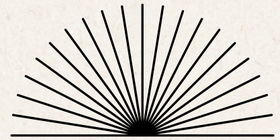
REAL-TIME EMOTION DETECTION USING FACIAL RECOGNITION AND EMOTION-BASED MOVIE RECOMMENDATION SYSTEM

PRESENTED BY:	PRN:
SUSHRUT NISAL	22070122227
SWARNIMA SINGH	22070122228
TIJIL PARAKH	22070122237
YAMINEE CHAUDHARY	22070122253



Agenda

01	Overview & Objectives
02	Architecture Diagram
03	Pipeline Flow
04	Challenges Faced
05	Lessons Learned
08	Steps

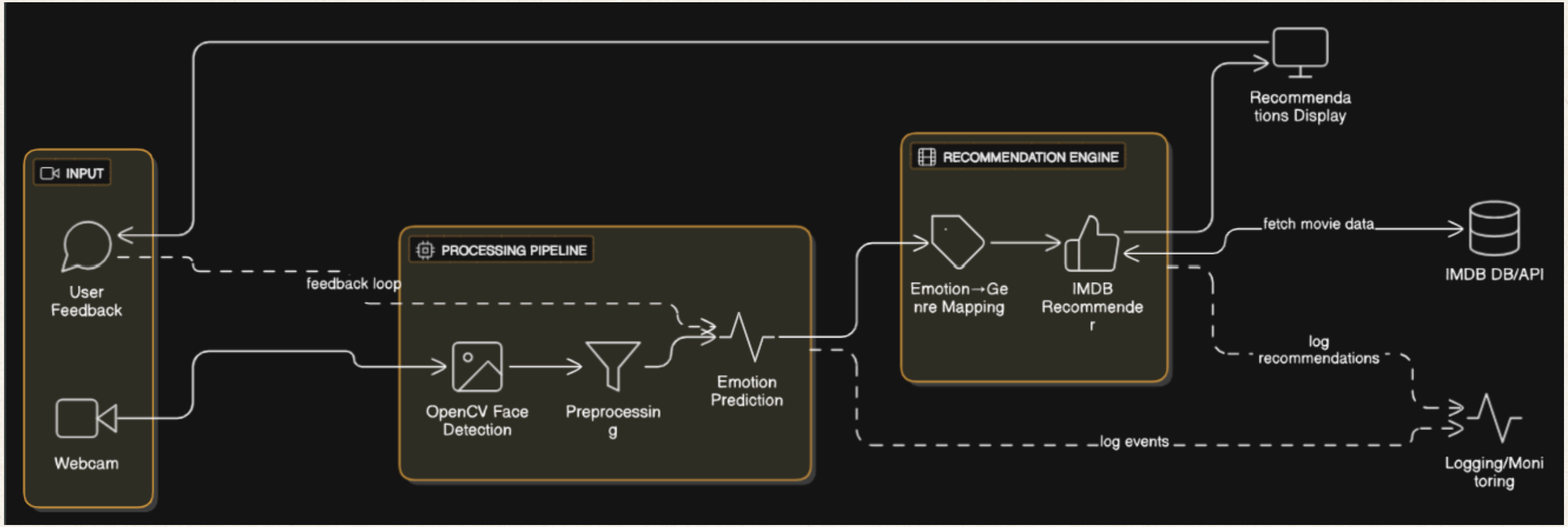
Overview

Emotion-aware computing has emerged as a key component in enhancing personalized user experiences. This project presents a real-time emotion detection system using facial recognition via convolutional neural networks (CNNs), coupled with a context-aware movie recommendation engine. The emotion detection module leverages the FER2013 dataset to classify human emotions through webcam input. Based on the identified emotional state, a movie recommendation engine suggests films aligned with the user's current mood by mapping emotions to genre-specific preferences. The model ensures fast and accurate emotion recognition through a pre-trained CNN architecture integrated with OpenCV for real-time performance. The recommender system utilizes the MovieLens dataset and applies a hybrid scoring function considering genre relevance, rating, and popularity to recommend personalized movie options.

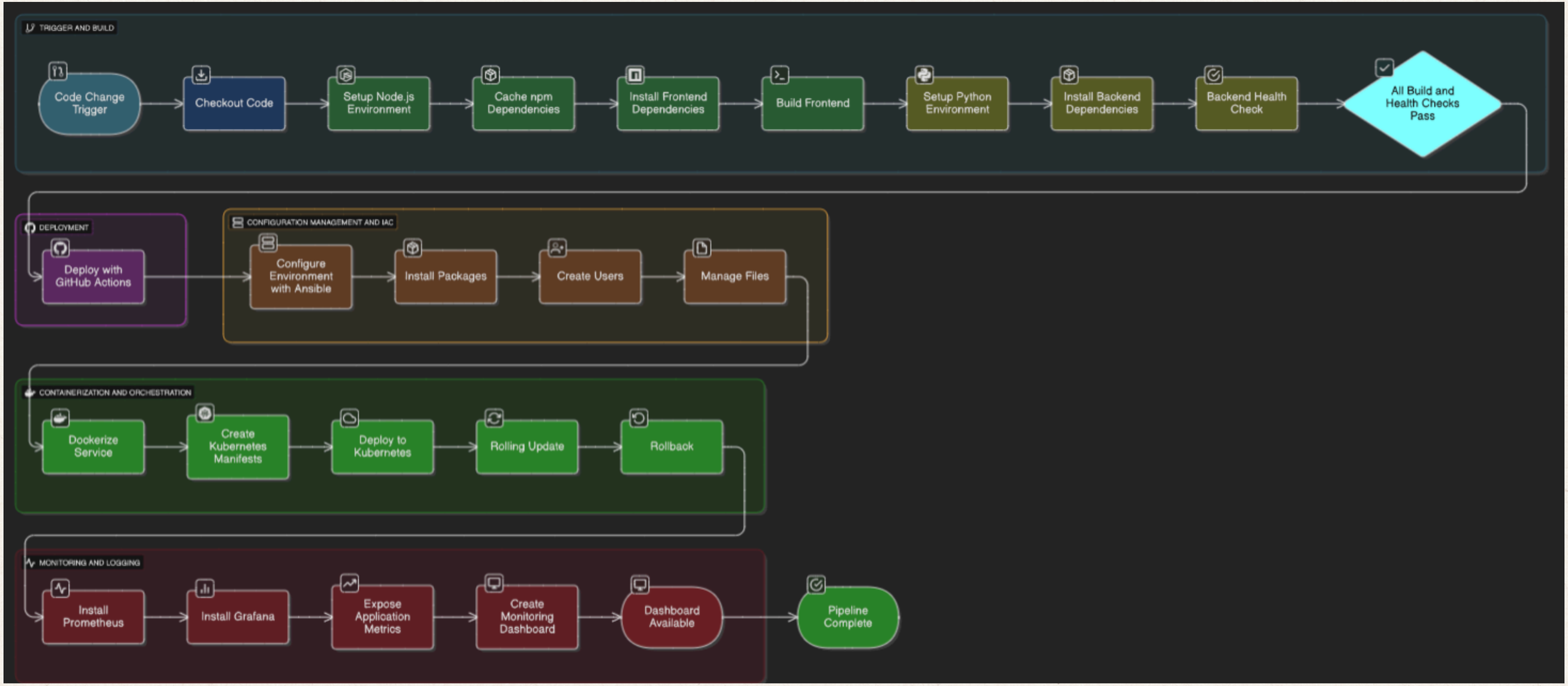
Objectives

- A real-time facial emotion detection module requires the implementation of pre-trained CNN technology from FER2013.
- Experts used a rule-based approach to link identified facial emotions with appropriate movie genres.
- A movie recommendation system will present recommendations from the MovieLens dataset by combining a scoring system which incorporates metrics related to average ratings and popularity along with genre relevance.
- The system evaluation occurs through webcam-based tests which measure the efficiency of its integrated components with user interaction.

Architecture Diagram



Pipeline Flow



Challenges Faced

- Integration Complexity: Connecting multiple modules (OpenCV, emotion prediction model, IMDB API) within the CI/CD pipeline caused dependency conflicts and version mismatches.
- Containerization Issues: While Dockerizing the application, ensuring that all Python libraries (OpenCV, TensorFlow, etc.) worked properly inside the container required several image rebuilds.
- Kubernetes Deployment Errors: YAML misconfigurations (ports, service selectors) initially prevented successful pod communication.
- Monitoring Setup: Configuring Prometheus exporters and Grafana dashboards to collect app metrics was challenging due to missing endpoints.
- Pipeline Debugging: GitHub Actions workflow often failed due to environment variable and permission issues for Docker Hub and Kubernetes cluster access.
- Runtime Environment Configuration: Using Ansible for environment setup required handling idempotency and dynamic inventory management.
- Data Privacy & Feedback Loop: Capturing webcam feed and storing logs safely needed additional care to avoid exposure of sensitive user data.

Lessons Learned

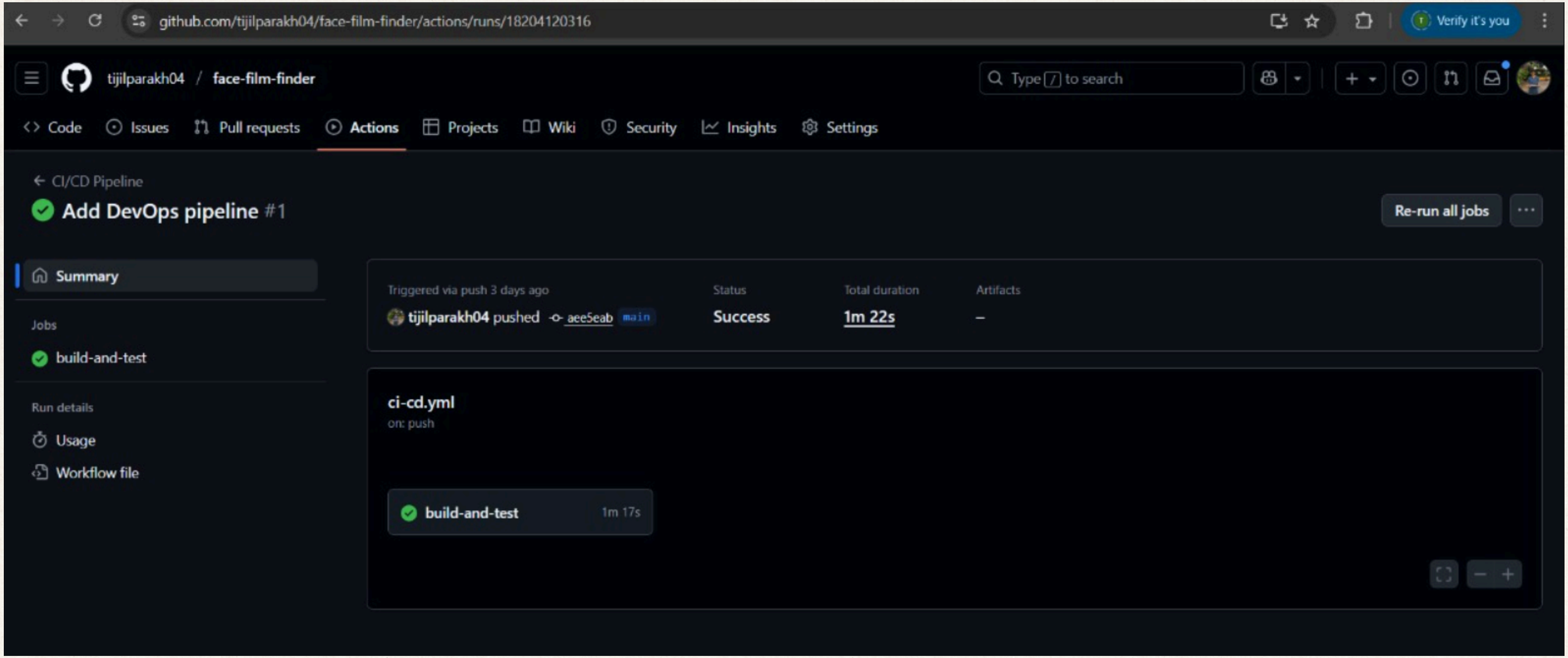
- End-to-End Automation: Understood how DevOps ensures faster, repeatable deployments using CI/CD and Infrastructure as Code.
- Container Efficiency: Learned how Docker and Kubernetes streamline scalability, rolling updates, and fault tolerance.
- Pipeline Design: Gained hands-on experience designing multi-stage GitHub Actions workflows including build, test, and deploy.
- Monitoring & Observability: Learned how Prometheus + Grafana enable proactive monitoring and real-time visualization of system health.
- Error Handling & Debugging: Improved troubleshooting skills by analyzing logs and pipeline failures.
- Collaboration Skills: Learned to coordinate effectively in a small DevOps team using GitHub for version control and task tracking.
- Security & Compliance: Understood the importance of handling user data securely and monitoring logs responsibly.

Step 1: Github pipeline

code -

ci-cd.yml

```
name: CI/CD Pipeline

on:
  push:
    branches: [ main ]
  pull_request:
    branches: [ main ]

jobs:
  build-and-test:
    runs-on: ubuntu-latest

    steps:
    - name: Checkout code
      uses: actions/checkout@v4

    - name: Setup Node.js
      uses: actions/setup-node@v4
      with:
        node-version: '18'
        cache: 'npm'

    - name: Install frontend dependencies
      run: npm install

    - name: Build frontend
      run: npm run build

    - name: Setup Python
      uses: actions/setup-python@v4
      with:
        python-version: '3.9'

    - name: Install backend dependencies
      run: |
        cd backend
        pip install -r requirements.txt

    - name: Test backend health
      run: |
        cd backend
        python -c "from app import app; print('Backend imports successfully')"
```


github.com/tijilparakh04/face-film-finder/actions/runs/18204120316
Verify it's you
tijilparakh04 / face-film-finder
Type / to search
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
CI/CD Pipeline
Add DevOps pipeline #1
Re-run all jobs
Summary
Jobs
build-and-test
Run details
Usage
Workflow file
Triggered via push 3 days ago
tijilparakh04 pushed aee5eab main
Status
Success
Total duration
1m 22s
Artifacts
–
ci-cd.yml
on: push
build-and-test
1m 17s

Step 2: Ansible

CODE:

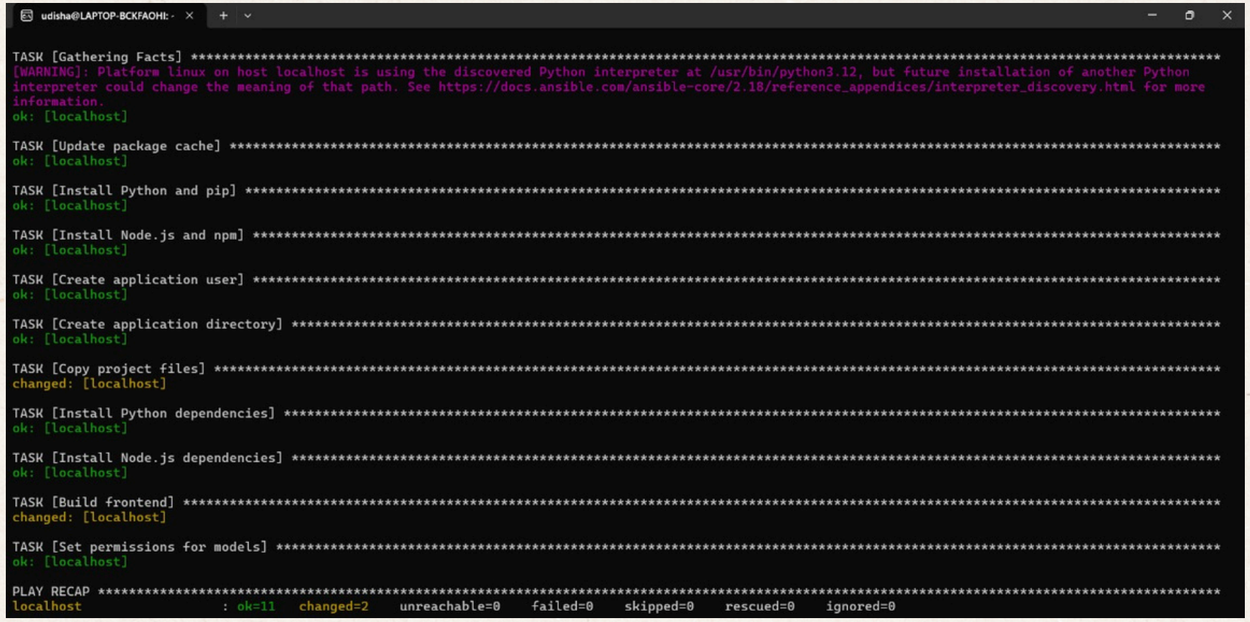
hosts.ini:

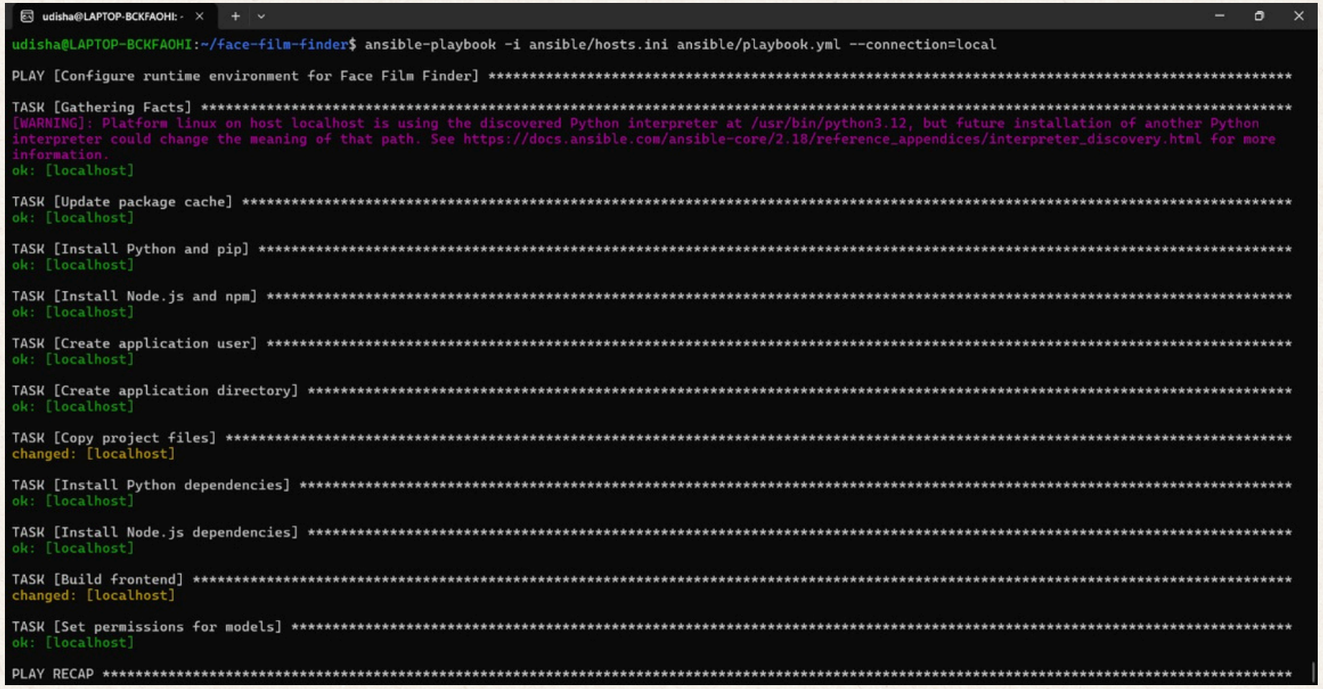
```
[local]
localhost ansible_connection=local

[webservers]
localhost
```

playbook.yml:

```
- name: Configure runtime environment for Face Film Finder
 hosts: webservers
 become: yes
 vars:
 app_user: facefilm
 app_dir: /opt/face-film-finder
 python_version: "3.9"
 tasks:
 - name: Update package cache
 apt:
 update_cache: yes
 when: ansible_os_family == 'Debian'
 - name: Install Python and pip
 apt:
 name:
 - python{{ python_version }}
 - python{{ python_version }}-pip
 - python{{ python_version }}-venv
 state: present
 when: ansible_os_family == 'Debian'
 - name: Install Node.js and npm
 apt:
 name:
 - nodejs
 - npm
 state: present
 when: ansible_os_family == 'Debian'
 - name: Create application user
 user:
 name: "{{ app_user }}"
 system: yes
 shell: /bin/bash
 home: "{{ app_dir }}"
 create_home: yes
 - name: Create application directory
 file:
 path: "{{ app_dir }}"
 state: directory
 owner: "{{ app_user }}"
 group: "{{ app_user }}"
 mode: '0755'
 - name: Copy project files
 copy:
 src: ../
 dest: "{{ app_dir }}"
 owner: "{{ app_user }}"
 group: "{{ app_user }}"
 ignore_errors: yes
 - name: Install Python dependencies
 pip:
 requirements: "{{ app_dir }}/backend/requirements.txt"
 virtualenv: "{{ app_dir }}/venv"
 virtualenv_python: python{{ python_version }}
 - name: Install Node.js dependencies
 npm:
 path: "{{ app_dir }}"
 state: present
 - name: Build frontend
 command: npm run build
 args:
 chdir: "{{ app_dir }}"
 become_user: "{{ app_user }}"
 - name: Set permissions for models
 file:
 path: "{{ app_dir }}/backend/models"
 owner: "{{ app_user }}"
 group: "{{ app_user }}"
 recurse: yes
```

Ansible


udisha@LAPTOP-BCKFAOHI: ~
TASK [Gathering Facts] **
[WARNING]: Platform linux on host localhost is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.18/reference_appendices/interpreter_discovery.html for more information.
ok: [localhost]
TASK [Update package cache] **
ok: [localhost]
TASK [Install Python and pip] **
ok: [localhost]
TASK [Install Node.js and npm] **
ok: [localhost]
TASK [Create application user] **
ok: [localhost]
TASK [Create application directory] **
ok: [localhost]
TASK [Copy project files] **
changed: [localhost]
TASK [Install Python dependencies] **
ok: [localhost]
TASK [Install Node.js dependencies] **
ok: [localhost]
TASK [Build frontend] **
changed: [localhost]
TASK [Set permissions for models] **
ok: [localhost]
PLAY RECAP **
localhost : ok=11 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0


udisha@LAPTOP-BCKFAOHI: ~
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ ansible-playbook -i ansible/hosts.ini ansible/playbook.yml --connection=local
PLAY [Configure runtime environment for Face Film Finder] **
TASK [Gathering Facts] **
[WARNING]: Platform linux on host localhost is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.18/reference_appendices/interpreter_discovery.html for more information.
ok: [localhost]
TASK [Update package cache] **
ok: [localhost]
TASK [Install Python and pip] **
ok: [localhost]
TASK [Install Node.js and npm] **
ok: [localhost]
TASK [Create application user] **
ok: [localhost]
TASK [Create application directory] **
ok: [localhost]
TASK [Copy project files] **
changed: [localhost]
TASK [Install Python dependencies] **
ok: [localhost]
TASK [Install Node.js dependencies] **
ok: [localhost]
TASK [Build frontend] **
changed: [localhost]
TASK [Set permissions for models] **
ok: [localhost]
PLAY RECAP **

Step 3: Docker and Kubernetes

codes:

1. backend/Dockerfile:

```
# Use official Python image as base
FROM python:3.9-slim

# Set working directory
WORKDIR /app

# Copy backend files
COPY backend/ /app/

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

# Expose port
EXPOSE 5000

# Run the app
CMD ["python", "app.py"]
```

2. src/Dockerfile:

```
# Use official Node.js image as base
FROM node:18-alpine

# Set working directory
WORKDIR /app

# Copy frontend files
COPY . .

# Install dependencies
RUN npm install

# Build the frontend
RUN npm run build

# Expose port (optional, for serving static files)
EXPOSE 3000

# Serve the built files using a simple server
RUN npm install -g serve

CMD ["serve", "-s", "dist", "-l", "3000"]
```

3. docker-compose.yml:

```
version: '3.8'

services:
  backend:
    build:
      context: .
      dockerfile: backend/Dockerfile
    ports:
      - "5000:5000"
    volumes:
      - ./backend:/app
    environment:
      - FLASK_ENV=development

  frontend:
    build:
      context: .
      dockerfile: src/Dockerfile
    ports:
      - "3000:3000"
    volumes:
      - ./:/app
    environment:
      - NODE_ENV=development
```

4. k8s/deployment.yml

```
apiVersion: apps/v1
kind: Deployment
metadata:
 name: face-film-finder-backend
spec:
 replicas: 2
 selector:
 matchLabels:
 app: backend
 template:
 metadata:
 labels:
 app: backend
 spec:
 containers:
 - name: backend
 image: face-film-finder-backend:latest
 ports:
 - containerPort: 5000
 env:
 - name: FLASK_ENV
 value: "production"
 resources:
 requests:
 memory: "512Mi"
 cpu: "250m"
 limits:
 memory: "1Gi"
 cpu: "500m"
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
 name: face-film-finder-frontend
spec:
 replicas: 2
 selector:
 matchLabels:
 app: frontend
 template:
 metadata:
 labels:
 app: frontend
 spec:
 containers:
 - name: frontend
 image: face-film-finder-frontend:latest
 ports:
 - containerPort: 3000
 env:
 - name: NODE_ENV
 value: "production"
 resources:
 requests:
 memory: "256Mi"
 cpu: "100m"
 limits:
 memory: "512Mi"
 cpu: "200m"
```

5. k8s/service.yml

```
apiVersion: v1
kind: Service
metadata:
  name: backend-service
spec:
  selector:
    app: backend
  ports:
    - protocol: TCP
      port: 5000
      targetPort: 5000
  type: ClusterIP
---
apiVersion: v1
kind: Service
metadata:
  name: frontend-service
spec:
  selector:
    app: frontend
  ports:
    - protocol: TCP
      port: 3000
      targetPort: 3000
  type: LoadBalancer
```

Docker containers running

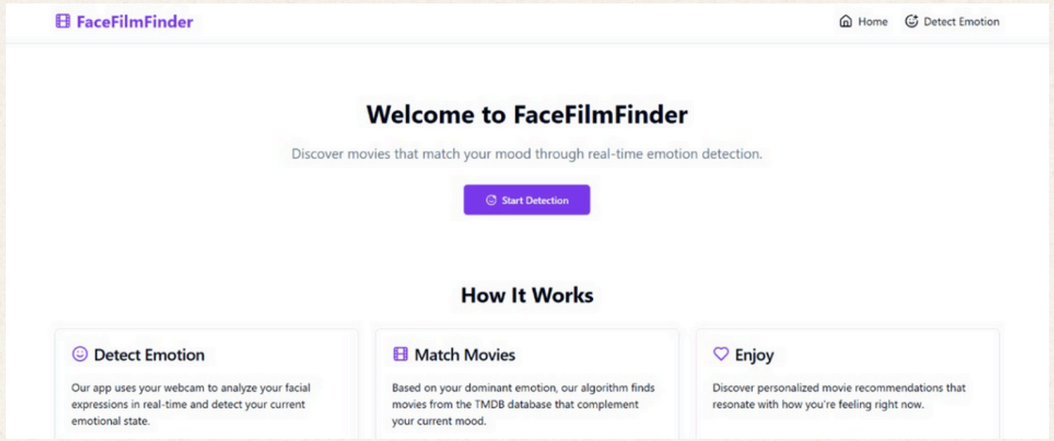
```
udisha@LAPTOP-BCKFAOHI:~/face-film-finder/k8s$ docker images
REPOSITORY                                TAG         IMAGE ID       CREATED          SIZE
face-film-finder-frontend                 latest      ba1290ef1a2d   8 minutes ago    958MB
face-film-finder-backend                  latest      ee181b8e95d1   10 minutes ago   2.51GB
registry.k8s.io/kube-apiserver            v1.34.0     90550c43ad2b   5 weeks ago      88MB
registry.k8s.io/kube-proxy                v1.34.0     df0860106674   5 weeks ago      71.9MB
registry.k8s.io/kube-scheduler            v1.34.0     46169d968e92   5 weeks ago      52.8MB
registry.k8s.io/kube-controller-manager   v1.34.0     a0af72f2ec6d   5 weeks ago      74.9MB
python                                    3.9-slim    abf3b76f84b9   8 weeks ago      120MB
registry.k8s.io/etcd                      3.6.4-0     5f1f5298c888   2 months ago     195MB
registry.k8s.io/pause                     3.10.1      cd073f4c5f6a   3 months ago     736kB
registry.k8s.io/coredns/coredns           v1.12.1     52546a367cc9   6 months ago     75MB
node                                      18-alpine   ee77c6cd7c18   6 months ago     127MB
gcr.io/k8s-minikube/storage-provisioner   v5          6e38f40d628d   4 years ago      31.5MB
udisha@LAPTOP-BCKFAOHI:~/face-film-finder/k8s$ kubectl apply -f k8s/deployment.yaml
error: the path "k8s/deployment.yaml" does not exist
udisha@LAPTOP-BCKFAOHI:~/face-film-finder/k8s$ cd ..
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl apply -f k8s/deployment.yaml
deployment.apps/face-film-finder-backend configured
deployment.apps/face-film-finder-frontend configured
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl apply -f k8s/service.yaml
service/backend-service unchanged
service/frontend-service unchanged
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl get pods
NAME                                      READY   STATUS              RESTARTS   AGE
face-film-finder-backend-57db8c4bdf-jxfmv    0/1     Running             0          15s
face-film-finder-backend-5f59785fff-grd46    0/1     ErrImageNeverPull   0          2m46s
face-film-finder-backend-5f59785fff-lk5xh    0/1     ErrImageNeverPull   0          2m46s
face-film-finder-frontend-575596f968-8qhg6   0/1     Terminating         0          2m46s
face-film-finder-frontend-575596f968-wn2j2   0/1     ErrImageNeverPull   0          2m46s
face-film-finder-frontend-6fdf97bc7d-lxtm4   1/1     Running             0          15s
face-film-finder-frontend-6fdf97bc7d-mcxb2   0/1     ContainerCreating   0          1s
```

Kubernetes pods running

```
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl get pods
NAME                                        READY   STATUS             RESTARTS      AGE
face-film-finder-backend-57db8c4bdf-jxfmv   0/1     Running            1 (7s ago)    25s
face-film-finder-backend-5f59785fff-grd46   0/1     ErrImageNeverPull  0             2m56s
face-film-finder-backend-5f59785fff-lk5xh   0/1     ErrImageNeverPull  0             2m56s
face-film-finder-frontend-575596f968-wn2j2  0/1     ErrImageNeverPull  0             2m56s
face-film-finder-frontend-6fdf97bc7d-lxtm4  1/1     Running            0             25s
face-film-finder-frontend-6fdf97bc7d-mcxb2  0/1     Running            0             11s
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ minikube service frontend-service
|-----------|-----------------|-------------|---------------------------|
| NAMESPACE |      NAME       | TARGET PORT |            URL            |
|-----------|-----------------|-------------|---------------------------|
| default   | frontend-service | 3000        | http://192.168.49.2:30495 |
|-----------|-----------------|-------------|---------------------------|
🏃  Starting tunnel for service frontend-service./
|-----------|-----------------|-------------|------------------------|
| NAMESPACE |      NAME       | TARGET PORT |          URL           |
|-----------|-----------------|-------------|------------------------|
| default   | frontend-service |             | http://127.0.0.1:40593 |
|-----------|-----------------|-------------|------------------------|
```

```
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl apply -f k8s/deployment.y
aml
deployment.apps/face-film-finder-backend created
deployment.apps/face-film-finder-frontend created
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl apply -f k8s/service.yaml
service/backend-service created
service/frontend-service created
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl get deployments
kubectl get pods -o wide
kubectl get svc
NAME                        READY   UP-TO-DATE   AVAILABLE   AGE
face-film-finder-backend    0/2     2            0           12s
face-film-finder-frontend   0/2     2            0           12s
NAME                                        READY   STATUS                  RES
TARTS   AGE   IP           NODE       NOMINATED NODE   READINESS GATES
face-film-finder-backend-7cfbd4488d-tf88v   0/1     ErrImagePull            0
        12s   <none>       minikube   <none>           <none>
face-film-finder-backend-7cfbd4488d-x995d   0/1     ImagePullBackOff        0
        12s   10.244.0.8   minikube   <none>           <none>
face-film-finder-frontend-64b59f464f-7g4pz  0/1     ContainerCreating       0
        12s   <none>       minikube   <none>           <none>
face-film-finder-frontend-64b59f464f-cbf7w  0/1     ContainerCreating       0
        12s   <none>       minikube   <none>           <none>
NAME               TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)
    AGE
backend-service    ClusterIP      10.99.91.178    <none>        5000/TCP
    7s
frontend-service   LoadBalancer   10.97.90.90     <pending>     3000:31559/TC
P   7s
kubernetes         ClusterIP      10.96.0.1       <none>        443/TCP
    3m4s
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$
```

Running - Kubernetes cluster



Rolling updates

```
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl set image deployment/face-film-finder-frontend frontend=face-fil
Flag --record has been deprecated, --record will be removed in the future
deployment.apps/face-film-finder-frontend image updated
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl rollout status deployment/face-film-finder-frontend
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 old replicas are pending termination...
deployment "face-film-finder-frontend" successfully rolled out
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl get pods -w
NAME                                     READY   STATUS             RESTARTS      AGE
face-film-finder-backend-57db8c4bdf-jxfmv    0/1     CrashLoopBackOff   6 (97s ago)   9m51s
face-film-finder-backend-5f59785fff-grd46    0/1     ErrImageNeverPull  0             12m
face-film-finder-backend-5f59785fff-lk5xh    0/1     ErrImageNeverPull  0             12m
face-film-finder-frontend-69bdbd5889-rsf4f   1/1     Running            0             30s
face-film-finder-frontend-69bdbd5889-w2w2q   1/1     Running            0             16s
```

```
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl set image deployment/face-film-finder-frontend frontend=face-film-finder-frontend:1.
Flag --record has been deprecated, --record will be removed in the future
deployment.apps/face-film-finder-frontend image updated
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl rollout status deployment/face-film-finder-frontend
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "face-film-finder-frontend" rollout to finish: 1 old replicas are pending termination...
deployment "face-film-finder-frontend" successfully rolled out
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl get pods -w
NAME                                     READY   STATUS             RESTARTS      AGE
face-film-finder-backend-57db8c4bdf-jxfmv    0/1     CrashLoopBackOff   6 (97s ago)   9m51s
face-film-finder-backend-5f59785fff-grd46    0/1     ErrImageNeverPull  0             12m
face-film-finder-backend-5f59785fff-lk5xh    0/1     ErrImageNeverPull  0             12m
face-film-finder-frontend-69bdbd5889-rsf4f   1/1     Running            0             30s
face-film-finder-frontend-69bdbd5889-w2w2q   1/1     Running            0             16s
^C
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl get pods -o wide
NAME                                     READY   STATUS             RESTARTS        AGE    IP            NODE       NOMINATED NODE
face-film-finder-backend-57db8c4bdf-jxfmv    0/1     CrashLoopBackOff   6 (2m51s ago)   11m    10.244.0.15   minikube   <none>
face-film-finder-backend-5f59785fff-grd46    0/1     ErrImageNeverPull  0               13m    10.244.0.12   minikube   <none>
face-film-finder-backend-5f59785fff-lk5xh    0/1     ErrImageNeverPull  0               13m    10.244.0.11   minikube   <none>
face-film-finder-frontend-69bdbd5889-rsf4f   1/1     Running            0               104s   10.244.0.18   minikube   <none>
face-film-finder-frontend-69bdbd5889-w2w2q   1/1     Running            0               90s    10.244.0.19   minikube   <none>
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl rollout history deployment/face-film-finder-frontend
deployment.apps/face-film-finder-frontend
REVISION  CHANGE-CAUSE
1         <none>
2         <none>
3         kubectl set image deployment/face-film-finder-frontend frontend=face-film-finder-frontend:1.1 --cluster=minikube --record=tru
```

Rollback and failure

```
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=frontend
  Containers:
   frontend:
    Image:      face-film-finder-frontend:latest
    Port:       3000/TCP
    Host Port:  0/TCP
    Limits:
      cpu:     200m
      memory:  512Mi
    Requests:
      cpu:         100m
      memory:      256Mi
    Liveness:      http-get http://:3000/ delay=15s timeout=1s period=20s #success=1 #failure=3
    Readiness:     http-get http://:3000/ delay=5s timeout=1s period=10s #success=1 #failure=3
    Environment:   <none>
    Mounts:        <none>
  Volumes:         <none>
  Node-Selectors:  <none>
  Tolerations:     <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    ReplicaSetUpdated
OldReplicaSets:  face-film-finder-frontend-575596f968 (0/0 replicas created), face-film-finder-frontend-69bdbd5889 (2/2 replicas created)
NewReplicaSet:   face-film-finder-frontend-6fdf97bc7d (1/1 replicas created)
Events:
  Type    Reason             Age                From                   Message
  ----    ------             ----               ----                   -------
  Normal  ScalingReplicaSet  14m                deployment-controller  Scaled up replica set face-film-finder-frontend-575596f968 from 0 to 2
  Normal  ScalingReplicaSet  12m                deployment-controller  Scaled down replica set face-film-finder-frontend-575596f968 from 2 to 1
  Normal  ScalingReplicaSet  12m                deployment-controller  Scaled up replica set face-film-finder-frontend-6fdf97bc7d from 1 to 2
  Normal  ScalingReplicaSet  11m                deployment-controller  Scaled down replica set face-film-finder-frontend-575596f968 from 1 to 0
  Normal  ScalingReplicaSet  3m2s               deployment-controller  Scaled up replica set face-film-finder-frontend-69bdbd5889 from 0 to 1
  Normal  ScalingReplicaSet  2m48s              deployment-controller  Scaled down replica set face-film-finder-frontend-6fdf97bc7d from 2 to 1
  Normal  ScalingReplicaSet  2m48s              deployment-controller  Scaled up replica set face-film-finder-frontend-69bdbd5889 from 1 to 2
  Normal  ScalingReplicaSet  2m35s              deployment-controller  Scaled down replica set face-film-finder-frontend-6fdf97bc7d from 1 to 0
  Normal  ScalingReplicaSet  2s (x2 over 12m)  deployment-controller  Scaled up replica set face-film-finder-frontend-6fdf97bc7d from 0 to 1
```

```
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ kubectl rollout undo deployment/face-film-finder-frontend
kubectl get pods
kubectl describe deployment face-film-finder-frontend   # shows event of rollback
deployment.apps/face-film-finder-frontend rolled back
NAME                                        READY   STATUS              RESTARTS       AGE
face-film-finder-backend-57db8c4bdf-jxfmv   0/1     CrashLoopBackOff    6 (4m8s ago)   12m
face-film-finder-backend-5f59785fff-grd46   0/1     ErrImageNeverPull   0              14m
face-film-finder-backend-5f59785fff-lk5xh   0/1     ErrImageNeverPull   0              14m
face-film-finder-frontend-69bdbd5889-rsf4f  1/1     Running             0              3m1s
face-film-finder-frontend-69bdbd5889-w2w2q  1/1     Running             0              2m47s
face-film-finder-frontend-6fdf97bc7d-bkvh6  0/1     ContainerCreating   0              1s
Name:                   face-film-finder-frontend
Namespace:              default
CreationTimestamp:      Sun, 05 Oct 2025 15:05:22 +0000
Labels:                 <none>
Annotations:            deployment.kubernetes.io/revision: 4
Selector:               app=frontend
Replicas:               2 desired | 1 updated | 3 total | 2 available | 1 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=frontend
  Containers:
   frontend:
    Image:      face-film-finder-frontend:latest
    Port:       3000/TCP
    Host Port:  0/TCP
    Limits:
      cpu:     200m
      memory:  512Mi
    Requests:
      cpu:         100m
      memory:      256Mi
    Liveness:      http-get http://:3000/ delay=15s timeout=1s period=20s #success=1 #failure=3
    Readiness:     http-get http://:3000/ delay=5s timeout=1s period=10s #success=1 #failure=3
    Environment:   <none>
    Mounts:        <none>
  Volumes:         <none>
  Node-Selectors:  <none>
```

step 4: Prometheus and Grafana

CODES:

1. Prometheus.yml

```
global:
 scrape_interval: 15s
scrape_configs:
 - job_name: 'face-film-finder-backend'
 static_configs:
 - targets: ['localhost:5000']
```

2. docker-compose.monitoring.yml

```
version: '3.8'
services:
 prometheus:
 image: prom/prometheus
 volumes:
 - ./prometheus.yml:/etc/prometheus/prometheus.yml
 ports:
 - "9090:9090"
 grafana:
 image: grafana/grafana
 ports:
 - "3001:3000"
 depends_on:
 - prometheus
 volumes:
 - grafana-storage:/var/lib/grafana
volumes:
 grafana-storage:
```

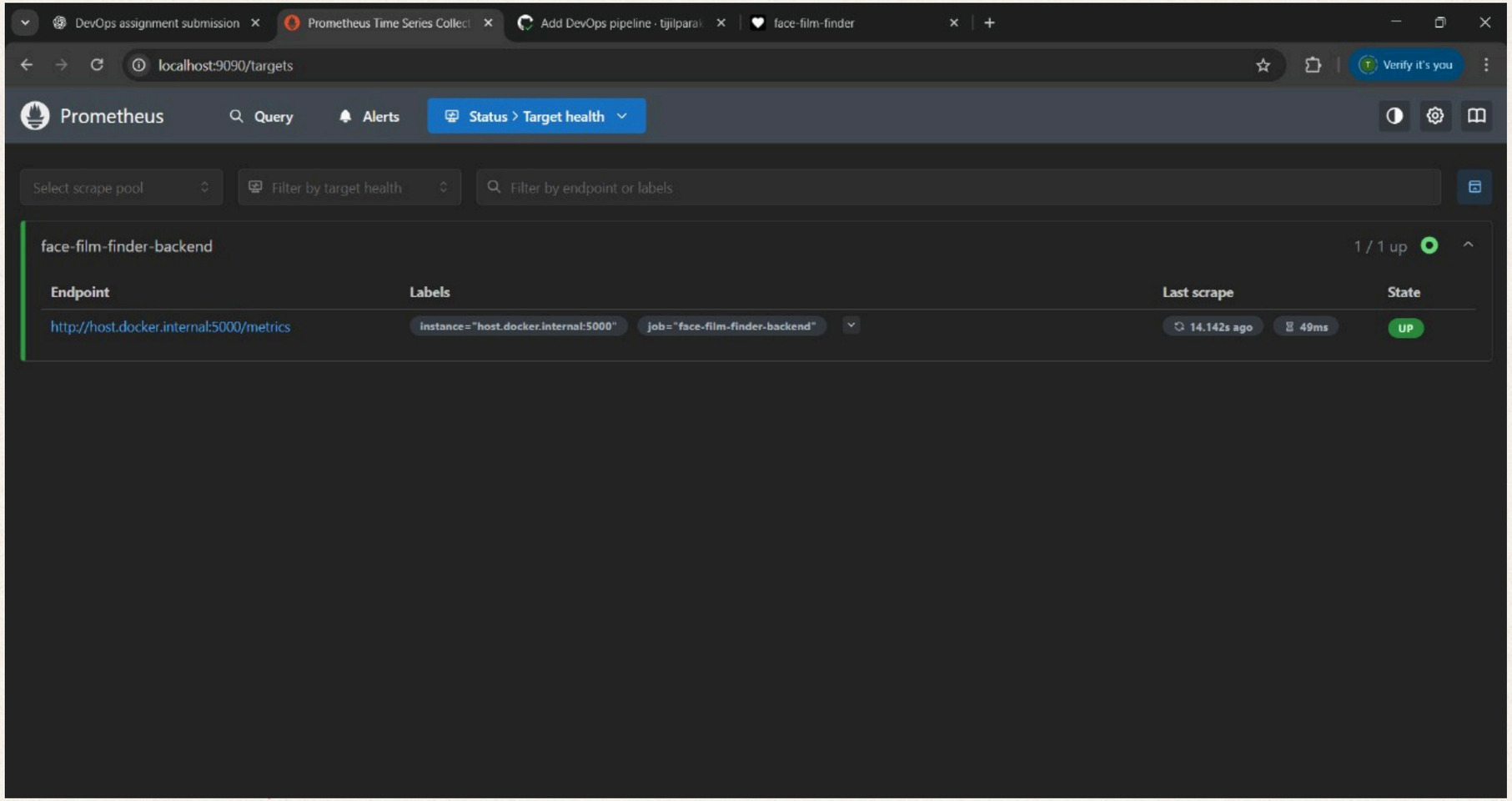
3. metrics.py (endpoint added in app.py in flask backend)

```
from prometheus_client import Counter, Histogram,
generate_latest, CONTENT_TYPE_LATEST
from flask import Response, request
import time

REQUEST_COUNT = Counter('request_count', 'App Request
Count', ['method', 'endpoint', 'http_status'])
REQUEST_LATENCY =
Histogram('request_latency_seconds', 'Request latency',
['endpoint'])

def before_request():
  request.start_time = time.time()

def after_request(response):
  request_latency = time.time() - request.start_time

REQUEST_LATENCY.labels(request.path).observe(request_l
atency)
  REQUEST_COUNT.labels(request.method, request.path,
response.status_code).inc()
  return response

def metrics():
  return Response(generate_latest(),
mimetype=CONTENT_TYPE_LATEST)
```

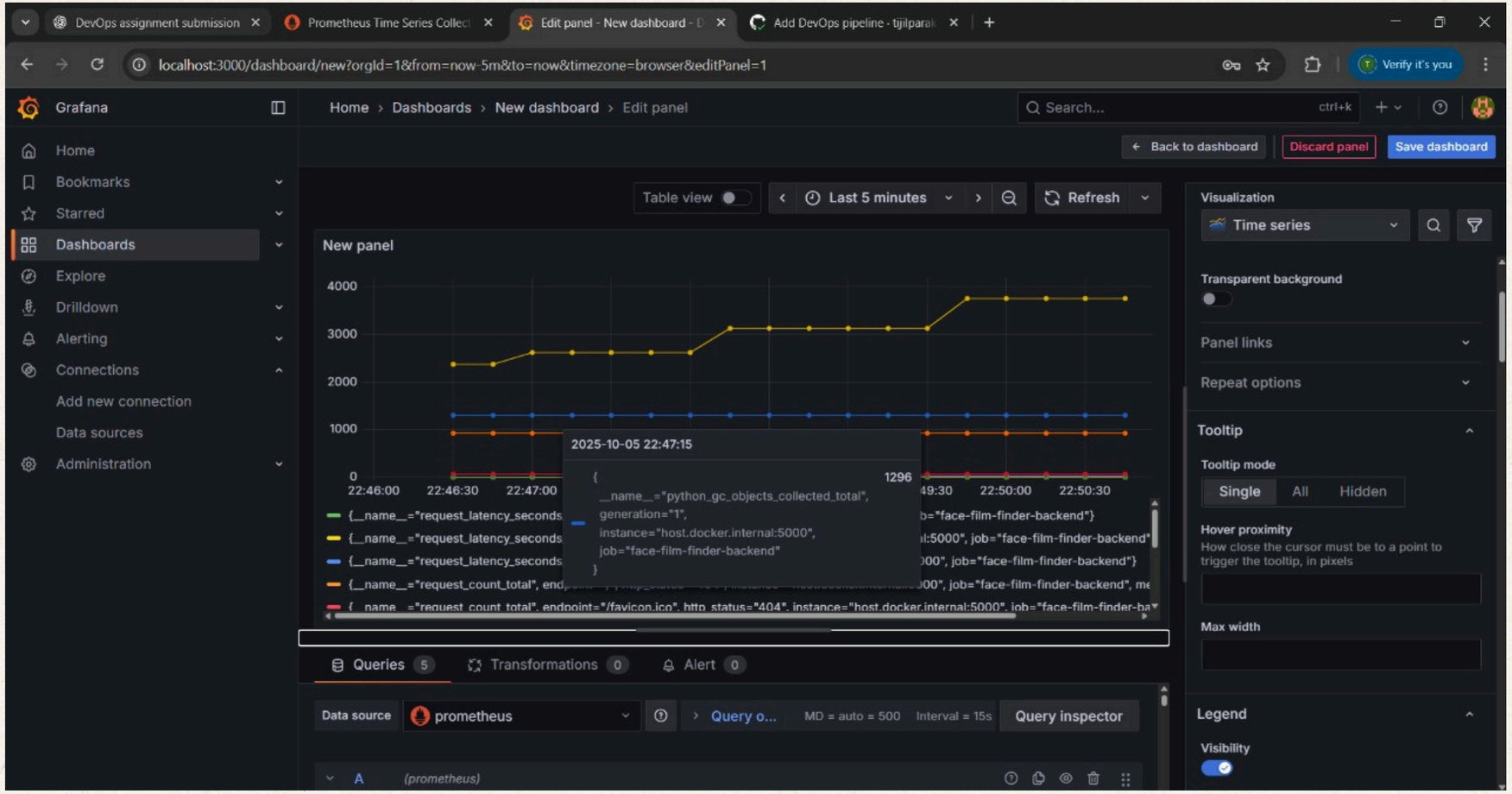
Prometheus and grafana pulled

```
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$ docker-compose -f docker-compose.monitoring.yml up -d
[+] Running 22/22
 ✔prometheus Pulled                                                                                  72.9s
   ✔e9fa37e588a8 Pull complete                                                                       69.3s
   ✔0357ac67262f Pull complete                                                                       69.2s
   ✔4314b14247f8 Pull complete                                                                        0.9s
   ✔7dc70dd519ad Pull complete                                                                        0.8s
   ✔15e2cd5823b3 Pull complete                                                                       58.6s
   ✔1617e25568b2 Pull complete                                                                        1.4s
   ✔03def9af9150 Pull complete                                                                        0.9s
   ✔9fa9226be034 Pull complete                                                                        1.2s
   ✔d14d9311398e Pull complete                                                                       68.8s
   ✔92fd369f57f0 Pull complete                                                                        1.1s
 ✔grafana Pulled                                                                                    114.7s
   ✔f04b4ef9a69c Pull complete                                                                        9.9s
   ✔b24c04af690f Pull complete                                                                        0.7s
   ✔f671127533d7 Pull complete                                                                        0.3s
   ✔ef84c8d2bf22 Pull complete                                                                        2.2s
   ✔01a7b2849125 Pull complete                                                                        9.8s
   ✔08b443d0d1f0 Pull complete                                                                        6.8s
   ✔9824c27679d3 Pull complete                                                                        6.5s
   ✔0a5ad0679b58 Pull complete                                                                      104.2s
   ✔41d23bd5dd1c Pull complete                                                                      110.7s
   ✔f28af41ca6c1 Pull complete                                                                        3.5s
[+] Running 4/4
 ✔Network face-film-finder_default            Created                                               0.1s
 ✔Volume "face-film-finder_grafana-storage"   Created                                               0.0s
 ✔Container face-film-finder-prometheus-1     Started                                               1.6s
 ✔Container face-film-finder-grafana-1        Started                                               1.5s
udisha@LAPTOP-BCKFAOHI:~/face-film-finder$
```

Prometheus ‘up’ state



Grafana dash board



THANK YOU