

Assignment (4.5)

1. Setting the environment variable for service.

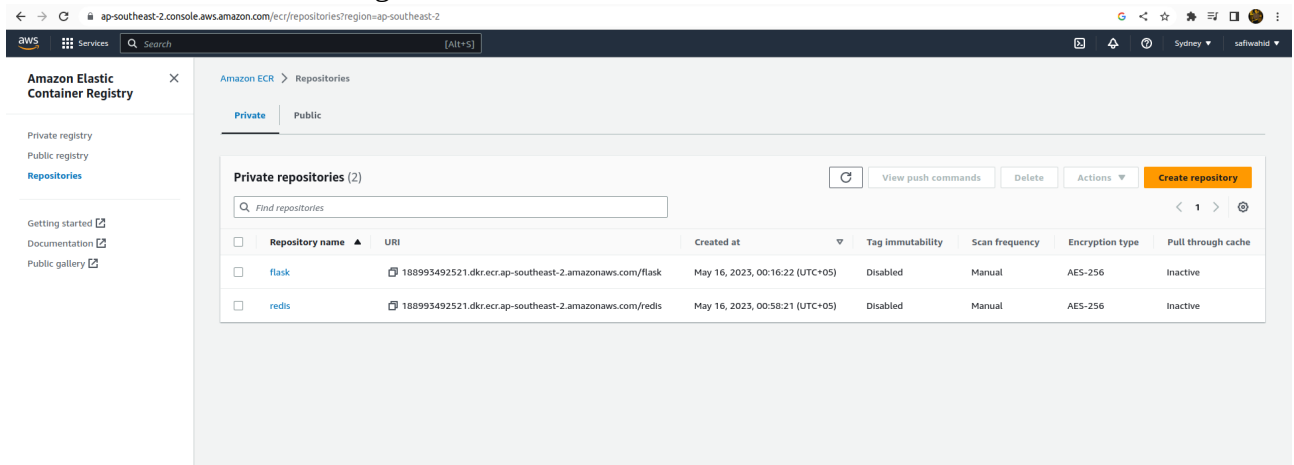
```
day_1_microservices > integrating_flask_redis > app.py > ...
1  import time
2  import os
3
4  import redis
5  from flask import Flask
6
7  app = Flask(__name__)
8  service=os.environ.get('SERVICE_DISCOVERY')
9  cache = redis.Redis(host=service, port=6379)
10
11
12 def get_and_increase_hit_count():
13     retries = 5
14     while True:
15         try:
16             return cache.incr("hits")
17         except redis.exceptions.ConnectionError as exc:
18             if retries == 0:
19                 raise exc
20             retries -= 1
21             time.sleep(0.5)
22
23
24 @app.route("/")
25 def hello():
26     count = get_and_increase_hit_count()
27     return "Hello World! I have been seen {} times.\n".format(count)
```

2.adding env to dockerfile

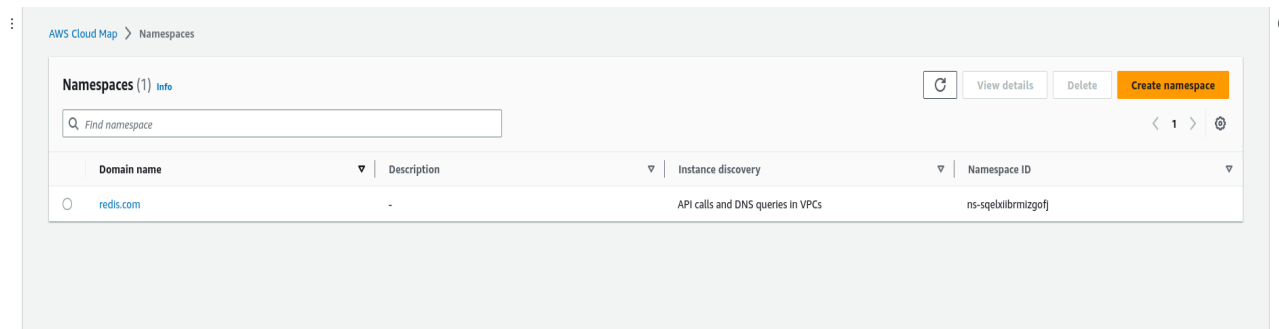
```
day_1_microservices > integrating_flask_redis > Dockerfile
1  FROM python:3.7-alpine
2
3  WORKDIR /code
4
5  ENV FLASK_APP=app.py
6  ENV FLASK_RUN_HOST=0.0.0.0
7  ENV SERVICE_DISCOVERY='redis'
8
9  RUN apk add --no-cache gcc musl-dev linux-headers
10 COPY requirements.txt requirements.txt
11
12 RUN pip install -r requirements.txt
13
14 EXPOSE 5000
15 |
16 COPY . .
17
18 CMD ["flask", "run"]
```

Muhammad Safi (2303.khi.deg.016)
Huzaiifa Ali (2303.khi.deg.023)
Sheikh Muhammad Sabih (2303.khi.deg.010)

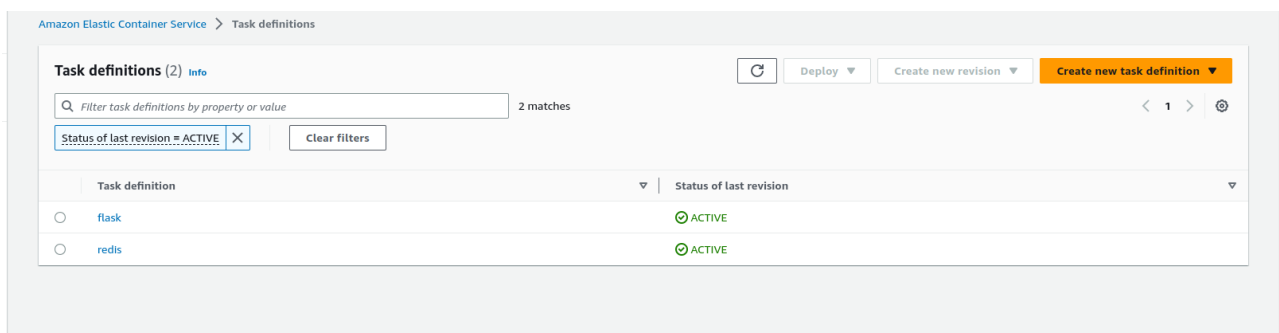
3. Pushed flask and redis images to ECR.



4. Creating namespaces and services.

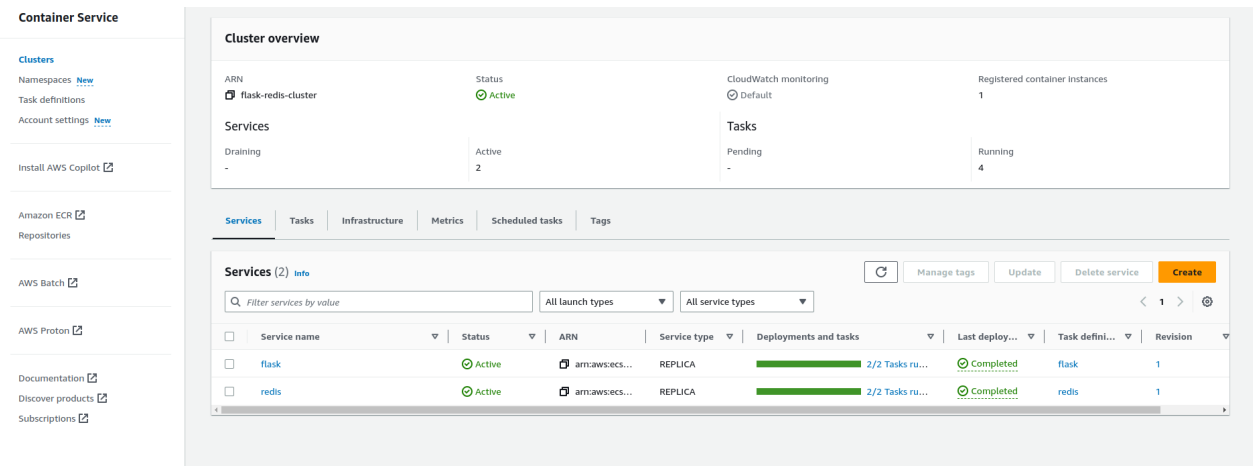
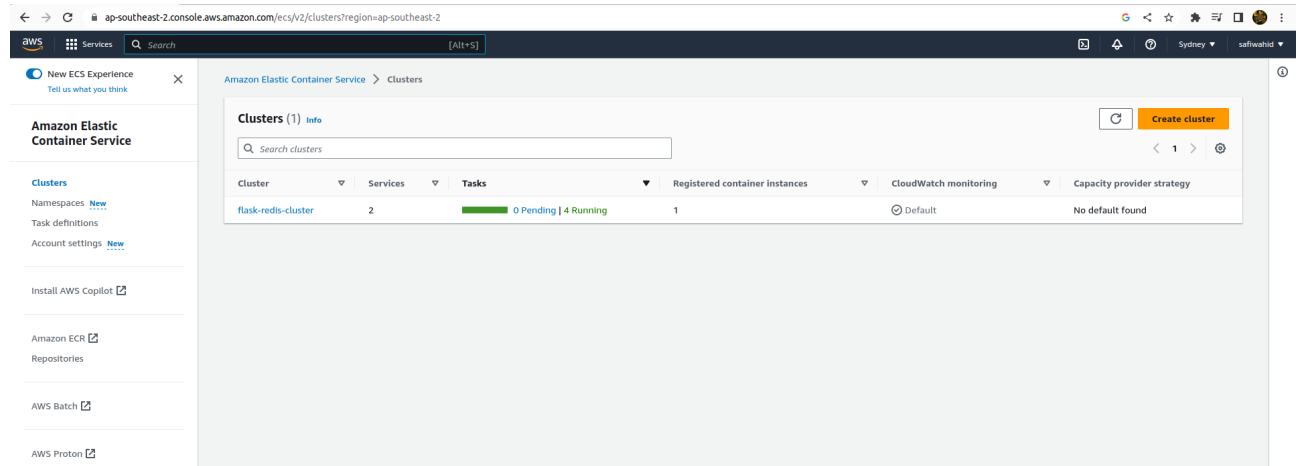


5. Creating Task definitions for flask and redis with port mappings and configurations.



Muhammad Safi (2303.khi.deg.016)
Huzafa Ali (2303.khi.deg.023)
Sheikh Muhammad Sabih (2303.khi.deg.010)

6. Adding services the cluster.



Muhammad Safi (2303.khi.deg.016)
Huzaifa Ali (2303.khi.deg.023)
Sheikh Muhammad Sabih (2303.khi.deg.010)

7. Checking the network interface of container and copying the public IP.

EC2 > Network Interfaces > eni-04e8d4f7c369150d6

Network interface summary for eni-04e8d4f7c369150d6

Refresh Delete network interface Actions

▼ Network interface details		
Network interface ID eni-04e8d4f7c369150d6	Name -	Description arn:aws:ecs:ap-southeast-2:188993492521:attachment/40804bc8-63e5-4e26-b40d-ad1609718eb9
Network interface status In-use	Interface type Elastic network interface	Security groups sg-03a73ad6eb44826a9 (flask-4045)
VPC ID vpc-0dda8ff0488842e9a	Subnet ID subnet-0fa69535419984494	Availability Zone ap-southeast-2a
Owner 188993492521	Requester ID 578734482556	Requester-managed True
Source/dest. check True		
▼ IP addresses		
Private IPv4 address 10.0.0.234	Private IPv4 DNS ip-10-0-0-234.ap-southeast-2.compute.internal	Elastic Fabric Adapter False
Public IPv4 address 13.55.209.230	Public IPv4 DNS ec2-13-55-209-230.ap-southeast-2.compute.amazonaws.com	IPv6 addresses -
Secondary private IPv4 addresses -	Association ID -	Elastic IP address owner amazon
MAC address 02:ae:5b:1c:53:04	IPv4 Prefix Delegation -	IPv6 Prefix Delegation -
▼ Instance details		
Instance ID -	Instance owner 672209591922	Device Index 1
Allocation ID		

Muhammad Safi (2303.khi.deg.016)
Huzaifa Ali (2303.khi.deg.023)
Sheikh Muhammad Sabih (2303.khi.deg.010)

8. Checking the deployed instance on 5000 port.

← → 🔒 Not secure | ec2-13-55-209-230.ap-southeast-2.compute.amazonaws.com:5000

Hello World! I have been seen 5 times.