

Cucumber Report

10-Feb-2025, 9:12:56 pm

Start : Feb 10, 9:12:54.089 pm

End : Feb 10, 9:12:55.531 pm

Duration : 1.442 s

Features

Scenarios

Steps

PASSED - 1
FAILED - 0
SKIPPED - 0

PASSED - 1
FAILED - 0
SKIPPED - 0

PASSED - 7
FAILED - 0
SKIPPED - 0



Feature		Scenario				Step			
Name	Duration	T	P	F	S	T	P	F	S
<u>This feature will test/verify basic functionalities of REST API [GET, POST, PUT & DELETE] using RESTAssured library.</u>	1.442 s	1	1	0	0	7	7	0	0

TAG	Scenario				Feature			
Name	<i>T</i>	<i>P</i>	<i>F</i>	<i>S</i>	<i>T</i>	<i>P</i>	<i>F</i>	<i>S</i>
@api-test2	1	1	0	0	1	1	0	0
@post-method	1	1	0	0	1	1	0	0





#	Feature Name	T	P	F	S	Duration
1	<u>This feature will test/verify basic functionalities of REST API [GET, POST, PUT & DELETE] using RESTAssured library.</u>	1	1	0	0	1.442 s



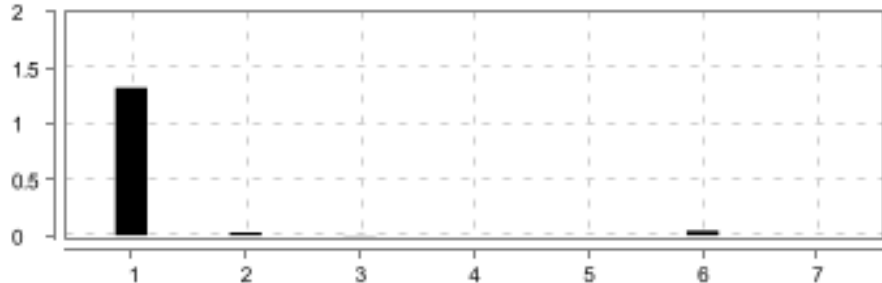

#	Feature Name	Scenario Name	T	P	F	S	Duration
1	<u>This feature will test/verify basic functionalities of REST API [GET, *</u>	<u>POST Method Test - create new user using POST method</u>	7	7	0	0	1.434 s

* The feature name and/or scenario name has been cropped to fit in the available space.

This feature will test/verify basic functionalities of REST API [GET, POST, PUT & DELETE] using RESTAssured library.

PASSED	DURATION - 1.442 s	Scenarios		Steps	
/ 9:12:54.089 pm // 9:12:55.531 pm /		Total - 1		Total - 7	
		Pass - 1		Pass - 7	
		Fail - 0		Fail - 0	
		Skip - 0		Skip - 0	

POST Method Test - create new user using POST method

<div>PASSED</div> <div>DURATION - 1.434 s</div>	<div></div>	<div>Steps</div> <div>Total - 7</div> <div>Pass - 7</div> <div>Fail - 0</div> <div>Skip - 0</div>	<div></div>
<div>/ 9:12:54.097 pm // 9:12:55.531 pm /</div>			
<div>This feature will test/verify basic functionalities of REST API [GET, POST, PUT & DELETE] using RESTAssu *</div>			
<div>@api-test2 @post-method</div>			

#	Step / Hook Details	Status	Duration
1	Given API server is up & running on "http://localhost:3000"	PASSED	1.324 s
2	When user access the endpoint "/users"	PASSED	0.026 s
3	Then user should get status as "200"	PASSED	0.002 s
4	And user should receive data with header content-type as "application/json"	PASSED	0.001 s
5	When user create and reads the payload "<payload>"	PASSED	0.001 s
6	And attach it to the request body and post	PASSED	0.040 s
7	Then user should get status as "201"	PASSED	0.001 s