**Steps to be taken to build the surya-sys-services-ocs API:**

* Discussion on Requirements,Name of the API, /resources,Flow-Design Completed
* System Connection Test thru mule-test-flow Completed
* Once Name Of API is confirmed then thru New Application Provisioning Jenkins Job which creates master n develop branch in github verify that n start development of the code by considering the develop branch code base into feature branch (if need to develop the Sync/Async API then after creating a git repo check the pom.xml all entries n check for -snapshot option in pom.xml)
* Raml Design with confirmed API NAME Completed
* Download n import the RAML in anypoint studio and do coding. Successful Build n test(postman) of the code locally,pom.xml change,jenkins file change,mule-artifacts.json change Completed
* MUNIT Test-Case creation n make sure that 60% code-coverage completed
* github commit of code completed
* mulesoft-configs policy(Endpoint mentioning place) n yaml file creation for surya-sys-services-ocs completed
* In Secret Manager,created & added secure properties for mule-surya-sys-services-ocs-env completed

QAT1

{

"json.logger.jms.password":"MuleAppLogsNP1234",

"da.x-api-key":"9QcJkS5ro25exS2BRZNr97gISy28wbZnwY7qQKqb",

"esbuser.ocs.db.password":"esbuser1",

"etauser.ocs.db.password":"c648690e49"

}

json.logger.jms.password=MuleAppLogsNP1234

etauser.ocs.db.password=c648690e49

esbuser.ocs.db.password=esbuser1

da.x-api-key =9QcJkS5ro25exS2BRZNr97gISy28wbZnwY7qQKqb

--------------------All coding part is done till here----------------------------------------------------------

* Verify the automatic generated API related Jenkins job is created,if not then connect @devopsTeam and then Deploy the code thru jenkins Completed
* Once deployed verify the Client Provider for the deployed API…If you are deploying the code into Dev then the Client Provider should be Okta-Dev n for every other env it should be Okta-Preview

Path to check:- Anypoint platform->API Manager->choose env->enter “api name’ in filter by box->settings->Under Runtime & Endpoint Configuration check for Client Provider…

If value mismatches like for DEV env it is Okta-Preview then you wont be able to use oauth token as it will say token has been revoked…

* Test the Deployed Code thru Postman Completed.
* Do the code review with Architect(Anubhav) n Team(Rahul,Sitansu,vinuthna)
* Completed the Exchange Documentation
* Promote the Code to higher environment for QA Testing by client Completed
* when promoted to CPV start Performance & load test using Jmeter Completed
* Create DATADOG DASHBOARD Completed

[Click here to access ETA API Dashboard](https://wastemanagement-esb.datadoghq.com/dashboard/d3n-xm8-fg6/mule-prod-eta?from_ts=1668612544350&to_ts=1668616144350&live=true)

* Create Confluence API Documentation Completed

---------------------- Preparation for PROD Release will be start from here-----------

prod connection details for OCS/DA

Hostname: ocsdprddb.surya.com

port:1521

esbusername: ESBUSER

esbpwd: cad51ac349

ServiceName for ESBUSER: ocsdprd\_esb\_srv

etausername: eta\_user

etapwd: fe8af07c07

ServiceName for eta\_user: ocsdprd\_srv

DA:

https://2xlh9nvt70.execute-api.us-east-1.amazonaws.com/v1/eta/

x\_api\_key = KkaPD6WXjF7fPQLFEI2GeaarwQyCJPkS4wnRxV48

* Verify Prod.yaml file in mulesoft-configs for connection strings,json loggers properties etc. Completed
* Create Secret Manager Entries for PROD Completed

Secret manager entries for prod ocs api:

{

"json.logger.jms.password":"MuleAppLogsPrd9Broker",

"da.x-api-key":"KkaPD6WXjF7fPQLFEI2GeaarwQyCJPkS4wnRxV48",

"esbuser.ocs.db.password":"cad51ac349",

"etauser.ocs.db.password":"fe8af07c07"

}

json.logger.jms.password=MuleAppLogsPrd9Broker

etauser.ocs.db.password=fe8af07c07

esbuser.ocs.db.password=cad51ac349

da.x-api-key =KkaPD6WXjF7fPQLFEI2GeaarwQyCJPkS4wnRxV48

* Create attachment.json file with deployment properties for Prod Deployment



[

{

"API": "surya-sys-services-ocs",

"RELEASE\_VERSION": "1.0.0-20221026.065315-8",

"API\_ASSET\_ID": "surya-sys-services-ocs",

"API\_VERSION": "1.0.0",

"WORKERS": "2",

"WORKER\_TYPE": "0.2",

"CH\_APP\_NAME": "surya-sys-services-ocs-v1-cloudapi",

"API\_PRODUCT\_VERSION": "v1",

"ENV": "PROD",

"CH\_RUNTIME\_VERSION": "4.4.0",

"DEPLOYMENT\_TEMPLATE": "Http API",

"STATIC\_IP": "false",

"APPROVED\_BY": "rvirani@surya.com"

}

]

* Create CRQ steps for Prod release to add Mule-predeployment steps n deployment steps completed



-----------------------PROD Validation--------------------

* once service deployed successfully,up n running in prod then take the prod postman collections GET endpoint n hit the API(if no GET endpoint is there then Hit /healthcheck) if getting successful response like 200,204,422 then validation is done Completed

-----------------------Hypercare Support------------------

* If any new or redeployment of apis is done where the Major code change is there(Minor code change like common- changes like json logger,big panda changes)then in datadog verify the logs for that API(recommended to look @prc APIs ,this is becoz all system api calls come n go thru prc api so if system api fails then that error will be logged in prc api also so look for prc level api) Completed
* Provide the performance stats for the APIs (new or redeployment of Major Changes in APIs) that got deployed. Datadog Dashboard Data will be helpful,for creating performance stats n you can take help from datadog logs also Completed

**Use below query to find exceptions**

host:cloudapi.surya.com⁠ source:muleapi⁠ @applicationName:(\*surya-prc-orders\* OR \*surya-prc-cust-tickets\* OR \*surya-sys-cust-tickets-mas\* OR \*surya-sys-cust-tickets-eadm\* OR \*surya-prc-cases\* OR \*surya-sys-cases-mas\* OR \*surya-sys-tkt-cas-empconsl\* OR \*surya-sys-tkt-cas-suryacom\*)⁠ @status:(EXCEPTION)⁠

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Time | Total ETA Requests ( this should include everything) | Total successful responses. | How many routed to | Out of those to Predictive ETA, how many were successful to get ETA from |
| Nov 13, 12:24 am – Nov 16, 9:55 am | 22875 | 20642 | EADM=6480 | EADM=320 |
|  |  |  | OCS=22867 | OCS=11942 |
|  |  |  | DA=8609 | DA=8380 |

The ETA – Service status Mulesoft API.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Time | API Name | Worker | WORKER\_TYPE | No. Of Request | No. Of Errors | No. Of Success | Average response time |
| Nov 13, 6:30 am – Nov 15, 6:30 am | surya-prc-locations-v2 | 2 | 0.2 | 9804 | 773 | 8859 | 1952msec |

|  |  |
| --- | --- |
| Errors | Occurences |
| The customer is not eligible to request ETA as Driver Assistance details not available | 55 |
| Invalid value 'null' for query parameter mas\_account\_number. expected type: String, found: Null | 7 |
| HTTP GET on resource '[https://cloudapi.surya.com:443/v1/sys-services-ocs/service-status'](https://cloudapi.wm.com/v1/sys-services-ocs/service-status%27) failed: Timeout exceeded. | 12 |
| org.mule.weave.v2.exception.UnsupportedTypeCoercionException: Cannot coerce Null (null) to DateTime | 1 |
| The customer is not eligible to request ETA | 698 |

------Creation of Release Branch from Develop code and then Release branch Code merge in Master, n deploy the code in QAT/QAT1 n test the API----------

* Creation of Release Branch code from Develop Branch and then merging release branch code into master branch completed
* Deploy the code in remaining env QAT1/QAT to make all development related branches like master,develop,release branch in sync n then test thru postman completed

Steps to consider while developing:-

* while commiting in github make sure no password,no sensitive info is commiteed like

in config.yaml replace sensitive info with empty string like below n then commit in github

da.x-api-key: "9QcJkS5ro25exS2BRZNr97gISy28wbZnwY7qQKqb"

esbuser.ocs.db.password: "esbuser1"

etauser.ocs.db.password: "c648690e49"

json.logger.jms.password: "MuleAppLogsNP1234"

da.x-api-key: ""

esbuser.ocs.db.password: ""

etauser.ocs.db.password: ""

json.logger.jms.password: ""

* at the end of the testing while deploying in cloudhub dev env enable/add the api autodiscovery wala part in global.xml me
* dont include api.id (api auto-discovery property) in any of the env.yaml file like dev,qat,qat1,prod,dr.yaml
* Make sure pom.xml has '-SNAPSHOT' in <version> element.

<groupId>com.surya</groupId>

<artifactId>surya-sys-services-ocs</artifactId>

<version>1.0.0-SNAPSHOT</version>

<packaging>mule-application</packaging>

* Make sure pom.xml has below entry to avoid error while building the project due to activemq error

</sharedLibraries>

<sharedLibrary>

<groupId>org.apache.activemq</groupId>

<artifactId>activemq-broker</artifactId>

</sharedLibrary>

<sharedLibrary>

<groupId>org.apache.activemq</groupId>

<artifactId>activemq-client</artifactId>

</sharedLibrary>

</sharedLibraries>

* Make sure mule-artifacts.json has below entry

"secureProperties": [

"anypoint.platform.client\_id",

"anypoint.platform.client\_secret",

"json.logger.jms.password",

"logging.aggregator.api.key.val",

"da.x-api-key",

"esbuser.ocs.db.password",

"etauser.ocs.db.password"

]

* Make sure jenkins file has below entries updated as below

@Library('mule-shared-library') \_

def apilist = [

serviceRepoUrl: 'https://github.surya.com/esb/surya-sys-services-ocs',

apiName: "surya-sys-services-ocs",

apiAssetID: "surya-sys-services-ocs",

apiSpecVersion: "1.0.0",

apiProductVersion: "v1",

clientProvider: "Okta-Dev",

staticIP: "false",

appType: "sync"

]

devDeploy(apilist)

* For sys-services-ocs,

for /service-status call, for all the loggers involved, please set the target app as OCS not OCS/DA

similarly for /eta call logger update target app as DA

source app in all the loggers should be surya-prc-locations-v2 and not SURYA becoz sys-services-ocs will be called from prc-locations-v2 api n not directly from client surya.

Postman auth token pre-request script:

**const** tokenUrl = '[https://integrationapiqat1.surya.com/v1/token'](https://integrationapiqat1.wm.com/v1/token%27);

**const** refresh\_token = "LwDB9iMZG6K3VARG5KjXVp3zXoEb4uOmRH14fXA8KKE";

**const** getTokenRequest = {

  method: 'POST',

  url: tokenUrl,

  header: { 'content-type': 'application/json' },

  body: {

      mode: 'raw',

      raw: {

            "grant\_type": "refresh\_token",

            "refresh\_token": refresh\_token

            }

  }

};

pm.sendRequest(getTokenRequest, (err, response) => {

**const** jsonResponse = response.json();

**const** newAccessToken = jsonResponse.access\_token;

  pm.variables.**set**('access\_token', newAccessToken);

});

Jenkins Attachment:

{  
    "API": "surya-prc-customer-setup",  
    "RELEASE\_VERSION": "1.0.13-20241018.062840-2",  
    "API\_ASSET\_ID": "prc-customer-setup",  
    "API\_VERSION": "1.0.20",  
    "CH\_APP\_NAME": "surya-prc-customer-setup-v1-cloudapi-qat1",  
    "API\_PRODUCT\_VERSION": "v1",  
    "ENV": "QAT1",  
    "WORKERS": "2",  
    "WORKER\_TYPE": "small",  
    "STATIC\_IP": "false",  
    "DEPLOYMENT\_TEMPLATE": "Http API",  
    "APPROVED\_BY": "adewagan@surya.com,kraut1@surya.com,sdubalgu@surya.com",  
    "CH\_RUNTIME\_VERSION": "4.6-java17",  
    "CH\_JAVA\_VERSION": "17",  
    "CH\_RELEASE\_CHANNEL": "LTS",  
    "CH\_LAST\_MILE\_SECURITY": "false",  
    "CH\_FORWARD\_SSL\_SESSION": "false",  
    "CH\_GENERATE\_PUBLIC\_URL": "false",  
    "CH\_OS\_V2": "false"  
  }