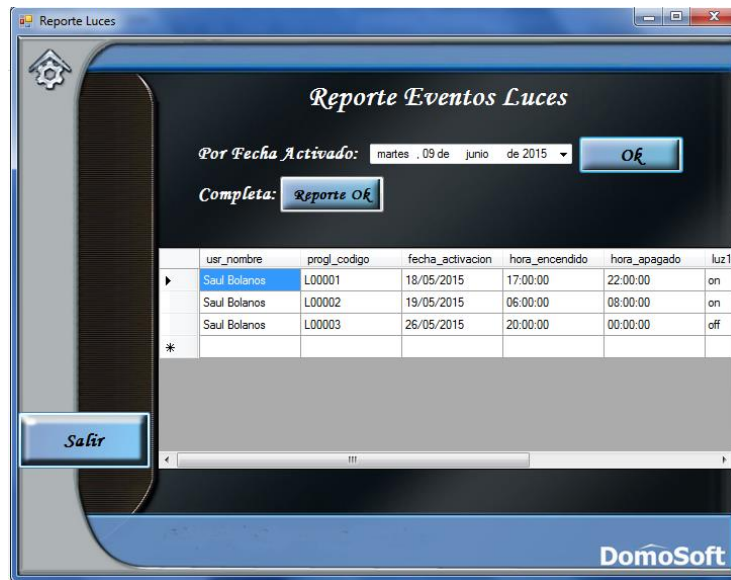
        End If
    End Sub

    Private Sub btnluz1_Click(sender As Object, e As EventArgs) Handles
btnluz1.Click
        FrmEstandar.L1 = FrmEstandar.L1 + 1
        If FrmEstandar.L1 = 1 Then
            btnluz1.Text = "Luz 1 On"
            FrmEstandar.BL1 = 1
        End If
        If FrmEstandar.L1 = 2 Then
            btnluz1.Text = "Luz 1 Off"
            FrmEstandar.BL1 = 4
            FrmEstandar.L1 = 0
        End If
    End Sub

    Private Sub btnluz2_Click(sender As Object, e As EventArgs) Handles
btnluz2.Click
        FrmEstandar.L2 = FrmEstandar.L2 + 1
        If FrmEstandar.L2 = 1 Then
            btnluz2.Text = "Luz 2 On"
            FrmEstandar.BL2 = 1
        End If
        If FrmEstandar.L2 = 2 Then
            btnluz2.Text = "Luz 2 Off"
            FrmEstandar.BL2 = 4
            FrmEstandar.L2 = 0
        End If
    End Sub

    Private Sub btnluz3_Click(sender As Object, e As EventArgs) Handles
btnluz3.Click
        FrmEstandar.L3 = FrmEstandar.L3 + 1
        If FrmEstandar.L3 = 1 Then
            btnluz3.Text = "Luz 3 On"
            FrmEstandar.BL3 = 1
        End If
        If FrmEstandar.L3 = 2 Then
            btnluz3.Text = "Luz 3 Off"
            FrmEstandar.BL3 = 4
            FrmEstandar.L3 = 0
        End If
    End Sub
End Class

```



**Figura 1.15** Ventana reporte de eventos de las luces.

**Fuente:** La Autora

Este formulario permite al administrador controlar los siguientes botones:

- **Encender todas:** Al presionarlo se encienden todas las luces, si se lo vuelve a presionar se apagan todas las luces.
- **Luz1:** Enciende o apaga desde el sistema la luz 1.
- **Luz 2:** Enciende o apaga desde el sistema la luz 2.
- **Luz 3:** Enciende o apaga desde el sistema la luz 3.
- **Guardar:** Para guardar un nuevo horario de control de la iluminación se debe:
  - A) Asignarle un código, este debe iniciar con la letra “L” para que se identifique que el registro pertenece a un horario de la iluminación.
  - B) Escoger la fecha en la cual se va a activar el dispositivo.
  - C) Seleccionar la hora para encender la luz, su formato debe ser 00:00 donde los dos primeros son la hora representada en veinticuatro horas y los dos siguientes son los minutos. Ejemplo, si desea que se active a las dos de la tarde deberá escribir 14:00, si desea que se active a las nueve de la mañana se debe digitar 09:00.
  - D) Seleccionar la hora para la apagar las luces.
  - E) Ingresa la acción de la luz 1 “on” para encendido, “off” para apagado.
  - F) Ingresa la acción de la luz 2 “on” para encendido, “off” para apagado

- G) Ingresa la acción de la luz 3 “on” para encendido, “off” para apagado
- **Editar:** Cambia un registro de horario de iluminación al ingresar sus datos en los campos especificados.
  - **Eliminar:** Ingresando un código de horario existente se elimina su registro de la base de datos.
  - **Salir:** Regresa al formulario estándar para escoger otra opción si es necesario.
  - **Reporte:** Muestra en otra ventana los horarios que se han registrado en la BDD.
    - A) Seleccionar la fecha para detallar los horarios registrados en una fecha específica.
    - B) Clic en el botón Fecha Ok.
    - C) Otra opción es dar clic en el botón reporte Ok para visualizar todos los registros.

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmReporLuces

    Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles
btnsalir.Click
        Close()
    End Sub

    Private Sub FrmReporLuces_Load(sender As Object, e As EventArgs) Handles
 MyBase.Load
        'TODO: esta línea de código carga datos en la tabla
'BDDdomoDataSet.usuario_luces' Puede moverla o quitarla según sea necesario.
        Me.Usuario_lucesTableAdapter.Fill(Me.BDDdomoDataSet.usuario_luces)

    End Sub

    Private Sub btnfecha_Click(sender As Object, e As EventArgs) Handles
btnfecha.Click
        Me.Usuario_lucesTableAdapter.FillBy(Me.BDDdomoDataSet.usuario_luces,
dtpfechaact.Value)
    End Sub

    Private Sub btncompleta_Click(sender As Object, e As EventArgs) Handles
btncompleta.Click
        Me.Usuario_lucesTableAdapter.Fill(Me.BDDdomoDataSet.usuario_luces)
    End Sub
End Class
```

## 1.9 Reportes del Sistema



**Figura 5.14** Ventana para visualización de reportes por el usuario general.

**Fuente:** La Autora

Esta ventana solamente se muestra al usuario general (común), donde puede escoger cualquier opción para visualizar el respectivo reporte.

El código fuente correspondiente a este formulario detallo a continuación:

```
Public Class FrmInvitado
```

```
    Private Sub FrmInvitado_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    End Sub
```

```
    Private Sub btnsalir_Click(sender As Object, e As EventArgs) Handles btnsalir.Click
```

```
        End  
    End Sub
```

```
    Private Sub btneliminar_Click(sender As Object, e As EventArgs) Handles btncortina.Click
```

```
        FrmReporCortina.Show()  
    End Sub
```

```
    Private Sub Button5_Click(sender As Object, e As EventArgs) Handles btnpuerta.Click
```

```
        FrmReporteIngreso.Show()  
    End Sub
```

```
Private Sub btniluminacion_Click(sender As Object, e As EventArgs) Handles  
btniluminacion.Click  
    FrmReporLuces.Show()  
End Sub  
  
Private Sub btナルarma_Click(sender As Object, e As EventArgs) Handles  
btナルarma.Click  
    FrmAlarma.Show()  
End Sub  
  
Private Sub btnseguridad_Click(sender As Object, e As EventArgs) Handles  
btnseguridad.Click  
    Frmseguridad.Show()  
End Sub  
End Class
```



# Código de los Micro controladores PIC

## Código PIC Principal

```
INCLUDE "modedefs.BAS"

define OSC 4

cmcon=7

I var byte

SC var byte

ST var byte

SP var byte

SerTecla var byte

SerPLC var byte

SerOrden var byte

STV var byte

BANDO VAR BYTE

BE VAR BYTE

BZ1 VAR BYTE

BZ2 VAR BYTE

BZ3 VAR BYTE

BE=0

BZ1=0

BZ2=0

BZ3=0

Inicio:

BANDO=0

Serin PORTB.0,N9600,serorden
```

```

if serorden = "S" then
    goto chequeo
endif
pause 27
if PORTB.6=1 then
    Serout PORTB.1,N9600,["+"]
    pause 27
    serout PORTB.3,N9600,["+"]

    serin PORTB.2,N9600,sertecla
    while (sertecla!="#")
        serout PORTB.1,N9600,[sertecla]
        serin PORTB.2,N9600,sertecla
        pause 27
    wend
    BANDO=1
    serout PORTB.1,N9600,["#"]
endif

```

```

if serorden = "O" then
    serout PORTB.3,N9600,["O"]
    PAUSE 27
    serout PORTB.4,N9600,[serorden]
    'serin PORTB.5,N9600,serorden
    'serout PORTB.1,N9600,[serorden]
    pause 1000
    serout PORTB.4,N9600,["A"]

```

```

    BANDO=1
endif
IF SERORDEN = "E" THEN
    serout PORTB.3,N9600,["E"]
    BANDO=1
ENDIF
if PORTA.2=0 and be=2 THEN
    LOW PORTA.3
    serout PORTB.1,N9600,["b"]
    BZ1=0
    BZ2=0
    BZ3=0
    BE=0
ENDIF
IF SERORDEN = "a" and PORTA.2=0 THEN
    HIGH PORTA.3
    BE=1
    bando=1
ENDIF
if serorden = "d" then
    low PORTA.3
    BE=2
    PAUSE 1000
endif
IF PORTA.2=1 AND BE=0 THEN
    SEROUT PORTB.1,N9600,["c"]
    BE=2

```

```

HIGH PORTA.3
be=2
ENDIF
IF PORTA.2=1 AND BE = 1 THEN
    SEROUT PORTB.1,N9600,["c"]
    BE=2
ENDIF
if PORTB.7=1 AND PORTA.2=1 AND BZ1=0 THEN
    SEROUT PORTB.1,N9600,["1"]
    PAUSE 25
    BZ1=1
ENDIF
if PORTA.0=1 AND PORTA.2=1 AND BZ2=0 THEN
    SEROUT PORTB.1,N9600,["2"]
    PAUSE 25
    BZ2=1
ENDIF
if PORTA.1=1 AND PORTA.2=1 AND BZ3=0 THEN
    SEROUT PORTB.1,N9600,["3"]
    PAUSE 25
    BZ3=1
ENDIF

IF BANDO=0 THEN
    serout PORTB.4,N9600,[serorden]
ENDIF

```

goto inicio

Chequeo:

!\*\*\*\*\* Chequeo Comunicacion con el Computador

sc=0

PAUSE 25

for i=0 to 4

serout PORTB.1,N9600,["?"]

serin PORTB.0,N9600,serorden

if serorden="?" then

sc=sc+1

endif

next

PAUSE 25

if sc>2 then

serout PORTB.1,N9600,["X"]

else

serout PORTB.1,N9600,["x"]

endif

PAUSE 25

!\*\*\*\*\* Chequeo Comunicacion con el Pic del teclado de Ingreso

serout PORTB.3,N9600,["?"]

pause 27

serout PORTB.1,N9600,["Y"]

PAUSE 25

!\*\*\*\*\* Chequeo Comunicacion con el PLC

```

sp=0
i=0
repeat
    serout PORTB.4,N9600,["?"]
    SERIN PORTB.5,N9600,SERPLC
    PAUSE 45
    i=i+1
    IF SERPLC="?" THEN
        SP=SP+1
    ENDIF
until i=3
PAUSE 25
IF SP>2 THEN
    SEROUT PORTB.1,N9600,["Z"]
ELSE
    SEROUT PORTB.1,N9600,["z"]
ENDIF
goto inicio
end

```

### **Código PIC Alarma**

```

X VAR BYTE :TIEMPOSAL VAR BYTE :ENTRADA VAR BYTE :TIEMPOSIR
VAR BYTE

```

```

ONOFF VAR PORTB.0 :LED VAR PORTA.0 : BIP VAR PORTB.5

```

RELE VAR PORTB.7 :SIRENA VAR PORTB.1 :TONOS VAR PORTB.6  
ZONA3 VAR PORTB.4 :ZONA2 VAR PORTB.3 :ZONA1T VAR PORTB.2  
LEDZ1 VAR PORTA.1 : LEDZ2 VAR PORTA.2 : LEDZ3 VAR PORTA.3  
:CMCON = 7

. \*\*\*\*\*  
;

@ device BOD\_OFF ;APAGA EL BODEN DE CONSUMO

@ device XT\_OSC ; Oscilador externo de 4 MHZ

TIEMPOSAL= 5 ;\*\*\*\*+ 5 DA EL TIEMPO DE SALIDA X 1 SEG. MAX 255 #

ENTRADA = 8 ;\*\*\*\*\*DA EL TIEMPO DE ENTRADA X 1 SEG. MAX 255 #

TIEMPOSIR= 250 ;\*\*\*\*\*DA EL TIEMPO DE SIRENA X 1 SEG. MAX 255  
#

GOTO TEST

TELF1: DTMFOUT TONOS , [0,9,6,1,3,6,5,6,4] ;\*\*\*COLOCAR 1# TELEFONO

RETURN

TELF2: DTMFOUT TONOS , [0,9,6,1,3,6,5,6,4] ;\*\*\*COLOCAR 2# TELEFONO

RETURN

PREARMADO: ;HIGH SIRENA ;AVISO DE PREARMADO SIRENA OMITIR  
ESTE si no desea

PAUSE 150

LOW SIRENA

PAUSE 150

RETURN

SENSIBILIDAD: PAUSE 100 ;\*\*\*SENSIBILIDAD DE LAS ZONAS DE 1 A 200  
mlSeg.

RETURN

. \*\*\*\*\*  
;

TEST:

```

GOSUB LUCESHIGH
    GOSUB PAUSA2000
    GOSUB PAUSA2000
GOSUB LUCESLOW
    FOR X = 1 TO 8
GOSUB LUCESHIGH
    GOSUB SENSIBILIDAD
GOSUB LUCESLOW
    GOSUB SENSIBILIDAD
    NEXT

GOTO PREND

LUCESHIGH: HIGH LED :HIGH BIP :HIGH LEDZ1 :HIGH LEDZ2 :HIGH LEDZ3
    RETURN

LUCESLOW: LOW LED :LOW BIP :LOW LEDZ1 :LOW LEDZ2 :LOW LEDZ3
    RETURN

PAUSA2000: PAUSE 2000
    RETURN

PAUSA1000: PAUSE 1000
    RETURN

PAUSA500: PAUSE 500
    RETURN

ONOFF1M:
    IF ONOFF = 1 THEN APAGADO
    PAUSE 500
    IF ONOFF = 1 THEN APAGADO

```



```

        PAUSE 500
    RETURN
ONOFF11:
    IF ONOFF = 1 THEN APAGADO
    RETURN

; ***** LLAMADA TELEFONICA *****
LLAMADA:
    HIGH RELE
    GOSUB ONOFF1M
    GOSUB ONOFF1M
    GOSUB TELF1
    FOR X = 1 TO 15
    GOSUB ONOFF1M
    NEXT
    FOR X = 1 TO 3
    GOSUB TELF1
    GOSUB ONOFF1M
    GOSUB ONOFF1M
    GOSUB ONOFF1M
    NEXT
    LOW RELE
    GOSUB ONOFF1M
    GOSUB ONOFF1M

    HIGH RELE
    GOSUB ONOFF1M

```

```
GOSUB ONOFF1M
    GOSUB TELF2
    FOR X = 1 TO 15
GOSUB ONOFF1M
    NEXT
    FOR X = 1 TO 3
    GOSUB TELF2
GOSUB ONOFF1M
GOSUB ONOFF1M
GOSUB ONOFF1M
    NEXT
LOW RELE
    RETURN
```

```
ENCENDERZ1: HIGH LEDZ1
```

```
    RETURN
```

```
ENCENDERZ2: HIGH LEDZ2
```

```
    RETURN
```

```
ENCENDERZ3: HIGH LEDZ3
```

```
    RETURN
```

```
GOTO PREND
```

```
.***** APAGADO *****
;
```

```
APAGADO:
```

```
    LOW SIRENA :LOW BIP :LOW LED :LOW RELE :LOW TONOS
```

```
    LOW LEDZ1 :LOW LEDZ2 :LOW LEDZ3
```

```

FOR X = 1 TO 3
HIGH LED : HIGH BIP
PAUSE 50
LOW LED :LOW BIP
PAUSE 50
NEXT
GOSUB PREARMADO
GOSUB PREARMADO
; ***** PRENDIDO *****

```

```

PREND:
IF ZONA3 = 1 THEN GOSUB ENCENDERZ3
IF ZONA2 = 1 THEN GOSUB ENCENDERZ2
IF ZONA1T =1 THEN GOSUB ENCENDERZ1
PAUSE 100
LOW LEDZ1 : LOW LEDZ2 :LOW LEDZ3
IF ONOFF = 1 THEN PREND

```

```

RETARDO:
GOSUB BIPBIP
GOSUB BIPBIP
GOSUB BIPBIP
GOSUB BIPBIP
FOR X = 1 TO TIEMPOSAL
HIGH LED
PAUSE 250
LOW LED : HIGH BIP

```

```
PAUSE 250
GOSUB ONOFF11
HIGH LED : LOW BIP
PAUSE 250
LOW LED
PAUSE 250
GOSUB ONOFF11
NEXT
GOTO RETARDO2
```

BIPBIP:

```
HIGH LED : HIGH BIP
PAUSE 60
LOW LED : LOW BIP
PAUSE 60
RETURN
```

INTERMIT:

```
HIGH LED
PAUSE 60
LOW LED
PAUSE 60
RETURN
```

RETARDO2:

```
FOR X = 1 TO 5
```

```
GOSUB BIPBIP
GOSUB BIPBIP
GOSUB BIPBIP
GOSUB BIPBIP
GOSUB ONOFF11
GOSUB INTERMIT
GOSUB INTERMIT
GOSUB INTERMIT
GOSUB INTERMIT
GOSUB ONOFF11
    NEXT
HIGH LED
```

```
GOSUB PREARMADO
```

```
; ***** ARMADO *****
```

```
ARMADO:
```

```
GOSUB ONOFF11
IF ZONA1T= 1 THEN ESPERACONF
IF ZONA2 = 1 THEN SIRENAS2CONF
IF ZONA3 = 1 THEN SIRENAS3CONF
PAUSE 10
```

```
GOTO ARMADO
```

```
ESPERACONF: GOSUB SENSIBILIDAD
```

```
IF ZONA1T = 1 THEN ESPERA
GOTO ARMADO
```

SIRENAS2CONF: GOSUB SENSIBILIDAD

IF ZONA2 = 1 THEN SIRENA2

GOTO ARMADO

SIRENAS3CONF: GOSUB SENSIBILIDAD

IF ZONA3 = 1 THEN SIRENA3

GOTO ARMADO

SIRENA2: HIGH LEDZ2

GOTO SIRENAS

SIRENA3: HIGH LEDZ3

GOTO SIRENAS

ESPERA:

FOR X = 1 TO ENTRADA

HIGH LED :HIGH BIP

GOSUB PAUSA500

LOW LED :LOW BIP

GOSUB PAUSA500

GOSUB ONOFF11

NEXT

HIGH LEDZ1

SIRENAS:

FOR X = 1 TO 10

HIGH SIRENA :HIGH BIP :HIGH LED

GOSUB ONOFF1M

```

NEXT
GOSUB LLAMADA
FOR X = 1 TO TIEMPOSIR
GOSUB ONOFF1M
NEXT
LOW SIRENA : LOW BIP

FOR X = 1 TO 250 ; DA EL TIEMPO ENTRE DISPARO Y DISPARO
SIGUIENTE
GOSUB ONOFF1M
NEXT

GOTO ARMADO

END

```

### **Código PIC del Teclado de la Alarma**

```

INCLUDE "modedefs.bas"
define OSC 4
cmcon=7
R VAR BYTE
SERPC VAR BYTE
SERIAL VAR BYTE
BIP VAR PORTA.0
LED VAR PORTA.1

```

BAND var byte

A VAR PORTB.2

B VAR PORTB.3

C VAR PORTB.4

D VAR PORTB.5

UNO VAR PORTB.6

DOS VAR PORTB.7

TRES VAR PORTA.3

HIGH LED : HIGH BIP

PAUSE 500

LOW LED

LOW BIP

band=0

GOTO TECLAUNO

BARRIDO:

LOW A

IF UNO = 0 THEN SERIAL ="1" :RETURN

IF DOS = 0 THEN SERIAL ="2" :RETURN

IF TRES = 0 THEN SERIAL ="3" :RETURN

HIGH A

LOW B

IF UNO = 0 THEN SERIAL ="4" :RETURN

IF DOS = 0 THEN SERIAL ="5" :RETURN

IF TRES = 0 THEN SERIAL ="6" :RETURN



```

HIGH B
LOW C
IF UNO = 0 THEN SERIAL ="7" :RETURN
IF DOS = 0 THEN SERIAL ="8" :RETURN
IF TRES = 0 THEN SERIAL ="9" :RETURN
HIGH C
LOW D
IF UNO = 0 THEN SERIAL ="*" :RETURN ;tecla *
IF DOS = 0 THEN SERIAL ="0" :RETURN
IF TRES = 0 THEN SERIAL ="#" :RETURN ;tecla #
HIGH D
pause 10

IF PORTA.4 = 1 AND PORTA.2=0 AND BAND=0 THEN
LOW PORTB.0
high PORTA.2
band=1
ENDIF
IF PORTA.4 = 0 AND PORTA.2=1 AND BAND=1 THEN
'high PORTB.0
'low PORTA.2
'pause 2000
'band=3
goto desactivar
ENDIF
IF PORTA.4=0 AND BAND >0 THEN
BAND=0

```

```
ENDIF  
GOTO BARRIDO
```

```
PTECLA:
```

```
    HIGH LED : HIGH BIP  
    PAUSE 100  
    LOW LED : LOW BIP
```

```
ESPACIO:
```

```
    IF UNO  = 0 THEN ESPACIO  
    IF DOS  = 0 THEN ESPACIO  
    IF TRES = 0 THEN ESPACIO  
    PAUSE 25  
    RETURN
```

```
TECLAUNO:
```

```
    GOSUB BARRIDO  
    GOSUB PTECLA  
    IF SERIAL="*" AND PORTA.4=0 THEN  
        LOW PORTB.0  
        high PORTA.2  
        BAND=1  
        pause 2000  
        GOTO TECLAUNO  
    ENDIF
```

```
    IF SERIAL="1" THEN TECLADOS  
        GOTO FALSO
```

TECLADOS:

GOSUB BARRIDO

GOSUB PTECLA

IF SERIAL="2" THEN TECLATRES

GOTO FALSO1

TECLATRES:

GOSUB BARRIDO

GOSUB PTECLA

IF SERIAL="3" THEN TECLACUATRO

GOTO FALSO2

TECLACUATRO:

GOSUB BARRIDO

GOSUB PTECLA

IF SERIAL="4" THEN DESACTIVAR

GOTO ERROR

goto TECLAuno

FALSO:

GOSUB BARRIDO

GOSUB PTECLA

FALSO1:

GOSUB BARRIDO

GOSUB PTECLA

FALSO2:

GOSUB BARRIDO

GOSUB PTECLA

ERROR:

FOR R = 1 TO 3

PAUSE 100

HIGH LED : HIGH BIP

PAUSE 100

LOW LED : LOW BIP

NEXT

GOTO TECLAUNO

DESACTIVAR:

FOR R = 1 TO 2

PAUSE 100

HIGH LED : HIGH BIP

PAUSE 100

LOW LED : LOW BIP

NEXT

HIGH PORTB.0

LOW PORTA.2

band=1

PAUSE 2000

'HIGH DOOR

'PAUSE 1000

```
'LOW DOOR
GOTO TECLAuno
END
```

### **Código PIC del Teclado de Ingreso de la Puerta**

```
include "modedefs.bas"
define OSC 4
CMCON=7
R var byte
CONT var byte
SERPC var byte
SERIAL var byte
BIP var PORTA.0
LED var PORTA.1
DOOR VAR PORTA.2
CLAV var byte[20]

A VAR PORTB.2
B VAR PORTB.3
C VAR PORTB.4
D VAR PORTB.5

UNO VAR PORTB.6
DOS VAR PORTB.7
TRES VAR PORTA.3
```

Iniciando:

HIGH LED : HIGH BIP

PAUSE 500

LOW LED

LOW BIP

CONT=0

r=0

GOTO TECLAS

BARRIDO:

LOW A

IF UNO = 0 THEN SERIAL ="1" :RETURN

IF DOS = 0 THEN SERIAL ="2" :RETURN

IF TRES = 0 THEN SERIAL ="3" :RETURN

HIGH A

LOW B

IF UNO = 0 THEN SERIAL ="4" :RETURN

IF DOS = 0 THEN SERIAL ="5" :RETURN

IF TRES = 0 THEN SERIAL ="6" :RETURN

HIGH B

LOW C

IF UNO = 0 THEN SERIAL ="7" :RETURN

IF DOS = 0 THEN SERIAL ="8" :RETURN

IF TRES = 0 THEN SERIAL ="9" :RETURN

HIGH C

LOW D

IF UNO = 0 THEN SERIAL = "\*" :RETURN

IF DOS = 0 THEN SERIAL = "0" :RETURN

IF TRES = 0 THEN SERIAL = "#" :RETURN

HIGH D

pause 10

GOTO BARRIDO

PTECLA:

HIGH LED : HIGH BIP

PAUSE 100

LOW LED : LOW BIP

ESPACIO:

IF UNO = 0 THEN ESPACIO

IF DOS = 0 THEN ESPACIO

IF TRES = 0 THEN ESPACIO

PAUSE 25

RETURN

TECLAS:

GOSUB BARRIDO

GOSUB PTECLA

CLAV[CONT]=SERIAL

CONT=CONT+1

IF SERIAL="#" THEN

HIGH DOOR

```
SERIN PORTB.0,N9600,SERPC
PAUSE 25
IF SERPC="+" THEN
    LOW DOOR
    PAUSE 25
    FOR R=0 TO CONT-1
        SEROUT PORTB.1,N9600,[CLAV[R]]
        PAUSE 30
    NEXT
    CONT=0
ENDIF
SERIN PORTB.0,N9600,SERPC
IF SERPC = "O" THEN
    GOTO OPEN
ELSE
    GOTO FALSO
ENDIF
ENDIF
```

GOTO TECLAS

OPEN:

```
FOR R = 1 TO 2
    PAUSE 100
    HIGH LED : HIGH BIP
    PAUSE 100
    LOW LED : LOW BIP
```



```

NEXT
GOTO TECLAS

FALSO:
    FOR R = 1 TO 3
        PAUSE 100
        HIGH LED : HIGH BIP
        PAUSE 100
        LOW LED : LOW BIP
    NEXT
GOTO TECLAS
END

```

### **Código PIC del PLC**

```

INCLUDE "modedefs.bas" 'Contiene velocidadesde modulacion para SERIN
SEROUT

DEFINE OSC 4

CMCON=7      'Convierte el puerto A en digital (Sirve para comunicacion serial)

ORDENES VAR BYTE

```

Inicio:

```

SERIN PORTA.2,N9600,ORDENES

PAUSE 25

IF ORDENES="?" THEN

    SEROUT PORTA.3,N9600,[ORDENES]

```

```
ENDIF
IF ORDENES="O" THEN
    HIGH PORTB.0
ENDIF
IF ORDENES="A" THEN
    LOW PORTB.0
ENDIF
IF ORDENES="B" THEN
    HIGH PORTB.1
ENDIF
IF ORDENES="C" THEN
    LOW PORTB.1
ENDIF
IF ORDENES="D" THEN
    HIGH PORTB.2
ENDIF
IF ORDENES="F" THEN
    LOW PORTB.2
ENDIF
IF ORDENES="G" THEN
    HIGH PORTB.3
ENDIF
IF ORDENES="H" THEN
    LOW PORTB.3
ENDIF
IF ORDENES="I" THEN
    HIGH PORTB.4
```

```
ENDIF
IF ORDENES="J" THEN
    LOW PORTB.4
ENDIF
IF ORDENES="K" THEN
    HIGH PORTB.5
ENDIF
IF ORDENES="L" THEN
    LOW PORTB.5
ENDIF
IF ORDENES="M" THEN
    HIGH PORTB.6
ENDIF
IF ORDENES="N" THEN
    LOW PORTB.6
ENDIF
IF ORDENES="P" THEN
    HIGH PORTB.7
ENDIF
IF ORDENES="Q" THEN
    LOW PORTB.7
ENDIF
GOTO INICIO
END
```