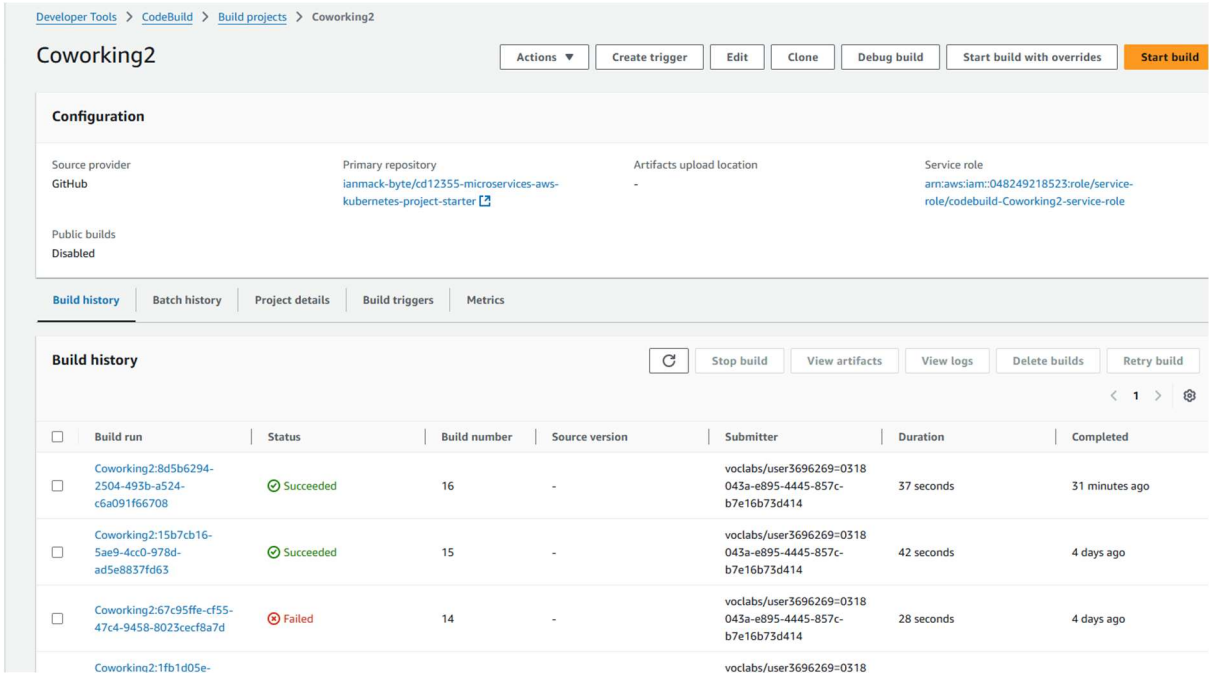
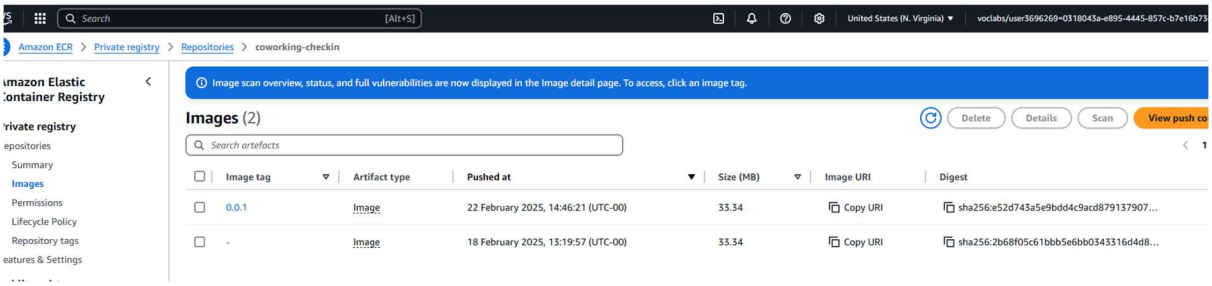
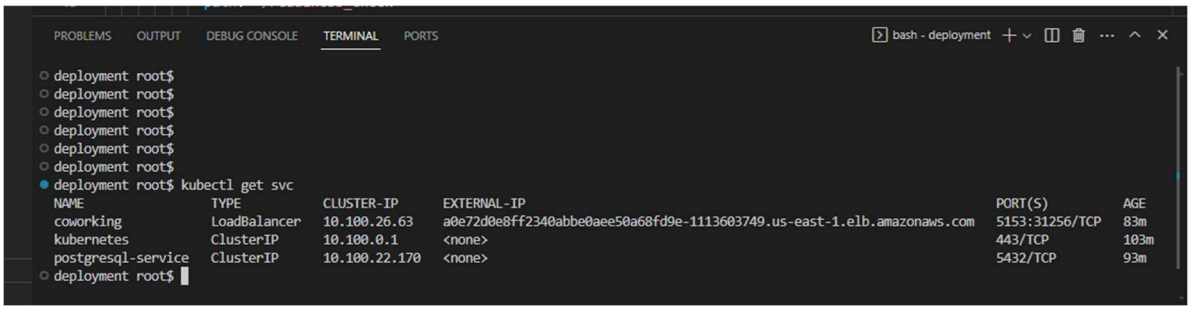
screenshot of AWS CodeBuild pipeline:



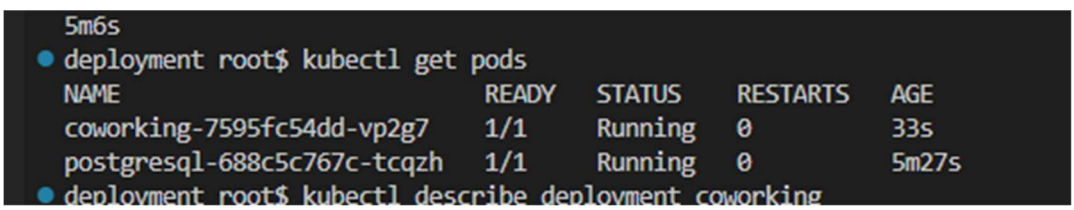
screenshot of AWS ECR repository for the application's repository.



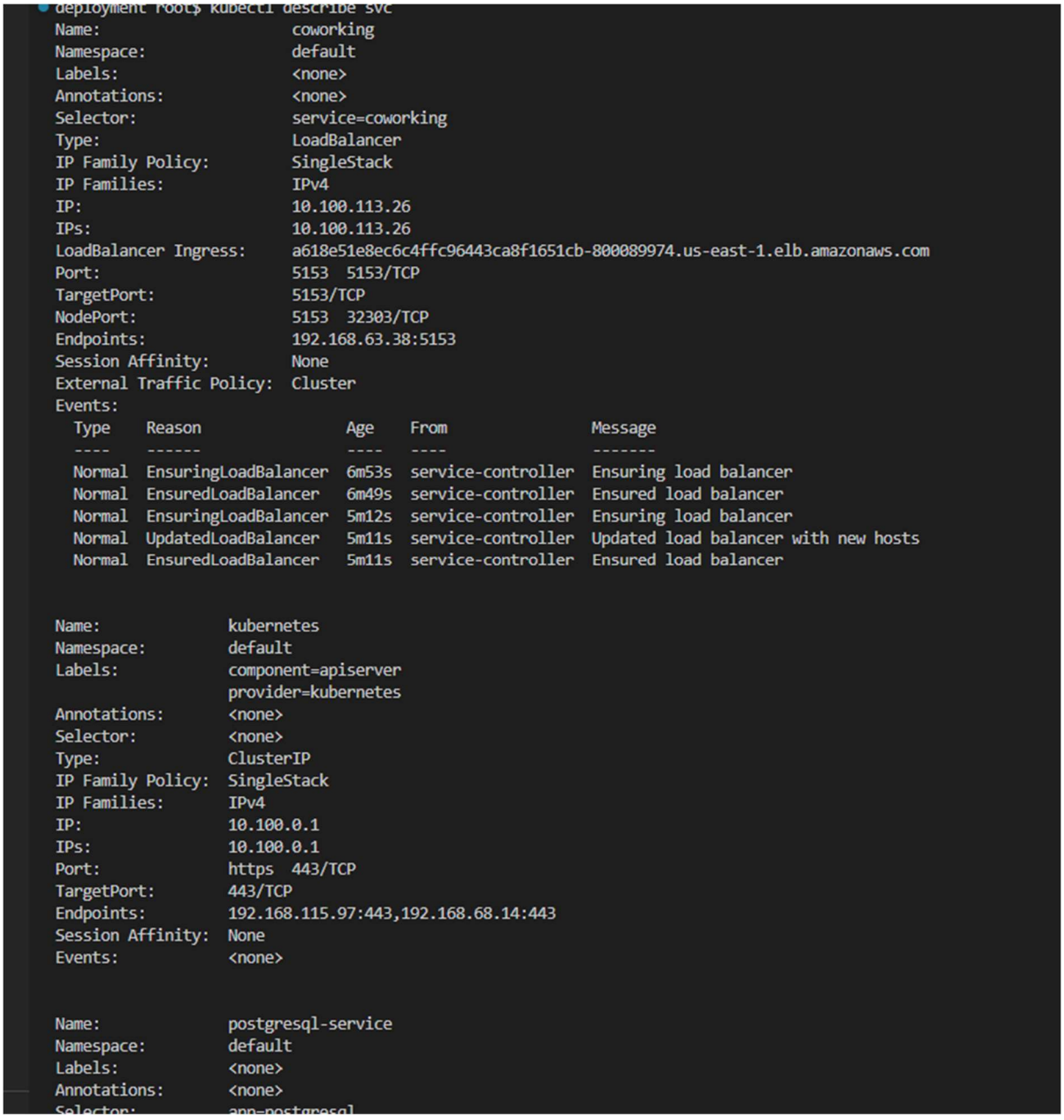
screenshot of running the kubectl get svc command.:



kubectl get pods:



kubectl describe svc:



```
deployment root$ kubectl describe svc
Name:                     coworking
Namespace:                default
Labels:                   <none>
Annotations:              <none>
Selector:                 service=coworking
Type:                     LoadBalancer
IP Family Policy:         SingleStack
IP Families:              IPv4
IP:                       10.100.113.26
IPs:                      10.100.113.26
LoadBalancer Ingress:     a618e51e8ec6c4ffc96443ca8f1651cb-800089974.us-east-1.elb.amazonaws.com
Port:                     5153  5153/TCP
TargetPort:               5153/TCP
NodePort:                 5153  32303/TCP
Endpoints:                192.168.63.38:5153
Session Affinity:         None
External Traffic Policy:  Cluster
Events:
  Type    Reason                Age    From                Message
  ----    ------                ----   ----                -------
  Normal  EnsuringLoadBalancer  6m53s  service-controller  Ensuring load balancer
  Normal  EnsuredLoadBalancer   6m49s  service-controller  Ensured load balancer
  Normal  EnsuringLoadBalancer  5m12s  service-controller  Ensuring load balancer
  Normal  UpdatedLoadBalancer   5m11s  service-controller  Updated load balancer with new hosts
  Normal  EnsuredLoadBalancer   5m11s  service-controller  Ensured load balancer


Name:              kubernetes
Namespace:         default
Labels:            component=apiserver
                   provider=kubernetes
Annotations:       <none>
Selector:          <none>
Type:              ClusterIP
IP Family Policy:  SingleStack
IP Families:       IPv4
IP:                10.100.0.1
IPs:               10.100.0.1
Port:              https  443/TCP
TargetPort:        443/TCP
Endpoints:         192.168.115.97:443,192.168.68.14:443
Session Affinity:  None
Events:            <none>


Name:              postgresql-service
Namespace:         default
Labels:            <none>
Annotations:       <none>
Selector:          app=postgresql
```

kubectl describe deployment coworking:

```
deployment root$ kubectl describe deployment coworking
Name:                   coworking
Namespace:              default
CreationTimestamp:      Wed, 12 Mar 2025 15:20:58 +0000
Labels:                 name=coworking
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               service=coworking
Replicas:               1 desired | 1 updated | 1 total | 1 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  service=coworking
  Containers:
   coworking:
    Image:      051150398741.dkr.ecr.us-east-1.amazonaws.com/coworking-checkin:0.0.1
    Port:       <none>
    Host Port:  <none>
    Liveness:   http-get http://:5153/health_check delay=5s timeout=2s period=10s #success=1 #failure=3
    Readiness:  http-get http://:5153/readiness_check delay=5s timeout=5s period=10s #success=1 #failure=3
    Environment Variables from:
      my-postgres-db-configmap  ConfigMap  Optional: false
    Environment:
      DB_PASSWORD:  <set to the key 'DB_PASSWORD' in secret 'my-postgres-db-password'>  Optional: false
    Mounts:         <none>
  Volumes:          <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   coworking-7595fc54dd (1/1 replicas created)
Events:
  Type    Reason             Age   From                   Message
  ----    ------             ----  ----                   -------
  Normal  ScalingReplicaSet  83s   deployment-controller  Scaled up replica set coworking-7595fc54dd to 1
```

screenshot of AWS CloudWatch Container Insights logs for the application.

/aws/containerinsights/coworking/application

Actions ▼ | View in Logs Insights | Start tailing | Se

Log group details

Log class Info
Standard

ARN
arn:aws:logs:us-east-1:048249218523:log-group:/aws/containerinsights/coworking/application:*

Creation time
43 minutes ago

Retention
Never expire

Stored bytes
-

Metric filters
0

Subscription filters
0

Contributor Insights rules
-

KMS key ID
-

Anomaly detection
Configure

Data protection
-

Sensitive data count
-

Field indexes
Configure

Transformer
Configure

Log streams | Tags | Anomaly detection | Metric filters | Subscription filters | Contributor Insights | Data protection | Field indexes | Transformer

Log streams (4)

Delete | Create log stream | Search all l

Filter log streams or try prefix search

Exact match | Show expired Info

Log stream	Last event time
ip-192-168-63-228.ec2.internal-application.var.log.containers.coworking-f9f8fc848-qrwm4_default_coworking-dd6	2025-02-22 15:11:29 (UTC)
ip-192-168-63-228.ec2.internal-application.var.log.containers.cloudwatch-agent-6rcgb_amazon-cloudwatch_otc-co	2025-02-22 14:53:44 (UTC)
ip-192-168-63-228.ec2.internal-application.var.log.containers.fluent-bit-gl4gb_amazon-cloudwatch_fluent-bit-0a6C	2025-02-22 14:41:32 (UTC)
ip-192-168-63-228.ec2.internal-application.var.log.containers.amazon-cloudwatch-observability-controller-manage	2025-02-22 14:41:29 (UTC)