**Framework Understanding Document**

JDK : 16.0

Build Tool : Maven

**Folder Structure**

**apiPOJO** : To create and save POJO classes for APIs.

**Utilities** : To create generic setups, that can be used irrespective of the application:

* DriverSetUp : Singleton class that returns WebDriver instance to be used in different Page object model classes.

ReportSetUp : Singleton class that return ExtentTest object to be used in all the main test scripts.

* Helpers : Contains static methods that are used repeatedly like methods to resd data from excel, from database, from json files etc.
* RestUtilities : Contains genric methods for multiple API requests like get, post, put etc, that can be customized according to the product/application needs.
* RetryListener : Implements IRetryListener and IAnnotationTransformer for reteying any failed test case multiple times.
* CommonActionsUI : Contains a list of generic actions performed on UI i.e. click, enter text, dynamic waits etc.

**appSpecificLibrary** : To create Page Objects for the application and customize generic api functions from Utilities according to the application.

* BBCWeatherPageObjects : Object Repository for BBC UI Page.
* BBCWeatherPageMethods : Methods for BBC UI Page.
* APIWrapper : Customized functions to use generic RestUtilities from Helpers.

**testScripts** : Main testing class for execution.

**Src/main/resources/TestData** : Input Data for parameterization.

**Files** :

* Testing.xml : For triggering the execution, specifying listener class and other dependencies.
* Pom.xml : Contains dependencies for building and compiling the code, surefireplugin for triggering testing file.
* Config.properties : Generic configuration parameters like path to input source, DB details, browser details etc.
* Reports.html : File generated in demo Folder after execution due to implementation of extent reports.

**Guidelines to execute :**

1. Change the value of required parameters, like browser, browser driver path, input data path, api base url etc. in config.properties file.
2. Execution can be triggered either through pom.xml or testing.xml as required. In case executing using an IDE, executioncan be triggered through “EntryTest” class in testScripts folder.