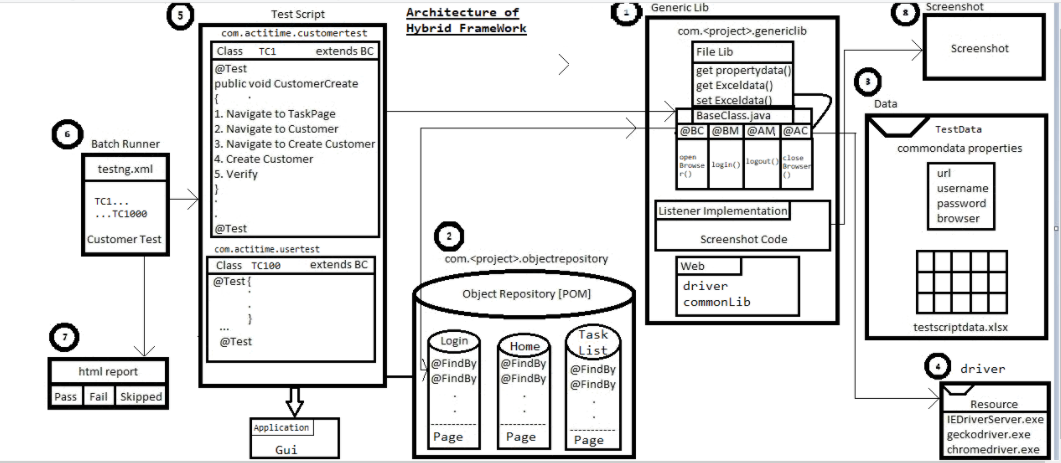
## Frame Work

Frame work is a set of rules and guidelines or best practice to be followed while automating any application.

Or Frame work is a collection of reusable components that makes automation test scripts development, execution and modification to be easier and faster.

Or Frame work is an instruction or procedure followed by every company to automate test script and achieve regression testing for every new build.

Or Frame work is a well organised structure of reusable components, where one driver script (TestNG.xml) will take care of entire batch execution without any manual interaction.



There are 8 major components available in Framework.

1. Generic Library
2. Object Repository (POM)
3. Test Data
4. Resources
5. Test Script
6. Batch Runner (TestNG.html)
7. Html report
8. ScreenShot
9. **Generics:**Generics is a library of classes,which contains reusable methdos which is used for any project. An UI framework of generic library can contains the following basic classes (interfaces also)

**FileLib :** all the methods related to reading / writing test data from an external resource are stored, it is implemented by usig Apache POI at properties class.

As per the rule of automation data should not be hard coded within a test script, so in order to read the data from excel and property files, we do have file lib class.

**Base Class**: Base class contains common annotations which is required for

all the test class like

@BeforeClass used to open the browser

@BeforeMethod used to login for the base class

@AfterMethod used to logout from the base class

@AfterClass used to close the browser

WaitForPageToLoad(), waitForElement() , select () ... etc.

**Listener implementation class**: it is implemented by using TestNG Listener feature which is used to take the screen shot whenever test is getting failed.

1. **Page Object Repository:**help us to store all the web elements locators likexpath, id, name etc.Insted of hard coding them which is aganist automation rule.in order to develop page object repository we use java design pattern such as POM, PageFactory. Page object model is a well organized design pattern where we can store all web elements at a page level (Number of web pages will be equal to number of POM classes). Make sure all the web elements available in the POM classes should be private so that test writer cannot modify the locator from their testclass.Page factory is an extended design pattern of POM which is used to initialise the element available in the POM class. PageFactory.initElements(driver,this);
2. **Test Data**: as per the rule of automation TestData should not be hard coded. Hence we maintain test data that is required to run a test case with the help of an external resource file like a) common data. Properties b) testScriptData.xlsx. CommonData.properties file contains global data which is common to all the test scripts like URL, UserName, PassWord, browseretc..
   1. URL: by changing URL we can run the test in any web server like staging, testing, production etc.
   2. UserName / PassWord: by changing credentials we can execute the test scipts in any credentials.
   3. Browser: by changing browser variable value we can run the test in any browser.
3. There are 3 types of external resources where test data can be stored
   1. Common data properties file
   2. Test data xlsx file
   3. JDBC file

This makes the readablilty , modification and maintenance of test related data easy. All the data which is required to run the test should be placed in excel file

1. Test Scriptes : Test scripts are the collections of actions to be performed on a system under test. As manual intervention should not happen in automation, we store test scripts in TestNG class for execution. All the test scripts related to a module should be stored in a single package and the package name should be <domain name>.<company name/project name><module name><test>

All the tests are automated using generic lib and object libraries. With the help of plugins like TestNG we can execute all the test scripts together.

1. TestNG: TestNGis a unit testing tool used to execute multiple test cases at a single time. Batch execution can be achieved in TestNG using TestNG.xml file where we can specify group suits, test case modules, exclusions etc.
2. Resources Jar: this components contains all the third party plugins which are required to run the tests. This may include third party tools like geko driver which is required for working with latest version of firefox. SimilarlyIEdirver Server.exe and ChromeDriver.exe which enable us to work with multiple browsers.
3. Reports: Test Reporting not only makes us aware of the status of the success or failure , but also helps in finding out the potential bugs which will also get to know the stability of the application soon after the test execution. TestNG generates the HTML reports which states the test execution results.
4. Screen Shots:in case of test cases/batch execution failure , we go for this mechanism to report where exactly the test case failed. This screen shot mechanism can be used along with test reporting for test robustness.

OR

Frame work is a set of rules and guidelines or best practice to be followed while automating any application.

1. The framework is developed using TestNG, POM and Genericlibrary(Excel and property).
2. It is a combination

of Data-Driven and Method-Driven framework, which we call as Hybrid Framework.

2. The exection is controlled by TestNG suite file which has list of TestNG classes which

are to be executed.

3. Each TestNG class has test method and also extend from base Test class which has

@BeforeClass@BeforeMethod and @AfterMethod,@AfterClass.

4. First @BeforeClass is executed which opens the browser and @BeforeMethod will be executed which contains enter the URL and login to the application.

WebDriver driver=new ChromeDriver();

5. After executing Before Method will start the execution of testmethod. The testMethod

takes the data from exel sheet and Perform the action by calling the method present in

the POM class.

Ex1:

FileLib f=new FileLib();

f.getExcelData(“SheetName”,row,cell);

Ex2:

LoginPage l=new LoginPage(driver);

l.setLogin(un,pw);

6.After executing all testmethod, it will exectesAfterMethod which contains logout and AfterClass which contains closes the browser.

driver.close();

7.After executing all the scripts it will generate the result it html format(test-output

folder) of the framework.

8.Since we implimented Listener it will take the screenshot of failel test cases.

Note :

**Modular driven Frame work (reusable methods):**When ever application contains lot of modules we prefer modular driven frame work. As per the modular driven frame work every component of the framework will be maintained by modules. Now a days modular driven frame work is not been used much because of maintanance is difficult.

**Hybrid Driven frame work:** combination of any two or three frameworks are called as Hybrid Framework. Frame works which we design is hybrid frame work because we are using data driven frame work, modular driven frame work and method driven frame work.

**Keyword Driven Frame work**: when ever we want to develop a test scipt in excel sheet, having less knowledge in selenium programming language we go for key word driven frame work. Key word driven frame work is user friendly where manual test engineers can also write test scripts. now a days no one is using key word driven frame work because of performance issue.

**Data Driven Frame Work**: when an application needs to be tested with huge amount of data or different set of data, we go for Data Driven Framework. In order to achieve data driven testing we go for property file, excel file, data base or @data provider available in TestNG. Generally Data Driven frame work is used in banking and e commerce frame work. In data driven frame work all the 8 components will be there, but every test script should have dedicated @data provider annotation and should have dedicated one excel sheet for one test case.

Data Provider: Data Provider is one of the annotation in testing which is used for Data driven testing.

@DataProvider annotation help us to provide different set of parameters(data) to run the same test case again and again.

The syntax is as shown in the given example.

## Framework Folder Structure:

