## Deploying a Flask Application On Kubernetes with Autoscaling And Load Testing

```
Activities 📝 👨 Workstation 🔻 📙 🔻 👨 👂 😩 🖺 🔲 🖂 😭 🖸 🐼 🐼 🐼 🐼 Worker1 💢 🗀 Worker2 💢
                                                                                                                                                                                                            · - - - X · · ·
                                                                                                     master@master-vm: ~/Desktop/flask-kube
             Reinitialized existing Git repository in /home/master/Desktop/flask-kube/.git/
master@master-vm:-/Desktop/flask-kube$ Is
app.py deployment-service.yaml Dockerfile README.md requirements.txt
master@master-vm:-/Desktop/flask-kube$ git clone https://github.com/Bhargavkulla/flask-kube.git
             Sending build context to Docker daemon 159.2kB

Step 1/7 : FROM python:3.11

3.11: Pulling from library/python
155ad54a8b28: Pull complete
8831108153cda: Pull complete
1d281e50d3e4: Pull complete
447713e77b4f: Pull complete
447773e74f5b: Pull complete
447749a24f5b: Pull complete
672d84e58157: Pull complete
072d84e58157: Pull complete
072d84e58157: Pull complete
772d84e58157: Pull complete
             ---> 7855334082Cb
Step 2/7 : WORKDIR /app
---> Running in d55b712c6e05
---> bfc2879be1f3
Step 3/7 : COPY . /app
---> cfb89a59692d
    •
             Step 4/7: RUN pip install --no-cache-dir -r requirements.txt ---> Running in 72e28ef54e90
master@master-vm:~/Desktop/flask-kube$ docker push bhargavakulla/flask-kube
Using default tag: latest
The push refers to repository [docker.io/bhargavakulla/flask-kube]
 ed2d2de7e13d: Pushed
 a4e99a22ba81: Pushed
 ff84030030d9: Pushed
b723da6e1cf4: Mounted from library/python
7af6b2a8a1a8: Mounted from library/python
71030c5d3283: Mounted from Library/python
4b017a36fd9c: Mounted from Library/python
20a9b386e10e: Mounted from Library/python
f8217d7865d2: Mounted from library/python
01c9a2a5f237: Mounted from library/python
latest: digest: sha256:c1654c6c11816344e2656ed12cde2f737465c81dc39ac3eb895a14bc0e52a8a<u>8 size: 2422</u>
                                -vm:~/Desktop/flask-kube$ kubectl apply -f deployment-service.yaml
 deployment.apps/flask-app created
service/flask-service created
                                  vm:~/Desktop/flask-kube$ kubectl create secret docker-registry docker-secret \
          --docker-server=https://index.docker.io/v1/ \
          --docker-username=bhargavakulla \
                                                                                                                                                                        I
          --docker-password=Bharu@12345 \
           --docker-email=bhargavaramcloud@gmail.com
secret/docker-secret created
                                                             kube$ kubectl patch serviceaccount default -p '{"imagePullSecrets": [{"name": "docker-secret"}]}
 serviceaccount/default patched
                                            op/flask-kube$ kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/latest/download/c
omponents.yaml
serviceaccount/metrics-server created
clusterrole.rbac.authorization.k8s.io/system:aggregated-metrics-reader created
clusterrole.rbac.authorization.k8s.io/system:metrics-server created
rolebinding.rbac.authorization.k8s.io/metrics-server-auth-reader created
clusterrolebinding.rbac.authorization.k8s.io/metrics-server:system:auth-delegator created
clusterrolebinding.rbac.authorization.k8s.io/system:metrics-server created
service/metrics-server created
deployment.apps/metrics-server created
apiservice.apiregistration.k8s.io/vibeta1.metrics.k8s.io created
master@master-vm:-/Desktop/flask-kube$
                                                                                                                                                                                           Activate Windows
```

```
master@master-vm:~/Desktop/flask-kube$ kubectl autoscale deployment flask-app --cpu-percent=50 --min=3 --max=10
horizontalpodautoscaler.autoscaling/flask-app autoscaled
master@master-vm:~/Desktop/flask-kube$ kubectl get hpa
                                              autoscaler.outo
-vm:~/Desktop/flask-kube$ kube
TARGETS
NAME REFERENCE TARGETS MINPODS MAXPODS REPLICAS AGE flask-app Deployment/flask-app cpu: <unknown>/50% 3 10 0 3s master@master-vm:~/Desktop/flask-kube$ kubectl run -it --rm busybox --image=busybox -- /bin/sh If you don't see a command prompt, try pressing enter.
 / #
/ # ls
  / # pwd
  /
/ # echo "hi bhargav"
hi bhargav
/ # exit
Session ended, resume using 'kubectl attach busybox -c busybox -i -t' command when the pod is running
pod "busybox" deleted
master@master-vm:~/Desktop/flask-kube$
Activities 🗷 👨 Workstation 🔻 📙 🔻 👵 🚇 🚇 🚇 🔛 🖃 🖫 🔁 🐼 🐼 Master 🗆 🕞 Worker1 🗵 🗎 Worker2 🗵
                                                                                                                                                                                                                                                                                                                                                        master@master-vm: ~
                                                                                                                                                                                                                                                                                                                                                          Q = - ø
                      master@master-vm:-$ minikube start

minikube v1.35.0 on Ubuntu 20.04

Using the docker driver based on existing profile

Starting "minikube" primary control-plane node in "minikube" cluster

Pulling base image v0.0.46 ...

Restarting existing docker container for "minikube" ...

Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...

Verifying Kubernetes components...

Using image gcr.io/k8s-minikube/storage-provisioner:v5

Enabled addons: storage-provisioner, default-storageclass
                     | /usr/bin/kubectl is version 1.28.15, which may have incompatibilities with Kubernetes 1.32.0.

■ Want kubectl v1.32.0? Try 'minikube kubectl -- get pods -A'

■ Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
master@master.vm:-$ kubectl get nodes

NAME STATUS ROLES AGE VERSION
minikube Ready control-plane 27h v1.32.0
master@master.vm:-$ kubectl get svc flask-service

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
flask-service NodePort 10.106.68.34 <none> 80:32665/TCP 77m
master@master.vm:-$ kubectl get svc

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
flask-service NodePort 10.106.68.34 <none> 80:32665/TCP 77m
kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 27h
master@master.vm:-$ minikube ip
192.168.49.2
                      192.168.49.2

master@master=vm:=$ kubectl run -it --rm load-generator --image=busybox -- /bin/sh

If you don't see a command prompt, try pressing enter.

/ # while true; do wget -q -0- http://192.168.49.2:32665; done

{"message":"Hello, World! This is a Flask app running in Docker."}

{"message":"Hello, World! This is a Flask app running in Docker."}

{"message":"Hello, World! This is a Flask app running in Docker."}

{"message":"Hello, World! This is a Flask app running in Docker."}

{"message":"Hello, World! This is a Flask app running in Docker."}

{"message":"Hello, World! This is a Flask app running in Docker."}

{"message":"Hello, World! This is a Flask app running in Docker."}
      D.V.D
                                                                                                                                                                                                                                                                                                                                Activate Windows
```

