

**VELLORE INSTITUTE OF
TECHNOLOGY, VELLORE**

SOFTWARE ENGINEERING

COURSE CODE: BCSE301L

CLASS NUMBER: VL2024250502304

FACULTY: DR NAVEENKUMAR J

DIGITAL ASSIGNMENT

NAME: DIVIJA V AGARWAL

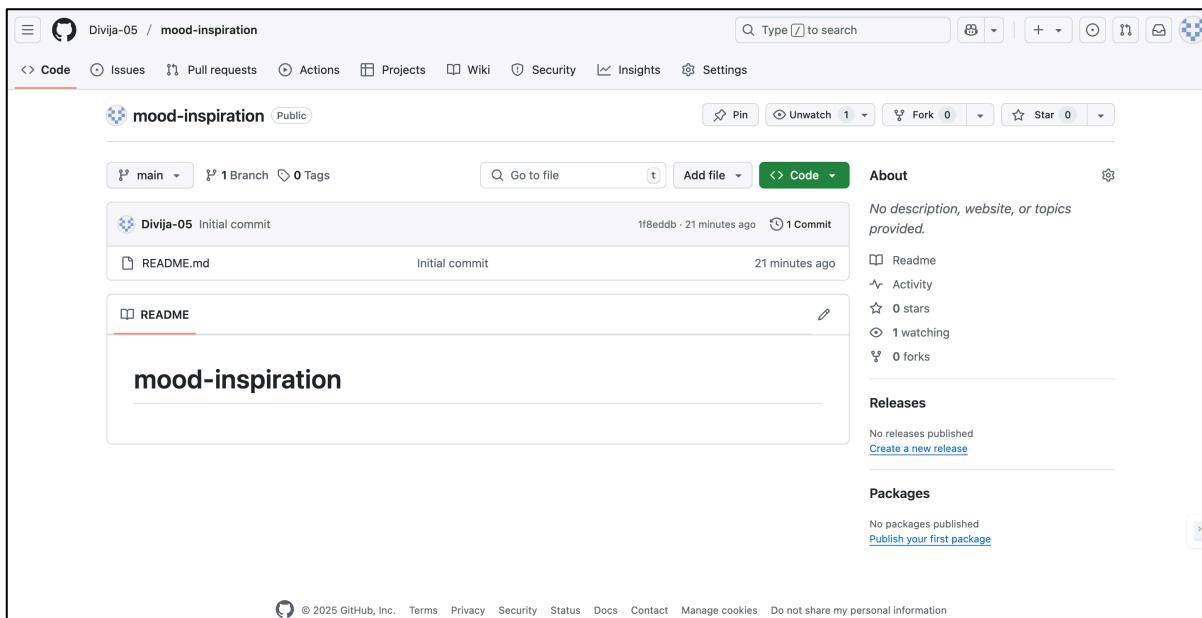
REG NO: 22BDS0329

INDEX

S.No	Topic	Page Number
1	GitHub Setup <ul style="list-style-type: none"> • Repository creation • Flask app creation • Pushing the web app on GitHub 	3-7
2	Docker and Kubernetes Setup <ul style="list-style-type: none"> • Installing docker • Setting up Kubernetes 	8-9
3	Deployment <ul style="list-style-type: none"> • Creating image on docker • Deploying using Kubernetes • Creating Kubernetes secret 	10-14
4	Auto Scaling	15
5	Rolling Updates and Rollbacks	16
6	Logging	17
7	Test Scenarios	18-21
Appendix		
A	Repository Link (Appendix B & C included)	22
B	Demo Video	22
C	Source Code	22

1. GitHub Setup

Step 1: Create a public/private GitHub repository



Step 2: Cloning the Repository to the local system

```
git clone https://github.com/Divija-05/mood-inspiration.git
```

```
cd mood-inspiration
```

```
[divijaagarwal@Divijas-MacBook-Air coding % git clone https://github.com/Divija-05/mood-inspiration.git ]
Cloning into 'mood-inspiration'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
[divijaagarwal@Divijas-MacBook-Air coding % cd mood-inspiration
[divijaagarwal@Divijas-MacBook-Air mood-inspiration % ]]
```

Step 3: Setting up the project in the GitHub folder

```
python3 -m venv venv
```

```
source venv/bin/activate
```

```
pip3 install flask
```

```
[divijaagarwal@Divijas-MacBook-Air mood-inspiration % python3 -m venv venv
[divijaagarwal@Divijas-MacBook-Air mood-inspiration % source venv/bin/activate
(venv) divijaagarwal@Divijas-MacBook-Air mood-inspiration % pip3 install flask
Collecting flask
  Downloading flask-3.1.0-py3-none-any.whl (102 kB)
    █████████████████████████████████████████ | 102 kB 1.3 MB/s
Collecting click>=8.1.3
  Downloading click-8.1.8-py3-none-any.whl (98 kB)
    █████████████████████████████████████████ | 98 kB 1.3 MB/s
Collecting importlib-metadata>=3.6
  Using cached importlib_metadata-8.6.1-py3-none-any.whl (26 kB)
Collecting itsdangerous>=2.2
  Downloading itsdangerous-2.2.0-py3-none-any.whl (16 kB)
Collecting blinker>=1.9
  Downloading blinker-1.9.0-py3-none-any.whl (8.5 kB)
Collecting Jinja2>=3.1.2
  Downloading jinja2-3.1.6-py3-none-any.whl (134 kB)
    █████████████████████████████████████████ | 134 kB 1.2 MB/s
Collecting Werkzeug>=3.1
  Downloading werkzeug-3.1.3-py3-none-any.whl (224 kB)
    █████████████████████████████████████████ | 224 kB 1.2 MB/s
Collecting zipp>=3.20
  Using cached zipp-3.21.0-py3-none-any.whl (9.6 kB)
Collecting MarkupSafe>=2.0
  Downloading MarkupSafe-3.0.2-cp39-cp39-macosx_11_0_arm64.whl (12 kB)
Installing collected packages: zipp, MarkupSafe, Werkzeug, Jinja2, itsdangerous, importlib-metadata, click, blinker, flask
Successfully installed Jinja2-3.1.6 MarkupSafe-3.0.2 Werkzeug-3.1.3 blinker-1.9.0 click-8.1.8 flask-3.1.0 importlib-metadata-8.6.1 itsdangerous-2.2.0 zipp-3.21.0
```

Step 4: Create the flask app

```
EXPLORER   ...   mood-inspiration
app.py >   .gitignore   quotes.json   index.html   .env
MOOD-INSPIRATION
static
  data
    activities.json
    quotes.json
script.js
style.css
templates
  index.html
  results.html
venv
.env
.gitignore
~$umentation...
app.py
documentation.docx
README.md
requirements.txt

app.py > ...
1  from flask import Flask, render_template, request, jsonify
2  import random
3  import json
4  import os
5  import openai
6
7  app = Flask(__name__)
8
9  from dotenv import load_dotenv # Import dotenv
10
11 # Load environment variables from .env file
12 load_dotenv()
13
14 # Get OpenAI API key from environment variable
15 openai.api_key = os.getenv("OPENAI_API_KEY")
16
17 if not openai.api_key:
18     raise ValueError("Missing OpenAI API Key! Make sure it's set in the .env file.")
19
20
21 # Load quotes data
22 with open('static/data/quotes.json', 'r') as f:
23     quotes_data = json.load(f)
24
25 # Load activities data
26 with open('static/data/activities.json', 'r') as f:
27     activities_data = json.load(f)

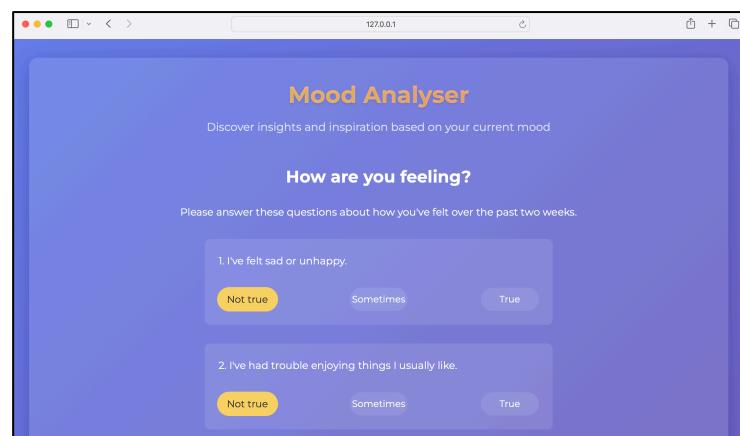
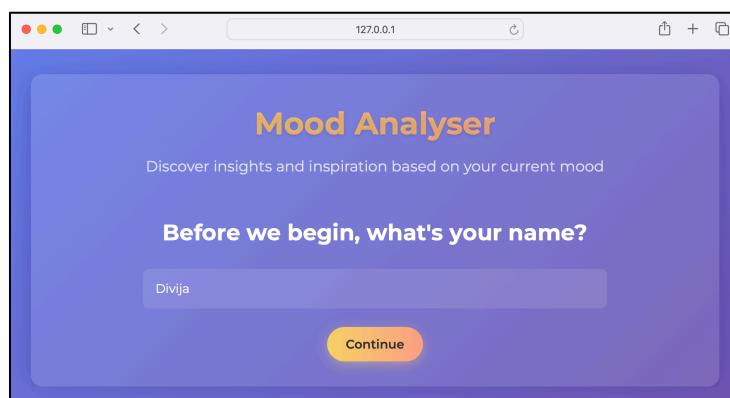
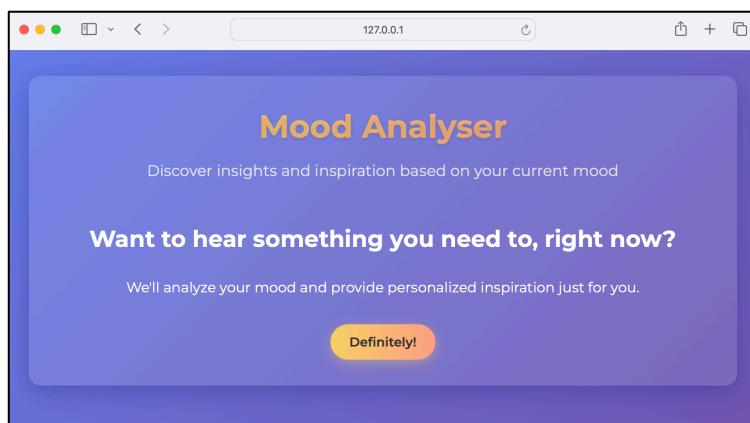
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS
(venv) divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

(for complete code refer Appendix)

Step 5: Run and check the app if it is running

Python3 app.py

Running <http://127.0.0.1:5000> on the browser displays the following :



8. I've felt anxious, worried, or fearful.

Not true Sometimes True

9. I've had sudden attacks of intense fear or panic.

Not true Sometimes True

10. My mood has changed rapidly for no apparent reason.

Not true Sometimes True

Note: This is not a diagnostic tool. If you're experiencing severe distress, please contact a mental health professional.

Analyse my Mood

Mood Analyser

Your personalized mood insights

Hello, Divija!

Here's your mood analysis and personalized inspiration.

Your Mood in a Nutshell

Primarily Trust

Your responses indicate that you're experiencing Trust as your primary emotion right now.

"Trust is the glue of life. It's the most essential ingredient in effective communication."

— Stephen Covey

Your Mood Profile

Trust	100%
Anticipation	80%
Surprise	20%
Joy	10%
Anger	10%
Sadness	0%
Fear	0%
Disgust	0%

Things You Could Try Today

- Trust Exercise**
Delegate a task to someone else without micromanaging them.
- Reflect on Trust**
Write down three people you trust completely and why you trust them.
- Trust Yourself**
Make a decision you've been postponing and commit to it.

Start Over

```

127.0.0.1 - - [29/Mar/2025 16:08:10] "POST /analyze HTTP/1.1" 200 -
127.0.0.1 - - [29/Mar/2025 16:08:10] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [29/Mar/2025 16:08:13] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [29/Mar/2025 16:08:13] "GET /static/script.js HTTP/1.1" 304 -
127.0.0.1 - - [29/Mar/2025 16:08:13] "GET /static/script.js HTTP/1.1" 304 -
127.0.0.1 - - [29/Mar/2025 16:08:13] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [29/Mar/2025 16:10:30] "POST /analyze HTTP/1.1" 200 -
127.0.0.1 - - [29/Mar/2025 16:10:30] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [29/Mar/2025 16:11:38] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [29/Mar/2025 16:11:38] "GET /static/style.css HTTP/1.1" 200 -
127.0.0.1 - - [29/Mar/2025 16:11:38] "GET /static/script.js HTTP/1.1" 200 -
127.0.0.1 - - [29/Mar/2025 16:11:39] "GET /favicon.ico HTTP/1.1" 404 -

```

Step 6: Setting up gitignore and pushing the code to repo

```
echo "venv/" >> .gitignore
```

```
echo "__pycache__/" >> .gitignore
```

(Refer appendix for full gitignore file)

```

[(venv) divijaagarwal@Divijas-MacBook-Air mood-inspiration % git add .
[(venv) divijaagarwal@Divijas-MacBook-Air mood-inspiration % git commit -m "Initial Flask app setup"
[main 9ba9e46] Initial Flask app setup
 4 files changed, 12 insertions(+)
  create mode 100644 .gitignore
  create mode 100644 app.py
  create mode 100644 documentation.docx
  create mode 100644 ~$cumentation.docx
[(venv) divijaagarwal@Divijas-MacBook-Air mood-inspiration % git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 949.38 KiB | 36.51 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com//Divija-05/mood-inspiration.git
  1f8eddb..9ba9e46  main -> main
(venv) divijaagarwal@Divijas-MacBook-Air mood-inspiration %

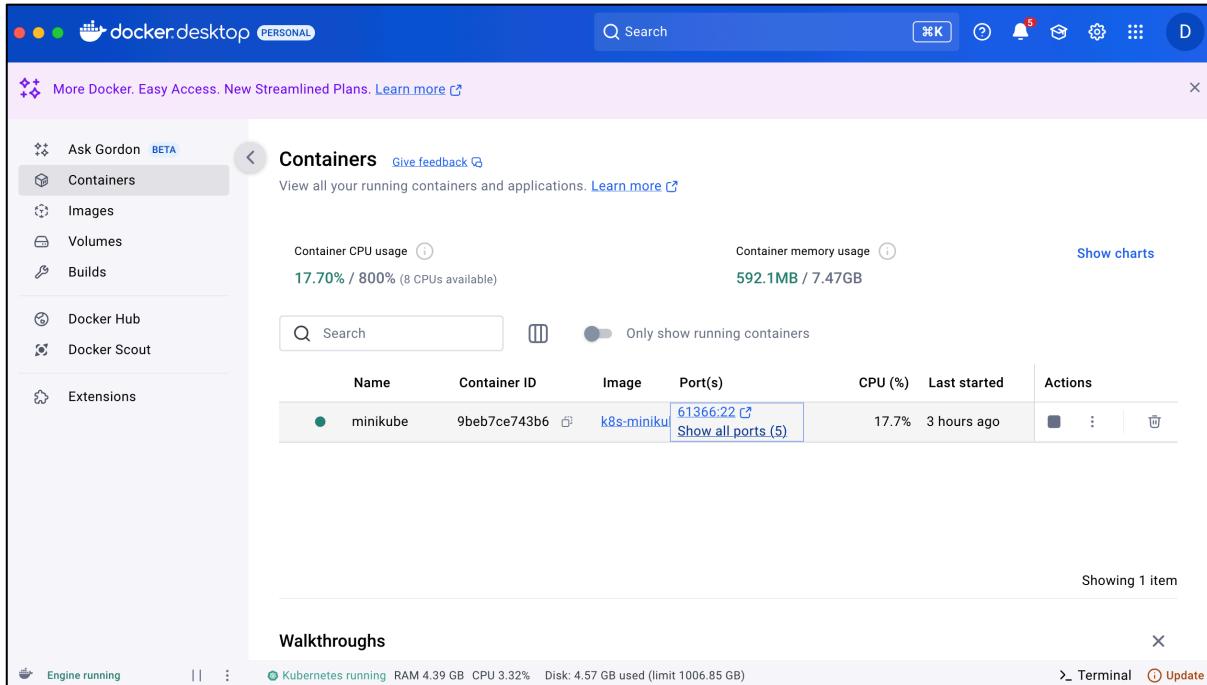
```

2. Setting up Docker and Kubernetes

Docker

Installing docker and running

```
(venv) divijaagarwal@Divijas-MacBook-Air mood-inspiration % docker --version
Docker version 28.0.1, build 068a01e
```



Kubernetes

Step 1: Install homebrew if not installed and then install minikube

For installing brew : /bin/bash -c "\$(curl -fsSL

[https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh\)](https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh))"

For installing minikube : brew install minikube

```
==> Installing minikube
==> Pouring minikube--1.35.0.arm64_sonoma.bottle.tar.gz
==> Caveats
zsh completions have been installed to:
  /opt/homebrew/share/zsh/site-functions
==> Summary
🍺 /opt/homebrew/Cellar/minikube/1.35.0: 10 files, 118.2MB
==> Running `brew cleanup minikube`...
Disable this behaviour by setting HOMEBREW_NO_INSTALL_CLEANUP.
Hide these hints with HOMEBREW_NO_ENV_HINTS (see `man brew`).
==> Caveats
==> minikube
zsh completions have been installed to:
  /opt/homebrew/share/zsh/site-functions
divijaagarwal@Divijas-MacBook-Air ~ %
```

Step 2 : Verification**Start minikube :** minikube start --driver=docker**Check if minikube is working :** minikube status**Set kubectl to use minikube :** kubectl config use-context minikube**Verify Kubernetes is working:** kubectl get nodes

```
divijaagarwal@Divijas-MacBook-Air ~ % minikube start --driver=docker

😄 minikube v1.35.0 on Darwin 14.6 (arm64)
💡 Using the docker driver based on user configuration
🚀 Using Docker Desktop driver with root privileges
👍 Starting "minikube" primary control-plane node in "minikube" cluster
🚜 Pulling base image v0.0.46 ...
💻 Downloading Kubernetes v1.32.0 preload ...
  > preloaded-images-k8s-v18-v1...: 314.92 MiB / 314.92 MiB 100.00% 392.75
  > gcr.io/k8s-minikube/kicbase...: 452.84 MiB / 452.84 MiB 100.00% 289.61
🔥 Creating docker container (CPUs=2, Memory=4000MB) ...
🐳 Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
  ▪ Generating certificates and keys ...
  ▪ Booting up control plane ...
  ▪ Configuring RBAC rules ...
🔧 Configuring bridge CNI (Container Networking Interface) ...
🔍 Verifying Kubernetes components...
  ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
💡 Enabled addons: storage-provisioner, default-storageclass
🎉 Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
divijaagarwal@Divijas-MacBook-Air ~ % kubectl config use-context minikube
Switched to context "minikube".
divijaagarwal@Divijas-MacBook-Air ~ % kubectl get nodes
NAME      STATUS    ROLES      AGE     VERSION
minikube  Ready     control-plane  159m   v1.32.0
divijaagarwal@Divijas-MacBook-Air ~ %
```

3. Deployment

Step 1: Building docker image

touch Dockerfile

nano Dockerfile

docker build -t mood-inspiration-app .

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % ls
README.md          documentation.docx      static           venv
app.py             requirements.txt       templates      ~$umentation.docx
divijaagarwal@Divijas-MacBook-Air mood-inspiration % touch Dockerfile
divijaagarwal@Divijas-MacBook-Air mood-inspiration % nano Dockerfile
```

```
UW PICO 5.09                                         File: Dockerfile

# Use a lightweight Python image
FROM python:3.9-slim

# Set the working directory
WORKDIR /app

# Copy the application files
COPY . /app

# Install dependencies
RUN pip install --no-cache-dir -r requirements.txt

# Expose the port
EXPOSE 5000

# Command to run the app
CMD ["python", "app.py"]
```

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % docker build -t mood-inspiration-app .

[+] Building 1.2s (9/9) FINISHED                               docker:desktop-linux
=> [internal] load build definition from Dockerfile          0.0s
=> => transferring dockerfile: 335B                          0.0s
=> [internal] load metadata for docker.io/library/python:3.9-slim 1.0s
=> [internal] load .dockerignore                            0.0s
=> => transferring context: 2B                           0.0s
=> [1/4] FROM docker.io/library/python:3.9-slim@sha256:e52ca5f579cc58fe 0.0s
=> [internal] load build context                          0.1s
=> => transferring context: 186.81kB                      0.1s
=> CACHED [2/4] WORKDIR /app                            0.0s
=> CACHED [3/4] COPY . /app                            0.0s
=> CACHED [4/4] RUN pip install --no-cache-dir -r requirements.txt 0.0s
=> exporting to image                                     0.0s
=> => exporting layers                                    0.0s
=> => writing image sha256:d091d9210a98493b3970cd5346760263bb0dcaf397e5 0.0s
=> => naming to docker.io/library/mood-inspiration-app 0.0s
divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

Step 2: Check if the docker image is built

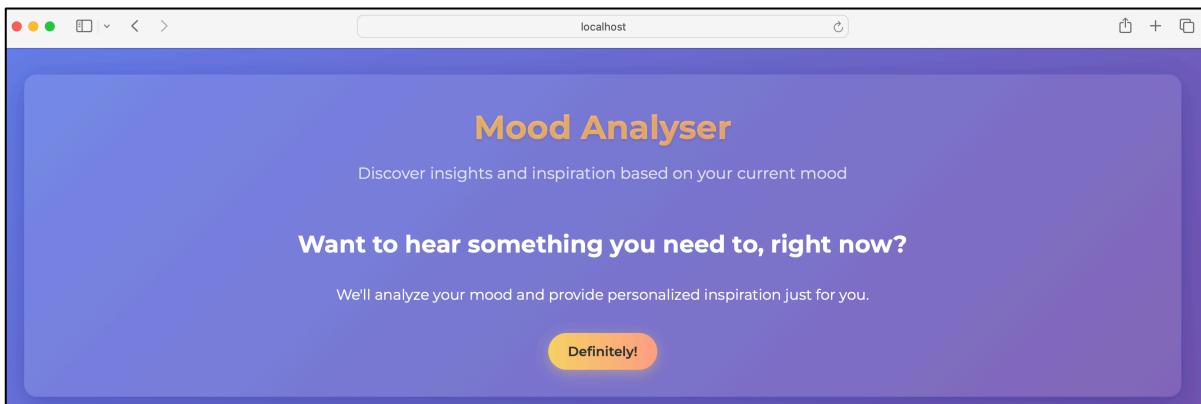
docker images

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % docker images
REPOSITORY          IMAGE ID      CREATED        SIZE
mood-inspiration-app   d091d9210a98  3 minutes ago  340MB
docker/desktop-kubernetes   ca2d12959459  5 weeks ago  400MB
debian               ca2d12959459  5 weeks ago  400MB
registry.k8s.io/kube-apiserver   6417e1437b6d  6 weeks ago  93.9MB
registry.k8s.io/kube-scheduler   82dfa03f692f  6 weeks ago  67.9MB
registry.k8s.io/kube-controller-manager   3c9285acf2f  6 weeks ago  87.2MB
registry.k8s.io/kube-proxy       e5aac5df76d9  6 weeks ago  97.1MB
gcr.io/k8s-minikube/kicbase    0434cf58b6db  2 months ago  1.22GB
registry.k8s.io/etcld        3fc9d4aa817a  6 months ago  142MB
registry.k8s.io/coredns/coredns  2f6c962e7b83  8 months ago  60.2MB
registry.k8s.io/pause         afb61768ce38  10 months ago  514KB
docker/desktop-vpnkit-controller  3750dfec169f  22 months ago  35MB
docker/desktop-storage-provisioner  c027a58fa0bb  3 years ago  39.8MB
divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

Step 4: Check on local host using docker

docker run -p 5001:5000 mood-inspiration-app

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % docker run -p 5001:5000 mood-inspiration-app
* Serving Flask app 'app' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on all addresses.
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://172.17.0.2:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 958-557-914
192.168.65.1 - - [29/Mar/2025 16:07:41] "GET / HTTP/1.1" 200 -
192.168.65.1 - - [29/Mar/2025 16:07:41] "GET /static/style.css HTTP/1.1" 200 -
192.168.65.1 - - [29/Mar/2025 16:07:41] "GET /static/script.js HTTP/1.1" 200 -
192.168.65.1 - - [29/Mar/2025 16:07:41] "GET /favicon.ico HTTP/1.1" 404 -
```



Step 5: Load the image into minikube

minikube image load mood-inspiration-app

eval \$(minikube docker-env)

```
docker build -t mood-inspiration-app .
```

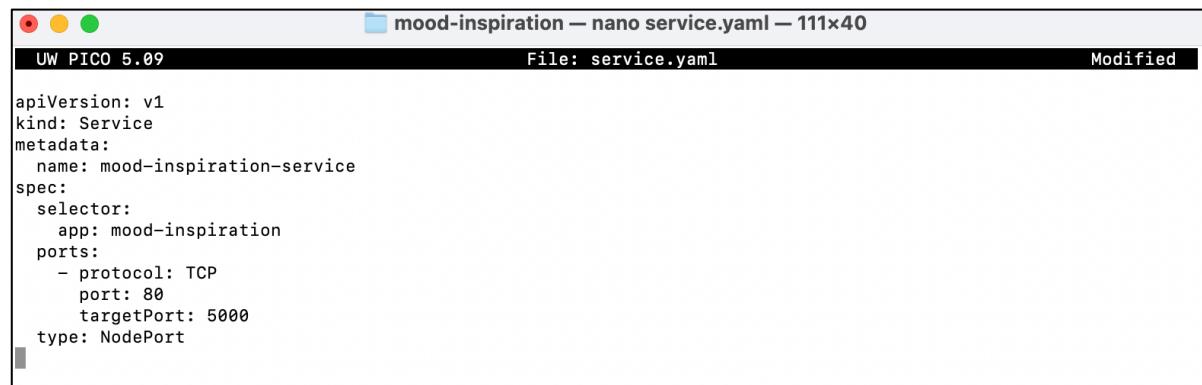
```
docker images | grep mood-inspiration-app
```

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % minikube image load mood-inspiration-app
divijaagarwal@Divijas-MacBook-Air mood-inspiration %

divijaagarwal@Divijas-MacBook-Air mood-inspiration % eval $(minikube docker-env)
divijaagarwal@Divijas-MacBook-Air mood-inspiration % docker build -t mood-inspiration-app .
[+] Building 84.1s (10/10) FINISHED
=> [internal] load build definition from Dockerfile                               docker:default
=> => transferring dockerfile: 335B                                              0.0s
=> [internal] load metadata for docker.io/library/python:3.9-slim                8.0s
=> [auth] library/python:pull token for registry-1.docker.io                      0.0s
=> [internal] load .dockerignore                                                 0.0s
=> => transferring context: 2B                                                 0.0s
=> [1/4] FROM docker.io/library/python:3.9-slim@sha256:e52ca5f579cc58fed41efccb55a0ed5dccf6c7a156cba76a  0.0s
=> => resolve docker.io/library/python:3.9-slim@sha256:e52ca5f579cc58fed41efccb55a0ed5dccf6c7a156cba76a  0.0s
=> => sha256:e52ca5f579cc58fed41efccb55a0ed5dccf6c7a156cba76a@sha256:4ef9ab751a99  0.0s
=> => sha256:c4cad0d3cd2e5af9a3e94c30e651f346930da73ce90a4b61657cee9e293b7fbc  1.75kB / 1.75kB  0.0s
=> => sha256:87c931ea7aa3ac24f3228671c64e77d3b160504164339603f5687d9e090a3a4c  5.29kB / 5.29kB  0.0s
=> [internal] load build context                                                 0.4s
=> => transferring context: 45.06MB                                            0.4s
=> [2/4] WORKDIR /app                                                       0.0s
=> [3/4] COPY . /app                                                       0.2s
=> [4/4] RUN pip install --no-cache-dir -r requirements.txt                  74.8s
=> exporting to image                                                       0.7s
=> => exporting layers                                                       0.7s
=> => writing image sha256:4ef9ab751a9985b090c80cd6739764345d28b38fcad000adcbb05d607c27b5b0  0.0s
=> => naming to docker.io/library/mood-inspiration-app                         0.0s
divijaagarwal@Divijas-MacBook-Air mood-inspiration % docker images | grep mood-inspiration-app
mood-inspiration-app          latest      4ef9ab751a99   24 seconds ago   342MB
divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

Step 6: Create service.yaml

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % touch service.yaml
divijaagarwal@Divijas-MacBook-Air mood-inspiration % nano service.yaml
```



```
mood-inspiration — nano service.yaml — 111x40
File: service.yaml
Modified

UW PICO 5.09

apiVersion: v1
kind: Service
metadata:
  name: mood-inspiration-service
spec:
  selector:
    app: mood-inspiration
  ports:
    - protocol: TCP
      port: 80
      targetPort: 5000
  type: NodePort
```

Step 7: Create Kubernetes deployment (create deployment.yaml)

```
touch deployment.yaml
```

```
nano deployment.yaml
```

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: mood-inspiration-deployment
spec:
  replicas: 3 # Updated to 3 replicas as per the project doc
  selector:
    matchLabels:
      app: mood-inspiration
  template:
    metadata:
      labels:
        app: mood-inspiration
    spec:
      containers:
        - name: mood-inspiration-app
          image: mood-inspiration-app # This should match the name you used in `minikube image load`
          ports:
            - containerPort: 5000

```

Step 6: Apply the deployment

kubectl apply -f deployment.yaml

kubectl get pods

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl apply -f deployment.yaml
deployment.apps/mood-inspiration-deployment created
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get pods
```

```
[divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get pods
NAME                      READY   STATUS    RESTARTS   AGE
mood-inspiration-deployment-8648967fbb-4kx79   1/1     Running   0          21s
mood-inspiration-deployment-8648967fbb-7gn87   1/1     Running   0          21s
mood-inspiration-deployment-8648967fbb-mfz9j    1/1     Running   0          21s
divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

Step 7: Expose the app and getting URL

kubectl apply -f service.yaml

kubectl get services

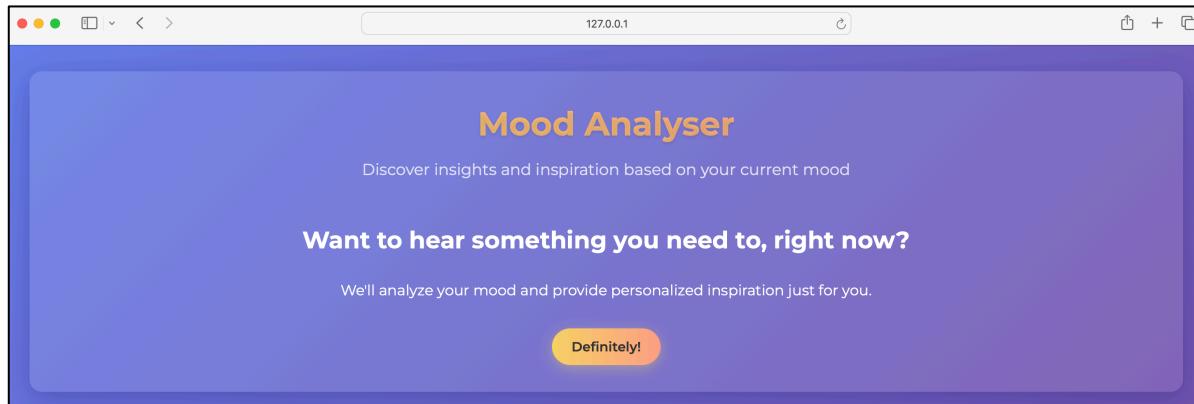
minikube service <your-service-name> --url (get the service name from the previous command)

minikube service mood-inspiration-service --url

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl apply -f service.yaml
service/mood-inspiration-service created
```

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get services
NAME           TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes     ClusterIP  10.96.0.1      <none>         443/TCP      4h27m
mood-inspiration-service   NodePort   10.98.193.102  <none>         80:31320/TCP  75m
divijaagarwal@Divijas-MacBook-Air mood-inspiration % minikube service mood-inspiration-service --url
http://127.0.0.1:63109
! Because you are using a Docker driver on darwin, the terminal needs to be open to run it.
```

<http://127.0.0.1:63109>



Step 6: Create Kubernetes Secret

```
echo -n "your_openai_api_key_here" | base64
```

```
mood-inspiration — nano secret.yaml — 111x40
File: secret.yaml
apiVersion: v1
kind: Secret
metadata:
  name: openai-secret
type: Opaque
data:
  OPENAI_API_KEY: c2stchJvai1qaVM4NWRhTUVUNmdZZ2VWRVVNT2xpb1JHVmNtdjN00GphWTIwbWlidTZpaS1HaUZmUC02NHU1YXdfcUo
```

```
kubectl apply -f secret.yaml
```

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl apply -f secret.yaml
secret/openai-secret created
divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

3. Auto Scaling

Step 1: Creating the metrics-server.yaml

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: metrics-server
  namespace: kube-system
  labels:
    k8s-app: metrics-server
spec:
  selector:
    matchLabels:
      k8s-app: metrics-server
  template:
    metadata:
      labels:
        k8s-app: metrics-server
    spec:
      containers:
        - name: metrics-server
          image: registry.k8s.io/metrics-server/metrics-server:v0.6.3
          imagePullPolicy: IfNotPresent
          args:
            - --cert-dir=/tmp
            - --secure-port=10250
            - --kubelet-preferred-address-types=InternalIP
            - --kubelet-insecure-tls
          ports:
            - containerPort: 10250
              name: https
              protocol: TCP
          livenessProbe:
            httpGet:
              path: /livez
              port: 10250
              scheme: HTTPS
          readinessProbe:
            httpGet:
              path: /readyz
              port: 10250
              scheme: HTTPS

```

Step 2: Checking metrics server status

kubectl get deployment metrics-server -n kube-system

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
metrics-server	1/1	1	1	10m

Step 2: Applying autoscaling [using CLI (kubectl autoscale) instead of hpa.yaml]

kubectl autoscale deployment mood-inspiration-deployment --cpu-percent=50 --min=2 --max=5

Verify : kubectl get hpa

NAME	REFERENCE	TARGETS	MINPODS	MAXPODS	REPLICAS	AGE
mood-inspiration-deployment	Deployment/mood-inspiration-deployment	cpu: <unknown>/50%	2	5	3	21s

Step 3: Monitoring the pods

NAME	READY	STATUS	RESTARTS	AGE
mood-inspiration-deployment-8648967fbb-4kx79	1/1	Running	0	56m
mood-inspiration-deployment-8648967fbb-7gn87	1/1	Running	0	56m
mood-inspiration-deployment-8648967fbb-mfz9j	1/1	Running	0	56m

4. Rolling updates and rollback

Step 1: Check current deployment

```
kubectl get deployment mood-inspiration-deployment -o yaml | grep image  
- image: mood-inspiration-app
```

Step 2: Build and push a new image

```
docker build -t your-dockerhub-username/mood-inspiration-app:v2 .  
docker push your-dockerhub-username/mood-inspiration-app:v2
```

Step 3: Performing a rolling update

```
kubectl set image deployment/mood-inspiration-deployment mood-inspiration-app=your-  
dockerhub-username/mood-inspiration-app:v2
```

```
kubectl rollout status deployment mood-inspiration-deployment
```

```
kubectl get pods -w
```

Step 4: Verify the update

```
kubectl describe deployment mood-inspiration-deployment | grep Image  
kubectl get pods
```

Step 5: Simulate a failure and rollback

```
kubectl rollout undo deployment mood-inspiration-deployment  
kubectl rollout history deployment mood-inspiration-deployment
```

5. Logging

Step 1: Monitoring the pods

kubectl get pods

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
mood-inspiration-deployment-8648967fbb-4kx79   1/1     Running   0          65m
mood-inspiration-deployment-8648967fbb-7gn87   1/1     Running   0          65m
mood-inspiration-deployment-8648967fbb-mfz9j   1/1     Running   0          65m
divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

Step 2: Log a pod's details

kubectl logs mood-inspiration-deployment-8648967fbb-4kx79

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl logs 8648967fbb-4kx79
error: error from server (NotFound): pods "8648967fbb-4kx79" not found in namespace "default"
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl logs mood-inspiration-deployment-8648967fbb-4kx79
* Serving Flask app 'app' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on all addresses.
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://10.244.0.16:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 412-128-148
10.244.0.1 - - [29/Mar/2025 16:19:26] "GET /static/style.css HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 16:19:26] "GET /static/script.js HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 16:44:58] "GET /static/style.css HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 16:44:58] "GET /static/script.js HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 17:06:42] "GET / HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 17:06:42] "GET /favicon.ico HTTP/1.1" 404 -
divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

Step 2 : Live monitoring of a pod

kubectl logs -f mood-inspiration-deployment-8648967fbb-4kx79

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl logs -f mood-inspiration-deployment-8648967fbb-4kx79
* Serving Flask app 'app' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on all addresses.
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://10.244.0.16:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 412-128-148
10.244.0.1 - - [29/Mar/2025 16:19:26] "GET /static/style.css HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 16:19:26] "GET /static/script.js HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 16:44:58] "GET /static/style.css HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 16:44:58] "GET /static/script.js HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 17:06:42] "GET / HTTP/1.1" 200 -
10.244.0.1 - - [29/Mar/2025 17:06:42] "GET /favicon.ico HTTP/1.1" 404 -
```

6. Test Scenarios

1. Application Availability Tests

Test: Check if the application is accessible via the Kubernetes service.

Command:

```
kubectl get services
```

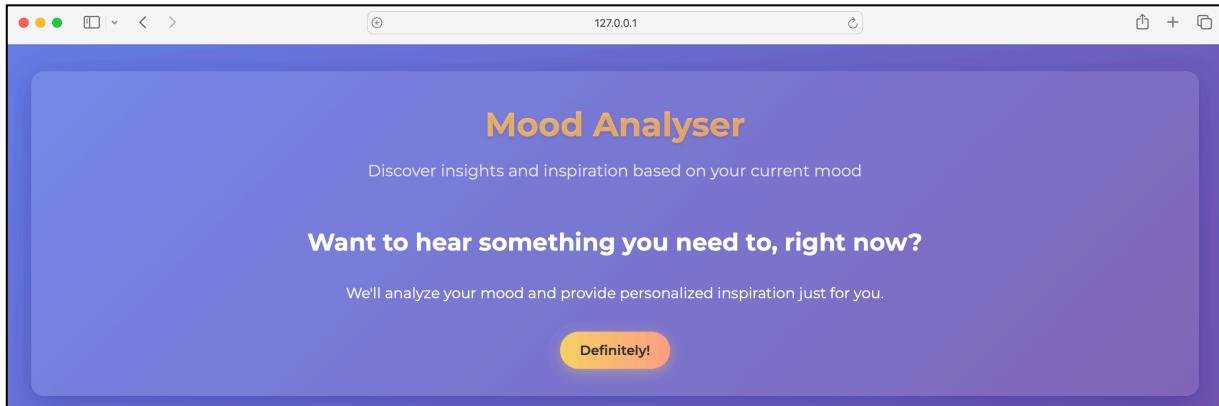
```
minikube service mood-inspiration-service --url
```

Expected Output: Should return the homepage or API response of the application.

Output:

<http://127.0.0.1:64358>

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get services
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)        AGE
kubernetes     ClusterIP  10.96.0.1   <none>       443/TCP      6h6m
mood-inspiration-service   NodePort   10.98.193.102 <none>       80:31320/TCP  174m
divijaagarwal@Divijas-MacBook-Air mood-inspiration % minikube service mood-inspiration-service --url
http://127.0.0.1:64358
! Because you are using a Docker driver on darwin, the terminal needs to be open to run it.
```



2. Scaling Tests

Test: Trigger high CPU usage to see if the **Horizontal Pod Autoscaler (HPA)** scales up pods.

Command:

```
kubectl get hpa
kubectl run --rm -it --image=busybox stress-test -- /bin/sh
```

Inside BusyBox shell:

```
while true; do wget -q -O- http://10.98.193.102:80; done
```

Expected Output: Number of pods should increase dynamically.

Output:

```
garwal@Divijas-MacBook-Air mood-inspiration % kubectl get hpa
NAME                      REFERENCE          TARGETS          MINPODS   MAXPODS   REPLICAS   AGE
mood-inspiration-deployment Deployment/mood-inspiration-deployment  cpu: <unknown>/50%  2          5          3          48m
divijaagarwal@Divijas-MacBook-Air mood-inspiration %

<input type="radio" id="q9-0" name="q9" value="0" required>
<label for="q9-0">Not true</label>
<input type="radio" id="q9-1" name="q9" value="1">
<label for="q9-1">Sometimes</label>
<input type="radio" id="q9-2" name="q9" value="2">
<label for="q9-2">True</label>
</div>
</div>

<div class="question">
<p>10. My mood has changed rapidly for no apparent reason.</p>
<div class="rating">
<input type="radio" id="q10-0" name="q10" value="0" required>
<label for="q10-0">Not true</label>
<input type="radio" id="q10-1" name="q10" value="1">
<label for="q10-1">Sometimes</label>
<input type="radio" id="q10-2" name="q10" value="2">
<label for="q10-2">True</label>
</div>
</div>

<div class="disclaimer">
<p>Note: This is not a diagnostic tool. If you're experiencing severe distress, please contact a mental health professional.</p>
</div>

<button type="submit" class="btn primary-btn">Analyse my Mood</button>
</form>
</div>
</main>
</div>

<script src="/static/script.js"></script>
</body>
</html>^C
/ # exit
Session ended, resume using 'kubectl attach stress-test -c stress-test -i -t' command when the pod is running
pod "stress-test" deleted
```

Verification: kubectl get pods -w

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get pods -w
NAME                  READY   STATUS    RESTARTS   AGE
mood-inspiration-deployment-8648967fbb-4kx79  1/1     Running   0          105m
mood-inspiration-deployment-8648967fbb-7gn87  1/1     Running   0          105m
mood-inspiration-deployment-8648967fbb-mfz9j   1/1     Running   0          105m
stress-test           1/1     Running   0          45s
```

3. Rolling Update & Rollback Test

Test: Perform a rolling update and verify zero downtime.

Command:

```
kubectl set image deployment/mood-inspiration-deployment mood-inspiration-app=your-dockerhub-username/mood-inspiration-app:v2
```

```
kubectl rollout status deployment/mood-inspiration-deployment
```

```
kubectl get pods -w
```

Expected Output: New version is deployed while keeping the app running.

Test: Rollback to the previous version in case of failure.

Command:

```
kubectl rollout undo deployment/mood-inspiration-deployment
kubectl rollout history deployment/mood-inspiration-deployment
```

Expected Output: Application reverts to the previous working version.

4. Pod Failure and Self-Healing Test

Test: Manually delete a pod and check if Kubernetes automatically recreates it.

Command:

```
kubectl delete pod <POD_NAME>
kubectl delete pod mood-inspiration-deployment-8648967fbb-4kx79
```

Verification: kubectl get pods -w

Expected Output: A new pod should be automatically created.

Output:

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get pods
  NAME                               READY   STATUS    RESTARTS   AGE
mood-inspiration-deployment-8648967fbb-4kx79   1/1     Running   0          111m
mood-inspiration-deployment-8648967fbb-7gn87   1/1     Running   0          111m
mood-inspiration-deployment-8648967fbb-mfz9j   1/1     Running   0          111m
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl delete pod mood-inspiration-deployment-8648967fbb-4kx79
pod "mood-inspiration-deployment-8648967fbb-4kx79" deleted
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get pods -w
  NAME                               READY   STATUS    RESTARTS   AGE
mood-inspiration-deployment-8648967fbb-6r897   1/1     Running   0          5s
mood-inspiration-deployment-8648967fbb-7gn87   1/1     Running   0          112m
mood-inspiration-deployment-8648967fbb-mfz9j   1/1     Running   0          112m
```

5. Logging Test

Test: Check if application logs are available.

Command:

```
kubectl get pods
```

```
kubectl logs mood-inspiration-deployment-8648967fbb-6r897
```

Expected Output: Should display application logs.

Output:

```
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
mood-inspiration-deployment-8648967fbb-6r897  1/1     Running   0          2m46s
mood-inspiration-deployment-8648967fbb-7gn87  1/1     Running   0          114m
mood-inspiration-deployment-8648967fbb-mfz9j   1/1     Running   0          114m
divijaagarwal@Divijas-MacBook-Air mood-inspiration % kubectl logs mood-inspiration-deployment-8648967fbb-6r897
* Serving Flask app 'app' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on all addresses.
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://10.244.0.40:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 763-309-311
divijaagarwal@Divijas-MacBook-Air mood-inspiration %
```

Appendix

Repository Link :

<https://github.com/Divija-05/mood-inspiration/>

Demo video and code in repository.