

# Rapport - Mini Application déployée avec Kubernetes

## 1. Explication du projet

L'objectif du projet était de concevoir et déployer une mini application web composée de deux services distincts, conteneurisés avec Docker, orchestrés avec Kubernetes, et exposés via une Gateway (Ingress Controller).

### Architecture mise en place :

- **Service 1** : Une application Python développée avec Flask, nommée **App** qui renvoie un message JSON simple : `{"message": "Bonjour depuis Flask!"}`.
- **Service 2** : Un second service Python (également en Flask) nommé **Time Service**, renvoyant l'heure actuelle sous format JSON : `{"current_time": "..."}.`
- **Base de données** : Intégration possible d'une base MySQL via un service supplémentaire, mais non directement utilisé dans les routes principales.
- **Gateway (Ingress)** : Configuration d'un Ingress Controller NGINX pour rediriger les requêtes vers les deux services via des chemins spécifiques (/flask et /time).

Les deux services sont déployés sous forme de conteneurs Docker, publiés sur Docker Hub, puis déployés sur un cluster Kubernetes local (via Minikube) avec un fichier de déploiement et un service Kubernetes pour chacun.

## 2. Lancement du projet

### Prérequis :

- Minikube installé avec le driver Docker sur Windows.
- Docker installé et configuré.
- Images Docker des deux services poussées sur Docker Hub.

### Étapes à suivre :

1. Démarrer Minikube et activer l'Ingress Controller :

```
minikube start  
minikube addons enable ingress
```

2. Déployer les deux services :

```
kubectl apply -f flask-deployment.yaml  
kubectl apply -f time-deployment.yaml
```

Ces fichiers YAML créent les déploiements et services pour les deux applications.

3. Déployer la Gateway (Ingress) :

```
kubectl apply -f ingress.yaml
```

L'Ingress est configuré sans host spécifique afin d'être accessible directement via l'IP du cluster Minikube (ou localhost via tunnel), avec les chemins suivants :

- /flask → redirige vers le service App.
- /time → redirige vers le service Time Service.

4. Lancer un tunnel Minikube pour exposer l'Ingress :

```
minikube tunnel
```

Cette commande doit rester active. Elle permet l'accès aux services via localhost sur le port 80.

### 3. Qu'attendre après l'exécution

Après avoir suivi les étapes précédentes, l'application est accessible depuis un navigateur ou via curl sur les URLs suivantes :

- <http://localhost/flask>

La réponse attendue est :

```
{"message": "Bonjour depuis Flask!"}
```

- <http://localhost/time>

La réponse attendue est similaire à :

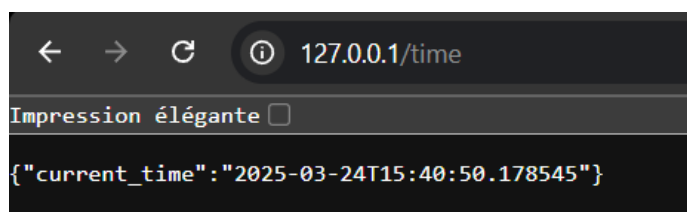
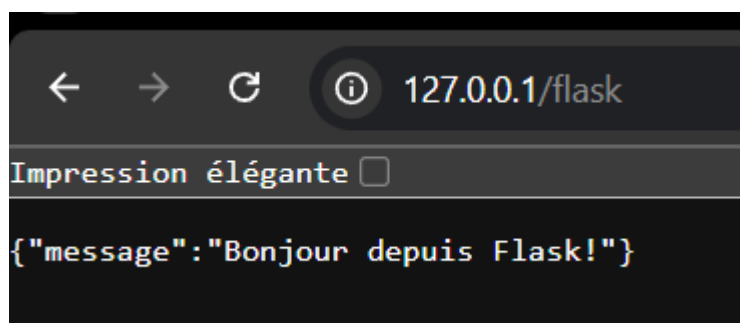
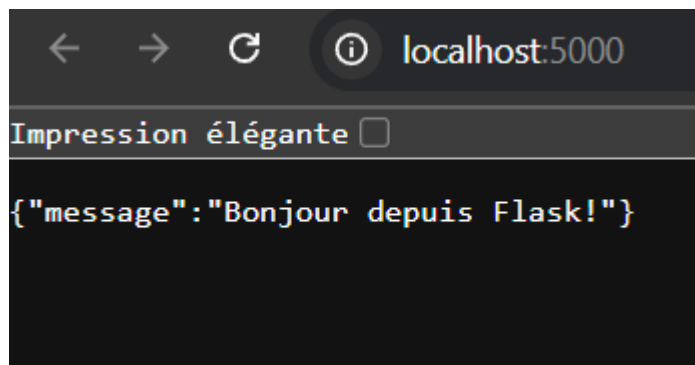
```
{"current_time": "2025-03-24T15:40:50.178545"}
```

(La réponse dépendra de l'heure et le jour à laquelle est lancé le programme)

#### Conclusion :

Le projet permet de démontrer le déploiement d'une architecture simple multi-services sous Kubernetes, incluant une Gateway, deux services distincts, et la gestion du routage via Ingress.

Ci-joint des captures d'écran après exécution :



Voici une capture d'écran des labs google effectués:

Activity	Type	Date started	Date finished	Score	Passed
<a href="#">Introduction to Docker</a>	Lab	2 hours ago	2 hours ago	Assessment: 100%	✓
<a href="#">Infrastructure as Code with Terraform</a>	Lab	3 hours ago	2 hours ago	Assessment: 100%	✓
<a href="#">Configuring Networks via gcloud</a>	Lab	3 hours ago	3 hours ago	Assessment: 100%	✓
<a href="#">Google Kubernetes Engine: Qwik Start</a>	Lab	3 hours ago	3 hours ago	Assessment: 100%	✓
<a href="#">Classify Images with TensorFlow on Google Cloud: Challenge Lab</a>	Lab	18 hours ago	18 hours ago	Assessment: 100%	✓
<a href="#">Identify Horses or Humans with TensorFlow and Vertex AI</a>	Lab	2 days ago	2 days ago	Assessment: 100%	✓
<a href="#">Classify Images with TensorFlow Convolutional Neural Networks</a>	Lab	2 days ago	2 days ago	Assessment: 100%	✓
<a href="#">Introduction to Convolutions with TensorFlow</a>	Lab	2 days ago	2 days ago	Assessment: 100%	✓
<a href="#">Introduction to Computer Vision with TensorFlow</a>	Lab	2 days ago	2 days ago	Assessment: 100%	✓
<a href="#">Classify Images with TensorFlow on Google Cloud</a>	Course	2 days ago	18 hours ago		✓
<a href="#">TensorFlow: Qwik Start</a>	Lab	2 days ago	2 days ago	Assessment: 100%	✓
<a href="#">Create a Virtual Machine</a>	Lab	Jan 30, 2025	Jan 30, 2025	Assessment: 100%	✓
<a href="#">A Tour of Google Cloud Hands-on Labs</a>	Lab	Jan 6, 2025	Jan 6, 2025	Assessment: 100%	✓

Le lien de la formation :

[https://www.cloudskillsboost.google/course\\_templates/646?catalog\\_rank=%7B%22rank%22%3A2%2C%22num\\_filters%22%3A1%2C%22has\\_search%22%3Afalse%7D](https://www.cloudskillsboost.google/course_templates/646?catalog_rank=%7B%22rank%22%3A2%2C%22num_filters%22%3A1%2C%22has_search%22%3Afalse%7D)

Et le repo github du projet :

<https://github.com/BenjaminBlitz/Projet-Virtualisation/>