

Nom : AMANI

Prénoms : Kouakou Jaurès De Salès

5SRC2

Repository :

Github : <https://github.com/Jauresamani/ProjetFinal-Kubernetes>

Rendu Projet ESGI Kubernetes - Partie 1

PayMyBuddy - De la Conteneurisation au Déploiement Kubernetes

Contexte

L'entreprise SUN Data Consulting vous a recruté pour moderniser le déploiement de l'application PayMyBuddy, une solution de gestion de transactions financières entre amis.

L'application était initialement monolithique et déployée manuellement. Votre mission est d'adopter les meilleures pratiques DevOps :

Conteneurisation avec Docker

Gestion des services avec Docker Compose

Orchestration avec Kubernetes

Gestion des configurations et secrets

Persistance des données

1. Installation des Prérequis

1.1 Installer Docker

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y docker.io
sudo systemctl enable --now docker
```

Vérifiez l'installation :

```
docker --version
```

1.2 Installer Docker Compose

```
sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o
/usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
```

Vérifiez l'installation :

```
docker-compose --version
```

1.3 Installer Kubernetes (K3s pour un environnement léger)

```
curl -sL https://get.k3s.io | sh -
```

Vérifiez l'installation :

```
kubectl get nodes
```

2. Conteneurisation de l'Application

2.1 Dockerfile du Backend (Spring Boot)

backend/Dockerfile

```
# Utilisation d'Amazon Corretto 17
FROM amazoncorretto:17-alpine
```

```
# Définir un répertoire de travail
WORKDIR /app
```

```
# Copier le fichier JAR du backend
COPY ../target/paymybuddy.jar .
```

```
# Exposer le port de l'application
EXPOSE 8080
```

```
# Exécuter l'application
CMD ["java", "-jar", "paymybuddy.jar"]
```

2.2 Dockerfile de MySQL

dbDockerfile

```
FROM mysql:8
```

```
# Définir les variables d'environnement
ENV MYSQL_ROOT_PASSWORD=password
ENV MYSQL_DATABASE=db_paymybuddy
ENV MYSQL_USER=paymybuddy
ENV MYSQL_PASSWORD=paymybuddy
```

```
# Copier les scripts de création de la base
COPY initdb /docker-entrypoint-initdb.d/
```

2.3 Construire et Pousser les Images Docker

```
root@kbs:~/projet-esgi-kubesfinal/Kubernetes# docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organization
docker.com/go/access-tokens/

Username: kamani3
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

docker build -t kamani3/paymybuddy-backend:latest ./backend

```
root@kbs:~/projet-esgi-kubesfinal/Kubernetes# docker build -t kamani3/paymybuddy-backend:latest ./backend
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
             Install the buildx component to build images with BuildKit:
             https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  43.41MB
Step 1/5 : FROM amazoncorretto:17-alpine
17-alpine: Pulling from library/amazoncorretto
f18232174bc9: Pull complete
584747a5afe9: Pull complete
Digest: sha256:7ce85412211246ff5ce362b59aa5dbdc9093d8bb088da7f7c8b9ef99a8704cdf
Status: Downloaded newer image for amazoncorretto:17-alpine
--> 653f9c68899d
Step 2/5 : WORKDIR /app
--> Running in d87cc24d57b8
--> Removed intermediate container d87cc24d57b8
--> 4f9a5ab6d6dc
Step 3/5 : COPY target/paymybuddy.jar .
--> 2c6b04366513
Step 4/5 : EXPOSE 8080
--> Running in f7b917a84b09
--> Removed intermediate container f7b917a84b09
--> 32ec1bd29e7d
Step 5/5 : CMD ["java", "-jar", "paymybuddy.jar"]
--> Running in f855ffedacbe
--> Removed intermediate container f855ffedacbe
--> 1e4e67d9ced7
Successfully built 1e4e67d9ced7
Successfully tagged kamani3/paymybuddy-backend:latest
```

docker build -t kamani3/paymybuddy-db:latest ./db

```
root@kbs:~/projet-esgi-kubesfinal/Kubernetes# docker build -t kamani3/paymybuddy-db:latest ./db
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
             Install the buildx component to build images with BuildKit:
             https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  3.584kB
Step 1/6 : FROM mysql:8
--> 78e93ded248
Step 2/6 : ENV MYSQL_ROOT_PASSWORD=password
--> Running in 6cef1977a054
--> Removed intermediate container 6cef1977a054
--> 79c9a446749e
Step 3/6 : ENV MYSQL_DATABASE=db_paymybuddy
--> Running in 6c78da457aaf
--> Removed intermediate container 6c78da457aaf
--> 9a846b9fbc88
Step 4/6 : ENV MYSQL_USER=paymybuddy
--> Running in c99caf1681a0
--> Removed intermediate container c99caf1681a0
--> 5693ab155b46
Step 5/6 : ENV MYSQL_PASSWORD=paymybuddy
--> Running in 7c466c59830f
--> Removed intermediate container 7c466c59830f
--> 9224539c52c5
Step 6/6 : COPY initdb ./docker-entrypoint-initdb.d/
--> bc2a0a027f64
Successfully built bc2a0a027f64
Successfully tagged kamani3/paymybuddy-db:latest
```

docker push kamani3/paymybuddy-backend:latest

docker push kamani3/paymybuddy-db:latest

```
root@kbs:~/projet-esgi-kubesfinal/Kubernetes# docker push kamani3/paymybuddy-backend:latest
docker push kamani3/paymybuddy-db:latest
The push refers to repository [docker.io/kamani3/paymybuddy-backend]
8527ac67b438: Pushed
f471d2a0aa97: Pushed
9599d4d8ca8c: Mounted from library/amazoncorretto
0800c18d16d: Mounted from library/amazoncorretto
latest: digest: sha256:d3820881d1608015498212f381509fc359ae14d450b7a63fba9ae4b6eb8e1cae size: 1159
The push refers to repository [docker.io/kamani3/paymybuddy-db]
caf45302176f: Pushed
b8f291e081bd: Mounted from library/mysql
3cd35c765079: Mounted from library/mysql
e1fdbc11f083: Mounted from library/mysql
d86ea8a52c44: Mounted from library/mysql
cd633769a867: Mounted from library/mysql
9596632bf3fc: Mounted from library/mysql
eb69d5145bb4: Mounted from library/mysql
470be3fb9da1: Mounted from library/mysql
0af30e70b238: Mounted from library/mysql
a75d05e9397f: Mounted from library/mysql
latest: digest: sha256:58a565ab4429153e46650397cd1a57728a5382892d80745076df9095e5599e1c size: 2619
```

3. Déploiement avec Docker Compose

Cat docker-compose.yml

version: "3.8"

services:

paymybuddy-db:

build: ./db

container_name: paymybuddy-db

ports:

- "3306:3306"

volumes:

```

- db_data:/var/lib/mysql
environment:
  MYSQL_ROOT_PASSWORD: password
  MYSQL_DATABASE: db_paymybuddy
  MYSQL_USER: paymybuddy
  MYSQL_PASSWORD: paymybuddy

paymybuddy-backend:
  build: ./backend
  container_name: paymybuddy-backend
  ports:
    - "8080:8080"
  depends_on:
    - paymybuddy-db
  environment:
    SPRING_DATASOURCE_URL: jdbc:mysql://paymybuddy-db:3306/db_paymybuddy
    SPRING_DATASOURCE_USERNAME: paymybuddy
    SPRING_DATASOURCE_PASSWORD: paymybuddy

volumes:
  db_data:

```

Démarrer l'application :

docker-compose up -d

```

root@kbs:~/projet-esgi-kubesfinal/Kubernetes# docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS        NAMES
root@kbs:~/projet-esgi-kubesfinal/Kubernetes# docker-compose up -d
Creating paymybuddy-db ... done
Creating paymybuddy-backend ... done

```

docker ps

```

root@kbs:~/projet-esgi-kubesfinal/Kubernetes# docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS        NAMES
15c14ee71eab   kamani3/paymybuddy-backend:latest   "java -jar paymybudd..."   3 minutes ago   Exited (1) 2 minutes ago        paymybuddy-backend
4477a606fbd6   kamani3/paymybuddy-db:latest        "docker-entrypoint.s..."   3 minutes ago   Exited (127) 3 minutes ago      paymybuddy-db

```

4. Orchestration avec Kubernetes

4.1 Creation Secret Config et ConfigMap

Creation du secret

```

apiVersion: v1
kind: Secret
metadata:
  name: mysql-secret
type: Opaque
data:
  MYSQL_ROOT_PASSWORD: cGFzc3dvcmQ= # Base64 de "password"

```

```

backend-deployment.yaml configMap.yaml config.yaml mysql-deployment.yaml README.md secret.yaml
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl apply -f secret.yaml
secret/mysql-secret created
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest#

```

Création des config map

cat config.yaml

```

apiVersion: v1
kind: ConfigMap
metadata:
  name: backend-config

```

```

namespace: default
data:
  DATABASE_HOST: mysql
  DATABASE_USER: root
  SPRING_DATASOURCE_URL: jdbc:mysql://mysql:3306/paymybuddy
  SPRING_DATASOURCE_USERNAME: root
  SPRING_JPA_HIBERNATE_DDL_AUTO: update
configmap/mysql-config created
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl apply -f config.yaml
configmap/backend-config created
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest#

```

cat configMap.yaml

```

apiVersion: v1
data:
  MYSQL_DATABASE: paymybuddy
  MYSQL_ROOT_PASSWORD: password
kind: ConfigMap
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |

{"apiVersion":"v1","data":{"MYSQL_DATABASE":"paymybuddy","MYSQL_ROOT_PASSWORD":"rootpass"},"kind":"
ConfigMap","metadata":{"annotations":{},"name":"mysql-config","namespace":"default"}}
creationTimestamp: "2025-03-14T11:54:33Z"
name: mysql-config

```

```

See "kubectl apply -h" for help and examples
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl apply -f configMap.yaml
configmap/mysql-config created
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest#

```

4.2 Déploiement de MySQL

```

cat mysql-deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: mysql
spec:
  replicas: 1
  selector:
    matchLabels:
      app: mysql
  template:
    metadata:
      labels:
        app: mysql
    spec:
      containers:
        - name: mysql
          image: kamani3/paymybuddy-db :latest
          envFrom:
            - configMapRef:
                name: mysql-config
            - secretRef:
                name: mysql-secret
          ports:
            - containerPort: 3306
          volumeMounts:
            - name: mysql-storage
              mountPath: /var/lib/mysql
      volumes:

```

```

- name: mysql-storage
  persistentVolumeClaim:
    claimName: mysql-pvc
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mysql-pvc
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
---
apiVersion: v1
kind: Service
metadata:
  name: mysql
spec:
  type: NodePort
  selector:
    app: mysql
  ports:
    - port: 3306
      targetPort: 3306
      nodePort: 30006

```

Appliquer la configuration :

```
kubectl apply -f mysql-deployment.yaml
```

```

root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl apply -f mysql-deployment.yaml
deployment.apps/mysql created
persistentvolumeclaim/mysql-pvc created
service/mysql created

```

4.3 Déploiement du Backend

```
cat backend-deployment.yaml
```

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  replicas: 1
  selector:
    matchLabels:
      app: backend
  template:
    metadata:
      labels:
        app: backend
    spec:
      containers:
        - name: backend
          image: kamani3/paymybuddy-backend:latest
          envFrom:
            - configMapRef:
                name: backend-config
          env:
            - name: SPRING_DATASOURCE_PASSWORD
              valueFrom:
                secretKeyRef:
                  name: mysql-secret
                  key: MYSQL_ROOT_PASSWORD

```

```
    ports:
      - containerPort: 8080
---
apiVersion: v1
kind: Service
metadata:
  name: backend
spec:
  type: NodePort
  selector:
    app: backend
  ports:
    - port: 8080
      targetPort: 8080
      nodePort: 30080
```

Appliquer la configuration :

```
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl apply -f mysql-deployment.yaml
deployment.apps/mysql created
persistentvolumeclaim/mysql-pvc created
service/mysql created
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl apply -f backend-deployment.yaml
deployment.apps/backend created
service/backend created
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest#
```

Vérifiez l'installation :

kubectl get nodes

```
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
backend-7bb9765dd5-925l8            1/1     Running   0           13s
mysql-565544fd55-q58m4             1/1     Running   0           71s
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl get nodes
NAME    STATUS   ROLES    AGE   VERSION
kbs     Ready    control-plane,master  2d2h  v1.31.6+k3s1
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# kubectl get nodes -o wide
NAME    STATUS   ROLES    AGE   VERSION   INTERNAL-IP   EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION   CONTAINER-RUNTIME
kbs     Ready    control-plane,master  2d2h  v1.31.6+k3s1  192.168.1.27   <none>        Ubuntu 24.04.2 LTS   6.8.0-55-generic  containerd://2.0.2-k3s2
root@kbs:~/projet-esgi-kubesfinal/Kubernetes/manifest# ls
```

Vérification dans le Browser

Non sécurisé 192.168.1.27:30080/login

YouTube Programme TV Maps OHA3 Im. O.C. Po... TousLesDrivers.co... Manuel Anglais Ge... cours d'anglais xp Français avec des... Live TV :

Pay My Buddy

☐ Remember me

Log in

New here? [Create an account](#)

Pay My Buddy | 2025 | © ernhollam

Création de compte

Sign up

We make money transfer easier.

Join the other buddies and have no worries for
your next trip organisation or restaurant bill share!

Pay My Buddy

Jaures

Amani

kamani3@myges.fr

.....

Sign up

Cancel

Connexion

Pay My Buddy

Home

Good to see you again, Jaures!

Most recent connection

No connections. [Add a connection](#)

Most recent transaction

No transactions. [Add transaction](#)

Current balance

0 ₪

Update balance

Pay My Buddy | 2025 | © ernhollan