PEARSON ALWAYS LEARNING

EQUELLA®

C# Soap Tutorial

Version 6.4

Document History

Document No.	Reviewed		Finalised		Published	
1	22/05/2015		22/05/2015			

May 2015 edition.

Information in this document may change without notice. EQUELLA® and its accompanying documentation are furnished under a non-disclosure, evaluation agreement or licence agreement. Copying, storing, transmitting, or otherwise reproducing the software or this document in any form without written permission from Pearson is strictly forbidden.

All products, other than EQUELLA®, named in this document are the property of their respective owners.

Property of:

Pearson
Level 1, 11 Elizabeth Street
Hobart, Australia, 7000

Copyright © 2015 Pearson

Contact documentation@equella.com for matters relating to documentation.

Table of Contents

C# SOAP tutorial 4
Running the examples 4
Guide to the example code 4
Writing your own code 5
Contact Client Support 6

C# SOAP tutorial

Running the examples

Visual Studio Express 2013 for Web is a free IDE for developing C# web applications and is available at http://www.visualstudio.com/downloads/download-visual-studio-vs#d-express-web

Setup a new Web Site project in Visual Studio Express 2013 for Web and point to the **samples\dotnet** folder as the root of the web site. To run the example files, edit **Settings.Config** with an EQUELLA username and password, and edit **Web.Config** at the path *system.serviceModel/client/endpoint/@address* (towards the bottom of the file) with the EQUELLA SOAP endpoint. The SOAP endpoint is of the format: *http://INSTITUTION_URL/services/SoapService51*

For example: http*://mydomain.edu.au/myinst/services/SoapService51*

Notice how the *system.serviceModel/bindings/basicHttpBinding/binding/@allowCookies* value is set to true. This attribute must be set to true for all new projects.

Feel free to customise or extend the sample files for your own purposes.

Guide to the example code

A web reference to SoapService51 has already been added to the Web Site project. To add other SOAP services, add a new Service Reference (right click the App_WebReferences folder in the Solution Explorer window and select *Add Service Reference...*) and enter the URL of the SOAP service's WSDL file. For example, *http://INSTITUTION_URL/services/calactivation.service?wsdl*

The example code contains a helper class (EQUELLASOAP) which you are free to use and modify. This class has been created to simplify the processing of the XML results and XML parameters returned from and supplied to the SOAP methods. The EQUELLASOAP class is not required to use the EQUELLA SOAP service, it is simply provided for convenience of extension.

Search

This example shows how you can retrieve a list of items from EQUELLA based on a few search parameters. The searching code is contained within the cmdSearch_Click method. Here is a summary of that code:

- It creates an EQUELLASOAP object which is a thin wrapper around the automatically generated SoapService51 client. The EQUELLASOAP object ensures that the user is logged in and maintains a reference to the logged in SOAP client.
- It invokes searchItems on the EQUELLASOAP object. This method is also a thin wrapper around the SoapService51 method. This method returns an XElement object for convenience, rather than raw XML string returned by the SoapService51.

- It retrieves information from the XML, such as the number of results returned, the number of results available and the actual result list. The format of the returned XML and the meanings/values of the parameters are documented in the *SOAP API* documentation.

Contribute

This example shows how you can contribute an item to a collection and give it a name, description and a file attachment. The contribution code is contained within the cmdSubmit_Click method. Here is a summary of that code:

- It creates an EQUELLASOAP object which is a thin wrapper around the automatically generated SoapService51 client. The EQUELLASOAP object ensures that the user is logged in and maintains a reference to the logged in SOAP client.
- It invokes newItem on the EQUELLASOAP object. Invoking newItem will create a new 'staging' area on the server for this item while it is being edited. The item will *not* be created on the server until saveItem is called, however you should edit the details of the item before doing so, as this example does.
- It sets the name and description of the item by setting the values of the XML
- It determines if a file has been uploaded and if so, it uploads the file to the staging area. Note that this does not create an attachment on the item, it simply uploads the file. The uploadFile method on the EQUELLASOAP class converts the binary file into a base 64 encoded string for passing to the SOAP method.
- It creates an attachment node for the uploaded file on the item XML. The format of the attachment node is documented in the *Item XML format* section of the *SOAP API* documentation.
- It invokes the saveItem method with a submit value of true. A submit value of true means that the item will now either go live or move onto the next moderation step, a value of false means the item will be saved with a status of '*draft*'.

Writing your own code

In general the process for invoking a SOAP method is as follows:

1. Create a SoapService51 object.
2. Call the login method:

```
client.login(username, password);
```

The user that you login as must have sufficient privileges to perform your required task. You cannot do anything over SOAP that you cannot do within the EQUELLA web UI.

3. Invoke a method:

```
collectionsXml = client.getContributableCollections();
```

(NOTE: All SOAP methods will return either a simple string, number or an XML string).

4. Call the logout method.

This is not strictly necessary, the user session will eventually time out on the server, but it is good practice to call logout to free server side resources.

Contact Client Support

We are always happy to help.

If your organisation has a support agreement with EQUELLA then help is available at http://equella.custhelp.com.