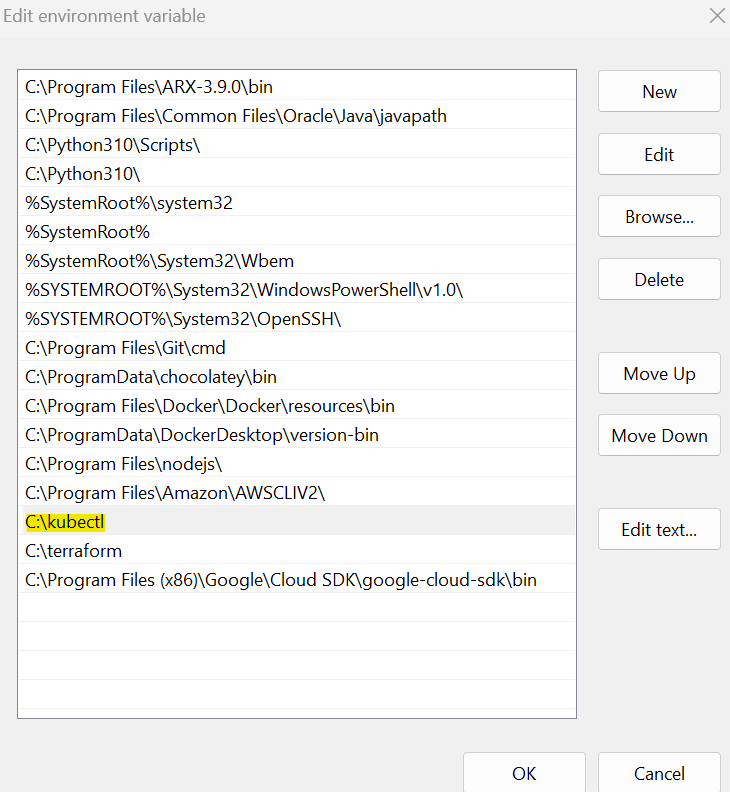
**7.1P: CREATING A KUBERNETES CLUSTER FOR A CONTAINERISED APPLICATION**

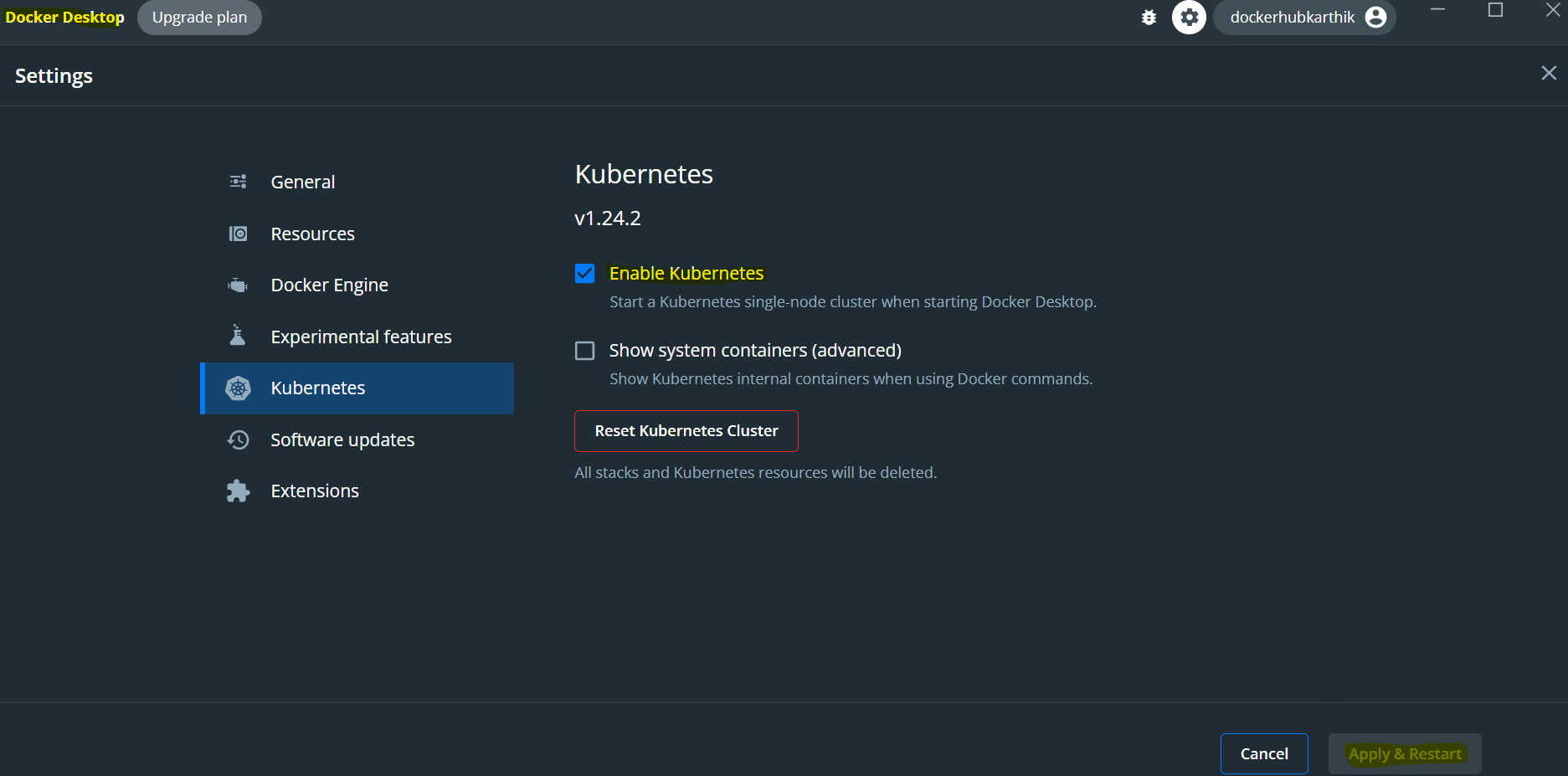
**• Setup the Kubernetes Cluster**

1. To setup the kubectl,

* Downloaded kubectl from <https://kubernetes.io/docs/tasks/tools/install-kubectl-windows/>
* Set the path environment variable to point to kubectl folder

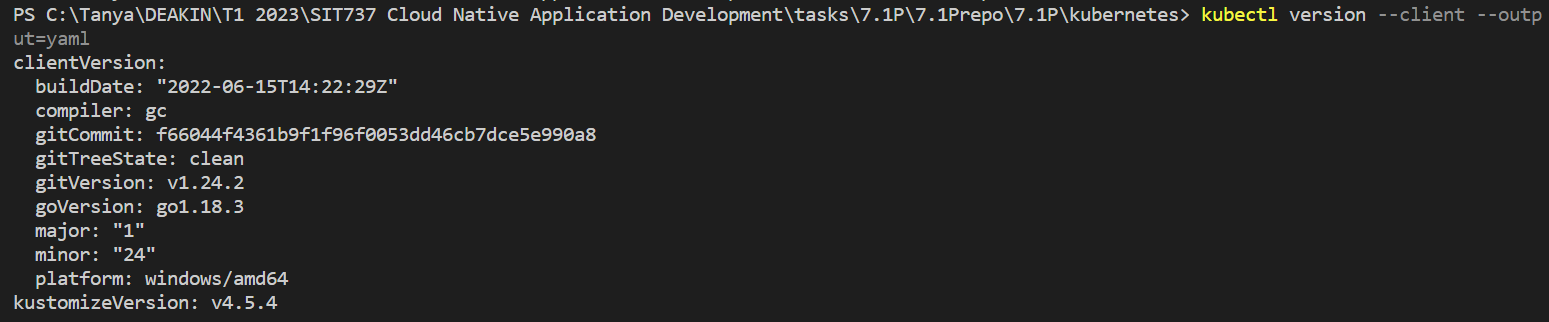


1. Enabled Kubernetes in Docker Desktop application > Apply & Restart

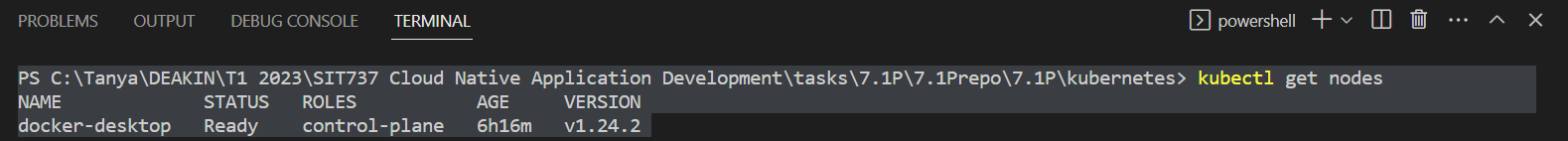


1. The below ensures that Kubernetes is enabled –

Command 🡪 kubectl version –client –output=yaml

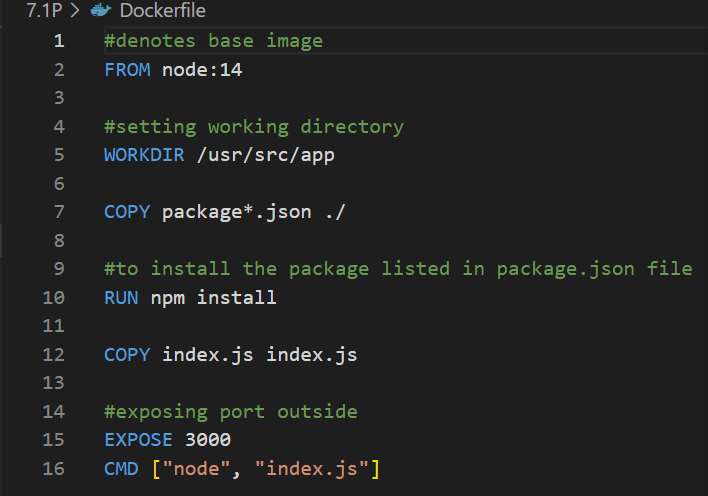


Also checked 🡪 kubectl get nodes

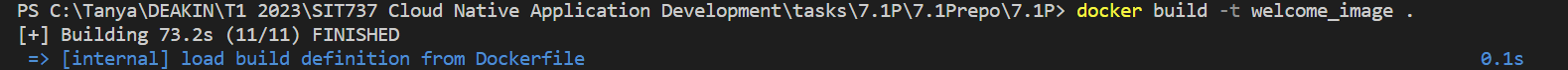


**• Create the Docker Image**

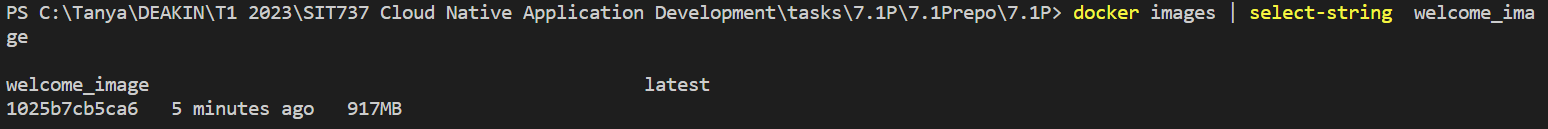
Created the docker file for Node.js application first –



Then, used the command 🡪 docker build -t welcome\_image .



This confirms the creation of the image, command 🡪 docker images | select-string welcome\_image



8. Push the Docker image to a registry

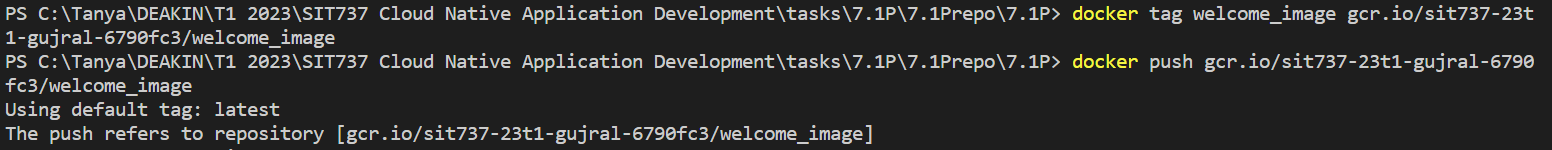
Tagged the image using 🡪 docker tag <image\_name> gcr.io/<project\_id>/<image\_name>

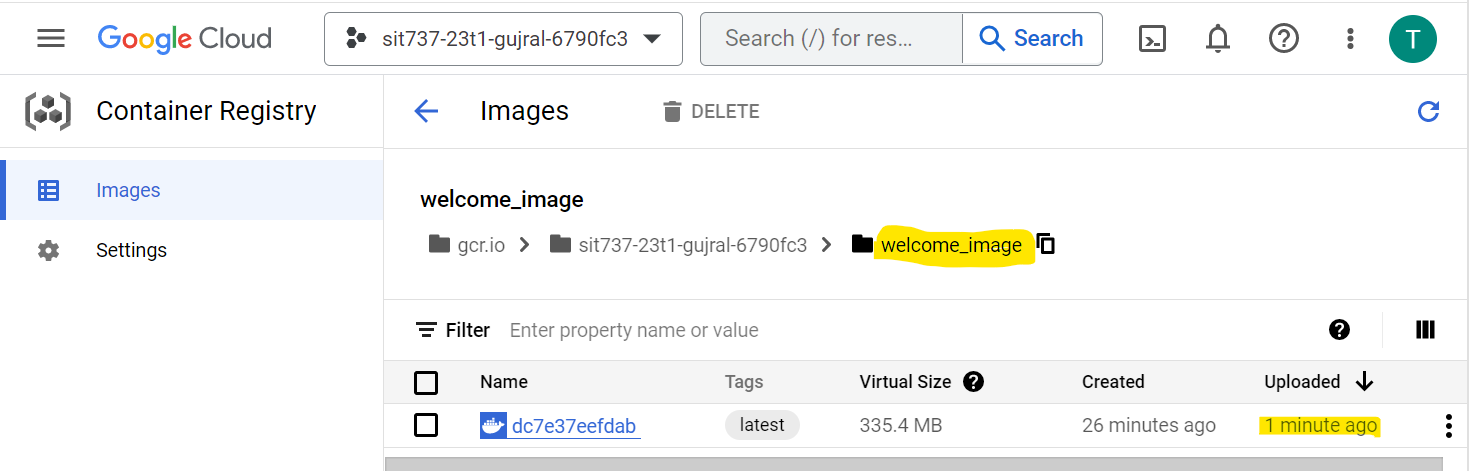
Command used 🡪 docker tag welcome\_image gcr.io/sit737-23t1-gujral-6790fc3/welcome\_image

And,

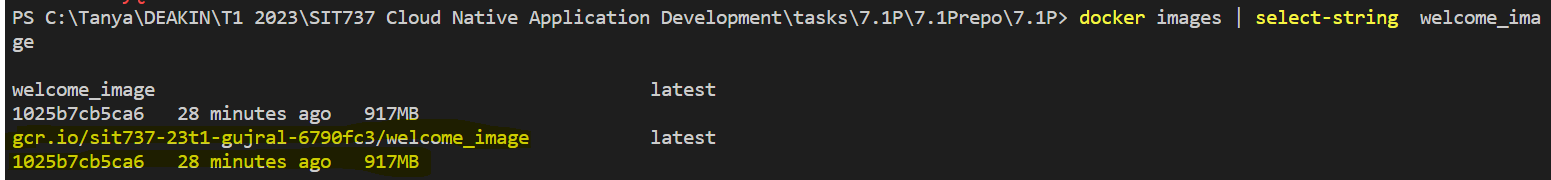
Pushed the image using 🡪 docker push gcr.io/<project\_id>/<image\_name>

Command used 🡪 docker push gcr.io/sit737-23t1-gujral-6790fc3/welcome\_image





Confirmed the push to GCR



**• Create the Kubernetes Deployment**

Kubernetes deployment configuration file for the application -

apiVersion: apps/v1

kind: Deployment

metadata:

  name: welcomemicroservice

spec:

  selector:

    matchLabels:

      app: welcomemicroservice

  replicas: 1

  template:

    metadata:

      labels:

        app: welcomemicroservice

    spec:

      containers:

      - name: welcomemicroservice

        image: gcr.io/sit737-23t1-gujral-6790fc3/welcome\_image

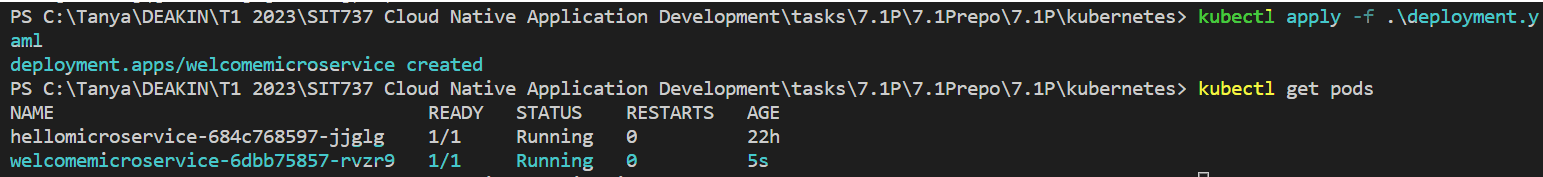
        ports:

        - containerPort: 3000

        imagePullPolicy: IfNotPresent

created the deployment service using the command 🡪 kubectl apply -f .\deployment.yaml

and checked using 🡪 kubectl get pods

****

**• Create the Kubernetes Service**

Kubernetes service configuration file for the application –

apiVersion: v1

kind: Service

metadata:

  name: welcomemicroservice-service

spec:

  type: LoadBalancer

  selector:

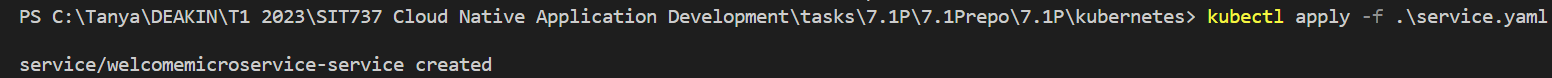
    app: welcomemicroservice

  ports:

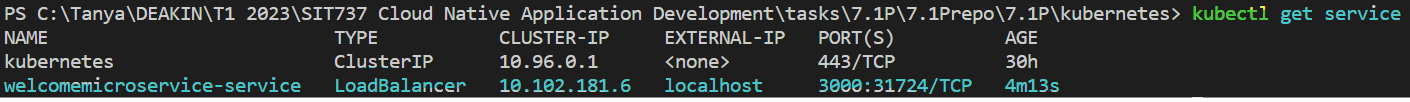
    - port: 3000

      targetPort: 3000

created the deployment service using the command 🡪 kubectl apply -f .\service.yaml



and checked using 🡪 kubectl get service



• Screenshots or video of the deployed application running on the Kubernetes cluster

