Github Link:
https://github.com/HaydenDuong/SIT323_Cloud_Native_Application_Development/tree/main/6.2C

Because 6.2C is a continuous task from 6.1P, thus, all the coding resources can be tracked back in folder “6.1P”

1/ Verifying the status of application through kubectl command

- Check all current-running pods by “kubectl get pods” (Image 1):

```
Lac T. Duong@Legion-7 MINGW64 ~/Desktop/SIT323_737 - Cloud Native Development/SIT323_Cloud_Native_Application_Development/6.1P (main)
$ kubectl get pods
NAME                                  READY   STATUS    RESTARTS   AGE
calculator-deployment-6c79f4fdfb-stlmw   1/1     Running   0          82m
```

Image 1

- Check all current-running services by “kubectl get services” (Image 2):

```
Lac T. Duong@Legion-7 MINGW64 ~/Desktop/SIT323_737 - Cloud Native Development/SIT323_Cloud_Native_Application_Development/6.1P (main)
$ kubectl get services
NAME                TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)          AGE
calculator-service  NodePort    10.102.230.221   <none>        80:30001/TCP     7s
kubernetes          ClusterIP   10.96.0.1        <none>        443/TCP          15h
```

Image 2

2/ Forward traffic from a local port to the Kubernetes service

- Type in command: “kubectl port-forward svc/calculator-service 8080:80” (Image 3):

```
Lac T. Duong@Legion-7 MINGW64 ~/Desktop/SIT323_737 - Cloud Native Development/SIT323_Cloud_Native_Application_Development/6.1P (main)
$ kubectl port-forward svc/calculator-service 8080:80
Forwarding from 127.0.0.1:8080 -> 3040
Forwarding from [::1]:8080 -> 3040
```

Image 3

3/ Test the application

- Can be tested by type in “/http:localhost:8080/add?a=20&b=30” (Image 4):

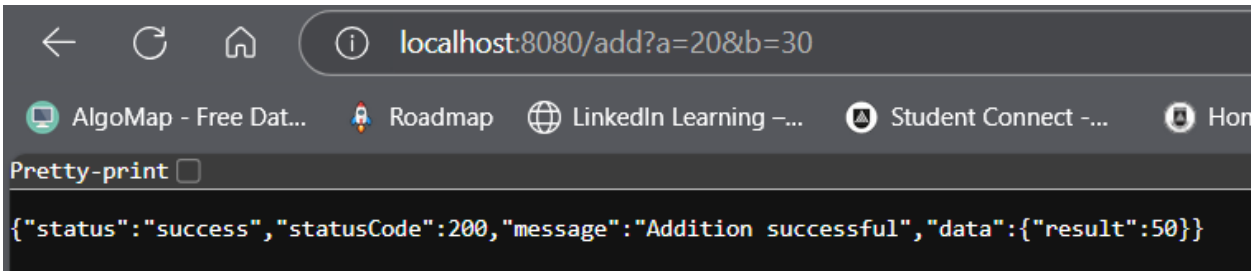

Image 4

- Or open a new terminal within vs code and type “curl localhost:8080”, it will display the html structure of the application (Image 5):

Image 5