# Progetto di Sistemi Distribuiti e Big Data

**Istruzioni per la build del docker compose**

Per effettuare il build del progetto basta utilizzare il file docker-compose.yaml:

avendo docker engine avviato, posizionarsi nella root directory del progetto e

digitare nel terminale il seguente comando:

docker compose up

in questo modo verranno costruite le immagini dei microservizi a partire dai dockerfile

Qualora, invece, si volesse effettuare la build del singolo microservizio, posizionarsi nella root specifica

e digitare il seguente comando, specificando nome dell'immagine e tag:

docker build -t <image\_name>:<image\_tag>

Per creare ed avviare un container basterà digitare il seguente comando:

docker run --name <container\_name> -p <local\_port>:<container\_port> -v <volume\_mapping> -e <env\_variable> <image\_name>:<image\_tag>

**Istruzioni per il deploy su K8S**

Per effettuare il deploy su K8S in locale utilizzare Kind ed installare Kubectl:

Posizionarsi all'interno della directory k8s e creare il cluster usando il seguente comando:

kind create cluster --config=config-yml

in questo modo verrà creato un cluster costituito da due nodi: un control plane node e

un worker node. Il cluster sarà raggiungibile tramite localhost, usando le porte 80 (HTTP) e

443 (HTTPS).

Creato il cluster effettuare il deploy dei microservizi col seguente comando:

kubectl apply -k .

Mentre per effettuare il delete dei microservizi:

#kubectl delete -k .

- DATABASE:

effettuare il deploy dei database (mysqlaut.yml, mysqlmain.yml e slaDB.yml), in quanto i relativi microservizi,

una volta avviati, proveranno subito a connettersi con essi.

- MainServer:

effettuare il deploy del bucket mainserver (mainserver.yml) per la stessa motivazione esposta prima;

- KAFKA:

effettuare il deploy del broker Kafka in quanto essenziale alla comunicazione tra mainserver e manager notification, dopo aver deployato kafka, bisognerà creare il topic.

Per fare ciò impostiamo le variabili d’ambiente KAFKA\_CREATE\_TOPICS con il nome del topic

- MICROSERVIZI:

infine, sarà possibile effettuare il deploy di tutti i microservizi che gestiranno backend e QoS