ExtentReporterNG

testiNG.xml

TestUtilsClass

Config.properties

Test RunnerClass

Extends

Extends

Analysts\_HomePg.feature

Analysts\_Career\_Pg.feature

Feature File

Test Layer/StepDefenition

Page Layer

Career Page

Home Page

AnalystCareerPgTest

AnalystHomePgTest

TestBaseClass

Call functions using oops concept

**Framework Approach: Page object Model**

**Tools/Plugin**: **Selenium WebDriver, TestNG Cucumber ,Natural, Maven**

**Folder Structure**

All the java packages are created under src/main/java

Package : com.qa.base contains : TestBaseClass.java

This Java class contains initialization of WebDriver, Pageload timeput and implicit wait utilities. Apart from this it also contains common functions like getting Curent url and page title.

Package : com.qa.Pages contains HomePage.java , CareerPage.java

Both files are used for writing specific function related to their respective page.

They are extending to testBaseClass.java

Package : com.qa.StepDefinition contains AnalystHomePgTest .java , AnalystCareerPgTest.java

Both files are used to automate respective test steps as per feature files

These java classes are accessing functions from respective page classes as well as TestBaseClass

They are extending to testBaseClass.java

Both files contain function to get screenshot for failed scenario. At this point of time this is hardcoded. Having said that common function can be written to append screenshot for all failed scenario.

Package : com.qa.Utils contains

ExtentReporterNG.java used for creating Extent Report

TestUtilClass : Used for utility function like Wait, Pageload timeout tec. Functions used by TestCaseClass.java

config.properties : contains data like page tilte , url browser information etc. used by all functions to load data at runtime.

Package : com.qa.Features contains

Analysts\_HomePg.feature

Analysts\_Career\_Pg.feature

Both feature files contains respective test Scenario for respective page.

Gherkin language is used here to write down test cases.

**Package :com.qa.TestRunner** contains cucumberOptions, TestNGCucumberRunner APIs for Execution

This files contains path of feature files and Step Definition as well as different reporting options.

Pom.xml : All dependencies and plug-in would be saved here.

**TestNg.xml** : contains listener for extent report and class name with path for TestRunnerClass.

Reporting folder : You can check various report under below folders.

Target folder, Test-output folder, cucumber-report

e.g

Analyst International Web Application Test.html

Extent.Html

This Framework is callable to do below functions:

1. We can Run project as TestNG Suite
2. This will invoke TestRunner class
3. TestRunnerclass read files from steps definition and execute scenarios.
4. TestBaseclass is fetching function/data from Config.properties and TestUtils
5. StepsDefinition classes will use functions from TestBase class,Page Class and data from Config.property to execute test steps written in feature files. These file will also generate screenshot for failed scenario.
6. ExtentReporterNG lass will listen all execution and create Extent.html Report
7. All required dependencies present in Pom.xml which provide jar files for respective APIs.

This Framework needs some enhancement like integration of excel report for test result , integration of screen shot for failed test cases and passing test data from excel sheet.