Automation Testing FRAMEWOK

Introduction

Name : Hybrid Framework

Frame work is the structural way of maintaining and executing the scripts.
Hybrid framework is a way to organize the code so that it can be

- ✓ Reusable
- ✓ Scalable
- ✓ Maintainable
- ✓ Understandable
- ✓ Workable

→**Reusable:** A framework once written should be used by multiple people across same or multiple team not just for one project, but it should be used across multiple projects.
Using FW one can reduce the effort spent on 'X' component (library of code) which is already in use. Using the already available code (reusing the code) will require less or no testing since the available code is not all new and it's already in use
Hint:

- Write library which can be reused in framework, instead writing same logic in multiple files with different names.
- Already written code which is in use require less or no testing

→**Scalable:** FW's written should be scalable, means it should be used from small to bigger project

- FW's should also support multiple projects
- FW's should be scaled to multiple technologies and tools.
- Technologies can be web or windows or even service based.
- Tools can be selenium, TestNG, Maven etc within same

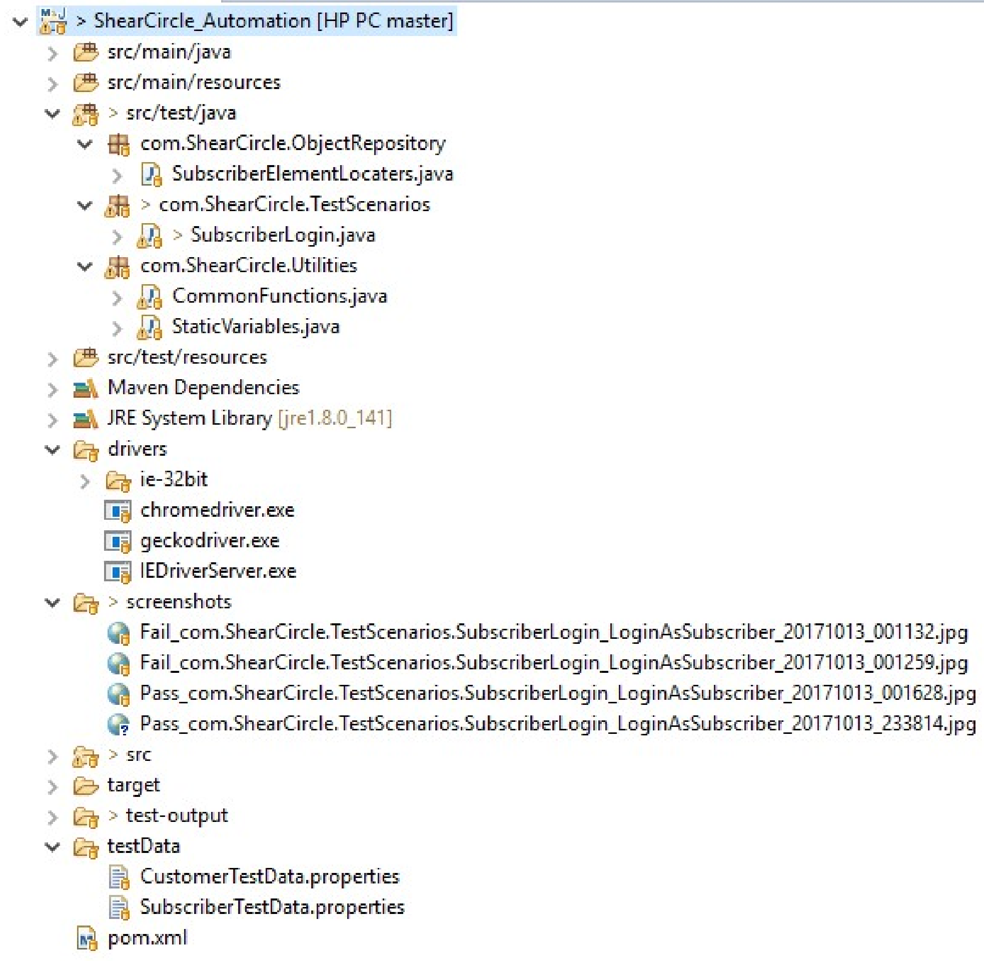
→**Maintainable:** FW should be easily maintainable, meaning the code need to be segregated as a logical groups of same type (classes) and functionalities.
Each and every code in framework should be documented, so that duplicate should will not emerge as a result lack of knowledge on existing code.
FW should be different entity from that of test project, so that changes to framework will go to framework project whereas a change to test goes to test project.

→**Workable:** FW should be usable by the team
FW should b pluggable and even less knowledge automation tet engineer should work with code using the FW methods.
It means write your FW as simple as possible.

→Establishing relationship with each page
Creating action methods within each page

- ✓ Instead of writing all the operation logic within each functionality, we can write all the operations as a method and call the method
- ✓ This will make our calling method very clean and will focus on what functionality the test method
- ✓ Write separate method for each operation which is related to that page and call them when needed.

- ✓ Instead of just returning the void type from each action method, lets return the page object itself, so that we will know how the page behaves while certain action is performed.



```
> ShearCircle_Automation [HP PC master]
    src/main/java
    src/main/resources
    > src/test/java
        com.ShearCircle.ObjectRepository
            SubscriberElementLocaters.java
        > com.ShearCircle.TestScenarios
            > SubscriberLogin.java
        com.ShearCircle.Utilities
            CommonFunctions.java
            StaticVariables.java
    src/test/resources
    Maven Dependencies
    JRE System Library [jre1.8.0_141]
    drivers
        ie-32bit
        chromedriver.exe
        geckodriver.exe
        IEDriverServer.exe
    > screenshots
        Fail_com.ShearCircle.TestScenarios.SubscriberLogin_LoginAsSubscriber_20171013_001132.jpg
        Fail_com.ShearCircle.TestScenarios.SubscriberLogin_LoginAsSubscriber_20171013_001259.jpg
        Pass_com.ShearCircle.TestScenarios.SubscriberLogin_LoginAsSubscriber_20171013_001628.jpg
        Pass_com.ShearCircle.TestScenarios.SubscriberLogin_LoginAsSubscriber_20171013_233814.jpg
    > src
    target
    > test-output
    testData
        CustomerTestData.properties
        SubscriberTestData.properties
    pom.xml
```

Explanation:

The above image is Structure of Frame work. Explaining below in detail.

Precondition: Install latest Java and Eclipse,

→ First step, in eclipse create a maven project (like: ShearCircle_Automation).

→Then, update JRE library files is up to date or not? If not add latest JRE file by using build path…

→Then, add dependencies in pom.xml.(from mvn repository website: https://mvnrepository.com/)

When we add dependencies, automatically all related jar file will be download and attach it the current project.

→Then, Create three folders(Right click on project>>New>>Folder) name it as below

- **i. drivers**
- **ii. screenshots**
- **iii. testData**

drivers folder for to store the browser supported exe files.

Screenshots folder for to store the screenshots once execution done. Whether status has pass or fail.

testData folder for to store the input(testdata). We will use ".properties" & ".xlsx" files.

→Then, go to "src/test/java" folder. Create 3 packages name it as below.

i. **com.ProjectName.ObjectRepository**
ii. **com.ProjectName.Testcases**
iii. **com.ProjectName.Utilities**

i: To store the application element locaters (Page wise or entire application wise).

ii: Develop the actual functional test scripts (as per the Business scenario's)

iii: Reusable scripts for all the projects. In addition to that, current project reusable scripts also we will implement in separate folder structure and call the existing classes into child classes.

→ In future any changes happened in locaters, will only change the respect object value from Object repository class. Don't want to modify the actual test script.

Suppose any new enhancement came, then we have to modify accordingly.

Will add any modification required in future….

Thank you.

QA-Automation Team.