



Software Developer Guide

SignMeIn

Copyright © Topaz Systems Inc. All rights reserved.

For Topaz Systems, Inc. trademarks and patents, visit www.topazsystems.com/legal.

Table of Contents

Overview.....	3
Properties.....	3
Methods	5
SignMeIn.INI File	7
Installation Notes.....	8
SignMeIn DLL and OCX List.....	9

Overview

NOTE: *SignMeIn™ Active-X Front Desk Sign-in System - MUST BE USED WITH SIGPLUS SOFTWARE SYSTEM. For use with Topaz SigPlus Active-X control version 4.29 or newer.*

INSTALLATION NOTE: *The control should be properly setup and registered when running the setup program. To register the control manually, copy signmein.ocx into the c:\windows\sigplus directory. Then run regsvr32 c:\windows\sigplus\signmein.ocx. Or, copy signmein.ocx into the c:\winnt\sigplus directory. Then run regsvr32 c:\winnt\sigplus\signmein.ocx*

SignMeIn.ocx is an Active-X control for creating easily-integrated, front desk applications using signature capture. Signatures, along with accompanying data (Visitee's name, appointment time/date, etc.) are stored in an Access database. Signature records can then be viewed using the SignMeInReport.ocx control. (See SignMeInReport.doc or SignMeInReport.pdf)

SignMeIn.ocx allows for secure, private sign-in applications. SignMeIn.ocx must be used in conjunction with the Topaz SigPlus Active-X signature capture control for capturing signatures.

FOR A VISUAL BASIC 6 DEMO USING SIGNMEIN.OCX, PLEASE SEE YOUR WIN\SigPlus\SignMeIn\VB6Demos\ DIRECTORY AFTER INSTALLATION.

Properties

.AutoSave =

Boolean. Set to TRUE, AutoSave allows the user the option to automatically archive the current database monthly. The first time the application is run after a new month has begun, SignMeIn.ocx will automatically ask the user if he/she wishes to archive the current database and begin a new one. User may choose to allow or deny. The user also has the opportunity to change the file name before archiving. The default name is a combination of the previous month, the current month and the year. Set to FALSE, it is up to the user to archive and begin a new database at their discretion.
THIS FEATURE CAN ONLY BE USED WITH THE DEFAULT DATABASE SCENARIO/LOCATION.

Example: SignMeIn1.AutoSave = TRUE

.DatabaseLoc =

String. DatabaseLoc allows the user to pass a path to a custom database location for the MDB database used to store the data. Call with double quotes (empty string) to set SignMeIn to use the default built-in database. (Although location of the database can be customized, the SignMeIn-provided MDB file must still be used!)

Example:

```
SignMeIn1.DatabaseLoc = "" 'set database to default location/use  
SignMeIn1.DatabaseLoc = "c:\somelocation\somefile.mdb" 'set custom db location
```

.ImageLoc =

String. ImageLoc allows the user to pass a path to a custom image to display on the “Welcome” screen during the sign-in process. Image MUST be black and white 1-bit BMP image of size no larger than 51 px sq. If no image is desired, pass an empty string ("").

Example:

```
SignMeIn1.ImageLoc = "" 'set image to none  
SignMeIn1.ImageLoc = "c:\somelocation\someimage.bmp" 'set custom image
```

.Visitee1, .Visitee 2, .Visitee 3, .Visitee 4, .Visitee5, . Visitee6, . Visitee7, . Visitee8 =

String. Eight separate properties, these allow the user to programmatically override the visitees assigned in the SignMeIn.ini. Additionally, the user can return the visitees currently assigned in the SignMeIn.ini (Visitee's names have a 12 character limit)

Example (overriding visitee from SignMeIn.ini): SignMeIn1.Visitee1 = “Visitee’s Name”

Example (returning visitee name from SignMeIn.ini):

```
Dim strVis As String  
strVisitee = SignMeIn1.Visitee1 'strVis now holds the Visitee1 value from the  
SignMeIn.ini
```

.Title1, .Title2, .Title3, .Title4, .Title5 =

String. Five separate properties, these allow the user to programmatically override the titles assigned in the SignMeIn.ini file. These titles are ordered as they appear on the screens when the SignMeIn application is run. Additionally, the user can return the titles currently assigned in the SignMeIn.ini

Example 1(overriding the first title from SignMeIn.ini):

```
SignMeIn1.Title1 = "Welcome to Some Hospital"
```

Example 2(returning the first title from from SignMeIn.ini):

```
Dim strTitle As String  
strTitle = SignMeIn1.Title1 'strTitle now holds Title1 from the SignMeIn.ini
```

.Question1, .Question2 =

String. Two separate properties, these allow the user to programmatically override the yes/no questions assigned in the SignMeIn.ini file. These questions appear on the fourth screen when the SignMeIn application is run. Additionally, the user can return the questions currently assigned in the SignMeIn.ini

Example (overriding the first question from SignMeln.ini):

```
SignMeln1.Question1 = "Has Your Telephone Number Changed?"
```

Example (returning the first title from from SignMeln.ini):

```
Dim strQuestion As String strQuestion = SignMeln1.Question1  
'strQuestion now holds first question from the SignMeln.ini
```

Methods

. OpenDb() As Integer

This method will open the SignMeln database. Prior to calling OpenDb(), the DatabaseLoc property should be set (see DatabaseLoc property). OpenDb() will open either the default, built-in SignMeln database, or the path to the database can be passed in for more customized location storage.

RETURN: Integer. 0=open successful, 1=database cannot be located, 2=database table "SignMeln" cannot be located, or incorrect MDB file has been selected

Example 1 (opening default database):

```
SignMeln1.DatabaseLoc = "" 'set up SignMeln to use the default, built-in database  
SignMeln1.OpenDb()
```

Example 2 (opening default database):

```
'set up SignMeln to use a database in a customized location SignMeln1.DatabaseLoc =  
"c:\somefolder\somefile.mdb" SignMeln1.OpenDb()  
Dim intRet As Integer
```

```
intRet = SignMeln1.OpenDB If intRet = 0 Then  
  
    'continue, database opened successfully  
ElseIf intRet = 1 Then  
    'cannot locate the default SignMeln database  
ElseIf intRet = 2 Then '  
    'database does not contain SignMeln table  
End If
```

.RunSignMeln()

This method begins the SignMeln application loop. Must be called after OpenDb() method.

Example:

```
SignMeln1.RunSignMeln
```

.ExitSignMeIn()

This method ends the SignMeIn application loop, and resets the tablet.

Example:

```
SignMeIn1.ExitSignMeIn
```

.ArchiveCurrentDatabase(strFilePath)

This method allows the user to manually archive the current database. The method takes a String argument, the file name to save with. The database will automatically be saved in the Archive directory of the SignMeIn directory structure, and a new database will be set up as the default.

Example:

```
Dim intAns As Integer
```

```
intAns = SignMeIn1.ArchiveCurrentDatabase("somefilename.mdb")
```

```
If intAns = 0 Then
    Successful archive
ElseIf intAns = 1 Then
    'database is open—cannot archive
ElseIf intAns = 2 Then
    'error occurred during archiving—cannot archive
End If
```

.GoFirstRecord()

This method returns the first record in the database. Must be called after OpenDb().

Example:

```
SignMeIn1.GoFirstRecord
```

.GoPreviousRecord()

This method returns the previous record in the database from the current. Must be called after OpenDb().

Example:

```
SignMeIn1.GoPreviousRecord
```

.GoNextRecord()

This method returns the next record in the database from the current. Must be called after OpenDb().

Example:

```
SignMeIn1.GoNextRecord
```

.GoLastRecord()

This method returns the last record in the database. Must be called after OpenDb().

Example:

```
SignMeIn1.GoLastRecord
```

.DeleteRecord()

This method deletes the current record in the database. Must be called after OpenDb().

Example:

```
SignMeIn1.DeleteRecord
```

SignMeIn.INI File

The SignMeIn.ini file allows the user to set various properties normally set by the developer at runtime. The SignMeIn.ini is automatically saved to the target machine's WIN directory (Windows or WINNT) during installation, and includes the following:

```
VisiteeName1=somevisname
VisiteeName2=somevisname
VisiteeName3=somevisname
VisiteeName4=somevisname
VisiteeName5=somevisname
VisiteeName6=somevisname
VisiteeName7=
VisiteeName8=
Title=Welcome To Named Hospital
TitlePg2=Please Choose Your Visitee:
TitlePg3>Select Appointment Time:
TitlePg4=Please Answer the Following:
TitlePg5=Sign or Print Your Name:
Question1=Has your address changed?
Question2=Has your insurance changed?
DatabaseLoc=
ImageLoc=
AutoSave=True
```

The user can specify in the INI up to 8 visitee's names to appear during the SignMeIn application loop. If any of the visitee's names are left blank (as in 7 and 8 above), then only those with values will be part of the application. (Limit 12 characters per visitee name)

There are a total of 5 screens in use during the SignMeIn application loop. Each screens title can be controlled via the Title parameters. (Limit to titles vary)

On screen 4 of the SignMeIn application loop, there are two questions posed to the user. These questions were designed to query the user about possible changes in their insurance and address, but can be changed using the Question parameters above. (Limit ?? characters)

The default database used by SignMeIn is located after install at WIN\SigPlus\SignMeIn\Current\SignMeIn.mdb. This is the database used if the DatabaseLoc parameter is left with no value (as shown above). If the user enters a pat to a different database, SignMeIn.ocx will use that database instead.

To include a small BMP image of choice on the “Welcome” screen (#1) of the SignMeIn application loop, the user can type in the path and filename using the ImageLoc parameter. (For details on type and size, please see the SignMeInLogo.bmp file located in your WIN\SigPlus\SignMEIn\Images\ directory after installation.

AutoSave, when set to True, will ask the user (the first time SignMeIn.ocx is instantiated after the first of a new month—based on the system clock) if they wish to archive the current database. The user can accept or deny at that time. If declined, the database is not archived. If accepted, the database is archived using the previous month/year and current month/year as the title. Alternate naming is optional. Databases archived in this manner will automatically be stored in the WIN\SigPlus\SignMeIn\Archived\ directory. A new database will automatically replace the archived database in the WIN\SigPlus\SignMeIn\Current\ directory.

Installation Notes

Topaz Systems SignMeIn post-installation directory structure is:

WIN

\SigPlus

\SignMeIn

SignMeIn.ocx (the Topaz Front Desk Sign-in Active-X Control)

SignMeInReport.ocx (the report creating Active-X Control for use with SignMeIn)

SignMeIn.exe (The SignMeIn application & demo using SignMeIn.ocx)

SignMeInReport.exe (Report application & demo using SignMeInReport.ocx)

MonthTracker.txt (keeps track of the current month for use with AutoSave feature)

\Archived (default archived database files are stored here)

\Clean (a clean copy of the SignMeIn.mdb database is stored here)

\Current (the current default SignMeIn.mdb database is stored here)

\Images (default images for the initial SignMeIn app screen stored here)

Files related to the SignMeInOCX and SignMeInReportOCX control:

SignMeIn.ocx	The SignMeIn Front Desk Sign-in Active-X Control.
SignMeIn.exe	Out-of-the-box front desk sign-in application using SignMeIn.ocx.
SignMeInReport.ocx	The SignMeIn Report Generator Active-X control.
SignMeInReport.exe	The out-of-the-box SignMeIn report generator demo and system.
SignMeIn.doc	The complete documentation on use of the SignMeIn Active-X control. Includes properties and methods documentation.
SignMeInReport.doc	The complete documentation on use of the SignMeInReport Active-X control. Includes properties and methods documentation.
SigPlusAdjust.exe	The GUI utility for quickly changing tablet model and connection settings for SigPlus.ocx.
TabLoc.exe	Utility for locating Topaz LCD serial tablets.
SignMeIn_UsersGuide.doc	The complete documentation on the use of SignMeIn.exe and SignMeInReport.exe.
SignMeIn.zip	VB source code for SignMeIn.exe.
...\\Clean\\SignMeIn.mdb	The clean SignMeIn MDB file; can be used by developers as the SignMeIn database when custom apps are developed.

SignMeIn DLL and OCX List

MSVBVM60.DLL

SignMeIn.ocx

SignMeInReport.ocx

Scrrun.dll

Comdlg32.ocx

ADO 2.0 Type Library (msado20.tlb)

(In addition, SigPlus dependencies are necessary — see SigPlus documentation at

www.topazsystems.com/Software/sigplus.pdf