

Projet Kubernetes :

PARTIE 1 : Docker

Création d'un fichier Dockerfile : >touch Dockerfile

Édition du Dockerfile : >nano Dockerfile

```
" FROM node:12.18.3
COPY package.json .
RUN npm install
COPY public ./public
COPY src ./src
CMD [ "npm", "start" ] "
```

Création de l'image : >docker build --tag nodeproject:1.0 .

Instanciation de l'image créée : > docker run -d -p 8989:3000 <IMAGE_ID>

On teste en local, sur <http://localhost:8989/> si tout fonctionne.

Liste des étapes pour la publication sur le docker hub :

On crée un repository sur le docker hub.

On se connecte depuis sur le terminal à son compte Docker Hub : >docker login

On ajoute le tag de notre image : >docker tag 22ddddc61141
luap33/firstdocker:firsttag

On push sur le docker hub notre image : >docker push luap33/firstdocker:firsttag

L'image est bien push sur le repository et disponible à cette adresse :

<https://hub.docker.com/repository/docker/luap33/firstdocker>

PARTIE 2 : Kubernetes

Installer le SDK Google Cloud : <https://cloud.google.com/sdk/docs/install#linux>

Créer un cluster sur Google Cloud Platform.

Ensuite dans le terminal faire : (à faire vers le dossier unzip du SDK)

```
./google-cloud-sdk/bin/ auth login
```

```
./google-cloud-sdk/bin/ auth configure-docker
```

Dans le terminal, on tag notre image docker mais sur notre projet google cloud platform :

```
>docker tag 22dddc61141 gcr.io/peppy-aileron-296112/myproject:firsttag
```

On push ensuite l'image :

```
>docker push gcr.io/peppy-aileron-296112/myproject:firsttag
```

On va sur Google Cloud Platform et on déploie notre image sous forme de GKE.

On affiche les pods :

```
paul_demets33@cloudshell:~ (peppy-aileron-296112)$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
gke-app-64bdbcd7f-m9stv             1/1     Running   0           19m
```

On créer le service sur le port 8989 et on l'affiche ensuite :

```
paul_demets33@cloudshell:~ (peppy-aileron-296112)$ kubectl expose deployment gke-app --name=gke-app-service --type=NodePort --port 8989 --target-port 3000
service/gke-app-service exposed
paul_demets33@cloudshell:~ (peppy-aileron-296112)$ kubectl get service
NAME                TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
gke-app-service     NodePort    10.48.11.25   <none>        8989:31212/TCP   4s
kubernetes           ClusterIP   10.48.0.1     <none>        443/TCP          23h
```

On créer notre ingress sur l'interface de Google Cloud Platform : (le fichier ingress.yaml se trouve dans les pièces-jointes du mail de rendu)

<input type="checkbox"/>	Nom ↑	État	Type	Points de terminaison	Pods	Espace de noms	Cluster
<input type="checkbox"/>	gke-app-service	OK	Port du nœud	10.48.11.25:8989 TCP	3/3	default	cluster-kubernetes
<input type="checkbox"/>	ingress	OK	Ingress	34.107.222.95/*	0/0	default	cluster-kubernetes

On obtient alors cette adresse IP : <http://34.107.222.95/>

On créer le HPA :

```
paul_demets33@cloudshell:~ (peppy-aileron-296112)$ kubectl autoscale deployment gke-app --cpu-percent=65 --min=3 --max=50
horizontalpodautoscaler.autoscaling/gke-app autoscaled
paul_demets33@cloudshell:~ (peppy-aileron-296112)$
```

Vérification de la création du HPA :

```
paul_demets33@cloudshell:~ (peppy-aileron-296112)$ kubectl get hpa
NAME                REFERENCE                TARGETS      MINPODS  MAXPODS  REPLICAS  AGE
gke-app             Deployment/gke-app        <unknown>/65%  3        50       1         5m29s
gke-app-hpa-cn2k    Deployment/gke-app        0%/80%       1        5        3         41m
paul_demets33@cloudshell:~ (peppy-aileron-296112)$
```