Week 2

Learnings

Problems

Minikube tries to pull images from a remote registry, but I want it to look at my local registry

When trying to apply the deployment specification, the deployment was unable to pull any of my local images. This is because Minikube has it's own image registry that it attempts to pull from. You can see the images in Minikube's registry by running

```
$ minikube image ls

registry.k8s.io/pause:3.9
registry.k8s.io/kube-scheduler:v1.27.4
registry.k8s.io/kube-proxy:v1.27.4
registry.k8s.io/kube-controller-manager:v1.27.4
registry.k8s.io/kube-apiserver:v1.27.4
registry.k8s.io/etcd:3.5.7-0
registry.k8s.io/coredns/coredns:v1.10.1
gcr.io/k8s-minikube/storage-provisioner:v5
docker.io/kubernetesui/metrics-scraper:<none>
docker.io/kubernetesui/dashboard:<none>
```

There are a few ways to solve this:

1. Build your images directly within the Minikube environment.
2. Use `minikube load` to load images from your registry into Minikube's registry.
3. Build your image directly in Minikube.

I went with solution 1 since it seems the most easy to me. To do this, I ran

```
eval $(minikube docker-env)

cd backend && ./scripts/buildDocker.sh && cd ..
```

```
kubectl apply -f kubernetes/deployments/webapplication.yml
```

This didn’t work, so I tried solution 2 which worked on the first try

```
$ minikube image load kubernetes-learning/backend:latest
$ minikube image ls
registry.k8s.io/pause:3.9
registry.k8s.io/kube-scheduler:v1.27.4
registry.k8s.io/kube-proxy:v1.27.4
registry.k8s.io/kube-controller-manager:v1.27.4
registry.k8s.io/kube-apiserver:v1.27.4
registry.k8s.io/etcd:3.5.7-0
registry.k8s.io/coredns/coredns:v1.10.1
gcr.io/k8s-minikube/storage-provisioner:v5
docker.io/kubernetesui/metrics-scraper:<none>
docker.io/kubernetesui/dashboard:<none>
docker.io/kubernetes-learning/backend:latest
```

Surprisingly, the deployment immediately started working. I didn’t have to reapply the deployment.

Unable to connect to the server: dial tcp 192.168.49.2:8443: connect: no route to host

When running any kubectl commands, this would come up. I realized that kubernetes was not running on my computer, so I ran

```
minikube start
```