PREVAIL

Technical Conference
2021

Web Application(Test Automation Framework) Deployment in OpenShift

Component Details

Nginx :

Nginx is a webserver, which can act as email proxy, reverse proxy, load balancer.

- In our case, Nginx serves all the static file and frontend application.
- Nginx can't directly communicate with Python base web applications that's why we need gateway interface (wsgi server) to interpret and handle request to Python. In our application waitress acts as wsgi server and Nginx is in front of it acting as reverse proxy.

Django :

Django is a Python-based free and open-source web framework that follows the MVT architectural pattern. Backend is written using Django

Celery :

Celery is a task queue which is a data structure maintained by a job scheduler containing jobs to run. It is designed for asynchronous operations, i.e., operations are executed in a non-blocking mode allowing the main operation to continue processing.

RabbitMQ :

RabbitMQ is a message broker which allows applications, systems, and services to communicate and exchange information with each other.

PostgreSQL :

PostgreSQL is an opensource SQL database (Relational). Stores data in Table based format.

CouchDB :

Apache CouchDB is an open-source document-oriented NoSQL database (Non-Relational).Stores data in key-value pairs, document-based, graph databases or wide-column based

Configuration Files

Deployment Configuration

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.16
        ports:
        - containerPort: 8080
```

Service Configuration

```
apiVersion: v1
kind: Service
metadata:
  name: nginx-service
spec:
  selector:
    app: nginx
  ports:
    - protocol: TCP
      port: 80
      targetPort: 8080
```

Deployment order in OpenShift

1. env-configmap.yaml
2. pgdata-persistentvolumeclaim.yaml
3. dbdata-persistentvolumeclaim.yaml (After step 3, wait till the persistent volume gets created)
4. hub-deployment.yaml
5. hub-service.yaml
6. chrome-deployment.yaml
7. chrome-service.yaml
8. firefox-deployment.yaml
9. firefox-service.yaml
10. postgres-deployment.yaml
11. postgres-service.yaml
12. couchserver-deployment.yaml
13. couchserver-service.yaml (After step 13, add Route to CouchDB and create database named “projects” [case sensitive])
14. uiaf-rabbit-deployment.yaml
15. uiaf-rabbit-service.yaml (After step 15, add Route to uiaf-rabbit)
16. web-deployment.yaml
17. web-service.yaml
18. nginx-deployment.yaml
19. nginx-service.yaml (After step 19, add Route to nginx)

THANK YOU