############################# Source Code Management ################################

# Create Application repo in github

https://github.com/ToanLeH/sd2079\_msa

# Create Infrastructure repo in github

<https://github.com/ToanLeH/sd2079_devops_ci_cd/tree/main>

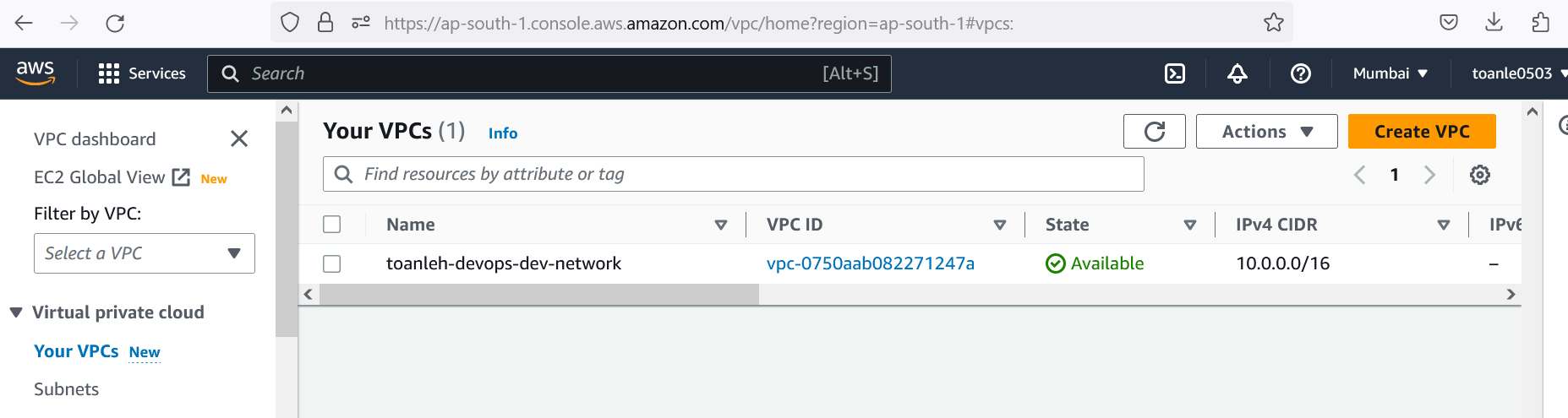
############################# Provision AWS Resource ################################

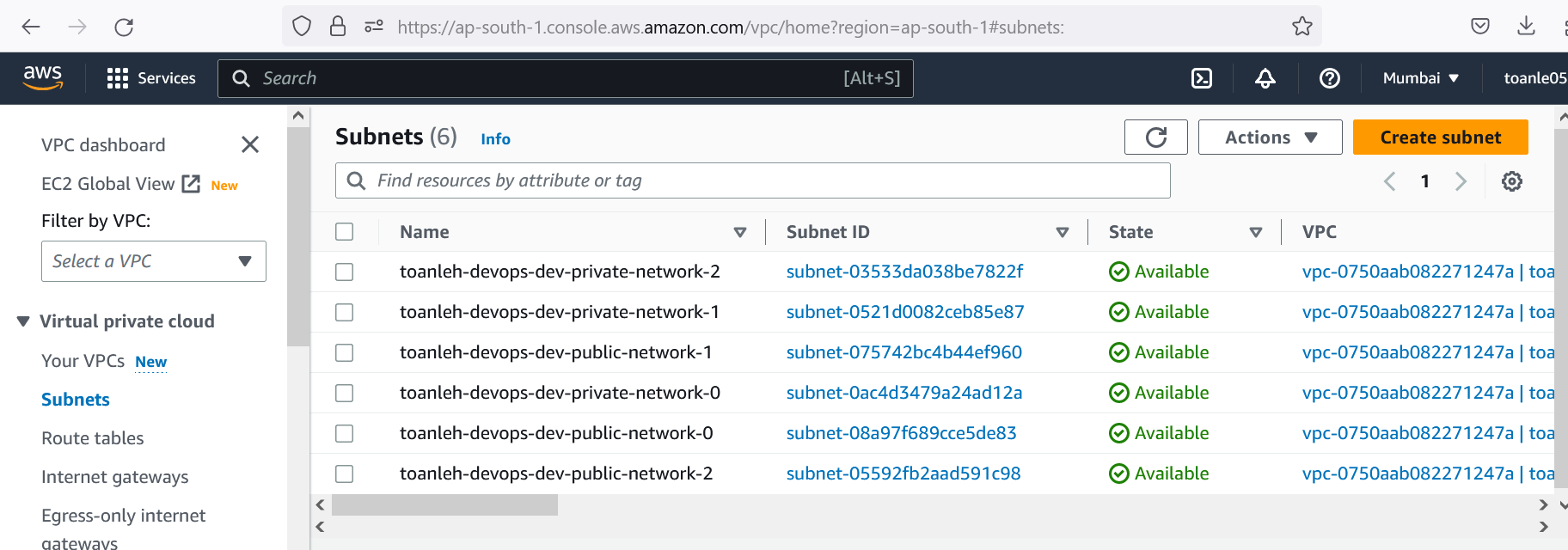
# Clone Infrastructure code and deploy aws resouce using terraform (PVC, EC2, EKS)

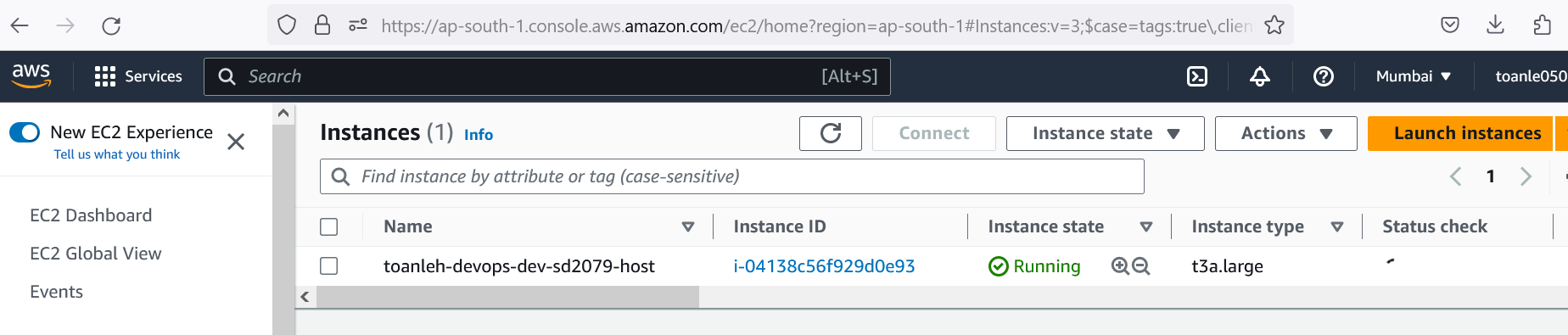
Terraform init

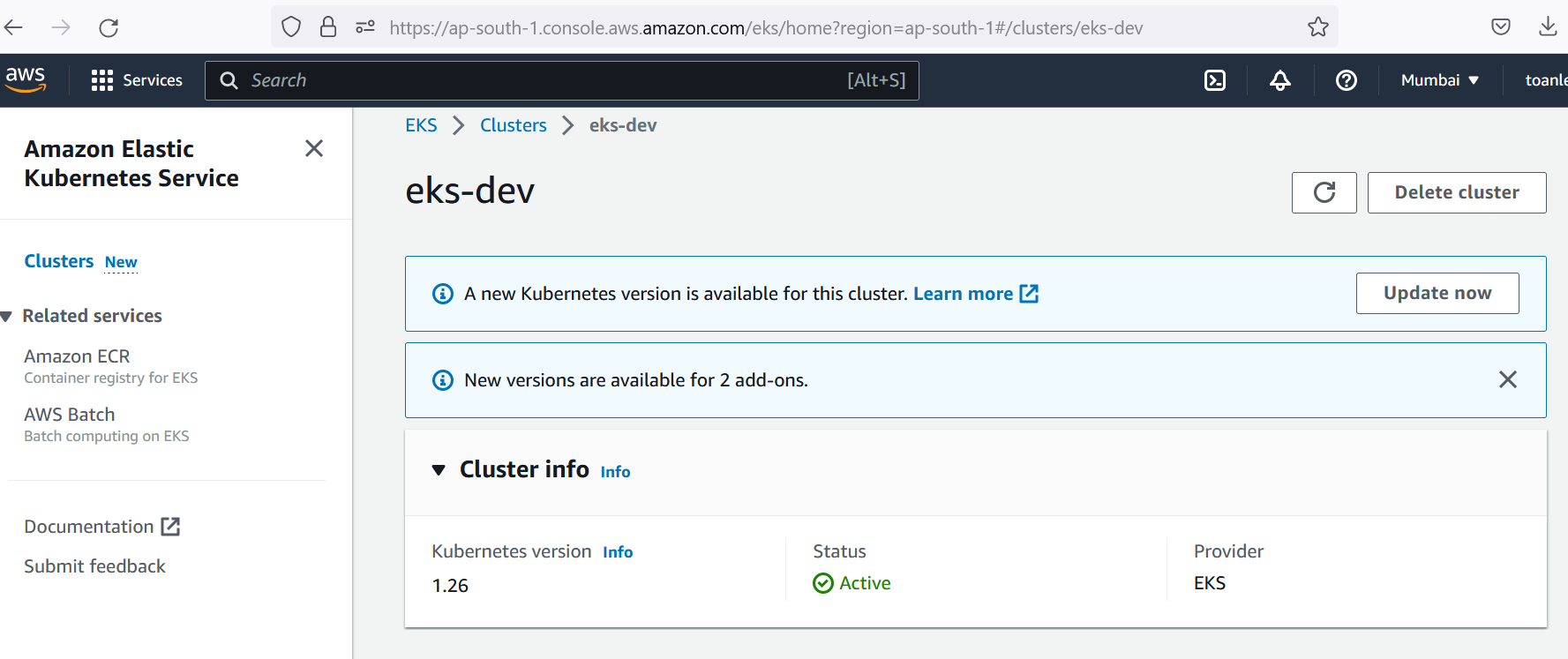
Terraform Plan

Terraform Apply



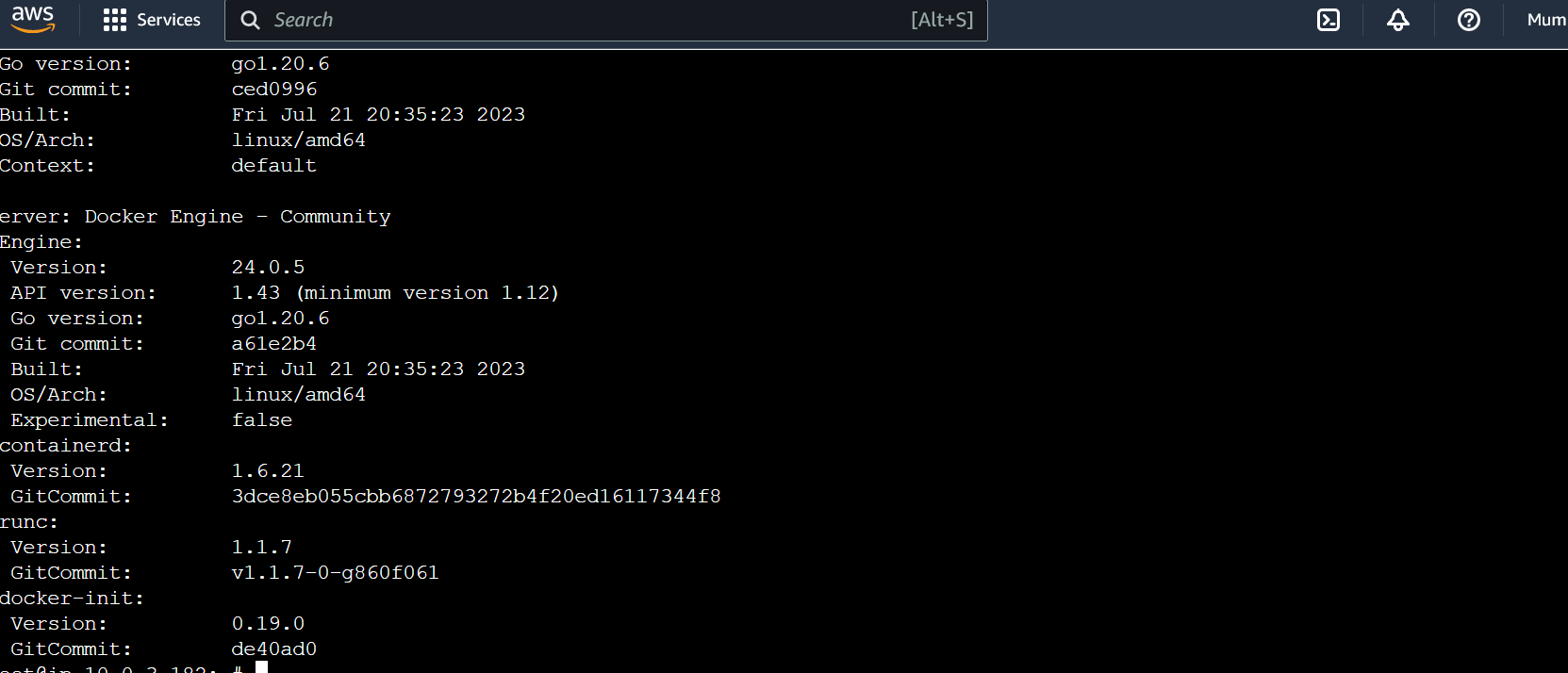


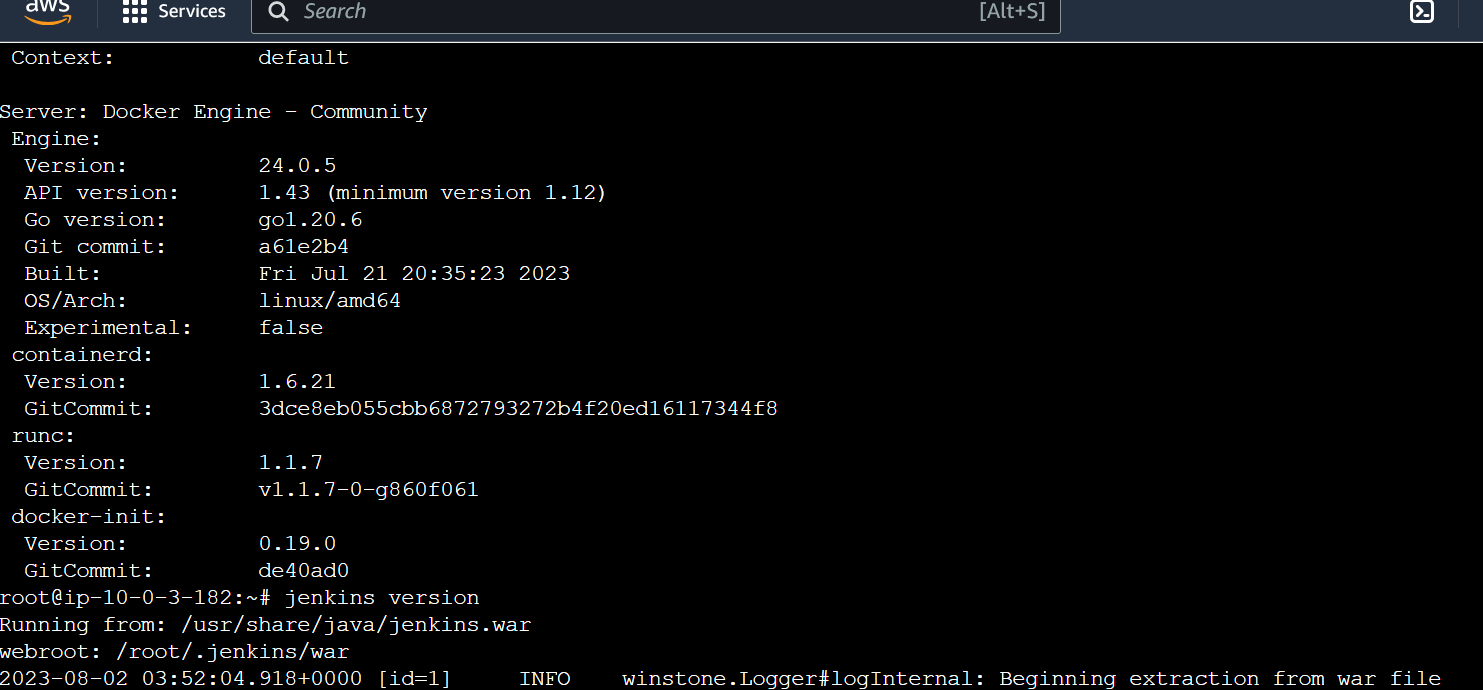


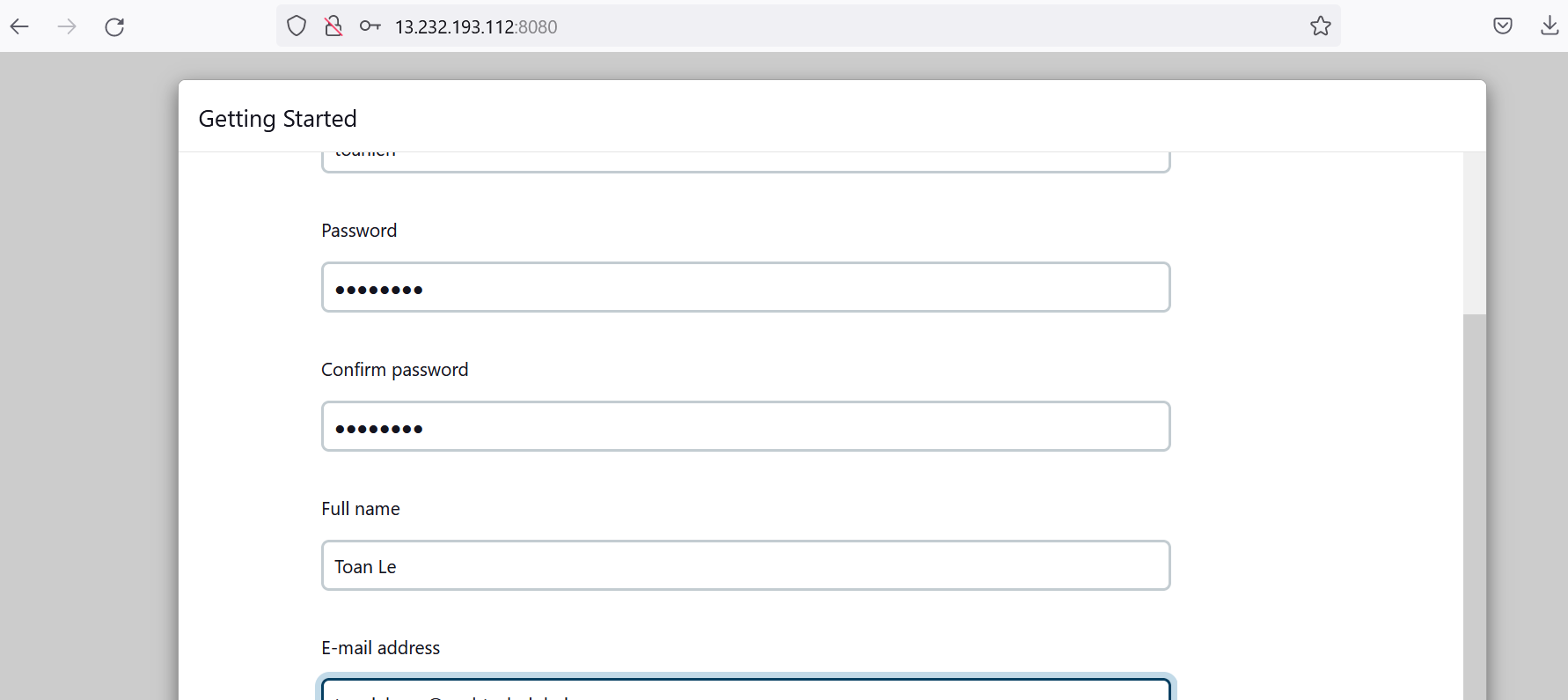


############################# Install Docker and Jenkins ################################

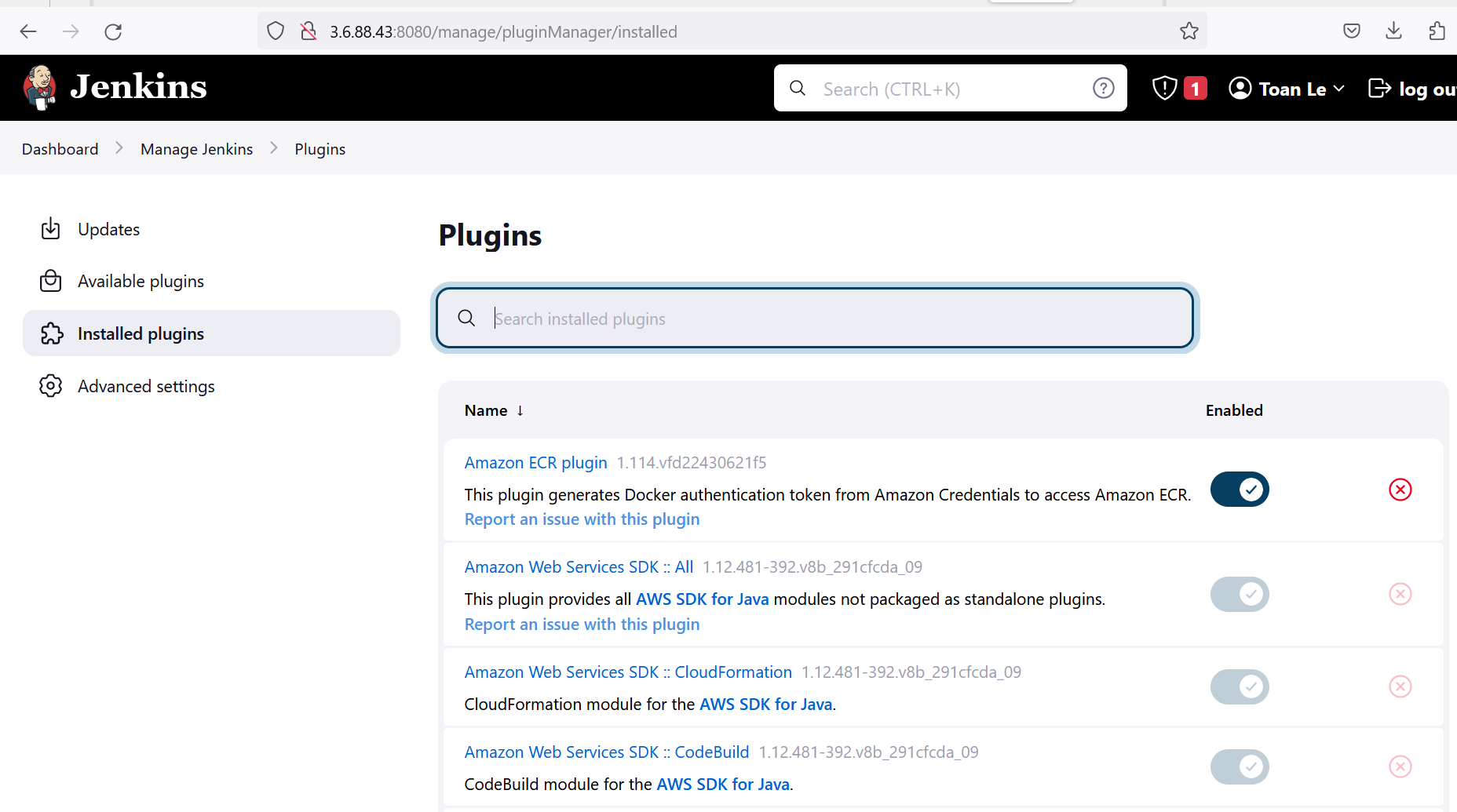
Install Jenkins via EC2 and login to set up Ci/CD pipeline





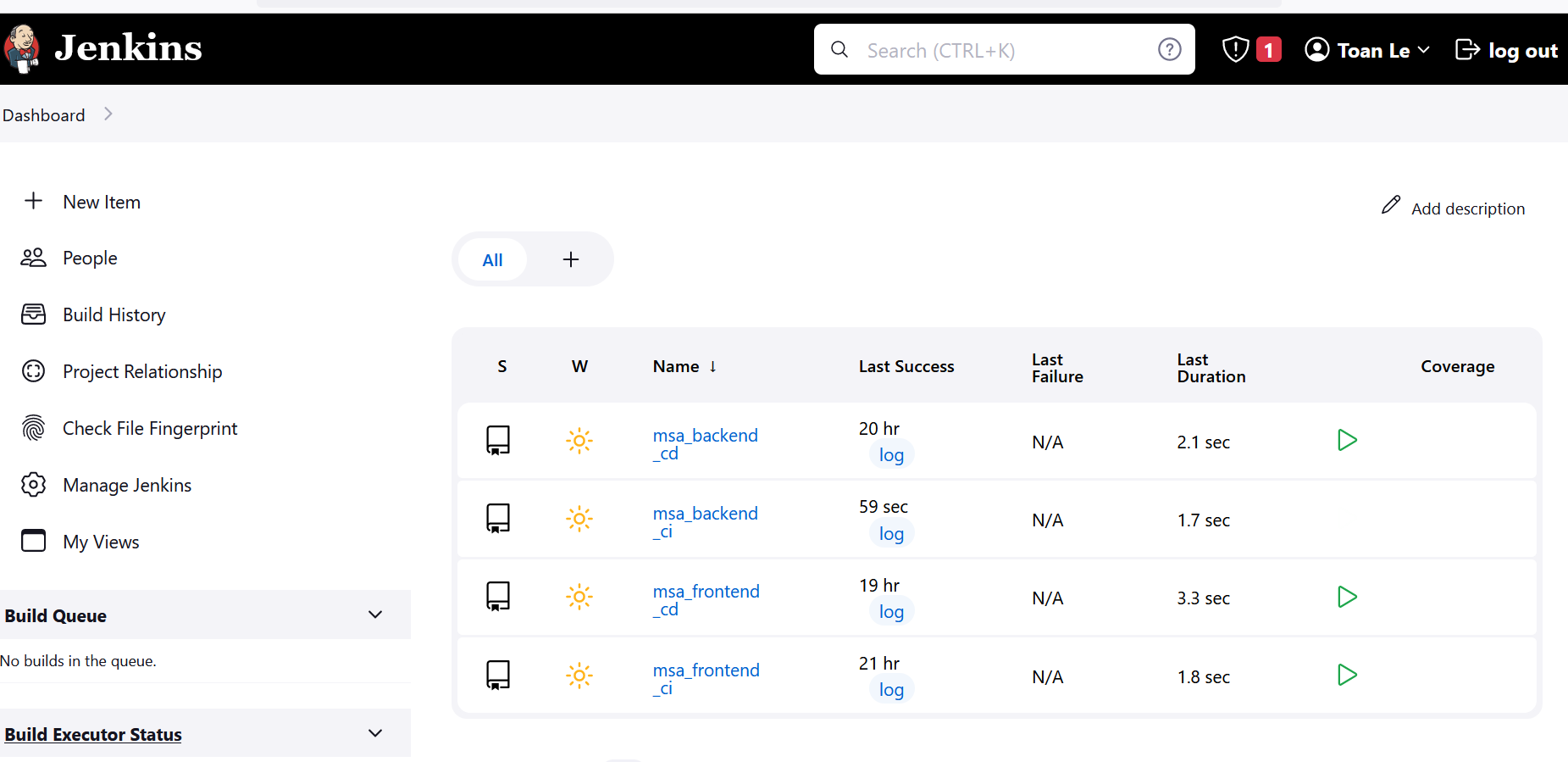


Follow prerequisite from <https://github.com/nashtech-garage/devops-ci-cd> to add require plugins and settings

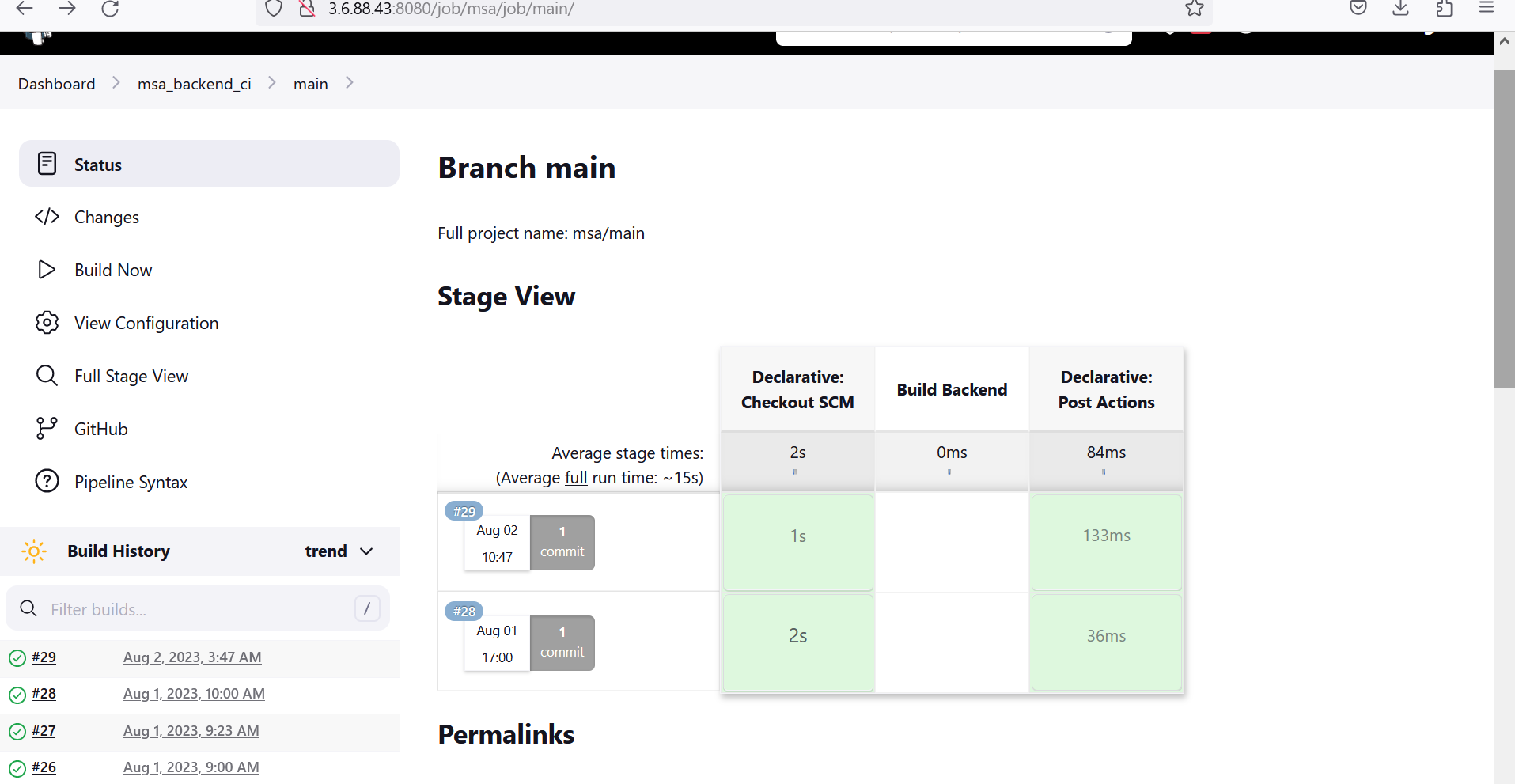


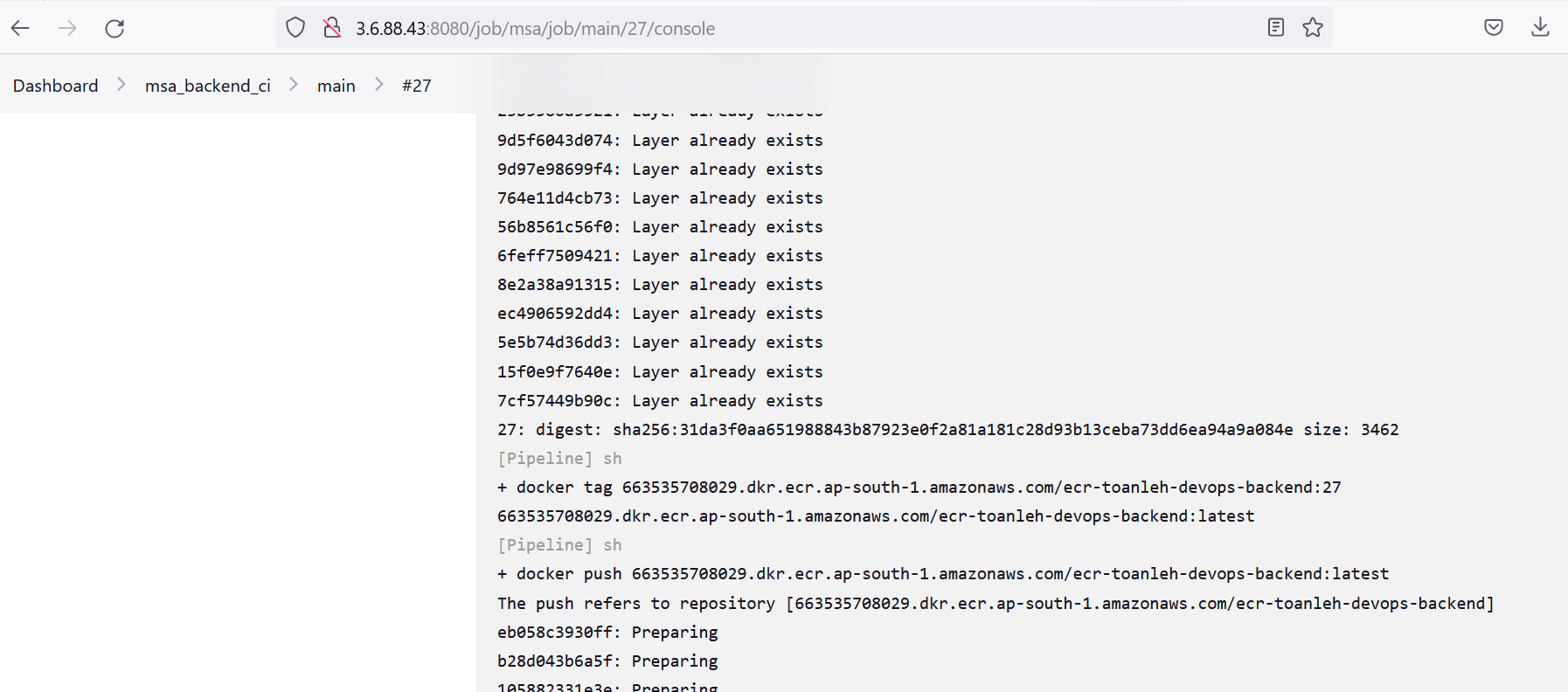
############################# Set up Jenkins CI/CD pipeline ############################

#set up CI/CD for backend and frontend

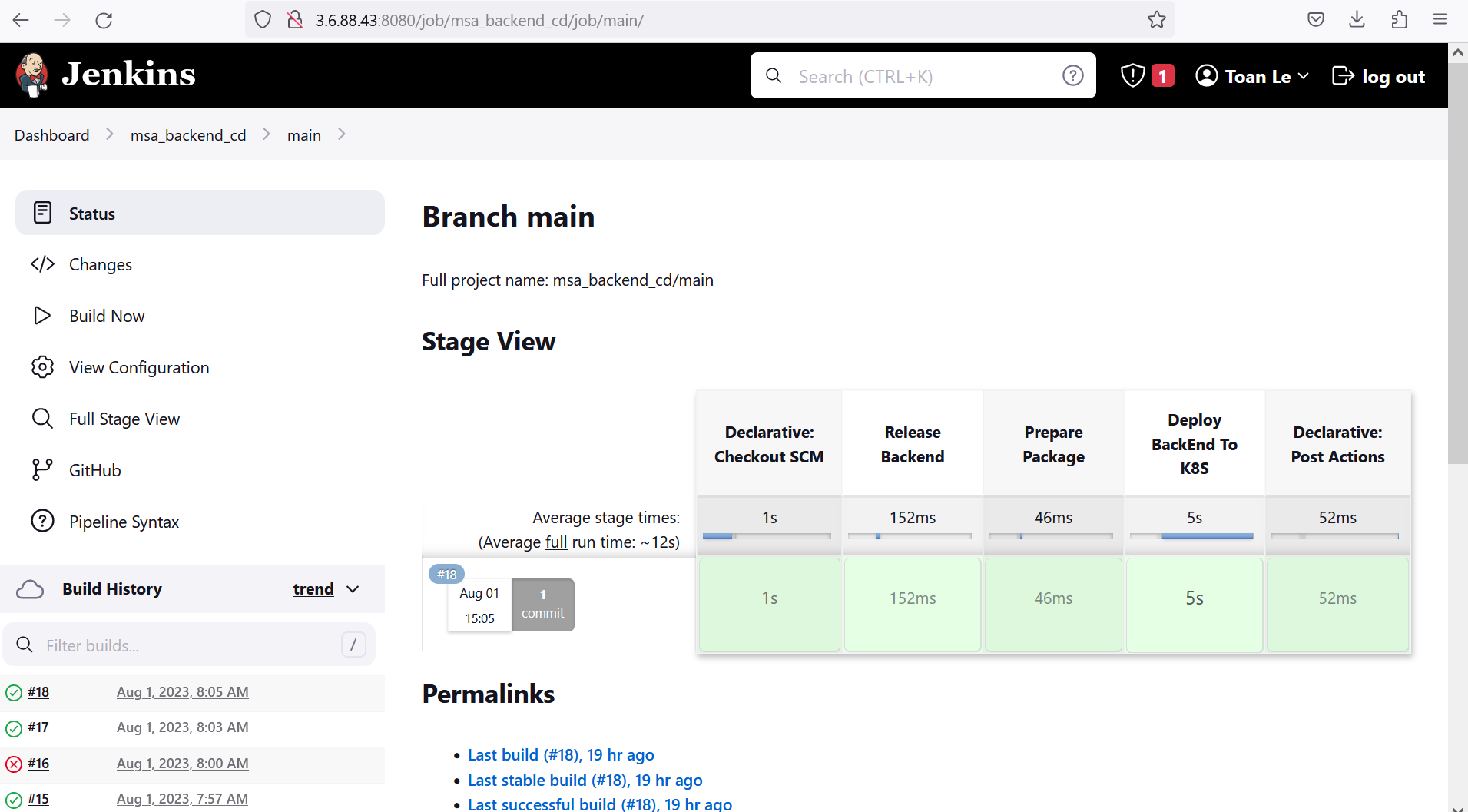


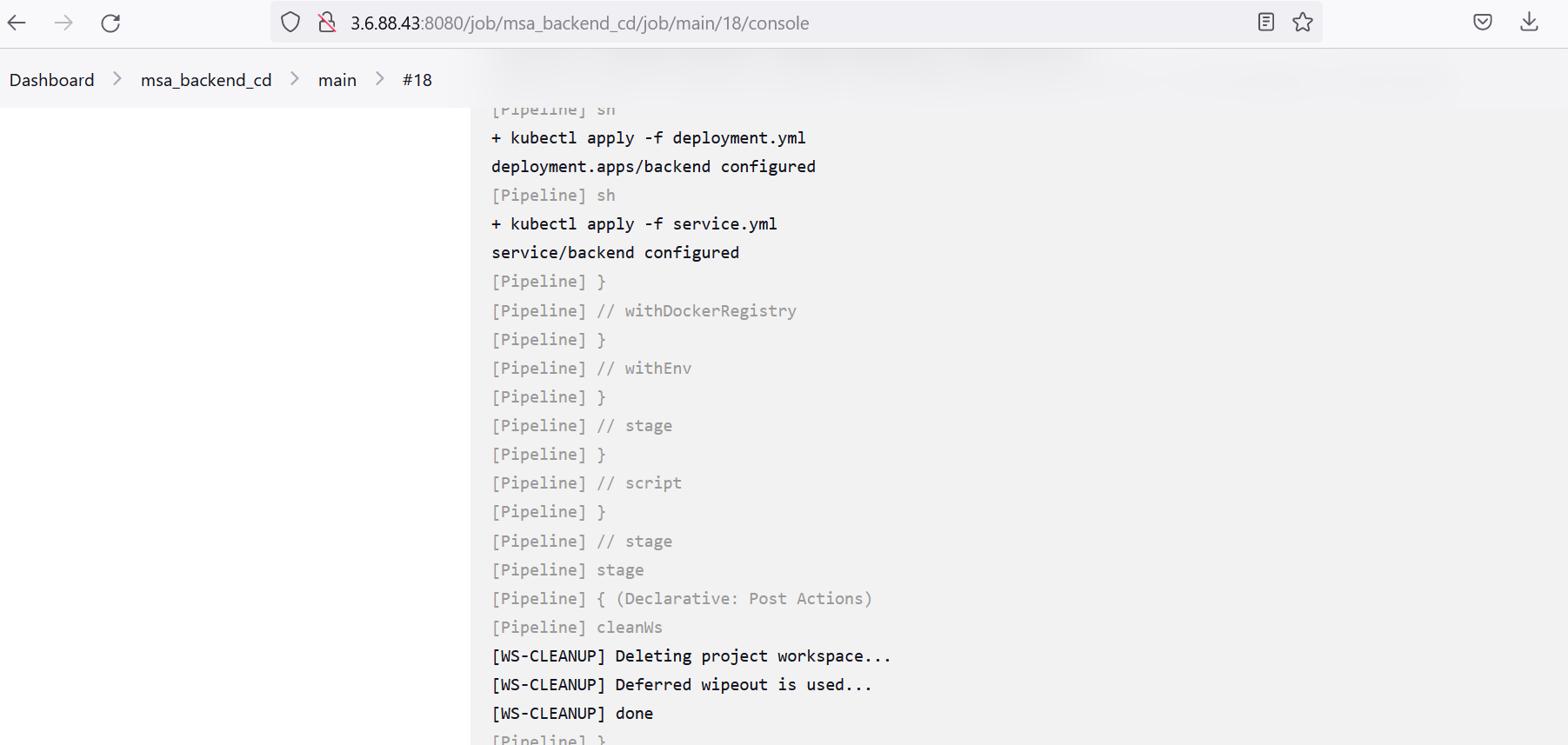
#backend ci



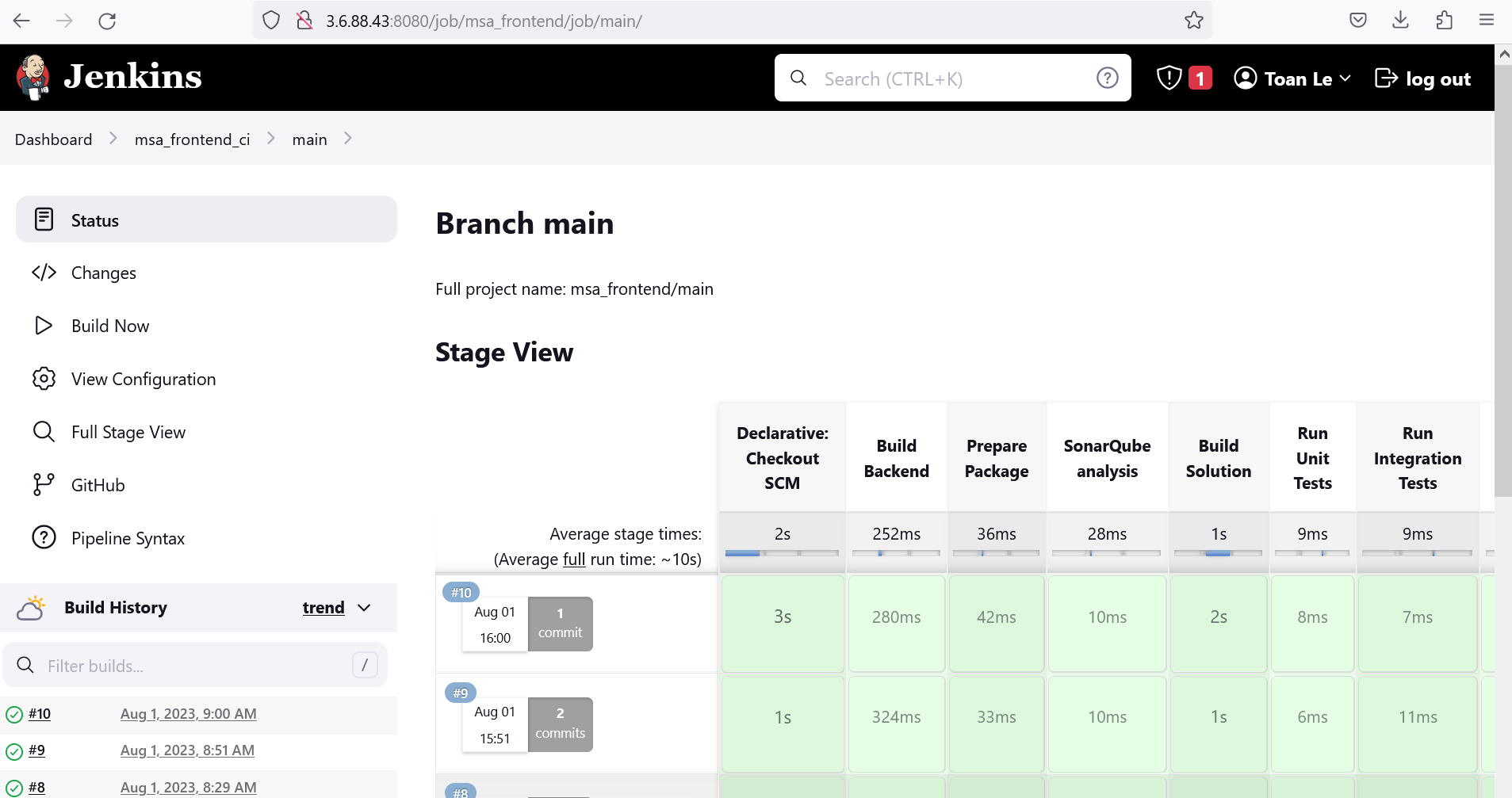


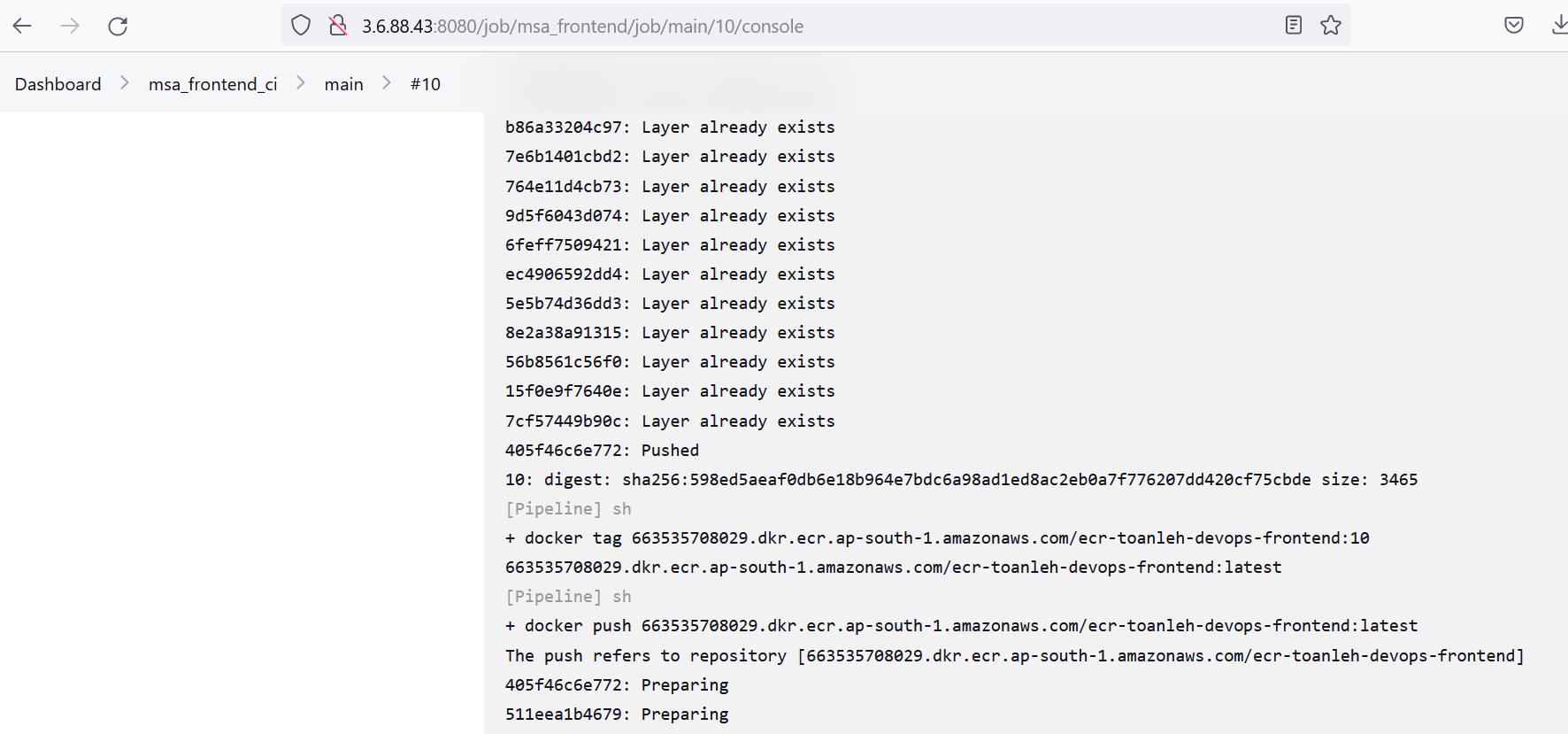
#Backend CD



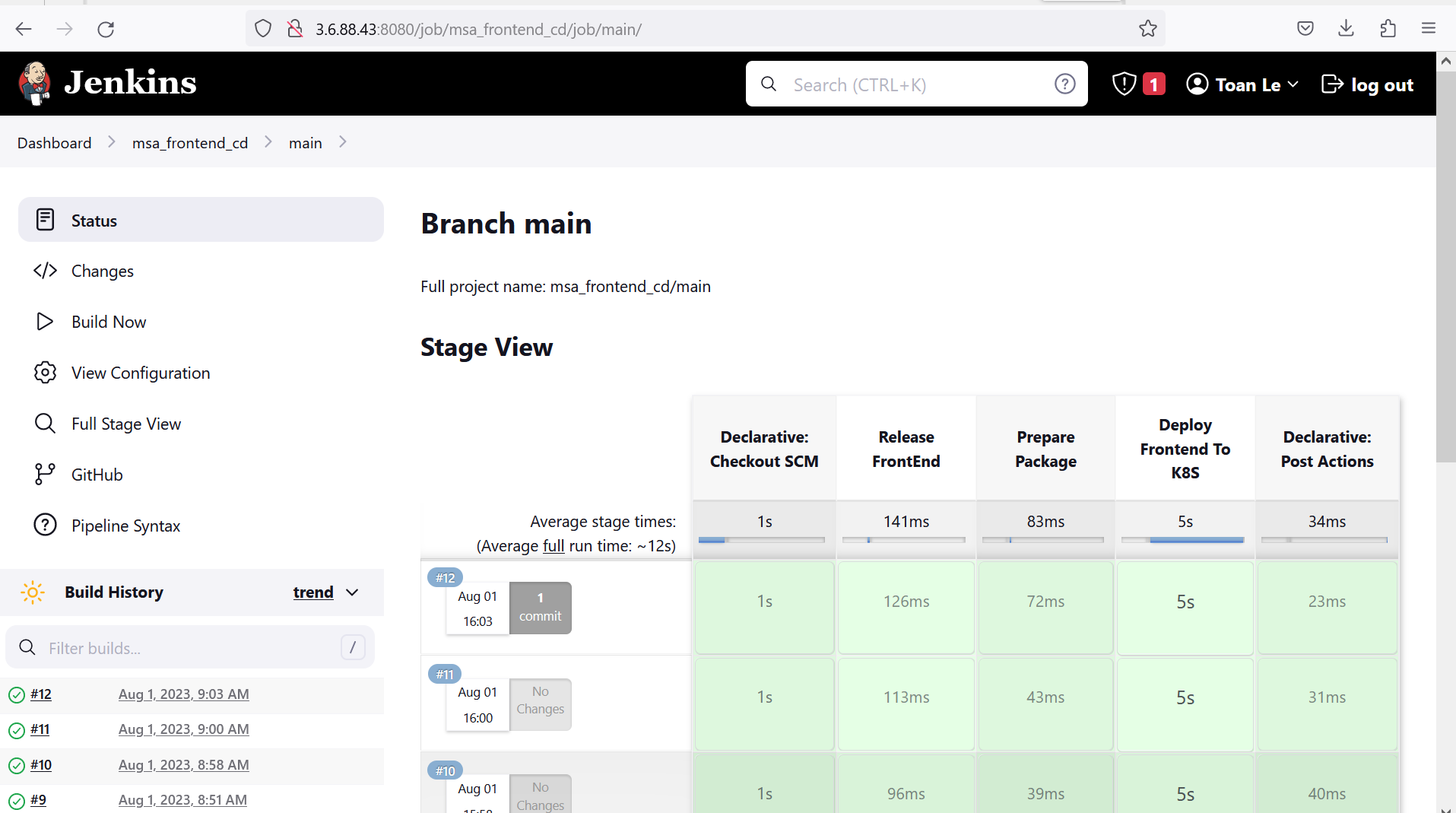


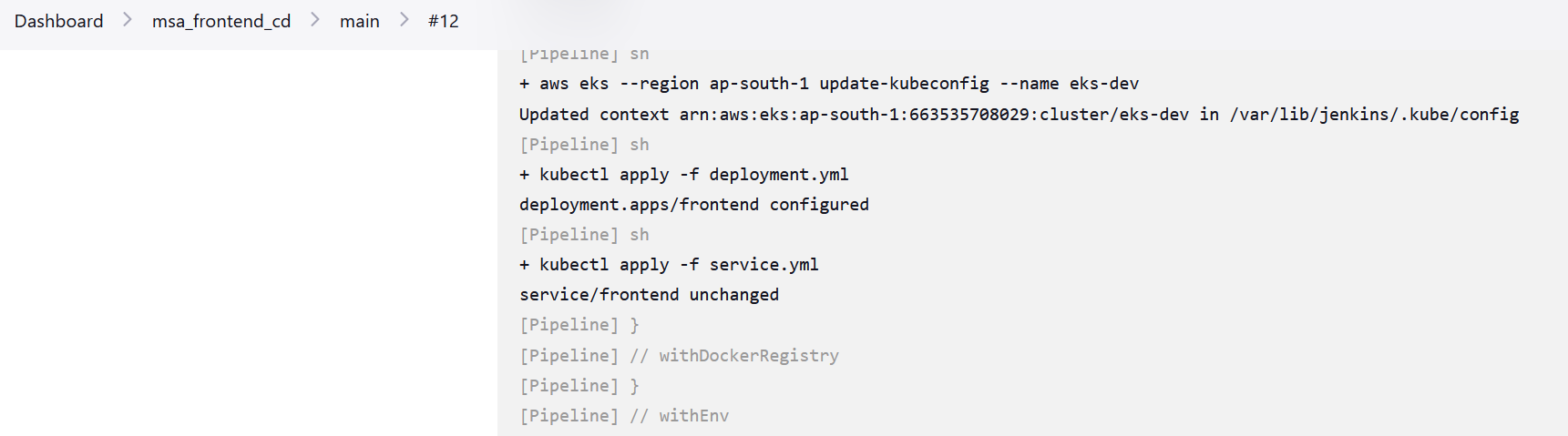
#FrontEnd CI



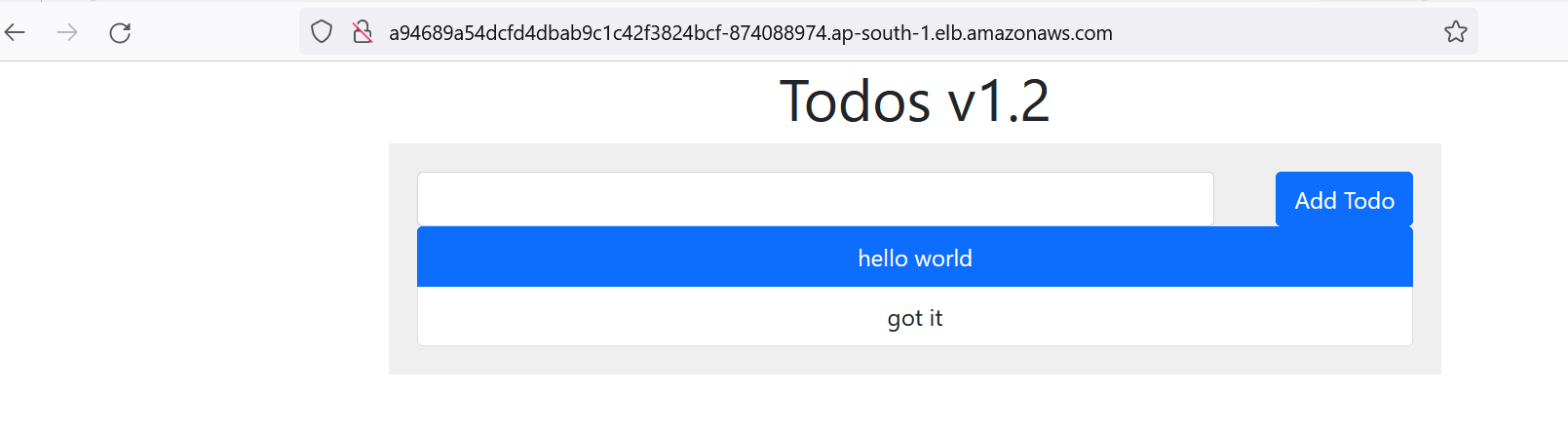


#FrontEnd CD





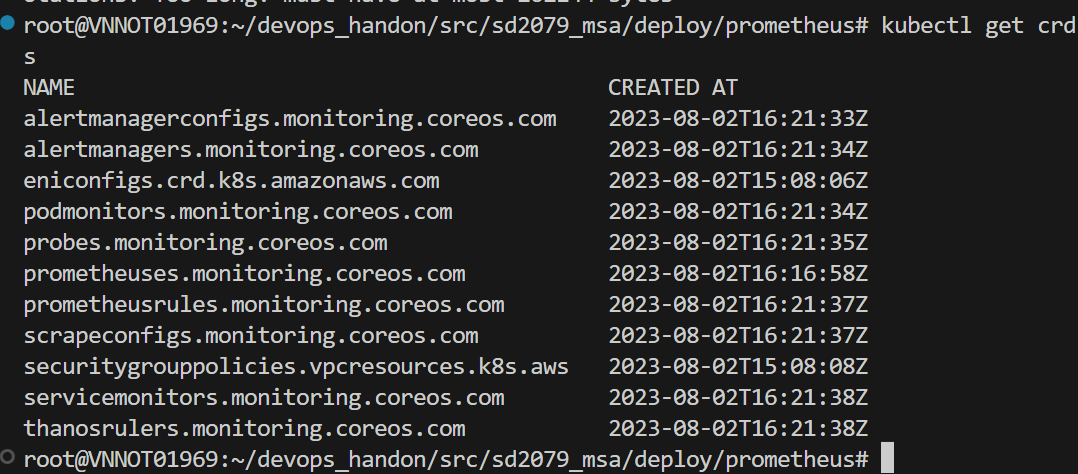
#Final Result



############################# Monitoring ############################

#Install CRD

kubectl apply -f <https://raw.githubusercontent.com/prometheus-operator/prometheus-operator/master/bundle.yaml>



#Install Prometheus

kubectl create namespace prometheus

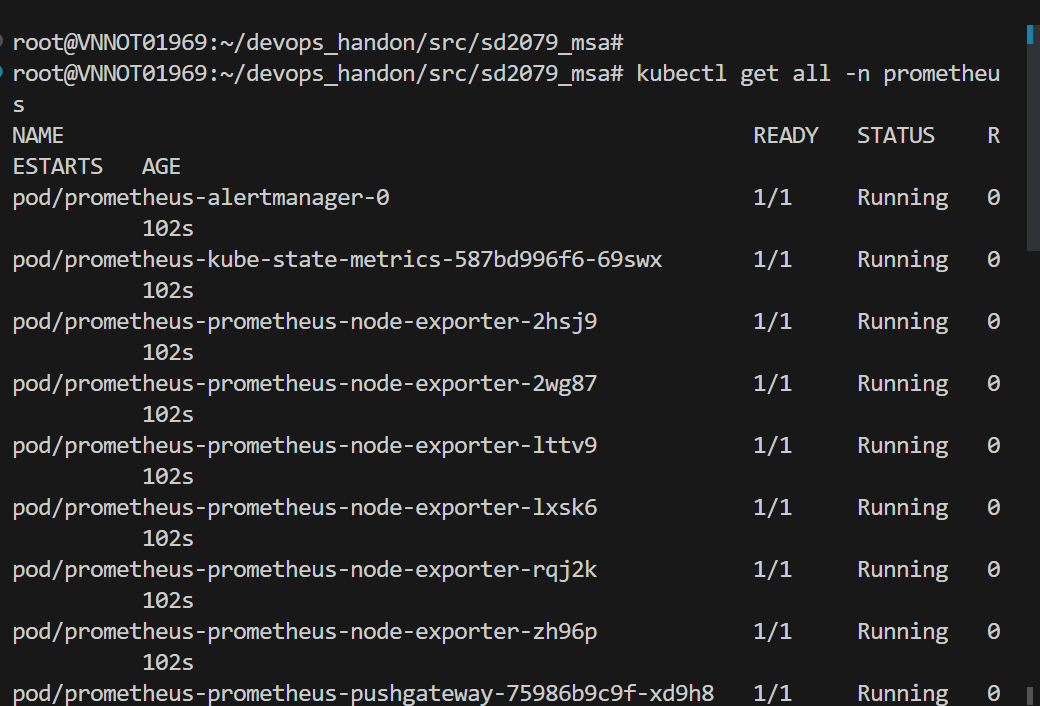
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts

helm install prometheus prometheus-community/prometheus \

--namespace prometheus \

--set alertmanager.persistentVolume.storageClass="gp2" \

--set server.persistentVolume.storageClass="gp2"



#Install Grafana

kubectl create namespace grafana

helm install grafana grafana/grafana \

--namespace grafana \

--set persistence.storageClassName="gp2" \

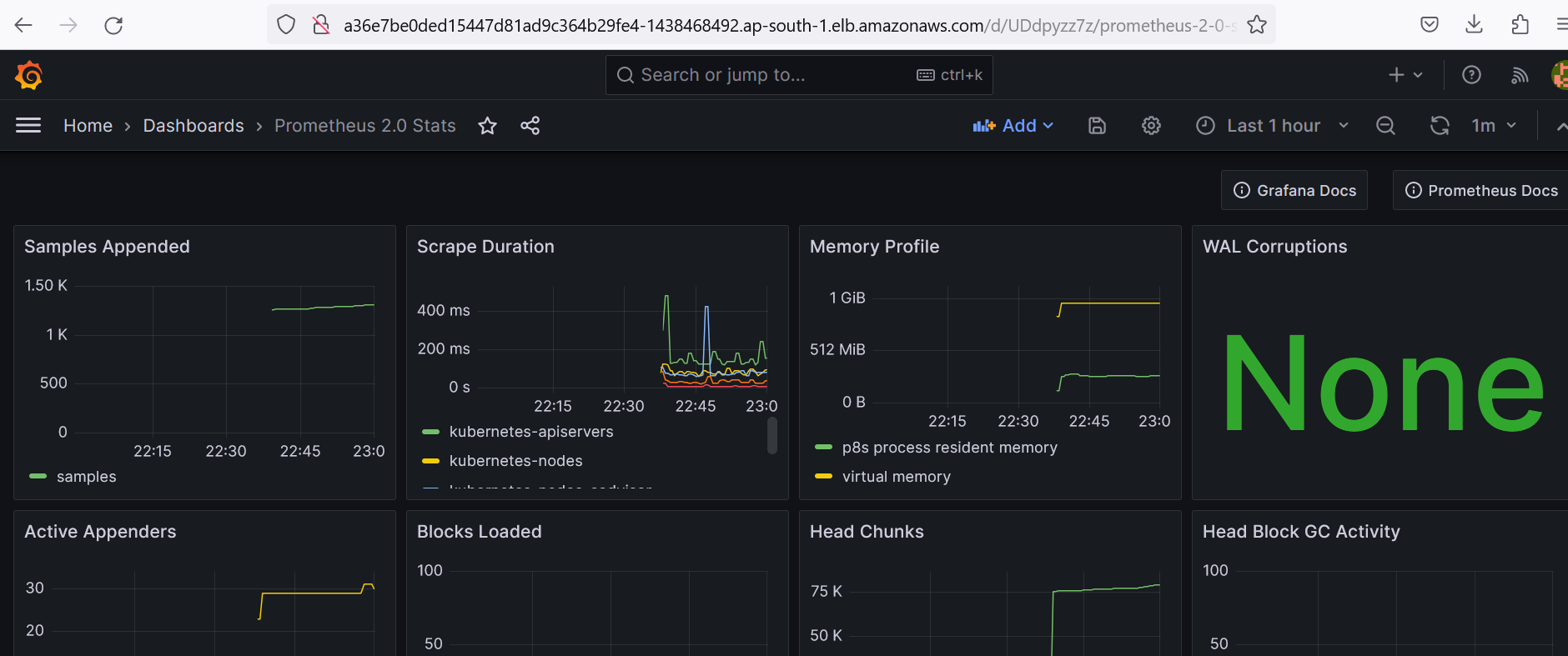
--set persistence.enabled=true \

--set adminPassword='Pa55w0rd' \

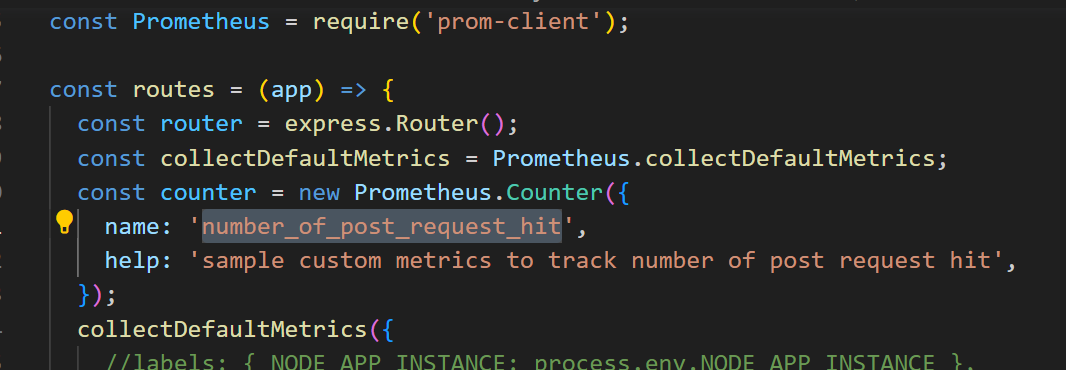
--values ./deploy/grafana/grafana.yaml \

--set service.type=LoadBalancer

Ensure you can access the Grafana dashboard after install



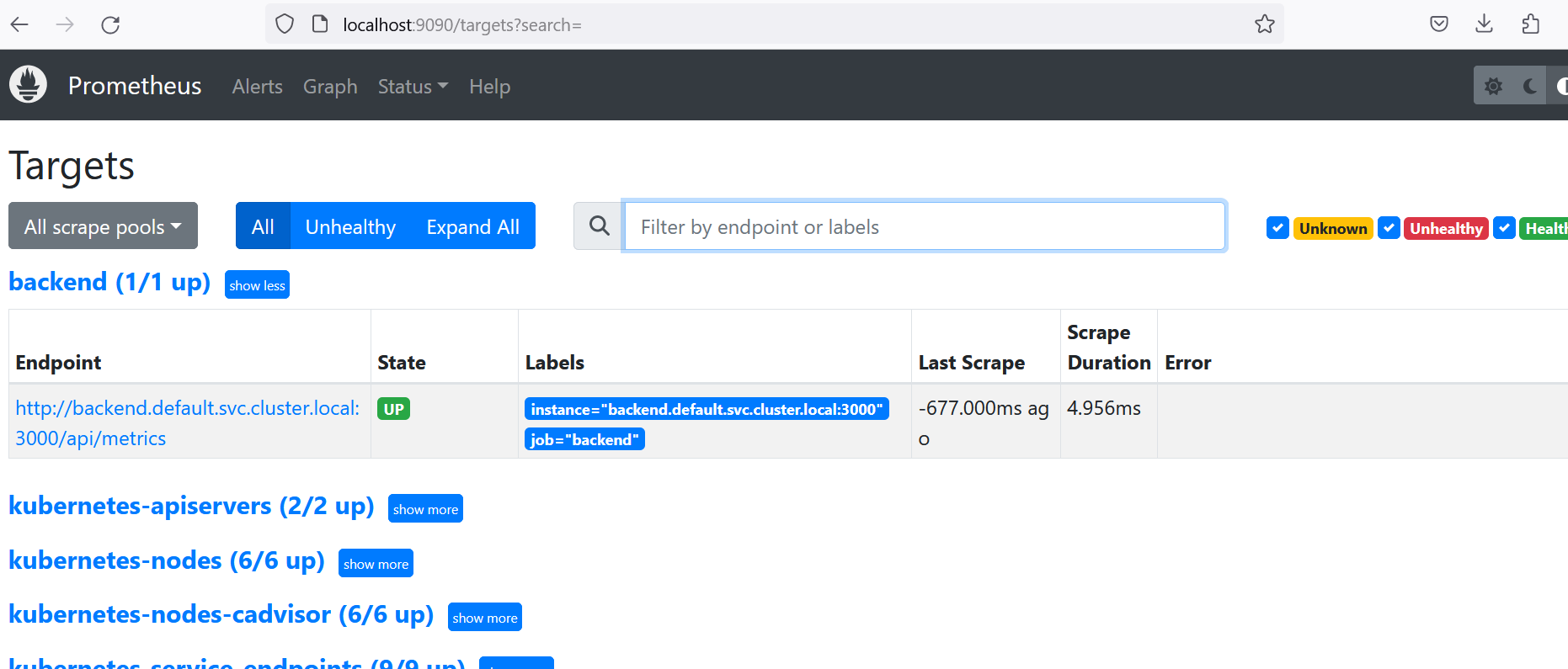
# Add the custom metrics to backend, Check and ensure the custom metrics is on frontend





#Instruct Prometheus to get the custom metrics from backend

helm upgrade prometheus prometheus-community/prometheus --install -f values.yaml -n prometheus --set alertmanager.persistentVolume.storageClass="gp2" --set server.persistentVolume.storageClass="gp2"



#Check and ensure the custom metrics can be accessed via Grafana dashboard (prometheus datasource), then you can build custom dashboard based on that custom metrics

