# Déploiement d'une Application Flask avec Docker et Kubernetes

#### Introduction

### Objectif du Projet

Déployer une application web Flask en utilisant Docker pour la conteneurisation et Kubernetes pour l'orchestration des conteneurs.

# **Prérequis**

- **Docker** : Pour la création et la gestion des conteneurs.
- **Kubernetes**: Pour l'orchestration des conteneurs.
- **Docker Registry**: Pour le stockage des images Docker.

# Étapes du Projet

#### 1. Création de l'Application Flask

Fichier: app.py

• **Description** : Code source de l'application Flask.

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080)
```

#### 2. Création du Dockerfile

- Fichier: Dockerfile
- **Description**: Instructions pour construire l'image Docker.

```
FROM python:3.9-slim
WORKDIR /app
COPY requirements.txt requirements.txt
RUN pip install -r requirements.txt
COPY . .
ENV PORT=8080
```

```
EXPOSE 8080
CMD ["python", "app.py"]
```

## 3. Construction et Push de l'Image Docker

Commandes :

```
docker build -t my-flask-app:latest .
docker tag my-flask-app:latest localhost:30336/my-flask-app:latest
docker push localhost:30336/my-flask-app:latest
```

## 4. Configuration de Kubernetes

#### a. Déploiement

• Fichier: deployment.yaml

• **Description** : Configuration du déploiement Kubernetes.

```
apiVersion: apps/v1
kind: Deployment
metadata:
 name: flask-app
spec:
 replicas: 1
 selector:
  matchLabels:
   app: flask-app
 template:
  metadata:
   labels:
    app: flask-app
  spec:
   containers:
   - name: flask-app
    image: localhost:30336/my-flask-app:latest
    ports:
    - containerPort: 8080
```

#### b. Service

• Fichier: service.yaml

• **Description**: Configuration du service Kubernetes pour exposer l'application.

```
apiVersion: v1
kind: Service
metadata:
name: flask-app-service
spec:
selector:
app: flask-app
```

ports:

- protocol: TCPport: 80

targetPort: 8080 nodePort: 30000 type: NodePort

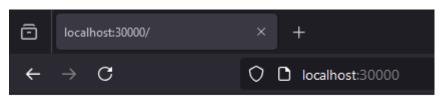
# 5. Déploiement Kubernetes

• Commandes:

kubectl apply -f deployment.yaml kubectl apply -f service.yaml

# Vérification

• **Accès à l'application** : Ouvrir http://localhost:30000 dans un navigateur pour vérifier que l'application Flask est correctement déployée.



Hello, World!