

Topaz Systems, Inc.
SignMeIn™ Front Desk Sign-in Active-X Control
Version 1.0.1 - 11/16/2011

Topaz Systems, Inc.
650 Cochran Street, Suite 6
Simi Valley, CA, 93065
805 520-8282 (Main)
805 520-0867 fax

Tech support:
805 520-8286
805 520-8282
www.topazsystems.com
support@topazsystems.com

Copyright 2011, all rights reserved

- 1. REQUIRES SIGPLUS ACTIVEX INSTALLATION**
- 2. REQUIRES MICROSOFT ACCESS XP, 2003, 2007, or 2010**
- 3. FOR USE WITH TL462 (SignatureGemLCD1X5) TABLET OR TL460 (SigLiteLCD) TABLETS ONLY**

SignMeIn™ Active-X Front Desk Sign-in System
(MUST BE USED WITH SIGPLUS SOFTWARE SYSTEM)

SignMeIn(tm) Front Desk Sign-in Active-X Control,
A member of the GemTools family.
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

DESCRIPTION:

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.

TECH SUPPORT

Call (805) 520-8286 or (805) 520-8282 for Technical support
Be sure to see <http://www.topazsystems.com/Software/faq.htm> for the most frequently asked questions and answers to software and hardware related issues.

**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 SignMeIn.ini):

```
SignMeIn1.Question1 = "Has Your Telephone Number Changed?"
```

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

```
Dim strQuestion As String
```

```
strQuestion = SignMeIn1.Question1
```

```
'strQuestion now holds first question from the SignMeIn.ini
```

Methods:

.OpenDb() As Integer

This method will open the SignMeIn database. Prior to calling OpenDb(), the DatabaseLoc property should be set (see DatabaseLoc property). OpenDb() will open either the default, built-in SignMeIn 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 "SignMeIn" cannot be located, or incorrect MDB file has been selected

Example 1 (opening default database):

```
SignMeIn1.DatabaseLoc = "" 'set up SignMeIn to use the default, built-in database  
SignMeIn1.OpenDb()
```

Example 2 (opening default database):

```
'set up SignMeIn to use a database in a customized location
```

```
SignMeIn1.DatabaseLoc = "c:\somefolder\somefile.mdb"
```

```
SignMeIn1.OpenDb()
```

```
Dim intRet As Integer
```

```
intRet = SignMeIn1.OpenDB
```

```
If intRet = 0 Then
```

```
'continue, database opened successfully
Elseif intRet = 1 Then
'cannot locate the default SignMeIn database
Elseif intRet = 2 Then '
'database does not contain SignMeIn table
End If
```

.RunSignMeIn()

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

Example:

```
SignMeIn1.RunSignMeIn
```

.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

Sccrun.dll

Comdlg32.ocx

ADO 2.0 Type Library (msado20.tlb)

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

<http://www.topazsystems.com/Software/sigplusdocs.zip>

Important Notices

License Agreement and Limited Warranty

IMPORTANT: Please read this document before continuing the software load procedure. By loading the software enclosed with this agreement, you are indicating acceptance of the terms of this legal agreement between you (herein call Licensee) and Topaz Systems, Inc. (herein called Topaz). If you do not agree to the terms of this agreement, do not load the enclosed software and promptly return the product.

1. **Limited use License:** Topaz and it's suppliers (if any) grant you the right to use the software for use with Topaz Gem-Series tablets only. The software is owned for distribution exclusively by Topaz and is protected by the United States Patent and Trademark laws and international treaties.

2. **Governing Law, Jurisdiction, and Forum:** This agreement is governed by the laws of the State of California, County of Ventura.

3. **YOU MAY:**

(a) Freely use, copy, and distribute the enclosed software only for use with Topaz Gem-Series tablets.

4. **YOU MAY NOT:**

(a) Use, distribute or modify the Topaz software for use with tablets not supplied by Topaz Systems, Inc.

(b) Reverse engineer, decompile, or dis-assemble, the software.

5. **LIMITED WARRANTY:**

Topaz Systems warrants to the original buyer only, that the media upon which the Software is recorded is free from defects in workmanship and material under normal use and service for a period of 30 days.

6. **EXCLUSIVE REMEDY:**

Topaz's entire liability and your exclusive remedy shall be, at Topaz's option, either (a) the repair or replacement of the software or (b) the refund of the price that was paid for the software package. The defective software, along with proof of payment, must be returned to Topaz within the terms of the warranty as set forth in this agreement.

7. **LIMITATIONS ON DAMAGES:**

In no event shall Topaz be liable for damages whatsoever, (including without limitation, damages for loss of profits, business interruption, loss of information, or other pecuniary loss) arising out of the use of or inability to use the software even it Topaz or it's suppliers, if any, have been advised have been advise of the possibility of such damages. In no event will Topaz's liability for any reason exceed the actual price paid for the license to use the specific program.

8. **NO OTHER WARRANTIES:**

With respect to the software, media, and written material, Topaz and it's suppliers, if any, disclaim all warranties, other than the above Topaz warranty, including but not limited to warranties merchantability or fitness for a particular use. Topaz does not warranty the software will meet or requirements or that the operations of the software will be uninterrupted or error free.

9. **GOVERNING LAW AND GENERAL PROVISIONS:**

If any part of this agreement is found void and unenforceable, it will not effect the validity of the balance of the agreement, which shall remain valid and enforceable according to it's terms. You agree that the software will not be shipped, transferred, or exported into any country or used in any manner prohibited by the United States Export Administration Act or any other export laws, restrictions or regulations. This agreement shall automatically terminate upon failure of you to comply with it's terms. This agreement may only be modified in writing signed by an officer of Topaz Systems, Inc. This agreement will not be governed by the United Nations convention on contracts for the International sale of goods, the application of which is expressly excluded.

Statement regarding patents, and warning against modification of Topaz products:

Topaz products are covered under US patents 5,120,908 and 5,122,623. Patent work is ongoing and pending. Use of Topaz's products in accordance with our instructions, to the best of our knowledge, does not infringe any patents. However, be aware that there are many patents out there, and modification of our products, or use of our products for other than their intended purpose, could run afoul of these patents. It is the responsibility of Topaz customers to honor relevant patents. U. S. Patent Nos. 5,544,255; 5,647,017; 5,818,955 and others are patents that users of Topaz products should consider if modifications are to be made to our products or if our products are to be used for other than their intended purpose. These patents may be obtained at <http://www.uspto.gov>. Only as examples, without first considering these patents, you should refrain from storing within the signature information (sig file or sig data):

1. A reason for signing or a statement of importance of the signature together with a set of measurements relating to the handwritten signature and either an indication of the signatory or means for comparing said measurements with a set of statistics of a genuine signature to obtain a similarity score.
2. A visual representation of a signature together with a document checksum, hash, or receipt, and the claimed identity of the signatory.
3. A set of measurements relating to biometric information together with a document checksum, hash, or receipt, and the claimed identity of the user.

The foregoing statement is not intended to be an interpretation of the patents or their application to any particular product or implementation.

It may be possible that Federal (FDA) and state digital signature regulations become violated if techniques in 1, 2, and 3 above are employed, so rest assured, Topaz software does not do any of them. Note the information below:

1. The FDA regulations state that "Signed electronic records shall contain information associated with the signing that clearly indicates all of the following ...printed name ... date and time ... The meaning ...". (see FDA guidelines section 11.50) The FDA regulations treat electronic records and electronic signatures as different entities (Section 11.70). Therefore, you should refrain from modifying the Topaz system on your own to store any of the name, date and meaning data in the signature file.
2. State regulations such as CA code of Regulations, Title 2, Division 7, Chapter 10 for example, require that the signature be bound to only the single message that is signed and not to any other message. Therefore you should refrain from modifying the Topaz system on your own to place a checksum in with the signature data and then adding other information in with the signature, and then encrypting them to a secret key. If your were to modify the Topaz system to do this, you would be binding the signature to multiple-messages which is prohibited by the state regulations.
3. State regulations require that the signature data be capable of verification by a forensic document examiner. (CA regulations paragraph 220003(b)(3)(B) for example). Some of the techniques in 1-3 above appear at odds with "lengthy process of handwriting analysis" required by state regulation (see state of CA FAQ section). With the Topaz system, you are assured of the most accurate forensic handwriting analysis results, since the Topaz .sig file is saving all of the original signature data.

In addition, there are potential security and refutability issues if too many things are crammed in with the signature. Leave the document data, reason for signing, receipt, and identity information in the document as GemTools is designed to work, and as complies with FDA and state regulations. That way, the signature is bound to all of the data, not just to part of the data. It is very easy to create all kinds of standard image files of the signature using GemTools software. Topaz signature & document security results from our patent pending methods of encrypting the .sig data directly to a document hash (not by putting the hash or receipt into the .sig file), and non-repudiation results from the proper use of (again patent pending) Topaz signature and document receipts. Please contact the factory if you have any additional questions or would like more information about your rights concerning patents as a customer or developer using Topaz products.

Important Notice:

This software or any or all additional documentation, guidelines, or examples do not constitute a warranty about the performance, security, or legal acceptability of GemTools software or the SigPlus control in any specific use or implementation. To the extent that GemTools or SigPlus are used to achieve regulatory or other specific objectives within an industry, you must consult competent experts or regulatory officials together with your own plan to achieve your desired business objectives using the Topaz tools.