

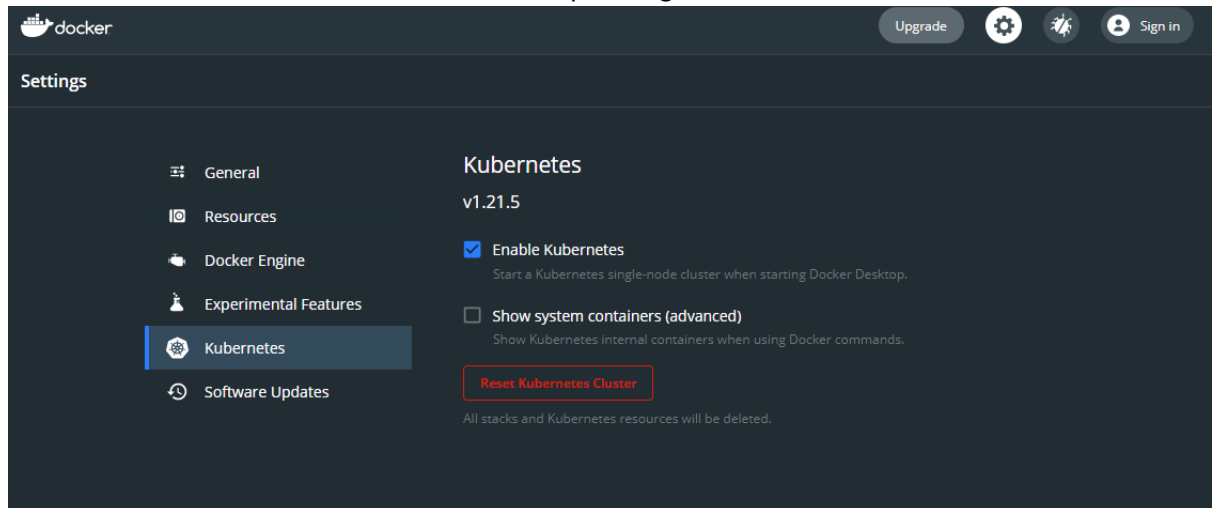
Justin Gnoh Kit Peow

A0202054Y

Github Repository: <https://github.com/justgnoh/Task-A-CS3219>

Instructions

1. Ensure Docker Desktop is running.
2. Ensure Kubernetes is enabled in Docker Desktop settings.



3. Enter the working directory `cd A2`
4. Do `docker-compose build`
5. Enter the 'Kubernetes directory' with `cd kubernetes`
6. Run `kubectl apply -f 01-mariadb-statefulset.yaml`

```
Justin@JustinsWorkstation MINGW64 ~/Desktop/New folder (2)/Task A2
$ kubectl apply -f 01-mariadb-statefulset.yaml
namespace/cs3219 created
statefulset.apps/cs3219-db created
```

7. Run `kubectl apply -f 02-mariadb-service.yaml`

```
Justin@JustinsWorkstation MINGW64 ~/Desktop/New folder (2)/Task A2
$ kubectl apply -f 02-mariadb-service.yaml
service/cs3219-db created
```

8. Run `kubectl apply -f 03-catalog-deployment.yaml`

```
Justin@JustinsWorkstation MINGW64 ~/Desktop/New folder (2)/Task A2
$ kubectl apply -f 03-catalog-deployment.yaml
deployment.apps/catalog-service-deployment created
```

9. Run `kubectl get all -n cs3219`

10. Check if all deployments are in RUNNING status.

```
Justin@JustinsWorkstation MINGW64 ~/Desktop/New folder (2)/Task-A-CS3219/A2/kubernetes (main)
$ kubectl get all -n cs3219
```

NAME	READY	STATUS	RESTARTS	AGE
pod/catalog-service-deployment-56fb6f65bc-fcbxj	1/1	Running	0	21m
pod/catalog-service-deployment-56fb6f65bc-lskfv	1/1	Running	0	21m
pod/cs3219-db-0	1/1	Running	0	21m

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
service/catalog-service	NodePort	10.109.32.195	<none>	8080:31307/TCP	2s
service/cs3219-db	NodePort	10.109.87.102	<none>	3306:31306/TCP	21m

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
deployment.apps/catalog-service-deployment	2/2	2	2	21m

NAME	DESIRED	CURRENT	READY	AGE
replicaset.apps/catalog-service-deployment-56fb6f65bc	2	2	2	21m

NAME	READY	AGE
statefulset.apps/cs3219-db	1/1	21m

11. Enter `localhost:31307/catalogs` into your browser

12. Observe the results returned



Additional Learning

The initial setup for Kubernetes was confusing and difficult at first, especially with need to redeploy Kubernetes after making changes.

A point to note is that:

1. Reset Kubernetes from Docker Desktop
2. Delete all images that need to be rebuilt
3. (IMPORTANT) Restart Docker Desktop [Not performing this step will lead to Kubernetes hanging when making requests]
4. Rerun ``docker-compose build`` when necessary
5. Rerun all Kubernetes commands
6. Observe that new changes are propagated to Kubernetes