

# Project Error

---

*Source:* <http://www.tech-archive.net/Archive/VB/microsoft.public.vb.bugs/2006-09/msg00024.html>

---

- *From:* "Ketu007" <ketu007\_in@xxxxxxxxxx>
  - *Date:* Wed, 20 Sep 2006 06:42:17 -0400
- 

"NOTE : Dt.10/05/2004  
"initially tmrReadFromSIM.interval=15000 now it's 1500  
Option Explicit

Private Declare Sub Sleep Lib "Kernel32" (ByVal dwMilliseconds As Long)

Const MSG\_LEN = 130

"Following variables are usefull to get connection data from  
zconfiguration.ini  
Dim strDataSrc As String  
Dim strUser As String  
Dim strPwd As String  
"Upto here

Dim strSMSC\_NUMBER As String  
\*Dim WithEvents SMSNotify As SMS3ASuiteLib.SMS\_SuiteAdapter  
\*Dim SMSSend As SMS3ASuiteLib.SMS\_SuiteAdapter  
\*Dim ShortMSG As SMS3ASuiteLib.ShortMessage  
\*Dim SMSMemory\_SIM As SMS3ASuiteLib.ISMSMemory

'Dim con As New ADODB.Connection  
'Dim rst As Recordset

Dim gstrCallerNo As String  
Public isClientConnected As Boolean 'by Ketan

"Related to user query from database (oracle 8i)  
Private Type SMSInformation  
PF\_ID As String  
PFBalance As String  
ChequeNo As String  
ChequeAmount As String  
ChequeDate As String  
CO7Number As String  
CO6Number As String  
BillRegistrationNumber As String  
PPONumber As String

## Project Error

```
PPODate As String
PensionAmount As String
EnhancedFamilyPension As String
NormalFamilyPension As String
CommutatedPensionAmount As String
NetPayablePensionAmount As String
CommutationAmount As String
DCRGAmount As String
PFSettlementAmount As String
DLIAmount As String
GISAmount As String
LeaveEncashmentAmount As String
LAP As String
LHAP As String
HalfSetPass As String
BillAmount As String
"June 07, 2004 – For some bank of Rajkot – Raj Info
Balance As String
"June 14, 2004 – To support SRCH option in Z.GSM GOLD 8.1
SRCHResult As String
End Type
Dim SMSInfo As SMSInformation
```

""Following structure is useful to display SMS details

```
"Private Type SMSDataStructure
" SMSDS_CallerID As String
" SMSDS_Type As String
" SMSDS_Request As String
" SMSDS_RequestDT As String
" SMSDS_Response As String
" SMSDS_ResponseDT As String
" SMSDS_Status As String
"End Type
"Dim SMSData As SMSDataStructure
""Upto here
```

```
Private Enum FillType
FT_ADD = 0
FT_UPDATE = 1
End Enum
```

```
Dim blnExplicitUnload As Boolean
```

```
Private Type SMSRequestType
SRT_PFI As Integer
SRT_PAI As Integer
SRT_PNI As Integer
SRT_PLI As Integer
SRT_BLI As Integer
SRT_UNKNOWN As Integer
End Type
```

Project Error

## Project Error

Dim SRT As SMSRequestType

""Following enum is usefull in fncRequestToCOM

"Private Enum ENUM\_RequestType

" enmInitialization = 0

" enmReadFromSIM = 1

" enmSendToSIM = 2

" enmDeleteFromSIM = 3

"End Enum

'Dim bytLastCmd As Byte

'For msCOMMSIM control and msCOMMSIM\_OnComm event

Dim DataRVFPort As Boolean

Dim mErrorMsg As String

'Dim mstrInputOne As String

Dim mbolDCStatus As Boolean

"

'Dim blnGoForRead As Boolean 'This variable will true if initalize is ok

'Dim blnGoForRespGen As Boolean 'This variable will true if read is ok

'Dim blnGoForSend As Boolean 'This variable will true if Response Generation is ok

'Dim blnGoForDelete As Boolean 'This variable will true if send is ok

'Dim blnGoForNextIteration As Boolean 'This variable will true if delete is ok

Dim blnSkipDeletion As Boolean "This variable will true if two smss are to be send for one request

Dim strResp1\$, strResp2\$ "User more variables like strResp3,strResp4 if u required to send more SMSs for one request

"

Dim mbln\_IsInitializeComplete As Boolean

Dim tmpTimer

Dim intIndex As Integer 'This variable will keep values of lines in lvwSMS

Dim LstItemCrntStatus As ListItem

Dim strFocusedLvw As String

Dim mblnDenyShowOnScreen As Boolean

Dim lstRequestID As Long

Dim blnInit As Boolean

## Project Error

Dim nClient 'Added by Ketan for Winsock Communication on Dt : 04/02/2005  
for Server Use Only

Dim m\_iBackColor As Long 'Added by Ketan on Dt : 13/07/2005 for Popup  
Window Color

```
'Added by Ketan On Dt : 12/02/2005
Private Sub cmdCloseOnlineUser_Click()
fraOnlineUser.Visible = False
cmdShowOnlineUsers.Enabled = True
If gblnIsServer = False Then
lvwOnLineUsers.ListItems.Clear
End If
End Sub
```

```
'Added by Ketan On dt : 12/02/3005
Private Sub cmdCloseSendMessage_Click()
fraMessage.Visible = False
txtMsg.Text = ""
'cmdSendMsg.Enabled = True
'cmdCloseOnlineUser.Enabled = True
fraOnlineUser.Enabled = True
ChkBroadCast.Value = 0
fraMessage.Tag = ""
ChkBroadCast.Enabled = True
cmdSendToUser.Caption = "&Send"
End Sub
```

```
'Developer : Ketan Dt : 12/02/2005
'Purpose : Send Message to Internal Selected User
Private Sub cmdSendMsg_Click()
Dim i As Double
Dim nSelected As Double
```

```
If lvwOnLineUsers.ListItems.Count > 0 Then
```

```
nSelected = 0
```

```
For i = 1 To lvwOnLineUsers.ListItems.Count
If lvwOnLineUsers.ListItems(i).Checked = True Then
nSelected = nSelected + 1
End If
Next
```

```
If nSelected > 0 Then
fraOnlineUser.Enabled = False
fraMessage.Visible = True
fraMessage.ZOrder 0
'cmdSendMsg.Enabled = False
'cmdCloseOnlineUser.Enabled = False
Else
```

Project Error

## Project Error

```
MsgBox "Please select user(s) for sending message",  
vbInformation, "Send Message"  
Exit Sub  
End If  
  
End If  
  
End Sub  
  
'by Ketan On Dt : 12/02/2005  
Private Sub cmdSendToUser_Click()  
Dim i As Double  
Dim strIPAddr As String  
  
If gblnIsServer = False Then  
If isClientConnected = False Then  
MsgBox "Z.GSMS Server is not Running, Please try  
later...", vbCritical, "Z.GSMS Server"  
Exit Sub  
End If  
End If  
  
If Len(Trim(txtMsg.Text)) = 0 Then  
MsgBox "Please enter message...", vbCritical, "Send Message"  
txtMsg.SetFocus  
Exit Sub  
End If  
  
strIPAddr = ""  
  
If gblnIsServer = False Then  
  
If Len(Trim(fraMessage.Tag)) > 0 Then  
strIPAddr = fraMessage.Tag  
End If  
  
For i = 1 To lvwOnLineUsers.ListItems.Count  
If lvwOnLineUsers.ListItems(i).Checked = True Then  
If Len(Trim(strIPAddr)) = 0 Then  
strIPAddr = lvwOnLineUsers.ListItems(i).Key  
Else  
strIPAddr = strIPAddr & "|" &  
lvwOnLineUsers.ListItems(i).Key  
End If  
End If  
Next  
  
If Len(Trim(strIPAddr)) > 0 Then  
If ChkBroadCast.Value = 0 Then  
'Send Message to Selected User  
tcpClient.SendData "SEND_MSG" & Chr(149) &
```

## Project Error

```
udfCompany.gstrCurrentUser & Chr(149) & txtMsg.Text & Chr(149) &
strIPAddr
Else
'Send Message to All Connected Users
tcpClient.SendData "BROADCAST" & Chr(149) &
udfCompany.gstrCurrentUser & Chr(149) & txtMsg.Text
End If
End If

Else

For i = 1 To lvwOnLineUsers.ListItems.Count
If ChkBroadCast.Value = 0 Then
If lvwOnLineUsers.ListItems(i).Checked = True Then
Send_Message_To_Client txtMsg.Text,
lvwOnLineUsers.ListItems(i).Key, udfCompany.gstrCurrentUser,
ZDLL.GetIPAddress
End If
Else
Send_Message_To_Client txtMsg.Text,
lvwOnLineUsers.ListItems(i).Key, udfCompany.gstrCurrentUser,
ZDLL.GetIPAddress
End If
Next

If Len(Trim(fraMessage.Tag)) > 0 Then
Send_Message_To_Client txtMsg.Text, fraMessage.Tag,
udfCompany.gstrCurrentUser, ZDLL.GetIPAddress
End If

End If

cmdCloseSendMessage_Click
End Sub

'by KEtan On dt : 12/02/2005
Private Sub cmdShowOnlineUsers_Click()
Dim strUsers As String

If gblnIsServer = True Then
fraOnlineUser.Visible = True
fraOnlineUser.ZOrder 0
Else
If tcpClient.State = sckConnected Then
cmdShowOnlineUsers.Enabled = False
tcpClient.SendData "ONLINE_USER" & Chr(149) 'Send Online User
Info Request to Server
Else
MsgBox "Server is not running...., Please try later...",
vbInformation, "Online Users"
'cmdShowOnlineUsers.Enabled = True
```

## Project Error

```
'fraOnlineUser.Visible = True  
End If  
End If
```

```
End Sub
```

```
"Dim TT As CTooltip
```

```
Private Sub Form_Load()  
On Error GoTo ErrH
```

```
Screen.MousePointer = vbHourglass
```

```
Dim LstItem As ListItem  
Dim SrvIP As String
```

```
'Me.Caption = gstrApplicationName  
Me.Caption = ""  
Me.Top = 0: Me.Left = 0
```

```
"Set limitations of Client – Server  
If gblnIsServer = True Then 'Server
```

```
'Added by Ketan for Winsock Communication on Dt : 04/02/2005  
'Open Winsock Port for Server
```

```
'-----  
nClient = 0 'Number of winsock  
tcpServer(nClient).LocalPort = strLocalPort  
tcpServer(nClient).Listen 'start listening (for clients)  
'-----
```

```
Add_OnLineUser udfCompany.gstrCurrentUser,  
udfCompany.gstrUserDesig, ZDLL.GetIPAddress
```

```
MnuStartCOMPort.Visible = True  
mnuF_Authority.Visible = True  
MnuCOMPort.Visible = True  
'MnuDelReadSMS.Visible = True  
MnuSettings_DBConfiguration.Visible = True
```

```
"Set Authority rights  
If UCase(AuthorityDesig) = UCase("Operator") Then  
MnuStartCOMPort.Enabled = False  
MnuStartSMSing.Enabled = False  
MnuSendSMS.Enabled = False
```

```
mnuF_Authority.Enabled = False
```

```
MnuCOMPort.Enabled = False  
MnuParameter.Enabled = False  
MnuSettings_DBConfiguration.Enabled = False
```

Project Error

## Project Error

```
MnuConnectedClients.Enabled = False
Else
MnuStartCOMPort.Enabled = True
MnuStartSMSing.Enabled = True
MnuSendSMS.Enabled = True

mnuF_Authority.Enabled = True

MnuCOMPort.Enabled = True
MnuParameter.Enabled = True
MnuSettings_DBConfiguration.Enabled = True
MnuConnectedClients.Enabled = True
End If

Else 'Client

If Len(Trim(strRemoteHost)) = 255 Or Len(Trim(strRemoteHost)) = 0
Then
SrvIP = InputBox("Enter Z.GSMS Server IP Address or Computer
Name", "Server IP")
If SrvIP = "" Then
MsgBox "Must be enter Z.GSMS Server IP Address"
End
Else
ZDLL.SaveToIni ZConfigurationINI, "COMSettings",
"RemoteHost", SrvIP
End If
End If

If Len(Trim(strRemotePort)) = 255 Or Len(Trim(strRemotePort)) = 0
Then
strRemotePort = 1001
End If

'Added by Ketan for Winsock Communication on Dt : 04/02/2005
'Connect with Winsock Server
'-----
tcpClient.RemoteHost = strRemoteHost 'IP address
here...localip=ip you're using so that you can test client/server on your
own comp
tcpClient.RemotePort = strRemotePort 'assign port server will
listen for client on
tcpClient.Connect 'try to connect to server
'-----

MnuStartCOMPort.Visible = False
MnuCOMPort.Visible = False
'MnuDelReadSMS.Visible = False
MnuSettings_DBConfiguration.Visible = True
MnuConnectedClients.Visible = False
```

## Project Error

```
"Set Authority rights
If UCase(AuthorityDesig) = UCase("Operator") Then
MnuStartSMSing.Enabled = False
MnuSendSMS.Enabled = False

mnuF_Authority.Enabled = False

MnuCOMPort.Enabled = False
MnuParameter.Enabled = False
MnuSettings_DBConfiguration.Enabled = False
ElseIf UCase(AuthorityDesig) = UCase("User") Then
mnuF_Authority.Enabled = False
MnuSettings_DBConfiguration.Enabled = False
Else
MnuStartSMSing.Enabled = True
MnuSendSMS.Enabled = True

MnuCOMPort.Enabled = True
MnuParameter.Enabled = True
MnuSettings_DBConfiguration.Enabled = True
End If
End If

If gblnIsServer = False Then
MnuSendSMS.Enabled = False
MnuPopupForward.Enabled = False
MnuPopupReply.Enabled = False
End If

" strSMSC_NUMBER = ZDLL.GetFromIni("ZConfigurationINI", "Settings",
"SMSCN", "+919825001002")
" strDataSrc = ZDLL.GetFromIni(ZConfigurationINI, "Data", "DataSrc",
"ZRLY90")
" strUser = ZDLL.GetFromIni(ZConfigurationINI, "Data", "User",
"AFRES")
" strPwd = ZDLL.GetFromIni(ZConfigurationINI, "Data", "Auth",
"GATES9")
"
" If (ZIDLL.cnORA.State = 0) Then
" 'open connection
" If ZIDLL.OPEN_Connection(strDataSrc, strUser, strPwd) = 0 Then
" gstrW2S = "Unable to connect DataSource." & vbCrLf & vbCrLf
_
" & "Check for following possibilities " & vbCrLf & "[1] Check
for " _
" & "parameters like DataSource, User, Password" & vbCrLf &
"[2] Oracle " _
" & "database is not stared yet" & vbCrLf & vbCrLf & "This
application will be " _
" & "closed."
```

## Project Error

```
" MsgBox gstrW2S, vbExclamation
"
" WriteToScreen txtCommStatus, gstrW2S
" WriteToFile ZERRLOG_FILE_PATH, gstrW2S
"
" blnExplicitUnload = True
" Form_QueryUnload 0, 0
" Else
" WriteToFile ZLOG_FILE_PATH,
"-----"
" WriteToFile ZLOG_FILE_PATH, "Database Connection Opened"
" WriteToScreen txtCommStatus, "Database Connection Opened"
" End If
" End If

' Call GetRequestCntUpToDate 'this procedure will retrieve latest
request count of each opt. code

'Set LstItem = lvwOnlineStatus.ListItems.Add(1, "A" & 1, "PFI"):
LstItem.SubItems(1) = SRT.SRT_PFI
'Set LstItem = lvwOnlineStatus.ListItems.Add(2, "A" & 2, "PNI"):
LstItem.SubItems(1) = SRT.SRT_PNI
'Set LstItem = lvwOnlineStatus.ListItems.Add(3, "A" & 3, "PAI"):
LstItem.SubItems(1) = SRT.SRT_PAI
'Set LstItem = lvwOnlineStatus.ListItems.Add(4, "A" & 4, "PLI"):
LstItem.SubItems(1) = SRT.SRT_PLI
'Set LstItem = lvwOnlineStatus.ListItems.Add(5, "A" & 5, "BLI"):
LstItem.SubItems(1) = SRT.SRT_BLI
'Set LstItem = lvwOnlineStatus.ListItems.Add(6, "A" & 6, "UNKNOWN"):
LstItem.SubItems(1) = SRT.SRT_UNKNOWN
Set LstItem = lvwOnlineStatus.ListItems.Add(1, "A" & 1, "Current
Status"): LstItem.SubItems(1) = 0
Set LstItemCrntStatus = lvwOnlineStatus.ListItems.Item(1)

If gblnIsServer = True Then
frmMain.stbrMain.Panels(3).Text = fncDisplayCOMSettings
Else
'frmMain.stbrMain.Panels(3).Visible = False
'Display Winsock Communication Setting
If isClientConnected = True Then
frmMain.stbrMain.Panels(3).Text = "Server : " & strRemoteHost
& ", Port : " & strRemotePort & " Connected"
Else
frmMain.stbrMain.Panels(3).Text = "Server : " & strRemoteHost
& ", Port : " & strRemotePort & " Disconnected"
'txtCommStatus.Text = txtCommStatus.Text & vbCrLf & " Server :
" & strRemoteHost & " not connected..."
End If
End If

"Populate all contacts from [AddressBook] into array to improve
```

## Project Error

```
performance
Call GetNPopulateContacts

' Call GetNPopulateContactsCollection

"Get GrpID from [GroupMaster] for 'General' group
udfGenDetail.udftGenDet_GeneralGrpID = fncGet_GrpID("General")

"Get values from ZConfiguration.Ini
tmrReadFromSIM.Interval = 1500 '5 Seconds ( 1000 mili second = 1
second)
' tmrReadFromSIM.Interval = ZDLL.GetFromIni(ZConfigurationINI,
"Properties", "ProcessInterval", 5) * 1000

If gblnIsServer = True Then
If ZDLL.GetFromIni(ZConfigurationINI, "COMSettings", "DCMode",
"Manually") = "Automatically" Then
Call MnuStartCOMPort_Click 'Start COM Port

'Changed by Ketan On Dt : 28/09/2005
If APPStat.appstCOMPortOn = True Then
If InitializeGSMDevice = False Then Exit Sub

If ZDLL.GetFromIni(ZConfigurationINI, "Properties",
"AutoRespReader", 0) = 1 Then
Call MnuStartSMSing_Click
End If
End If

End If
Else
If ZDLL.GetFromIni(ZConfigurationINI, "Properties",
"AutoRespReader", 0) = 1 Then
Call MnuStartSMSing_Click
End If
End If

'Read Response
ReadSMSFromDB gblnIsServer, False

'Read Request
ReadSMSFromDB gblnIsServer, True

'Added by Ketan On Dt : 05/03/3005 for Viewing sms by sent by all
user...
If gblnIsSpecialClientNode = True Then
ReadRequest
End If

'ReadSMSFromDB gblnIsServer, True
```

## Project Error

```
Load Tray
udfOPTCode.udftOPTCode_SRCH = "SRCH"
gstrFncABID_SRCH = fncGetFncAuthority(udfOPTCode.udftOPTCode_SRCH)

' Set TT = New CTooltip
' TT.Style = TTBalloon
' TT.Icon = TTIconInfo
'
' TT.Title = "Multiline tooltip"
' TT.TipText = "GATES"
' "TT.Create Command1.hwnd
' TT.Create Tray.hwnd

Screen.MousePointer = vbDefault
Exit Sub

ErrH:
If Err.Number = 401 Then Resume Next
Screen.MousePointer = vbDefault

Call LogEntry("frmMain-->Form_Load", ErrLog, True, Err.Description,
Err.Number, Err.Description, Err.Source)

gstrW2S = "frmMain-->Form_Load"
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &
Err.Source
WriteToScreen txtCommStatus, gstrW2S
WriteToFile ZERRLOG_FILE_PATH, gstrW2S
End
End Sub

Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)
On Error GoTo ErrH

Dim lngErrNo As Long, strErrDesc As String, strErrSrc As String

Screen.MousePointer = vbDefault
If gblnDoUWantToLogOff = False Then
If blnExplicitUnload = False Then
If MsgBox("Are you sure you want to quit application?",
vbExclamation + vbYesNo + vbDefaultButton2) = vbYes Then
UpdateClients False, AuthorityPerson, gblnIsServer

ZDLL.SaveToIni ZConfigurationINI, "Properties", "AppHwnd",
0
'ZDLL.WriteToUnknownCallLogFile App.Path &
"\ZUnknown.txt", "1"
'ZDLL.CloseWindow "ZToolTip"
ZDLL.DestroyWindows (ZDLL.GetHandle("ZToolTip"))
'ZDLL.WriteToUnknownCallLogFile App.Path &
```

## Project Error

```
"\ZUnknown.txt", "2"
```

```
If CloseApp = 1 Then 'This procedure will destroy dll  
instance e.g. ZIVRS.dll, ZLINK.dll
```

```
'ZDLL.WriteToUnknownCallLogFile App.Path &  
"\ZUnknown.txt", "3"
```

```
Call ClosePort
```

```
'ZDLL.WriteToUnknownCallLogFile App.Path &  
"\ZUnknown.txt", "4"
```

```
"This will remove icon from system tray
```

```
Call Shell_NotifyIcon(NIM_DELETE, nfIconData)
```

```
WriteToFile ZLOG_FILE_PATH, "Connection Closed":
```

```
WriteToFile ZLOG_FILE_PATH,
```

```
"-----"
```

```
ZDLL.WriteToUnknownCallLogFile App.Path &
```

```
"\ZUnknown.txt", "5"
```

```
ZDLL.SaveToIni ZConfigurationINI, "Properties",
```

```
"AppStatus", 1 '1-normal shutdown,0-abnormal shutdown
```

```
Cancel = 0
```

```
End
```

```
End If
```

```
Else
```

```
Cancel = 1: gblnCloseApp = False
```

```
End If
```

```
Else
```

```
"This will remove icon from system tray
```

```
Call Shell_NotifyIcon(NIM_DELETE, nfIconData)
```

```
Cancel = 0: End
```

```
End If
```

```
Else
```

```
If gblnIsServer = True Then
```

```
Call ClosePort: APPStat.appstCOMPortOn = False
```

```
End If
```

```
Call STOP_SMSing: APPStat.appstSMSingOn = False
```

```
Cancel = 0
```

```
End If
```

```
Exit Sub
```

```
ErrH:
```

```
lngErrNo = Err.Number
```

```
strErrDesc = Err.Description
```

```
strErrSrc = Err.Source
```

```
gstrW2S = "frmMain-->Form_QueryUnload"
```

```
gstrW2S = gstrW2S & "|" & lngErrNo & "|" & strErrDesc & "|" &
```

```
strErrSrc
```

```
WriteToScreen txtCommStatus, gstrW2S
```

```
WriteToFile ZERRLOG_FILE_PATH, gstrW2S
```

```
Call LogEntry("frmMain-->Form_QueryUnload", ErroLog, True, strErrDesc,  
lngErrNo, strErrDesc, strErrSrc)
```

## Project Error

End Sub

```
Private Sub Form_Resize()  
On Error Resume Next
```

```
Dim intXPos As Integer  
Dim strRes As String
```

```
intXPos = InStr(udfGenDetail.udftGenDet_ScreenResolution, "x")  
strRes = Mid(udfGenDetail.udftGenDet_ScreenResolution, 1, intXPos -  
1)
```

```
If Me.WindowState <> vbMinimized Then  
Me.WindowState = vbMaximized
```

```
Screen.MousePointer = vbHourglass  
'Select Case udfGenDetail.udftGenDet_ScreenResolution  
Select Case strRes  
'Case "800x600"  
Case "800"  
lvwOnScreen(0).Left = 75: lvwOnScreen(0).Top = 270:  
lvwOnScreen(0).Height = 2430: lvwOnScreen(0).Width = 11790  
fraCommnicationStatus.Left = 4545:  
fraCommnicationStatus.Top = 5445: fraCommnicationStatus.Height = 2130:  
fraCommnicationStatus.Width = 7350  
txtCommStatus.Left = 120: txtCommStatus.Top = 225:  
txtCommStatus.Height = 1785: txtCommStatus.Width = 7170  
fraOnlineStatus.Left = 45: fraOnlineStatus.Top = 5445:  
fraOnlineStatus.Height = 2130: fraOnlineStatus.Width = 4485  
lvwOnlineStatus.Left = 120: lvwOnlineStatus.Top = 225:  
lvwOnlineStatus.Height = 1815: lvwOnlineStatus.Width = 4230  
stbrMain.Left = 0: stbrMain.Top = 7605: stbrMain.Height =  
345: stbrMain.Width = 11940  
FraControls.Left = 10395: FraControls.Top = 60:  
FraControls.Height = 750: FraControls.Width = 1500  
txtResp.Left = 765: txtResp.Top = 420: txtResp.Height =  
285: txtResp.Width = 180  
txtReq.Left = 1185: txtReq.Top = 360: txtReq.Height = 285:  
txtReq.Width = 180  
lvwOnScreen(1).Left = 75: lvwOnScreen(1).Top = 2970:  
lvwOnScreen(1).Height = 2430: lvwOnScreen(1).Width = 11790  
lblResponseTitle(0).Left = 60: lblResponseTitle(0).Top =  
30: lblResponseTitle(0).Height = 270: lblResponseTitle(0).Width = 1110  
lblResponseTitle(1).Left = 105: lblResponseTitle(1).Top =  
2730: lblResponseTitle(1).Height = 270: lblResponseTitle(1).Width = 1200  
lvwOnScreen(0).ColumnHeaders(5).Width = 5523  
lvwOnScreen(1).ColumnHeaders(5).Width = 5523  
cmdShowOnlineUsers.Move cmdShowOnlineUsers.Left,  
lblResponseTitle(1).Top, cmdShowOnlineUsers.Width,  
cmdShowOnlineUsers.Height
```

Project Error

## Project Error

fraOnlineUser.Top = 2190 – 300  
fraOnlineUser.Left = 3810 – 200  
fraMessage.Top = 3180 – 300  
fraMessage.Left = 4485 – 200

'Case "1024x740"  
Case "1024"

lvwOnScreen(0).Left = 75: lvwOnScreen(0).Top = 270:  
lvwOnScreen(0).Height = 3645: lvwOnScreen(0).Width = 15075  
fraCommnicationStatus.Left = 4545:  
fraCommnicationStatus.Top = 7905: fraCommnicationStatus.Height = 2130:  
fraCommnicationStatus.Width = 10635  
txtCommStatus.Left = 120: txtCommStatus.Top = 225:  
txtCommStatus.Height = 1785: txtCommStatus.Width = 10395  
fraOnlineStatus.Left = 45: fraOnlineStatus.Top = 7905:  
fraOnlineStatus.Height = 2130: fraOnlineStatus.Width = 4485  
lvwOnlineStatus.Left = 120: lvwOnlineStatus.Top = 225:  
lvwOnlineStatus.Height = 1815: lvwOnlineStatus.Width = 4230  
stbrMain.Left = 0: stbrMain.Top = 10425: stbrMain.Height =  
345: stbrMain.Width = 15180  
FraControls.Left = 10395: FraControls.Top = 285:  
FraControls.Height = 750: FraControls.Width = 1500  
txtResp.Left = 765: txtResp.Top = 420: txtResp.Height =  
285: txtResp.Width = 180  
txtReq.Left = 1185: txtReq.Top = 360: txtReq.Height = 285:  
txtReq.Width = 180  
lvwOnScreen(1).Left = 75: lvwOnScreen(1).Top = 4215:  
lvwOnScreen(1).Height = 3645: lvwOnScreen(1).Width = 15075  
lblResponseTitle(0).Left = 60: lblResponseTitle(0).Top =  
30: lblResponseTitle(0).Height = 270: 'lblResponseTitle(0).Width = 1110  
lblResponseTitle(1).Left = 105: lblResponseTitle(1).Top =  
3975: lblResponseTitle(1).Height = 270: 'lblResponseTitle(1).Width = 1200  
lvwOnScreen(0).ColumnHeaders(5).Width = 8800  
lvwOnScreen(1).ColumnHeaders(5).Width = 8800

Case Else '800x600

lvwOnScreen(0).Left = 75: lvwOnScreen(0).Top = 270:  
lvwOnScreen(0).Height = 2430: lvwOnScreen(0).Width = 11790  
fraCommnicationStatus.Left = 4545:  
fraCommnicationStatus.Top = 5445: fraCommnicationStatus.Height = 2130:  
fraCommnicationStatus.Width = 7350  
txtCommStatus.Left = 120: txtCommStatus.Top = 225:  
txtCommStatus.Height = 1785: txtCommStatus.Width = 7170  
fraOnlineStatus.Left = 45: fraOnlineStatus.Top = 5445:  
fraOnlineStatus.Height = 2130: fraOnlineStatus.Width = 4485  
lvwOnlineStatus.Left = 120: lvwOnlineStatus.Top = 225:  
lvwOnlineStatus.Height = 1815: lvwOnlineStatus.Width = 4230  
stbrMain.Left = 0: stbrMain.Top = 7605: stbrMain.Height =

## Project Error

```
345: stbrMain.Width = 11940
FraControls.Left = 10395: FraControls.Top = 60:
FraControls.Height = 750: FraControls.Width = 1500
txtResp.Left = 765: txtResp.Top = 420: txtResp.Height =
285: txtResp.Width = 180
txtReq.Left = 1185: txtReq.Top = 360: txtReq.Height = 285:
txtReq.Width = 180
lvwOnScreen(1).Left = 75: lvwOnScreen(1).Top = 2970:
lvwOnScreen(1).Height = 2430: lvwOnScreen(1).Width = 11790
lblResponseTitle(0).Left = 60: lblResponseTitle(0).Top =
30: lblResponseTitle(0).Height = 270: lblResponseTitle(0).Width = 1110
lblResponseTitle(1).Left = 105: lblResponseTitle(1).Top =
2730: lblResponseTitle(1).Height = 270: lblResponseTitle(1).Width = 1200
lvwOnScreen(0).ColumnHeaders(5).Width = 5923
lvwOnScreen(1).ColumnHeaders(5).Width = 5923
```

```
End Select
Screen.MousePointer = vbDefault
End If
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
Dim i
```

```
'DEVELOPER : Ketan Dt. 04/02/2005
'Purpose : Close Server's Clients Connections
'=====
If gblnIsServer = True Then
For i = 0 To tcpServer.UBound
If tcpServer(i).State = sckOpen Then
tcpServer(i).Close
End If
Next
End If
```

```
'DEVELOPER : Ketan Dt. 04/02/2005
'Purpose : Close Server Connection
'=====
If gblnIsServer = False Then
If tcpClient.State = sckOpen Then
tcpClient.Close
End If
End If
```

```
End Sub
```

```
Private Sub lvwOnLineUsers_ItemCheck(ByVal Item As MSComctlLib.ListItem)
If gblnIsServer = True Then
If lvwOnLineUsers.ListItems.Count > 0 Then
If Item.Checked = True Then
If
```

## Project Error

```
lvwOnLineUsers.ListItems(lvwOnLineUsers.SelectedItem.Index).SubItems(2) =  
ZDLL.GetIPAddress Then
```

```
lvwOnLineUsers.ListItems(lvwOnLineUsers.SelectedItem.Index).Checked =  
False  
End If  
End If  
End If  
End If  
End Sub
```

```
"  
"Private Sub Form_Resize()  
" On Error Resume Next  
" If Me.WindowState <> vbMinimized Then  
" Me.WindowState = vbMaximized  
"" lvwOnScreen(0).Width = Me.Width - 150: lvwOnScreen(1).Width =  
Me.Width - 150  
"" fraCommnicationStatus.Width = (lvwOnScreen(0).Width + 100) -  
(fraCommnicationStatus.Left)  
""  
"" txtCommStatus.Width = fraCommnicationStatus.Width - 200  
""  
"" fraCommnicationStatus.Top = Me.Height -  
(fraCommnicationStatus.Height + stbrMain.Height)  
"" fraOnlineStatus.Top = fraCommnicationStatus.Top  
"  
" End If  
"  
"" Dim ctrl As Control  
""  
"" For Each ctrl In Me.Controls  
"" Debug.Print ctrl.Name & " - L:" & ctrl.Left & " - T:" &  
ctrl.Top & " - H:" & ctrl.Height & " - W:" & ctrl.Width  
"" Next ctrl  
"End Sub
```

```
Private Sub lvwOnScreen_DblClick(Index As Integer)  
On Error GoTo ErrH
```

```
If lvwOnScreen(Index).ListItems.Count > 0 Then  
If lvwOnScreen(Index).SelectedItem.Index = 0 Then  
Exit Sub  
End If  
Else  
Exit Sub  
End If
```

```
If Index = 1 Then
```

```
If
```

Project Error

## Project Error

```
InStr(lvwOnScreen(Index).ListItems(lvwOnScreen(Index).SelectedItem.Index).SubItems(1),
".") > 0 Then
txtMsg.Text =
lvwOnScreen(1).ListItems(lvwOnScreen(1).SelectedItem.Index).SubItems(5)
fraMessage.Tag =
lvwOnScreen(Index).ListItems(lvwOnScreen(Index).SelectedItem.Index).SubItems(1)
cmdSendToUser.Caption = "&Reply"
ChkBroadCast.Enabled = False
fraMessage.Visible = True
fraMessage.ZOrder 0
Exit Sub
End If

End If

If lvwOnScreen(Index).ListItems.Count > 0 Then

If
InStr(lvwOnScreen(Index).ListItems(lvwOnScreen(Index).SelectedItem.Index).SubItems(1),
".") > 0 Then
frmOnScreenDetail.cmdReply.Enabled = False
frmOnScreenDetail.cmdForward.Enabled = False
End If

If Len(Trim(lvwOnScreen(Index).SelectedItem.Tag)) > 0 Then

frmOnScreenDetail.nMsgID =
Val(lvwOnScreen(Index).SelectedItem.Tag)
If Len(Trim(lvwOnScreen(Index).SelectedItem.Tag)) = 0 Then
WriteToFile ZERRLOG_FILE_PATH, "Message ID Not Found"
End If
End If

frmOnScreenDetail.mbytIsRequest = Index
frmOnScreenDetail.mintLVWId =
lvwOnScreen(Index).ListItems(1).Text
frmOnScreenDetail.Show

End If

Exit Sub

ErrH:
Call LogEntry("frmMain-->lvwOnScreen_DblClick", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub

Private Sub lvwOnScreen_GotFocus(Index As Integer)
On Error GoTo ErrH

strFocusedLvw = Index
```

## Project Error

Exit Sub

ErrH:

```
Call LogEntry("frmMain-->lvwOnScreen_GotFocus", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub
```

```
Private Sub lvwOnScreen_KeyPress(Index As Integer, KeyAscii As Integer)  
On Error GoTo ErrH
```

```
If KeyAscii = 13 Then Call MnuSMSDetail_Click  
Exit Sub
```

ErrH:

```
Call LogEntry("frmMain-->lvwOnScreen_KeyPress", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub
```

```
Private Sub lvwOnScreen_MouseDown(Index As Integer, Button As Integer,  
Shift As Integer, X As Single, Y As Single)  
On Error GoTo ErrH
```

```
If Index = 1 Then  
If lvwOnScreen(1).ListItems.Count > 0 Then
```

```
If  
InStr(lvwOnScreen(1).ListItems(lvwOnScreen(1).SelectedItem.Index).SubItems(1),  
".") = 0 Then  
If Button = vbRightButton Then  
MnuPopupForward.Visible = True  
PopupMenu MnuPopup  
End If  
Else  
If Button = vbRightButton Then  
MnuPopupForward.Visible = False  
PopupMenu MnuPopup  
End If  
End If  
End If  
End If  
Exit Sub
```

ErrH:

```
Call LogEntry("frmMain-->lvwOnScreen_MouseDown", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub
```

```
"Private Sub lvwOnScreen_MouseMove(Index As Integer, Button As Integer,  
Shift As Integer, x As Single, y As Single)  
" If lvwOnScreen(1).ListItems.Count > 0 Then  
" TT.Title = App.Title
```

## Project Error

```
" TT.TipText = "Caller ID : " &
lvwOnScreen(1).ListItems(1).SubItems(1) & vbCrLf _
" & "SMS From : " &
lvwOnScreen(1).ListItems(1).SubItems(2) & vbCrLf _
" & "Date Time : " &
lvwOnScreen(1).ListItems(1).SubItems(3) & vbCrLf _
" & "Message : " &
lvwOnScreen(1).ListItems(1).SubItems(4)
" TT.Create lvwOnScreen(1).hWnd
" End If
"End Sub
```

```
Private Sub MnuAbout_Click()
On Error GoTo ErrH
```

```
frmAbout.Show 1
Exit Sub
```

```
ErrH:
Call LogEntry("frmMain-->MnuAbout_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuCalc_Click()
On Error GoTo ErrH
```

```
Dim strCalc$
strCalc = ZDLL.GetSystemDir & "\Calc.exe"
If Dir(strCalc) <> "" Then Shell strCalc
Exit Sub
```

```
ErrH:
gstrW2S = "frmMain-->Form_MnuCalc_Click"
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &
Err.Source
WriteToScreen txtCommStatus, gstrW2S
WriteToFile ZERRLOG_FILE_PATH, gstrW2S
```

```
Call LogEntry("frmMain-->MnuCalc_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuCalendar_Click()
On Error GoTo ErrH
```

```
FrmCalendar.Show 1
Exit Sub
```

```
ErrH:
```

Project Error

## Project Error

```
gstrW2S = "frmMain-->Form_MnuCalendar_Click"  
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &  
Err.Source  
WriteToScreen txtCommStatus, gstrW2S  
WriteToFile ZERRLOG_FILE_PATH, gstrW2S
```

```
Call LogEntry("frmMain-->MnuCalendar_Click", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub
```

```
Private Sub MnuClearScreen_Click()  
On Error GoTo ErrH
```

```
If MsgBox("Are you sure you want to clear the screen?", vbYesNo +  
vbQuestion) = vbYes Then  
'IvwSMS.ListItems.Clear: intIndex = 0  
End If  
Exit Sub
```

```
ErrH:  
gstrW2S = "frmMain-->MnuClearScreen_Click"  
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &  
Err.Source  
WriteToScreen txtCommStatus, gstrW2S  
WriteToFile ZERRLOG_FILE_PATH, gstrW2S  
Call LogEntry("frmMain-->MnuClearScreen_Click", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub
```

```
Private Sub MnuCOMPort_Click()  
On Error GoTo ErrH
```

```
'If gblnStartDC = True Then  
If APPStat.appstCOMPortOn = True Then  
MsgBox "Please stop COM Port to access COM Settings",  
vbExclamation  
Exit Sub  
Else  
frmCOMDetails.Show 1  
End If  
Exit Sub
```

```
ErrH:  
gstrW2S = "frmMain-->MnuCOMPort_Click"  
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &  
Err.Source  
WriteToScreen txtCommStatus, gstrW2S  
WriteToFile ZERRLOG_FILE_PATH, gstrW2S  
Call LogEntry("frmMain-->MnuCOMPort_Click", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub
```

Project Error

## Project Error

```
Private Sub MnuConnectedClients_Click()  
On Error GoTo ErrH
```

```
FrmConnectedClients.Show  
Exit Sub
```

```
ErrH:  
Call LogEntry("frmMain-->MnuDeleteReq_Click", ErrLog, True,  
Err.Description, Err.Number, Err.Description)  
End Sub
```

```
Private Sub MnuDeleteReq_Click()  
On Error GoTo ErrH
```

```
If MsgBox("Are you sure you want to delete selected Request?",  
vbExclamation + vbYesNo + vbDefaultButton2) = vbYes Then
```

```
End If  
Exit Sub
```

```
ErrH:  
Call LogEntry("frmMain-->MnuDeleteReq_Click", ErrLog, True,  
Err.Description, Err.Number, Err.Description)  
End Sub
```

```
"Run this procedure to delete all read messages
```

```
Private Sub MnuDelReadSMS_Click()  
On Error GoTo ErrH
```

```
Dim i As Integer  
Screen.MousePointer = vbHourglass  
If APPStat.appstSMSingOn = True Then  
MsgBox "Please stop Response Reader before deleting all Read  
messages", vbExclamation  
Screen.MousePointer = vbDefault  
Exit Sub  
End If  
If APPStat.appstCOMPortOn = False Then  
MsgBox "Please start COM Port before deleting all Read messages",  
vbExclamation  
Screen.MousePointer = vbDefault  
Exit Sub  
End If
```

```
For i = 1 To 30  
"Delete From SIM  
bytLastCmd = enmDeleteFromSIM: mstrInputOne = ""  
Call fncRequestToCOM(msCOMMSIM, enmDeleteFromSIM, i)  
Do Until blnGoForNextIteration = True: DoEvents: Loop  
Next i
```

Project Error

## Project Error

```
gstrW2S = "User has tried to delete all Read messages"
WriteToScreen txtCommStatus, gstrW2S
WriteToFile ZERRLOG_FILE_PATH, gstrW2S
gstrW2S = ""
Screen.MousePointer = vbDefault
Exit Sub

ErrH:
Screen.MousePointer = vbDefault
gstrW2S = "frmMain-->MnuDelReadSMS_Click"
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &
Err.Source
WriteToScreen txtCommStatus, gstrW2S
WriteToFile ZERRLOG_FILE_PATH, gstrW2S
Call LogEntry("frmMain-->MnuDelReadSMS_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub

Private Sub MnuExit_Click()
On Error GoTo ErrH

' If APPStat.appstDatabaseRunning = True Then
' MsgBox "Please stop Database before closing application",
vbExclamation
' Screen.MousePointer = vbDefault
' Exit Sub
' End If
' If APPStat.appstCOMPortOn = True Then
' MsgBox "Please stop COM Port before closing application",
vbExclamation
' Screen.MousePointer = vbDefault
' Exit Sub
' End If
' If APPStat.appstSMSingOn = True Then
' MsgBox "Please stop SMS'ing before closing application",
vbExclamation
' MsgBox "Please stop Response Reading before closing application",
vbExclamation
' Screen.MousePointer = vbDefault
' Exit Sub
' End If
gblnDoUWantToLogOff = False
Unload Me
" If gblnStartDC = True Then
" MsgBox "Please stop SMS'ing before closing application",
vbExclamation
" Else
" Unload Me
" End If
Exit Sub
```

## Project Error

```
ErrH:
gstrW2S = "frmMain-->MnuExit_Click"
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &
Err.Source
WriteToScreen txtCommStatus, gstrW2S
WriteToFile ZERRLOG_FILE_PATH, gstrW2S
Call LogEntry("frmMain-->MnuExit_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub mnuF_A_AssignRights_Click()
On Error GoTo ErrH
```

```
'Assign Authority Rights
Set_Status "Authority Rights"
Load frmAuthorityMaster
Set_Status ""
frmAuthorityMaster.Show vbModal
Exit Sub
```

```
ErrH:
Call LogEntry("frmMain-->mnuF_A_AssignRights_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub mnuF_A_AuthorityGroup_Click()
On Error GoTo ErrH
```

```
'Assign Authority Group
Set_Status "Authority Group"
Load frmAuthorityGroupMaster
Set_Status ""
frmAuthorityGroupMaster.Show vbModal
Exit Sub
```

```
ErrH:
Call LogEntry("frmMain-->mnuF_A_AuthorityGroup_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
"Private Sub MnuFile_AcBalEntry_Click()
" On Error GoTo ErrH
"
```

```
" frmAcBalEntry.AddFlag = True
" frmAcBalEntry.Show vbModal
" Exit Sub
"
```

```
"ErrH:
" Call LogEntry("frmMain-->MnuFile_AcBalEntry_Click", ErrLog, True,
```

## Project Error

```
Err.Description, Err.Number, Err.Description, Err.Source)  
"End Sub
```

```
Private Sub MnuFile_AddressBook_Click()  
frmContacts.Show  
End Sub
```

```
Private Sub MnuFile_Functions_Click()  
frmFunctions.AddFlag = True  
frmFunctions.Show  
End Sub
```

```
Private Sub MnuFile_QuickMessage_Click()  
frmQuickMsg.AddFlag = True  
frmQuickMsg.Show  
End Sub
```

```
Private Sub MnuLogOff_Click()  
gblnDoUWantToLogOff = True  
'frmMDI.Hide  
'Call MnuStartSMSing_Click  
'frmAuthenticate.Show 1  
Unload frmMain  
frmAuthenticate.Show vbModal  
End Sub
```

```
Private Sub MnuParameter_Click()  
On Error GoTo ErrH
```

```
frmParameter.Show 1  
Exit Sub
```

```
ErrH:  
gstrW2S = "frmMain-->MnuParameter_Click"  
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &  
Err.Source  
WriteToScreen txtCommStatus, gstrW2S  
WriteToFile ZERRLOG_FILE_PATH, gstrW2S  
Call LogEntry("frmMain-->MnuParameter_Click", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub
```

```
"Private Function CloseApp() As Integer  
" On Error GoTo ErrH  
"  
" CloseApp = 0  
" If ZIDLL.cnDB.State = 1 Then ZIDLL.cnDB.Close  
"  
" Set ZIDLL.cnDB = Nothing  
" Set ZDLL = Nothing  
" CloseApp = 1
```

Project Error

## Project Error

```
" Exit Function
"
"ErrH:
" CloseApp = 0
" gstrW2S = "frmMain-->CloseApp"
" gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &
Err.Source
" WriteToScreen txtCommStatus, gstrW2S
" WriteToFile ZERRLOG_FILE_PATH, gstrW2S
"End Function
```

```
Private Sub FillOnlineStatus(strReqType As String)
On Error GoTo ErrH
```

```
Dim LstItem As ListItem
Dim intInc As Integer
```

```
Select Case UCase(SMSData.SMSDS_Type)
Case UCase("PFI")
SRT.SRT_PFI = SRT.SRT_PFI + 1
Case UCase("PNI")
SRT.SRT_PNI = SRT.SRT_PNI + 1
Case UCase("PLI")
SRT.SRT_PLI = SRT.SRT_PLI + 1
Case UCase("PAI")
SRT.SRT_PAI = SRT.SRT_PAI + 1
Case UCase("BLI")
SRT.SRT_BLI = SRT.SRT_BLI + 1
Case Else
SRT.SRT_UNKNOWN = SRT.SRT_UNKNOWN + 1
End Select
```

```
Select Case UCase(SMSData.SMSDS_Type)
Case UCase("PFI")
Set LstItem = lvwOnlineStatus.ListItems.Item(1)
LstItem.SubItems(1) = SRT.SRT_PFI
Case UCase("PNI")
Set LstItem = lvwOnlineStatus.ListItems.Item(2)
LstItem.SubItems(1) = SRT.SRT_PNI
Case UCase("PLI")
Set LstItem = lvwOnlineStatus.ListItems.Item(4)
LstItem.SubItems(1) = SRT.SRT_PLI
Case UCase("PAI")
Set LstItem = lvwOnlineStatus.ListItems.Item(3)
LstItem.SubItems(1) = SRT.SRT_PAI
Case UCase("BLI")
Set LstItem = lvwOnlineStatus.ListItems.Item(5)
LstItem.SubItems(1) = SRT.SRT_BLI
Case Else
Set LstItem = lvwOnlineStatus.ListItems.Item(6)
LstItem.SubItems(1) = SRT.SRT_UNKNOWN
```

## Project Error

```
End Select
intInc = intInc + 1
lvwOnlineStatus.ListItems(intInc).Selected = True
Exit Sub

ErrH:
gstrW2S = "frmMain-->FillOnlineStatus"
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &
Err.Source
WriteToScreen txtCommStatus, gstrW2S
WriteToFile ZERRLOG_FILE_PATH, gstrW2S
Call LogEntry("frmMain-->FillOnlineStatus", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub

Private Sub FillSMS(bytFT As FillType)
' On Error GoTo ErrH
'
' Dim LstItem As ListItem
'
' If intIndex = 0 Then intIndex = 1
''' "Z.GSM will reply same request from same mobile but following
lines are for checking of duplicate request
''' "generated anyhow. e.g. if
smsdata.smsds_type,smsdata.smsds_request,smsdata.smsds_requestdt are same
then
''' "consider this request as duplicate and avoid to display
''' "Upto Here
'
' If bytFT = FT_ADD Then
' Set LstItem = lvwSMS.ListItems.Add(intIndex, "A" & intIndex,
SMSData.SMSDS_CallerID)
' If SMSData.SMSDS_Type = LstItem.SubItems(1) And _
' SMSData.SMSDS_Request = LstItem.SubItems(2) And _
' SMSData.SMSDS_RequestDT = LstItem.SubItems(3) Then
' Exit Sub
' End If
'
' LstItem.SubItems(1) = SMSData.SMSDS_Type
' LstItem.SubItems(2) = SMSData.SMSDS_Request
' LstItem.SubItems(3) = SMSData.SMSDS_RequestDT
' LstItem.SubItems(5) = SMSData.SMSDS_Status
'
' LstItem.Tag = intIndex
' intIndex = intIndex + 1
' Else
' Set LstItem = lvwSMS.ListItems.Item(intIndex - 1)
' LstItem.SubItems(4) = SMSData.SMSDS_ResponseDT
' LstItem.SubItems(5) = SMSData.SMSDS_Status
' LstItem.SubItems(6) = SMSData.SMSDS_Response
```

## Project Error

```
' End If
' lvwSMS.ListItems(intIndex - 1).Selected = False
' lvwSMS.ListItems(1).Selected = True
' Exit Sub
'
'ErrH:
" Resume
' gstrW2S = "frmMain-->FillSMS"
' gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &
Err.Source
' WriteToScreen txtCommStatus, gstrW2S
' WriteToFile ZERRLOG_FILE_PATH, gstrW2S
'End Sub

Private Sub MakeSMSDataEmpty()
SMSData.SMSDS_CallerID = ""
SMSData.SMSDS_Request = ""
SMSData.SMSDS_RequestDT = ""
SMSData.SMSDS_Response = ""
SMSData.SMSDS_ResponseDT = ""
SMSData.SMSDS_Status = ""
SMSData.SMSDS_Type = ""
End Sub

Private Sub MakeOtherVariablesEmpty()
blnGoForDelete = False
blnGoForNextIteration = False
blnGoForRead = False
blnGoForRespGen = False
blnGoForSend = False

strResp1 = ""
strResp2 = ""
End Sub

Private Sub MnuPopupForward_Click()
On Error GoTo ErrH

'frmSendSMS.txtSMSToContacts.Text =
lvwOnScreen(1).SelectedItem.ListSubItems(2)
'frmSendSMS.txtSMSTo.Text =
lvwOnScreen(1).SelectedItem.ListSubItems(1)
frmSendSMS.Caption = frmSendSMS.Caption & " - Forward"
frmSendSMS.txtSMSMsg.Text =
lvwOnScreen(1).SelectedItem.ListSubItems(4)
frmSendSMS.Show vbModal
Exit Sub

ErrH:
Call LogEntry("frmMain-->MnuPopupReply_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
```

## Project Error

End Sub

```
Private Sub MnuPopupReply_Click()
On Error GoTo ErrH
If
InStr(lvwOnScreen(1).ListItems(lvwOnScreen(1).SelectedItem.Index).SubItems(1),
".") = 0 Then
frmSendSMS.txtSMSToContacts.Text =
lvwOnScreen(1).SelectedItem.ListSubItems(2)

frmSendSMS.txtSMSTo.Text =
lvwOnScreen(1).SelectedItem.ListSubItems(1)
frmSendSMS.Caption = frmSendSMS.Caption & " – Reply"
frmSendSMS.Show vbModal
Else
fraMessage.Tag =
lvwOnScreen(1).ListItems(lvwOnScreen(1).SelectedItem.Index).SubItems(1)
ChkBroadCast.Enabled = False
fraMessage.Visible = True
fraMessage.ZOrder 0
End If
```

Exit Sub

```
ErrH:
Call LogEntry("frmMain-->MnuPopupReply_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuRefreshSMS_Click()
On Error GoTo ErrH
```

Static nCnt As Byte

```
If APPStat.appstSMSingOn = True Then
MsgBox "Please stop SMS'ing before deleting all Read messages",
vbExclamation
MsgBox "Please stop Response Reader before deleting all Read
messages", vbExclamation
Screen.MousePointer = vbDefault
Exit Sub
End If
If APPStat.appstCOMPortOn = False Then
MsgBox "Please start COM Port before deleting all Read messages",
vbExclamation
Screen.MousePointer = vbDefault
Exit Sub
End If
```

```
Screen.MousePointer = vbHourglass
stbrMain.Panels(1).Text = "Re-initializing GSM Device, Please wait
```

Project Error

## Project Error

```
..."  
"Initialize SIM  
TTPTryAgain:  
If nCnt >= 255 Then nCnt = 0  
nCnt = nCnt + 1  
bytLastCmd = enmInitialization: mstrInputOne = ""  
Call fncRequestToCOM(msCOMMSIM, enmInitialization)  
'Do Until blnGoForRead = True: DoEvents: Loop  
" Wait for few seconds  
tmpTimer = ""  
tmpTimer = Timer + 10  
Do Until Timer > tmpTimer  
If blnGoForRead = True Then  
tmpTimer = "": Screen.MousePointer = vbDefault  
stbrMain.Panels(1).Text = "": Exit Do  
End If  
DoEvents  
Loop  
tmpTimer = ""  
If blnGoForRead = False Then  
If MsgBox("Initialization failed. Check cable connection between  
GSM device and computer" & vbCrLf & "Do you want to re-initialize GSM  
now?", vbExclamation + vbYesNo + vbDefaultButton2, App.title & " - Attempt  
:" & nCnt) = vbYes Then  
GoTo TTPTryAgain  
End If  
End If  
Screen.MousePointer = vbDefault  
stbrMain.Panels(1).Text = ""  
Exit Sub  
  
ErrH:  
Screen.MousePointer = vbDefault  
stbrMain.Panels(1).Text = ""  
gstrW2S = "frmMain-->MnuRefreshSMS_Click"  
gstrW2S = gstrW2S & "|" & Err.Number & "|" & Err.Description & "|" &  
Err.Source  
WriteToScreen txtCommStatus, gstrW2S  
WriteToFile ZERRLOG_FILE_PATH, gstrW2S  
Call LogEntry("frmMain-->MnuRefreshSMS_Click", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub  
  
Private Sub MnuRep_ConnectedClients_Click()  
On Error GoTo ErrH  
  
udfenmREPORTS = enmCONNECTEDCLIENTS  
ZReports.Visible = True  
ZReports.Show  
Exit Sub
```

## Project Error

```
ErrH:
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuRep_ConnectedClients_Click", ErrLog,
True, Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuRep_Contacts_Click()
On Error GoTo ErrH
```

```
udfenmREPORTS = enmContacts
ZReports.Visible = True
ZReports.Show
Exit Sub
```

```
ErrH:
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuRep_Contacts_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuRep_Functions_Click()
On Error GoTo ErrH
```

```
udfenmREPORTS = enmFunctions
ZReports.Visible = False
ZReports.Show
Set ZReports = Nothing
Exit Sub
```

```
ErrH:
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuRep_Functions_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuRep_GroupDetail_Click()
On Error GoTo ErrH
```

```
udfenmREPORTS = enmGroupDetail
ZReports.Visible = False
ZReports.Show
Set ZReports = Nothing
Exit Sub
```

```
ErrH:
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuRep_GroupDetail_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuRep_OrganizationDetail_Click()
```

## Project Error

On Error GoTo ErrH

```
udfenmREPORTS = enmOrganizationDetail
ZReports.Visible = False
ZReports.Show
Set ZReports = Nothing
Exit Sub
```

ErrH:

```
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuRep_OrganizationDetail_Click", ErrLog,
True, Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuRep_QuickMessages_Click()
On Error GoTo ErrH
```

```
udfenmREPORTS = enmQuickMessages
ZReports.Visible = False
ZReports.Show
Set ZReports = Nothing
Exit Sub
```

ErrH:

```
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuRep_QuickMessages_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuRep_SMSTransactions_Click()
On Error GoTo ErrH
```

```
udfenmREPORTS = enmSMSTransaction
ZReports.Visible = True
ZReports.Show
Exit Sub
```

ErrH:

```
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuRep_SMSTransactions_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub
```

```
Private Sub MnuReports_Click()
' MsgBox "You are not registered to view reports", vbExclamation
End Sub
```

```
Private Sub MnuSendSMS_Click()
On Error GoTo ErrH
```

```
Screen.MousePointer = vbHourglass
```

Project Error

## Project Error

```
If gblnIsServer = True Then
" If APPStat.appstSMSingOn = True Then
" MsgBox "Please stop Response Reader before sending SMS",
vbExclamation
" Screen.MousePointer = vbDefault
" Exit Sub
" End If
If APPStat.appstCOMPortOn = False Then
MsgBox "Please start COM Port before sending SMS",
vbExclamation
Screen.MousePointer = vbDefault
Exit Sub
End If

End If
Screen.MousePointer = vbDefault
frmSendSMS.Show 1
Exit Sub

ErrH:
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuSendSMS_Click", ErrLog, True,
Err.Description, Err.Number, Err.Description, Err.Source)
End Sub

Private Sub MnuSettings_DBConfiguration_Click()
On Error GoTo ErrH

frmDBConfiguration.bytdbProvider = dbProvider
frmDBConfiguration.Show vbModal
Exit Sub

ErrH:
Screen.MousePointer = vbDefault
Call LogEntry("frmMain-->MnuSettings_DBConfiguration_Click", ErrLog,
True, Err.Description, Err.Number, Err.Description, Err.Source)
End Sub

Private Sub MnuSMSDetail_Click()
On Error GoTo ErrH

'lvwOnScreen.Item.Index
If Len(strFocusedLvw) = 0 Then Exit Sub
'MsgBox lvwOnScreen(CInt(strFocusedLvw)).SelectedItem
If lvwOnScreen(CInt(strFocusedLvw)).ListItems().Count > 0 Then
frmOnScreenDetail.mbytIsRequest = CByte(strFocusedLvw)
frmOnScreenDetail.mintLVWid =
lvwOnScreen(CInt(strFocusedLvw)).ListItems(1).Text
frmOnScreenDetail.Show
End If
```

## Project Error

Exit Sub

ErrH:

```
Call LogEntry("frmMain-->MnuSMSDetail_Click", ErrLog, True,  
Err.Description, Err.Number, Err.Description, Err.Source)  
End Sub
```

```
Private Sub MnuStartDatabase_Click()  
On Error GoTo ErrH
```

```
Screen.MousePointer = vbHourglass  
If APPStat.appstDatabaseRunning = False Then  
' strSMSC_NUMBER = ZDLL.GetFromIni("ZConfigurationINI", "Settings",  
"SMSCN", "+919825001002")  
strDataSrc = ZDLL.GetFromIni(ZConfigurationINI, "Data", "DataSrc",  
"ZRLY90")  
strUser = ZDLL.GetFromIni(ZConfigurationINI, "Data", "User",  
"AFRES")  
strPwd = ZDLL.GetFromIni(ZConfigurationINI, "Data", "Auth",  
"GATES9")
```

```
If (ZIDLL.cnDB.State = 0) Then  
'open connection  
'If ZIDLL.OPEN_Connection(strDataSrc, strUser, strPwd) = 0  
Then  
If ZIDLL.OPEN_Connection(enmSQL, False) = 0 Then  
gstrW2S = "Unable to connect D 'Dialing.... alert...  
'-----  
If ComString Like "*ATD*" Then
```

```
Debug.Print ComString & " Input Len : " & msCOMMSIM.InputLen & "  
Buffer : " & msCOMMSIM.InBufferCount  
spl = Split(ComString, "ATD")
```

```
If UBound(spl) = 1 Then
```

```
strFilter = Mid(spl(1), 1, InStr(1, spl(1), ";"))  
spl(1) = strFilter  
spl(1) = Replace(spl(1), ";", "")
```

```
cCallerNAME = fncGetSenderNameFromDB(ZDLL.AV(spl(1)))
```

```
bytServerValue = ZDLL.GetFromIni(ZConfigurationINI,  
"ApplicationType", "Server")
```

```
strCmdToolTip = vbCrLf & " Dailing to ..." & vbCrLf & vbCrLf &  
" Date Time : " & Format(Now, "dd/MM/yyyy hh:mm:ss") & vbCrLf & " Number :  
& spl(1) & vbCrLf & " Name : " & cCallerNAME
```

```
txtCommStatus.Text = txtCommStatus.Text & vbCrLf &
```

Project Error

## Project Error

strCmdToolTip

fncBroadcast\_CALL ZDLL.AV(strCmdToolTip) 'Broadcast Caller  
Number to all client...

If bytServerValue = 1 Then

```
*****  
'Comment by Ketan On Dt : 13/07/2005
```

```
*****  
'strCmdToolTip = App.Path & "\ZToolTip.exe" & " " &  
Chr(34) & strCmdToolTip & Chr(34)  
'Shell strCmdToolTip, vbNormalFocus
```

```
*****
```

```
*****  
'Added by Ketan On dt : 13/07/2005 Adding new Popup Style
```

```
*****  
If ZDLL.GetFromIni(ZConfigurationINI, "Notification",  
"OUTGOINGCALL", "1") = "1" Then  
Set AlertWindow = New frmAlertWindow  
m_iBackColor = RGB(160, 195, 255)  
AlertWindow.DisplayMessage strCmdToolTip, 4, _  
True, True, True, m_iBackColor, False  
End If
```

```
*****
```

End If

End If

End If

```
'-----  
'Incomming Call..... alert....  
'-----
```

```
If ComString Like "*CLIP*" Then  
spl = Split(ComString, Chr(34))
```

```
If UBound(spl) > 1 Then  
Debug.Print "Call Detected.... " & spl(1)
```

```
cCallerNAME = fncGetSenderNameFromDB(ZDLL.AV(spl(1)))
```

```
bytServerValue = ZDLL.GetFromIni(ZConfigurationINI,  
"ApplicationType", "Server")
```

## Project Error

```
strCmdToolTip = vbCrLf & " Calling From ... " & vbCrLf &  
vbCrLf & " Date Time : " & Format(Now, "dd/MM/yyyy hh:mm:ss") & vbCrLf & "  
Caller ID: " & spl(1) & vbCrLf & " Caller Name : " & cCallerNAME
```

```
txtCommStatus.Text = txtCommStatus.Text & vbCrLf &  
strCmdToolTip
```

```
If Len(Trim(glbCalle
```