



Deployment Checklist & Testing Report

THE QLIKTAG PLATFORM V3.0.3

Version 1.0

Published Date 27 April 2020

Last Revision Date 27 April 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
27 April 2020	Mohit Kumar	Ram Prasad Palugula	

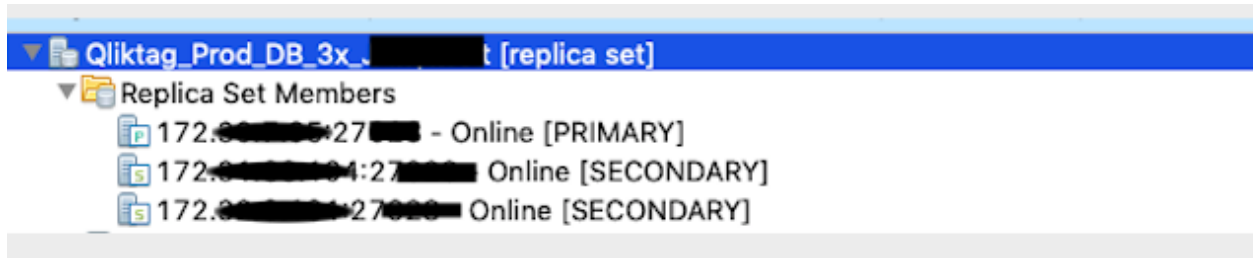
Environment	Description
Production Live	Production environment
Production DR Environment	Disaster Recovery environment
Sandbox	A sandbox environment on same production code version which is used for testing's or pre-production development by enterprise customers

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

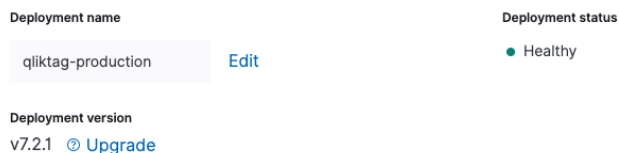
Below are the proof of screen shots for each review item:

SNo #1



SNo #2

qliktag-production



SNo #3

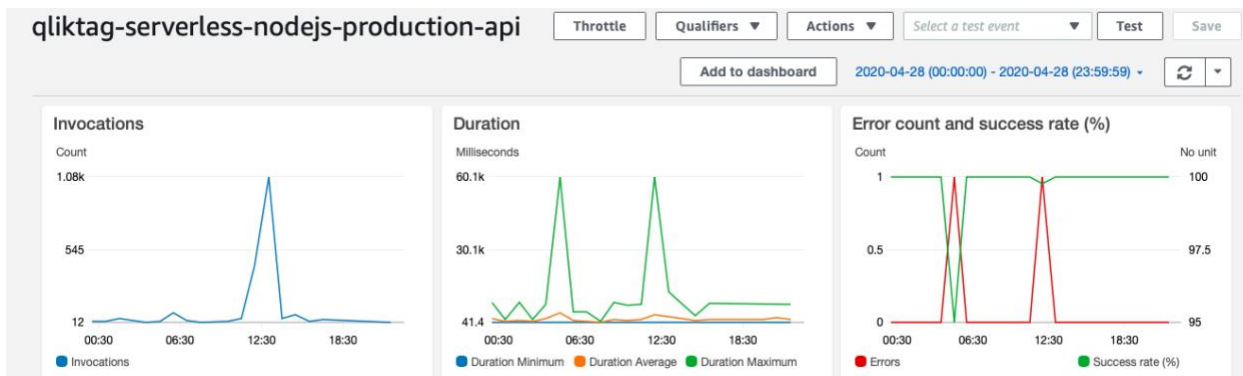
QliktagWeb application deployment

```
Build
Build setup 30s >
echo "Pipeline Started for Production3x(qliktagweb) enviroment branc... <-1s >
node -v <-1s >
npm install 44s >
npm run-script build 4m 17s >
aws s3 rm --recursive s3://qliktag-ui-static/production/qliktagweb/ 7s >
aws s3 cp dist/ s3://qliktag-ui-static/production/qliktagweb/ --recu... 9s >
aws s3 rm --recursive s3://qliktag-ui-static-dr/production/qliktagwe... 18s >
aws s3 cp dist/ s3://qliktag-ui-static-dr/production/qliktagweb/ --r... 31s >
echo "Pipeline Completed for Production3x(qliktagweb) enviroment bra... <-1s >
Build teardown 59s >
```

QliktagAPI application deployment

```
Build docker
echo "Serverless cloudformation deployment is completed successfully" <-1s >
apt-get update && apt-get install -y python3-pip 18s >
pip3 install -U awscli 7s >
eval $(aws ecr get-login --no-include-email --region us-east-1) <-1s >
docker build -f docker/Dockerfile -t qliktag-3x-container-production... 26s >
echo "Docker image build is completed successfully" <-1s >
docker tag qliktag-3x-container-production:latest 701463471155.dkr.e... <-1s >
docker push 701463471155.dkr.ecr.us-east-1.amazonaws.com/qliktag-3x-... 29s >
echo "Successfully pushed the docker image to repository" <-1s >
aws ecs update-service --cluster qliktag-3x-cluster-production --ser... <-1s >
aws ecs update-service --cluster qliktag-3x-cluster-production --ser... <-1s >
echo "Successfully updated production cluster of container services" <-1s >
Build teardown <-1s >
```

SNo #4



SNo #5

	Function name	Description	Runtime
<input type="radio"/>	qliktag-serverless-nodejs-production-adminapi		Node.js 10.x
<input type="radio"/>	qliktag-serverless-nodejs-production-api		Node.js 10.x
<input type="radio"/>	qliktag-serverless-nodejs-production-redapi		Node.js 10.x

SNo #6

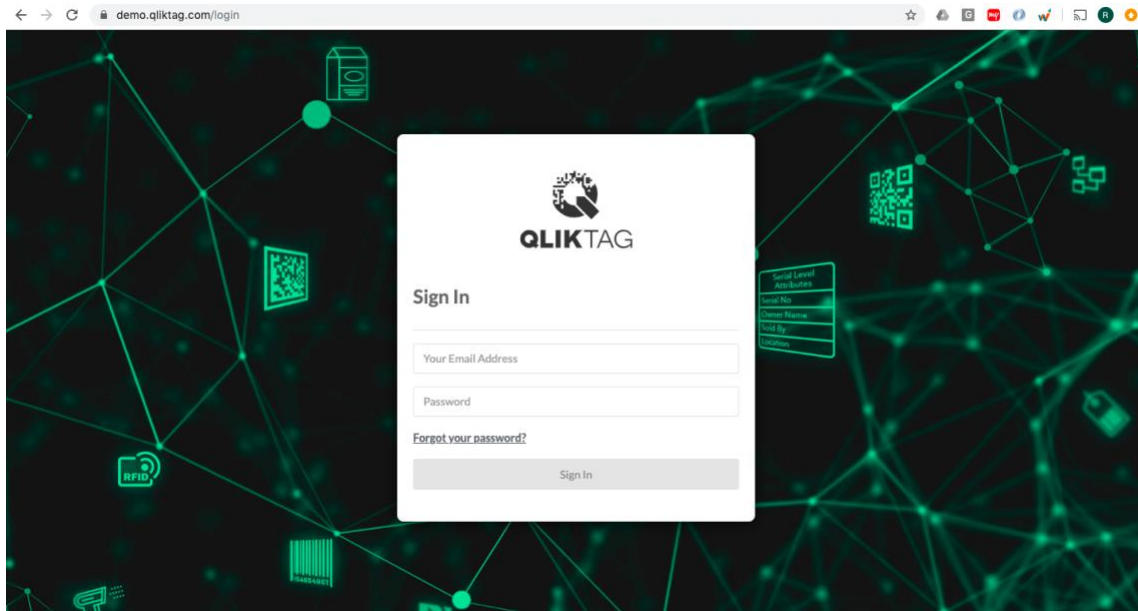
```
Angular CLI: 6.2.8
Node: 10.14.1
OS: darwin x64
Angular: 6.1.10
... animations, common, compiler, compiler-cli, core, forms
... http, language-service, platform-browser
... platform-browser-dynamic, router

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)
```

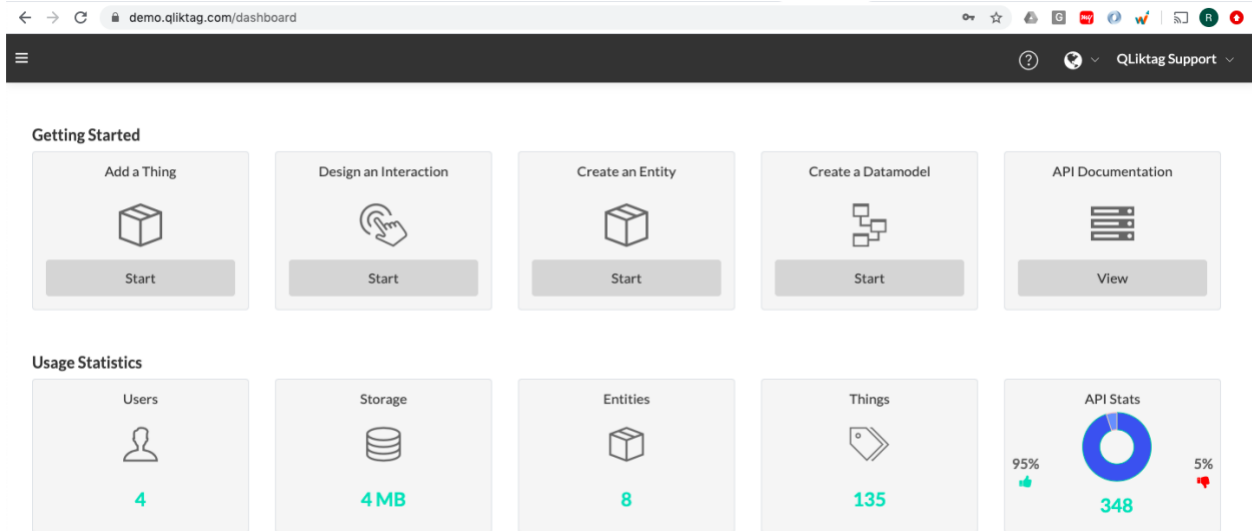
SNo #7

```
← → ↻ demo.qliktag.com/api/health
{
  healthStatus: "green",
  time: "2020-04-27T10:23:35.283Z"
}
```

SNo #8



SNo #9



SNo #10

demo.qliktag.com/users

Users + Create User

🏠 / Users

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

4 total

SNo #11

GET API

GET /api/account

Get account details in controller

Parameters Cancel

Name	Description
content-type *required string (header)	Content-type of Response application/json
page integer (query)	Page number page - Page number
items_per_page integer (query)	Items per page items_per_page - Items pe
sort string(\$string) (query)	Sort by field Name sort - Sort by field Name
sort_order string (query)	Sort Order -
queryString string(\$string) (query)	&key1=val1&key2=val2 queryString - &key1=val1&

Execute Clear

Responses

Curl

```
curl -X GET "https://demo.qliktag.com/api/account" -H "accept: */*" -H "content-type: application/json" -H "Authorization: Bearer 89e8739d-c50c-4dbf-a398-c49b94248fdb"
```

Request URL

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

Server response

Code Details

200 Response body

```
{
  "requestStatus": true,
  "errors": [],
  "warnings": [],
  "requestedAt": "2020-04-27T12:44:34.593Z",
  "data": /
```


POST API

POST /api/oauth

Create and Refresh the Authorization Token for validating the incoming request to the system

Parameters Cancel

No parameters

Request body required password

Edit Value | Model

```
{
  "data": {
    "email": "#####@qliktag.com",
    "password": "#####",
    "grantType": "password"
  }
}
```

Cancel Reset

Execute Clear

Responses

Curl

```
curl -X POST "https://demo.qliktag.com/api/oauth" -H "accept: */*" -H
"Authorization: Bearer 89e8739d-c50c-4dbf-a398-c49b94248fdb" -H "Content-Type:
application/json" -d "{\"data\":
```

Request URL

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

Server response

Code **Details**

200 **Response body**

```
{
  "requestStatus": true,
  "errors": [],
  "warnings": [],
  "requestedAt": "2020-04-27T12:43:03.507Z",
  "data": {
    "userId": "4fe9be90-063a-11ea-8bd6-2183ec0290b1",
    "accountUserId": "4fe9be90-063a-11ea-8bd6-2183ec0290b1"
```

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:

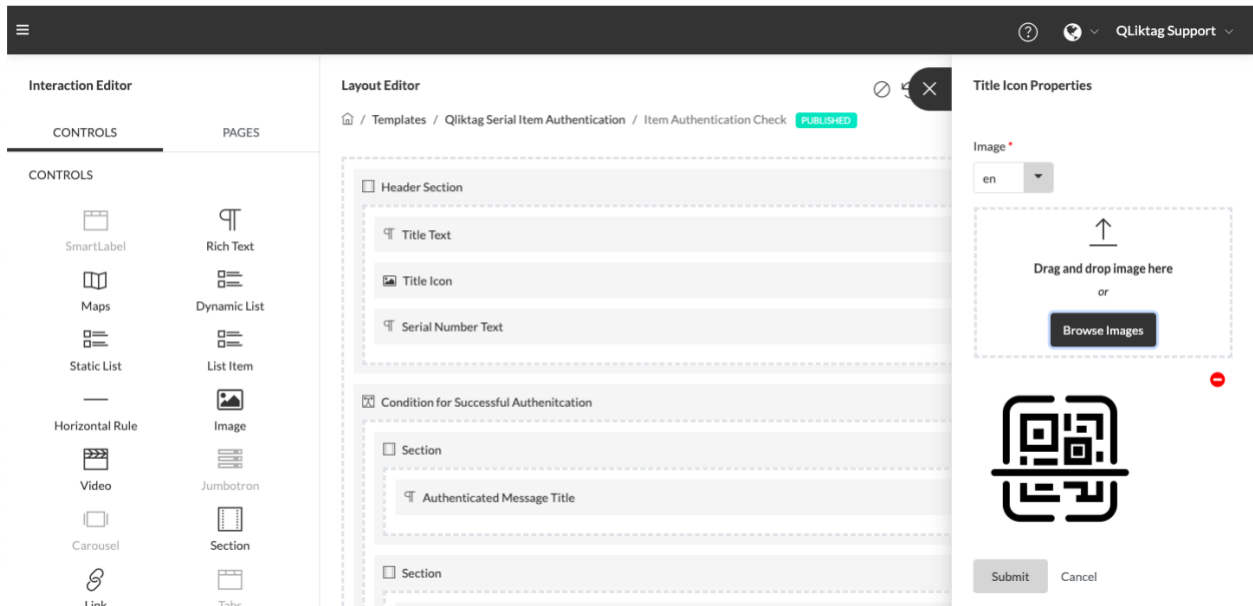
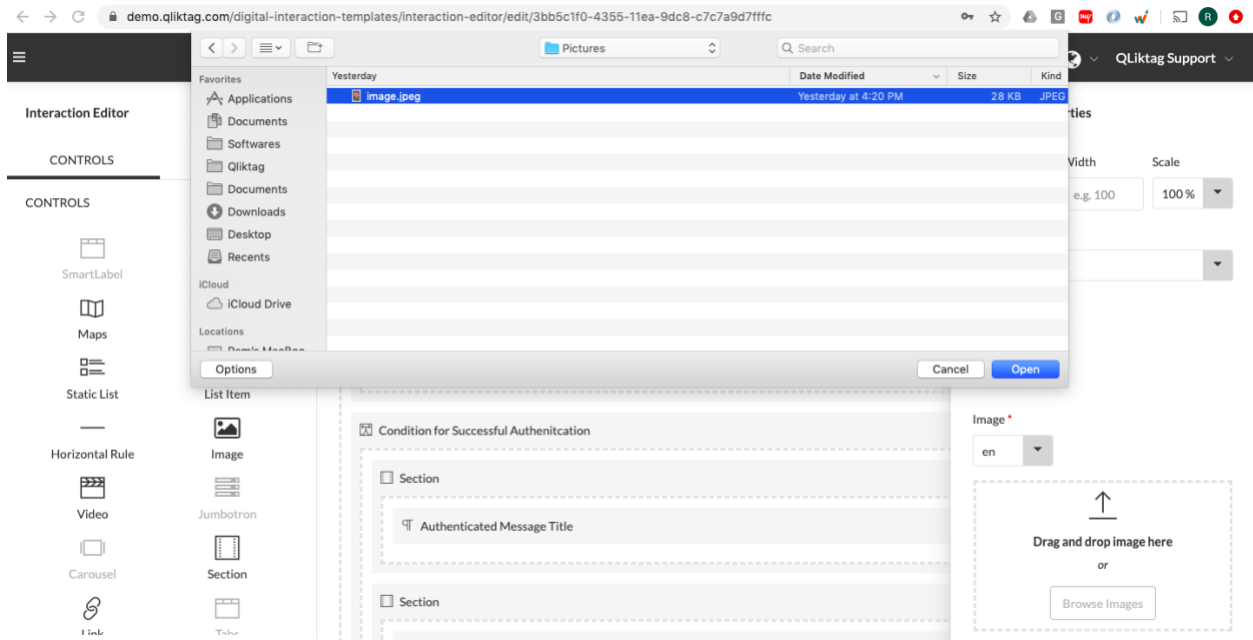
SNo #13

Through API

The screenshot shows an API client interface for a POST request to the endpoint `/api/assets/upload/file`. The description states: "Create new digital asset (Single or Bulk assets data) using file uploads for the particular Controller and environment". The "Parameters" section is empty. The "Request body" is set to "multipart/form-data". The form includes three fields: `use_file_name` (boolean, true), `unique_file` (boolean, true), and `file` (string (binary), with a file selection button for `roberto-nick_unsplash.jpg`). Below the form are "Execute" and "Clear" buttons. The "Responses" section shows a successful response with a status code of 200. The response body is a JSON object:


```
{  "requestStatus": true,  "errors": {},  "warnings": [],  "data": {    "filename": "486302",    "url": "https://cdn-qliktag-mm/product-images/56a2ndeb97b647206188bc15/roberto-nick-4qwr_4d698-unsplash.jpg"  }}
```

Through UI




SNo #14


demo.qliktag.ag/ip/fc3c8540-4354-11ea-8396-55402ef3e162/f2081290-3c35-11ea-9943-d9dbbde19345/p/smartlabel#usage-and-handling





Head & Shoulders
Head & Shoulders Anti-Dandruff Shampoo
 180 ml





GPC: 10000834
04902430774130


 Ingredients & Contents


 Health, Safety & Environment


 Usage & Handling





 Features & Benefits


 Company, Brand & Sustainability

Usage Instructions
 Shake before use. Apply onto wet hair, lather and rinse thoroughly. Repeat if desired.

Disclaimer: This label is for demo purposes only.

[Feedback for Smartlabel](#)

SNo #15

qliktag-serverless-nodejs-production-api

2020-04-27 (00:00:00) - 2020-04-27 (23:59:59)

Recent invocations

#	Timestamp	RequestID	LogStream	DurationInMS	BilledDurationInMS	
▶ 1	2020-04-27T23:59:33.554Z	c64d52d1-ac83-45c4-84fa-f31b4dfbf530	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	44.65	100	5
▶ 2	2020-04-27T23:51:33.568Z	d9d7a5bf-99d9-4fdd-a2e0-040e83cec7ac	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	94.74	100	5
▶ 3	2020-04-27T23:51:10.179Z	c451f9d2-51c3-481e-95ec-b0ccc14aa258	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	44.43	100	5
▶ 4	2020-04-27T23:43:10.168Z	4d194298-7f1c-4a35-a0ab-616d6936016a	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	45.64	100	5
▶ 5	2020-04-27T23:36:54.877Z	d209bb9f-c914-4d46-8400-593cbf539851	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	1143.22	1200	5
▶ 6	2020-04-27T23:36:34.546Z	61275fa1-930b-45a2-a2a6-025b3aa1ffb2	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	448.04	500	5
▶ 7	2020-04-27T23:35:45.043Z	02e3ac76-1a03-4f81-8e16-bfadaa768c0f	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	995.34	1000	5
▶ 8	2020-04-27T23:35:10.202Z	b16d7c44-731b-420f-b800-4fdb08c31618	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	97.66	100	5
▶ 9	2020-04-27T23:27:10.360Z	20b6179f-b825-41cd-8798-518a4e9f5e8b	2020/04/27/(\$LATEST)36e8295b81424229b40d8c5620ecc5a6	236.74	300	5

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 same production release code. Only below verification items are applicable as it runs in 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