# Identity

Constructor:

When you instantiate the library, the constructor loads the configuration that is defined in oauthconf.xml file of the environment

**Params**:

*xmlConfigPath (string):* full path to the config file (for example: C:\\myfilepath\\DLL\\oauthconf.xml)

*environment (string):* environment to load settings

public void synchronizeSessionWithServer()

This method verifies the authorization tokens (client\_token, access\_token and refresh\_token)

Also updates the web client status, storing the client\_token, access\_token and refresh tokend and login\_status in the library.

You should invoke it on each request in order to check and update the status of the user (not logged, logged or connected), and verify that every token that you are gonna use before is going to be valid.

public bool isConnected()

Check if the user is connected. A user conected is a user logged on DruID.

**return:**

*Boolean:* True is logged, False is not logged

public void authorizeUser(string code)

If a user succesfully logs-in DruID, the application, will redirect to post-login url of the web client.

In that case, the post-login URL will revieve an authorization code as a GET parameter.

Once the authorization code is provided to the web client, the SDK will send again to DruID the token\_endpoint to obtain the access\_token of the user and create the cookie

The metod is needed to authorize de user when the web client takes back the control of the browser.

public bool checkUserComplete(string scope)

This method checks if the user have completed all the fields needed for that section.

The "scope" (section) is a group of fields configured in DruID for a web client.

A section can be also defined as a "part" (section) of the website (web client) that only can be accesed by a user who have filled a set of personal information configured in DruID (all of the fields required for that section).

This method is commonly used for promotions or sweepstakes: if a user wants to participate in a promotion, the web client must ensure that the user have all the fields filled in order to let him participate.

**Params:**

*Scope (string):* section-key identifier of the web client. The section-key is located in oauthconf.xml file

**return:**

*boolean:* TRUE if the user have already completed all the fields needed for that section

public void logoutUser()

This method is needed to logout a user.

It makes

* the logout call to DruID
* clear cookies
* tokens and local data for the logged user

public User getUserLogged()

This method returns the PII of the logged user

**return**:

*User:* instance of logged user PII

public User getDataUsers(string uid, StoredToken access\_token)

Returns the User PII trough the DruID personal identifier.

**Params:**

*uid (string):* string uid DruID personal identifier

*access\_token (string):* access\_token

**return:**

*User:* instance

public bool checkLoginStatus()

Check if the user is logged-in or not

**return:**

*boolean:* true if the user is logged-in, false if not.