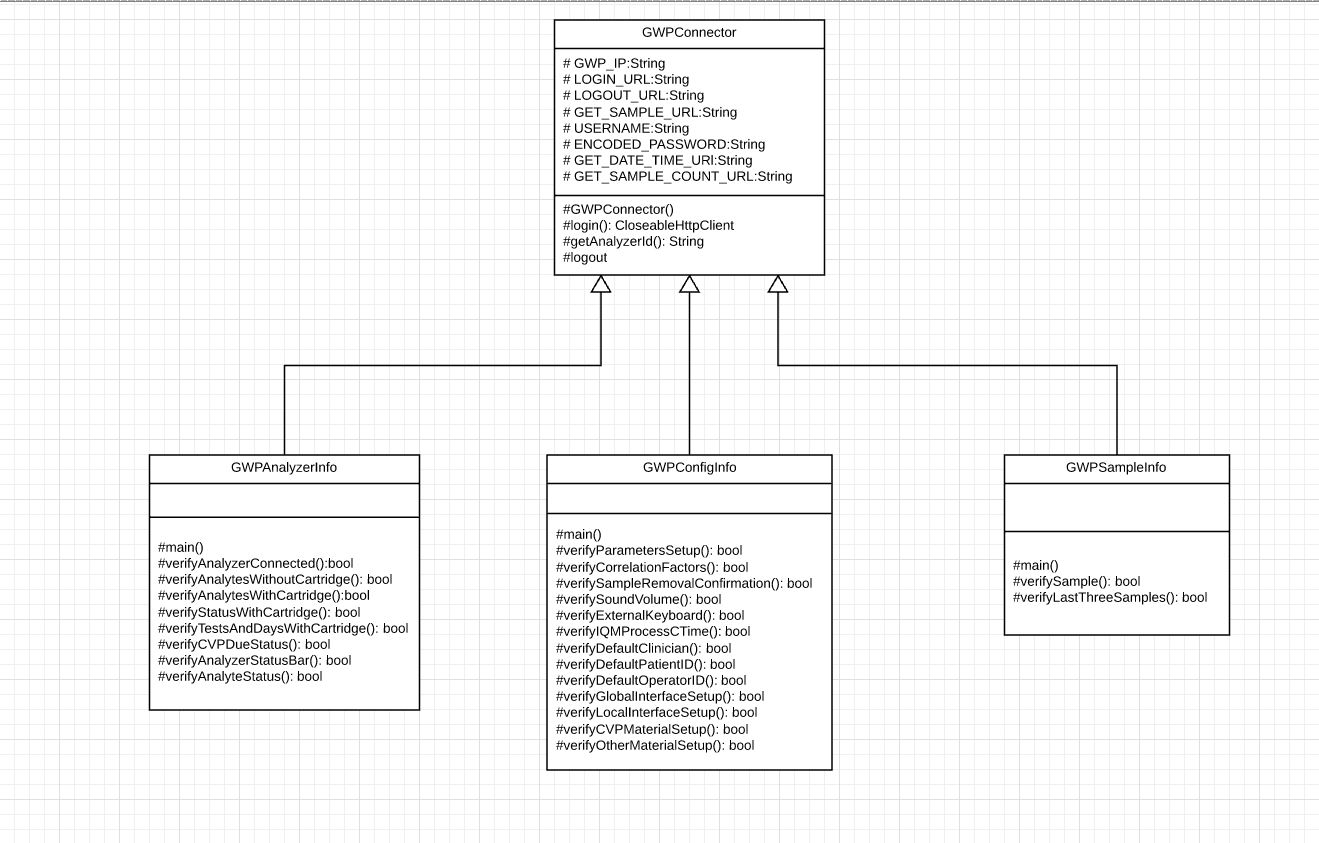
**GWP Integration**

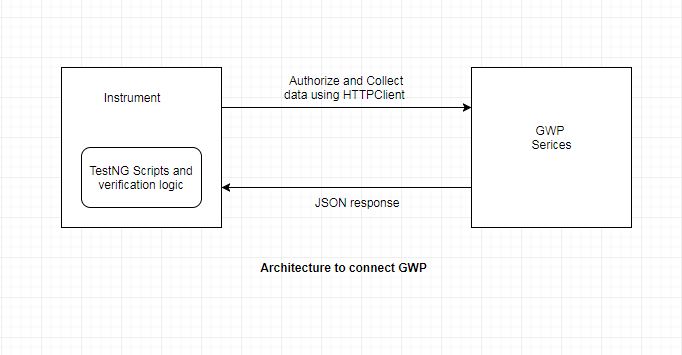
**Purpose:**

* Create a library which can be used to verify results of instrument test cases on GEMweb Plus
* Implement a library which is easy to integrate with existing TestNG Scripts
* Write a systematic function for each test case scenario of an instrument

**Class Diagram:**

****

**Architecture:**

****

**Integration with TestNG Scripts:**

* After the logic of test case, create object of GWP class specific to that test case inside TestNG function.
* Using that object call particular method related to that test case and pass required parameters of that method.
* This function will verify those parameters in GWP and return true or false.
* For Example: We want to verify if status of an analyzer connection in GWP.
* We will create an object of GWPAnalyzerInfo class.
* Then we will call verifyAnalyzerConnected(String analyzerName) method using that object and passing analyzer name.
* And will assert true or false.
* Code will look like:

GWPAnalyzerInfo connector = **new** GWPAnalyzerInfo();

**assertTrue(**connector.verifyAnalyzerConnected("gemtestpc8”));

* verifyAnalyzerConnected(String analyzerName) will login to GWP and will use its existing REST API to get status of an Analyzer.
* It will logout that connection and will return the status (true or false).

**Get API information from GWP:**

* Open GEMweb Plus in the browser. Right click on the page and click on the Inspect.
* Click on the Network tab in Developer’s Console.
* You can observe all the network calls of GWP.
* Click on any of the network call, you can observe Headers, Preview, Response, etc. of that HTTP request.
* For example: To get encoded password from request body
* Open developer console tab by clicking inspect on GWP web page.
* Go to Network tab and clear all calls.
* Now, click on sign in and fill login form.
* When you click on Sign In button of form, you can see “login” network call in network tab.
* Click on that link, you can observe detailed information of that request.
* You can observe GWP is using “https://192.168.237.241/api/login” API to authenticate user.
* In headers under Form Data you can get username and encoded password for that login.