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

Assignment (4.5)

1. Setting the environment variable for service.

day_1_microservices > integrating_flask_redis > app.py > ...

```python
import time
import os

import redis
from flask import Flask

app = Flask(__name__)
service=os.environ.get('SERVICE_DISCOVERY')
cache = redis.Redis(host=service, port=6379)


def get_and_increase_hit_count():
    retries = 5
    while True:
        try:
            return cache.incr("hits")
        except redis.exceptions.ConnectionError as exc:
            if retries == 0:
                raise exc
            retries -= 1
            time.sleep(0.5)


@app.route("/")
def hello():
    count = get_and_increase_hit_count()
    return "Hello World! I have been seen {} times.\n".format(count)
```

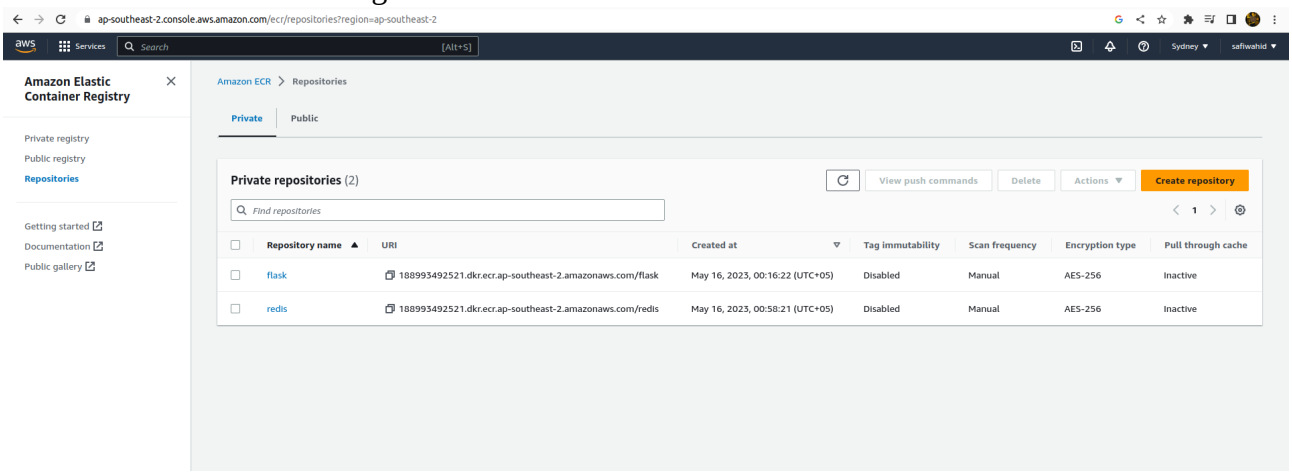
2.adding env to dockerfile

day_1_microservices > integrating_flask_redis > Dockerfile

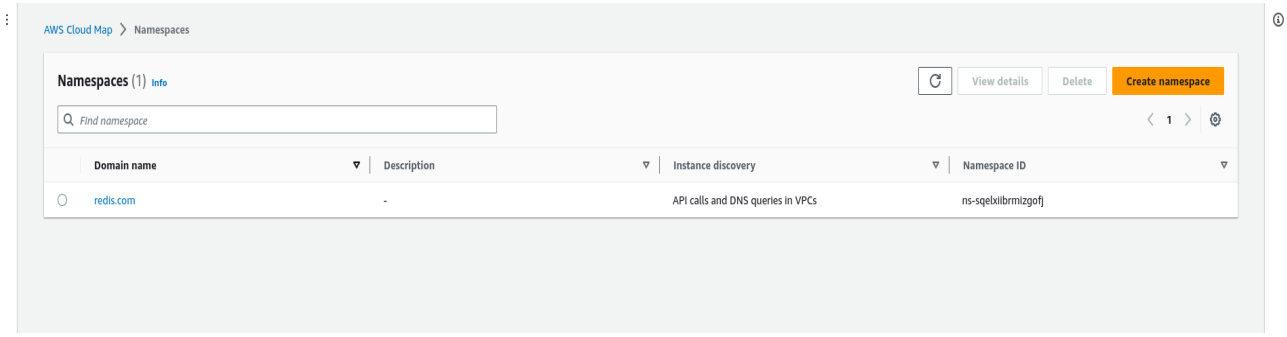
```
FROM python:3.7-alpine

WORKDIR /code

ENV FLASK_APP=app.py
ENV FLASK_RUN_HOST=0.0.0.0
ENV SERVICE_DISCOVERY='redis'

RUN apk add --no-cache gcc musl-dev linux-headers
COPY requirements.txt requirements.txt

RUN pip install -r requirements.txt

EXPOSE 5000

COPY . .

CMD ["flask", "run"]
```

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

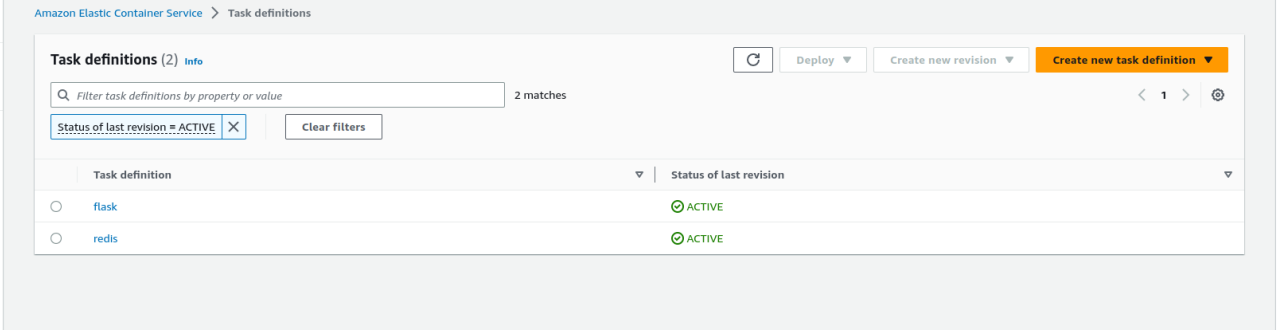
3. Pushed flask and redis images to ECR.



4. Creating namespaces and services.

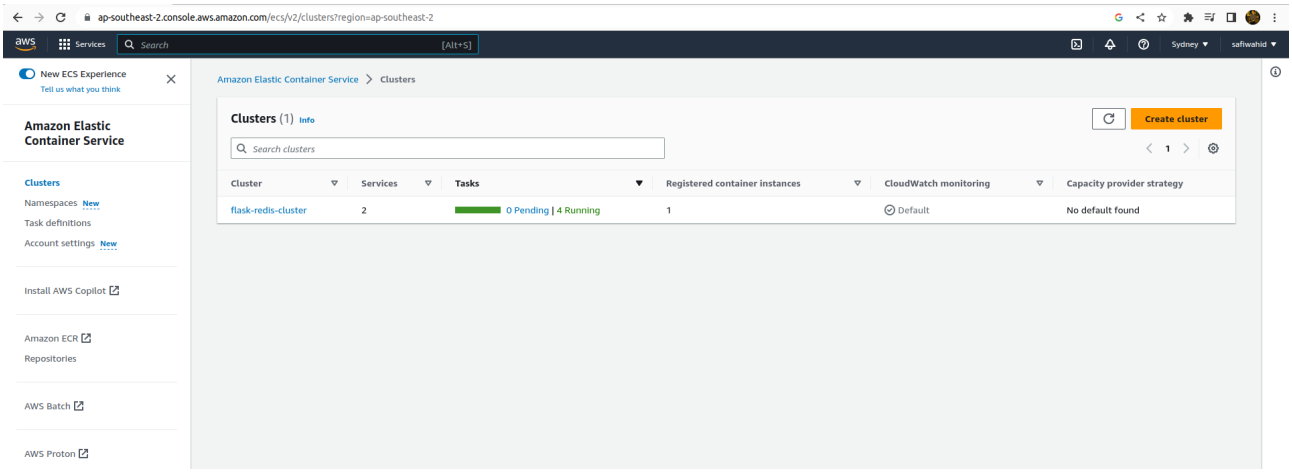


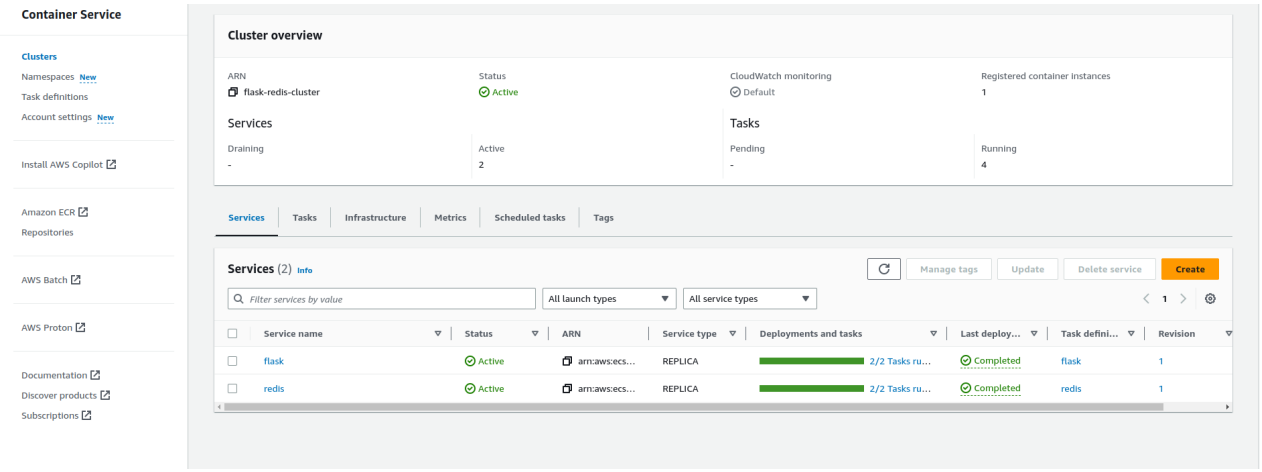
5. Creating Task definitions for flask and redis with port mappings and configurations.



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

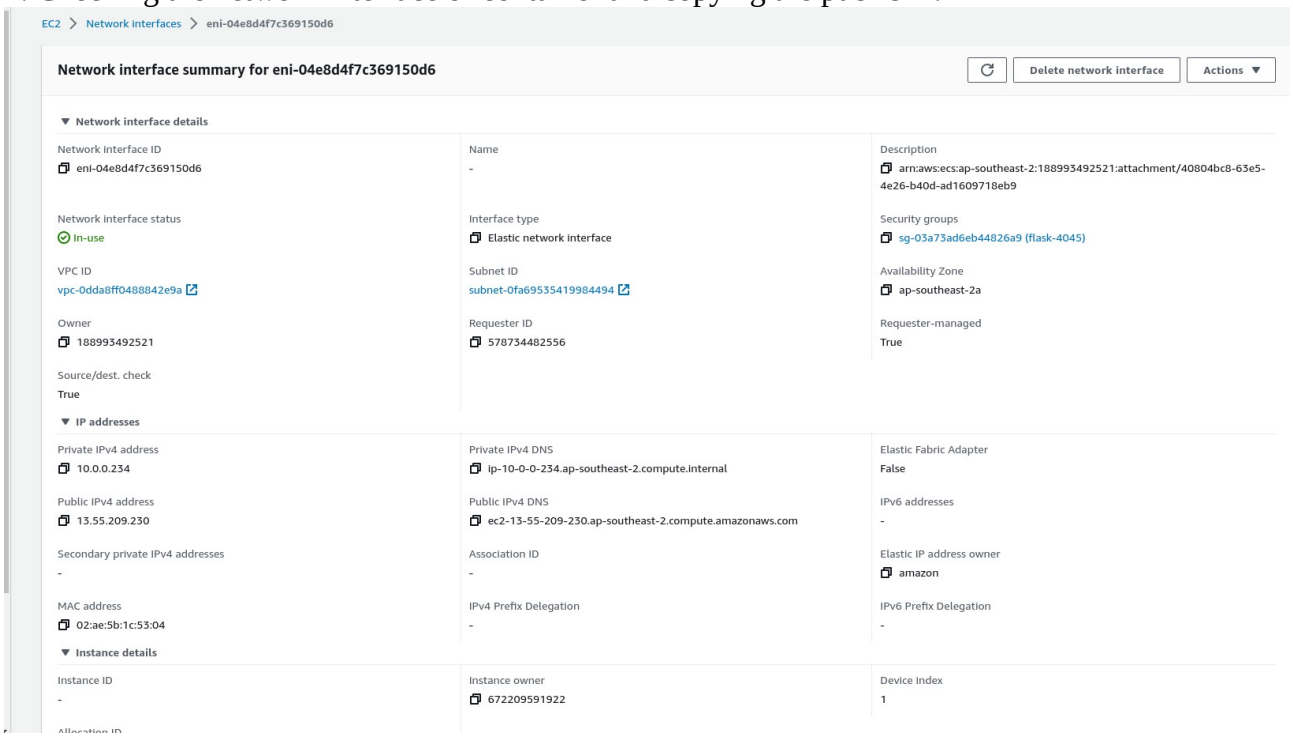
6. Adding services the cluster.





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.



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.