

Date	19 NOVEMBER 2022
Team ID	PNT2022TMID41651
Project Name	Customer Care Registry

## Sprint 1

### Containerization and deployment

#### Docker file

```

ranjith@ranjith-ubu2:~/ibmproj1/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_registry1 latest     58a507abff64  About an hour ago  631MB

```



```
ranjith@ranjith-ubuntu2:~/ibmproj/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 only. 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'

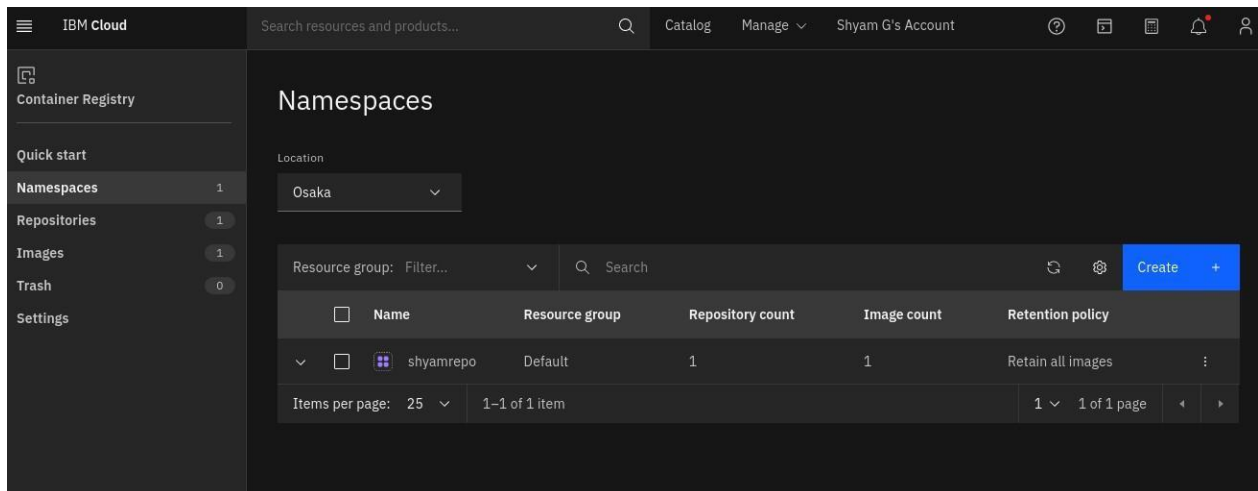
OK
+ finalccproj ibmcloud cr namespace-list -v
Listing namespaces for account 'Shyam G's Account' in registry 'jp2.icr.io'...

Namespace Resource Group Created
shyamrepo Default 24 seconds ago

OK
+ finalccproj
```

## Kubernetes cluster creation

The screenshot displays the IBM Cloud console interface for a Kubernetes cluster. The top navigation bar shows the IBM Cloud logo and search bar. The main content area is titled 'customerarent' and includes a 'Kubernetes dashboard' link. A prominent warning banner states '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](#).' Below this, the 'Overview' tab is selected, showing a status summary: '1 of 1' nodes are 'Pending', '0 of 0' add-ons are 'Normal', the 'Master status' is 'Unknown', and the 'Ingress status' is 'Pending'. A 'Details' section provides specific information: Cluster ID 'cdpj34uf0zn1p...', Version '1.24.7\_1542', Infrastructure 'Classic', and Zones 'Milan 01'. The right sidebar contains a 'Help' menu with options like 'Log in to your cluster', 'Deploy your app', 'Expose your app', 'Add storage to your app', 'Connect integrations', 'Install add-ons', and 'Troubleshoot'.



**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>Complaint created Successfully</strong>')
    response = api.send(mail)
    print(response.status_code)
    print(response.headers)
```

```
ranger
2223 MiB 196.84 GB 71%
sendmail.py x
1 import sendgrid
2 from sendgrid.helpers.mail import Mail
3 import configparser
4 config = configparser.ConfigParser()
5 import base64
6 config.read('mail.env')
7 # APIKEY = config.get('API', 'APIKEY')
8 api = sendgrid.SendGridAPIClient('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')
9 FROM_EMAIL = 'rakshan486@gmail.com'
10 def sendemail(user,type):
11     TO_EMAIL = user
12     if type == 'Account_creation':
13         mail = Mail(from_email=FROM_EMAIL,to_emails=TO_EMAIL,subject='Account Created successfully',html_content='<strong>Account created Succes
14     if type == 'complaint_creation':
15         mail = Mail(from_email=FROM_EMAIL,to_emails=TO_EMAIL,subject='Complaint Created successfully',html_content='<strong>Compliant created Su
16     response = api.send(mail)
17     print(response.status_code)
18     print(response.headers)
19 def forget_password_mail(user,otp):
20     TO_EMAIL = user
21     mail = Mail(from_email=FROM_EMAIL,
22                 to_emails=TO_EMAIL,
23                 subject='Your Customer care Passowrd reset request',
24                 html_content='<h2 style='text-align:center;'>Your One Time Password</h2><br><h1><strong style='text-align:center;'>"+str(otp)+"</str
25     response = api.send(mail)
26     print(response.status_code)
27     print(response.headers)
28 def updated_password_mail(user):
29     TO_EMAIL = user
30     mail = Mail(from_email=FROM_EMAIL,
31                 to_emails=TO_EMAIL,
32                 subject='Your Password reset successfully.',
33                 html_content='<h2 style='text-align:center;'>Your Password Reset Successfully.</h2>')
34     response = api.send(mail)
35
NORMAL sendmail.py utf-8 python Top 1:1
2409:4072:6dbf:4b14:e867:1485:ab07:48d4 | W: show | E: show | BAT 66.16% 03:47 | 196.8 GiB | 0.90 | 1.8 GiB | 1.4 GiB | 2022-11-14 11:07:55 | EN
```