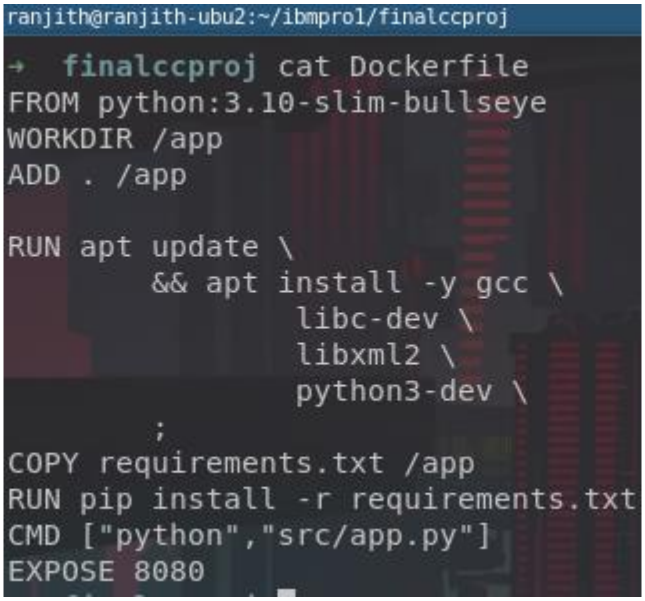
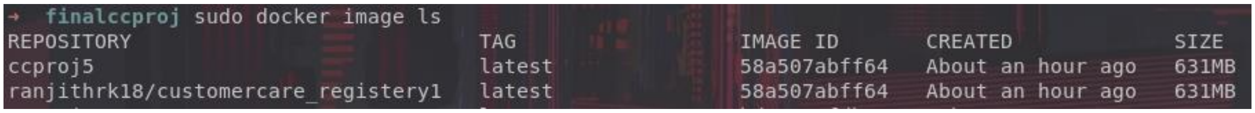
Sprint 4
Containerization and deployment

Docker file



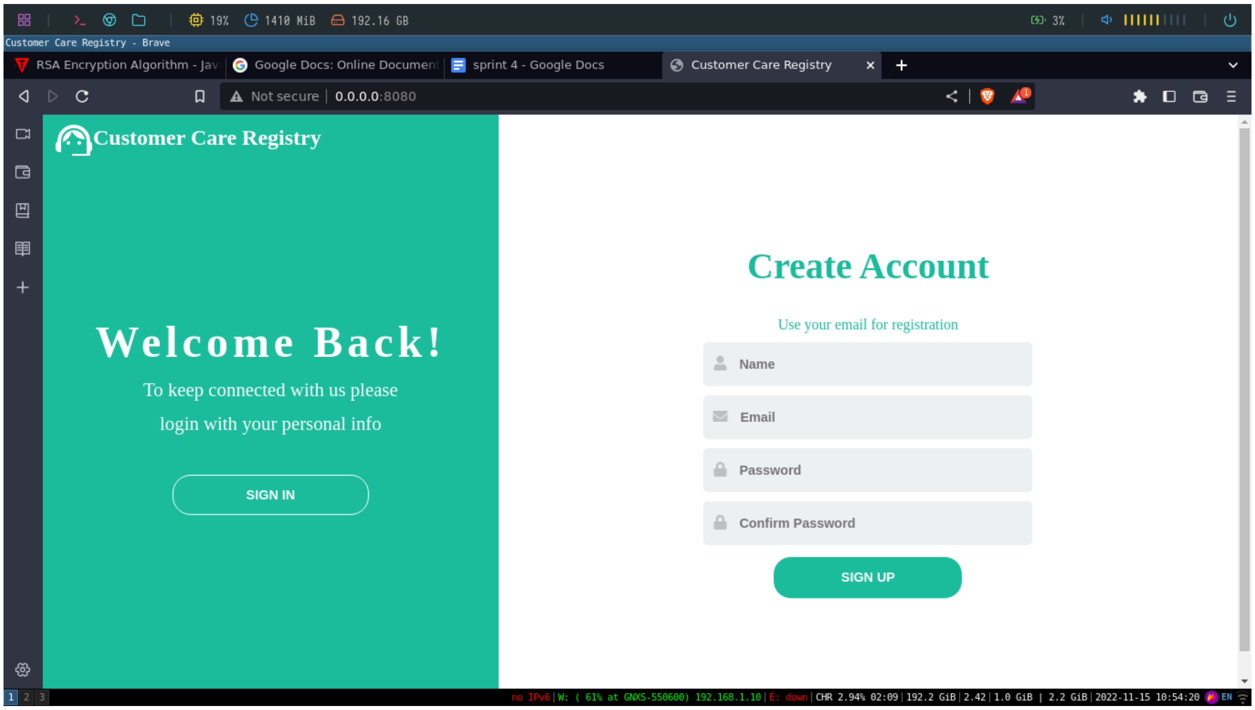
```
ranjith@ranjith-ubu2:~/ibmpro1/finalccproj

→  finalccproj cat Dockerfile
FROM python:3.10-slim-bullseye
WORKDIR /app
ADD . /app

RUN apt update \
        && apt install -y gcc \
                libc-dev \
                libxml2 \
                python3-dev \
        ;
COPY requirements.txt /app
RUN pip install -r requirements.txt
CMD ["python","src/app.py"]
EXPOSE 8080
```

Docker images



```
→  finalccproj sudo docker image ls
REPOSITORY                          TAG       IMAGE ID       CREATED              SIZE
ccproj5                             latest    58a507abff64   About an hour ago    631MB
ranjithrk18/customercare_registery1 latest    58a507abff64   About an hour ago    631MB
```

Running container



Customer Care Registry - Brave

Not secure | 0.0.0.0:8080

Customer Care Registry

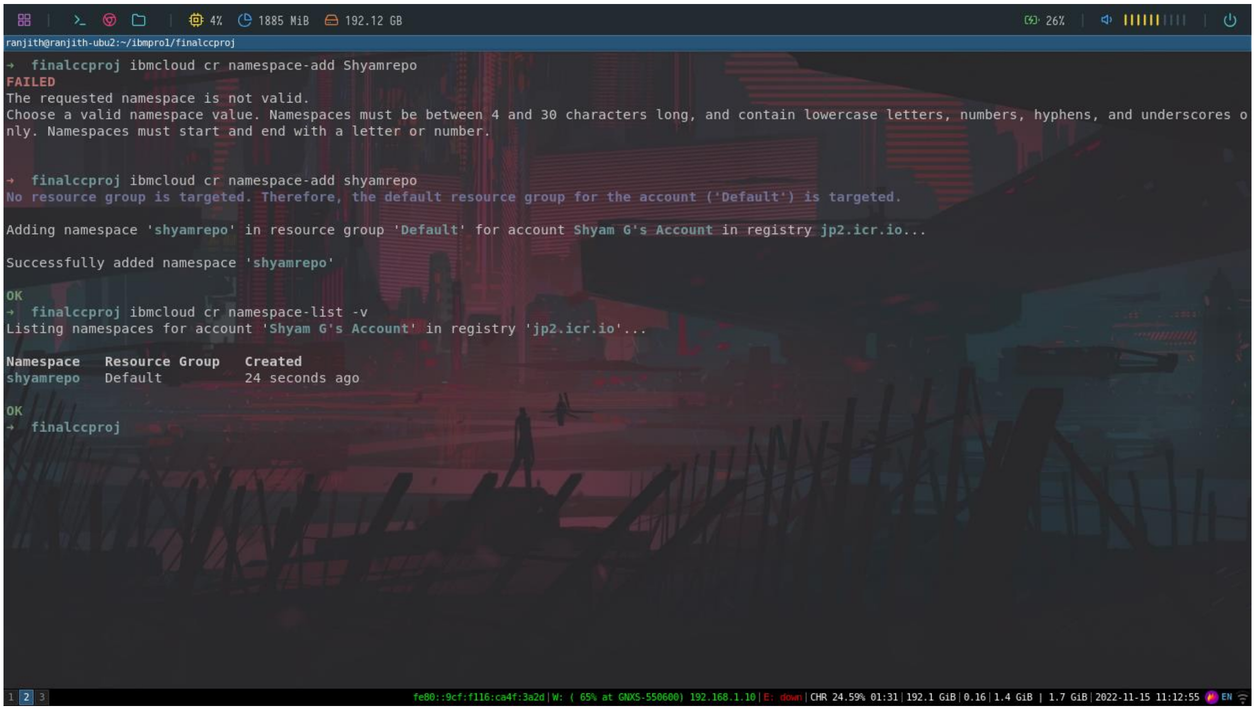
Welcome Back!

To keep connected with us please login with your personal info

Create Account

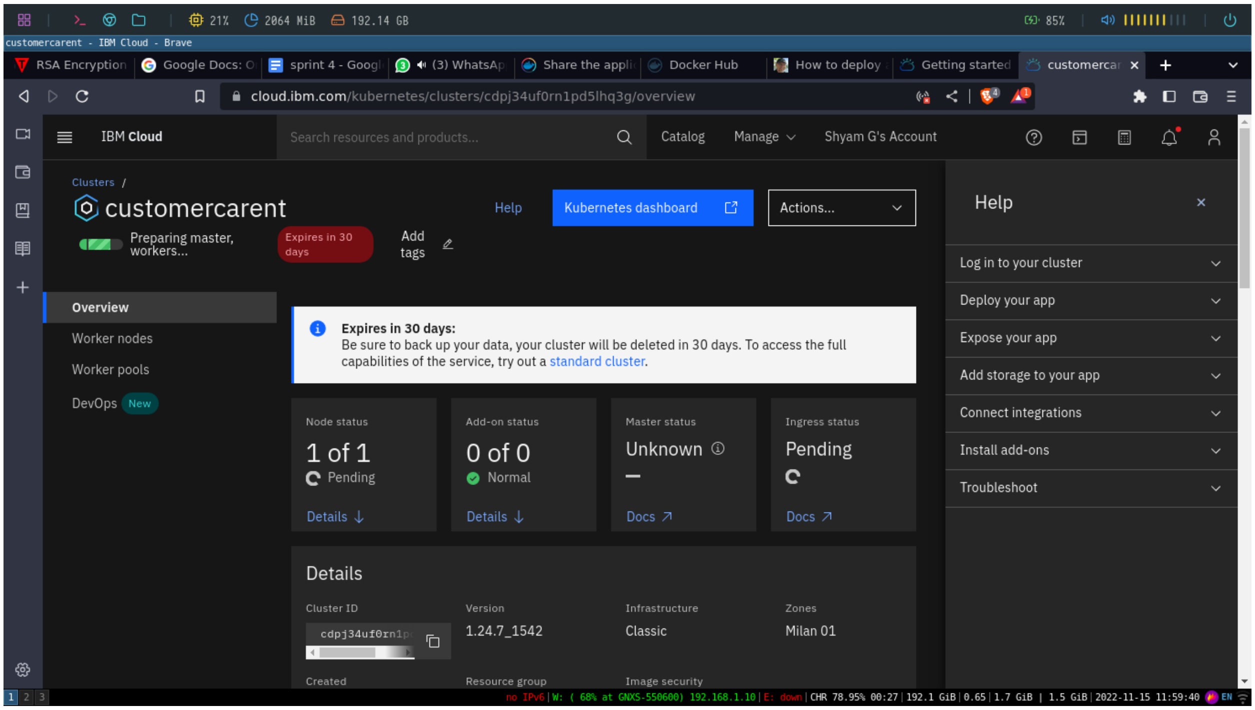
Use your email for registration

19%
1410 MiB
192.16 GB
3%
RSA Encryption Algorithm - Jav
Google Docs: Online Document
sprint 4 - Google Docs
Customer Care Registry
SIGN IN
Name
Email
Password
Confirm Password
SIGN UP
1 2 3
no IPv6 | W: (61% at GNXS-550600) 192.168.1.10 | E: down | CHR 2.94% 02:09 | 192.2 GiB | 2.42 | 1.0 GiB | 2.2 GiB | 2022-11-15 10:54:20 EN



```
ranjith@ranjith-ubu2:~/ibmprol/finalccproj
→  finalccproj ibmcloud cr namespace-add Shyamrepo
FAILED
The requested namespace is not valid.
Choose a valid namespace value. Namespaces must be between 4 and 30 characters long, and contain lowercase letters, numbers, hyphens, and underscores o
nly. Namespaces must start and end with a letter or number.


→  finalccproj ibmcloud cr namespace-add shyamrepo
No resource group is targeted. Therefore, the default resource group for the account ('Default') is targeted.

Adding namespace 'shyamrepo' in resource group 'Default' for account Shyam G's Account in registry jp2.icr.io...

Successfully added namespace 'shyamrepo'

OK
→  finalccproj ibmcloud cr namespace-list -v
Listing namespaces for account 'Shyam G's Account' in registry 'jp2.icr.io'...

Namespace    Resource Group   Created
shyamrepo    Default          24 seconds ago

OK
→  finalccproj
```

Kubernetes cluster creation



Expires in 30 days:
Be sure to back up your data, your cluster will be deleted in 30 days. To access the full capabilities of the service, try out a standard cluster.

Node status
1 of 1
Pending
Details

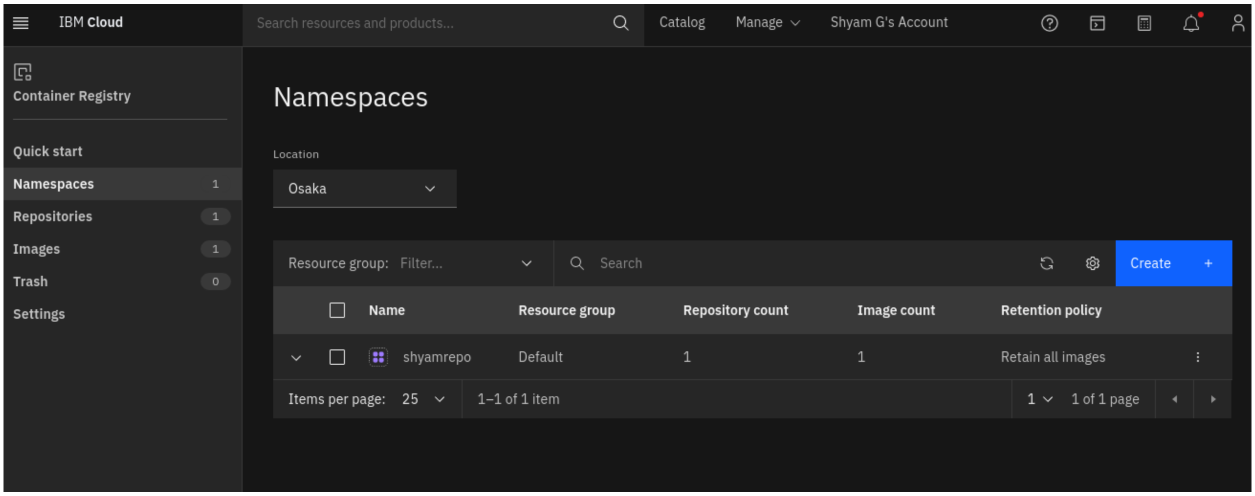
Add-on status
0 of 0
Normal
Details

Master status
Unknown
—
Docs

Ingress status
Pending
Docs

Details

Cluster ID: cdpj34uf0rn1p
Version: 1.24.7_1542
Infrastructure: Classic
Zones: Milan 01
Created
Resource group
Image security



Namespaces

Location: Osaka

Resource group: Filter...

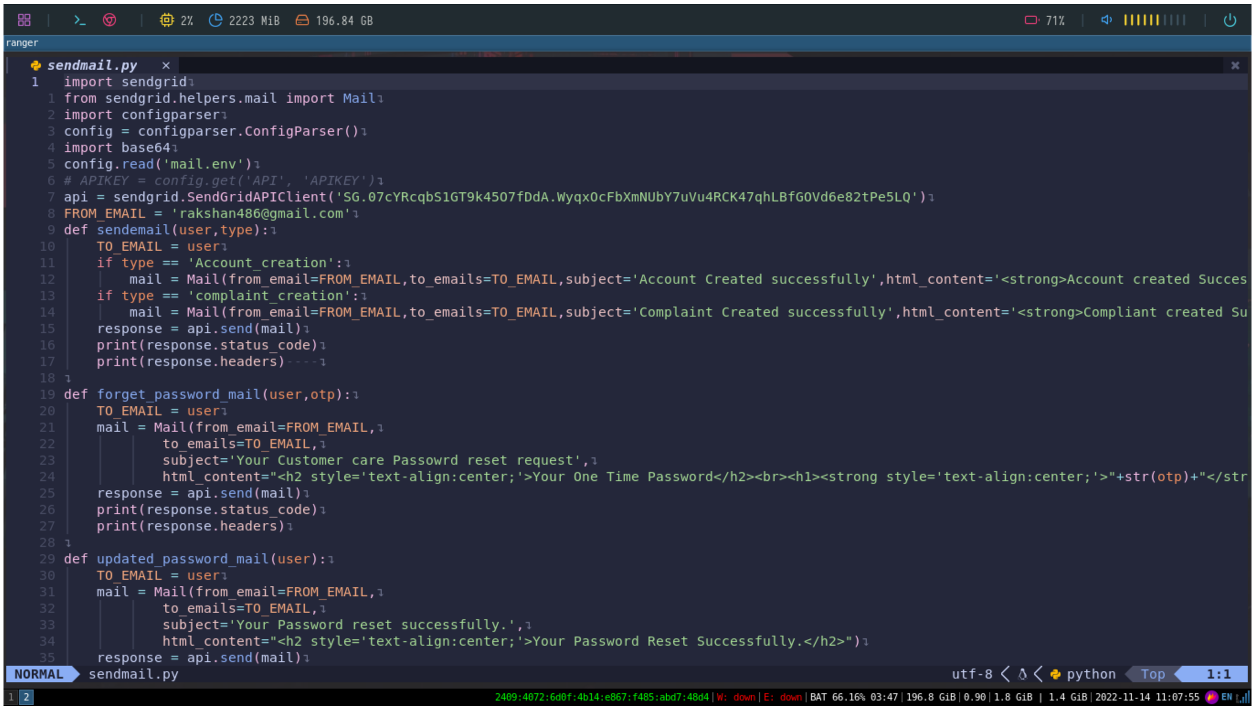
Name	Resource group	Repository count	Image count	Retention policy
shyamrepo	Default	1	1	Retain all images

Items per page: 25 | 1–1 of 1 item | 1 | 1 of 1 page

Sendgrid Acc:

Sendgrid code:

```
import sendgrid
from sendgrid.helpers.mail import Mail
import configparser
config = configparser.ConfigParser()
import base64
config.read('mail.env')
# APIKEY = config.get('API', 'APIKEY')
api =
sendgrid.SendGridAPIClient('SG.07cYRcqbS1GT9k45O7fDdA.WyqxOcFbXmNUbY7uVu4RCK
47qhLBfGOVd6e82tPe5LQ')
FROM_EMAIL = 'rakshan486@gmail.com'
def sendemail(user,type):
    TO_EMAIL = user
    if type == 'Account_creation':
        mail = Mail(from_email=FROM_EMAIL,to_emails=TO_EMAIL,subject='Account Created
successfully',html_content='<strong>Account created Successfully</strong>')
    if type == 'complaint_creation':
        mail = Mail(from_email=FROM_EMAIL,to_emails=TO_EMAIL,subject='Complaint Created
successfully',html_content='<strong>Compliant created Successfully</strong>')
    response = api.send(mail)
    print(response.status_code)
    print(response.headers)
```


2%
2223 MiB
196.84 GB
71%
ranger
sendmail.py ×
NORMAL
sendmail.py
utf-8
python
Top
1:1
2409:4072:6d0f:4b14:e867:f485:abd7:48d4 | W: down | E: down | BAT 66.16% 03:47 | 196.8 GiB | 0.90 | 1.8 GiB | 1.4 GiB | 2022-11-14 11:07:55
EN

```python
import sendgrid
from sendgrid.helpers.mail import Mail
import configparser
config = configparser.ConfigParser()
import base64
config.read('mail.env')
# APIKEY = config.get('API', 'APIKEY')
api = sendgrid.SendGridAPIClient('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')
FROM_EMAIL = 'rakshan486@gmail.com'
def sendemail(user,type):
    TO_EMAIL = user
    if type == 'Account_creation':
        mail = Mail(from_email=FROM_EMAIL,to_emails=TO_EMAIL,subject='Account Created successfully',html_content='<strong>Account created Succes
    if type == 'complaint_creation':
        mail = Mail(from_email=FROM_EMAIL,to_emails=TO_EMAIL,subject='Complaint Created successfully',html_content='<strong>Compliant created Su
    response = api.send(mail)
    print(response.status_code)
    print(response.headers)

def forget_password_mail(user,otp):
    TO_EMAIL = user
    mail = Mail(from_email=FROM_EMAIL,
            to_emails=TO_EMAIL,
            subject='Your Customer care Passowrd reset request',
            html_content="<h2 style='text-align:center;'>Your One Time Password</h2><br><h1><strong style='text-align:center;'>"+str(otp)+"</str
    response = api.send(mail)
    print(response.status_code)
    print(response.headers)

def updated_password_mail(user):
    TO_EMAIL = user
    mail = Mail(from_email=FROM_EMAIL,
            to_emails=TO_EMAIL,
            subject='Your Password reset successfully.',
            html_content="<h2 style='text-align:center;'>Your Password Reset Successfully.</h2>")
    response = api.send(mail)
```