



Deployment Checklist & Testing Report

THE QLIKTAG PLATFORM V3.1.1

Version 1.0
Published Date 05 Oct 2020
Last Revision Date 05 Oct 2020

DEPLOYMENT CHECKLIST

This document describes the activities performed after production environment setup and deployment is completed. The aim of this is to verify and confirm the environment is functioning as expected.

Date	Verified By	Reviewed By	Comments
05 Oct 2020	Mohit Kumar	Ram Prasad Palugula	

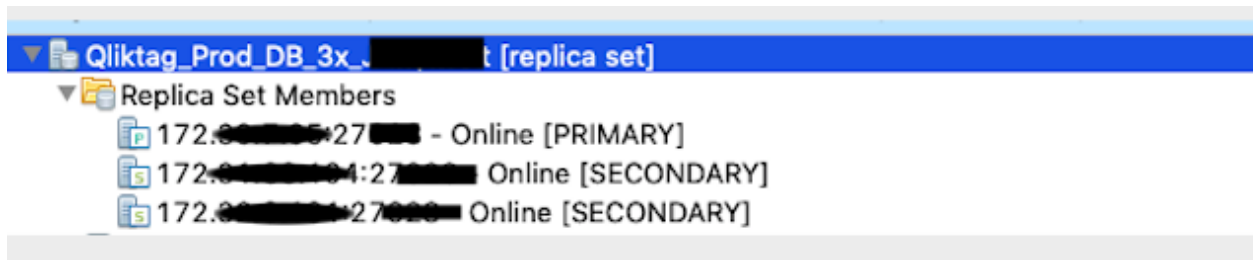
Environment	Description
Production Live	Production environment
Production DR Environment	Disaster Recovery environment
Sandbox	A sandbox environment on same production code version which is used for testing's or pre-production development by enterprise customers

Production Environment

S No	Review Item	Result (Pass/Fail)
1	Verify the database replica set connection	Pass
2	Verify the Elasticsearch status	Pass
3	Verify the bitbucket pipeline status	Pass
4	Verify lambda service status in AWS console	Pass
5	Verify the lambda service running on NodeJS 10.x	Pass
6	Verify the web application is deployed with Angular correctly	Pass
7	Verify the API health status /api/health	Pass
8	Browse the web application and verify the page loading	Pass
9	Verify the login to the web application	Pass
10	Navigate inside the web application and verify few pages	Pass
11	Test few API GET, POST operations	Pass
12	Verify emails send through application	Pass
13	Verify image uploads through API and web application	Pass
14	Verify few visual Interaction landing pages	Pass
15	Verify the cloudwatch logs	Pass

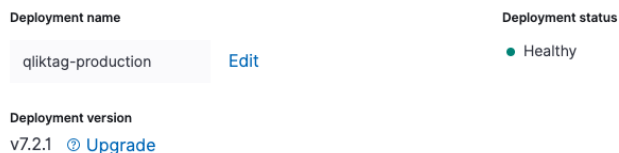
Below are the proof of screen shots for each review item:

SNo #1



SNo #2

qliktag-production



SNo #3

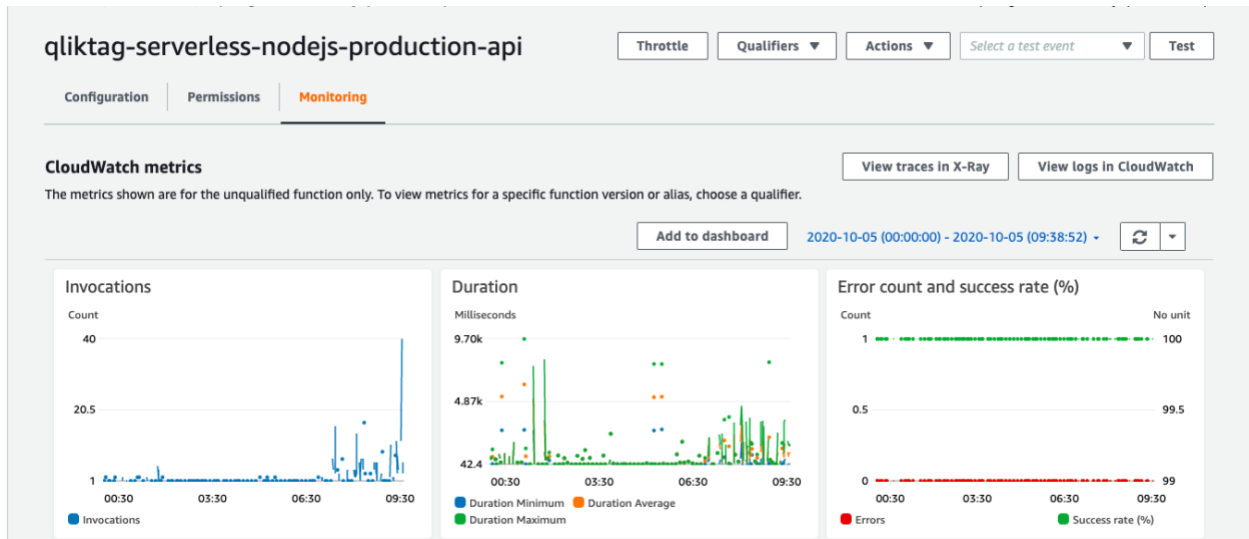
QliktagWeb application deployment

```
Build 🔍 🔄 🗑️ ℹ️  
  
Build setup 25s ➤  
echo "Pipeline Started for Production3x(qliktagweb) enviroment branc... <1s ➤  
node -v <1s ➤  
npm install 42s ➤  
npm run-script build 4m 42s ➤  
aws s3 rm --recursive s3://qliktag-ui-static/production/qliktagweb/ 7s ➤  
aws s3 cp dist/ s3://qliktag-ui-static/production/qliktagweb/ --recu... 12s ➤  
aws s3 rm --recursive s3://qliktag-ui-static-dr/production/qliktagwe... 16s ➤  
aws s3 cp dist/ s3://qliktag-ui-static-dr/production/qliktagweb/ --r... 28s ➤  
echo "Pipeline Completed for Production3x(qliktagweb) enviroment bra... <1s ➤  
Build teardown 28s ➤
```

QliktagAPI application deployment

```
Build docker 🔍 🔄 🗑️ ℹ️  
  
Build setup 13s ➤  
echo "Pipeline Started for Production3x enviroment branch" <1s ➤  
npm install 56s ➤  
npm install -g serverless 19s ➤  
serverless plugin install --name serverless-webpack 25s ➤  
sls deploy --stage production 4m 14s ➤  
sls deploy --stage production --config serverless-dr.yml 3m 52s ➤  
echo "Pipeline Completed for production3x enviroment branch" <1s ➤  
echo "Serverless cloudformation deployment is completed successfully" <1s ➤  
apt-get update && apt-get install -y python3-pip 19s ➤  
pip3 install -U awscli 18s ➤  
eval $(aws ecr get-login --no-include-email --region us-east-1) <1s ➤  
docker build -f docker/Dockerfile -t qliktag-3x-container-production . 41s ➤  
echo "Docker image build is completed successfully" <1s ➤  
docker tag qliktag-3x-container-production:latest 701463471155.dkr.ecr.us-east-1.amazonaws.com/qliktag-3x-container-production:latest <1s ➤  
docker push 701463471155.dkr.ecr.us-east-1.amazonaws.com/qliktag-3x-container-production:latest 53s ➤  
echo "Successfully pushed the docker image to repository" <1s ➤  
aws ecs update-service --cluster qliktag-3x-cluster-production --service qliktag-service-api-production --force-new-deployment 1s ➤  
aws ecs update-service --cluster qliktag-3x-cluster-production --service qliktag-service-redapi-production --force-new-deployment <1s ➤  
echo "Successfully updated production cluster of container services" <1s ➤  
Build teardown 57s ➤
```

SNo #4



SNo #5

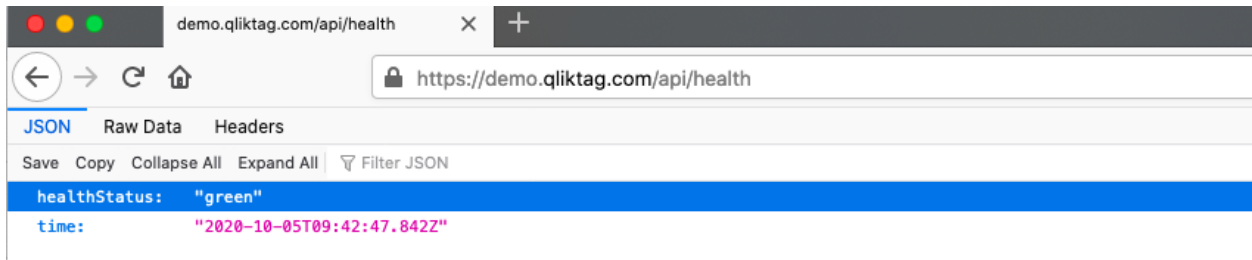
Function name	Description	Runtime	Code size	Last modified
<input type="radio"/> qliktag-serverless-nodejs-production-jobs		Node.js 10.x		20 minutes ago
<input type="radio"/> qliktag-serverless-nodejs-production-adminapi		Node.js 10.x		20 minutes ago
<input type="radio"/> qliktag-serverless-nodejs-production-api		Node.js 10.x		20 minutes ago
<input type="radio"/> qliktag-serverless-nodejs-production-redapi		Node.js 10.x		20 minutes ago

SNo #6

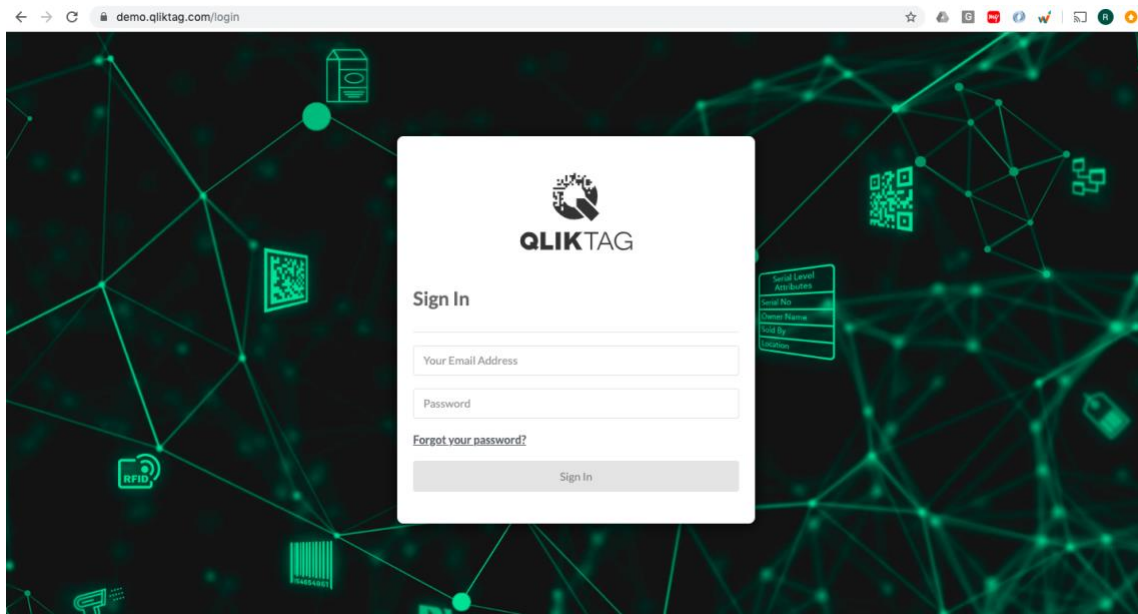
```
Angular CLI: 6.2.8
Node: 10.14.1
OS: darwin x64
Angular: 6.1.10
... animations, common, compiler, compiler-cli, core, forms
... http, language-service, platform-browser
... platform-browser-dynamic, router

Package                       Version
-----
@angular-devkit/architect     0.11.4
@angular-devkit/build-angular 0.11.4
@angular-devkit/build-optimizer 0.11.4
@angular-devkit/build-webpack 0.11.4
@angular-devkit/core          7.1.4
@angular-devkit/schematics    0.8.8
@angular/cdk                  6.4.7
@angular/cli                   6.2.8
@angular/flex-layout          6.0.0-beta.18
@angular/material             6.4.7
@ngtools/webpack              7.1.4
@schematics/angular           0.8.8
@schematics/update             0.8.8 (cli-only)
```

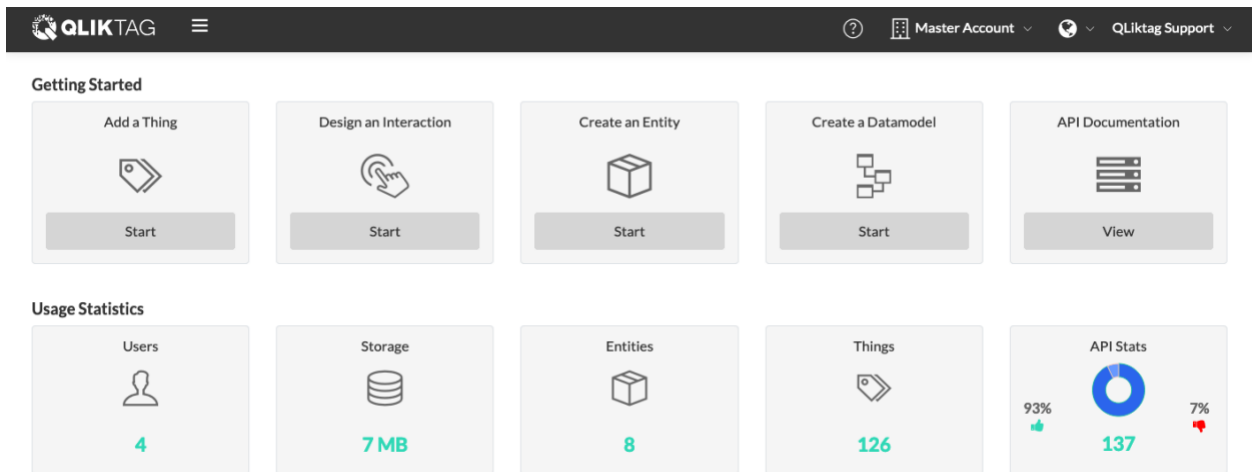
SNo #7



SNo #8



SNo #9



SNo #10

Browser address bar: <https://demo.qliktag.com/users>

Navigation: QLIKTAG | Master Account | Qliktag Support

Users

[+ Add User](#)

Home / Users

First Name	Last Name	Email	Status	
neil	sequeira	neil.sequeira@qliktag.com		
Demo	Qliktag	support@qliktag.com		
Arjun	Viswanathan	arjun.viswanathan@qliktag.com		
Dilip	Daswani	dilip.daswani@qliktag.com		

4 total

SNo #11

GET API

GET /api/account 🔒

Get account details in controller

Parameters Cancel

Name	Description
content-type * <small>required</small> string <i>(header)</i>	Content-type of Response <input type="text" value="application/json"/>
page integer <i>(query)</i>	Page number <input type="text" value="page - Page number"/>
items_per_page integer <i>(query)</i>	Items per page <input type="text" value="items_per_page - Items per page"/>
sort string(Setting) <i>(query)</i>	Sort by field Name <input type="text" value="sort - Sort by field Name"/>
sort_order string <i>(query)</i>	Sort Order <input type="text" value="--"/>
queryString string(Setting) <i>(query)</i>	&key1=val1&key2=val2 <input type="text" value="queryString - &key1=val1&key2=val2"/>

Responses

Curl

```
curl -X GET "https://demo.qliktag.com/api/account" -H "accept: */*" -H "content-type: application/json" -H "Authorization: Bearer a767a29a-28de-47f7-aa56-1ed7bf940f3c"
```

Request URL

```
https://demo.qliktag.com/api/account
```

Server response

Code	Details
200	<p>Response body</p> <pre>{ "requestStatus": true, "errors": [], "warnings": [], "requestedAt": "2020-10-05T09:59:54.350Z", "data": [</pre>

POST API

SNo #12



Welcome!

Mohit,

Demo is now using The Qliktag Platform – An IoT Platform for Everyday Consumer Products! The system Administrator Demo has invited you to join. In order to activate your user account, you will have to first set a secure password. Once you've set a password, you can login to the system using the password and details below:

SNo #13

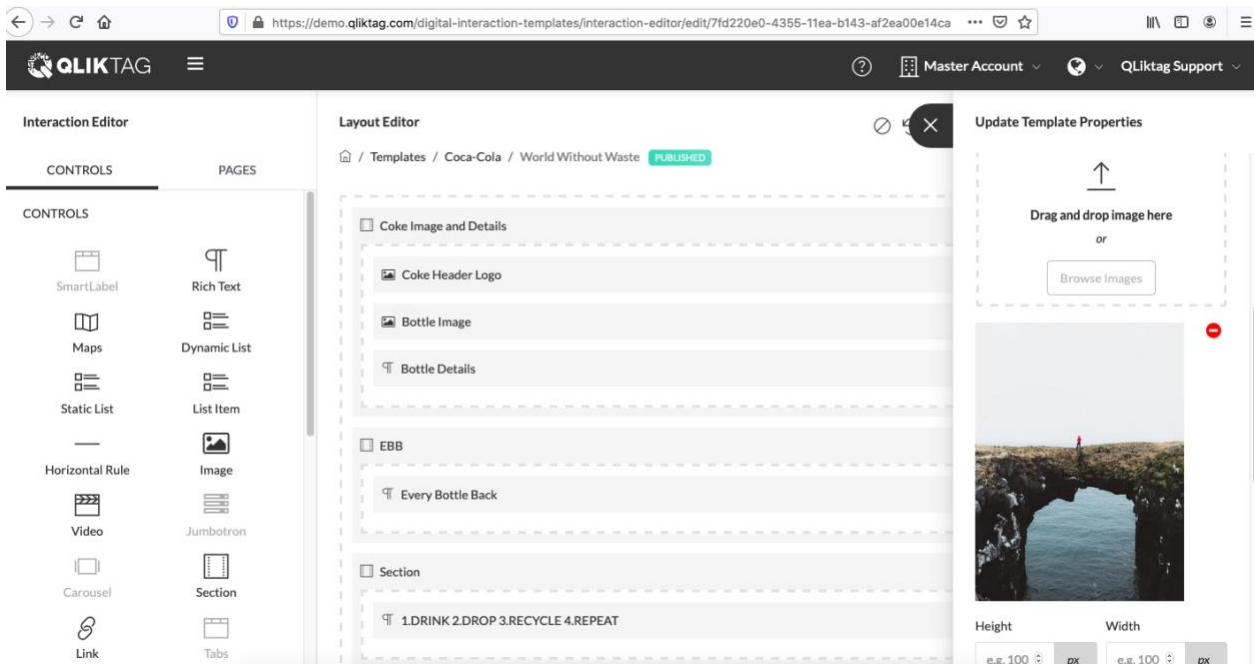
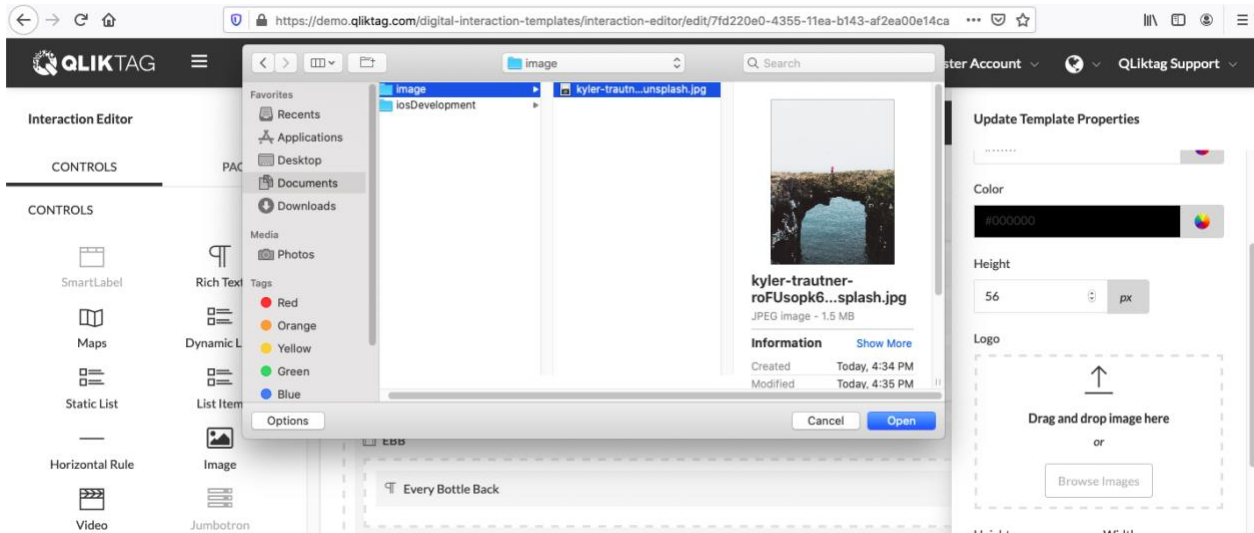
Through API

The screenshot displays an API client interface for a POST request to the endpoint "/api/assets/upload/file". The interface includes a "Parameters" section with a "Cancel" button, a "Request body" section with a dropdown menu set to "multipart/form-data", and a "Responses" section. The "Request body" section contains three fields: "use_file_name" (boolean, true), "unique_file" (boolean, true), and "file" (string (binary), with a "Choose file" button and the selected file "roberto-elik_unsplash.jpg"). The "Responses" section shows the request details, including the request URL "https://demo-qliktag.com/api/assets/upload/file", the server response code "200", and the response body: {"requestStatus": true, "errors": {}, "warnings": [], "data": [{"fileId": 486302, "url": "https://cdn.qliktag.com/production/5da20deb97867206198ba15/roberto-elikom-6gw_8da098-unsplash.jpg"}]}.

```
curl -X POST "https://demo-qliktag.com/api/assets/upload/file" -H "Accept: */*" -H "Authorization: Bearer 1f8f8ba-7d4d-41f8-97cf-ea348ea321d" -H "Content-Type: multipart/form-data" -F "use_file_name=true" -F "unique_file=true" -F "file@roberto-elikom-6gw_8da098-unsplash.jpg;type=image/jpeg"
```

Code	Details
200	<pre>{ "requestStatus": true, "errors": {}, "warnings": [], "data": [{ "fileId": 486302, "url": "https://cdn.qliktag.com/production/5da20deb97867206198ba15/roberto-elikom-6gw_8da098-unsplash.jpg" }] }</pre>

Through UI



SNo #14

The screenshot shows a product page for 'Head & Shoulders Anti-Dandruff Shampoo' (180 ml) on the smartlabel platform. The page includes a navigation menu with categories like 'Ingredients & Contents', 'Health, Safety & Environment', 'Usage & Handling', 'Features & Benefits', and 'Company, Brand & Sustainability'. Below the menu, there are sections for 'Human Health', 'Storage and Disposal', and 'Environment'. A disclaimer states 'This label is for demo purposes only.' and a 'Feedback for Smartlabel' link is provided. Social media icons for Facebook, Pinterest, and Twitter are at the bottom.

SNo #15

The screenshot displays the AWS CloudWatch Logs Insights interface for the 'qliktag-serverless-nodejs-production-api'. It shows a table of recent function invocations with columns for #, Timestamp, RequestID, LogStream, DurationInMS, BilledDurationInMS, and MemorySetInMB. The data shows various invocations with durations ranging from approximately 316.72ms to 1002.45ms.

#	Timestamp	RequestID	LogStream	DurationInMS	BilledDurationInMS	MemorySetInMB
1	2020-10-05T09:45:30.694Z	028dfaf4-1f37-42d5-bebf-6ab9c658420e	2020/10/05/[\$LATEST]d2c5589901f6449ca617749666d327c8	882.7	900	512
2	2020-10-05T09:45:30.235Z	bfe86fd6-3868-4fe1-a641-dcc248fc98a7	2020/10/05/[\$LATEST]c217490d63ad411cb643082499623551	832.83	900	512
3	2020-10-05T09:45:29.016Z	41867866-7c7e-4cd1-a6dc-3dbbb66d9986	2020/10/05/[\$LATEST]c217490d63ad411cb643082499623551	841	900	512
4	2020-10-05T09:45:27.395Z	907ca681-7095-4ecf-8dde-2a78f283906	2020/10/05/[\$LATEST]c217490d63ad411cb643082499623551	316.72	400	512
5	2020-10-05T09:44:58.056Z	3e1e52d2-bc11-445e-a4e0-6102791b2207	2020/10/05/[\$LATEST]c217490d63ad411cb643082499623551	1002.45	1100	512
6	2020-10-05T09:44:57.530Z	d6bf456e-d5b6-4e6b-9043-dad9be9b03ba	2020/10/05/[\$LATEST]042cd22e01f647e49fa1def1fd99441	524.78	600	512
7	2020-10-05T09:44:57.511Z	3b2e0451-d0c9-4aab-9123-2613c466a4bd	2020/10/05/[\$LATEST]d2c5589901f6449ca617749666d327c8	542.25	600	512
8	2020-10-05T09:44:56.293Z	f810d69b-856d-493c-b1ff-455cdd0608c8	2020/10/05/[\$LATEST]d2c5589901f6449ca617749666d327c8	1715.93	1800	512
9	2020-10-05T09:44:40.870Z	c3274d57-47c8-496d-8be9-fbffb43a2e54	2020/10/05/[\$LATEST]d2c5589901f6449ca617749666d327c8	737.46	800	512

Production Disaster Recovery (DR) Environment

S No	Review Item	Result (Pass/Fail)
1	Verify the database replica set connection	Pass
2	Verify the Elasticsearch connection	Pass
3	Verify the bitbucket pipeline status	Pass
4	Verify lambda service status in AWS console	Pass
5	Verify the lambda service running on NodeJS 10.x	Pass
6	Verify the web application is deployed with Angular correctly	Pass
7	Verify the API health status /api/health	Pass
8	Browse the web application and verify the page loading	Pass
9	Verify cloudwatch logs	Pass

Sandbox Environment

Sandbox is a cloud account configuration with “<<controllerCode>>-sandbox” for enterprise customers in the same production release code. Only below verification items are applicable as it runs in the same production environment and verified under production environment deployment.

S No	Review Item	Result (Pass/Fail)
1	Browse the web application and verify the page loading	Pass
2	Verify the login to the web application	Pass
3	Navigate inside the web application and verify few pages	Pass
4	Test few API GET, POST operations	Pass
5	Verify emails send through application	Pass
6	Verify image uploads through API and web application	Pass
7	Verify few visual Interaction landing pages	Pass