SAE – Automation - **S**olid **A**nd **E**ffectively working partner for development of **YOUR** products and industrial automation

XlsSmsSender

The client application example for SAEAUT SMS Service for sending SMS implemented in VBA within xls-sheet

Main goal by SAEAUT SMS Service design was to create a server application aplication **easily usable by different client applications in LAN or Internet.** There is a few possibilities to fulfil this goal – for example, the dynamic linked library. Such solution is offered by SAE- Automation as well.

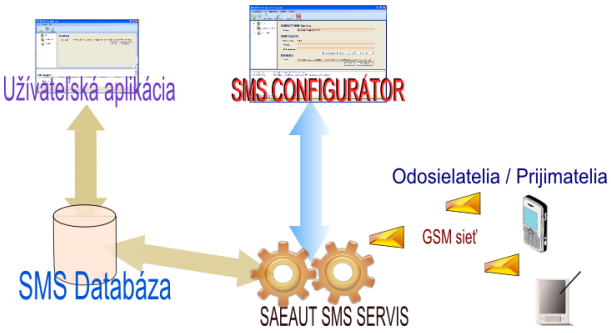


Figure. 1. Cooperation of a user client application with SAEAUT SMS Service using shared database

We have tried to create even easier solution (Figure 1.) – to enable a cooperation of arbitrary application, written in arbitrary programming language or script, over shared database.

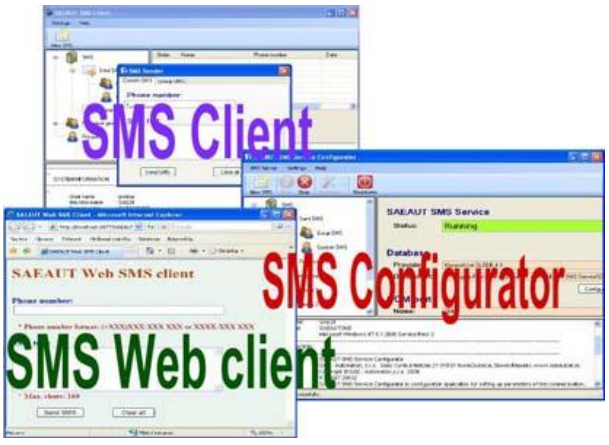


Figure 2. Client applications for SAEAUT SMS Service

We offer (1) own client desktop application enabling a SAEAUT SMS Service usage from arbitrary computer in LAN, (2) web application enabling its using in Internet (3)client application enabling interconnecting almost arbitrary database with the SMS Service database (4) our application **OpcDbGateway** which can use the application database as own process database (5) the configuration application of the SAEAUT SMS Service itself is also client application for sending/receiving individual or group SMS.

We suppose that SAEAUT SMS Service is useful also for skilled Office applications (as e.g. MS Excel™) users, who would like to implement sending/receiving SMS functionality according own needs.

For these users category we have prepared an example of client application implemented using VBA (Visual Basic for Applications). The file containing this application can be downloaded from na www.saeautom.sk/download/XlsSMSList.xls . Experimenting with look and functionality of this application is easy also for none programming professionals.

After opening of this file, it is necessary to enable using of macros and to start the macro *TimeScaning* according to the Figure.3.

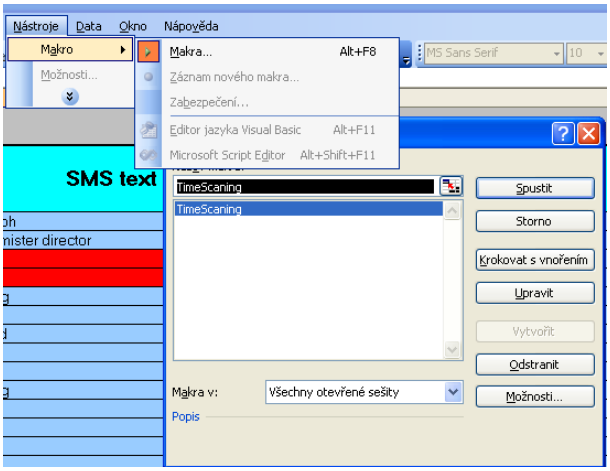


Figure 3. Starting of the TimeScaning macro

Then, you will be asked to define a path to the SAEAUT SMS SERVICE applications database, which (if you have not changed its placement after installation) will be in the directory *c:\Program Files\SAE - Automation, s.r.o\SAEAUT SMS Service\Database* .

Now, you can already start writing of SMS texts to the column *SMS Text* (Figure 4.) and telephone numbers of the SMS receivers to the column *Tel.*

http://www.saeautom.sk, sae-automation@saeautom.sk, tel.:+421-(0)42-445 07 01, fax:+421-(0)42-445 07 02,
Adresa: Sady Cyrila a Metoda, 21/18, 018 51 Nová Dubnica, Slovakia

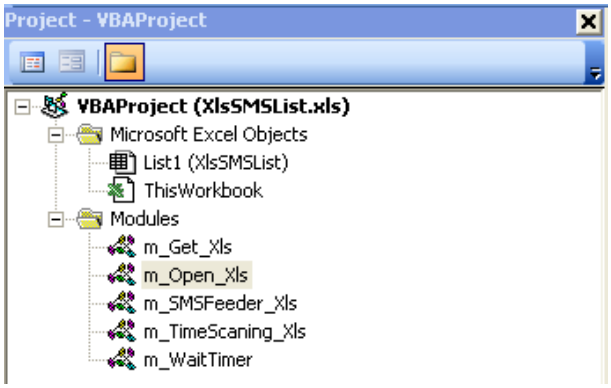
number. After that, the programme is waiting for confirmation sign „*“ in the column *Confirm Sign*. Programme evaluates if given input data are correct, and eventual errors displays in the column *Result*.

Tel. number	SMS text	Confirm. Sign	Result
+421915749096	Video o SAEAUT SMS service: www.saeautom.sk/sk/products/smsservice/download.htm	*	ok
+421915749096	Cena: 410,-EUR	*	ok
0421	Nevrhovsky termin stretnutia: zajtra, 18:00	*	B6 Tel.is too short! C6 is empty!
0421915		*	B7 Tel.is too short! C7 is empty!

Figure 4.XLS-table used for SMS sending

If you are requiring to resend an already sent SMS again, you can remove the confirmation sign „*“ and to put it to the *Confirm Sign* column again. The application enables also transfer of the whole list with SMS from various text files using Windows™ clipboard.

We intended first of all to offer inspiration for creation of alike applications. Because of this, please look at the application description and the VBA source code and adapt it according to your needs. The VBA modules overview can be found in the Figure 5.



Obr. 5. Program modules overview

Main application cycle placed in the module ***m_TimeScaning_Xls*** initiates periodic pooling of new SMS items in the table according to the Figure 4.

m_Open_Xls jis used for opening of the Excel form before of the main application cycle.

m_Get_Xls module uses OLE Automation functionality to transfer data between MS Excel and MS Access environment. It provides identification of new SMS, proving their correctness and transforming of new SMS to the 2-dimesional array given as return value from the *Get_Xls* function value.

Module ***m_SMSFeder_Xls*** provides correct formatting and transfer of the 2-dimenional array containing new SMS to the database table *SimpleSMSList.*

Module ***Wait_Timer*** contains function starting of the commands in defined time.

Complete program listing can be found in next pages.

http://www.saeautom.sk, sae-automation@saeautom.sk, tel.:+421-(0)42-445 07 01, fax:+421-(0)42-445 07 02,
Adresa: Sady Cyrila a Metoda, 21/18, 018 51 Nová Dubnica, Slovakia

```
Public strXls As String
Public oldRec As Long
Public allRec As Long
Public newRec As Long

Public objSht As Excel.Worksheet
Public MDBpath As String

Sub TimeScaning()

Dim telProd As String
Dim outSms As String
Dim RetRec As Variant
Dim IsChosen As Boolean

    IsChosen = IsChosenFile()
    If Not IsChosen Then
        MsgBox "Mdb file was not found" _
        & Chr(13) & Chr(10) & "Macro will be stopped"
        Exit Sub
    End If

strXls = "Provider='Microsoft.Jet.OLEDB.4.0';" _
& "Data Source='" & MDBpath & " ';"

allRec = 100
oldRec = 1
newRec = 0

OpenXls

'Main cycle repeating by period of every 15 second
Do

RetRec = GetXls
If newRec > 0 Then
   FeedXlsSMS RetRec
End If
wait 15

Loop

End Sub

Function IsChosenFile() As Boolean


MDBpath = Application.GetOpenFilename("MS access databse (*.mdb),*.mdb")

If MDBpath <> "False" Then
   IsChosenFile = True
Else
   IsChosenFile = False
End If

End Function
```

Tab 1: Modul m_TimeScaning_Xls

http://www.saeautom.sk, sae-automation@saeautom.sk, tel.:+421-(0)42-445 07 01, fax:+421-(0)42-445 07 02,
Adresa: Sady Cyrila a Metoda, 21/18, 018 51 Nová Dubnica, Slovakia

```
Public Function OpenXls()
On Error GoTo ErrorHandler


Set objSht = Excel.ActiveWorkbook.ActiveSheet
Exit Function

ErrorHandler:


    If Err <> 0 Then
        MsgBox Err.Source & "-->" & Err.Description, , "Error"
    End If
End Function
```

Tab 2: Modul m_Open_Xls

```
Public Function GetXls() As Variant
 On Error GoTo ErrorHandler

'Create an array with 3 columns and 100 rows

Dim TmpArr(1 To 100, 1 To 4) As Variant
Dim DatArr(1 To 100, 1 To 4) As Variant

Dim r, i, StartPointX, StartPointY As Integer
Dim buff As String
Dim col1, col2, col3, col4 As Boolean
Dim ActiveRow As Long


ActiveRow = RowNumber

i = 0

'address of start point for the xls data table

StartPointX = 1
StartPointY = 3


For r = 1 To ActiveRow
  col1 = True
  col2 = True
  col3 = True
  col4 = True
  buff = ""
  TmpArr(r, 1) = objSht.Cells(r + StartPointY, 1 + StartPointX).Value
  TmpArr(r, 2) = objSht.Cells(r + StartPointY, 2 + StartPointX).Value
  TmpArr(r, 3) = objSht.Cells(r + StartPointY, 3 + StartPointX).Value
  TmpArr(r, 4) = objSht.Cells(r + StartPointY, 4 + StartPointX).Value


 'End of table
  If TmpArr(r, 1) = "" And TmpArr(r, 2) = "" And TmpArr(r, 3) = "" Then
     Exit For
  End If

      'Is format right?
```


http://www.saeautom.sk, sae-automation@saeautom.sk, tel.:+421-(0)42-445 07 01, fax:+421-(0)42-445 07 02,
Adresa: Sady Cyrila a Metoda, 21/18, 018 51 Nová Dubnica, Slovakia

```
'col1*************************************************
    If TmpArr(r, 1) = "" Then
      buff = buff & "B" & r + StartPointY & " Is empty! "
      col1 = False
    End If
    If Val(TmpArr(r, 1)) = 0 Then
      buff = buff & "B" & r + StartPointY & " Is not digit! "
      col1 = False
    End If
    If Len(TmpArr(r, 1)) <= 9 Then
      buff = buff & "B" & r + StartPointY & " Tel.is too short! "
      col1 = False
    End If
    If Len(TmpArr(r, 1)) >= 14 Then
      buff = buff & "B" & r + StartPointY & " Tel.is too long! "
      col1 = False
    End If


    'Finsal evalution for col1
    If Not col1 Then
      objSht.Cells(r + StartPointY, 1 + StartPointX).Interior.ColorIndex = 3
    Else
      objSht.Cells(r + StartPointY, 1 + StartPointX).Interior.ColorIndex = 37
    End If

'col2*************************************************
    If TmpArr(r, 2) = "" Then
      buff = buff & "C" & r + StartPointY & " Is empty! "
      col2 = False
    End If

         'Finsal evalution for col2
    If Not col2 Then
      objSht.Cells(r + StartPointY, 2 + StartPointX).Interior.ColorIndex = 3
    Else
      objSht.Cells(r + StartPointY, 2 + StartPointX).Interior.ColorIndex = 37
    End If


'col3*************************************************
    If TmpArr(r, 3) <> "*" Then
      buff = buff & "D" & r + StartPointY & " Confirm sign is not right! "
      col3 = False
    End If

         'Finsal evalution for col3
    If Not col3 Then
      objSht.Cells(r + StartPointY, 3 + StartPointX).Interior.ColorIndex = 3
    Else
      objSht.Cells(r + StartPointY, 3 + StartPointX).Interior.ColorIndex = 37
    End If


'sum col1 col2 col3***********************************

    If col1 And col2 And col3 Then
      objSht.Cells(r + StartPointY, 4 + StartPointX).Value = "ok"
      objSht.Cells(r + StartPointY, 4 + StartPointX).Font.ColorIndex = 10
    Else
      objSht.Cells(r + StartPointY, 4 + StartPointX).Value = buff
```


http://www.saeautom.sk, sae-automation@saeautom.sk, tel.:+421-(0)42-445 07 01, fax:+421-(0)42-445 07 02,
Adresa: Sady Cyrila a Metoda, 21/18, 018 51 Nová Dubnica, Slovakia

```
      objSht.Cells(r + StartPointY, 4 + StartPointX).Font.ColorIndex = 9
    End If

'*******************************************************

   'Inset SMS record to the final output table
    If col1 And col2 And col3 And TmpArr(r, 4) <> "ok" Then

      i = i + 1

      DatArr(i, 1) = TmpArr(r, 1)
      DatArr(i, 2) = TmpArr(r, 2)
      DatArr(i, 3) = TmpArr(r, 3)
      DatArr(i, 4) = TmpArr(r, 4)

    End If

 Next

 GetXls = DatArr
 newRec = i

Exit Function

ErrorHandler:

  If Err <> 0 Then
      MsgBox Err.Source & "-->" & Err.Description, , "Error"
  End If
End Function

Function RowNumber() As Long 'Compute the number of row
    RowNumber = ActiveSheet.Cells.Find(What:="*", _
    SearchDirection:=xlPrevious, _
    SearchOrder:=xlByRows).Row
End Function
```

Tab 3: Modul m_Get_Xls

```
'Declaring of function important for the geting of Computer name

Private Declare Function GetComputerNameXls Lib "kernel32" _
    Alias "GetComputerNameA" _
    (ByVal lpBuffer As String, nSize As Long) As Long

'Procedure for exact formating and inserting a new SMS to the SimpleSMSList
table
'SimpleSMSList table servs as a outputing gate for the outgoing SMS
Sub FeedXlsSMS(ByVal xlsArr As Variant)

    On Error GoTo ErrorHandler
    'recordset and connection variables
    Dim rstFeedSms As ADODB.Recordset
    Dim Cnxn As ADODB.Connection
    Dim strSQLFeedSms As String

    'data and time format variables
    Dim time As Date
    Dim datTim As Date

    datTim = DateValue(Now) & " " & TimeValue(Now)
```


http://www.saeautom.sk, sae-automation@saeautom.sk, tel.:+421-(0)42-445 07 01, fax:+421-(0)42-445 07 02,
Adresa: Sady Cyrila a Metoda, 21/18, 018 51 Nová Dubnica, Slovakia

```
    'open connection
    Set Cnxn = New ADODB.Connection
    Cnxn.Open strXls

    'open recordset server-side for indexing
    Set rstFeedSms = New ADODB.Recordset
    rstFeedSms.CursorLocation = adUseServer
    strSQLFeedSms = "SimpleSMSList"

    rstFeedSms.Open strSQLFeedSms, strXls, adOpenKeyset, _
    adLockOptimistic, adCmdTableDirect



    'inserting of the formated strings
    'to the specific fields of the table SimpleSMSList

    For r = 1 To newRec
        rstFeedSms.AddNew
        rstFeedSms!usComputerName = ReturnComputerNameXls          '"SAE18"
        rstFeedSms!usSenderPhone = formatTel(xlsArr(r, 1))       'telSum
        rstFeedSms!usSMSText = xlsArr(r, 2)          'smsText
        rstFeedSms!usTimeStamp = datTim
        rstFeedSms.Update
        blnRecordAdded = True
    Next r



    'clean up
    rstFeedSms.Close
    Cnxn.Close
    Set rstFeedSms = Nothing
    Set Cnxn = Nothing
    Exit Sub

ErrorHandler:
    ' clean up
    If Not rstFeedSms Is Nothing Then
        If rstFeedSms.State = adStateOpen Then rstFeedSms.Close
    End If
    Set rstFeedSms = Nothing

    If Not Cnxn Is Nothing Then
        If Cnxn.State = adStateOpen Then Cnxn.Close
    End If
    Set Cnxn = Nothing

    If Err <> 0 Then
        MsgBox Err.Source & "-->" & Err.Description, , "Error"
    End If
End Sub

'Functionality important for the geting of Computer name

Function ReturnComputerNameXls() As String
Dim rString As String * 255, sLen As Long, tString As String
    tString = ""
    On Error Resume Next
    sLen = GetComputerNameXls(rString, 255)
```


http://www.saeautom.sk, sae-automation@saeautom.sk, tel.:+421-(0)42-445 07 01, fax:+421-(0)42-445 07 02,
Adresa: Sady Cyrila a Metoda, 21/18, 018 51 Nová Dubnica, Slovakia

```
    sLen = InStr(1, rString, Chr(0))
    If sLen > 0 Then
        tString = Left(rString, sLen - 1)
    Else
        tString = rString
    End If
    On Error GoTo 0
    ReturnComputerNameXls = UCase(Trim(tString))
End Function

Function formatTel(tel)
    'preformating of inputing tel. +444222111111 number
    'to the standard world format  +444(222)111111

    lenTelPreselection = Len(tel) - 9

    tel1 = Left(tel, lenTelPreselection)
    tel2 = Left(Right(tel, 9), 3)
    tel3 = Right(tel, 6)

    formatTel = tel1 & "(" & tel2 & ")" & tel3
End Function
```

Tab 4: Modul m_SMSFeeder_Xls

http://www.saeautom.sk, sae-automation@saeautom.sk, tel.:+421-(0)42-445 07 01, fax:+421-(0)42-445 07 02,
Adresa: Sady Cyrila a Metoda, 21/18, 018 51 Nová Dubnica, Slovakia