Implementation of Docker and DevOps Principles in Django-React Web Project

Reference Paper - https://ieeexplore.ieee.org/document/10117715

Team Members
- Sachchida Nand Tiwari (M23CSA527)
- Balakrishna Kariveda (M23CSA511)
- Ritesh Lamba (M23CSA544)

Introduction

- Implements principles from the IEEE paper on Docker and DevOps.

- Focus: Scalable and maintainable deployment for a Django-React web app.

- Tools: Docker, Kubernetes, Prometheus/Grafana.

System Overview

Backend: Django + PostgreSQL.

Frontend: React.

Containerization: Docker for all components.

Orchestration: Kubernetes for scalability and high availability.

Alignment with Paper

Aspect	Paper Recommendations	Our Implementation
Containerization	Use Docker	Docker for Django, React, PostgreSQL
Microservices	Modularize services	Separate containers

Alignment with Paper

Aspect	Paper Recommendations	Our Implementation
Orchestration	Use Kubernetes for cluster management	Kubernetes for scalability
High Availability	Redundancy, self-healing	Kubernetes manages failovers

Key Features

Containerized Deployment:

- Django, React, PostgreSQL in isolated Docker environments.

Kubernetes Orchestration:

- Ensures availability and resource optimization.

Load Balancing:

- Kubernetes ingress for routing and load handling.

Benefits Achieved

- **Scalability:** Adjusts to workload changes dynamically.

- **Reliability:** Redundancy and Kubernetes self-healing.

- **Efficiency:** Optimized resource usage.

- **Observability:** Actionable insights via monitoring tools.

Kubernetes Configurations Overview

File	Purpose
namespace.yaml	Isolates CRS resources.
django-config.yaml	Configures Django settings.
django-deployment.yaml	Deploys Django backend.
postgres-pv.yaml	Defines PostgreSQL storage.
ingress.yaml	Routes frontend/backend traffic.

Django Configuration

django-config.yaml: ConfigMap for environment variables.

django-secret.yaml: Secret for database credentials.

django-deployment.yaml: Deploys backend container on port 8000.

PostgreSQL Configuration

postgres-pv.yaml: PersistentVolume for database storage.

postgres-secret.yaml: Secret for PostgreSQL credentials.

postgres-deployment.yaml: Deploys PostgreSQL container on port 5432.

React Configuration

react-deployment.yaml: Deploys frontend container.

react-service.yaml: Exposes React app on port 3000.

ingress.yaml: Manages traffic routing for React and Django.

Conclusion

- Integrated Docker, Kubernetes, and DevOps principles.

- Achieved a scalable, efficient, and maintainable architecture.

- Monitoring ensures proactive issue resolution.

Thank You!