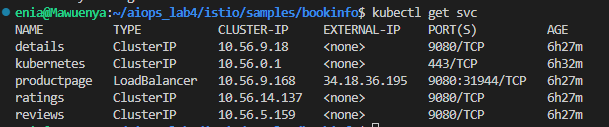
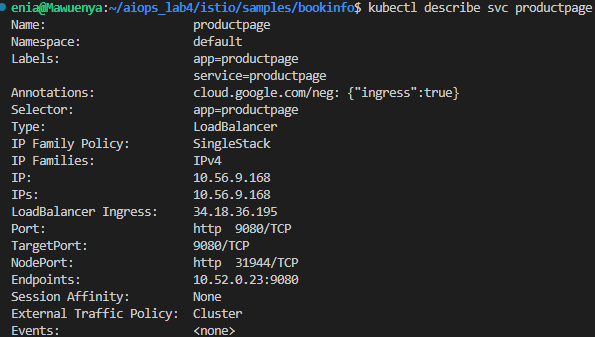
Lab 4: Task 1 Question 7  
  
Explore in detail these resources created by your deployment and write out fully how they function in response to user requests: all Services, Deployments, and Pods:  
kubectl get <resource type> for a given resource type to see the list of each  
kubectl describe <resource type> <resource ID> to see the details of a given resource item  
Kubectl logs <pod ID> to see the logs for a given pod

Answer:

1. Services: In Kubernetes, a service is an abstraction that specifies a logical collection of pods and an access policy for those pods. Services allow dependent Pods to be loosely coupled. One Service distributes traffic among several Pods.
   1. kubectl get svc to see a list of all services



* 1. kubectl describe svc <service-name> to describe a specific service



1. Deployments: Declarative updates for Pods and ReplicaSets are provided by a deployment. When you specify a desired state in a deployment, the controller gradually shifts the current state toward the desired state. You can provide deployments to make new ReplicaSets or delete current deployments and replace them with new ones that use all of their resources.
   1. kubectl get deployments to see a list of all deployments

A screen shot of a black screen

Description automatically generated

* 1. kubectl describe deployment <deployment-name> to describe a specific deployment

A screenshot of a computer program

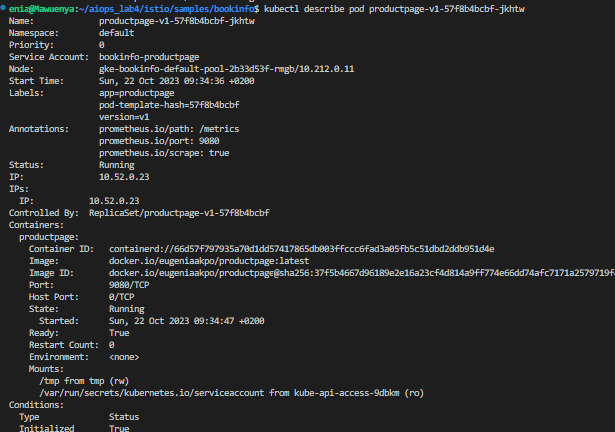
Description automatically generated

1. Pods: A Pod is the smallest and most basic unit of the Kubernetes object model that may be created or deployed. A Pod is a container that contains one or more processes operating on your cluster.
   1. kubectl get pods to see a list of all pods in the default namespace

A screenshot of a computer

Description automatically generated

* 1. kubectl describe pod to describe a specific pod



* 1. kubectl logs <pod-name> to view the logs for a specific pod

A computer screen shot of a computer program

Description automatically generated