

PUSH e-MAIL (FreePOP Edition)

version 1.0

Guide for System Integrator and Software Developer

Table of Contents

COPYRIGHT.....	2
DISCLAIMER.....	2
ACKNOWLEDGEMENT.....	2
REVISION HISTORY.....	2
INTRODUCTION.....	3
BENEFITS.....	3
REQUIREMENT.....	3
LICENSE AGREEMENT.....	3
SECTION 1: INSTALLATION.....	4
SECTION 2: SOURCE CODE (VISUAL BASIC 6).....	5
2.1:MAIN PROGRAM CODE.....	5
2.2 GSM MODEM SETTING CODE	12
2.3 EMAIL SETTING CODE	15
SECTION 3: GRAPICAL USER INTERFACE.....	19
SECTION 4: WARRANTY AND SUPPORT.....	21

COPYRIGHT

Copyright © MOBITEK System Sdn. Bhd. 2008. All rights reserved. No part of this document may be reproduced, distributed, stored in a retrieval system or translated into any language, in any form or by any means, electronic, mechanical, magnetic, optical, photocopying, manual or otherwise, without the prior written permission of MOBITEK System Sdn. Bhd.

DISCLAIMER

MOBITEK makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Further, MOBITEK reserves the right to revise this publication and to make changes from time to time in the contents hereof without obligation to notify any person of such revision or changes. Trademarks and Registered Trademarks Products and product names mentioned in this document may be trademarks or registered trademarks of their respective owners.

ACKNOWLEDGEMENT

1. **Push e-Mail (FreePOP Edition)** is developed by Mr. Aizuddin Aizat B. Abdul Rahman, a student from MULTIMEDIA UNIVERSITY MALAYSIA.
2. **FreePOP ActvieX Control** is developed by Glenn Harm, who is employed by a large computer company as a computer consultant. His current expertise is working on large scale Microsoft Exchange Server deployments. He also enjoys all possible ways to keep up with the changing face of computer technology and spends way too much time with his computer and the Internet. For more information, please refer to -- http://www.harm.net/FreePOP/FreePOP_Doc.asp

REVISION HISTORY

<i>EDITION</i>	<i>ISSUED DATE</i>	<i>REMARK</i>
1 st	16 th of August, 2007	Draft release
2 nd	21 st of March, 2008	Revised guide.

INTRODUCTION

Push e-Mail is getting popular in business. However, to implement Push e-Mail, you need to fork-out a large sum of money to invest in BlackBerry system or Microsoft Exchange Server as well as BlackBerry handsets or Windows Mobile handsets. Good news is **Push e-Mail (FreePOP Edition)** is a low cost alternative to BlackBerry and Microsoft Exchange.

Push e-Mail (FreePOP Edition) will connect to an e-mail account (POP3), will check for e-mails and push the e-mail to recipient's mobile phone via SMS. It works with any mobile phones.

BENEFITS

1. Easy to set-up.
2. Works with any brands of hand phone.
3. Low cost as you do not need to invest in BlackBerry Enterprise Server, Microsoft Exchange Server, BlackBerry handsets, nor Windows Mobile handsets.
4. Source code can be modified to suite your business rule eg. What kind of e-mail to push, push to whom, when to push etc.

LIMITATIONS

1. Only works with e-mail with POP3. Does not work with IMAP.
2. Only can push e-mail with 160 characters limit.

REQUIREMENT

1. You must be a software developer or system integrator having knowledge in:
 - a) programming skill;
 - b) POP3;
 - c) ActiveX DLL (COM);
 - d) ActiveX Control; and
 - e) Windows Operating System
2. An e-mail account.
3. A Windows PC (98 – 2003) with internet connection; *Vista* is not tested.
4. A *SMS modem* supplied by MOBITEK System.
5. A SIM card.
6. A mobile phone.

LICENSE AGREEMENT

MOBITEK System grants you the right to use the **Push e-Mail (FreePOP Edition)** subject to the following terms and conditions:

1. **Push e-Mail (FreePOP Edition)** is **Free Of Charge** when SMS modem is purchased from MOBITEK System. It is free from any registration, licensing or deployment fees. It may be deployed freely for commercial or non-commercial.
2. You have the right to modify and distribute the source code of **Push e-Mail (FreePOP Edition)** for commercial use; however you must give credit the to the author of **FreePOP ActiveX control**.
3. **Push e-Mail (FreePOP Edition)** must be used with the SMS modem supplied by MOBITEK System.
4. **Push e-Mail (FreePOP Edition)** is provided on "as-is" basis; without warranty of any kind, express, implied or otherwise; without any technical support.
5. MOBITEK System must not be held liable for any data loss, damages, loss of profits or any other kind of loss while using or misusing **Push e-Mail (FreePOP Edition)**.

SECTION 1: INSTALLATION

1. Connect the GSM modem (refer to installation guide in the CD)
2. Install the Push E-mail in your operating system
3. Run the the application

SECTION 2: SOURCE CODE (VISUAL BASIC 6)

You have the right to modify and distribute the source code of Push e-Mail (FreePOP Edition) for commercial use; however you must give credit the to the author of FreePOP ActiveX control.

2.1:MAIN PROGRAM CODE

```
Private Declare Function GetPrivateProfileString Lib "kernel32" Alias "GetPrivateProfileStringA" (ByVal
lpApplicationName As String, ByVal lpKeyName As String, ByVal lpDefault As String, ByVal lpReturnedString
As String, ByVal nSize As Long, ByVal lpFileName As String) As Long
```

```
Private Declare Function WritePrivateProfileString Lib "kernel32" Alias "WritePrivateProfileStringA" (ByVal
lpApplicationName As String, ByVal lpKeyName As Any, ByVal lpString As Any, ByVal lpFileName As String)
As Long
```

```
Dim blnStop As Boolean
```

```
Dim blnChecking As Boolean
```

```
Dim sSMSText As String
```

```
Dim strPath As String
```

```
Dim sFrom As String
```

```
Dim sDate As String
```

```
Dim sSubject As String
```

```
Dim sBody As String
```

```
Dim sReceiver As String
```

```
Dim sPort As String
```

```
Dim sPIN As String
```

```
Public Function ReadINI(strsection As String, strkey As String, strfullpath As String) As String
```

```
.....
```

```
'read ini
```

```
'x = readini("Example INI", "Example", "C:\Example.ini")
```

```
'MsgBox X
```

```
.....
```

```
Dim strbuffer As String
```

```
Let strbuffer$ = String$(750, Chr$(0&))
```

```
Let ReadINI$ = Left$(strbuffer$, GetPrivateProfileString(strsection$, ByVal LCase$(strkey$), "", strbuffer,
Len(strbuffer), strfullpath$))
```

```
End Function
```

```
Public Sub WriteINI(strsection As String, strkey As String, strkeyvalue As String, strfullpath As String)
```

```
.....
```

```
'write ini
```

```
'Call writeini("Example INI", "Example", "Yes", "C:\Example.ini")
```

```
.....
```

```
Call WritePrivateProfileString(strsection$, UCase$(strkey$), strkeyvalue$, strfullpath$)
```

```
End Sub
```

```
Private Sub btnStart_Click()
```

```
Dim intCount As Integer
```

```
strPath = App.Path
```

```
'when user click start button
```

```
btnStart.Enabled = False
```

```
btnStop.Enabled = True
```

```
blnStop = False
```

```
'SV: start the timer
```

```
Me.Timer.Enabled = True
```

```
Me.Timer.Interval = 30000
```

```
Me.txtLog.Text = "Service is started!"
```

```
'A:read data from EMail Setting
```

```
sFrom = ReadINI("EMail_Setting", "Sender", strPath & "\Setting.INI")
```

```
sDate = ReadINI("EMail_Setting", "Date", strPath & "\Setting.INI")
```

```
sSubject = ReadINI("EMail_Setting", "Subject", strPath & "\Setting.INI")
```

```
sBody = ReadINI("EMail_Setting", "Body", strPath & "\Setting.INI")
```

```
sReceiver = ReadINI("EMail_Setting", "Receiver", strPath & "\Setting.INI")
```

```
'A:read data from GSM Modem Setting
```

```
sPort = ReadINI("GSM_Modem_Setting", "Port", strPath & "\Setting.INI")
```

```
sPIN = ReadINI("GSM_Modem_Setting", "PIN", strPath & "\Setting.INI")
```

```
sModemNumber = ReadINI("GSM_Modem_Setting", "SIM Number", strPath & "\Setting.INI")
```

End Sub

Private Sub btnStop_Click()

'A: When user push stop button, timer will be stop
'the application will be suspend
blnStop = True

If blnChecking = False Then

Me.btnStart.Enabled = True

Me.btnStop.Enabled = False

Me.Timer.Enabled = False

Me.txtLog.Text = "Service is stopped!"

Exit Sub

End If

End Sub

Private Sub Form_Load()

'A:when load the main form, timer and stop button will be disable

Me.Timer.Enabled = False

Me.btnStart.Enabled = True

Me.btnStop.Enabled = False

End Sub

Private Sub mnuemail_Click()

'A:call EMail_Setting window

Call EMail_Setting.Show

End Sub

Private Sub mnuModem_Click()

'A:call GSM_Modem_Setting window

```
Call GSM_Modem_Setting.Show
End Sub
```

```
Private Sub Timer_Timer()
```

```
Dim SMS As New MobitekSMSAPI5.SMS
Dim intModemStatus As Integer
Dim intChr As Integer
```

```
    blnChecking = True
```

```
    'if user click stop, then do not run the code
```

```
    If blnStop = True Then
```

```
        Me.btnStart.Enabled = True
```

```
        Me.btnStop.Enabled = False
```

```
        Me.Timer.Enabled = False
```

```
        Me.txtLog.Text = "Servie is stopped!"
```

```
        Exit Sub
```

```
    End If
```

```
'A:when the click start button, when timer reaches 1 min, then run the code here ....
```

```
'A:connect to the host using pop3 using setting from EMail_Setting data from .ini file
```

```
FreePOP1.POPHostname = ReadINI("EMail_Setting", "Email address", strPath & "\Setting.INI")
```

```
FreePOP1.POPUsername = ReadINI("EMail_Setting", "ID", strPath & "\Setting.INI")
```

```
FreePOP1.POPPassword = ReadINI("EMail_Setting", "Password", strPath & "\Setting.INI")
```

```
'SV: display status
```

```
Me.txtLog.Text = "Connecting to mail server ..."
```

```
'A:connect to email
```

```
FreePOP1.Connect
```

```
'A:if there is no error connecting to email account
```

```
If FreePOP1.Error = 0 Then
```

```
    'SV: if there is no e-mail in inbox, then exit
```

```
If FreePOP1.MsgCount = 0 Then
  Me.txtLog = "There is no e-mail."
  Exit Sub
End If
```

```
'A: set the message to the 1st message
FreePOP1.SetCurrentMsg (1)
```

```
If FreePOP1.Error = 0 Then
  'A;retrieve email from email account
  FreePOP1.RetrieveCurrentMsg
```

```
DoEvents
```

```
If FreePOP1.Error = 0 Then
```

```
  'sender
```

```
  If sFrom = "1" Then
```

```
    sSMSText = "Fr:" & FreePOP1.MsgFrom
```

```
  Else
```

```
    sSMSText = ""
```

```
  End If
```

```
  'date of the email
```

```
  If sDate = "1" Then
```

```
    sSMSText = sSMSText & vbCrLf & "Date:" & FreePOP1.MsgDate
```

```
  Else
```

```
    sSMSText = sSMSText
```

```
  End If
```

```
  'title/subject of the email
```

```
  If sSubject = "1" Then
```

```
    sSMSText = sSMSText & vbCrLf & "Sub: " & FreePOP1.MsgSubject
```

```
  Else
```

```
    sSMSText = sSMSText
```

```
  End If
```

```
  'content/nody of the email
```

```
  If sBody = "1" Then
```

```
    sSMSText = sSMSText & vbCrLf & "Body: " & FreePOP1.MsgContents
```

```
  Else
```

```
sSMSText = sSMSText  
End If
```

```
Else  
    MsgBox FreePOP1.ErrorText, vbOKOnly, "Error During RetrieveCurrentMsg"  
End If
```

```
Else  
    MsgBox FreePOP1.ErrorText, vbOKOnly, "Error during SetCurrentMsg"  
End If
```

```
Else  
    MsgBox FreePOP1.ErrorText, vbOKOnly, "Cannot connect to mail server!"  
    blnChecking = False  
    Exit Sub  
End If
```

```
Me.txtLog.Text = sSMSText  
MsgBox sSMSText
```

```
'send using smsApi v5  
'initialize modem  
intModemStatus = SMS.ModemInit(sPort, sPIN)  
Select Case intModemStatus
```

```
Case 0  
    txtLog = txtLog & vbCrLf & "GSM Modem Not Connected!"  
    Exit Sub
```

```
Case 1  
    txtLog = txtLog & vbCrLf & "GSM Modem Connected!"
```

```
Case 2  
    txtLog = txtLog & vbCrLf & "PIN Required!"  
    Exit Sub
```

```
Case 3  
    txtLog = txtLog & vbCrLf & "Incorrect PIN entered! Warning: after 3 tries of incorrect PIN entered, your  
SIM card will be blocked by TELCO!"  
    Exit Sub
```

```
Case 4
```

```
txtLog = txtLog & vbCrLf & "Your SIM card is blocked by TELCO!"
Exit Sub
Case 5
txtLog = txtLog & vbCrLf & "Your SIM card has problem!"
Exit Sub
Case Else
txtLog = txtLog & vbCrLf & "GSM Modem Not Connected!"
Exit Sub
End Select
```

"You can use this API function anywhere in your code to check the modem's connection to GSM network
If SMS.ConnectToGSM = True Then

```
txtLog = txtLog & vbCrLf & "GSM modem is connected to GSM Network."
Else
txtLog = txtLog & vbCrLf & "GSM modem is disconnected from the GSM Network. Please re-initialised
GSM modem."
```

```
'disconnect email account
FreePOP1.Disconnect
```

```
'Close modem connection to PC
SMS.ModemClose
Set SMS = Nothing
```

```
Exit Sub
End If
```

```
'SV: limit to 160?
If bln160 = True Then
sSMSText = Mid(sSMSText, 1, 160)
'SV: send out SMS here ...
If SMS.SendSMS(sReceiver, sSMSText) = True Then
'after sent = True then delete the current e-mail
FreePOP1.DeleteCurrentMsg
'disconnect email account
```

FreePOP1.Disconnect

Me.txtLog = "Mail successfully sent to " & sReceiver & " as SMS!"

Else

Me.txtLog = "Mail fail to sent to " & sReceiver & " as SMS!"

End If

Else

Me.txtLog = "only 160 character will be sent out. "

End If

blnChecking = False

'Close modem connection to PC

SMS.ModemClose

Set SMS = Nothing

End Sub

2.2 GSM MODEM SETTING CODE

Private Declare Function GetPrivateProfileString Lib "kernel32" Alias "GetPrivateProfileStringA" (ByVal lpApplicationName As String, ByVal lpKeyName As String, ByVal lpDefault As String, ByVal lpReturnedString As String, ByVal nSize As Long, ByVal lpFileName As String) As Long

Private Declare Function WritePrivateProfileString Lib "kernel32" Alias "WritePrivateProfileStringA" (ByVal lpApplicationName As String, ByVal lpKeyName As Any, ByVal lpString As Any, ByVal lpFileName As String) As Long

Private Sub btnCancel_Click()

'A:show main window

Status.Show

'A:unload GSM_Modem_Setting window

Unload GSM_Modem_Setting

End Sub

Private Sub btnSave_Click()

Dim strPath As String

'A: Get the path of this application

strPath = App.Path

'A: Write GSM_Modem_Setting in .ini file

WriteINI "GSM_Modem_Setting", "Port", Me.txtPort.Text, strPath & "\Setting.INI"

WriteINI "GSM_Modem_Setting", "PIN", Me.txtPIN.Text, strPath & "\Setting.INI"

MsgBox ("Seeting have been save")

'A: show main window

Status.Show

'A: unload GSM_Modem_Setting window

Unload GSM_Modem_Setting

End Sub

Private Sub Form_Load()

Dim strPath As String

'A: Get the path of this application

strPath = App.Path

Status.Hide

txtPort.Text = ReadINI("GSM_Modem_Setting", "Port", strPath & "\Setting.INI")

txtPIN.Text = ReadINI("GSM_Modem_Setting", "PIN", strPath & "\Setting.INI")

End Sub

Public Function ReadINI(strsection As String, strkey As String, strfullpath As String) As String

.....

'read ini

'x = readini("Example INI", "Example", "C:\Example.ini")

'MsgBox X

.....

Dim strbuffer As String

```
Let strbuffer$ = String$(750, Chr$(0&))
```

```
Let ReadINI$ = Left$(strbuffer$, GetPrivateProfileString(strsection$, ByVal LCase$(strkey$), "", strbuffer,  
Len(strbuffer), strfullpath$))
```

```
End Function
```

```
Public Sub WriteINI(strsection As String, strkey As String, strkeyvalue As String, strfullpath As String)
```

```
.....
```

```
'write ini
```

```
'Call writeini("Example INI", "Example", "Yes", "C:\Example.ini")
```

```
.....
```

```
Call WritePrivateProfileString(strsection$, UCase$(strkey$), strkeyvalue$, strfullpath$)
```

```
End Sub
```

2.3 EMAIL SETTING CODE

```
Private Declare Function GetPrivateProfileString Lib "kernel32" Alias "GetPrivateProfileStringA" (ByVal  
lpApplicationName As String, ByVal lpKeyName As String, ByVal lpDefault As String, ByVal lpReturnedString  
As String, ByVal nSize As Long, ByVal lpFileName As String) As Long
```

```
Private Declare Function WritePrivateProfileString Lib "kernel32" Alias "WritePrivateProfileStringA" (ByVal  
lpApplicationName As String, ByVal lpKeyName As Any, ByVal lpString As Any, ByVal lpFileName As String)  
As Long
```

```
Public bln160 As Boolean
```

```
Private Sub btnCancel_Click()
```

```
'A:hide main program
```

```
Status.Show
```

```
Unload EMail_Setting
```

```
End Sub
```

```
Private Sub btnSave_Click()
```

```
Dim strPath As String
```

```
'A:Get the path of this application
```

```
strPath = App.Path
```

```
'A:write/save data in .ini file
```

```
WriteINI "EMail_Setting", "Email address", Me.txtEmail.Text, strPath & "\Setting.INI"
```

```
WriteINI "EMail_Setting", "ID", Me.txtID.Text, strPath & "\Setting.INI"
```

```
WriteINI "EMail_Setting", "Password", Me.txtPassword.Text, strPath & "\Setting.INI"
```

```
WriteINI "EMail_Setting", "Receiver", Me.txtReceiver.Text, strPath & "\Setting.INI"
```

```
'A: check if checkbox are true or false then save it in the .ini file
```

```
If Me.chkFrom.Value = 1 Then
```

```
WriteINI "EMail_Setting", "Sender", "1", strPath & "\Setting.INI"
```

```
Else
```

```
WriteINI "EMail_Setting", "Sender", "0", strPath & "\Setting.INI"
```

```
End If
```

```
If Me.chkDate.Value = 1 Then
```

```
WriteINI "EMail_Setting", "Date", "1", strPath & "\Setting.INI"
```

```
Else
```

```
WriteINI "EMail_Setting", "Date", "0", strPath & "\Setting.INI"  
End If
```

```
If Me.chkSubject.Value = 1 Then
```

```
WriteINI "EMail_Setting", "Subject", "1", strPath & "\Setting.INI"
```

```
Else
```

```
WriteINI "EMail_Setting", "Subject", "0", strPath & "\Setting.INI"
```

```
End If
```

```
If Me.chkBody.Value = 1 Then
```

```
WriteINI "EMail_Setting", "Body", "1", strPath & "\Setting.INI"
```

```
Else
```

```
WriteINI "EMail_Setting", "Body", "0", strPath & "\Setting.INI"
```

```
End If
```

```
MsgBox ("Seeting have been save")
```

```
Status.Show
```

```
Unload EMail_Setting
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
Dim strPath As String
```

```
Dim strValue As String
```

```
Status.Hide
```

```
'A:retrieve path of application
```

```
strPath = App.Path
```

```
txtEmail.Text = ReadINI("EMail_Setting", "Email address", strPath & "\Setting.INI")
```

```
txtID.Text = ReadINI("EMail_Setting", "ID", strPath & "\Setting.INI")
```

```
txtPassword.Text = ReadINI("EMail_Setting", "Password", strPath & "\Setting.INI")
```

```
txtReceiver.Text = ReadINI("EMail_Setting", "Receiver", strPath & "\Setting.INI")
```

```
'A:read value True/False from .ini file
```

```
strValue = ReadINI("EMail_Setting", "Sender", strPath & "\Setting.INI")
```

```
If strValue = "1" Then
    Me.chkFrom.Value = 1
Else
    Me.chkFrom.Value = 0
End If
```

```
strValue = ReadINI("EMail_Setting", "Date", strPath & "\Setting.INI")
If strValue = "1" Then
    Me.chkDate.Value = 1
Else
    Me.chkDate.Value = 0
End If
```

```
strValue = ReadINI("EMail_Setting", "Subject", strPath & "\Setting.INI")
If strValue = "1" Then
    Me.chkSubject.Value = 1
Else
    Me.chkSubject.Value = 0
End If
```

```
strValue = ReadINI("EMail_Setting", "Body", strPath & "\Setting.INI")
If strValue = "1" Then
    Me.chkBody.Value = 1
Else
    Me.chkBody.Value = 0
End If
```

End Sub

Public Function ReadINI(strsection As String, strkey As String, strfullpath As String) As String

.....

```
'read ini
'x = readini("Example INI", "Example", "C:\Example.ini")
'MsgBox X
```

.....

Dim strbuffer As String

Let strbuffer\$ = String\$(750, Chr\$(0&))

Let ReadINI\$ = Left\$(strbuffer\$, GetPrivateProfileString(strsection\$, ByVal LCase\$(strkey\$), "", strbuffer, Len(strbuffer), strfullpath\$))

End Function

Public Sub WriteINI(strsection As String, strkey As String, strkeyvalue As String, strfullpath As String)

.....

'write ini

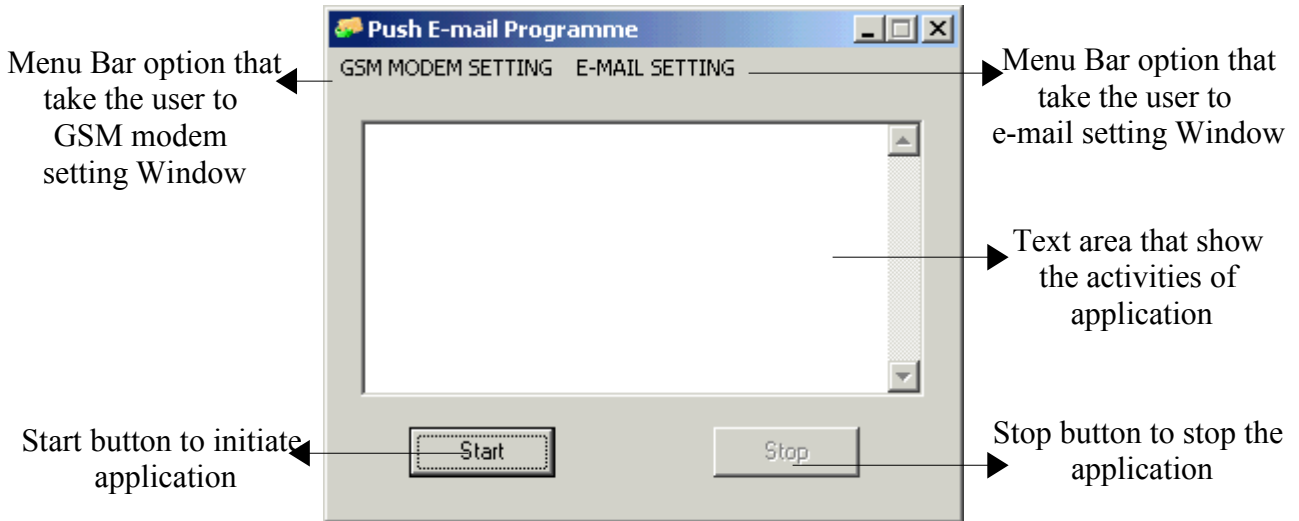
'Call writeini("Example INI", "Example", "Yes", "C:\Example.ini")

.....

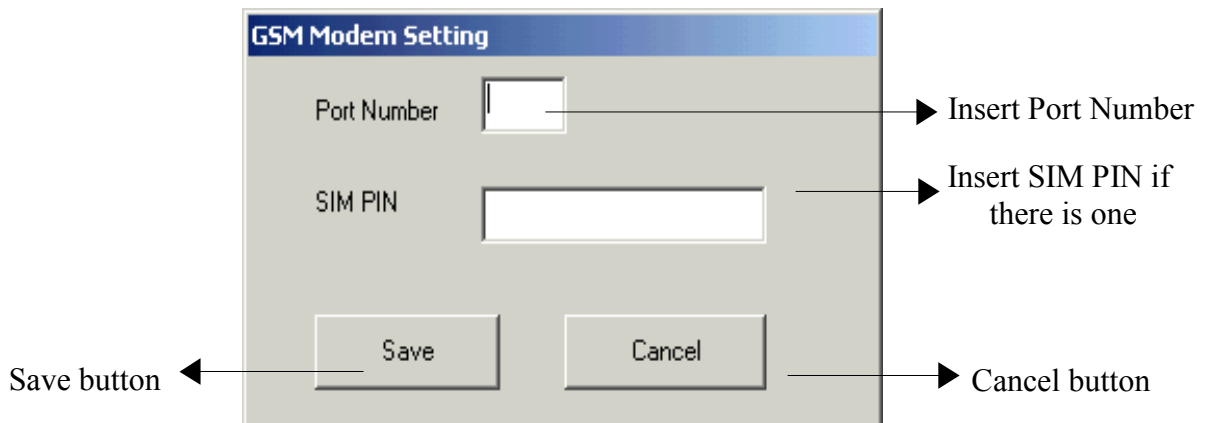
Call WritePrivateProfileString(strsection\$, UCase\$(strkey\$), strkeyvalue\$, strfullpath\$)

End Sub

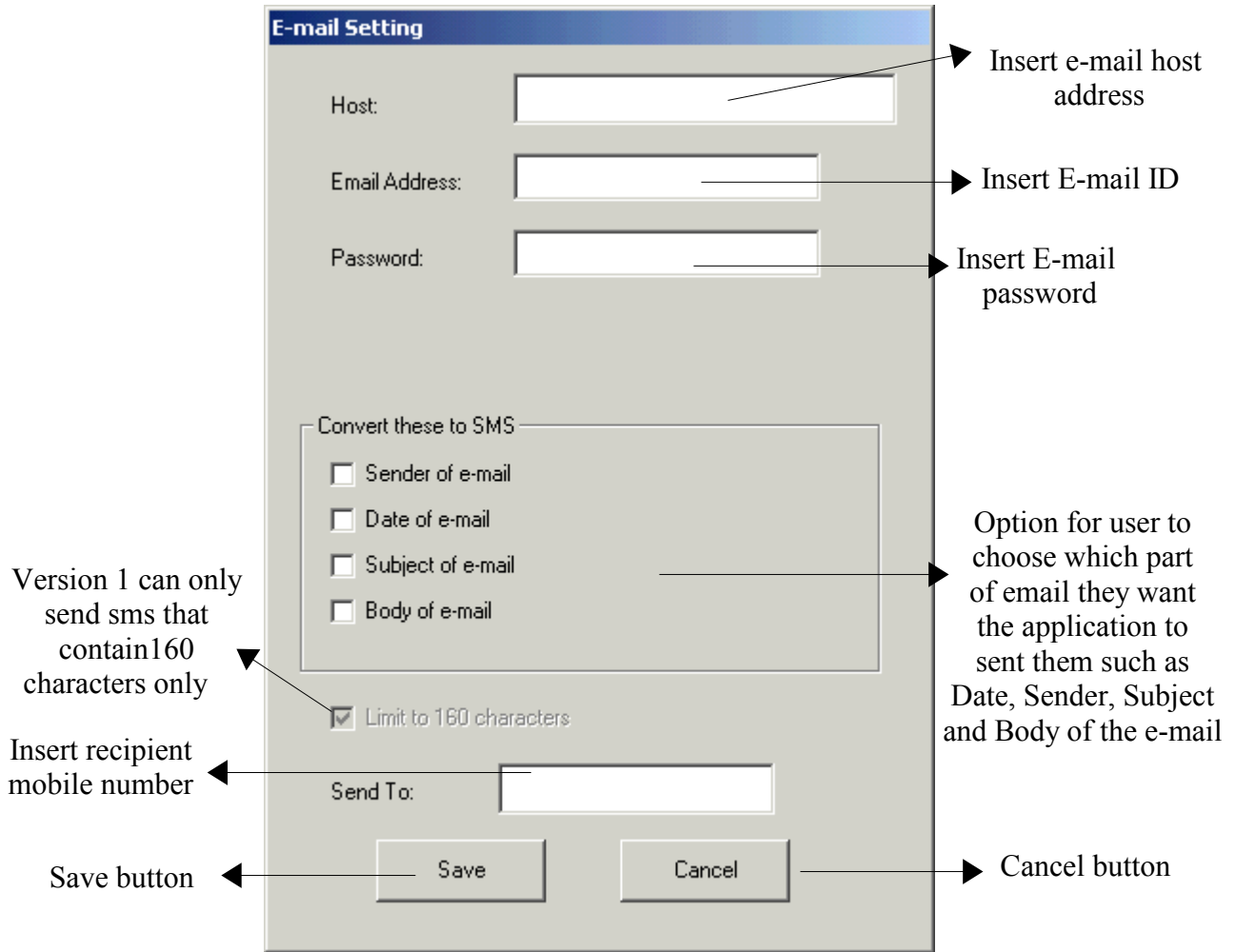
SECTION 3: GRAPICAL USER INTERFACE



Main window



GSM modem setting window



E-mail Settings window

SECTION 4: WARRANTY AND SUPPORT

MOBITEK System **does not provide any warranty** for the use of **Push e-Mail (FreePOP Edition)**.

MOBITEK System **does not provide any technical assistance nor support** for the use of **Push e-Mail (FreePOP Edition)**.