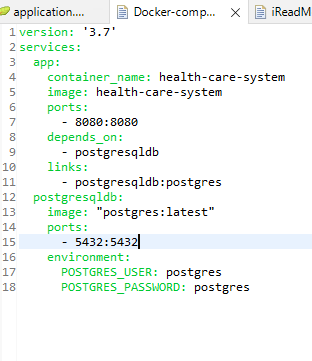
Sprint 2

*Documentation on Healthcare System*

1. Docker –compose
2. Create the Dockerfile
3. Create the docker –compose.yaml file
4. Define the services in the docker-compose file



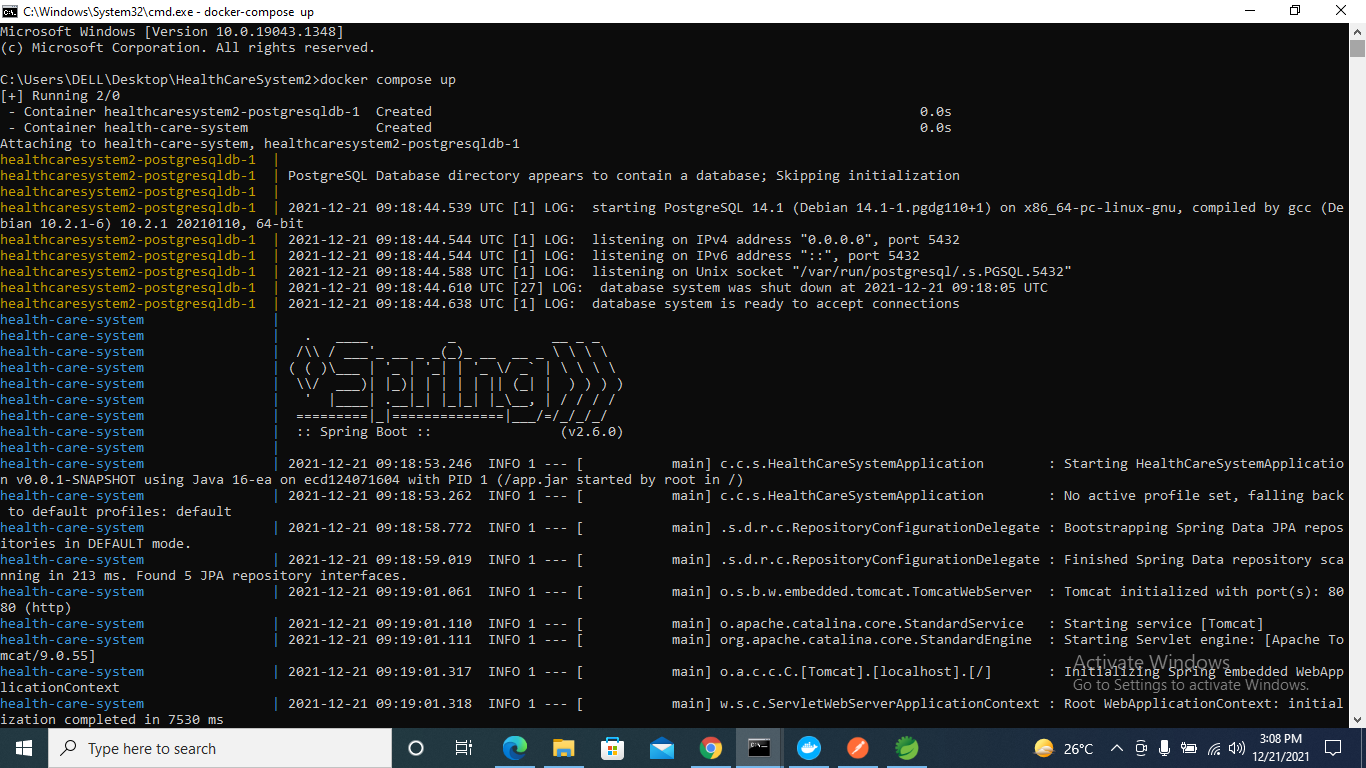
1. Build the image by using the below command

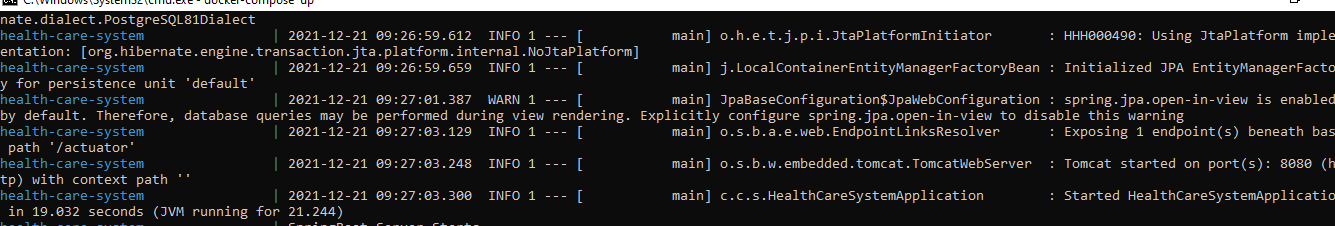
$ docker build –t health-care-system .

1. For execution write the below command

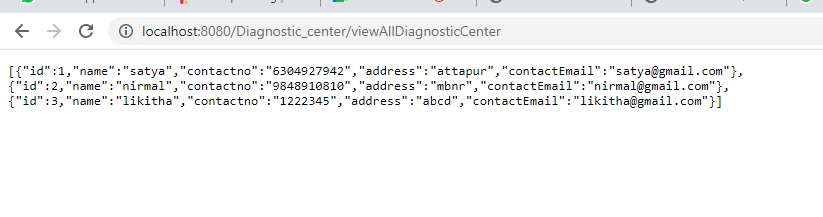
$ docker-compose up

1. Check wheather the application with port is working or not

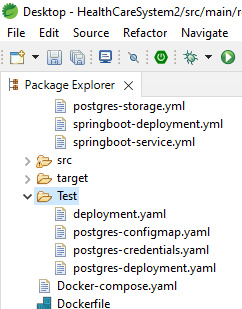




Getting the data in browser

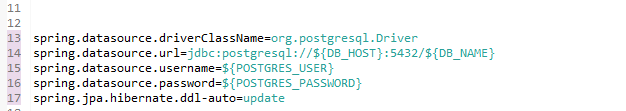


Kubernetes



* $kubectl deployment.yaml
* $kubectl postgres-configmap.yaml
* $kubectl postgres-credentials.yaml
* $kubectl postgres-service.yaml

1. Change the application properties in Springboot .



1. Build the maven , while building the maven with skipping the test cases the jar files are created.
2. Now, open the cmd where the application was present build the images using the below command

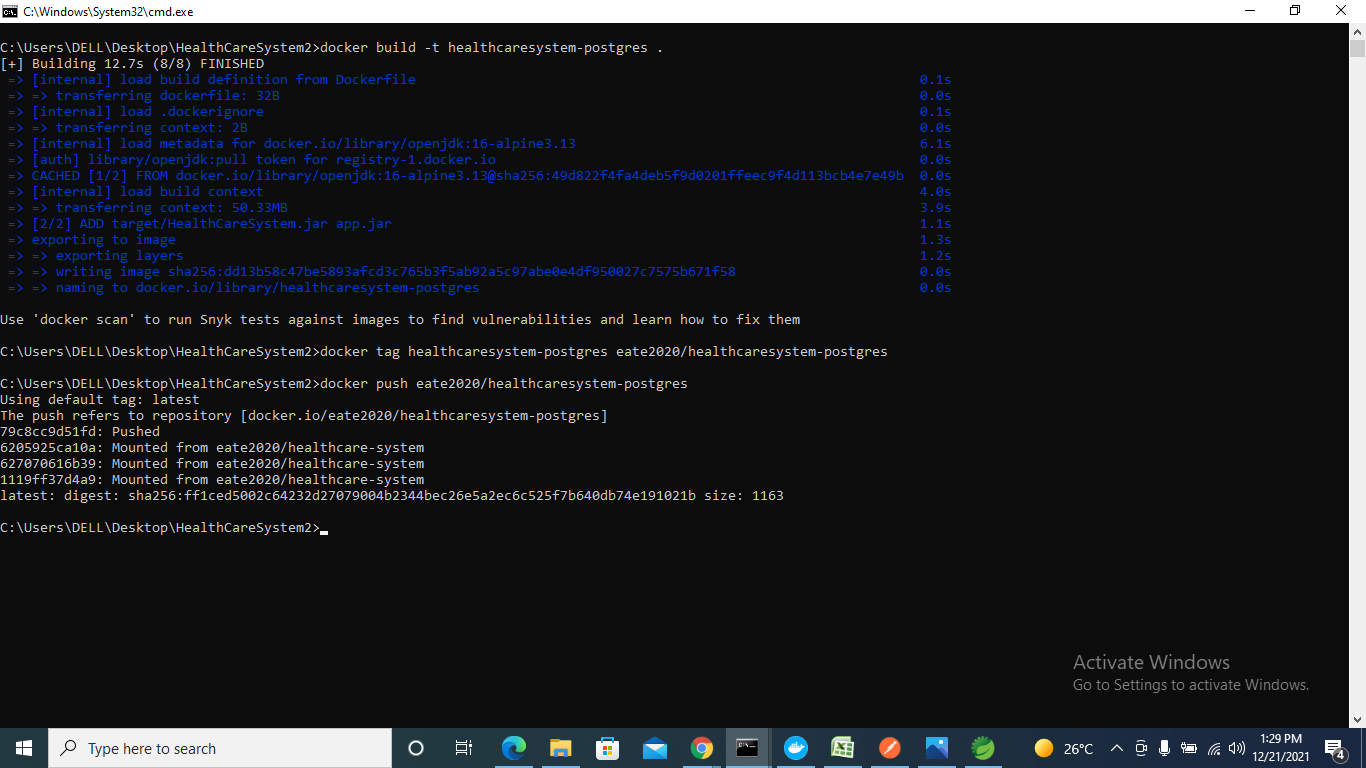
$docker build –t healthcaresystem-postgres .

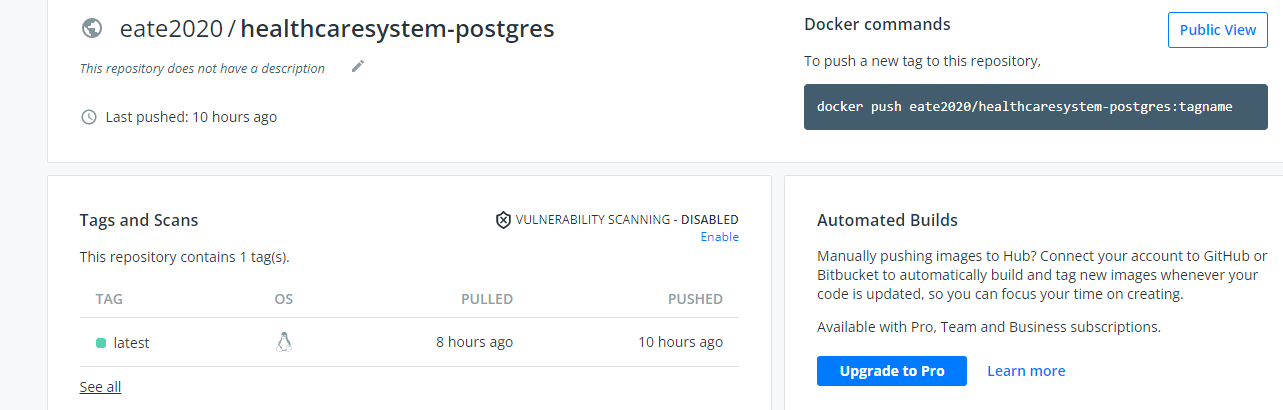
1. Give the tag to the image using the below command

$docker tag healthcaresystem-postgres eate2020/ healthcaresystem-postgres

1. Push the image into docker hub using the command

$docker push eate2020/healthcaresystem-posgres





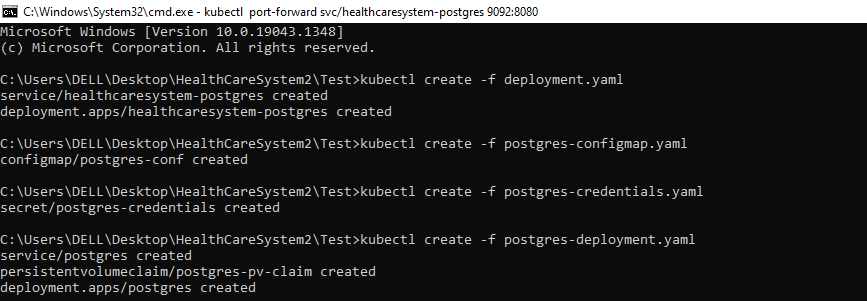
1. Before starting the deploying we need to check that kubernetes is present in the docker or not by using the command

$kubectl –-version

1. If it present then proceed with minikube installation else install kubectl from the chrome and set the path in environments.
2. After installing the minikube ,start the minikube by using the command

$minikube start

1. Now, deploy the application on kubernetes by creating the yamls files in the cmd by using command



$kubectl create -f deployment.yaml

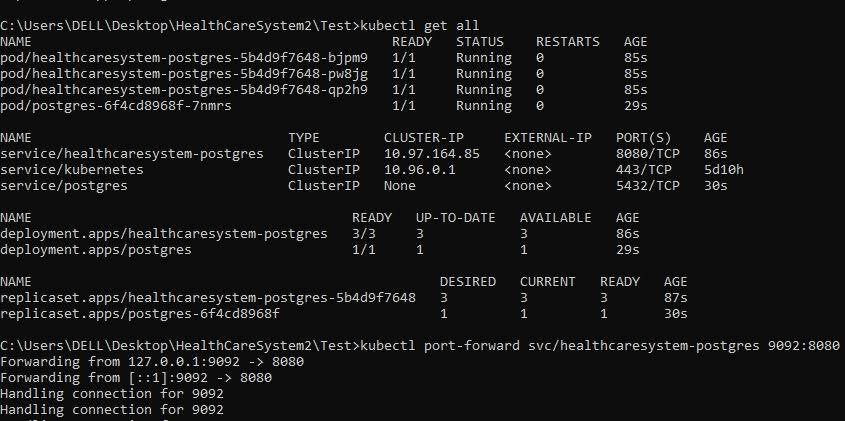
$kubectl create –f postgres-configmap.yaml

$kubectl create –f postgres-credentials.yaml

$kubectl create –f postgres-deployment.yaml

1. To know the running of the application use the command

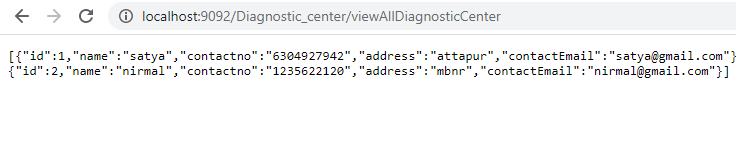
$kubectl get all



1. After successful deployment forward the port number to check the application in the browser

$kubectl port-forward svc/healthcaresystem-postgres 9092:8080

Browser snapshots



EKS Cluster

1. For deploying the application in the Eks cluster first need to download the AWS CLI from chrome.
2. To confirm the installation open cmd from start menu, use command

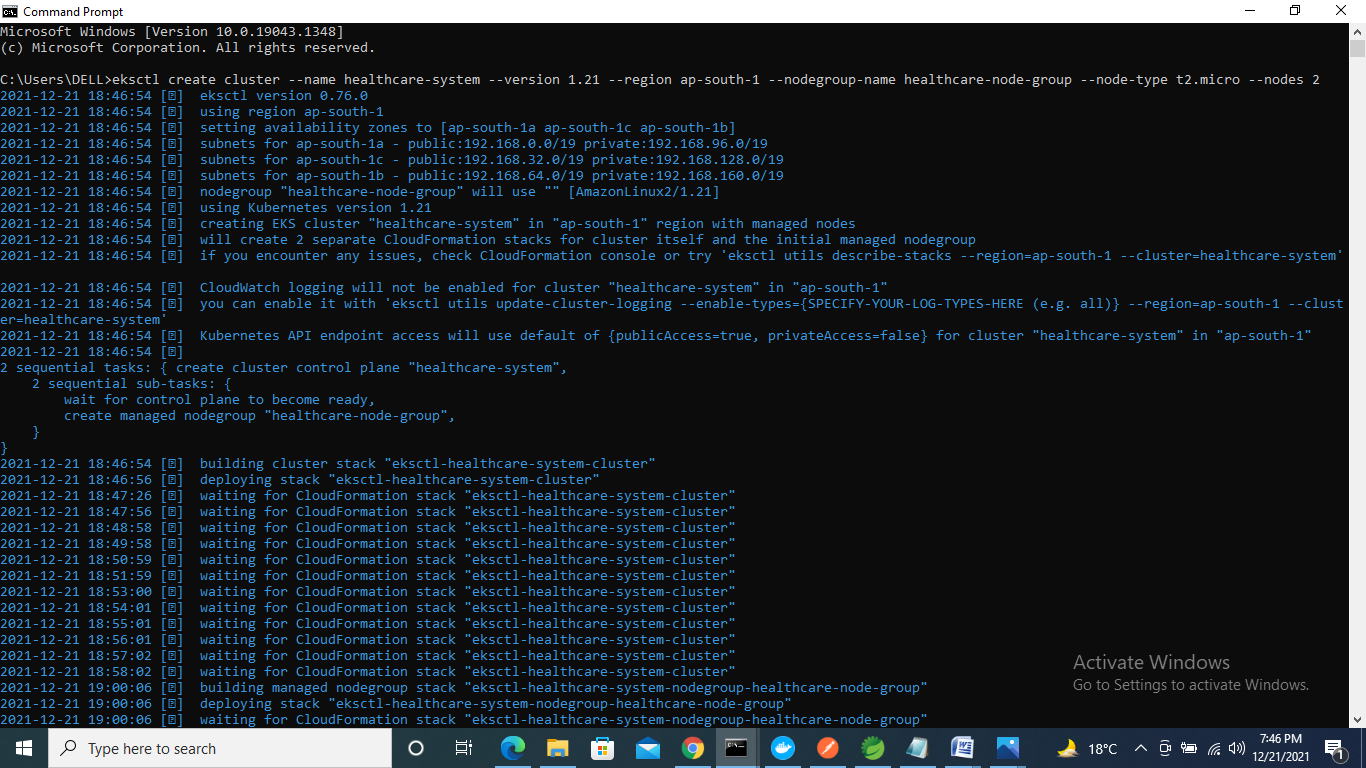
$aws –version

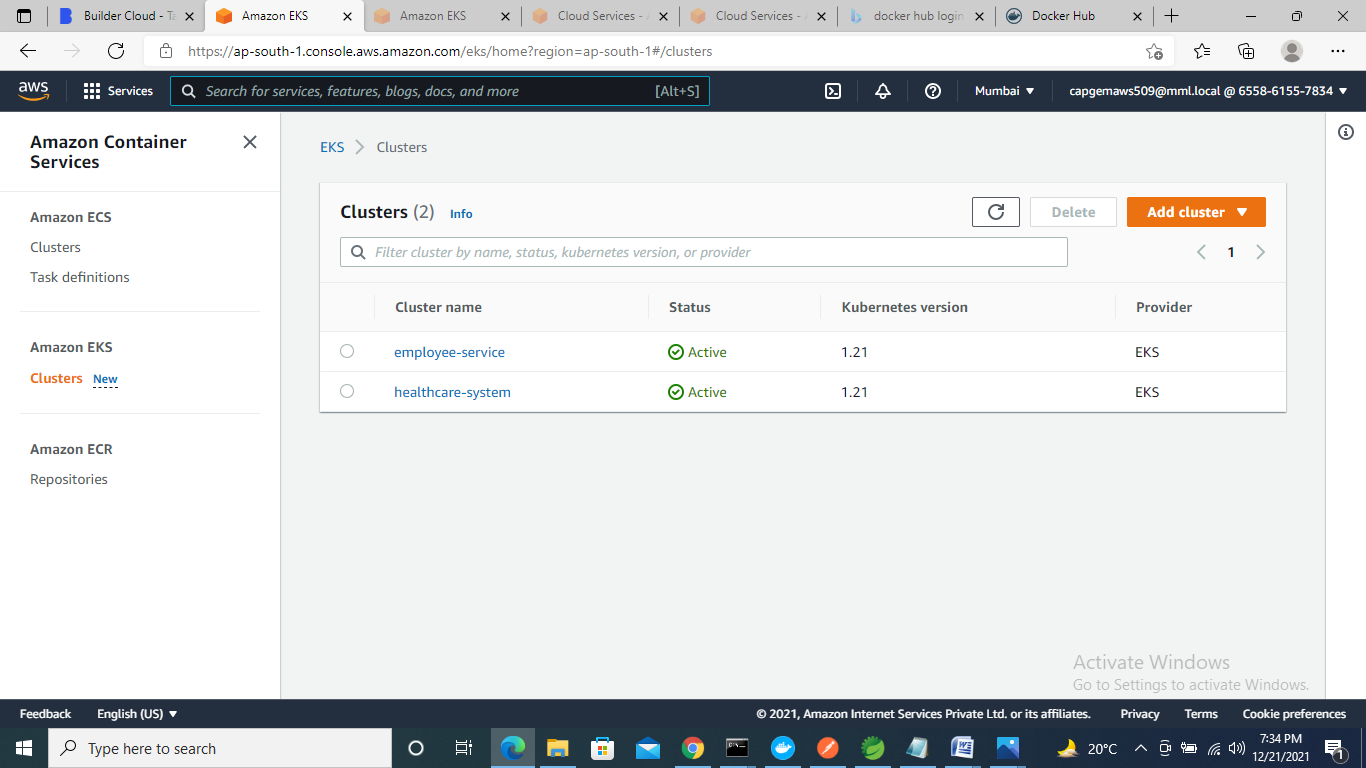
1. To configure the Aws in cmd get the secrets from AWS IAM account.
2. For installation of eksctl, chocolatey has to be installed first.
3. In order to install chocolatey, first, ensure that yu are using an administrative shell.
4. Copy the specific test to the cmd.
5. Wait a few seconds to complete the command and to confirm use command

$ eksctl version

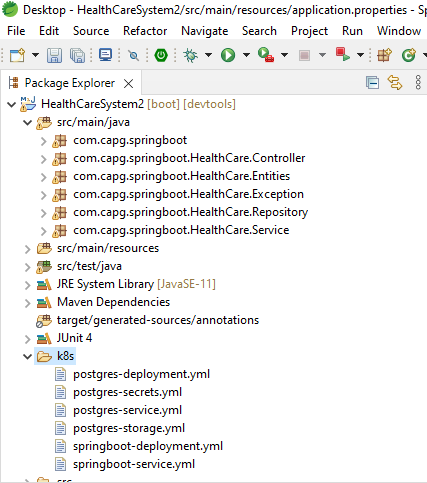
1. For creating the cluster use the below command

$ eksctl create cluster --name healthcare-system --version 1.21 --region ap-south-1 --nodegroup-name healthcare-node-group --node-type t2.micro --nodes 2



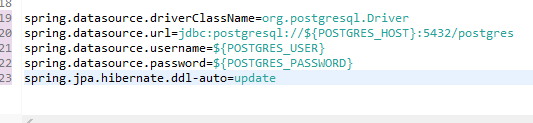


1. Create the yaml files



* Postgres-storage.yaml
* Postgres-secrets.yaml
* Postgres-deployment.yaml
* Postgres-service.yaml
* Springboot-deployment.yaml
* Springboot-service.yaml

1. Change the application properties in the springboot



1. Build the maven so that the new jar files are creates.
2. Open the cmd where the project is present build the image

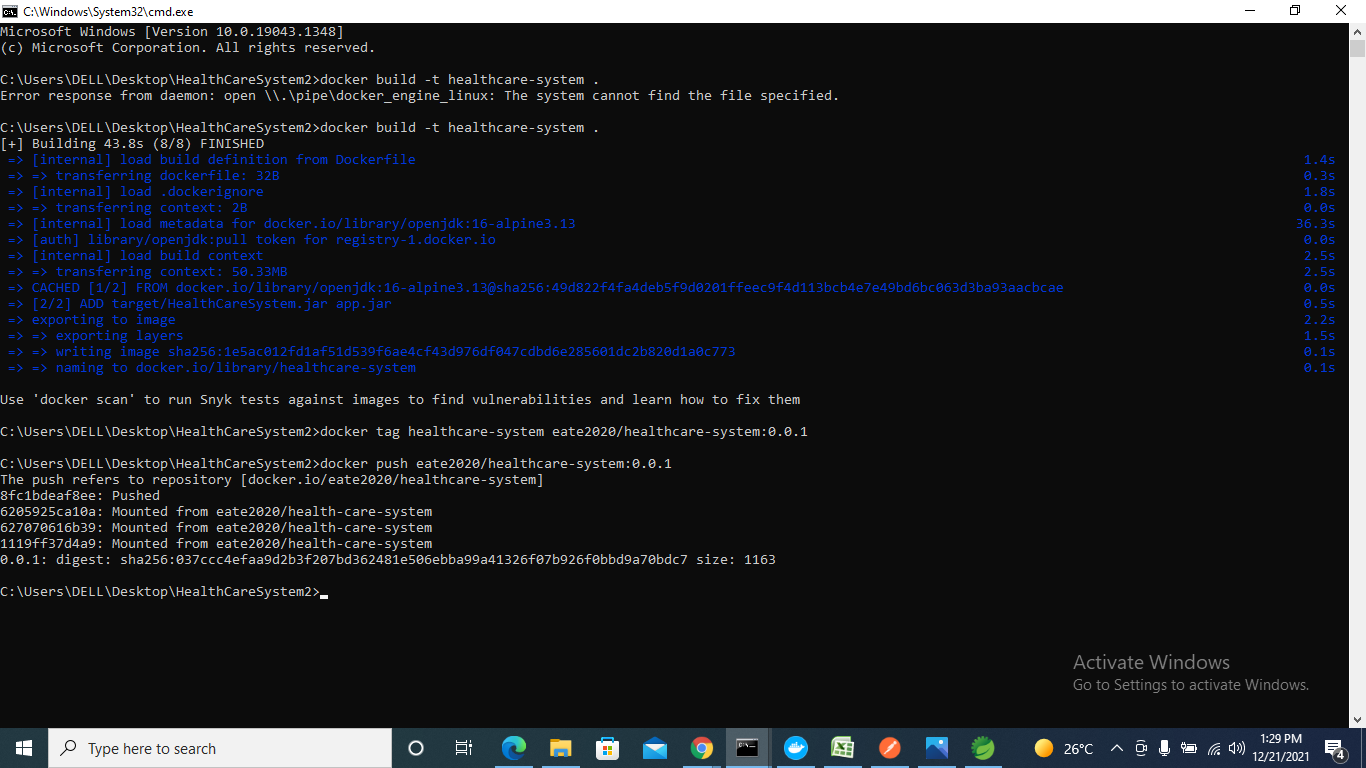
$docker build –t healthcare-system .

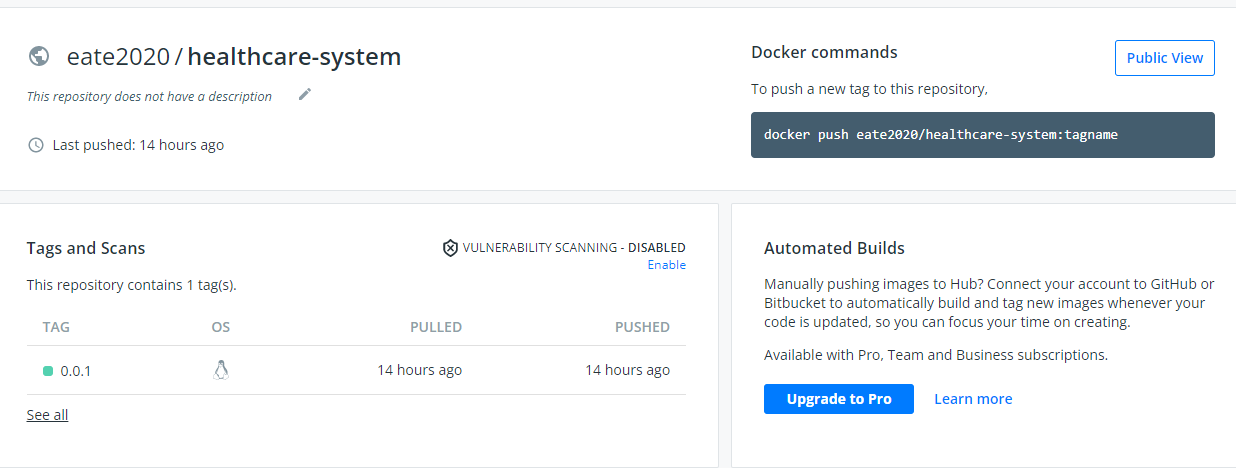
1. Giving the tag to the image

$docker tag healthcare-system eate2020/healthcare-system:0.0.1

1. Push the image into docker hub

$docker push eate2020/healthcare-system:0.0.1





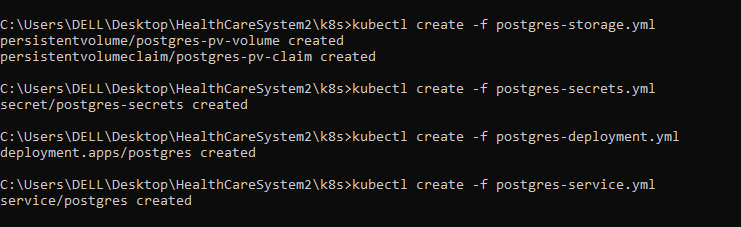
1. Open the cmd from the location of project where(k8s folder) the deployemts files are present.
2. Create the deployment files in the cmd

$kubectl create -f postgres-storage.yaml

$kubectl create -f postgres-secrets.yaml

$kubectl create -f postgres-deployments.yaml

$kubectl create -f postgres-service.yaml



1. To know the status of application use the command

$kubectl get all

1. To set configmap and get the IP address use the below command

$kubectl get svc postgres –o jsonpath=”{.spec.clusterIP}”

1. Get the IP address put in the below command

$kubectl create configmap hostname-config --from-literal=postgres\_host=10.100.253.208

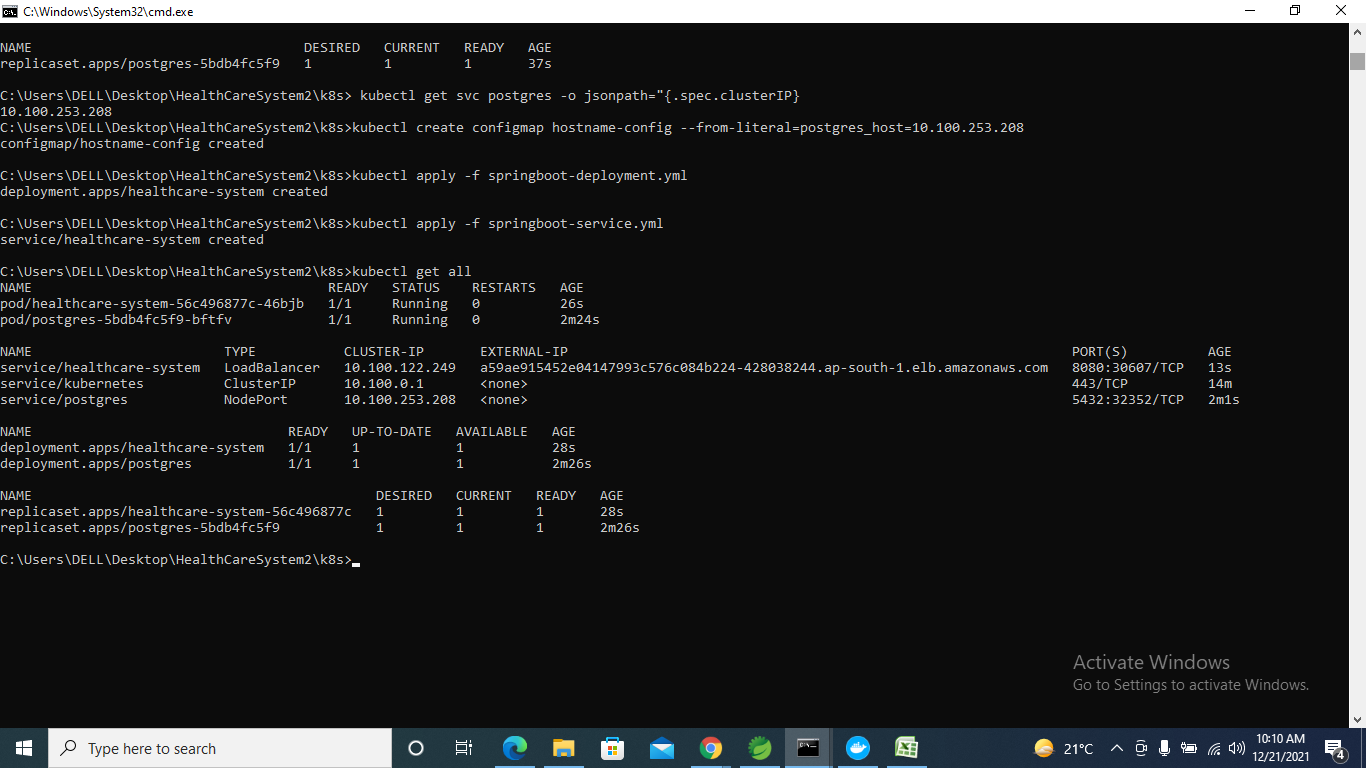
1. To create the remaining deployment files

$kubectl create –f spring-deployment.yaml

$kubectl create –f spring-service.yaml

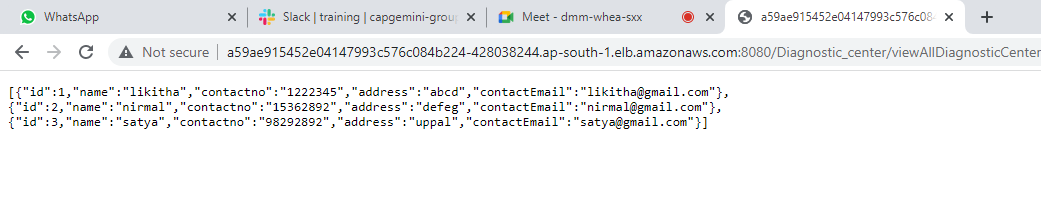
1. To know the status of successful deployments

$kubectl get all



Now, check in the browser by pasting IP address in the browser

http://a59ae915452e04147993c576c084b224-428038244.ap-south-1.elb.amazonaws.com:8080/Diagnostic\_center/viewAllDiagnosticCenter



.