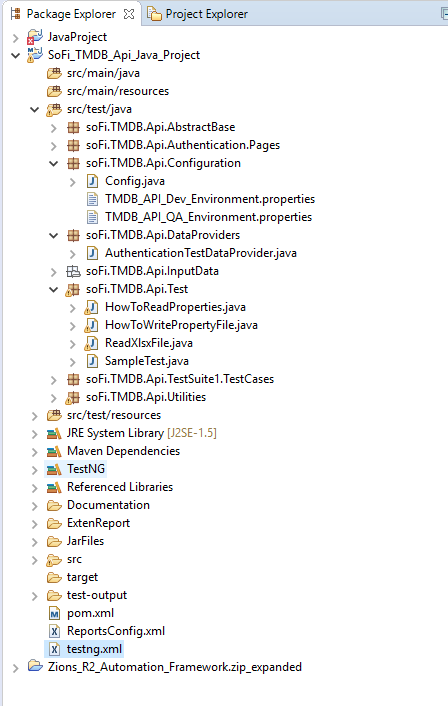
Installation Configuration and Run Project Manual

Step1: Configuration for Personal Environment



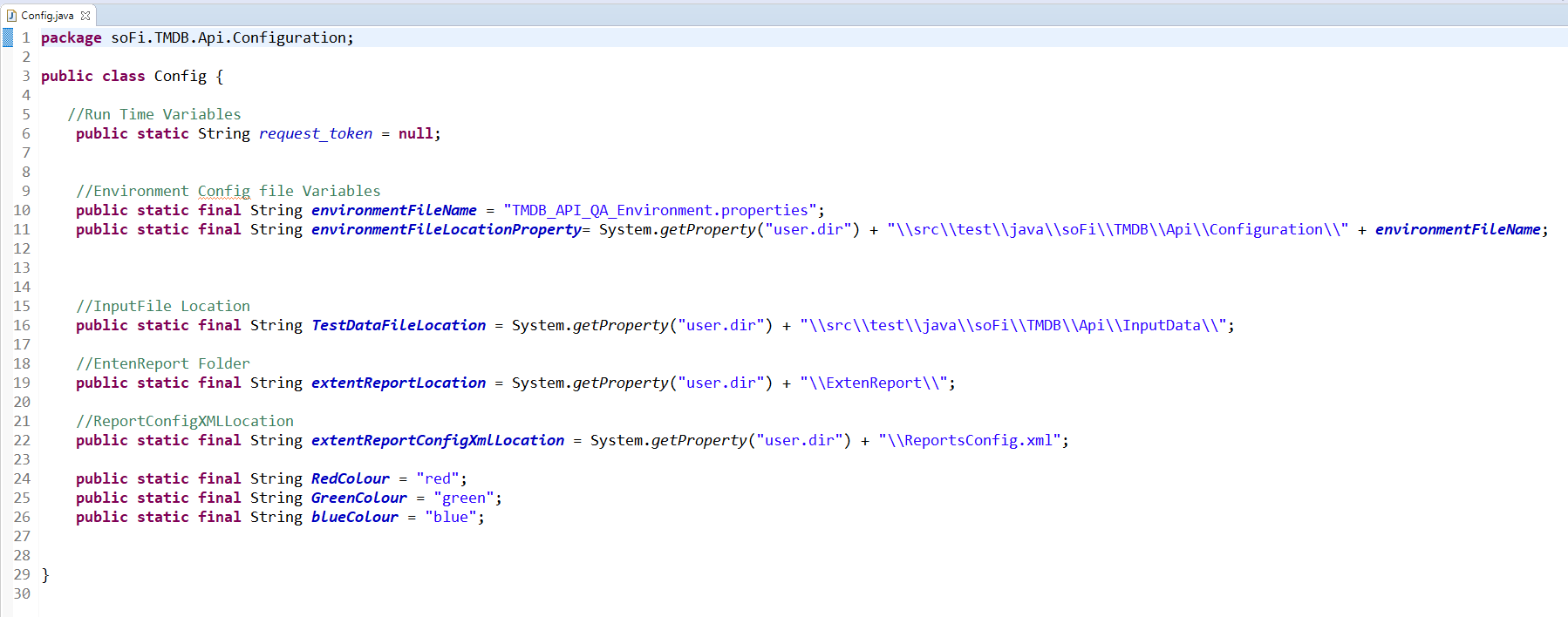
Configuration Package:

It includes three files:

Config.java: Here we have stored some project related variables. Like paths configuration for extent report, input file, and Name of Environment File (We are going to have a different Environment File to run the project on different end point, database etc).

Image of Config file.

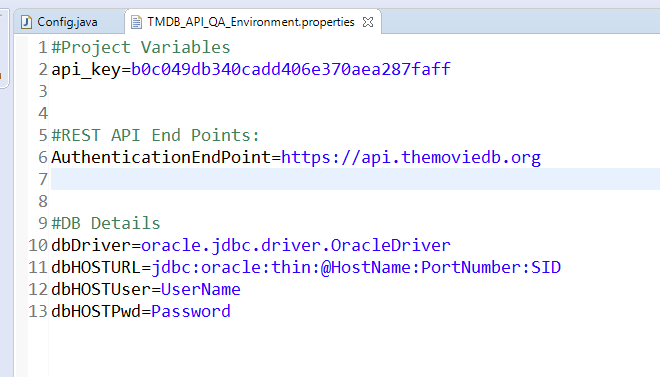
**Don’t change anything in this file except the “environmentFileName” unless you don’t want to run your project on your different Environment, That File name will decide on which environment you are going to run your project.** I have set it as “TMDB\_API\_QA\_Environment.properties” to run my project on QA Environment. You can set it as “TMDB\_API\_DEV\_Environment.properties” if you want to run the project on Dev environment**.**



TMDB\_API\_QA\_Environment.proprties : file, That will have values of the properties as per QA Environment.

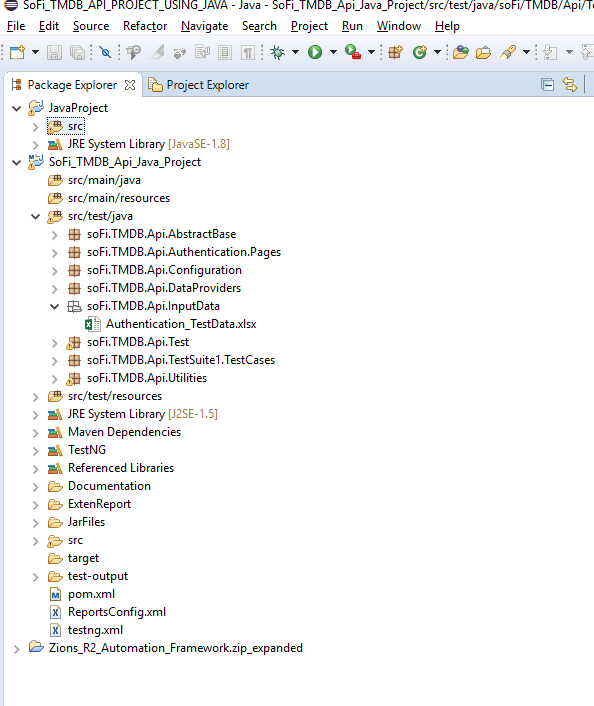
Here you change your api\_key value with your api key.

\*Don’t change key name, only values can be changed as per environment.

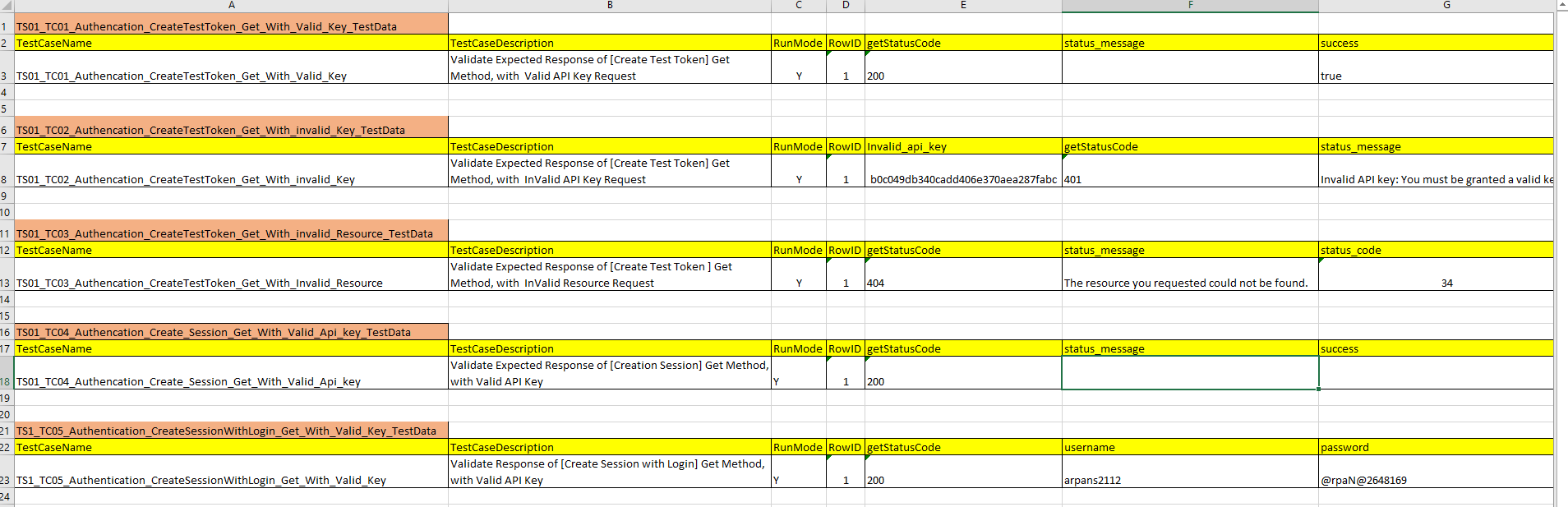


ALL SET We are Ready To Run over Project.

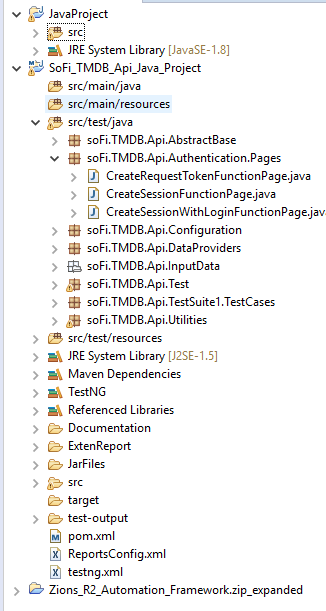
**Input File Location:** Highlighted in black box, Input data, Expected Data are set in this file.



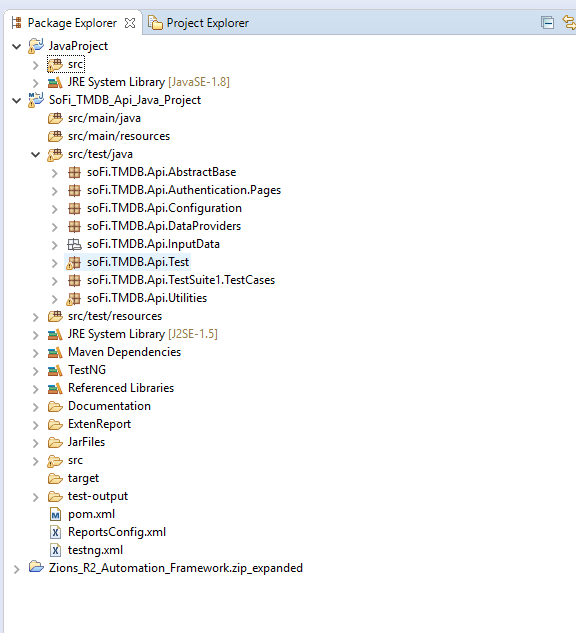
This is How input File Looks:



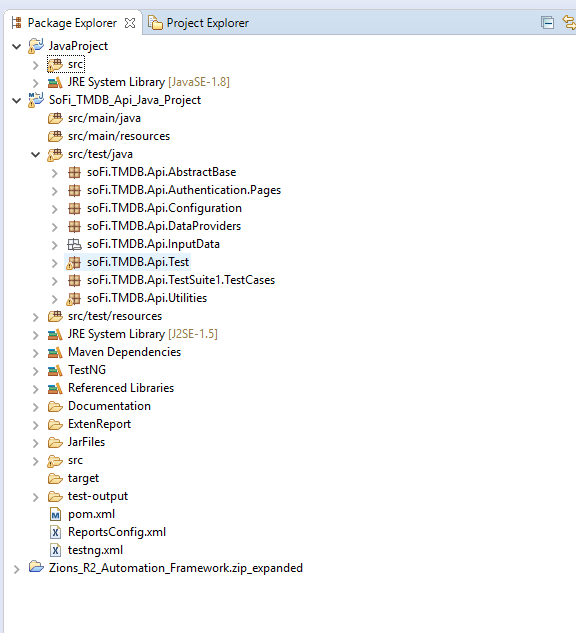
**JSON Paths: Here we will have the individual files for Each function to Maintain the JSON Paths of the data we want to Retrieve. If tomorrow something changes we can easily Maintain. [Easy Maintainability]**



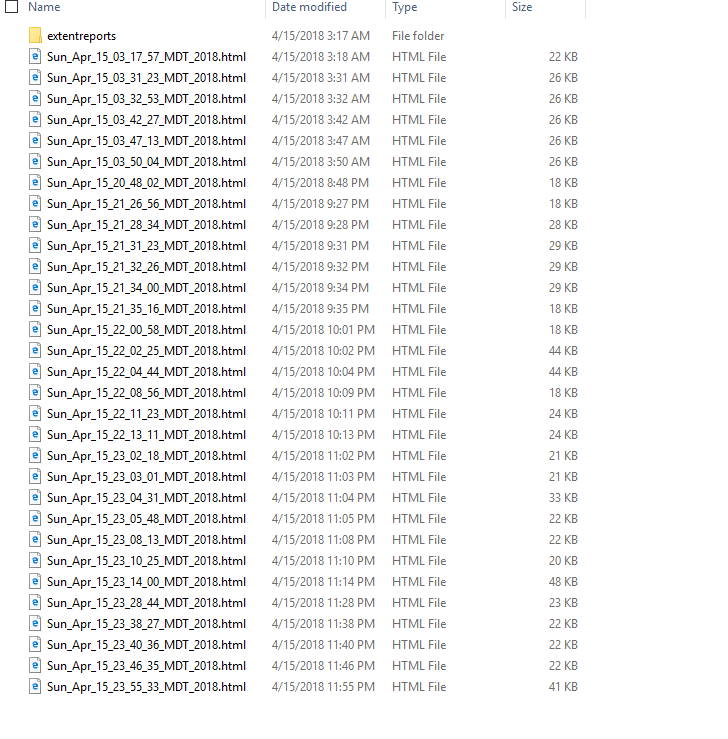
Run Project: Right Click on the TestNG.xml and Run as TestNG



Status Report in Extent Reporting: Report will be generated in the Folder Extent Report. Check with the Most Recent html File after you run your project as Run as TestNG.



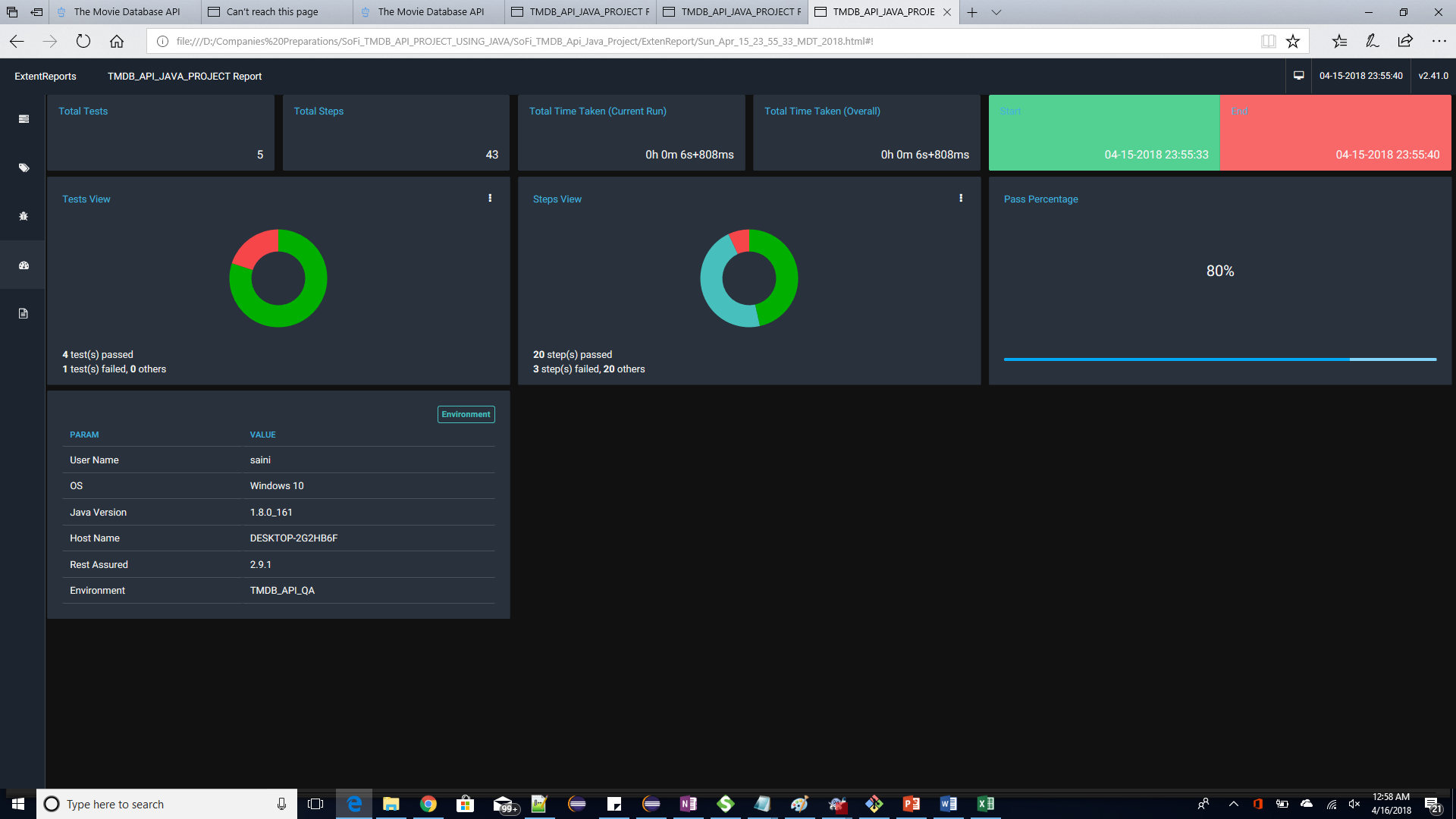
In My Case Most Recent html Report is highlighted below. Open with Chrome.



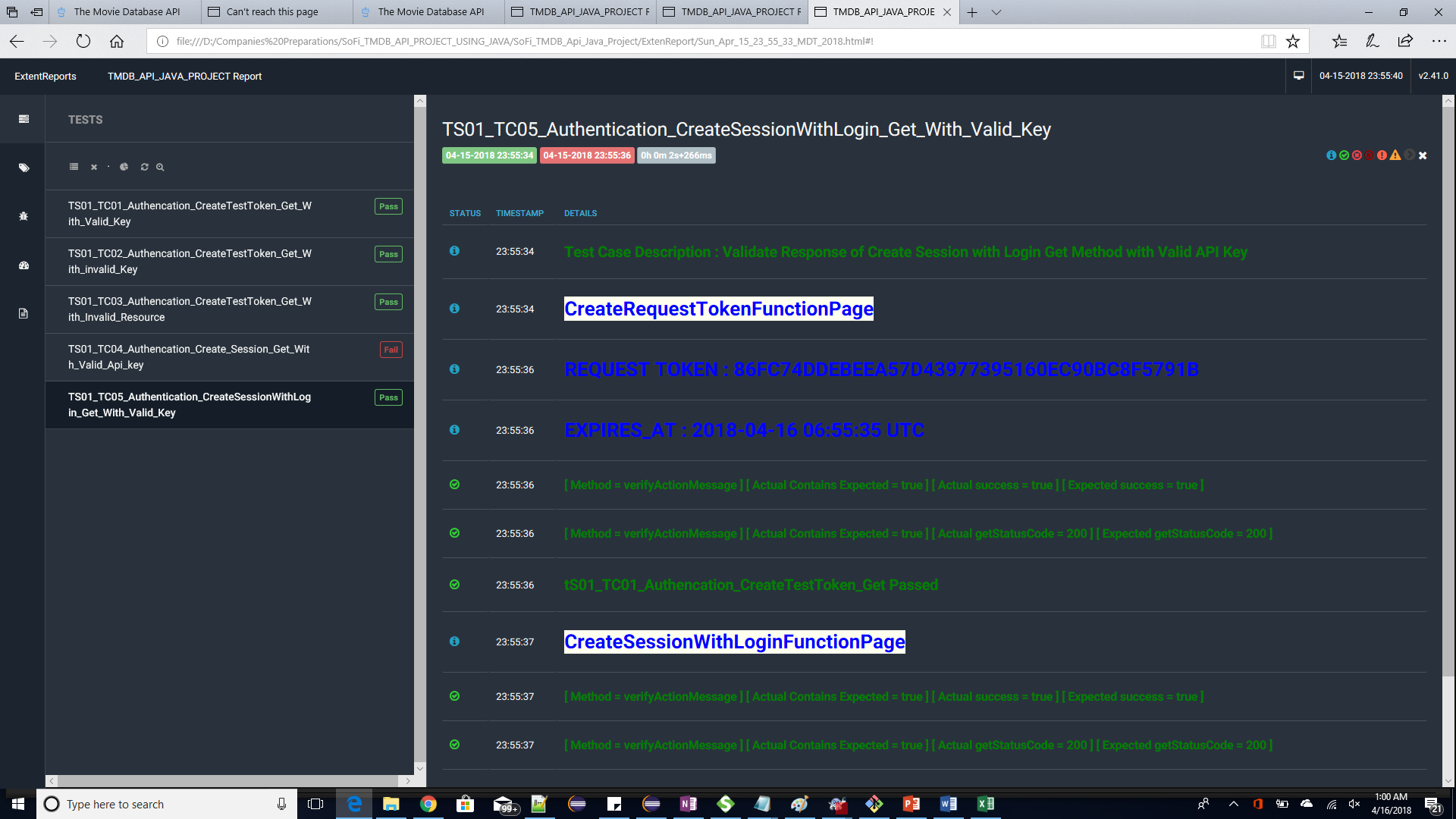
Extent Report looks Like:

Here you can see the over all status of the Execution. Like

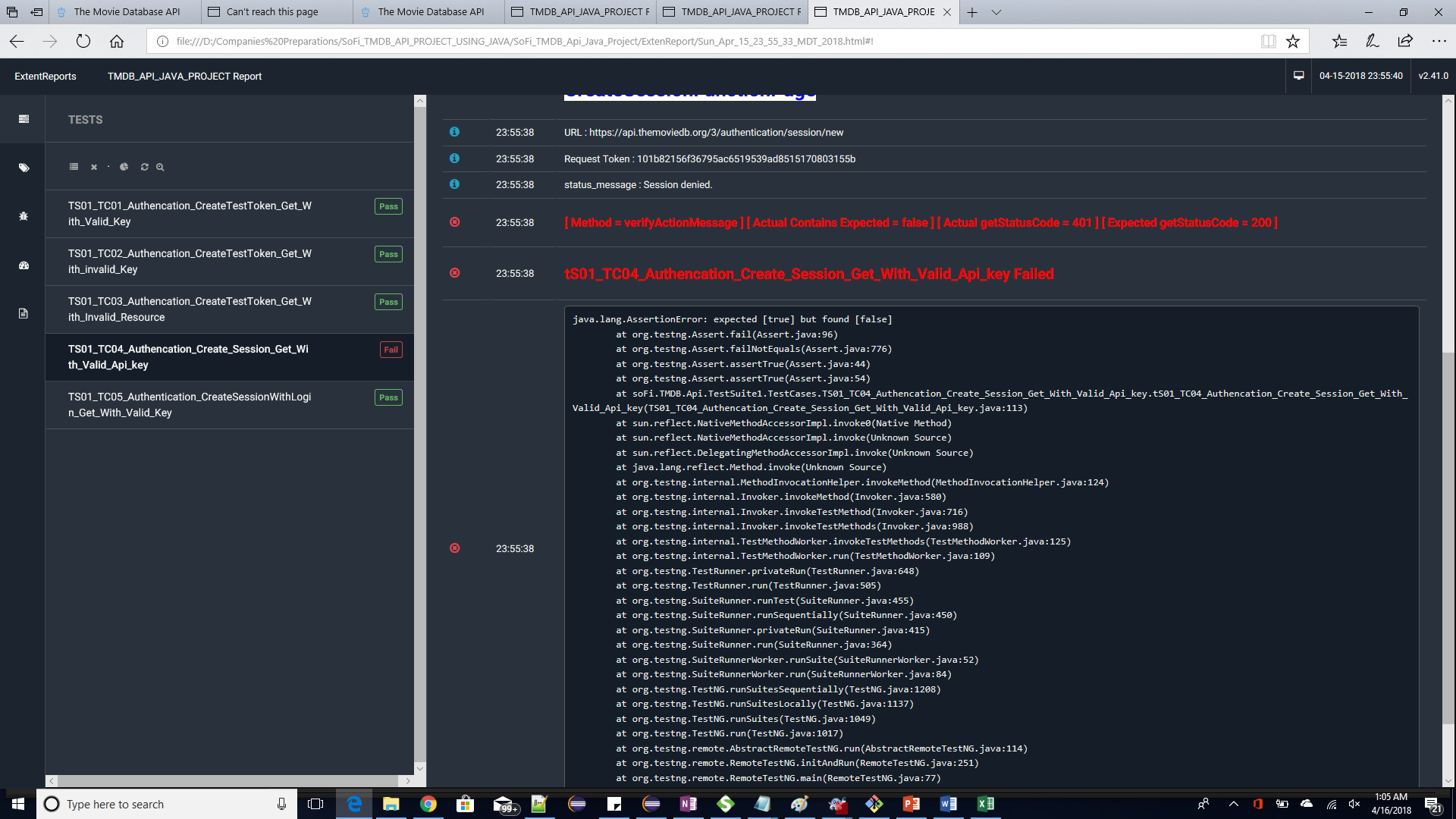
1. Total no of Test Case Run, How many are passed, How many are failed, Total time taken, Over All percentage, and The Environment Details. Those can be configurable can be changed later



**Here You can see the Details of each Step of Individual Test Case:** Assertion validation, passed or Failed and More details you need for analysis.



Here we can see the Reason of Failure of Test Case: Got **“Session denied”** and **Http response code as 401** instead of 200. So, Test Case got failed and Marked as Failed





\*\*\*\*\*\*\*\*\*\*\*\*\*End of Document\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*