**Cahllenge 1**

I have automated setup for this problem including the infrastructure setup using GCP IAC code , also I have designed this solution Kubernetes

1. **To create a Google Kubernetes Engine (GKE) cluster**

export CLUSTER=mediawiki-cluster

export ZONE=us-west1-a

export PROJECT\_ID=credible-bay-375810

gcloud container clusters create "$CLUSTER" --zone "$ZONE"

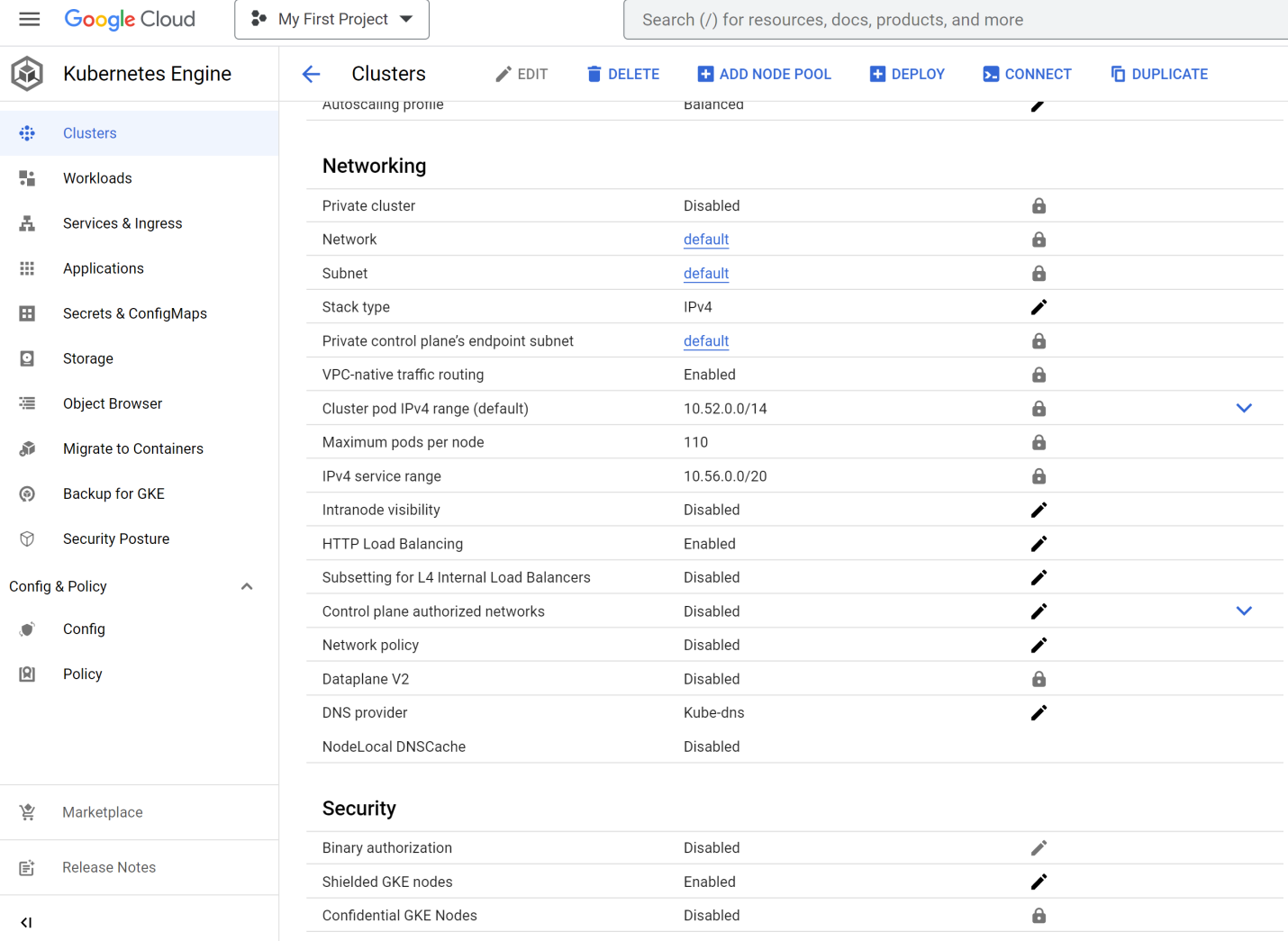
1. **Configure kubectl to connect to the new cluster:**

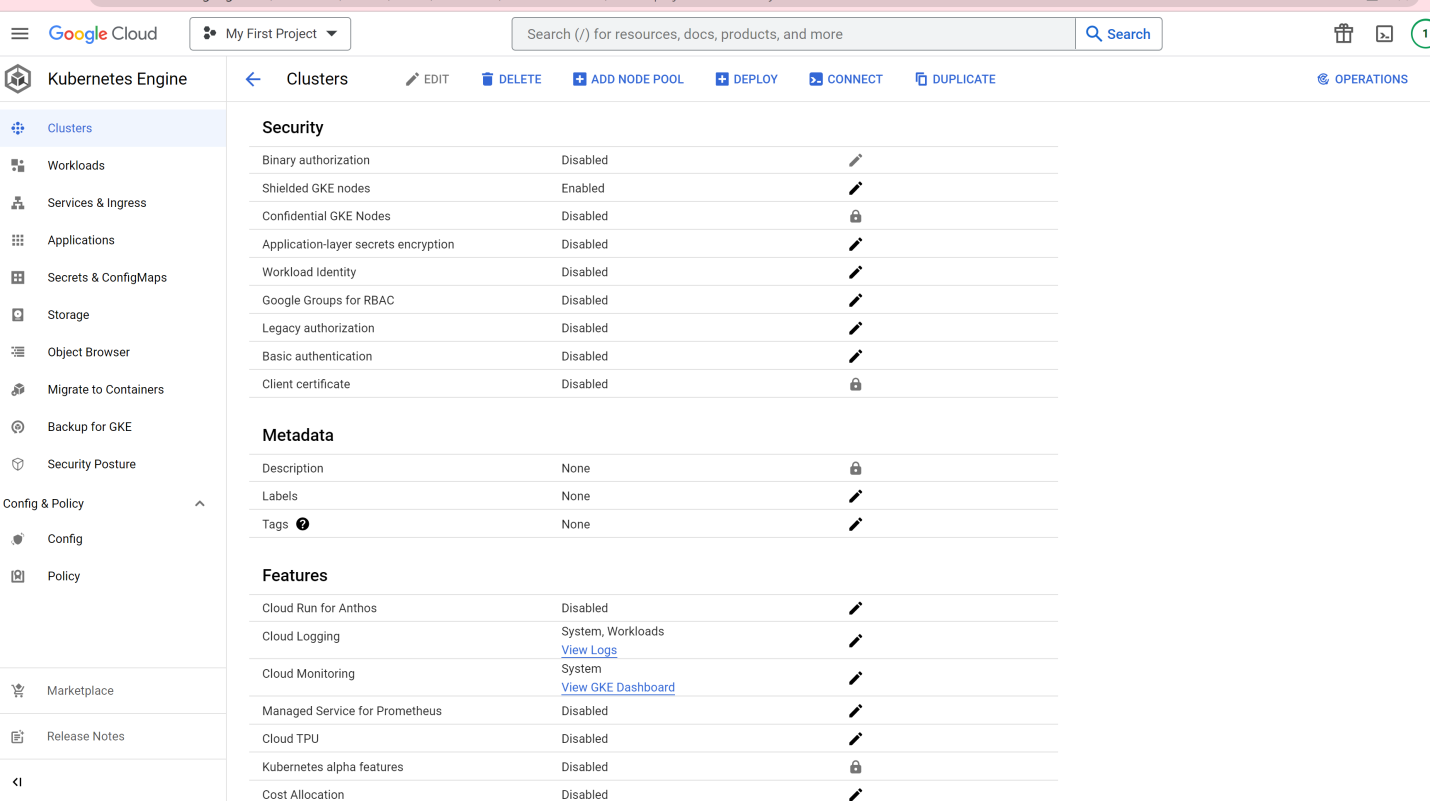
gcloud container clusters get-credentials "$CLUSTER" --zone "$ZONE"

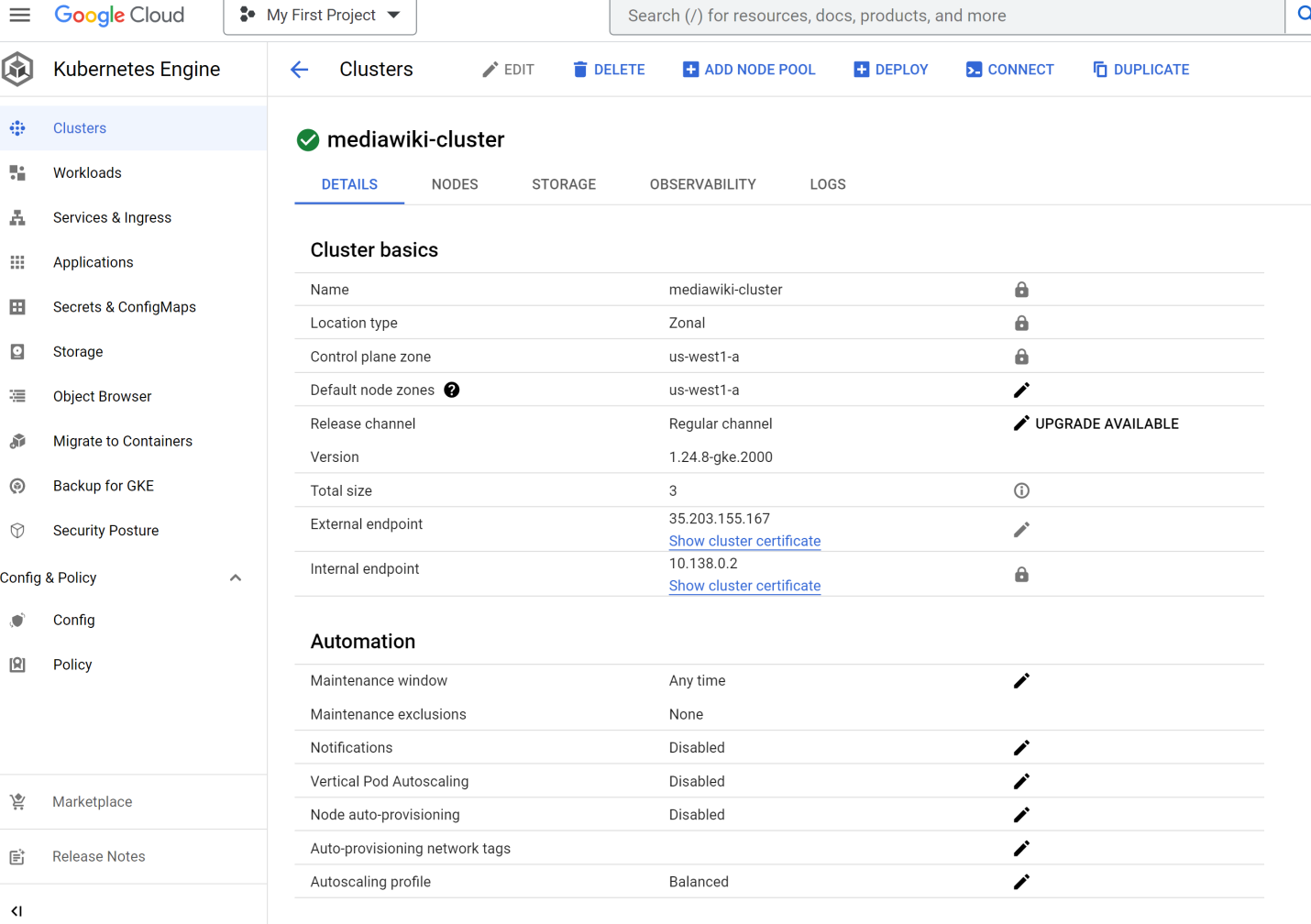
1. **Clone this repo and the associated tools repo**

git clone --recursive https://github.com/GoogleCloudPlatform/click-to-deploy.git

**Screenshot of Google Kubernetes Engine (GKE) cluster**







**Install the Application resource definition**

1. To set up your cluster to understand Application resources

kubectl apply -f "https://raw.githubusercontent.com/GoogleCloudPlatform/marketplace-k8s-app-tools/master/crd/app-crd.yaml"

### Install the app

1. Navigate to the mediawiki directory:
   1. cd click-to-deploy/k8s/mediawiki
   2. Configure the app with environment variables

export APP\_INSTANCE\_NAME=mediawiki-1

export NAMESPACE=default

1. Set the StorageClass name and Disk
   1. export STORAGE\_CLASS="standard"
   2. export MEDIAWIKI\_PERSISTENT\_DISK\_SIZE="5Gi"
   3. export DB\_PERSISTENT\_DISK\_SIZE="7Gi"
2. Expose the Service externally
   1. export PUBLIC\_SERVICE\_AND\_INGRESS\_ENABLED=true
3. Setup Image tag
4. export TAG="1.39.1-20230202-094708”
5. **Configure the container images:**

export IMAGE\_REGISTRY="marketplace.gcr.io/google"

export IMAGE\_MEDIAWIKI="${IMAGE\_REGISTRY}/mediawiki"

export IMAGE\_APACHE\_EXPORTER="${IMAGE\_REGISTRY}/mediawiki/apache-exporter:${TAG}"

export IMAGE\_MARIADB="${IMAGE\_REGISTRY}/mediawiki/mariadb:${TAG}"

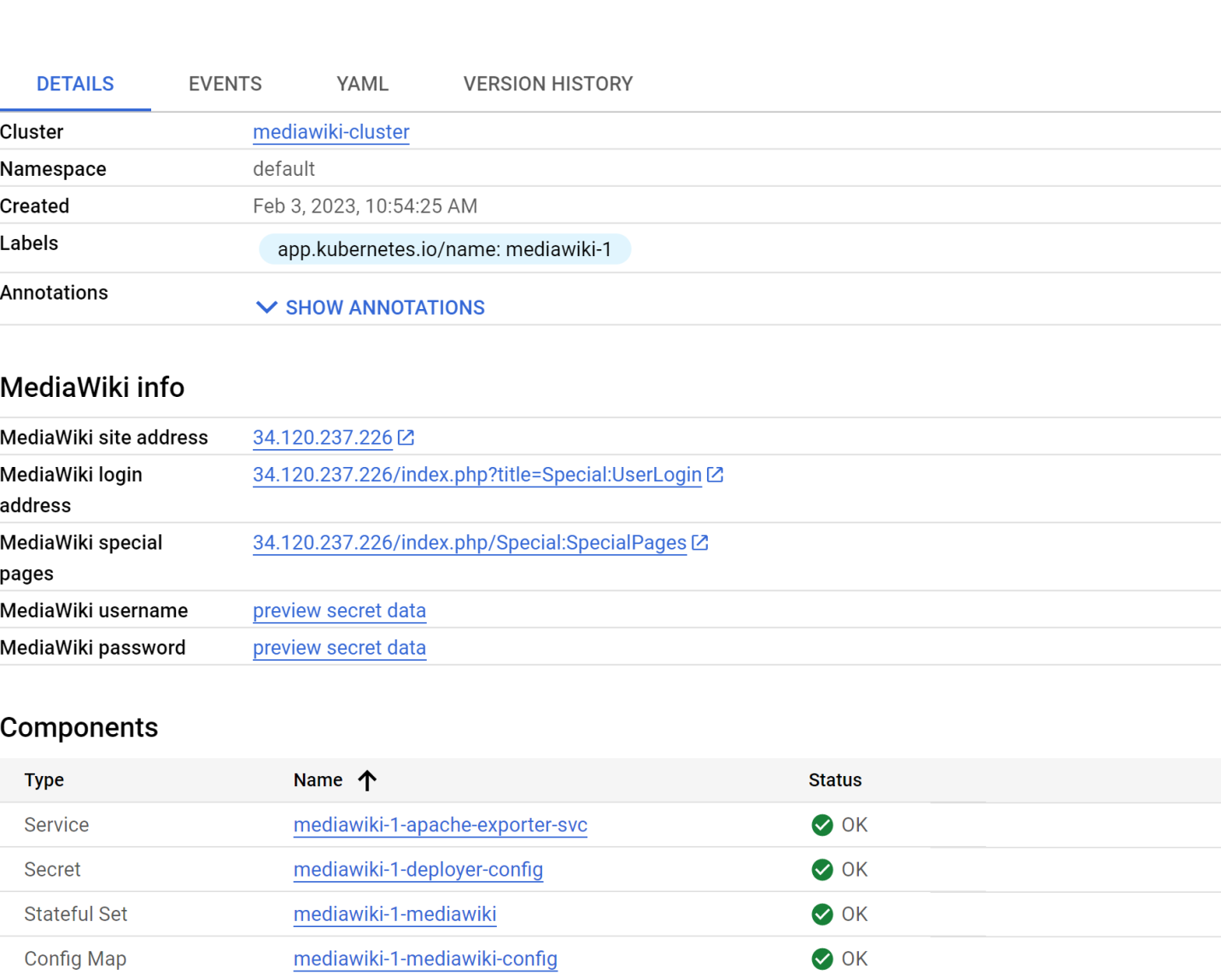
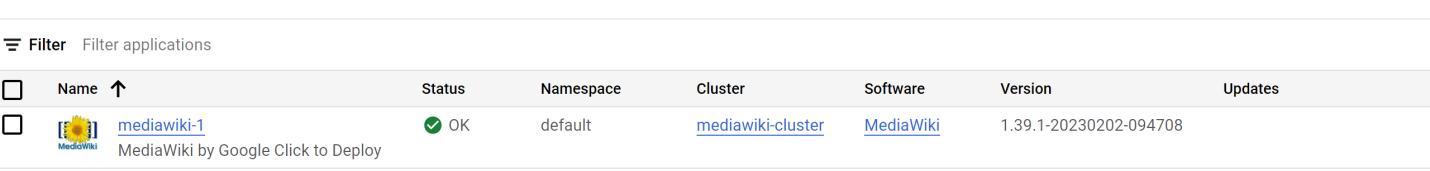
export IMAGE\_MYSQL\_EXPORTER="${IMAGE\_REGISTRY}/mediawiki/mysql-exporter:${TAG}"

export IMAGE\_METRICS\_EXPORTER="${IMAGE\_REGISTRY}/mediawiki/prometheus-to-sd:${TAG}"

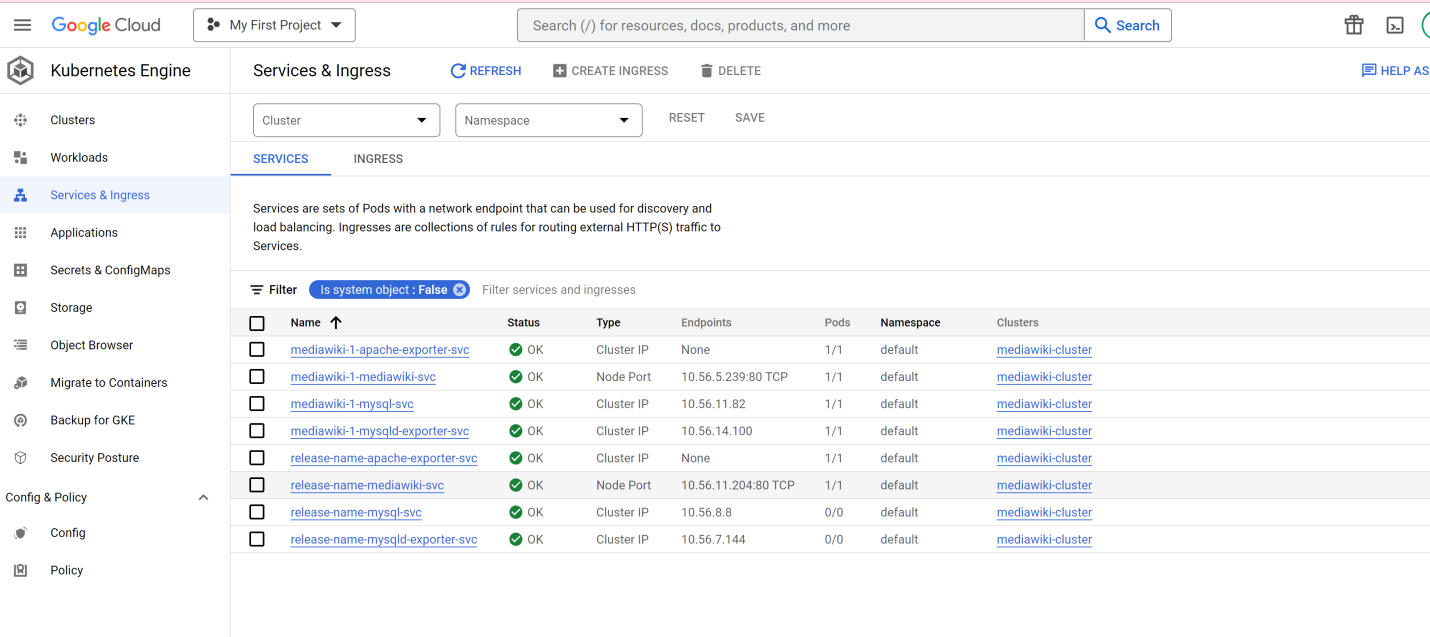
1. Expand the manifest template
   1. helm template chart/mediawiki \
   2. --namespace "${NAMESPACE}" \
   3. --set mediawiki.image.repo="${IMAGE\_MEDIAWIKI}" \
   4. --set mediawiki.image.tag="${TAG}" \
   5. --set mediawiki.admin.username="${MEDIAWIKI\_ADMIN\_USERNAME}" \
   6. --set mediawiki.admin.password="${MEDIAWIKI\_ADMIN\_PASSWORD}" \
   7. --set mediawiki.persistence.storageClass="${STORAGE\_CLASS}" \
   8. --set mediawiki.persistence.size="${MEDIAWIKI\_PERSISTENT\_DISK\_SIZE}" \
   9. --set db.image="${IMAGE\_MARIADB}" \
   10. --set db.rootPassword="${ROOT\_DB\_PASSWORD}" \
   11. --set db.mediawikiPassword="${MEDIAWIKI\_DB\_PASSWORD}" \
   12. --set db.persistence.size="${DB\_PERSISTENT\_DISK\_SIZE}" \
   13. --set db.exporter.image="${IMAGE\_MYSQL\_EXPORTER}" \
   14. --set db.exporter.password="${EXPORTER\_DB\_PASSWORD}" \
   15. --set apache.exporter.image="${IMAGE\_APACHE\_EXPORTER}" \
   16. --set enablePublicServiceAndIngress="${PUBLIC\_SERVICE\_AND\_INGRESS\_ENABLED}" \
   17. --set tls.base64EncodedPrivateKey="${TLS\_CERTIFICATE\_KEY}" \
   18. --set tls.base64EncodedCertificate="${TLS\_CERTIFICATE\_CRT}" \
   19. --set metrics.exporter.enabled="${METRICS\_EXPORTER\_ENABLED}" \
   20. --set metrics.image="${IMAGE\_METRICS\_EXPORTER}" \
   21. > "${APP\_INSTANCE\_NAME}\_manifest.yaml"

#### Apply the manifest to your Kubernetes cluster

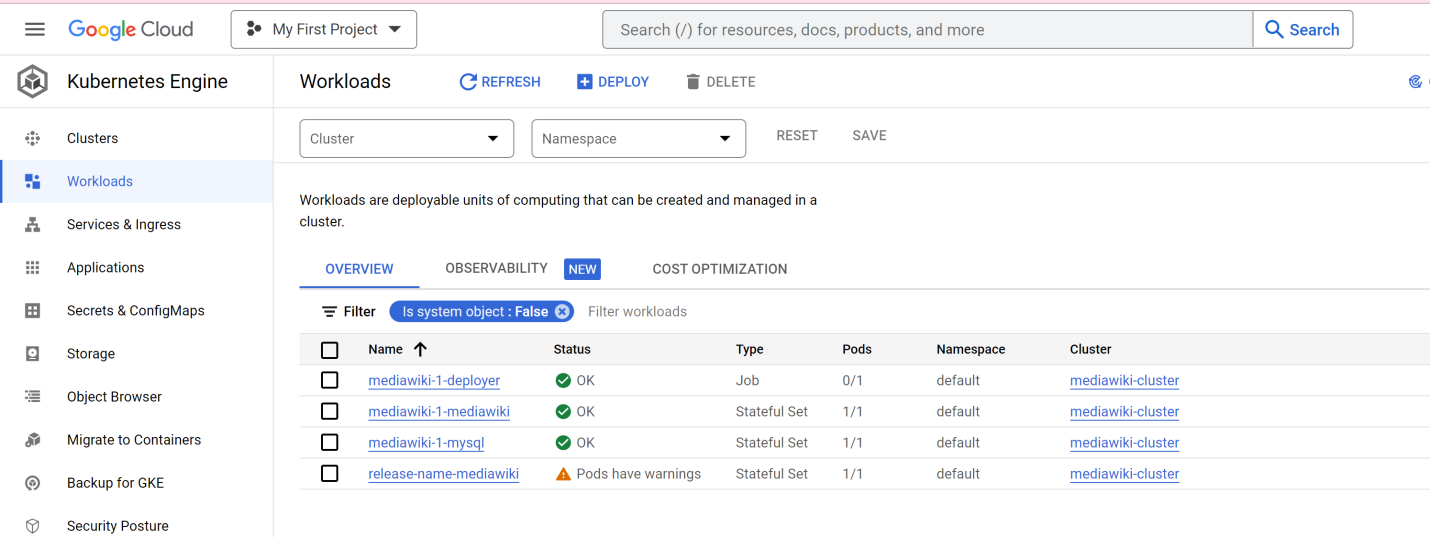
kubectl apply -f "${APP\_INSTANCE\_NAME}\_manifest.yaml" --namespace "${NAMESPACE}"

**Screenshot of App Installed** 

**Screenshot Of services**

****

**Screenshots of Workload**

****