

Project 3 Submission

GitHub Repo: <https://github.com/kevahwin/kubernetes-aws-project>

Dockerfile: <https://github.com/kevahwin/kubernetes-aws-project/blob/main/analytcs/Dockerfile>

Screenshot of AWS CodeBuild pipeline

The screenshot displays the AWS CodeBuild console interface. The left sidebar shows the navigation menu with options like Source, Artifacts, Build, and Pipeline. The main content area shows the build details for the project 'kev-coworking:5a11b482-ee3a-4421-9dd6-e33dfd7ece59'. The build status is 'Succeeded'. The build logs show the following steps:

- 1 [Container] 2024/12/30 15:56:03.461932 Running on CodeBuild On-demand
- 2 [Container] 2024/12/30 15:56:03.461981 Waiting for agent ping
- 3 [Container] 2024/12/30 15:56:03.864135 Waiting for DOWNLOAD_SOURCE
- 4 [Container] 2024/12/30 15:56:07.297558 Phase is DOWNLOAD_SOURCE
- 5 [Container] 2024/12/30 15:56:07.452971 CODEBUILD_SRC_DIR=/codebuild/output/src231031217/src/github.com/kevahwin/kubernetes-aws-project
- 6 [Container] 2024/12/30 15:56:07.453608 YML location is /codebuild/output/src231031217/src/github.com/kevahwin/kubernetes-aws-project/analytcs/buildspec.yml
- 7 [Container] 2024/12/30 15:56:07.456611 Setting HTTP client timeout to higher timeout for Github and Github Enterprise sources
- 8 [Container] 2024/12/30 15:56:07.456941 Processing environment variables
- 9 [Container] 2024/12/30 15:56:07.740633 No runtime version selected in buildspec
- 10 [Container] 2024/12/30 15:56:07.737055 Moving to directory /codebuild/output/src231031217/src/github.com/kevahwin/kubernetes-aws-project
- 11 [Container] 2024/12/30 15:56:07.880298 Unable to initialize cache download: no paths specified to be cached
- 12 [Container] 2024/12/30 15:56:07.873786 Configuring ssm agent with target id: codebuild:5a11b482-ee3a-4421-9dd6-e33dfd7ece59
- 13 [Container] 2024/12/30 15:56:07.987369 Successfully updated ssm agent configuration
- 14 [Container] 2024/12/30 15:56:07.987848 Registering with agent
- 15 [Container] 2024/12/30 15:56:08.069998 Phases found in YML: 3
- 16 [Container] 2024/12/30 15:56:08.069525 PRE_BUILD: 2 commands

Screenshot of AWS ECR repository for the application's repository

The screenshot displays the AWS ECR console interface. The left sidebar shows the navigation menu with options like Private registry, Public registry, and Settings. The main content area shows the 'Images' tab for the repository 'kev-udacity-coworking'. The table lists the following images:

Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest
13	Image	30 December 2024, 15:57:00 (UTC-00)	178.52	Copy URI	sha256:c55785feb1b8d9026d96467382a6d...
12	Image	29 December 2024, 01:09:19 (UTC-00)	178.51	Copy URI	sha256:d78e176bc566f5b3fa1215a26640c4...
11	Image	29 December 2024, 00:56:37 (UTC-00)	178.51	Copy URI	sha256:e14c31b4e759c5c8d9853773780a...
latest	Image	29 December 2024, 00:06:46 (UTC-00)	178.51	Copy URI	sha256:4893861cb2bc144649d13b12052e1...

Screenshot of `kubectl get svc` and `kubectl get pods` before coworking deployment

```
kevahwin@MacBookPro kubernetes-aws-project % kubectl get svc
NAME                TYPE        CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
kubernetes           ClusterIP   10.100.0.1     <none>         443/TCP    6h48m
postgresql-service   ClusterIP   10.100.48.158  <none>         5432/TCP   5h20m
kevahwin@MacBookPro kubernetes-aws-project % kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
postgresql-7fbffd5bff-hdjfq        1/1     Running   0          5h21m
kevahwin@MacBookPro kubernetes-aws-project %
```

Screenshot of `kubectl get svc`

```
kevahwin@MacBookPro deployment % kubectl get svc
NAME                TYPE        CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
coworking            LoadBalancer  10.100.129.148 aa305e3ffb61a46e190f025e69af44ef-1224298312.us-east-1.elb.amazonaws.com 5153:31941/TCP 12s
kubernetes           ClusterIP   10.100.0.1     <none>         443/TCP    6h56m
postgresql-service   ClusterIP   10.100.48.158  <none>         5432/TCP   5h29m
```

Screenshot of `kubectl get pods`

```
kevahwin@MacBookPro kubernetes-aws-project % kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
coworking-65fd597f94-rdc6m         1/1     Running   0          74s
postgresql-7fbffd5bff-fnr52        1/1     Running   0          18m
kevahwin@MacBookPro kubernetes-aws-project %
```

Screenshot of `kubectl describe svc <DATABASE_SERVICE_NAME>`

```
kevahwin@MacBookPro deployment % kubectl describe svc postgresql-service
Name:                postgresql-service
Namespace:           default
Labels:              <none>
Annotations:         <none>
Selector:            app=postgresql
Type:                ClusterIP
IP Family Policy:    SingleStack
IP Families:         IPv4
IP:                  10.100.48.158
IPs:                 10.100.48.158
Port:                <unset> 5432/TCP
TargetPort:          5432/TCP
Endpoints:           192.168.2.93:5432
Session Affinity:    None
Internal Traffic Policy: Cluster
Events:              <none>
```

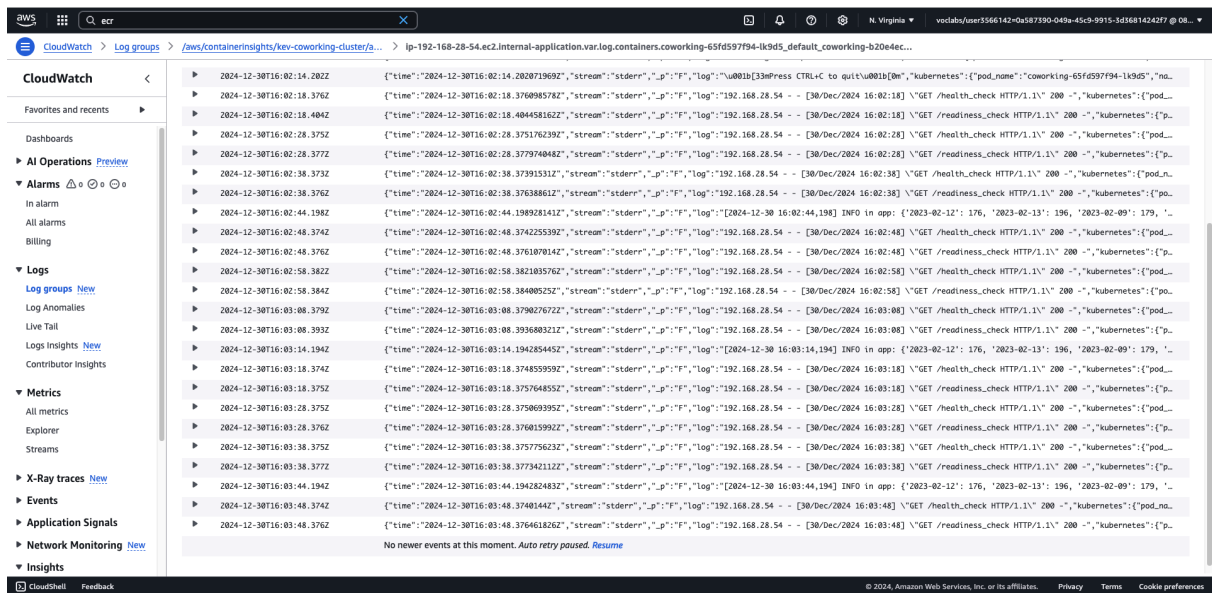
Screenshot of `kubectl describe deployment <SERVICE_NAME>`

```
kevahwin@MacBookPro kubernetes-aws-project % kubectl describe deployment coworking
Name: coworking
Namespace: default
CreationTimestamp: Sun, 29 Dec 2024 00:10:13 +0000
Labels: name=coworking
Annotations: deployment.kubernetes.io/revision: 2
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: public.ecr.aws/l2t4g5t0/kev-udacity-coworking:11
      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:
        DB_USERNAME: <set to the key 'POSTGRES_USER' in secret 'postgresql-secret'> Optional: false
        DB_PASSWORD: <set to the key 'POSTGRES_PASSWORD' in secret 'postgresql-secret'> Optional: false
        DB_HOST: <set to the key 'DB_HOST' of config map 'postgres-config'> Optional: false
        DB_PORT: <set to the key 'DB_PORT' of config map 'postgres-config'> Optional: false
        DB_NAME: <set to the key 'DB_NAME' of config map 'postgres-config'> Optional: false
      Mounts: <none>
      Volumes: <none>
      Node-Selectors: <none>
      Tolerations: <none>
  Conditions:
    Type          Status    Reason
    ----          -
    Available      True      MinimumReplicasAvailable
    Progressing    True      NewReplicaSetAvailable
    OldReplicaSets: coworking-cc6db8b8f (0/0 replicas created)
    NewReplicaSet:  coworking-65fd597f94 (1/1 replicas created)
  Events:
    Type          Reason          Age    From          Message
    ----          -
    Normal        ScalingReplicaSet   58m    deployment-controller    Scaled up replica set coworking-cc6db8b8f to 1
    Normal        ScalingReplicaSet   8m5s    deployment-controller    Scaled up replica set coworking-65fd597f94 to 1
    Normal        ScalingReplicaSet   7m47s    deployment-controller    Scaled down replica set coworking-cc6db8b8f to 0 from 1
```

All Kubernetes config files used for deployment (ie YAML files):

<https://github.com/kevahwin/kubernetes-aws-project/tree/main/deployment>

Screenshot of AWS CloudWatch logs for the application



`README.md` file in your solution that serves as documentation for your user to detail how your deployment process works:
<https://github.com/kevahwin/kubernetes-aws-project/blob/main/deployment/README.md>