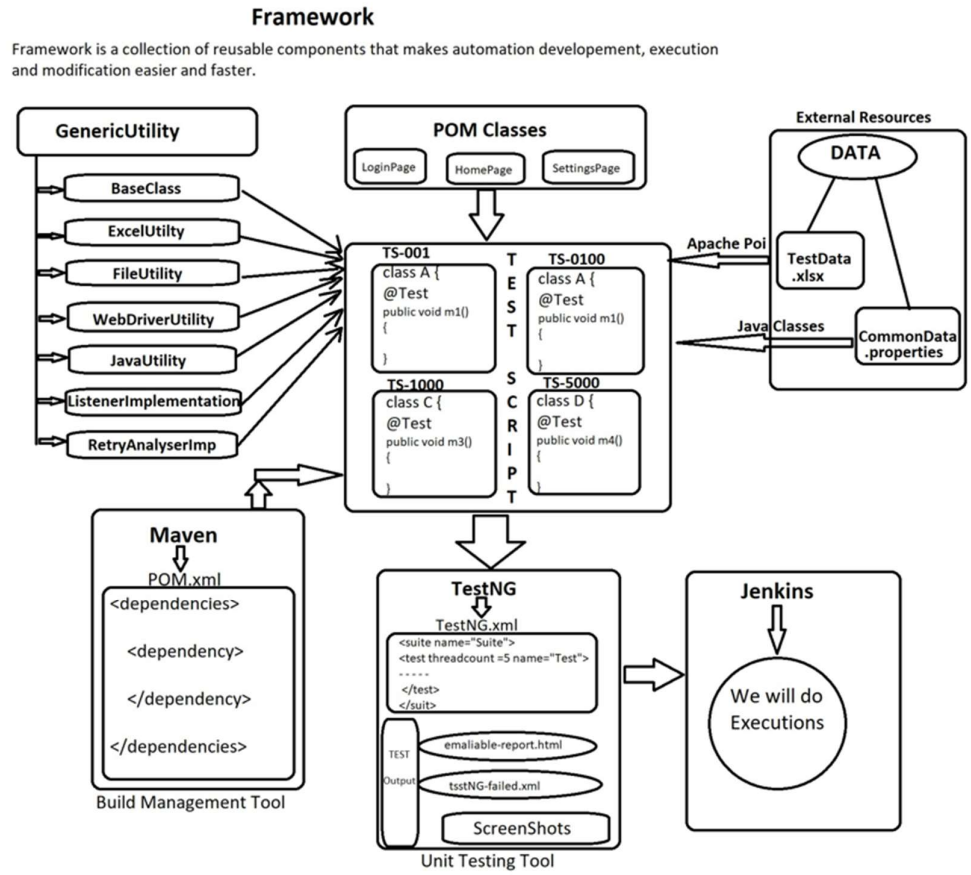
FRAMEWORK

Framework is a collection of reusable components that makes automation development, execution and modification easier and faster.

Components of Framework

1. Generic Utility
2. POM classes/Object Repository
3. External Resources (like Test Data, Common Data)
4. TestNG.xml file
5. Test Output (like reports.html)
6. Screenshots
7. Maven POM.xml file
8. Test Scripts
9. CI/CD Tools Jenkins



1 Generic Utility :- It is a component in a framework which contains multiple classes with generic methods to develop scripts very fast and easily. And the classes are like

i)BaseClass :- It is a class which contains different methods with TestNG annotations like @BeforeSuite, @BeforeTest, @BeforeClass, @BeforeMethod, @AfterMethod, @AfterClass, @AfterTest, @AfterSuite.

@BeforeSuite :- It will execute before starting of the executions and it will execute only once in entire execution. Here we configure the data base connections.

@BeforeTest :- Here we will do the parallel execution configurations.

@BeforeClass :- It will execute before execution of any classes. Here we do the configuration to launch the browser.

@BeforeMethod :- It will execute every time before execution of any @Test annotation. Here we do the configuration to do the login actions.

@AfterMethod :- It will execute every time after completing execution of any @Test annotation. Here we do the configuration to do the logout actions.

@AfterClass :- It will execute after completing execution of any classes. Here we do the configuration to close the browser.

@AfterTest :- Here we will do the configurations to close parallel execution.

@AfterSuite :- It will execute after the execution ends and it will execute only once in entire execution. Here we configure to close the data base connections.

ii)ExcelUtility :- In excel utility, it contains generic methods with arguments(sheetName, rowNumber, cellNumber) to fetch the data from excel, store the data in excel. To achieve this we have to use Apache POI external jars.

iii)FileUtility :- In file utility, it contains generic method with argument(key) to fetch the data from properties file.

iv)JavaUtility :- In java utility, it contains different methods like, to get random numbers, get date, get date and time,.. And methods are developed using java classes.

v)WebDriverUtility :- In WebDriver utility, it contains generic methods to perform some actions during execution like, take screenshot, switching to different window, selecting options from listbox, moving cursor to element,…. And this methods are developed using different classes of selenium.

vi)ListenerImplementationClass :- It is the implementation class of the ITestListener interface which contains abstract methods which will get triggered on the basis of the status(passes, failed, skipped) of the script.

vii)RetryAnalyserImplementationClass :- It is the implementation class of the IRetryAnalyser interface which contain abstract method which will get triggered if we want to rerun the failed test scripts.

2 POM Classes/Object Repository :- This is the component of framework where we will create separate pom class for each and every web page which will contain elements of that page, constructor to initialize elements through initElements method of pagefactory class, getters and setters method, some business logics.

3 Resources :- This is a framework component which contains data like test data and common data.

i)TestData :- The data which is specific to particular test script is called as test data.

ii)CommonData :- The data which is required in each and every test script is called as common data.

4 TestScript :- This is the framework component which will be developed using @Test annotation. @Test annotation is provided by TestNG and it has an internal implementation of main method.

5 TestNG.xml :- It is a the component of TestNG which is used to execute the test scripts in batch, parallel, groups, cross browser and we can also provide listener implementation in TestNG.xml file to monitor the execution of the scripts.

6 HTML report :- It is also the component of TestNG which is generated after every execution in test output folder in emailable-report.html format which contains passed/failed/skipped status of the executed script.

7 Screenshot :- It is generated whenever our script got failed during execution by using listeners implementation. It is used as a reference for debugging and if there is no mistake in the script we can add the screenshot as a proof while logging the bug.

8 POM.xml :- It is the component of maven where we add the dependencies of external jars and tools. By using pom.xml we can avoid manually downloading and adding the jars and drivers in the framework every time after new updates.