

## ASSIGNMENT 4

Student Name	P.JAYASHREE
Student Roll Number	951919CS031
Team ID	PNT2022TMID13229

### 1. Pull an Image from docker hub and run it in docker playground.

```
docker pull uifd/ui-for-docker
```

```
docker run -d -p 9000:9000 --privileged -v
```

```
/var/run/docker.sock:/var/run/docker.sock uifd/ui-for- docker
```

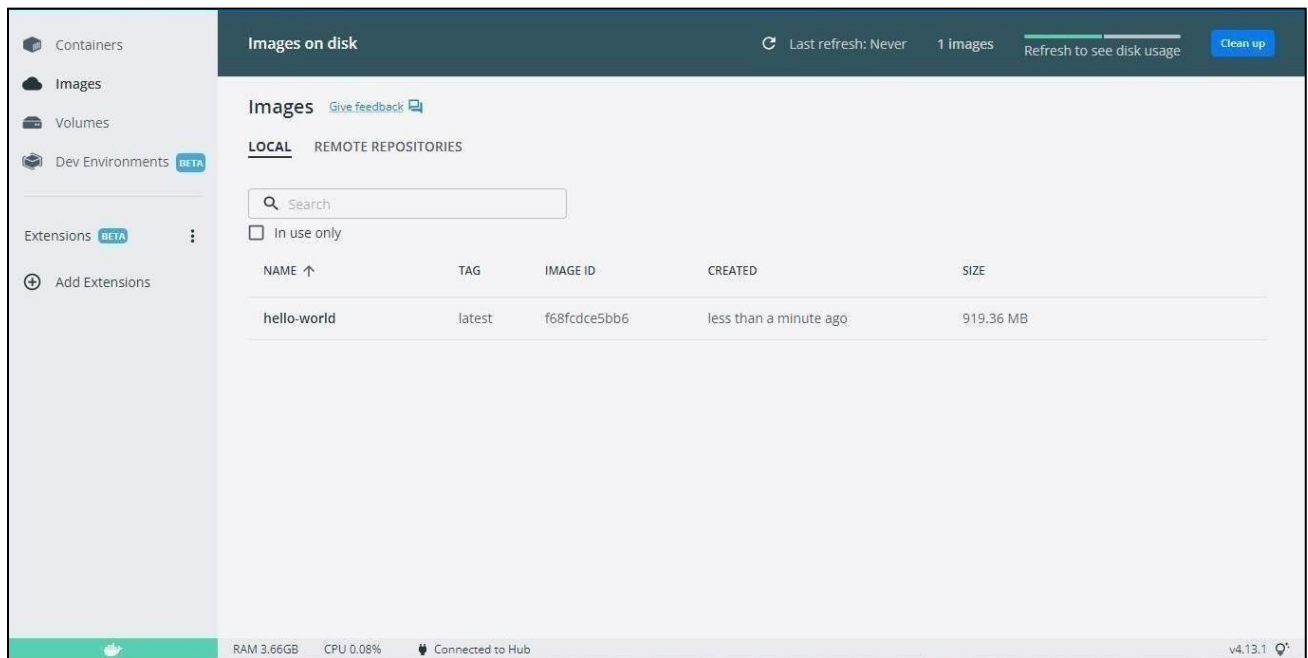
The screenshot displays the Docker Playground interface. At the top, a timer shows 03:57:05. Below it, a 'CLOSE SESSION' button is visible. The 'Instances' section on the left shows a list of instances, including one named 'node1' with IP 192.168.0.13. The main area shows details for a container named 'cdi0ji60\_cdi0jpe0qau0008f9u8g'. It displays the IP 192.168.0.13, an 'OPEN PORT' button for port 9000, and resource usage: Memory 1.59% (63.77MiB / 3.906GiB) and CPU 0.45%. An SSH command is provided: 'ssh ip172-18-0-40-cdi0ji60qau0008f9u80@direct.labs.play-v'. Below this are 'DELETE' and 'EDITOR' buttons. A terminal window shows the following output:

```
# This is a sandbox environment. Using personal credentials #
# is HIGHLY! discouraged. Any consequences of doing so are #
# completely the user's responsibilities. #
# #
# The PWD team. #
#####
[node1] (local) root@192.168.0.13 ~
$ docker pull uifd/ui-for-docker
Using default tag: latest
```

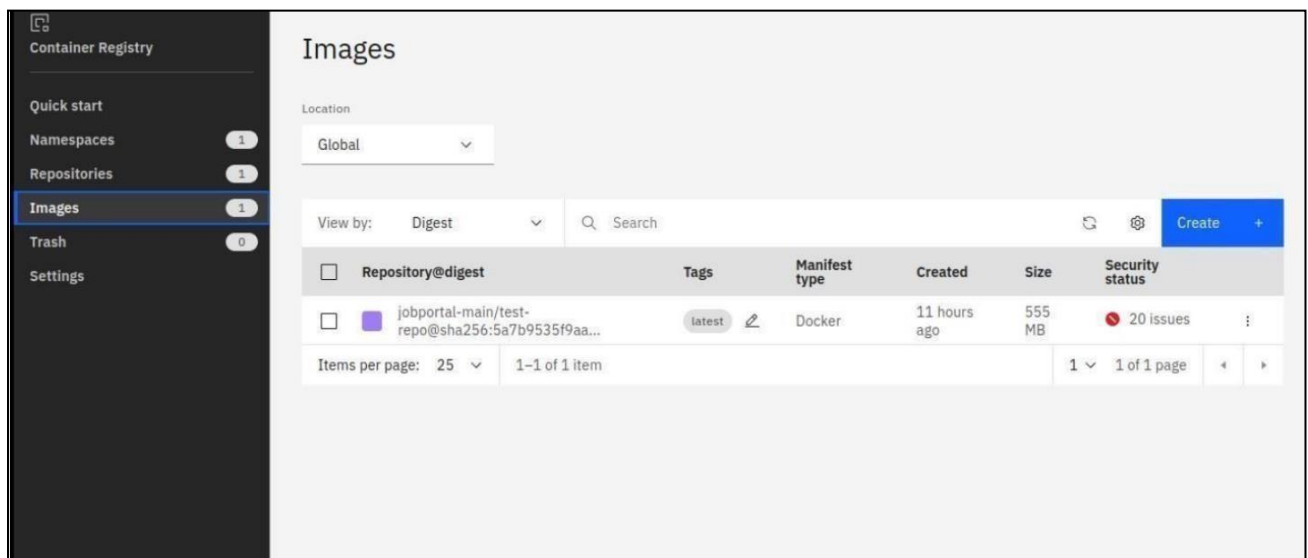
Below the terminal, the 'UI For Docker' dashboard is shown. It has a navigation bar with tabs: Dashboard, Containers, Containers Network, Images, Networks, Volumes, and Info. A 'Refresh' button is on the right. The 'Running Containers' section shows a list with 'serene\_keller' and 'Up 47 seconds'. A 'Status' donut chart shows 100% Running (green), 0% Stopped (red), and 0% Ghost (grey). The 'Containers created' section shows a bar chart with a value of 1. The 'Images created' section shows a bar chart with a value of 1.

## 2. Create a docker file for the jobportal application and deploy it in Docker desktop application.

```
FROM helloworld:latest
WORKDIR ~/Desktop/
ADD . helloworld/
WORKDIR ~/Desktop/htmlfile
RUN pip install -r requirements
RUN chmod +x app.sh
CMD
```



## 3. Create a IBM container registry and deploy helloworld app or jobportalapp.



**4. Create a Kubernetes cluster in IBM cloud and deploy helloworld image or jobportal image and also expose the same app to run in nodeport.**

apiVersion: v1 kind:

Service metadata:

name: hello-world- deployment spec:

ports: - port:

5000 targetPort:

5000 selector: app:

hello-world --- apiVersion: apps/v1 kind: Deployment metadata: name:

hello-world- deployment spec:

replic as: 1 select or:

matchLabels: app: helloworld

template: meta da ta :

la be ls :

app: hello- world spec:

containers:

- name: hello-world image: au.icr.io/hello-worldapp/helloworld imagePullPolicy:

Always ports:

- containerPort: 5000