**Ninad Pantane - NatWest Java API Assignment**

Automation Framework Guideline Document -

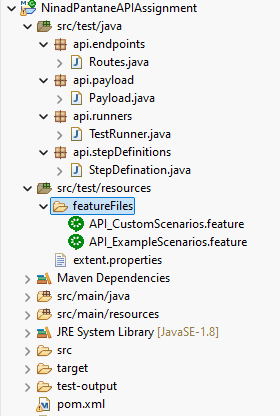
1. Maven Project with all the required dependencies added in pom.xml file

* RestAssured
* Cucumber
* Junit
* Extent Report

1. BDD Structure Implemented with 2 Feature Files

* API\_ExampleScenarios.feature = 3 Scenarios
* API\_CustomScenarios.feature = 5 Scenarios

1. Well Defined Framework Design Structure Implemented



4) All the Endpoints are kept in one File -> Routes

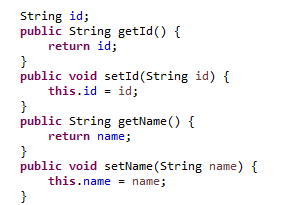
POST: https://api.restful-api.dev/objects

GET: https://api.restful-api.dev/objects/{id}

GET: https://api.restful-api.dev/objects

DELETE: https://api.restful-api.dev/objects/{id}

5) POJO Class Created for all required attributes for API are “Getters” and “Setters“ for ‘id’, ‘name’ etc.

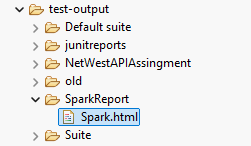


6) Test Runner File with @RunWith and @CucumberOptions, mapping Feature File with Step Definition File.

7) Extent Report Implemented “aventstack.extentreports.cucumber.adapter.ExtentCucumberAdapter”

8) BDD Hooks implemented as @Before and @After :: No Code Implemented -> Just Added Loggers for Extent Report

9) Property File “extent.properties” added at Project Root Level for generating Spark Report as test-output



10) Feature File “API\_ExampleScenarios.feature” has exactly same 3 scenarios automated as given the Problem Statement

Scenario 1: Verify an item can be created

Scenario 2: Ability to return an item

Scenario 3: Ability to list multiple items

11) Feature File “API\_CustomScenarios.feature” includes custom test cases such as verifying Success and Error Message for multiple API calls, extracting values from JSON Response for response validation –

Total 5 Test Scenarios implemented in Custom Scenario File

NOTE below all the points were considered while developing the framework

* Automate a set of scenarios that will include multiple rest calls -> **DONE**
* Be able to share information between rest calls in a given test case -> **DONE**
* Demonstrate ability to design and implement effective error and edge cases -> **DONE**
* Utilise BBD (cucumber) when automating the tests -> **DONE**
* Demonstrate ability to design good assertions ideally utilising JSON Path -> **DONE**
* Demonstrate ability to go beyond the given set of scenarios with justification -> **DONE**
* Follow clean code practices such as good naming and method design -> **DONE**
* Provide the solution using Git (Can also be provided via email) -> **DONE**
* Demonstrate ability to design and implement setup and clean-up steps -> **DONE**

Project Hosted on GIT –

Git URL – <https://github.com/NinadPantane/NatWestAPIAssingment>

Report Execution Screenshot –

