DevOps deployment options.

Step 1

- push docker images to amazon ECR.
- Create ECR vpc endpoint so downloading of images is across AWS backbone.

Step 2

- For all these below the roles should be created with the least permissions possible and all infrastructure should be created using IAC.

An EC2 instance.

- SSH into instance.
- Install docker
- download both instances onto this ec2 instance and run them locally.
- Since port 5000 is the api port, need to add security group rule for this as well so it can accept connections to this port from within the security group so only the ec2 instance can call the endpoint.
- Since port 8080 is the webapp endpoint, would need this to be publicly accessible.

ECS cluster

- deploy both docker containers on this ECS cluster.
- Would need to create a task definition per container and deploy both in the same container.
- Deploy the api cluster in a private subnet.
- Deploy webap in a public subnet.
- Add security group rules to allow for connections between the two respective containers security groups and subnets to work successfully.
- Since there are 3 environments for this
  - code pipeline which would get src code
  - code build step to build the image
  - code deploy the image to staging
    - manual approval
    - tear down staging image or leave it up depending.
  - Code deploy to QA
    - manual approval and teardown
  - code deploy to production.

S3 Bucket

- I know you can launch static websites in s3 so you could run the index.html in s3.
- I cant remember the specifics of it but I know it is possible as I have done it before.

Api Gateway

- create lambda function which contains the flask code.
- Create an api gateway trigger for this lambda function which opens it up as a rest endpoint.
- API gateway creating a microservices style endpoint which is a fully managed service from AWS and can scale enormously and manage 1000s of requests at the same time without latency issues.

Jenkins Server

- I am not familiar with a jenkins server.