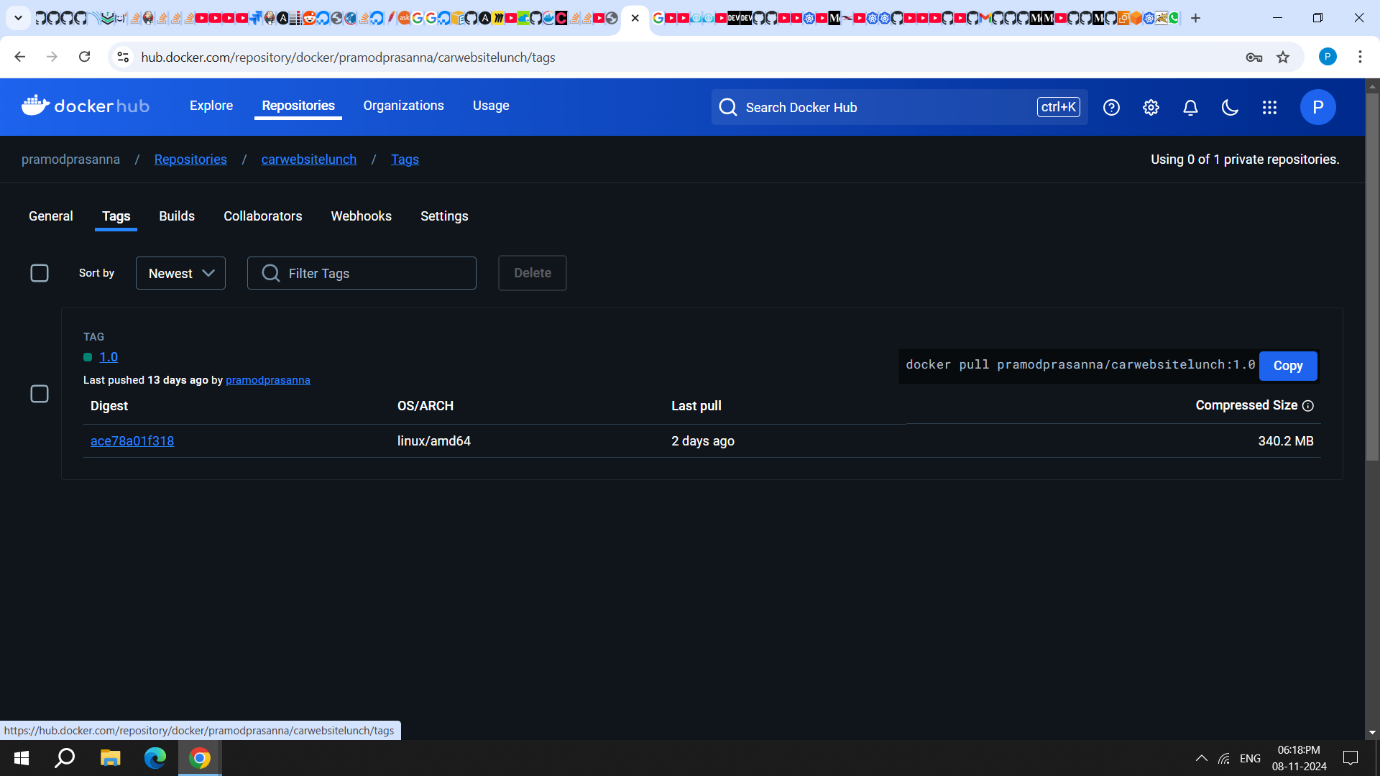
CAR WEBSITE

Below is the docker image of carwebsite. (in this image tomcat installed and webapp is also got downloaded from nexus to webapps folder ) so In this Kubernetes I am using that image only.



Vi cardeployment.yaml

---

apiVersion: apps/v1

kind: Deployment

metadata:

name: car

spec:

replicas: 1

selector:

matchLabels:

app: car

template:

metadata:

labels:

app: car

spec:

containers:

- name: car

image: pramodprasanna/carwebsitelunch:1.0

ports:

- containerPort: 80

Vi service.yaml

1>

---

apiVersion: v1

kind: Service

metadata:

name: car

spec:

selector:

app: car

ports:

- protocol: TCP

port: 80

targetPort: 8080

type: NodePort

or

2>

---

apiVersion: v1

kind: Service

metadata:

name: car

spec:

selector:

app: car

type: NodePort

ports:

- protocol: TCP

port: 80

targetPort: 8080

nodePort: 30583 # Manually setting the NodePort in service file not go and made change in edit svc

commands:

kubectl apply -f car.yaml

kubectl apply -f service.yaml

check once pods are in running state.

kubectl get nodes -w -o wide

kubectl get pods

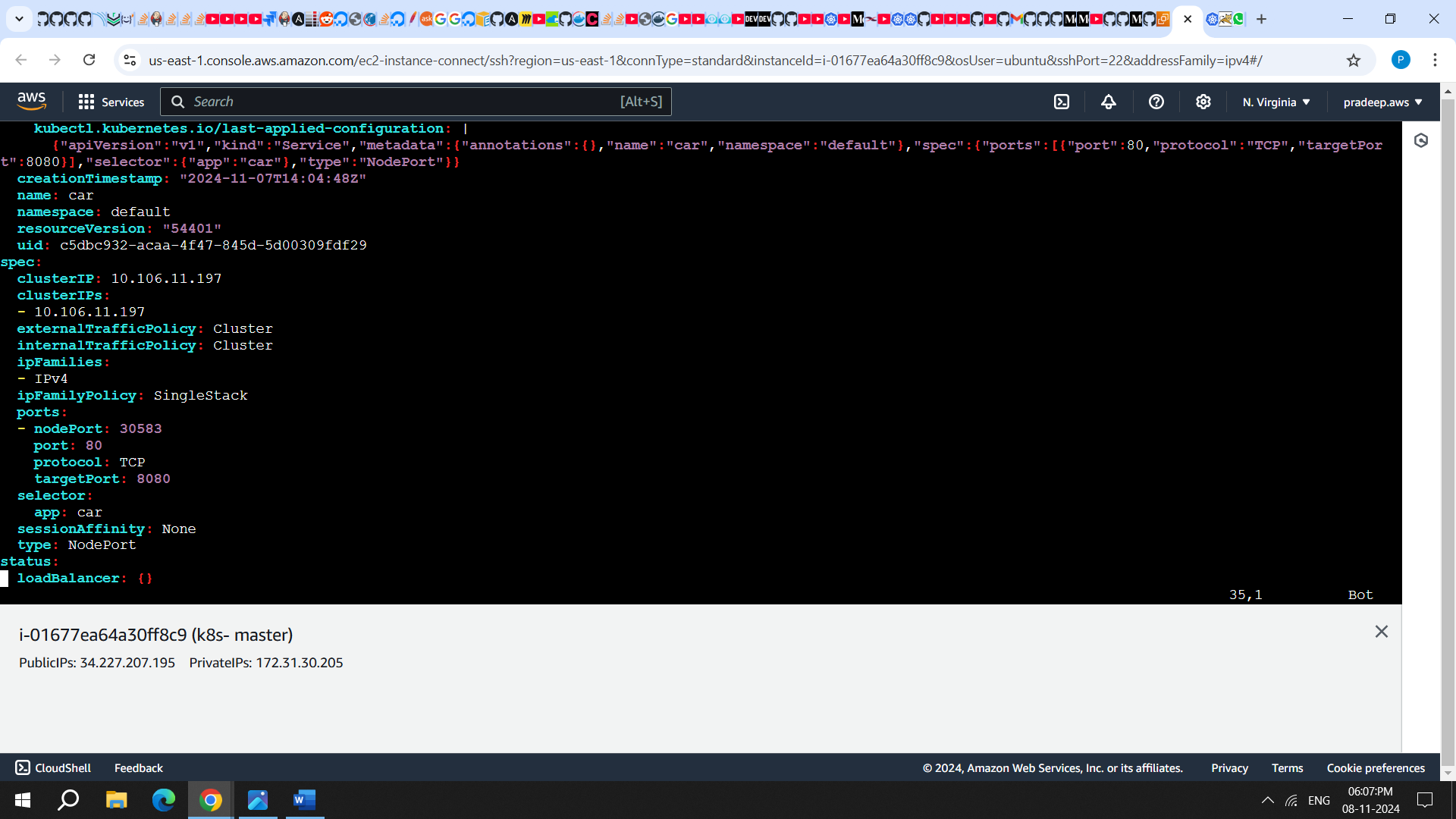
kubectl get svc

kubectl logs car-69478c79f-lxqxz

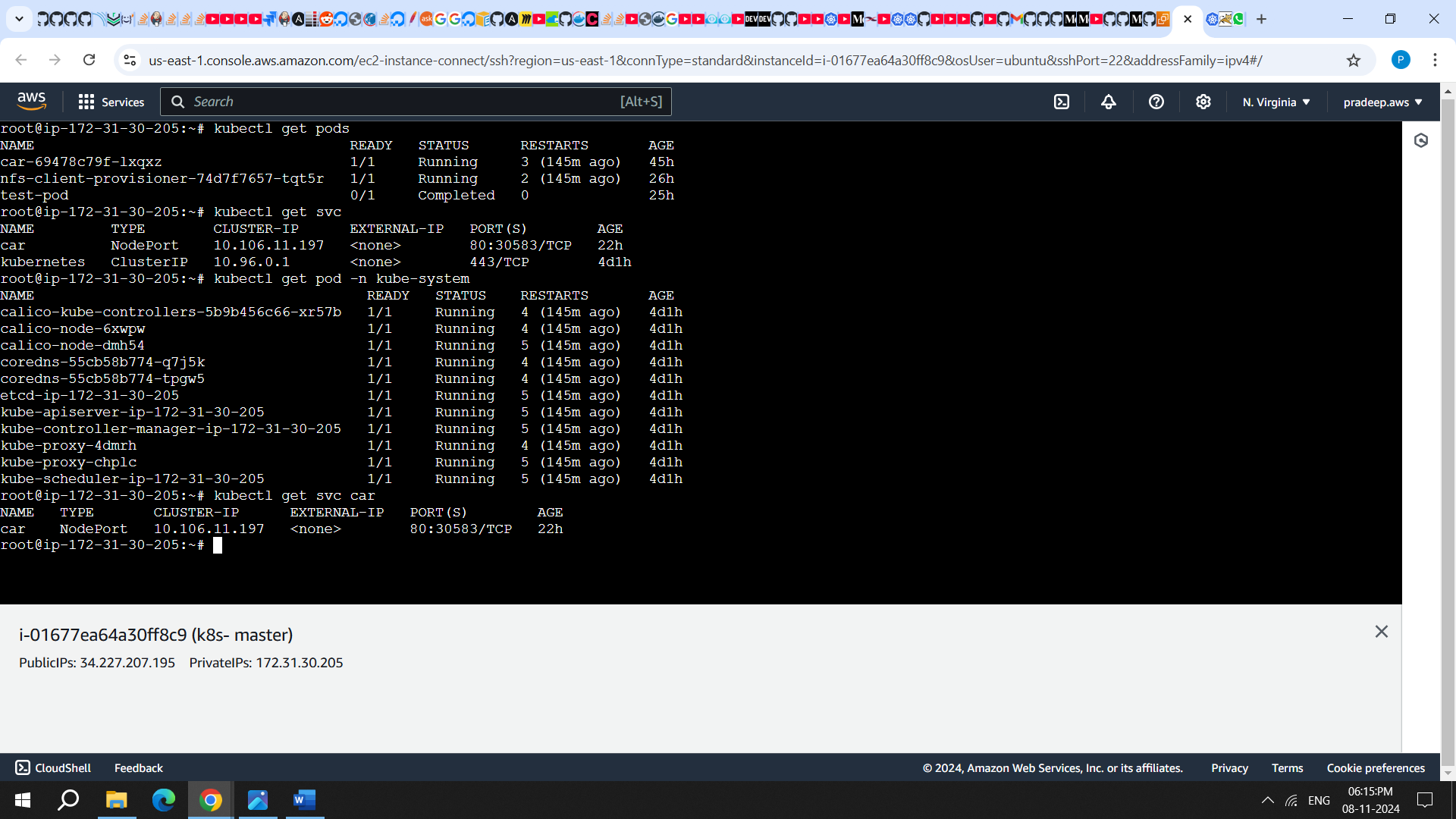
if you used 1> service.yaml file

kubectl edit svc car

then edit in svc car nodeport: 30583 after changes do apply command

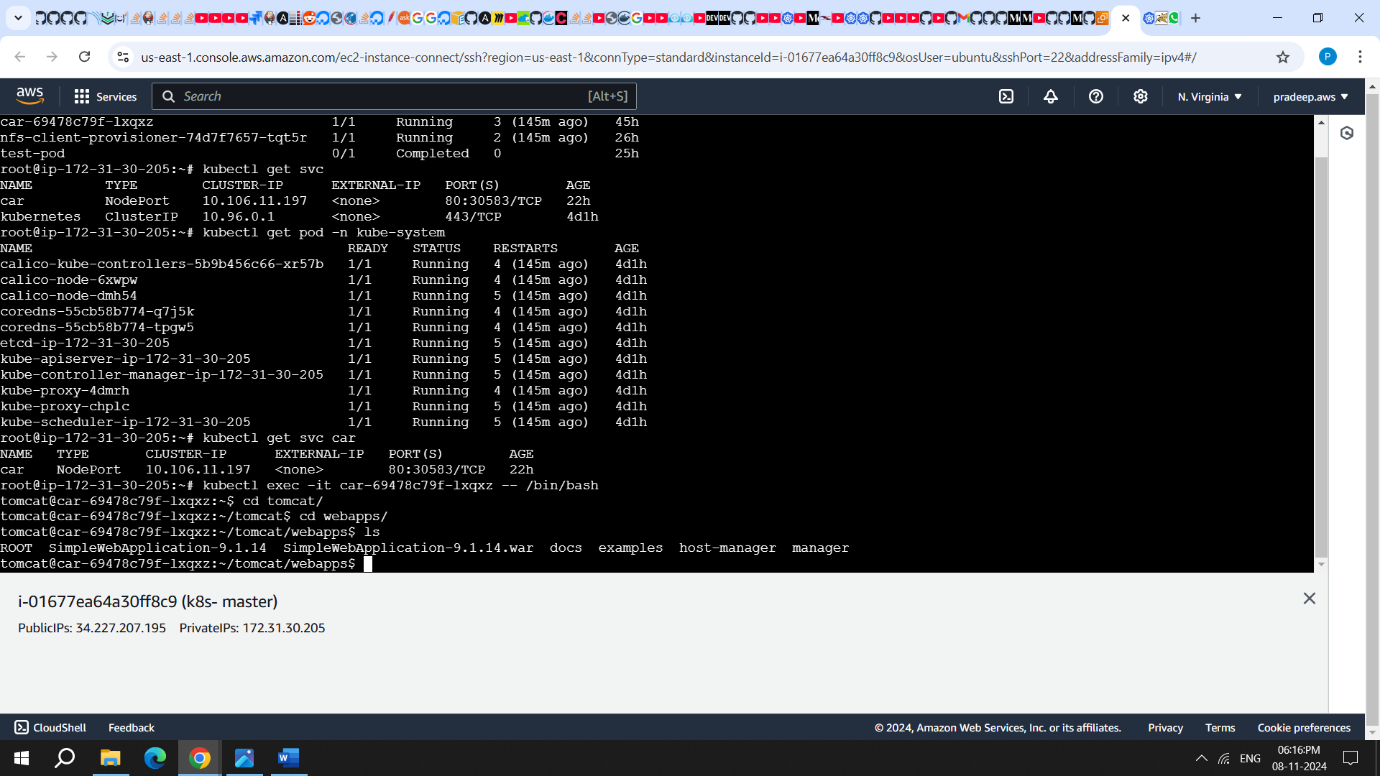


kubectl apply -f service.yaml



below command is used to go inside the container pod

kubectl exec -it car-69478c79f-lxqxz -- /bin/bash



then here check your webapps folder is your project (.war) is there are not

then also check tomcat-user.xml and META/context file

if its not configured do changes then exit from pod.

To describe.

then if you want to describe the pod of yours

kubectl describe pod car-69478c79f-lxqxz

URL:

http://slavepublicIP:30583/SimpleWebApplication-9.1.14/