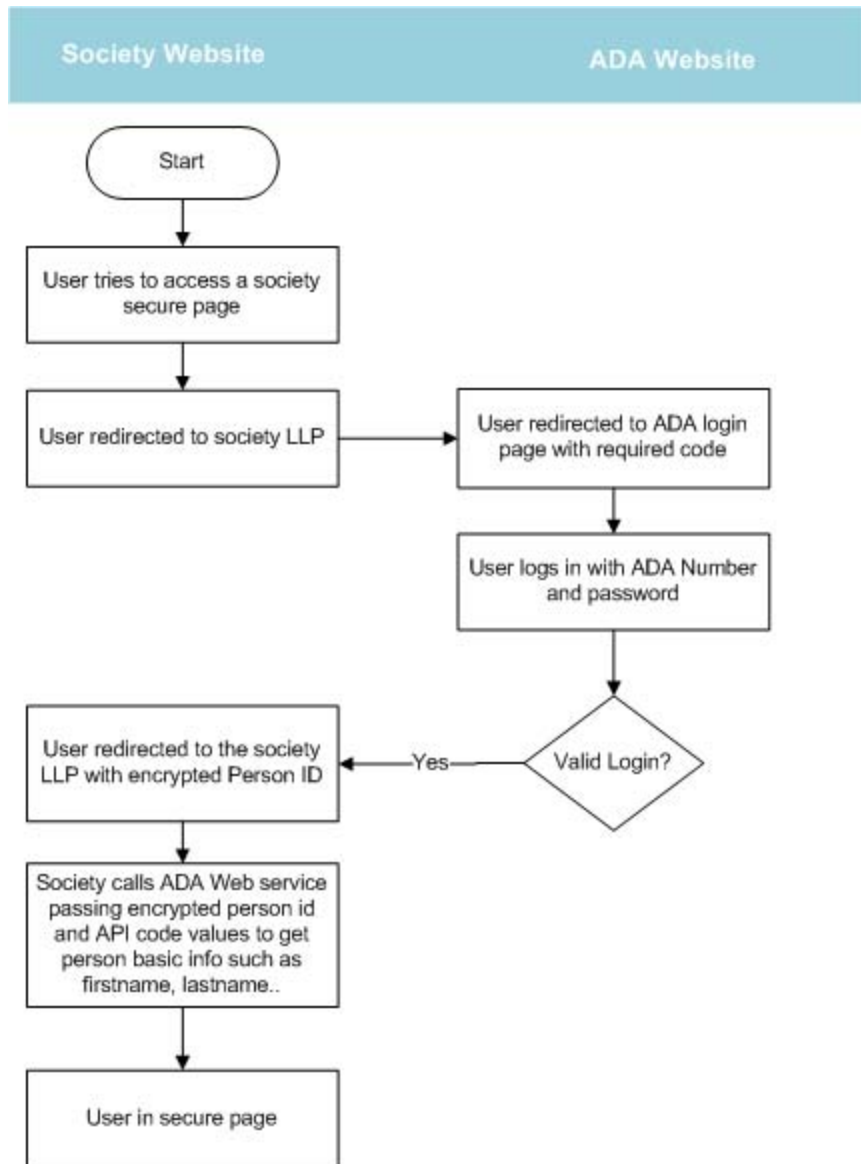# ADA –SSO Solution Guide

Welcome to the ADA-SSO Developer's Guide. This guide provides the information you need to develop the ADA-SSO solution on your website.

The ADA-SSO solution offers a fast and convenient way to access the ADA-SSO functionality. With this solution, the ADA lets you incorporate the ADA-SSO solution technology in your website. Your users can use one credential to login in to both your website and the ADA's website.

Business Processes and Rules

- This solution is restricted to Societies using Aptify.

- Societies are provided both testing and production access to the ADA's web services.

- Societies need to provide the URL of their login landing page to the ADA. If the URL changes at a later date, you will need to inform the ADA immediately. Failing to do so will result in processing errors.

SSO flow diagram

Society Website	ADA Website

```
                    ┌─────────┐
                    │  Start  │
                    └─────────┘
                         │
                         ▼
            ┌──────────────────────┐
            │ User tries to access │
            │  a society secure    │
            │        page          │
            └──────────────────────┘
                         │
                         ▼
   ┌─────────────────────────┐        ┌──────────────────────┐
   │ User redirected to      │───────▶│ User redirected to   │
   │ society LLP             │        │ ADA login page with  │
   │                         │        │ required code        │
   └─────────────────────────┘        └──────────────────────┘
                                                │
                                                ▼
                                      ┌──────────────────────┐
                                      │ User logs in with    │
                                      │ ADA Number and       │
                                      │ password             │
                                      └──────────────────────┘
                                                │
                                                ▼
   ┌─────────────────────────┐               ◇
   │ User redirected to the  │◀───Yes───  Valid Login?
   │ society LLP with        │               ◇
   │ encrypted Person ID     │
   └─────────────────────────┘
                │
                ▼
   ┌─────────────────────────┐
   │ Society calls ADA Web   │
   │ service passing         │
   │ encrypted person id and │
   │ API code values to get  │
   │ person basic info such  │
   │ as firstname, lastname..│
   └─────────────────────────┘
                │
                ▼
   ┌─────────────────────────┐
   │ User in secure page     │
   └─────────────────────────┘
```

Setting up the solution:

Step 1: Add a new web reference to ADA web service

In your project solution add a web reference to this URL:
http://test.ebusiness.ada.org/webservices/PO3Info.asmx
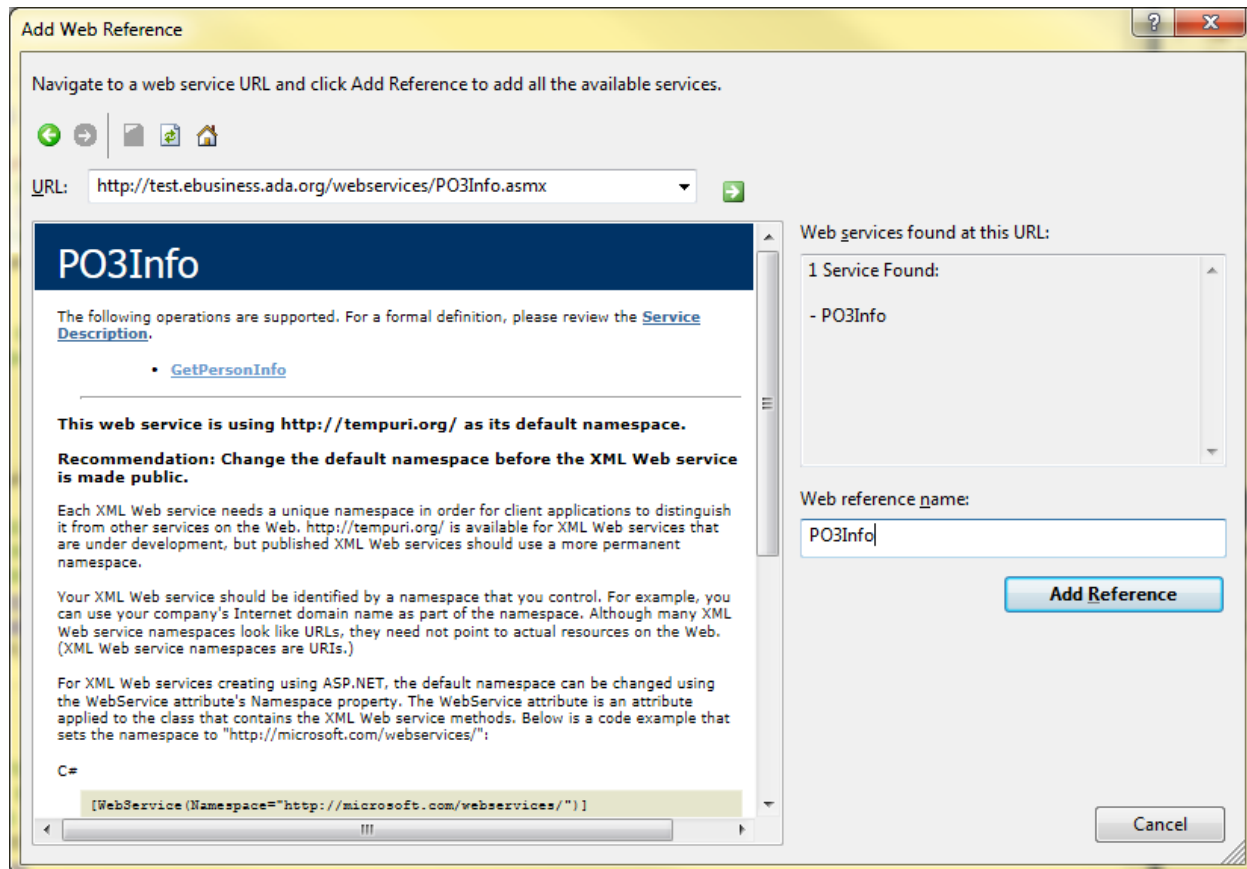
Step 2: Create a new login landing page

Create a new login landing page (LLP). All secure pages in your website should take the user to the LLP if the user hasn't already logged in. The LLP will redirect to the ADA's login page.

The ADA's login page accepts two parameters from your website:

Parameter name	Required	Description
PO3ORGAPICODE	Yes	This is the code the ADA will use to uniquely identify your Society. ADA will supply you with this value
PO3OrgTargetURL	No	ADA passes this parameter to your website after authentication is complete. You can use the parameter's value to redirect to this page

ADA test login page URL: https://test.ebusiness.ada.org/login/loginpo3.aspx

The request to the ADA login page looks like:
```
https://test.ebusiness.ada.org/login/loginpo3.aspx?PO3ORGAPICODE=TS2424J2H5J2HK5&PO3OrgTarget
URL=memberinfo.aspx
```

In this example, the code we will use to uniquely identify the test society is: TS2424J2H5J2HK5

Example:

```
Sub Call_ADA_LoginPage(ByVal pTargetUrl As String)
    Dim strPath As String
    'IT TAKES USER TO ADA LOGIN PAGE, YOU NEED TO PASS YOUR API CODE IN PO3ORGAPICODE PARAMETER
    'PO3OrgTargetURL PARAMETER IS OPTIONAL, BASED ON YOUR SETTING YOU CAN PASS IT TO ADA LOGIN PAGE OR IGNORE IT
    'ADA PROVIDE YOUR API CODE
    strPath = "https://test.ebusiness.ada.org/login/loginpo3.aspx?PO3ORGAPICODE=TS2424J2H5J2HK5&PO3OrgTargetURL=" & pTargetUrl
    Response.Redirect(strPath)
End Sub
```

Step 3: ADA authenticates the user

ADA renders the login page with both the ADA's logo and the society's logo. After authentication completes, the ADA's login page will redirect to the society's LLP with an encrypted Person ID in the EPID parameter.

ADA response example:

```
SocietyWebsite/LoginLandingPage.aspx?EPID=xbtBShJ0mX4=&OrgTargetURL=memberinfo.aspx
```

Step 3: Society calls ADA web server to get user info

This response means the authentication was successful. The society should call the ADA web service to get the user information and based on the society business rules decides if user have access to the society website or not.

ADA web service (http://test.ebusiness.ada.org/webservices/PO3Info.asmx) includes two functions: GetPersonInfo and GetPersonAdditionalInfo. State API code and Encrypted person ID should be passed to these functions to get the user information in xml format.

<u>**GetPersonInfo Function:**</u>
This function returns the person basic information:

Xml result example:

```xml
<?xml version="1.0" encoding="utf-16" standalone="yes"?>
 <ADAPersonInfo>
  <Person>
      <PersonID>187202</PersonID>
      <FirstName>Peter</FirstName>
      <LastName>Bradley</LastName>
      <ADANumber>161960813</ADANumber>
      <PrimaryFunction>Dentist</PrimaryFunction>
      <PrimaryFunctionID>11</PrimaryFunctionID>
      <MemberStatus>Tripartite Member</MemberStatus>
      <MemberStatusID>12</MemberStatusID>
      <ComponentName>South Florida District Dental Association</ComponentName>
      <ComponentID>163</ComponentID>
      <ComponentJurisdiction>10B</ComponentJurisdiction>
      <ConstituentName>Florida Dental Association</ConstituentName>
      <ConstituentID>16</ConstituentID>
      <ConstituentJurisdiction>10</ConstituentJurisdiction>
      <PrimaryEmail>bradleyp@ada.org</PrimaryEmail>
      <Birthday>12-25-1940</Birthday>
      <Gender>Male</Gender>
      <PracticeTypeID>0</PracticeTypeID>
      <PracticeTypeName />
      <LabelName>Dr Peter B Bradley, PhD</LabelName>
      <FirstLast>Peter Bradley</FirstLast>
  </Person>
 </ADAPersonInfo>
```

Your LLP can read the xml result and based on the information (dentist, member, etc) decide if the user should have access to the site or not and takes the user to the proper page.

Call GetPersoninfo Function:

```vbnet
'ADD ADA WEBSERVICE TO YOUR WEB REFERENCES
'CALL ADA WEBSERVICE FUNCTIONS BY PASSING EPID(ENCRYPTED PERSON ID) ID AND API CODE
' GETPERSONINFO FUNCTION RETURNS THE XML STRING IN RESULT WHICH INCLUDES FIRSTNAME, LASTNAME, ....
Dim wsPInfo As New PO3Info.PO3Info()
Dim strMyXML As String
Dim xnlxnlPersonNodes As XmlNodeList
Dim xnlPersonNode As XmlNode
Dim xnlbaseDataNodes As XmlNodeList
Dim xnlbFirstInRow As Boolean
strMyXML = wsPInfo.GetPersonInfo("TS2424J2H5J2HK5", Request.QueryString("ePID"))
Dim MyDoc As New System.Xml.XmlDocument
MyDoc.LoadXml(strMyXML)
xnlxnlPersonNodes = MyDoc.GetElementsByTagName("Person")
Dim strUserInfo As String = ""
'GET XML STRING VALUE
For Each xnlPersonNode In xnlxnlPersonNodes
    xnlbaseDataNodes = xnlPersonNode.ChildNodes
    xnlbFirstInRow = True
    For Each baseDataNode As XmlNode In xnlbaseDataNodes
        If (xnlbFirstInRow) Then
            xnlbFirstInRow = False
        Else
            strUserInfo += "<br>"
        End If
        strUserInfo += baseDataNode.Name & ": " & baseDataNode.InnerText
    Next
Next
'ADD A SESSION FOR THIS USER
Session.Add("UserInfo", strUserInfo)
```

GetPersonAdditionalInfo Function:

This function includes more details about the user:

Xml result example:

```xml
<?xml version="1.0" encoding="utf-16" standalone="yes"?>
 <ADAPersonAdditionalInfo>
  <Person>
        <PersonID>187202</PersonID>
        <PreferredCity>Chicago</PreferredCity>
        <PreferredZip>60611-2637</PreferredZip>
        <licenseNumber>1111</licenseNumber>
        <licenseExpirationDate>2-26-2017</licenseExpirationDate>
        <GraduationDate>6-1-2005</GraduationDate>
        <GraduationSchool>Howard University</GraduationSchool>
        <SecondaryMembership />
        <WebSiteAccess>False</WebSiteAccess>
        <EmployeeOrganizationId>0</EmployeeOrganizationId>
        <IsEmployee>False</IsEmployee>
        <ComponentMembershipProductWebName />
        <ADAMembershipProductWebName />
        <ConstituentMembershipProductWebName />
        <MembershipRenewalDate>12/31/2013</MembershipRenewalDate>
        <PermissionLevel1>True</PermissionLevel1>
        <PermissionLevel2>False</PermissionLevel2>
        <PermissionLevel3>False</PermissionLevel3>
        <PermissionLevel1ID>24572</PermissionLevel1ID>
        <PermissionLevel2ID>0</PermissionLevel2ID>
        <PermissionLevel3ID>0</PermissionLevel3ID>
        <PersonPhoto>http://test.ebusiness.ada.org/myada/profileimages.aspx?id=187202&type=photo</PersonPhoto>
   </Person>
</ADAPersonAdditionalInfo>
```

Call GetPersonAdditionalinfo Function:

```vb
'CALL SECOND FUNCTION: GETPERSONADDITIIONALINFO
Dim strMyXML2 As String
Dim xnlxnlPersonNodes2 As XmlNodeList
Dim xnlPersonNode2 As XmlNode
Dim xnlbaseDataNodes2 As XmlNodeList
Dim xnlbFirstInRow2 As Boolean
strMyXML2 = wsPInfo.GetPersonAdditionalInfo("TS2424J2H5J2HK5", Request.QueryString("ePID"))
Dim MyDoc2 As New System.Xml.XmlDocument
MyDoc2.LoadXml(strMyXML2)
xnlxnlPersonNodes2 = MyDoc2.GetElementsByTagName("Person")
Dim strUserInfo2 As String = ""
'GET XML STRING VALUE
For Each xnlPersonNode2 In xnlxnlPersonNodes2
    xnlbaseDataNodes2 = xnlPersonNode2.ChildNodes
    xnlbFirstInRow2 = True
    For Each baseDataNode2 As XmlNode In xnlbaseDataNodes2
        If (xnlbFirstInRow2) Then
            xnlbFirstInRow2 = False
        Else
            strUserInfo2 += "<br>"
        End If
        strUserInfo2 += baseDataNode2.Name & ": " & baseDataNode2.InnerText
    Next
Next
Session.Add("UserInfo2", strUserInfo2)
```

Terms and Definition

- **Person ID**: Person Aptify internal ID
    - o **Type**: Number
    - o **Allow Empty**: NO

- **First name**: Person first name
    - o **Type**: String
    - o **Allow Empty**: NO

- **Last name:** Person last name
    - o **Type**: String
    - o **Allow Empty**: NO

- **ADA Number:** The number that the ADA assigns to dentists and dental students
    - o **Type**: String
    - o **Allow Empty**: YES

- **Primary Function**: Each person record has a primary function in the system, primary function describes what type of person they are. The main functions are:
    - o 11 Dentist
    - o 12 Hygienist
    - o 18 Students

    Possible values:

Primary Function ID	Primary Function
11	Dentist
12	Hygienist
13	ADA Affil
14	ADA Assoc
15	ADA Honor
16	ADA Intl
17	Intl Stud
18	Student
25	ADA Staff
26	Assistant
28	Comp Staff
30	Constit Staff

- o **Type**: String/Number
- o **Allow Empty**: NO

- **Member Status:** ADA has a different type of members
  - Member status ID 3, 10, 11, 12, 13 or 14 are known as Member in ADA level.
  - Member status 13 is known as Tripartite Member, which is society member.

  All Possible values for member status are:

Member Status ID	Member Status
1	All Types
2	Non-Member
3	Member
4	ADA Reseller
6	ADA Constituent
8	ADA Commercial
9	ADA Special Non-Member
10	Applicant
11	ADA Direct Member
12	Tripartite Member
13	Student Member
14	Member-Pending Cutoff
5	ADA International
7	ADA Staff

  - **Type**: String/Number
  - **Allow Empty**: NO

- **Constituent and Component:**
  The ADA is made up of 3 organizations: National, State and Local. National is ADA, state is constituent and local is component, for example a dentist in Chicago would join the following 3 organizations: ADA, Illinois, Chicago.

  ConstituentID: Aptify Internal ID of the state society that they are a member of (or would be a member of if they are currently a non-mbr)
  - **Type**: Number
  - **Allow Empty**: YES

  ConstituentName: Name of the state society that they are a member of (or would be a member of if they are currently a non-mbr).
  - **Type**: String
  - **Allow Empty**: YES

  ComponentID: Aptify Internal ID of the Component that they are a member of (or would be a member of if they are currently a non-mbr)

- o **Type**: Number
- o **Allow Empty**: YES

Component Name: Name of the Component that they are a member of (or would be a member of if they are currently a non-mbr)

- o **Type**: String
- o **Allow Empty**: YES

- **Constituent Jurisdiction**: The tripartite number for the jurisdiction
  - o **Type**: String
  - o **Allow Empty**: YES

- **PrimaryEmailAddress:** Person email address
  - o **Type**: String
  - o **Allow Empty**: YES

- **Birthday**: Person Birthday
  - o **Type**: Date
  - o **Allow Empty**: YES

- **Gender**: Person's gender, Possible values for gender are: Unknown, Male, Female
  - o **Type**: String
  - o **Allow Empty**: YES

- **Practice Type:** Possible values:

Practice Type ID	Practice Type Name
1	Owner
2	Non Owner
3	Associate
4	Independent Contractor
5	Owner - Group Practice
6	Non-owner - Group Practice
7	Associate - Group Practice
8	Unknown
1	Owner

- o **Type**: String/Number
- o **Allow Empty**: YES

- **Label Name:** Aptify generated label by person name and suffix
  - o **Type**: String

        o **Allow Empty**: NO

- **First Last :** Person first and last name
  - o **Type**: String
  - o **Allow Empty**: NO

- **Preferred City**: The dentist may supply the ADA with multiple addresses. One of their addresses must be identified as their "Preferred Address". This is where the ADA mails to. Do not assume PreferredCity is the city their office is in. It may be their Office, Home, Vacation Home, PO Box, or any other address type that they have identified as their Preferred Address.
  - o **Type**: String
  - o **Allow Empty**: YES

- **PreferredZip:** Similar to above, the zip code tied to their Preferred Address.
  - o **Type**: String
  - o **Allow Empty**: YES

- **License Number:** This is the Active license in the state related to the API Code. Many dentists have several licenses.. ADA is only supplying "active" licenses in this field, and then only active licenses for the state making the API call.
  - o **Type**: String
  - o **Allow Empty**: YES

- **LicenseExpirationDate:** The date the license expires.
  - o **Type**: Date
  - o **Allow Empty**: YES

- **GraduationDate**: Date graduated
  - o **Type**: Date
  - o **Allow Empty**: YES

- **GraduationSchool:** The name of the school from which the dentist graduated
  - o **Type**: String
  - o **Allow Empty**: YES

- **SecondaryMembership:** A secondary membership such as "Allied" or "Affiliate" that the state has sold to the dentist/hygienist/etc. Until the states begin selling these (probably for 2015), this field will be blank.
  - o **Type**: String
  - o **Allow Empty**: YES

- **ISEmployee:** In Aptify each ADA/State/Component's employee has an employee record and web credential which they can use to login to ADA/State website. if the person who is trying to login is ADA/State/Component employee this field return True
  - **Type**: String
  - **Allow Empty**: YES

- **EmployeeOrganizatioinID:** Similar to above, this field return the organization ID of the employee.
  - **Type**: String
  - **Allow Empty**: YES

- **ADAMembershipProductWebName:** Web name of the ADA Membership Product for the current membership year
  - **Type**: String
  - **Allow Empty**: YES

- **ConstituentMembershipProductWebName:** Web name of the Constituent Membership Product for the current membership year
  - **Type**: String
  - **Allow Empty**: YES

- **ComponentMembershipProductWebName**: Web name of the component Membership Product for the current membership year
  - **Type**: String
  - **Allow Empty**: YES

- **MembershipRenewalDate:** The date the Membership expires
  - **Type**: String
  - **Allow Empty**: YES

- **PermissionLevel1:** Society can assign 3 different list to their API code in Aptify. PermissionLevel1 shows if the person name is in the first list of society or not. If yes, the value would be True.
  - **Type**: String
  - **Allow Empty**: YES

- **PermissionLevel2:** Society can assign 3 different list to their API code in Aptify. PermissionLevel2 shows if the person name is in the second list of society or not. If yes, the value would be True.
  - **Type**: String
  - **Allow Empty**: YES

- **PermissionLevel3:** Society can assign 3 different list to their API code in Aptify. PermissionLevel3 shows if the person name is in the third list of society or not. If yes, the value would be True.
  - **Type**: String
  - **Allow Empty**: YES

- **PermissionLevel1ID:** Society can assign 3 different list to their API code in Aptify. PermissionLevel1ID shows the first list ID If the user belongs to that list.

- o **Type**: Number
- o **Allow Empty**: YES
- **PermissionLevel2ID:** Society can assign 3 different list to their API code in Aptify. PermissionLevel2ID shows the second list ID If the user belongs to that list.
  - o **Type**: Number
  - o **Allow Empty**: YES
- **PermissionLevel3ID:** Society can assign 3 different list to their API code in Aptify. PermissionLevel3ID shows the third list ID If the user belongs to that list.
  - o **Type**: Number
  - o **Allow Empty**: YES
- **PersonPhoto:**The URL of the person photo
  - o **Type**: String
  - o **Allow Empty**: YES

Note: The society can use these 3 different lists for different purposes, for example

- They have a member access to a particular section of their website

- They have a list of non-member non-dentist that should have access to their website

1) They create a list in Aptify, add all privileged person records to the list

2) Provide the list ID to ADA

3) Work with developer team to implement the related solution in the web

4) When the user from the list logs in, ADA webservice return the true flag, for related list id and the user should have access to the particular area

Note: As long as a person has an Aptify ID and password they are able to log in to the ADA's and Society's websites. It is up to the Societies to decide what to display to the person logging on to their site.

If they are a member of the society their Member Status will be "Tripartite Member" and Constituent ID will be the society ID.

If they are a member of ADA, their Member Status will be one of "Member", "Applicant", "ADA Direct Member", "Tripartite Member", "Student Member" or "Member-Pending Cutoff" values, and Constituent ID will be any other value other than society Constituent ID.

Members who have not paid dues as of mid-April (date is determined every year), get switched from Tripartite Member to Member-Pending Cutoff. ADA still considers them members but it is up to the society to decide if they want to consider them members. In early July, everyone with a status of M-PC gets changed to Non-Member.

ADA will provide the "Constituent ID" of the society in a separate email.

Log out

For the logout process, the society needs to have a page to kill all the sessions and cookies of the current user, and then call https://test.ebusiness.ada.org/login/logoutpo3.aspx page, this page will kill all the session from the ADA side for the current user, by passing PO3ReturnURL parameter to the page, system redirect the user to parameter value page after killing all the sessions

For example by calling
https://test.ebusiness.ada.org/login/logoutpo3.aspx?PO3ReturnURL=http://www.statedental.org system will take the user to the http://www.statedental.org after log out at ADA part.

Example:

```
|
}Partial Class logout
    Inherits System.Web.UI.Page

}    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
        If Not Session("UserInfo") Is Nothing Then
            Session.Clear()
            Response.Redirect("https://test.ebusiness.ada.org/login/logoutpo3.aspx?PO3ReturnURL=http://localhost:50988/ADA-P(
        End If
    End Sub
End Class
```