**DEMO PROJECT**

**TASK:**

Create Build and Deploy Pipeline for K8s application deployment.

Create Job DSL code to create build and deploy pipeline.

Create Free Style Job to Run Job DSL code to create above pipeline jobs.

Build pipeline is image pipeline to build docker file and push image to AWS elastic container registry (ECR).

Deploy pipeline is used to deploy the application using kubectl into k8s cluster.

Outcome is application deployed and run in k8s cluster. Able to access nginx website with url <http://<ip_address>:<port>.>

**Solution:**

**Step 1: AWS Login**

* Creating the one new instance
* Installation Jenkins
* Installation of minikube
* Installation of Docker
* Installation AWS Cli

**Step 2: Create Jenkins Job**

* Install plugins Docker ,kubernetes, Job DSL
* Create free style Project
* Click on the DSL script
* write DSL script pipeline on build and deploy
* Build the sample code using git repository

**Step 3: Docker images creation**

* Inside Sample project creating Docker file
* Create ECR repository.
* Build the Docker images.
* Write Code on ECR Credential in pipeline pushing the image on ECR repository **.**
* After pushing image Deploying the application on Kubernetes.

**Step 4: Deploy the Kubernetes Application**

* Write Yaml file on deployment and service
* The minikube application Started deploy the deployment.yaml file .
* Checking Ec2 terminal
* Go to root user sudo -i
* Inside root user Jenkins working
* Checking kubectl get pods
* Checking the SVC

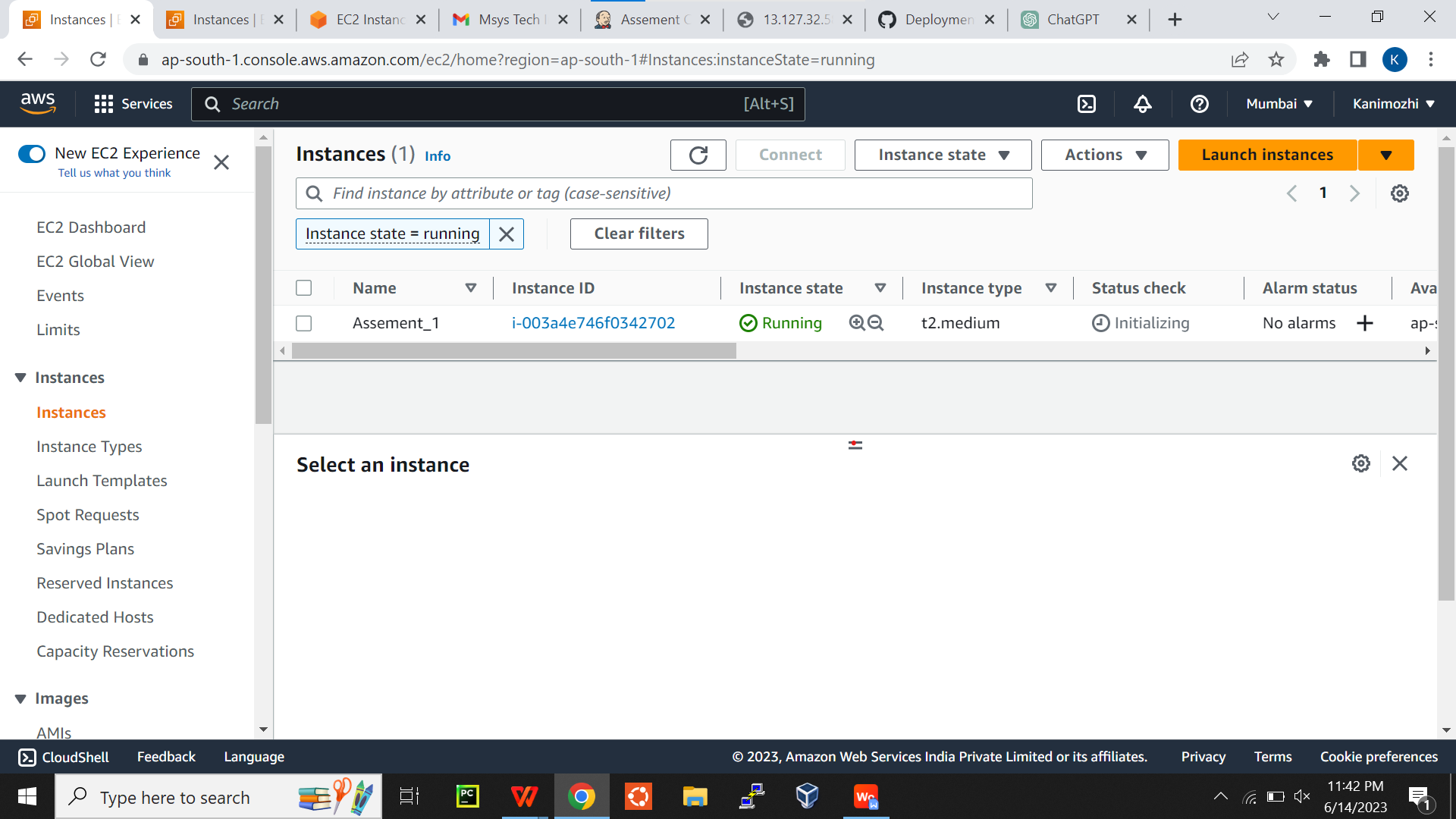
**Step 5: Validate Deployment**

After the "Deploy Pipeline" job completes successfully, verify that the application is deployed and running in the K8s cluster.

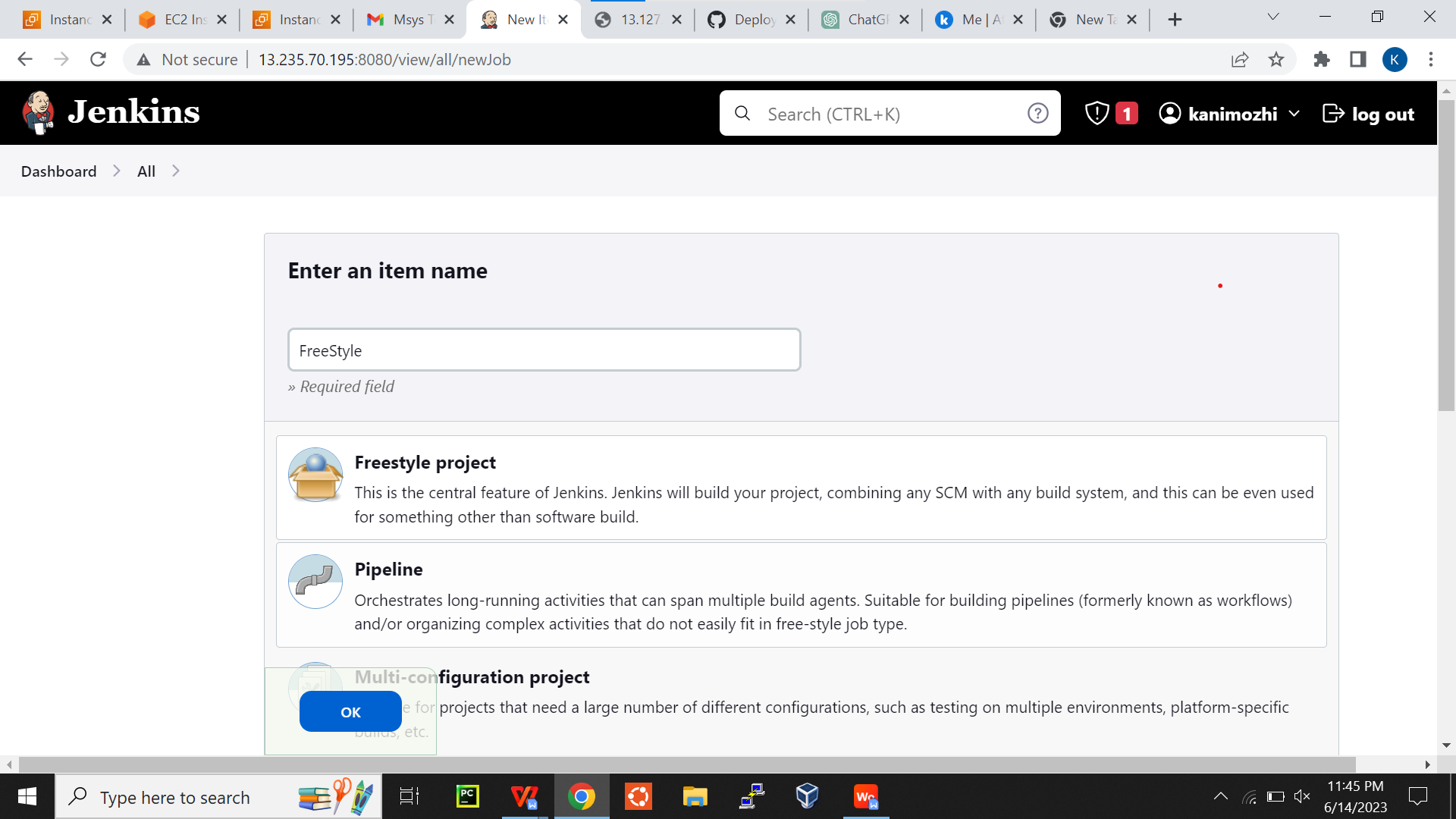
* Obtain the IP address and port of th**e.**

**Screen Shots:**

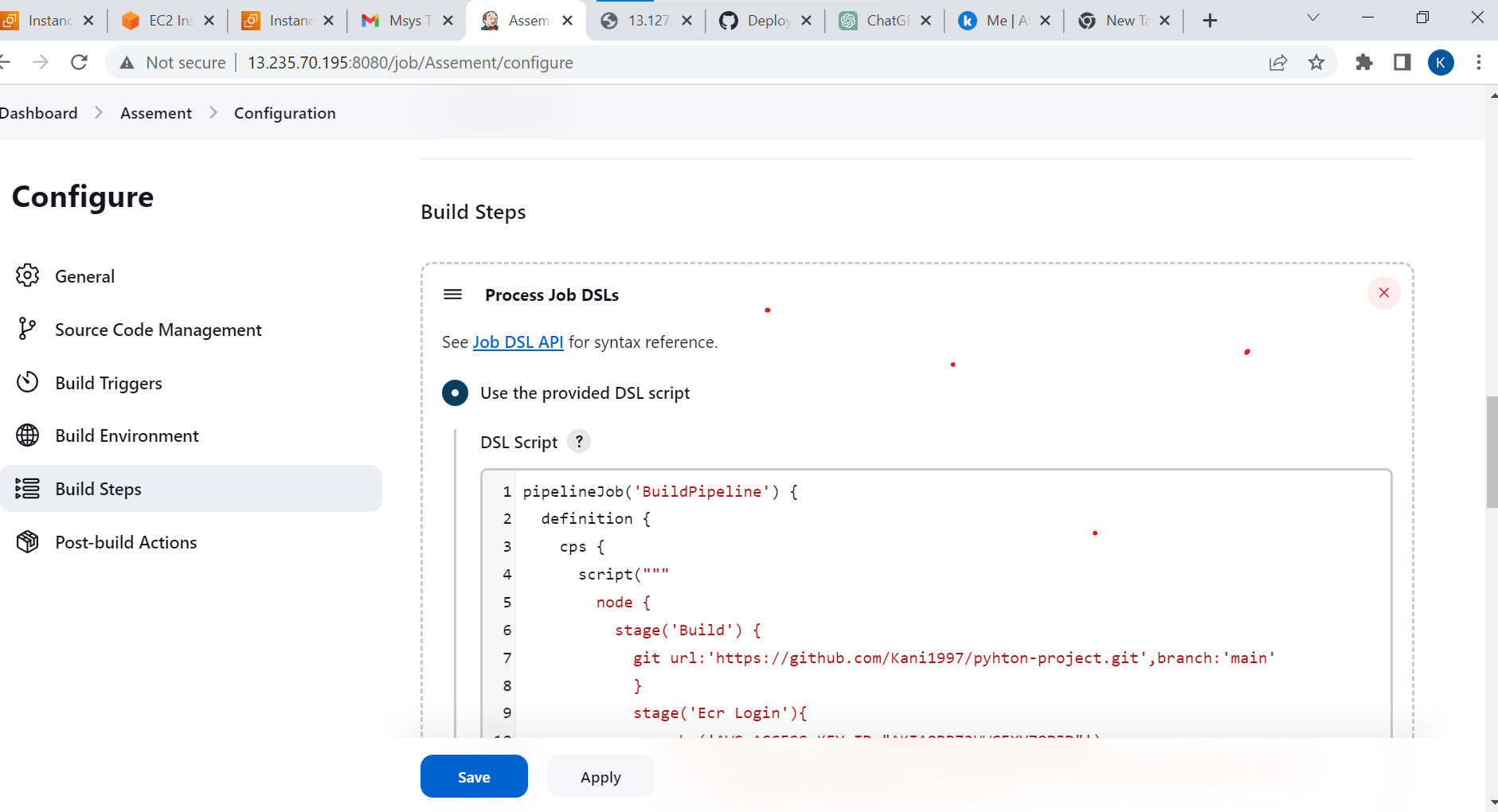
1. **Lanuch Instance**



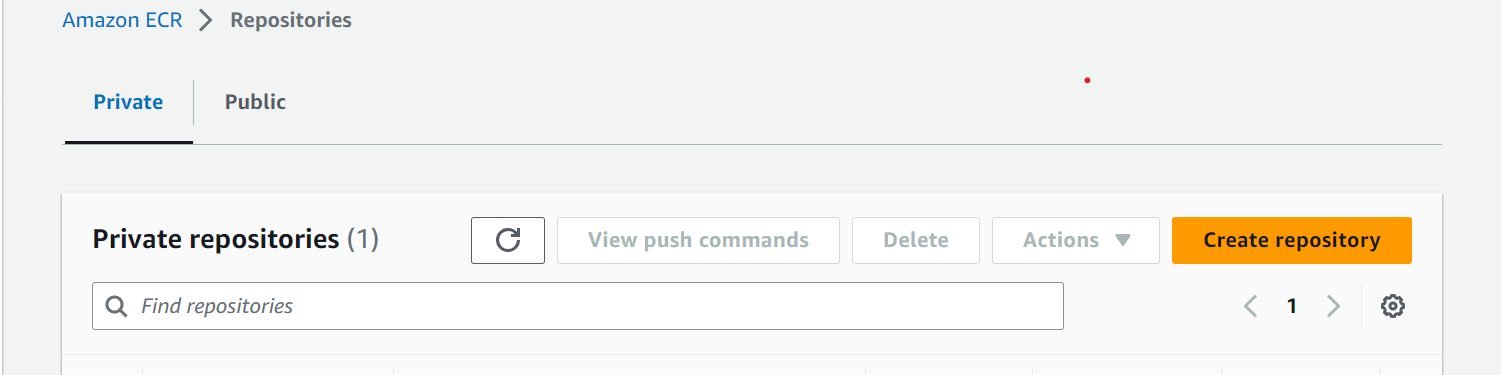
1. **Create Freestyle Project**



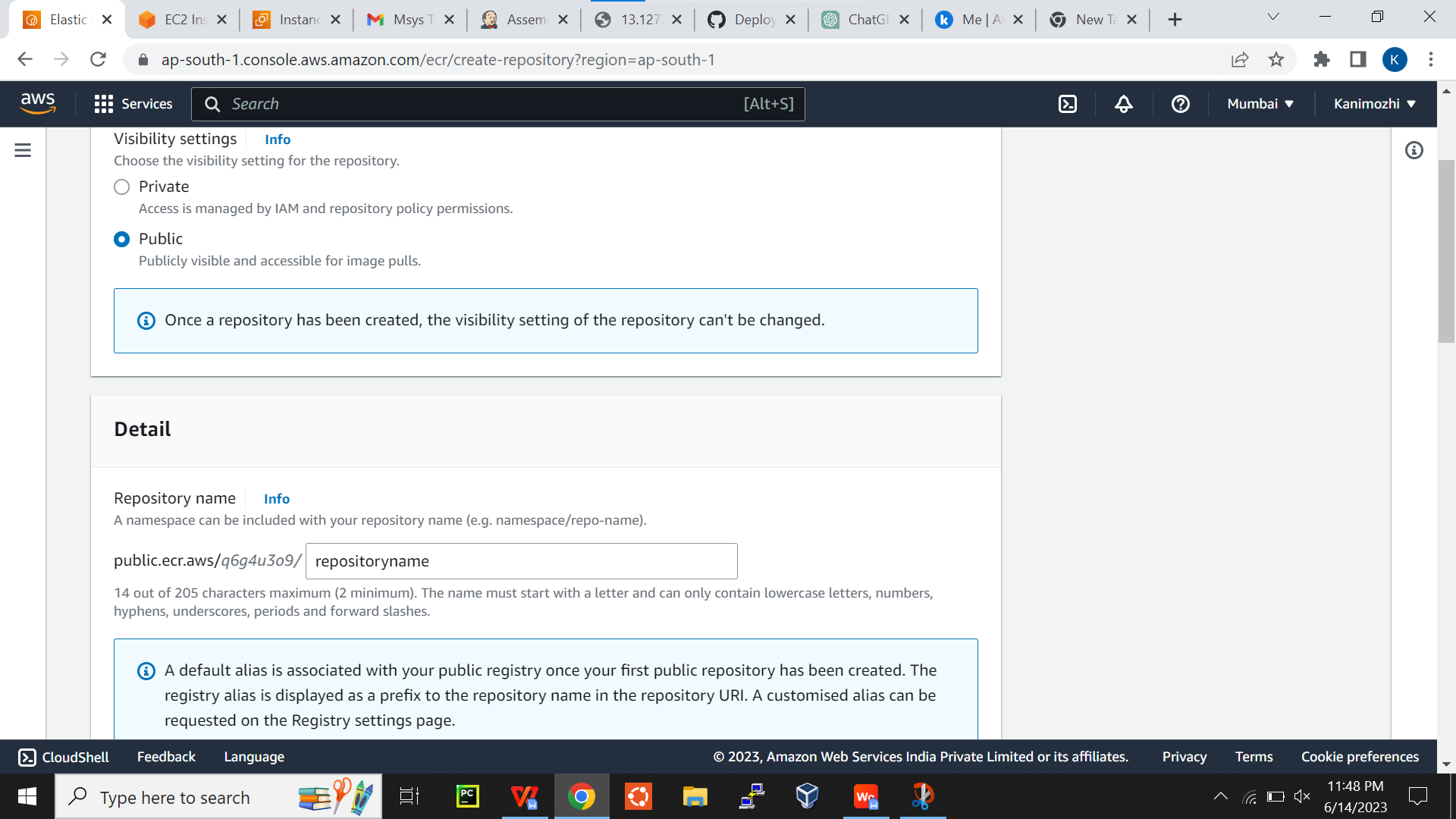
1. **Create DSL script**



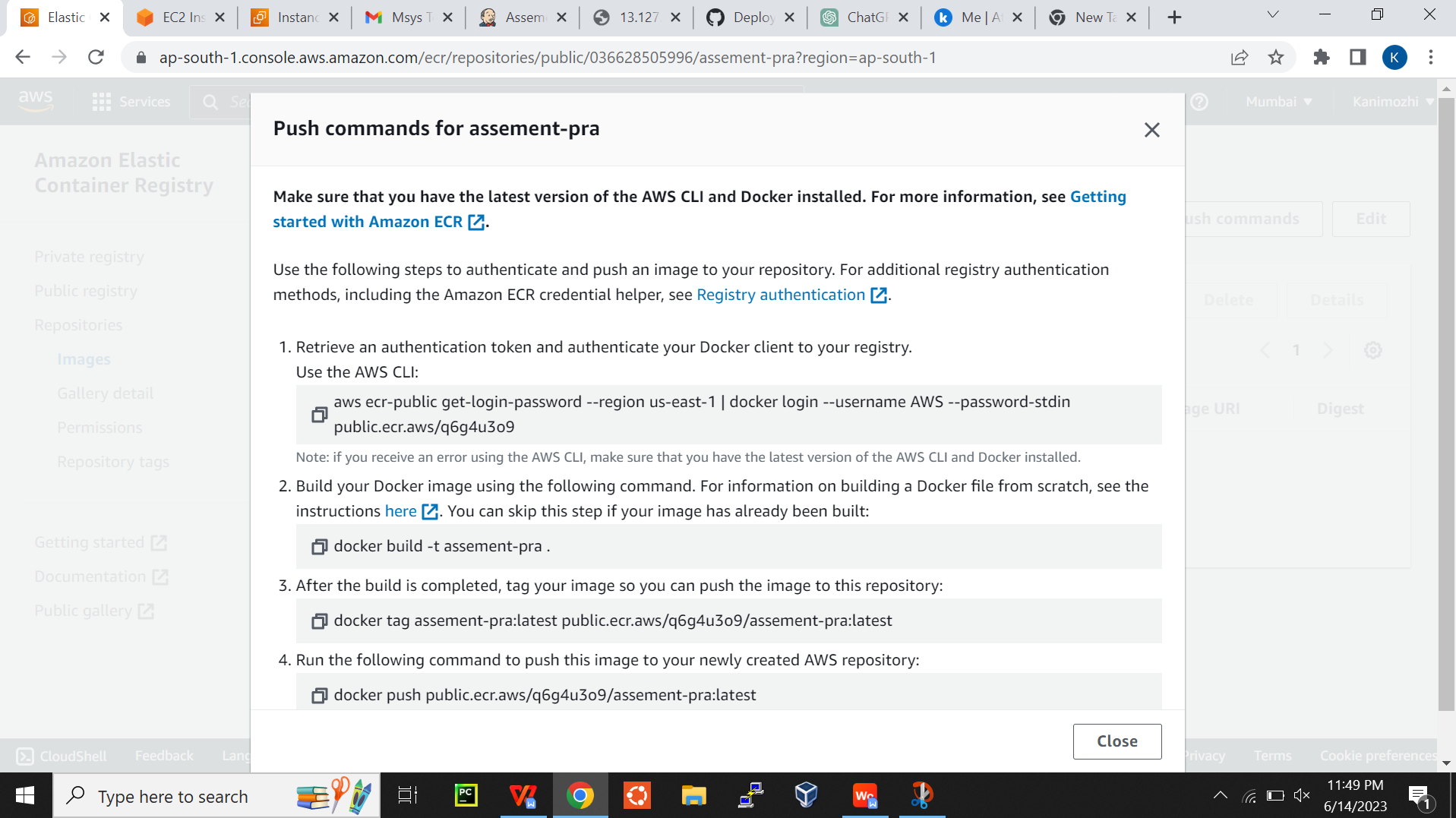
**4.Create ECR Repository**



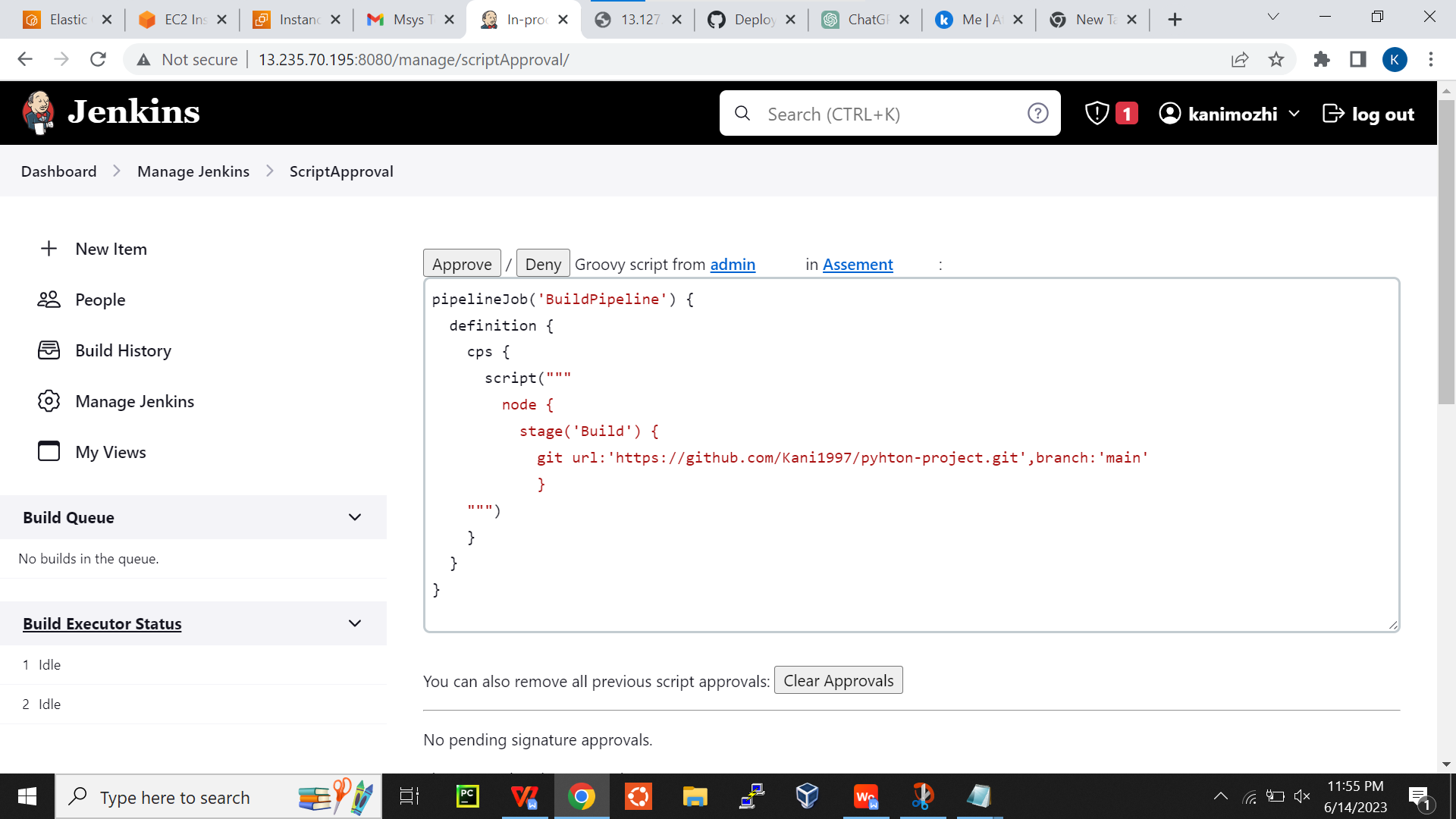
**Create Public Repository**



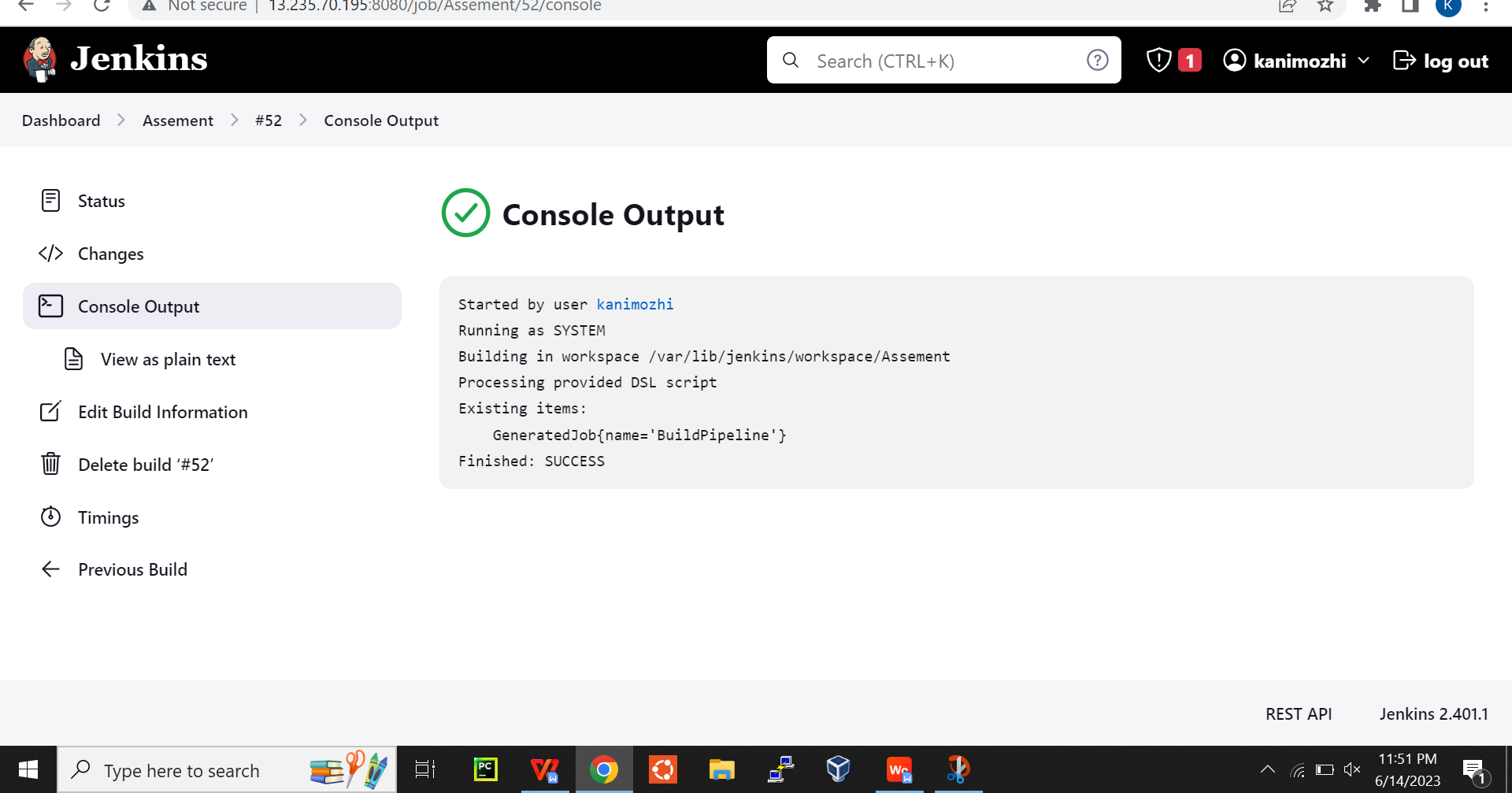
**5. Viewing the Pushing Commands**

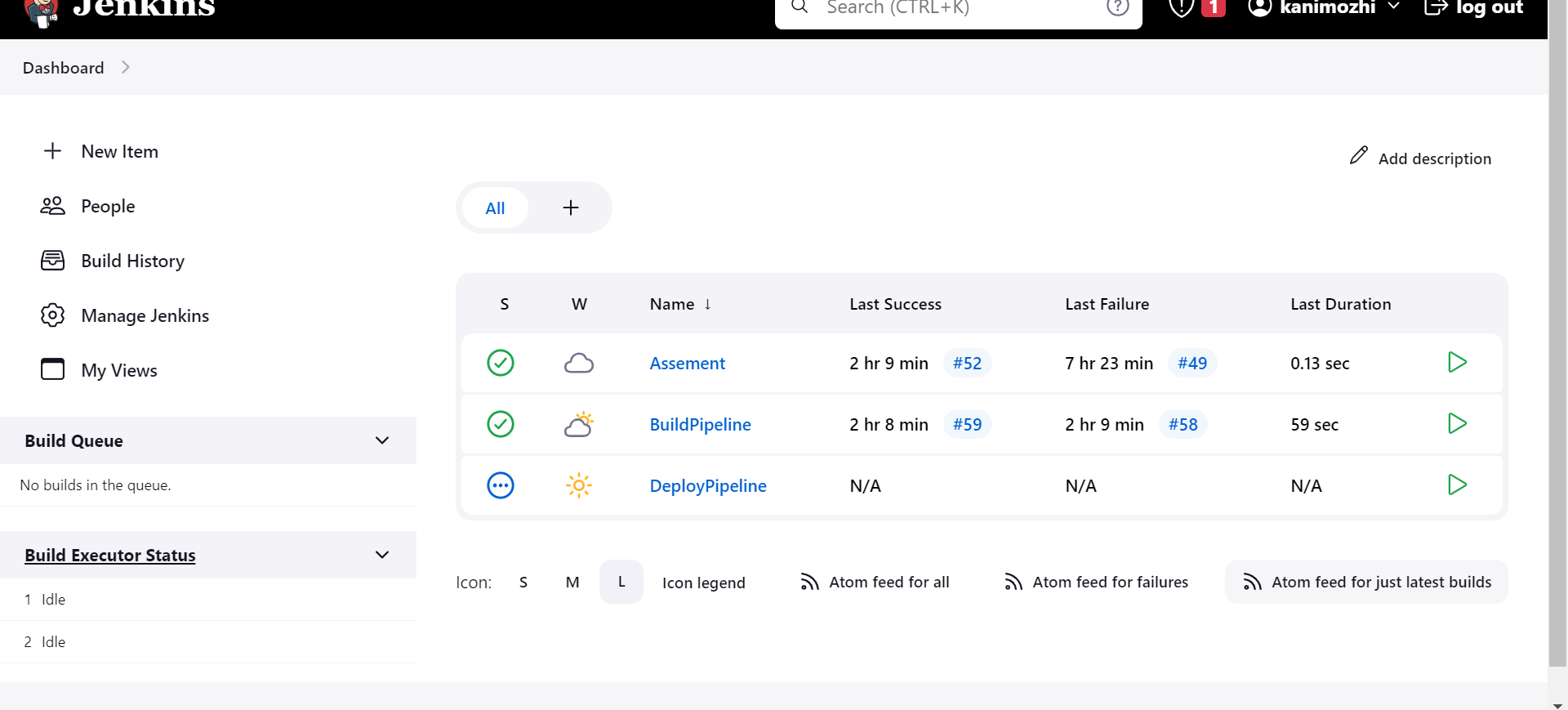


1. **Write DSLPipeline Build pipeline**

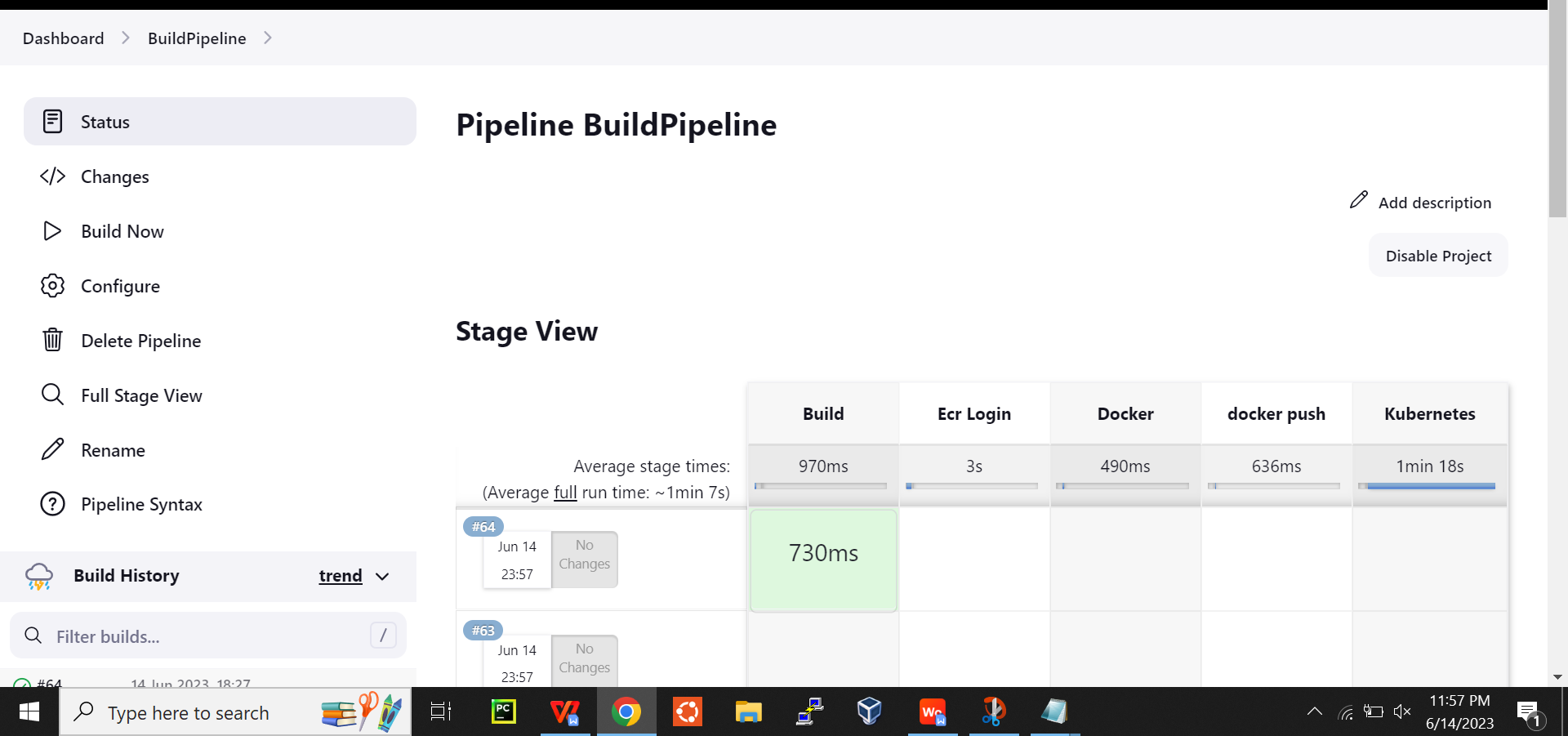


**7.Build pipeline successfully**

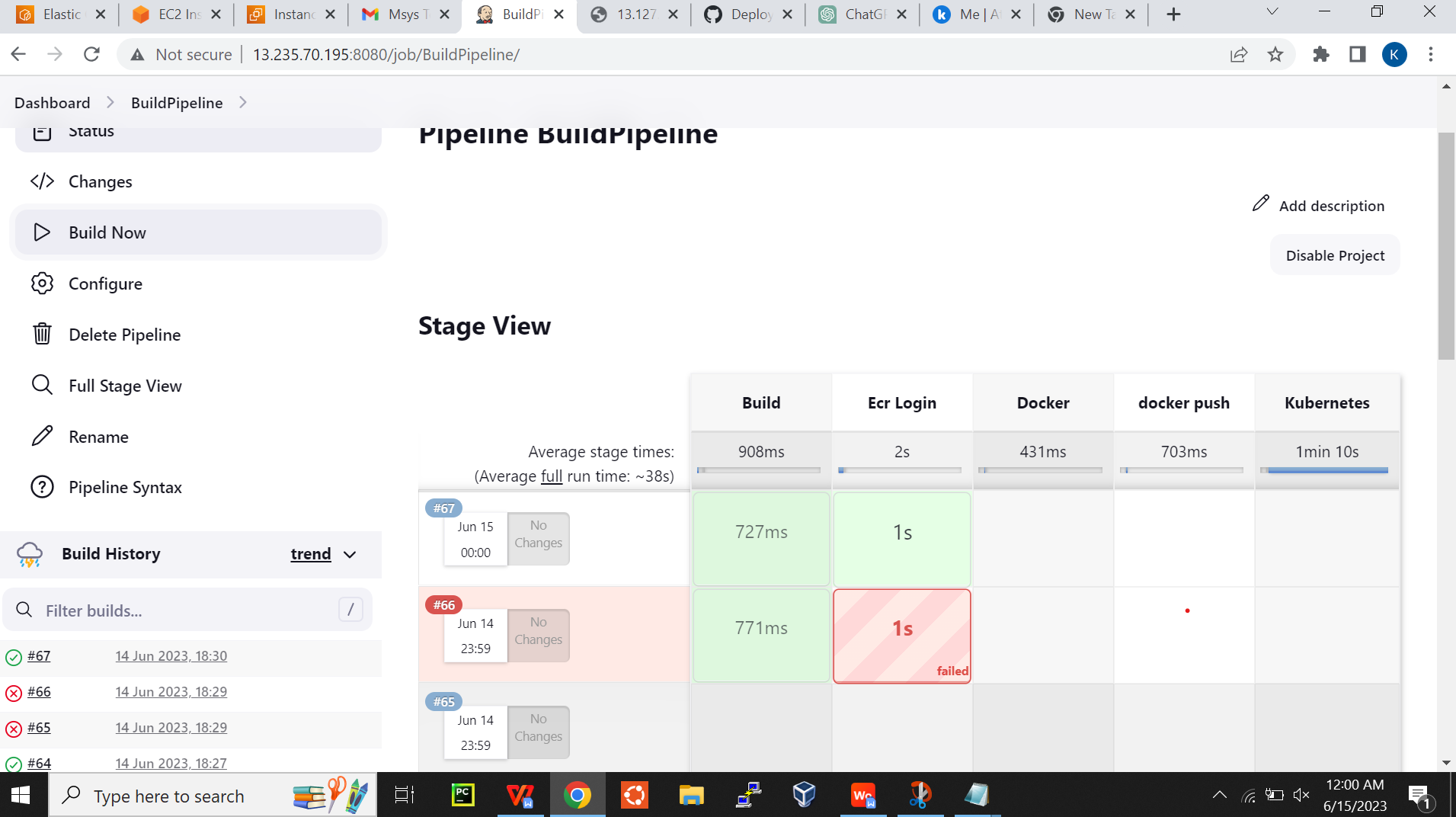




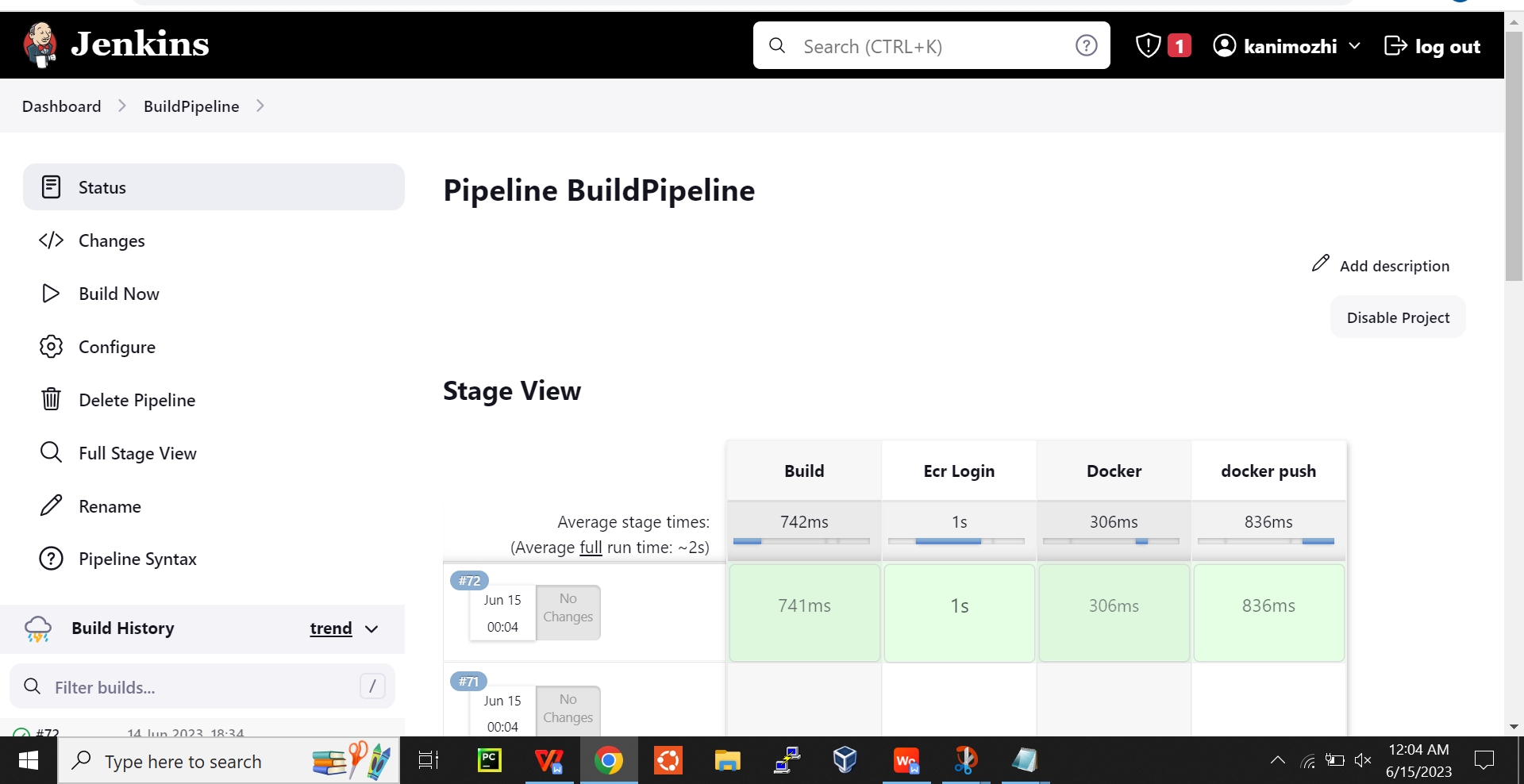
1. **Viewing stage of build pipeline**



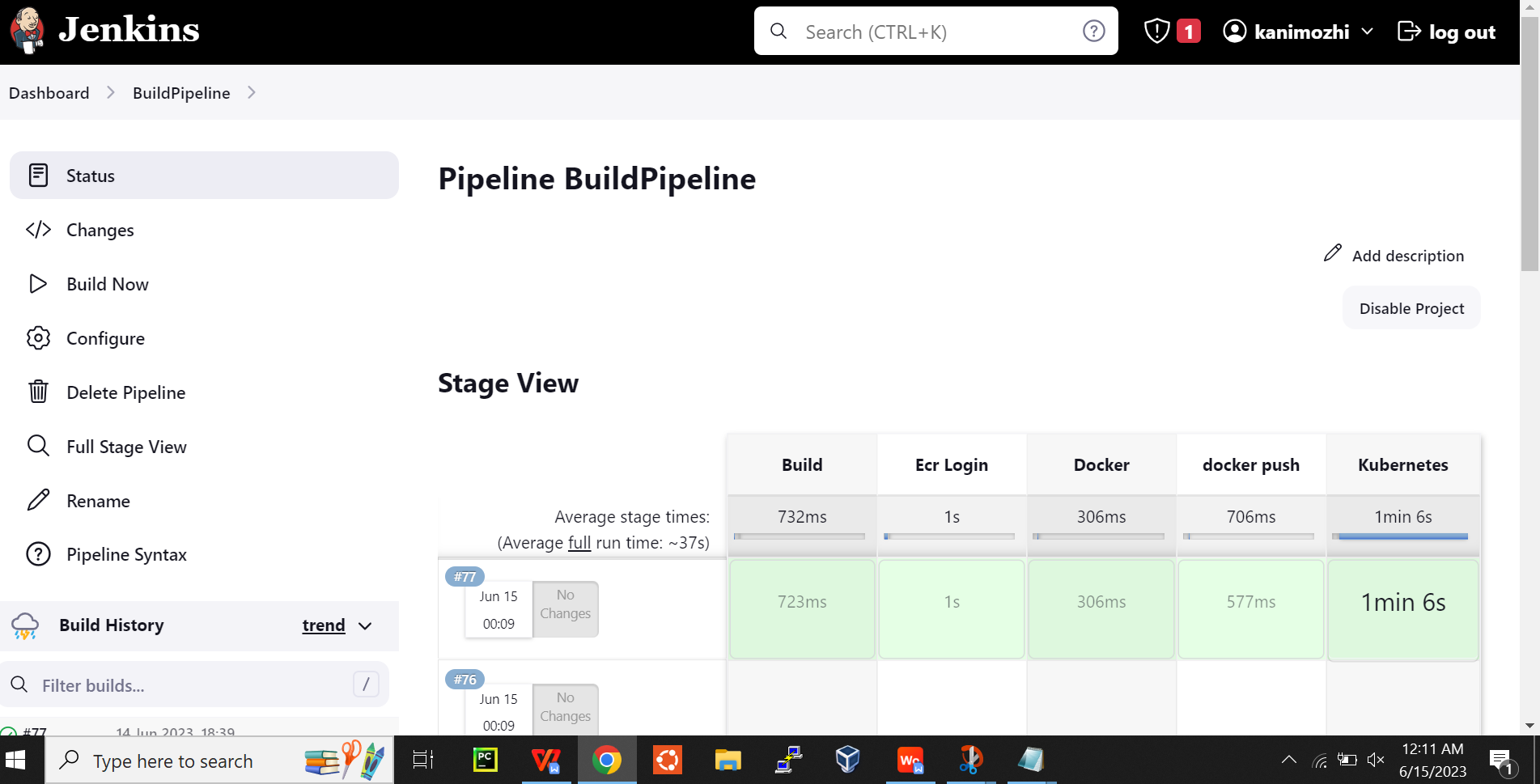
**9.ECR login Build stage view**



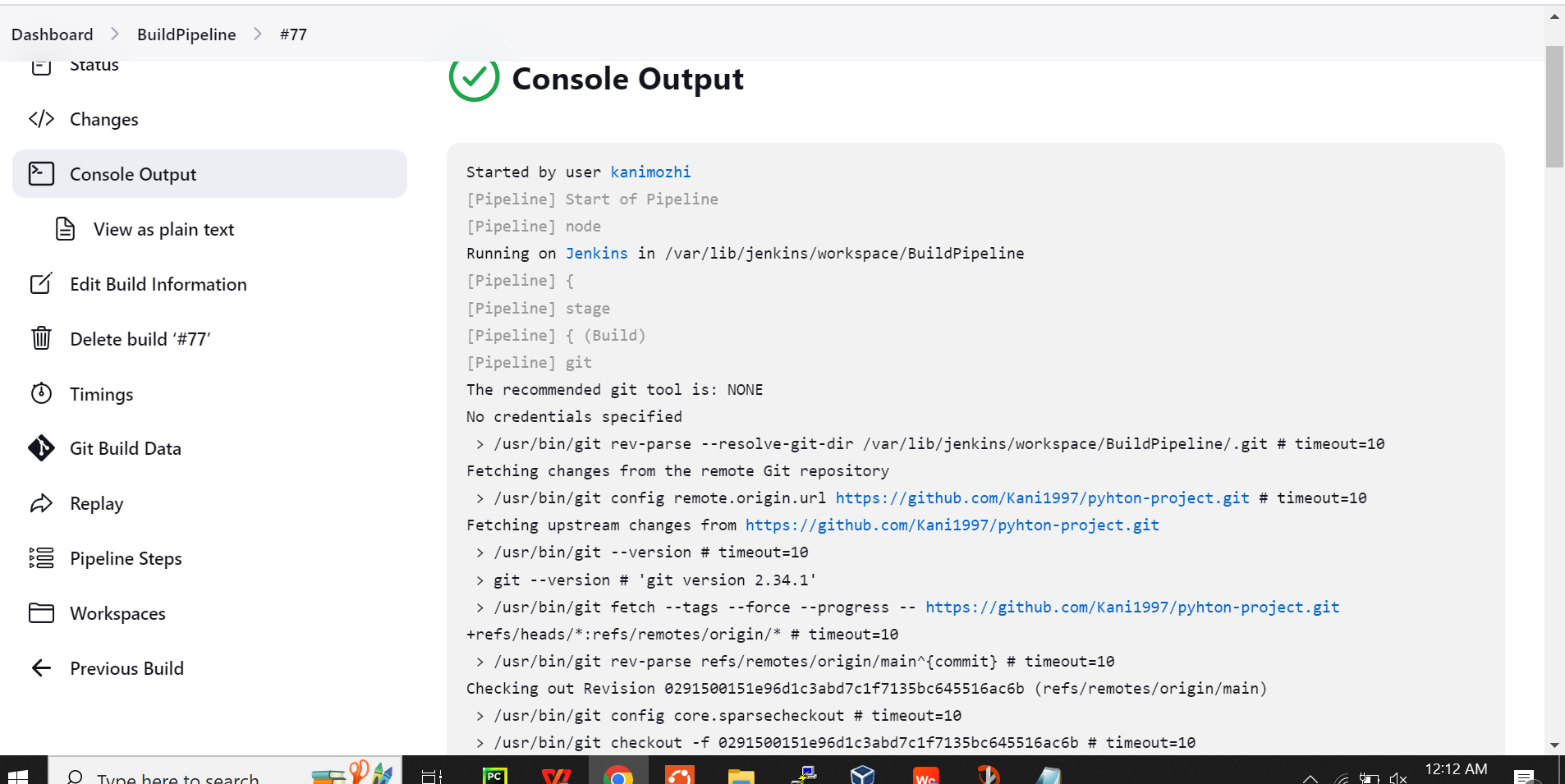
**10.Docker Build image Pushing to ECR**

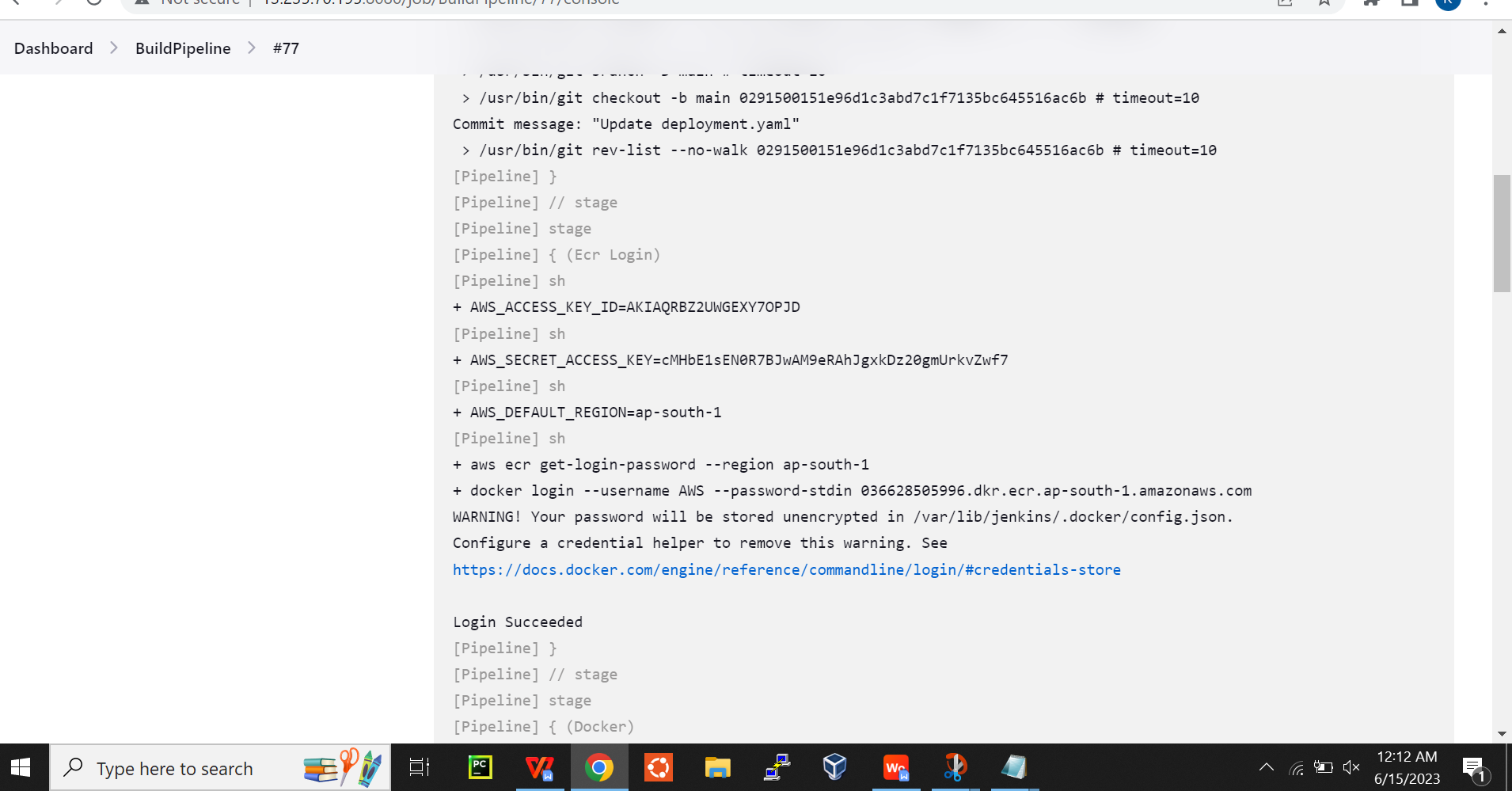


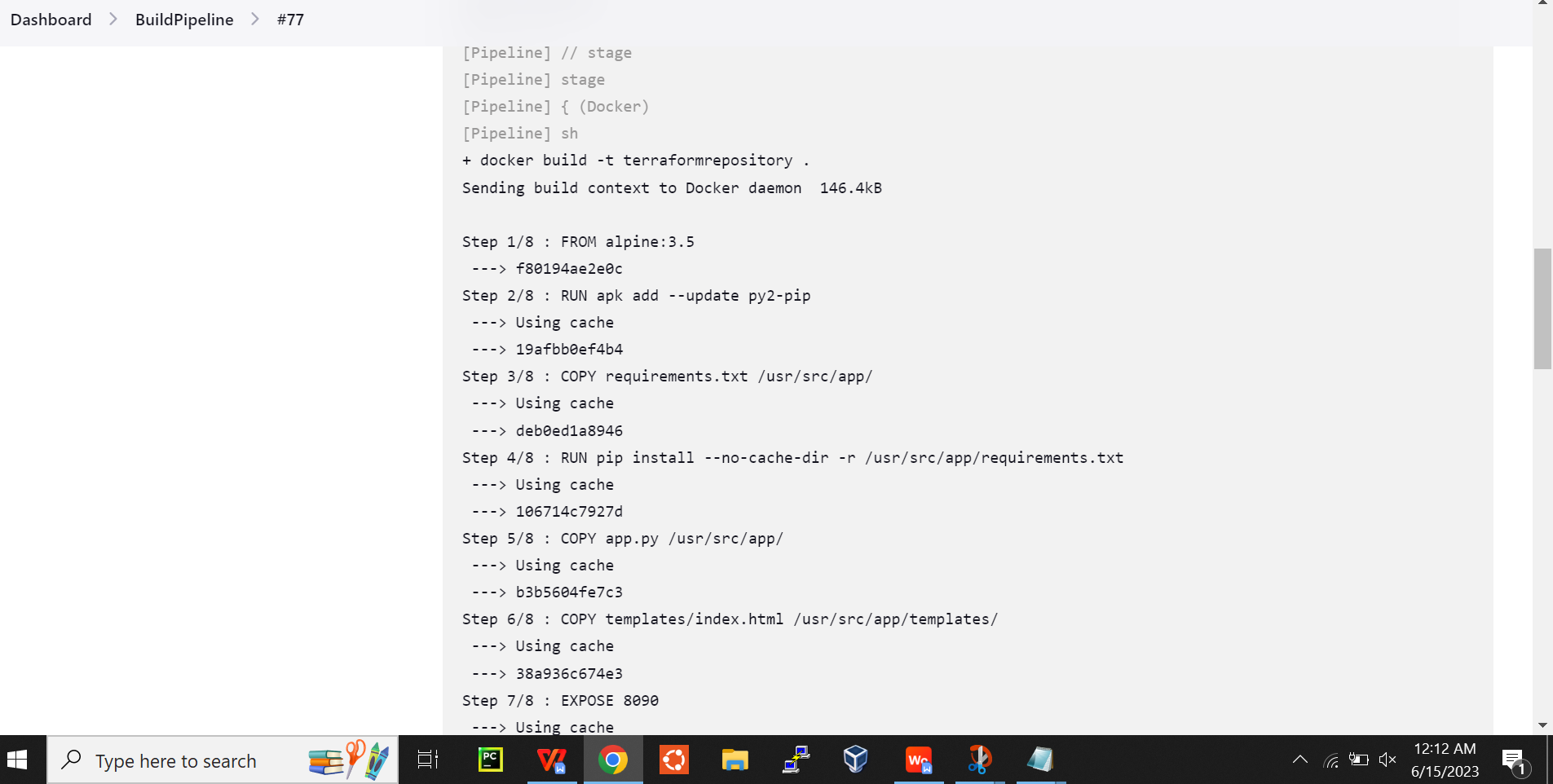
**11.Kubernetes creation**

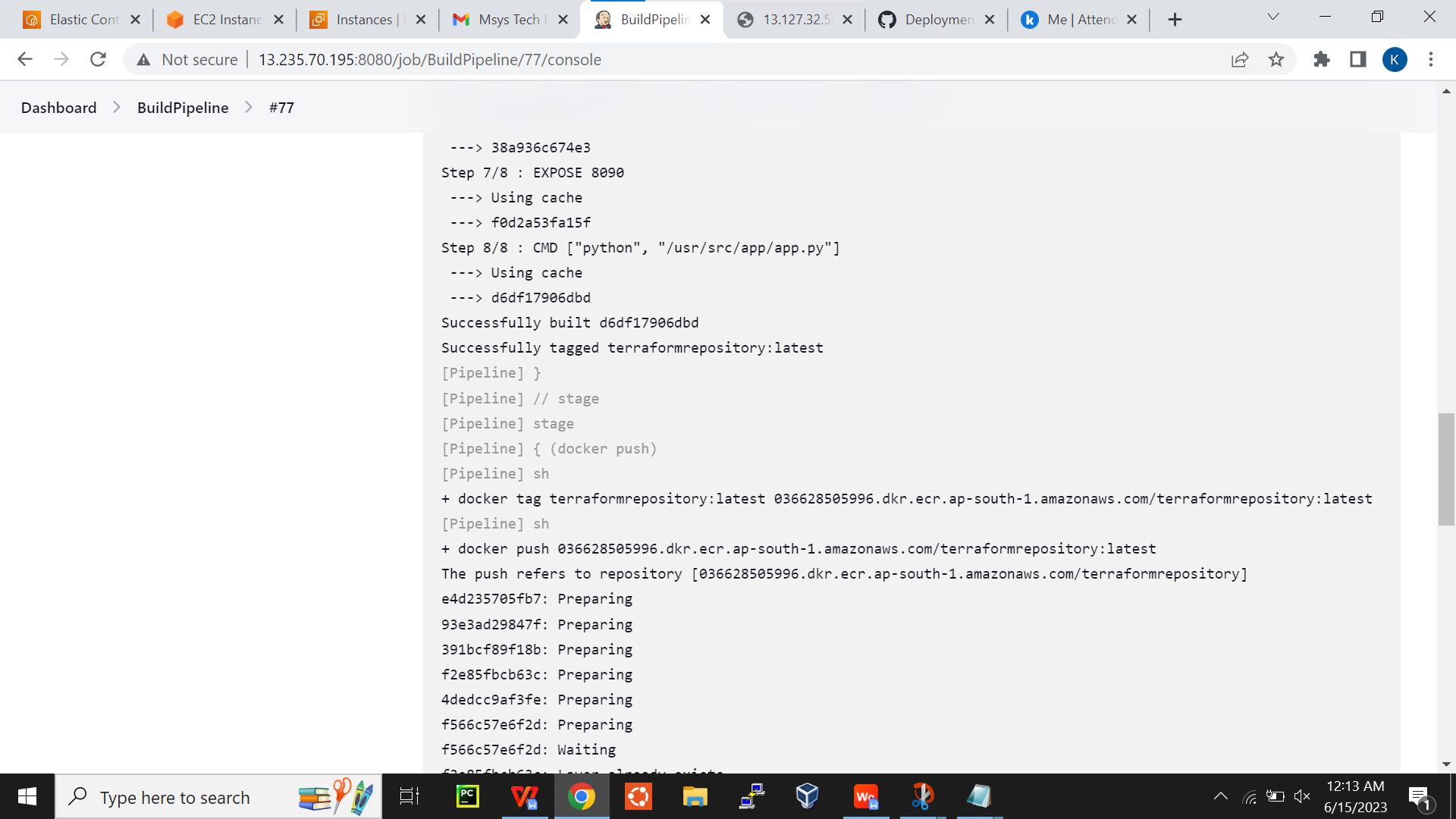


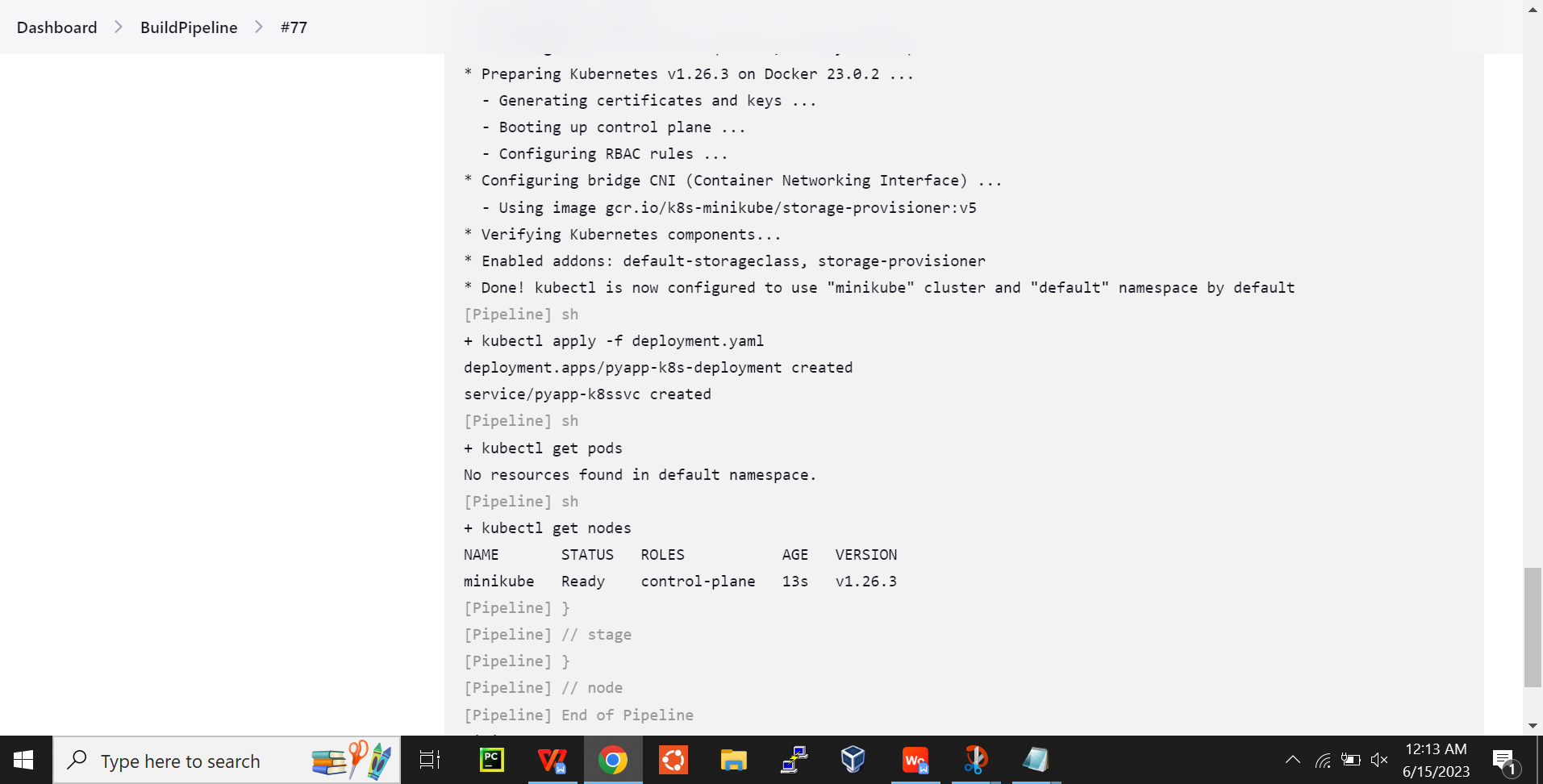
**OverAll Console Output:**

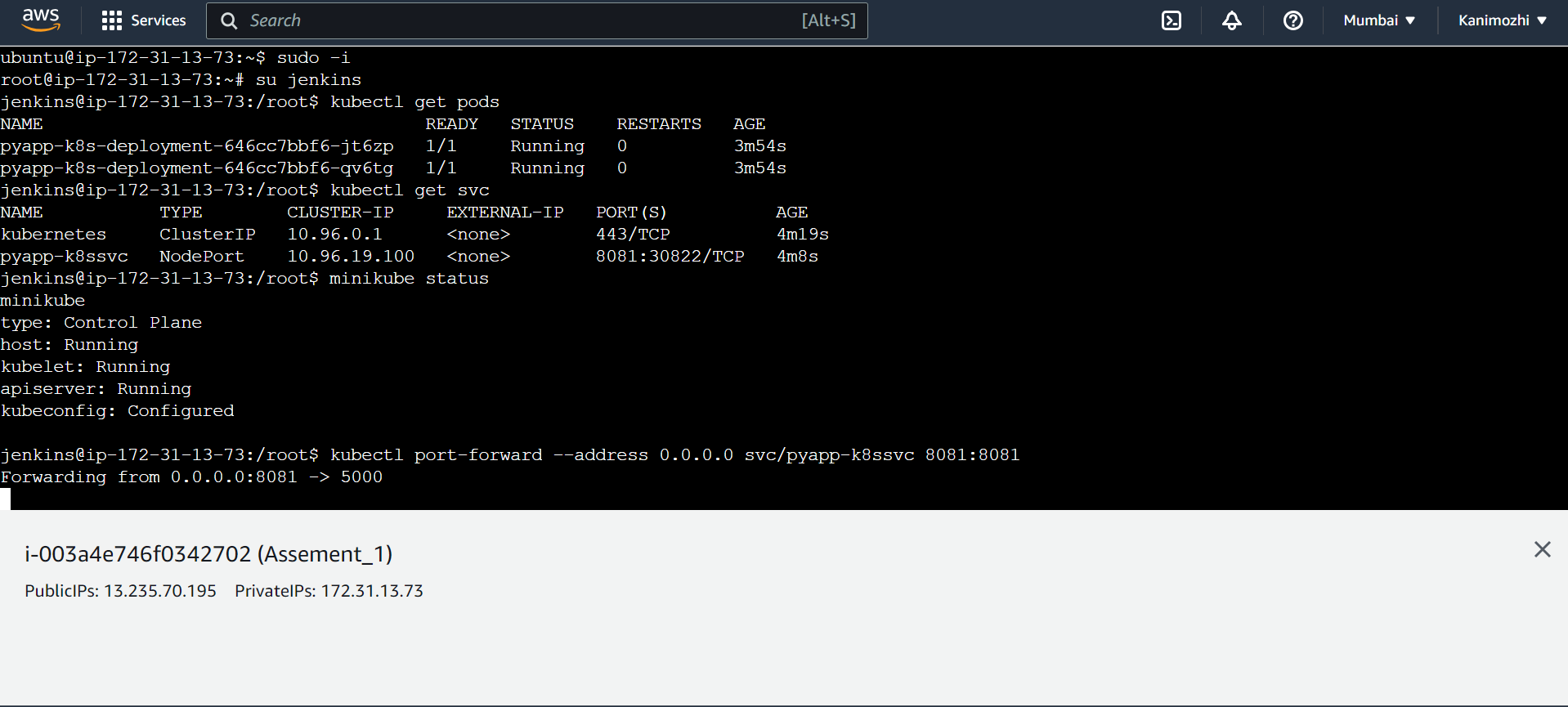












Output:

