Script #: 7.0 POST /jmx/registration & /jmx/metricInfo

Scenario: verb POST, HOST value http://localhost:8080/rest

Path Value: /jmx/registration

Purpose: Create Registration in Monitoring and Alerting

The POST request is asynchronous and no response is returned. If an error occurs before the asynchronous request is sent, the error returns to the caller.

Version Control

Version #	Date	Author	Description
1.0	03/01/2013	Ryan Marinello	Initial Draft
1.1	April, 2013	Ryan Marinello	Updates, Review

Test Scripts

The following scripts cover this scenario:

7.0 POST /jmx/registration Content-Type: application/json and Encoded Payload Test Requirements {"component":"component","server":"localhost:8080-MnA",

"node":"node1","description":"test1","metricInfos":[{"name":"invokeCount","description":"Numb er of times test1 has been invoked",

"category":"UTILIZATION","displayName":"Invocation Count",

"intervalPeriodInSeconds":60},{"name":"Availability",

"description":"Is test1 component up and running?","category":"AVAILABILITY",

"displayName":"Availability of Component test1","intervalPeriodInSeconds":60,"defaultValue":0}

This test scenario covers the following high-level test requirements (see scripts below for specific requirements covered by each test script):

PRMA.5

RADMA.1.1

RFP.88.1.4

NFRMA.6

RADMA.5

RADMA.5.1

User Groups

Technical Users

Script Description

This script tests the URI /jmx/registration via POST

The status result is 201 Created and the JSON response is correctly constructed with required metadata.

Setup

Monitoring and Alerting is operational and MongoDB is connected

QA staff must have Rest Client available to perform test. It is also possible to see test results in the custom MnA UI.

QA staff has either a Local or Cloud environment deployed for testing to interact with UI.

Teardown

- None

Script Steps

Step #	Test Action	Expected Results
1	Access the Rest Client verb POST	REST functional
2	Enter HOST value	http://localhost:8080/jmx
3	Enter PATH value	/jmx/registration
4	Enums	categoryType: "AVAILABILITY", "PERFORMANCE", "THROUGHPUT", "UTILIZATION"
5	Encoded Payload {"component":"test item bank", "server":"localhost:8080-MnA", "node":"node1", "description":"test1", "metricInfos":[{"name":"invokeCount", "description":"Number of times test1 has been invoked", "category":"UTILIZATION", "displayName":"Invocation Count", "intervalPeriodInSeconds":60}, {"name":"Availability", "description":"Is test1 component up and running?", "category":"AVAILABILITY", "displayName":"Availability of Component test1", "intervalPeriodInSeconds":60,	THROUGHPUT UTILIZATION AVAILABILITY PERFORMANCE

	"defaultValue":0}] }	
6	Content-Type:	application/json
7	Click Send	
8	Validation: POST empty payload, {}	requires at minimum: one non-null element in MetricInfo array (see Script24.0)
9	DB collection search	Stored data. It is available in 'registration' DB collection
10	DB Date Time	ISODate and time stamp in DB
11		successful status of 201 Created and a Response returns
12	Search DB, UI, API endpoint	metadata available in a DB 'registration' collection, matches UI and API endpoint structure
	Logic - POST metric THEN	check internal cache for already registered metrice IF YES then continue on, IF not registerd POST metricInfo
13	Get by {id}	Searchable {id} or {alertnateKey}. API endpoint UI has matching data IF not registered POST a metricInfo to JMX ObjectName
14	Get by {alternateKey}	
15	**View M&A registration page = state**	Registration will be displayed on UI page View Registrations
16	resources	Registration will be displayed on UI page View Registrations Metadata for item will be displayed in Response
17	View Response from REST client resource	Registration will be displayed on UI page View Registrations Metadata for item will be displayed in Response Metadata for item will be displayed in Response DB registration collection holds metadata
18 19	View Response from REST client resource View DB 'registration' collection	Metadata for item will be displayed in Response DB registration collection holds metadata
20	Requirements	PRMA.5, RADMA.1.1, RFP.88.1.4, NFRMA.6, RADMA.5, RADMA.5.1

Script #: 7.0 POST /jmx/registration & /jmx/metricInfo

Scenario: verb POST, HOST value http://localhost:8080/rest

Path Value: /jmx/metricInfo

Purpose: Add metricInfo to Monitoring and Alerting

The POST request is asynchronous and no response is returned. If an error occurs before the asynchronous request is sent, the error returns to the caller.

Version Control

Version #	Date	Author	Description
1.0	03/01/2013	Ryan Marinello	Initial Draft
1.1	April, 2013	Ryan Marinello	Updates, Review

Test Scripts

The following scripts cover this scenario:

7.0 POST /jmx/metricInfo Content-Type: application/json and Encoded Payload contents {"alternateKey": {"component": "test", "node": "node1", "server": "server"}, "insertTimestamp": "2013-04-23T11:45:08.353-05:00", "message": "message", "metricName": "name", "description": "WARN", "categoryType": “test", "displayName": "test”}

Test Requirements

This test scenario covers the following high-level test requirements (see scripts below for specific requirements covered by each test script):

PRMA.5

RADMA.1.1

RFP.88.1.4

NFRMA.6

RADMA.5

RADMA.5.1

User Groups

Technical Users

Script Description

This script tests the URI /jmx/metricInfo via POST

The status result is 201 Created and the JSON response is correctly constructed with required metadata.

Setup

Monitoring and Alerting is operational and MongoDB is connected

QA staff must have Rest Client available to perform test. It is also possible to see test results in the custom MnA UI.

QA staff has either a Local or Cloud environment deployed for testing to interact with UI.

Teardown

- None

Script Steps

Step #	Test Action	Expected Results
1	Access REST client verb POST	REST functional
2	Enter HOST value	http://localhost:8080/jmx
3	Enter PATH value	/jmx/metricInfo
	Payload requirements	either "regID" or "server", "node", "component" are required
4	Encode Payload	"alternateKey", "name", "description", "category", "displayName", regID", "insertTimestamp", }
5	Content-Type	application/json
6	Click Send	201 created JSON reponse headers return
		metric key name used to cache
		hyperic alerts listed
7	DB collection	stored data in 'metricInfo' collection

Test Execution

Date	Tester
April,2013	RM