




Assignment-4

Date	22 October 2022
Student Name	Naveen P
Student Roll No	737819ECR115
Max Marks	2

1.Pull an Image from docker hub and run it in Docker Playground

 **dockerhub** [Explore](#) [Repositories](#) [Organizations](#) [Help](#) [Upgrade](#)  naveen9221

[Explore](#) [uifd/ui-for-docker](#)

 **uifd/ui-for-docker** ☆ 10M+ Pulls
By [uifd](#) • Updated 6 years ago
A web interface for Docker, formerly known as DockerUI. Deprecated, use Portainer for new features.
[Image](#)

[Overview](#) [Tags](#)

UI For Docker

This repo is deprecated. Development continues at: [portainer/portainer](#)

[chat](#) [on github](#)



UI For Docker is a web interface for the Docker Remote API. The goal is to provide a pure client side implementation so it is effortless to connect and manage docker.

Docker Pull Command

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

03:50:01

CLOSE SESSION

Instances  


+ ADD NEW INSTANCE

192.168.0.18
node1

cdpgm160_cdp3gm3e0qau000fkqf90

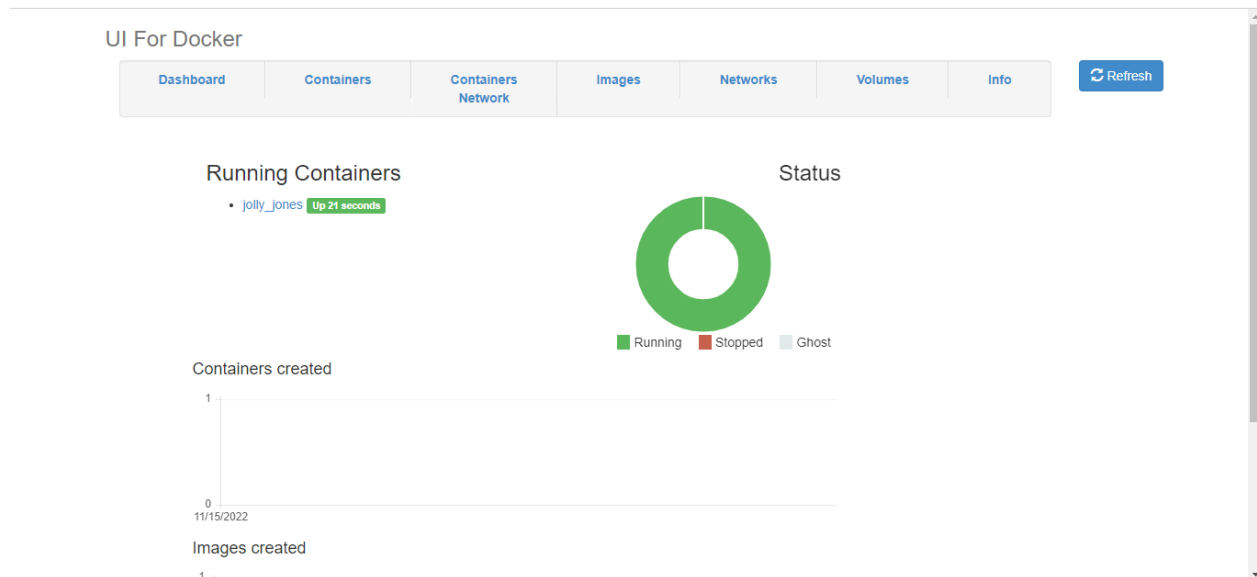
IP: 192.168.0.18 [OPEN PORT](#) 9000

Memory: 1.70% (68.12MiB / 3.906GiB) CPU: 0.80%

SSH: ssh ip172-18-0-22-cdp3gm160qau000fkqf8g@direct.labs.pla 

[DELETE](#) [EDITOR](#)

```
# The PWD team.
#####
[node1] (local) root@192.168.0.18 ~
$ docker pull uifd/ui-for-docker
Using default tag: latest
latest: Pulling from uifd/ui-for-docker
041194d080c8: Pull complete
Digest: sha256:fe371ff5a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1cb749
Status: Downloaded newer image for uifd/ui-for-docker:latest
docker.io/uifd/ui-for-docker:latest
[node1] (local) root@192.168.0.18 ~
$ docker run -d -p 9000:9000 --privileged -v /var/run/docker.sock:/var/run/docker.sock uifd/ui-for-docker
53ba2dfcb5ad7a6d80d3507e0f4d83f23e9258185c5c3436b017b4df7c62d52e
[node1] (local) root@192.168.0.18 ~
$
```



2. Create a docker file for the job portal application and deploy it in Docker desktop application

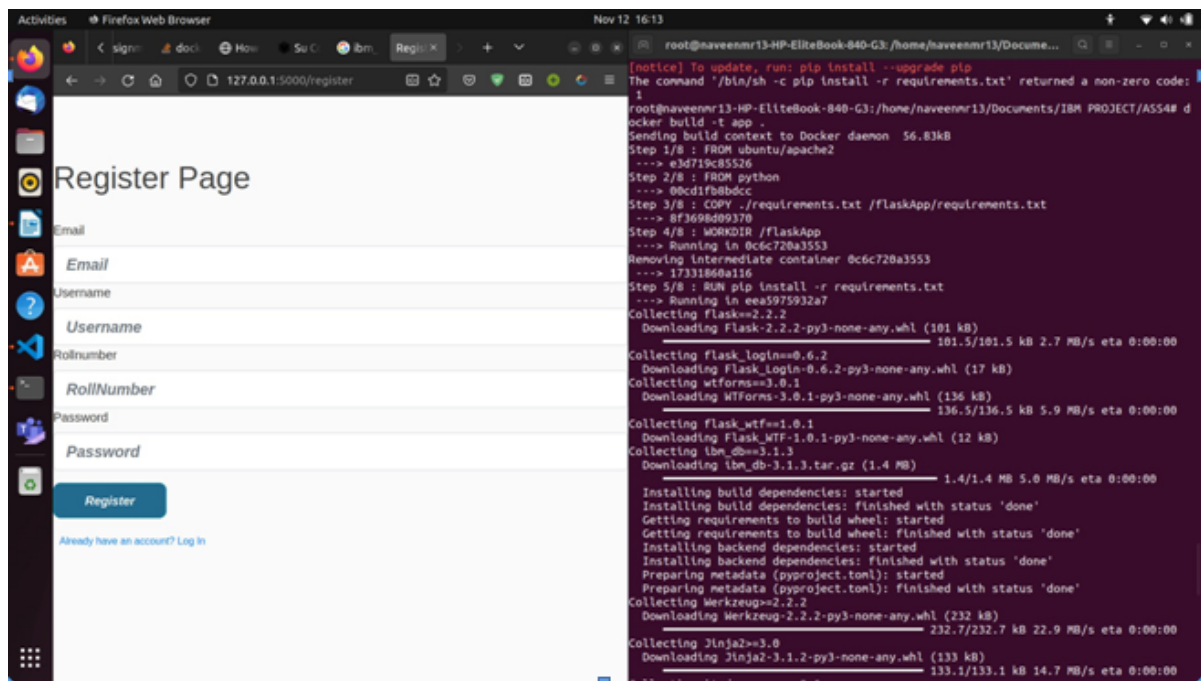
Dockerfile:

```
FROM
ubuntu/apache2 FROM python
COPY ./requirements.txt /flaskApp/requirements.txt
WORKDIR /flaskApp
RUN pip install -r requirements.txt
COPY . /flaskApp
ENTRYPOINT [ "python" ]
CMD [ "app.py" ]
```

```

root@naveenmr13-HP-EliteBook-840-G3:/home/naveenmr13/Documents/IBM PROJECT/ASS4# d
ocker run -p 5000:5000 app
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. U
se a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.2:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 107-635-278
172.17.0.1 - - [12/Nov/2022 10:42:44] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Nov/2022 10:42:44] "GET /static/style.css HTTP/1.1" 304 -
172.17.0.1 - - [12/Nov/2022 10:43:34] "GET /register HTTP/1.1" 200 -
172.17.0.1 - - [12/Nov/2022 10:43:34] "GET /static/style.css HTTP/1.1" 304 -

```



The screenshot shows a web browser window with the 'Register Page' and a terminal window displaying Docker build logs.

Register Page: The page has a title 'Register Page' and a form with the following fields: Email, Username, RollNumber, Password, and Password (repeated). There is a 'Register' button and a link 'Already have an account? Log In'.

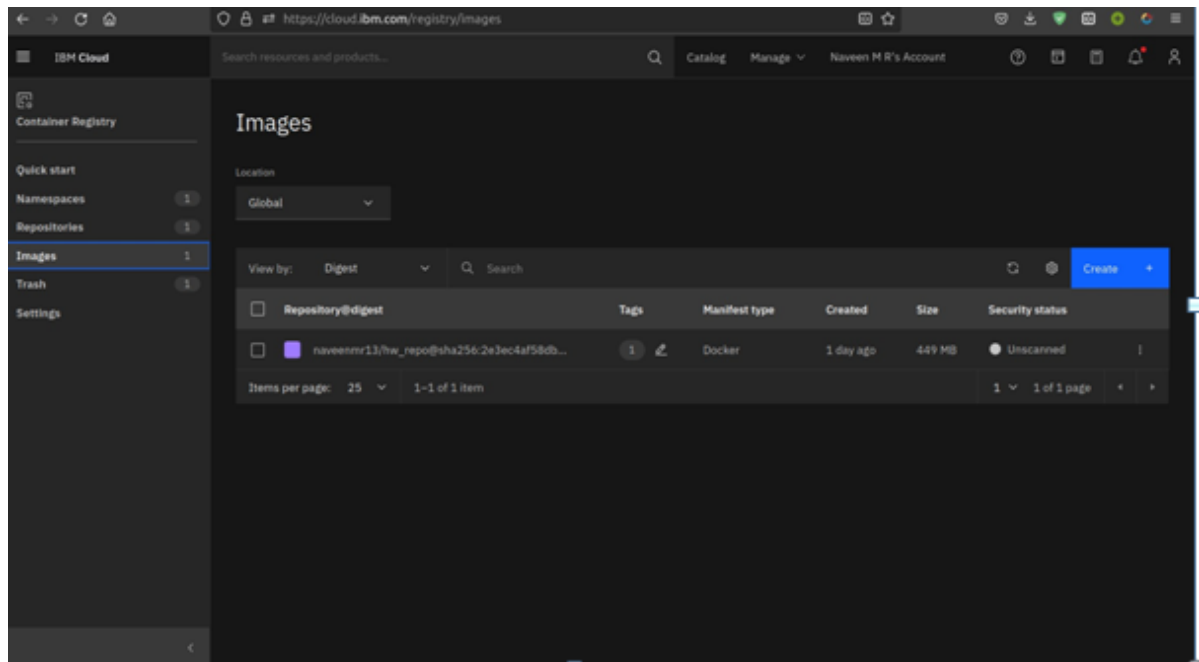
Terminal Output: The terminal shows the Docker build process for the 'app' container. It includes the following steps and output:

```

[notice] To update, run: pip install --upgrade pip
The command '/bin/sh -c pip install -r requirements.txt' returned a non-zero code: 1
root@naveenmr13-HP-EliteBook-840-G3:/home/naveenmr13/Documents/IBM PROJECT/ASS4# d
ocker build -t app .
Sending build context to Docker daemon 56.83kB
Step 1/8 : FROM ubuntu/apache2
--> e3d719c85526
Step 2/8 : FROM python
--> 00cd1fb8bdcc
Step 3/8 : COPY ./requirements.txt /flaskApp/requirements.txt
--> 8f3698d09370
Step 4/8 : WORKDIR /flaskApp
--> Running in 0c6c720a3553
Removing intermediate container 0c6c720a3553
--> 17331860a116
Step 5/8 : RUN pip install -r requirements.txt
--> Running in eea5975932a7
Collecting flask==2.2.2
  Downloading Flask-2.2.2-py3-none-any.whl (101 kB)
Collecting flask_login==0.6.2
  Downloading Flask_Login-0.6.2-py3-none-any.whl (17 kB)
Collecting wtforms==3.0.1
  Downloading WTForms-3.0.1-py3-none-any.whl (136 kB)
Collecting flask_wtf==1.0.1
  Downloading Flask_WTF-1.0.1-py3-none-any.whl (12 kB)
Collecting ibm_db==3.1.3
  Downloading ibm_db-3.1.3.tar.gz (1.4 MB)
Installing build dependencies: started
Installing build dependencies: finished with status 'done'
Getting requirements to build wheel: started
Getting requirements to build wheel: finished with status 'done'
Installing backend dependencies: started
Installing backend dependencies: finished with status 'done'
Preparing metadata (pyproject.toml): started
Preparing metadata (pyproject.toml): finished with status 'done'
Collecting Werkzeug==2.2.2
  Downloading Werkzeug-2.2.2-py3-none-any.whl (232 kB)
Collecting Jinja2==3.0
  Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)

```

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



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

