**Source Code for Swarm Microservice Deployment**

**Docker-Compose File:**

version: "3.9"

services:

frontend:

image: dockersamples/examplevotingapp\_vote:latest

ports:

- "8080:80" # Expose frontend on port 8080

networks:

- voting-network

deploy:

replicas: 2

resources:

limits:

cpus: "0.5"

memory: "128M"

restart\_policy:

condition: on-failure

backend:

image: dockersamples/examplevotingapp\_result:latest

ports:

- "5000:80" # Expose backend API on port 5000

networks:

- voting-network

deploy:

replicas: 2

resources:

limits:

cpus: "0.5"

memory: "128M"

restart\_policy:

condition: on-failure

worker:

image: dockersamples/examplevotingapp\_worker:latest

networks:

- voting-network

deploy:

replicas: 1

restart\_policy:

condition: on-failure

redis:

image: redis:alpine

networks:

- voting-network

deploy:

replicas: 1

restart\_policy:

condition: on-failure

postgres:

image: postgres:13

environment:

POSTGRES\_USER: user

POSTGRES\_PASSWORD: password

POSTGRES\_DB: votes

volumes:

- postgres-data:/var/lib/postgresql/data

networks:

- voting-network

deploy:

replicas: 1

restart\_policy:

condition: on-failure

visualizer:

image: dockersamples/visualizer:latest

ports:

- "8081:8080" # Expose Docker Visualizer on port 8081

volumes:

- "/var/run/docker.sock:/var/run/docker.sock" # Allow visualizer to access Docker

networks:

- voting-network

deploy:

placement:

constraints: [node.role == manager] # Ensure visualizer runs only on the manager node

networks:

voting-network:

driver: overlay

volumes:

postgres-data: