TECH ASSIGNMENT

Setup instruction:

This setup instruction assumes that the user has already installed the below dependencies.

Dependencies:

- Java (version 7 or above)
- Maven
- IDE
- GIT client

Note: All the setup instructions have been explained with IntelliJ IDE, the options may slightly vary in other IDE platforms.

The setup can be done in two ways:

A) Importing from existing sources:

This is a maven project. The checked out git repository can be directly imported as maven project by clicking 'File -> New ->Project from Existing Sources.' as shown in Image 1.
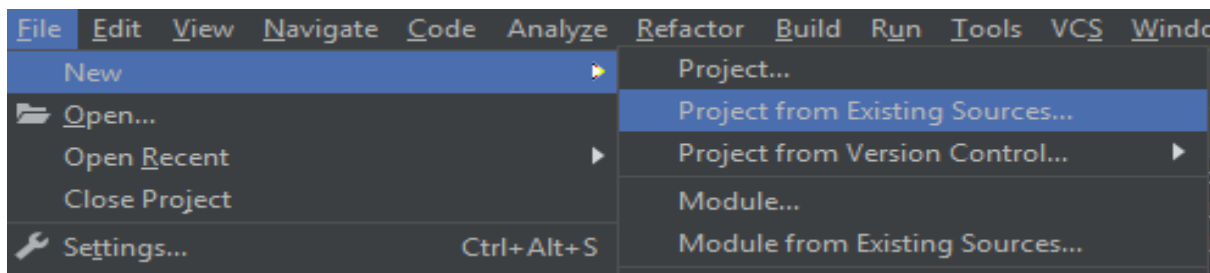

Image 1: Importing from existing sources

B) Cloning the source code:

The source code is already committed to the GitHub repository. It can be cloned locally by clicking 'File -> New -> Project from Version Control -> Git' as shown in Image2.
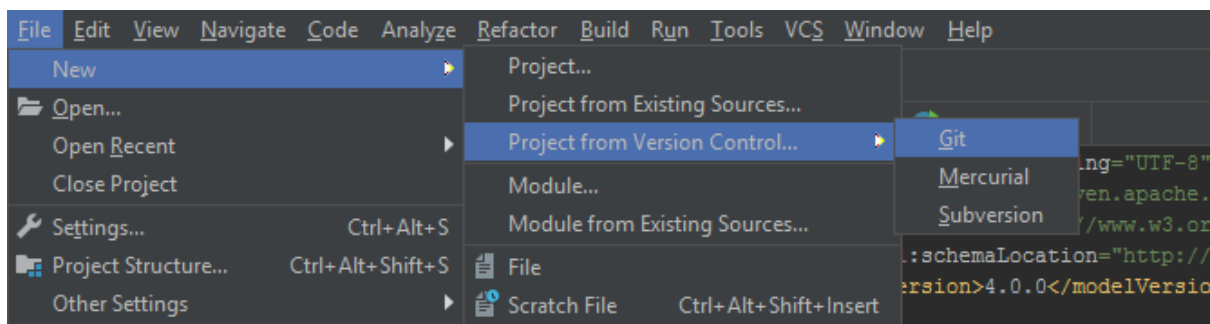

Image 2: Selecting the Version Control
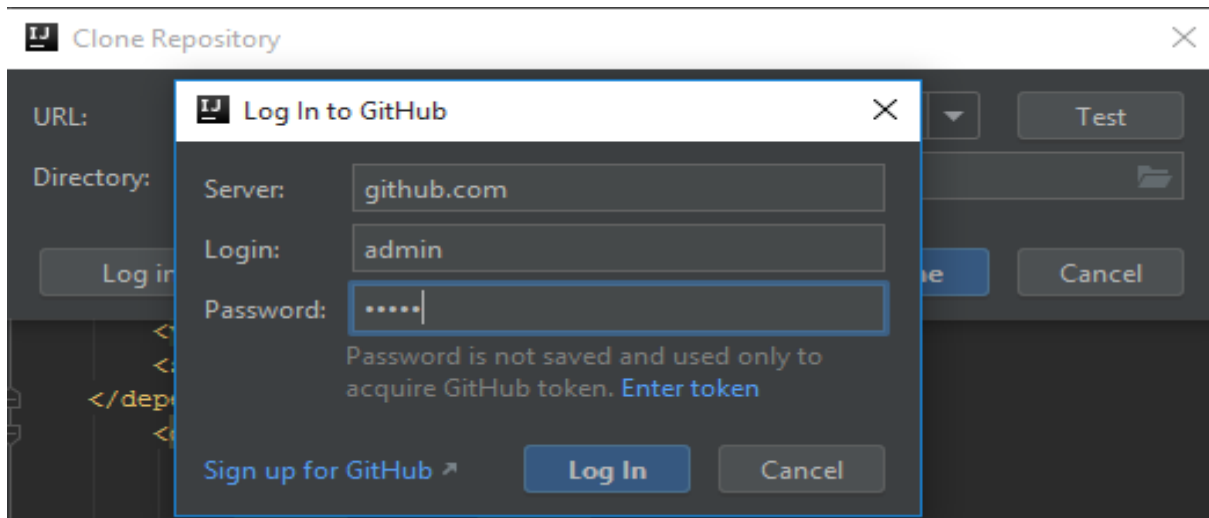
Provide the valid GitHub credentials as in Image 3.

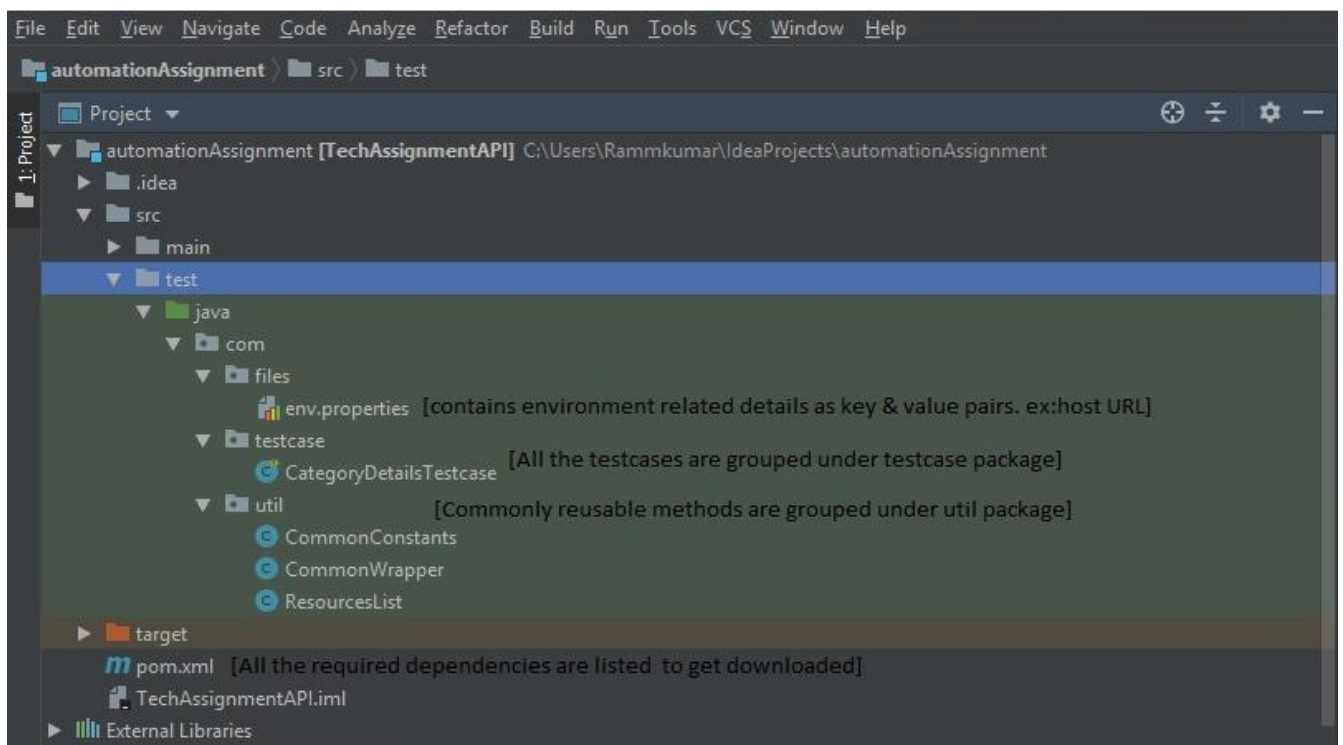Image 3: Provide GitHub credentials

Project Structure:

Image 4: Project structure

Libraries used:

- Rest Assured
- TestNG
- JSON simple
- JSON Path

How to use?

1. Open the env.properties file and enter the hostname against which the test cases needs to be executed.
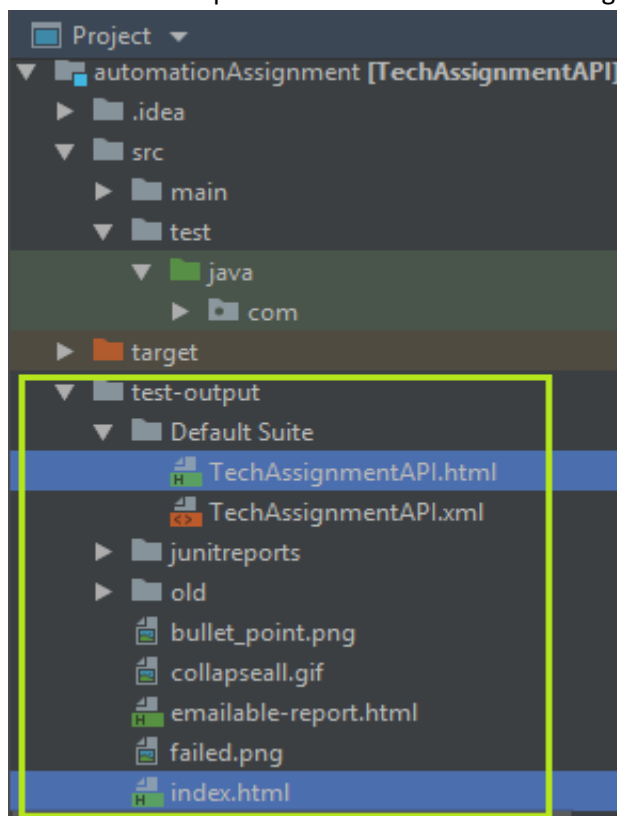
```
HOST_NAME=https://api.tmsandbox.co.nz
CATALOGUE=false
```

2. Parse the category id (ex:6327) in the get method

```java
@BeforeClass
public void getData() throws IOException
{
    File file = new File(CONFIG_PROP_FILE);
    FileInputStream fis = new FileInputStream(file);
    prop.load(fis);
    RestAssured.baseURI= prop.getProperty("HOST_NAME");

    jsonResponse = RestAssured.given().
            param( s: "catalogue", prop.getProperty("CATALOGUE")).
            when().get(ResourcesList.CategoryResourcesID(6327)).
            then().assertThat().statusCode(200).and().contentType("application/json").
            extract().response();
    jsonResult = jsonResponse.asString();
    jsp = CommonWrapper.rawToJSON(jsonResponse);

}
```

Note: This category id can be stored in an external file (ex: xlsx files) and can be reused for other calls.

3. Right click on the test case file and select 'Run'
4. The test report will be generated under test-output folder
5. The HTML test report under the Default suite will give us the test results.

```
Project ▼
▼ automationAssignment [TechAssignmentAPI]
  ▶ .idea
  ▼ src
    ▶ main
    ▼ test
      ▼ java
        ▶ com
  ▶ target
  ▼ test-output
    ▼ Default Suite
        TechAssignmentAPI.html
        TechAssignmentAPI.xml
    ▶ junitreports
    ▶ old
      bullet_point.png
      collapseall.gif
      emailable-report.html
      failed.png
      index.html
```

Note: To view the Test Reports enable checkbox "Use default reporters" in Run ->EditConfiguration ->Listeners tab

Sample screenshot of Test Report:

TechAssignmentAPI

Tests passed/Failed/Skipped:	3/0/0
Started on:	Mon Nov 04 11:41:15 NZDT 2019
Total time:	3 seconds (3754 ms)
Included groups:	
Excluded groups:	

(Hover the method name to see the test class name)

PASSED TESTS			
Test method	**Exception**	**Time (seconds)**	**Instance**
validateDescription Test class: com.testcase.CategoryDetailsTestcase Test method: AC3: Verifying whether element with Name 'Gallery' has a Description that contains the text '2x larger image'		0	com.testcase.CategoryDetailsTestcase@6a5fc7f7
validateListFlag Test class: com.testcase.CategoryDetailsTestcase Test method: AC2: Verifying the list flagging is enabled		0	com.testcase.CategoryDetailsTestcase@6a5fc7f7
validateName Test class: com.testcase.CategoryDetailsTestcase Test method: AC1: Verifying the Name		0	com.testcase.CategoryDetailsTestcase@6a5fc7f7

Credits:

Rest Assured [http://rest-assured.io/]

Stack Overflow [https://stackoverflow.com/]