**Demo App Assignment**

**Approach Followed:**

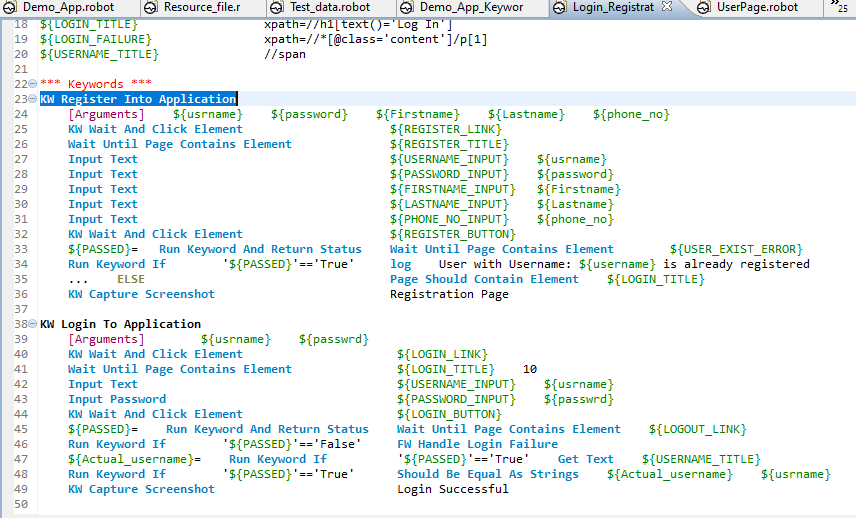
* UI Automation – Robot Framework
* API Automation – Python
* Unit test – Unit test framework

**UI Automation:**

Using robot framework, a page object mechanism is followed which helps in efficient way of storing the locators.

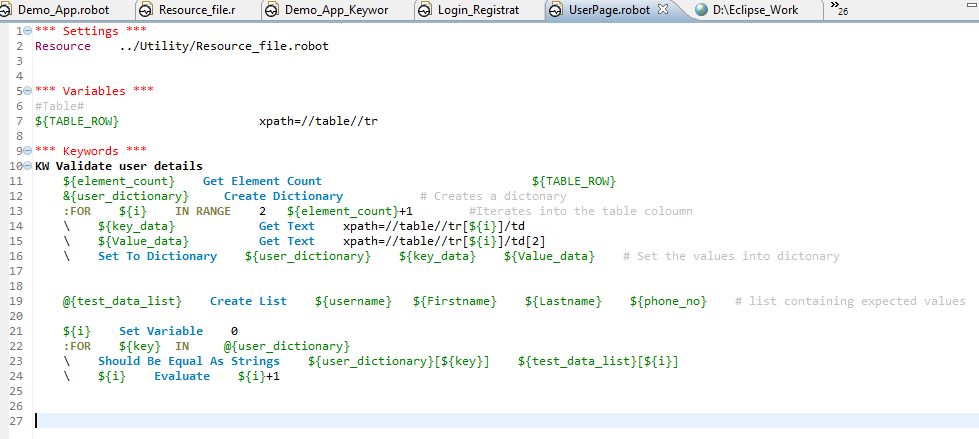
**Login\_Registration.robot :**

Keywords for registering into Demo page and logging into the application are present in this robot file. Most of the locators are **id** based as id has the highest order of precedence in accessing the locators.

****

**UserPage.robot**

Keywords for validating the user details post login are maintained in this robot file. Dictionary keyword is used to store the values from the table and compared it with the values in the list where expected values are stored.



**API Automation:**

Python has been used for API testing of the application. Post and get methods are used for validating response from the request and a session is used throughout the validation after logging in.

**Instruction to run the project**

Pre-requisite

* Python 3.7 or above
* Install libraries in requirement.txt

1. Install python and set the environment variables for python.
2. Install the requirments.txt using “**pip install -r requirments.txt**”

**To run Robot project**

* Open command prompt from the test folder in the project and run the below command

**robot Demo\_App.robot**

* “report.html” gets generate in the project folder along with screenshots in the Screenshots folder.

**To run API project**

* Open command prompt from API\_Module folder in project and run the below command

**Pytest Test\_DemoApp.py**

**FAQ**

1. How do you review code?

**I review the code by looking for the appropriate log of test cases and capturing the screenshots after every important functionality.**

1. How do you enforce coding standards?

**Every test case is made independent of each other, so if a test case fails the other test case should not be interrupted. Comments are used in appropriate place for every tester to understand the code. Proper naming convention is followed.**

1. How do you plan what kind of approach you take for test automation - what libraries to use, how does it work in couple of years, how to make it easy to maintain, etc? What are the main points to consider?

**POM approach is followed for test automation to avoid redundancy and efficient way of maintaining the locators. Since python 3.8 is used it serves the purpose of working for upcoming years and to supports backward compatibility.**

1. Code testability, how do you enforce it?

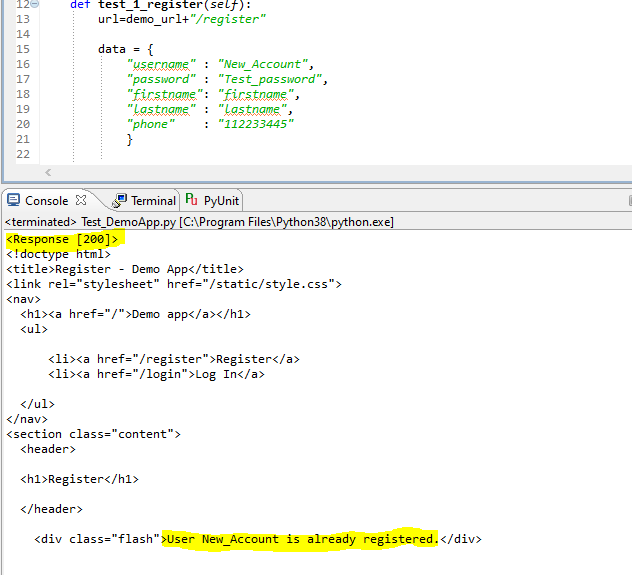
**By using appropriate assert statement code testability of every functionality is ensured**

1. How do you make sure that the product is testable?

**To be testable, a requirement must be clear, measurable, and complete, without any ambiguity**

**Report of found Issues/bugs:**

Response code for registration and login failure was **200,** which should be 400 instead.



**Tell us what improvement would you propose for the app**

Mandatory field and mobile number validation should be included in the registration page. Length and alphanumeric password validation should be included.

**How much time it took, there is no time limit as such. Remember to return the task when agreed or let us know if you run late.**

**7 Days**