FilePro Connector Web Service API

Overview

The FilePro web service API is composed of the following two web services:

1. https://norfolk.civicweb.net/Global/WebServices/Login.asmx
2. https://norfolk.civicweb.net/Global/WebServices/Document.asmx

These web services use cookies to maintain the login session.

A FilePro document Id number returned from the web service can be used to link to FilePro using the following format:

"https://norfolk.civicweb.net/Documents/DocumentList.aspx?ID=<Id>"

Login.asmx

This web service has the following two methods:

1. bool LoginUser(string userName, string password)
2. void LogoutUser()

LoginUser

This method logs the user with the supplied user name and password into CivicWeb. Setting the user name and password to empty strings will log on the public user. The method returns true if the login succeeded and false if not.

This method has the following parameters:

Parameter	Type	Description
userName	string	The user name.
password	string	The password.

If the login succeeded, the response from the web service will contain the following cookie:

- CurrentSession

This cookie must be sent with the request to the other web service methods in order for the user session to be recognized and maintained. The session is maintained for 60 minutes after the most recent web service request.

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Login.asmx HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://civicweb.net/LoginUser"
```

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LoginUser xmlns="http://civicweb.net/">
      <userName>string</userName>
      <password>string</password>
    </LoginUser>
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LoginUserResponse xmlns="http://civicweb.net/">
      <LoginUserResult>boolean</LoginUserResult>
    </LoginUserResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced
with actual values.

```
POST /Global/WebServices/Login.asmx HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LoginUser xmlns="http://civicweb.net/">
      <userName>string</userName>
      <password>string</password>
    </LoginUser>
  </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LoginUserResponse xmlns="http://civicweb.net/">
      <LoginUserResult>boolean</LoginUserResult>
    </LoginUserResponse>
  </soap12:Body>
</soap12:Envelope>
```

HTTP POST

The following is a sample HTTP POST request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Login.asmx/LoginUser HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: application/x-www-form-urlencoded
Content-Length: length

userName=string&password=string

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<boolean xmlns="http://civicweb.net/">boolean</boolean>
```

LogoutUser

This method logs the user out of the system. The cookie is cleared in the response.

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Login.asmx HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://civicweb.net/LogoutUser"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LogoutUser xmlns="http://civicweb.net/" />
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
```

```
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LogoutUserResponse xmlns="http://civicweb.net/" />
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Login.asmx HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LogoutUser xmlns="http://civicweb.net/" />
  </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LogoutUserResponse xmlns="http://civicweb.net/" />
  </soap12:Body>
</soap12:Envelope>
```

HTTP POST

The following is a sample HTTP POST request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Login.asmx/LogoutUser HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: application/x-www-form-urlencoded
Content-Length: length

HTTP/1.1 200 OK
```

Document.asmx

This web service has four methods:

1. IdCollection GetSearchResults(string exactPhrase, string allWords, string atLeastOneWord, string wordsInProximity1, string wordsInProximity2, string wordsNotIncluded)
2. DocumentSearchResultCollection GetSearchResultsDetails(IdCollection searchResultsSubset, string exactPhrase, string allWords, string atLeastOneWord, string wordsInProximity1, string wordsInProximity2, string wordsNotIncluded)
3. DocumentChildListResult GetChildList(int id, string path, bool includeDocuments, string documentProvider, string controlID)
4. DocumentFileResult GetOriginalFile(int id)

GetSearchResults

This method returns a list of document ids that result from a search of FilePro based on the criteria submitted.

This method has the following parameters:

Parameter	Type	Description
exactPhrase	string	Searches for documents containing this exact phrase
allWords	string	Searches for documents containing all of these words. The words must be separated by spaces.
atLeastOneWord	string	Searches for documents containing at least one of these words. The words must be separated by spaces.
wordsInProximity1	string	Searches for documents containing this word or phrase in close proximity to wordsInProximity2.
wordsInProximity2	string	Searches for documents containing this word or phrase in close proximity to wordsInProximity1.
wordsNotIncluded	string	Searches for documents that do not contain any of these words. The words must be separated by spaces.

The IdCollection returned by the method is a list of integer document ids.

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx HTTP/1.1
Host: laurentia
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://civicweb.net/GetSearchResults"

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetSearchResults xmlns="http://civicweb.net/">
      <exactPhrase>string</exactPhrase>
      <allWords>string</allWords>
      <atLeastOneWord>string</atLeastOneWord>
      <wordsInProximity1>string</wordsInProximity1>
      <wordsInProximity2>string</wordsInProximity2>
      <wordsNotIncluded>string</wordsNotIncluded>
    </GetSearchResults>
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetSearchResultsResponse xmlns="http://civicweb.net/">
      <GetSearchResultsResult>
        <int>int</int>
        <int>int</int>
      </GetSearchResultsResult>
    </GetSearchResultsResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx HTTP/1.1
Host: laurentia
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetSearchResults xmlns="http://civicweb.net/">
      <exactPhrase>string</exactPhrase>
      <allWords>string</allWords>
      <atLeastOneWord>string</atLeastOneWord>
      <wordsInProximity1>string</wordsInProximity1>
      <wordsInProximity2>string</wordsInProximity2>
      <wordsNotIncluded>string</wordsNotIncluded>
```

```
      </GetSearchResults>
   </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
   <soap12:Body>
      <GetSearchResultsResponse xmlns="http://civicweb.net/">
         <GetSearchResultsResult>
            <int>int</int>
            <int>int</int>
         </GetSearchResultsResult>
      </GetSearchResultsResponse>
   </soap12:Body>
</soap12:Envelope>
```

HTTP POST

The following is a sample HTTP POST request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx/GetSearchResults HTTP/1.1
Host: laurentia
Content-Type: application/x-www-form-urlencoded
Content-Length: length

exactPhrase=string&allWords=string&atLeastOneWord=string&wordsInProximity1=string&wor
dsInProximity2=string&wordsNotIncluded=string

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<ArrayOfInt xmlns="http://civicweb.net/">
   <int>int</int>
   <int>int</int>
</ArrayOfInt>
```

GetSearchResultsDetails

This method returns a list of DocumentSearchResult objects, which contain detailed search result information, formatted in html, for a specific list of documents. This information is intended to be displayed on a search result html page and include marking of search criteria and sample text from the document.

This method has the following parameters:

Parameter	Type	Description
searchResultsSubset	IdCollection	This is a list of integer document

		ids, for which search result information will be returned. The function will only return information for up to 50 documents at a time, no matter how may more ids are in the list.
exactPhrase	string	Searches for documents containing this exact phrase
allWords	string	Searches for documents containing all of these words. The words must be separated by spaces.
atLeastOneWord	string	Searches for documents containing at least one of these words. The words must be separated by spaces.
wordsInProximity1	string	Searches for documents containing this word or phrase in close proximity to wordsInProximity2.
wordsInProximity2	string	Searches for documents containing this word or phrase in close proximity to wordsInProximity1.
wordsNotIncluded	string	Searches for documents that do not contain any of these words. The words must be separated by spaces.

The DocumentSearchResult object has the following properties:

Parameter	Type	Description
Id	int	The document id. This is used to link to the search result using the format "https://norfolk.civicweb.net/Documents/DocumentList.aspx?ID=<Id>"
TitleHtml	string	The document title, with search criteria marked in html.
SampleHtml	string	Sample text from the document, with search criteria marked in html.
PathHtml	string	The path of the document, with html links to each parent folder, separated by > (>) characters.

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx HTTP/1.1
Host: laurentia
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://civicweb.net/GetSearchResultDetails"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```

```
    <soap:Body>
      <GetSearchResultDetails xmlns="http://civicweb.net/">
        <searchResultsSubset>
          <int>int</int>
          <int>int</int>
        </searchResultsSubset>
        <exactPhrase>string</exactPhrase>
        <allWords>string</allWords>
        <atLeastOneWord>string</atLeastOneWord>
        <wordsInProximity1>string</wordsInProximity1>
        <wordsInProximity2>string</wordsInProximity2>
        <wordsNotIncluded>string</wordsNotIncluded>
      </GetSearchResultDetails>
    </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetSearchResultDetailsResponse xmlns="http://civicweb.net/">
      <GetSearchResultDetailsResult>
        <DocumentSearchResult>
          <Id>int</Id>
          <TitleHtml>string</TitleHtml>
          <SampleHtml>string</SampleHtml>
          <PathHtml>string</PathHtml>
        </DocumentSearchResult>
        <DocumentSearchResult>
          <Id>int</Id>
          <TitleHtml>string</TitleHtml>
          <SampleHtml>string</SampleHtml>
          <PathHtml>string</PathHtml>
        </DocumentSearchResult>
      </GetSearchResultDetailsResult>
    </GetSearchResultDetailsResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx HTTP/1.1
Host: laurentia
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetSearchResultDetails xmlns="http://civicweb.net/">
      <searchResultsSubset>
        <int>int</int>
        <int>int</int>
      </searchResultsSubset>
      <exactPhrase>string</exactPhrase>
      <allWords>string</allWords>
      <atLeastOneWord>string</atLeastOneWord>
      <wordsInProximity1>string</wordsInProximity1>
      <wordsInProximity2>string</wordsInProximity2>
      <wordsNotIncluded>string</wordsNotIncluded>
    </GetSearchResultDetails>
  </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetSearchResultDetailsResponse xmlns="http://civicweb.net/">
      <GetSearchResultDetailsResult>
        <DocumentSearchResult>
          <Id>int</Id>
          <TitleHtml>string</TitleHtml>
          <SampleHtml>string</SampleHtml>
          <PathHtml>string</PathHtml>
        </DocumentSearchResult>
        <DocumentSearchResult>
          <Id>int</Id>
          <TitleHtml>string</TitleHtml>
          <SampleHtml>string</SampleHtml>
          <PathHtml>string</PathHtml>
        </DocumentSearchResult>
      </GetSearchResultDetailsResult>
    </GetSearchResultDetailsResponse>
  </soap12:Body>
</soap12:Envelope>
```

GetChildList

This method returns a DocumentChildListResult object, which contains a list of child document information.

This method has the following parameters:

Parameter	Type	Description

Id	int	This is the id of the parent folder. An id of 0 get a list of the root folders and documents.
Path	string	This parameter must be an empty string.
includeDocuments	bool	This if true if you want to return documents as well as folders, and false if not.
documentProvider	string	This must be the string "iCompass.CivicWeb.Items.DocumentProvider".
controlID	string	This parameter must be an empty string.

The DocumentChildListResult object has the following properties:

Parameter	Type	Description
Id	int	The parent folder id.
Nodes	List<DocumentTreeNodeInformation>	A list of DocumentTreeNodeInformation objects, which contain the information for each child document.

The DocumentTreeNodeInformation object has the following properties:

Parameter	Type	Description
Id	int	The document or folder id.
Name	string	The title of the document or folder.
Folder	bool	If this is a folder, true, otherwise, false.

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx HTTP/1.1
Host: laurentia
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://civicweb.net/GetChildList"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetChildList xmlns="http://civicweb.net/">
      <id>int</id>
      <path>string</path>
      <includeDocuments>boolean</includeDocuments>
      <documentProvider>string</documentProvider>
      <controlID>string</controlID>
    </GetChildList>
  </soap:Body>
</soap:Envelope>
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetChildListResponse xmlns="http://civicweb.net/">
      <GetChildListResult>
        <ControlID>string</ControlID>
        <ID>int</ID>
        <Path>string</Path>
        <Nodes>
          <DocumentTreeNodeInformation>
            <ID>int</ID>
            <Name>string</Name>
            <Path>string</Path>
            <Folder>boolean</Folder>
          </DocumentTreeNodeInformation>
          <DocumentTreeNodeInformation>
            <ID>int</ID>
            <Name>string</Name>
            <Path>string</Path>
            <Folder>boolean</Folder>
          </DocumentTreeNodeInformation>
        </Nodes>
      </GetChildListResult>
    </GetChildListResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx HTTP/1.1
Host: laurentia
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetChildList xmlns="http://civicweb.net/">
      <id>int</id>
      <path>string</path>
      <includeDocuments>boolean</includeDocuments>
      <documentProvider>string</documentProvider>
      <controlID>string</controlID>
```

```
        </GetChildList>
      </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetChildListResponse xmlns="http://civicweb.net/">
      <GetChildListResult>
        <ControlID>string</ControlID>
        <ID>int</ID>
        <Path>string</Path>
        <Nodes>
          <DocumentTreeNodeInformation>
            <ID>int</ID>
            <Name>string</Name>
            <Path>string</Path>
            <Folder>boolean</Folder>
          </DocumentTreeNodeInformation>
          <DocumentTreeNodeInformation>
            <ID>int</ID>
            <Name>string</Name>
            <Path>string</Path>
            <Folder>boolean</Folder>
          </DocumentTreeNodeInformation>
        </Nodes>
      </GetChildListResult>
    </GetChildListResponse>
  </soap12:Body>
</soap12:Envelope>
```

HTTP POST

The following is a sample HTTP POST request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx/GetChildList HTTP/1.1
Host: laurentia
Content-Type: application/x-www-form-urlencoded
Content-Length: length

id=string&path=string&includeDocuments=string&documentProvider=string&controlID=strin
g

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
```

```
<DocumentChildListResult xmlns="http://civicweb.net/">
  <ControlID>string</ControlID>
  <ID>int</ID>
  <Path>string</Path>
  <Nodes>
    <DocumentTreeNodeInformation>
      <ID>int</ID>
      <Name>string</Name>
      <Path>string</Path>
      <Folder>boolean</Folder>
    </DocumentTreeNodeInformation>
    <DocumentTreeNodeInformation>
      <ID>int</ID>
      <Name>string</Name>
      <Path>string</Path>
      <Folder>boolean</Folder>
    </DocumentTreeNodeInformation>
  </Nodes>
</DocumentChildListResult>
```

GetOriginalFile

This method returns the file referenced by the selected document ID.

This method has the following parameters:

Parameter	Type	Description
id	int	The document ID

The DocumentFileResult object has the following properties:

Property	Type	Description
Html	string	The html version of the document, if it exists (usually for word documents)
FileBytes	byte[]	The bytes of the document file, if it exists
Extension	string	The file extension of the document file (includes leading period)

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://civicweb.net/GetOriginalFile"

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetOriginalFile xmlns="http://civicweb.net/">
      <id>int</id>
    </GetOriginalFile>
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetOriginalFileResponse xmlns="http://civicweb.net/">
      <GetOriginalFileResult>
        <Html>string</Html>
        <FileBytes>base64Binary</FileBytes>
        <Extension>string</Extension>
      </GetOriginalFileResult>
    </GetOriginalFileResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetOriginalFile xmlns="http://civicweb.net/">
      <id>int</id>
    </GetOriginalFile>
  </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetOriginalFileResponse xmlns="http://civicweb.net/">
      <GetOriginalFileResult>
        <Html>string</Html>
        <FileBytes>base64Binary</FileBytes>
        <Extension>string</Extension>
      </GetOriginalFileResult>
    </GetOriginalFileResponse>
  </soap12:Body>
</soap12:Envelope>
```

HTTP POST

The following is a sample HTTP POST request and response. The placeholders shown need to be replaced with actual values.

```
POST /Global/WebServices/Document.asmx/GetOriginalFile HTTP/1.1
Host: norfolk.civicweb.net
Content-Type: application/x-www-form-urlencoded
Content-Length: length

id=string

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<DocumentFileResult xmlns="http://civicweb.net/">
  <Html>string</Html>
  <FileBytes>base64Binary</FileBytes>
  <Extension>string</Extension>
</DocumentFileResult>
```

Code samples

.Net

The following code ensures that a client has been logged into the FilePro Connector.

```
private void EnsureCivicWebLogin()
{
        CivicWebLogin.Login loginWebService = new CivicWebLogin.Login();
        loginWebService.CookieContainer = CivicWebCookies;
        bool loginSucceeded = loginWebService.LoginUser(string.Empty, string.Empty);
}
```

CivicWebCookies is a System.Net.CookieContainer object, which stores the login cookie for the FilePro Connector Web Service. CivicWebLogin.Login is a class automatically generated by Visual Studio for interacting with the Web Service.

After this code runs the `CivicWebCookies` variable contains the login cookes for the FilePro Connector.

The `CivicWebCookies` variable should be reused to interact with the other FilePro Connector Web Services. The easiest way to do this is to store it in a Session variable.

```
using System;
using System.Collections.ObjectModel;
using System.Globalization;
using System.Net;

public class WebServiceSample
{
      private static CookieContainer cookies;
      private static string currentSessionId;

      private bool LogOn(string userName, string password)
      {
            bool success = false;
            CurrentSessionId = string.Empty;
            try
            {
                  LoginWebService.Login loginWebService = new
LoginWebService.Login() { CookieContainer = Cookies };
                  loginWebService.Url = GetWebServiceUrl(loginWebService.Url);

                  if (loginWebService.LoginUser(userName, password))
                  {
                        CurrentSessionId = Cookies.GetCookies(new
Uri("https://sample.civicweb.net"))["CurrentSession"].Value;
                  }
            }
            catch (Exception e)
            {
                  //Handle invalid connection
            }

            return success;
      }

      protected int[] GetSearchResults(string username, string password, string
exactPhrase, string allWords, string atLeastOneWord, string proximityWord1, string
proximityWord2, string notIncludingWords)
      {
            int[] searchResultDocumentIds = null;
            if (LogOn(userName, password))
            {
                  try
                  {
                        DocumentsWebService.Document documentWebService = new
DocumentsWebService.Document() { CookieContainer = Cookies };
                        documentWebService.Url =
GetWebServiceUrl(documentWebService.Url);
```

```
                                    searchResultDocumentIds =
documentWebService.GetSearchResults(exactPhrase, allWords, atLeastOneWord,
proximityWord1, proximityWord2, notIncludingWords);
                    }
                    catch
                    {
                            //Handle invalid connection
                    }
            }

            return meetings;
      }

      public string GetWebServiceUrl(string webServiceUrl)
      {
            string strWebServiceUrl = "https://sample.civicweb.net";
            int intPosition = webServiceUrl.IndexOf("//", 0,
StringComparison.InvariantCultureIgnoreCase);
            if (intPosition > 0 && webServiceUrl.Length > intPosition + 2)
            {
                    intPosition = webServiceUrl.IndexOf("/", intPosition + 2,
StringComparison.InvariantCultureIgnoreCase);
                    if (intPosition > 0)
                    {
                            strWebServiceUrl = (strWebServiceUrl.EndsWith("/") ?
strWebServiceUrl.Substring(0, strWebServiceUrl.Length - 1) : strWebServiceUrl) +
webServiceUrl.Substring(intPosition, webServiceUrl.Length - intPosition);
                    }
            }

            return strWebServiceUrl;
      }

      public static string CurrentSessionId
      {
            get
            {
                    return currentSessionId;
            }
            set
            {
                    currentSessionId = value;
            }
      }

      //This maintains the cookies for web service requests
      public static CookieContainer Cookies
      {
            get
            {
                    if (cookies == null)
                    {
                            cookies = new CookieContainer();
                    }
```

```
                    return cookies;
            }
            set
            {
                    cookies = value;
            }
        }
    }
}
```

PHP

```php
<?php
/**
 * Log on
 */
function log_on($user_name, $password) {
  $results = array(
    'success' => FALSE,
    'session_id' => '',
  );

  $civicweb_url = 'https://sample.civicweb.net/';
  if (strlen($civicweb_url) > 0) {
    try {
      $client = @new SoapClient($civicweb_url . 'Global/WebServices/Login.asmx?wsdl',
array('exceptions' => 1,));
      $parameters = new stdClass();
      $parameters->userName = $user_name;
      $parameters->password = $password;
      $retval = $client->LoginUser($parameters);
      $results['success'] = $retval->LoginUserResult;
      $results['session_id'] = $client->_cookies['CurrentSession'][0];
    }
    catch (SoapFault $E) {
      $results['success'] = FALSE;
      $results['session_id'] = '';
    }
  }

  return $results;
}

/**
 * Get Search Result Document IDs.
 */
function get_search_results($user_name, $password, $exact_phrase, $all_words,
$at_least_one_word, $proximity_word_1, $proximity_word_2, $not_including_words) {
  $logon_result = log_on($user_name, $password);
  $search_result_document_ids = array();
  if ($logon_result['success']) {
    $civicweb_url = 'https://sample.civicweb.net/';
    if (strlen($civicweb_url) > 0) {
      //Get results
      try {
```

```php
        $client = @new SoapClient($civicweb_url .
Global/WebServices/Document.asmx?wsdl', array('exceptions' => 1,));
        $client->__setCookie('CurrentSession', $logon_result['session_id']);

        $parameters = new stdClass();
        $parameters->exactPhrase = $exact_phrase;
        $parameters->allWords = $all_words;
        $parameters->atLeastOneWord = $at_least_one_word;
        $parameters->wordsInProximity1 = $proximity_word_1;
        $parameters->wordsInProximity2 = $proximity_word_2;
        $parameters->wordsNotIncluded = $not_including_words;

        $retval = $client->GetSearchResults($parameters);
        if (property_exists($retval->GetSearchResultsResult, 'int')) {
          $search_result_document_ids = $retval->GetSearchResultsResult->int;
        }
        else {
          $search_result_document_ids = array();
        }
      }
      catch (SoapFault $E) {
        $search_result_document_ids = array();
      }
    }
  }

  return $search_result_document_ids;
}
?>
```