**Problem statement**

Deploy a sample application to a Kubernetes Cluster by following the steps below

1) Clone the following application from the GitHub repository https://github.com/AzureSamples/azure-voting-app-redis.git

2) Create a docker image using the files cloned above

3) Create an Azure Container Registry(ACR) Instance

4) Push the image created above to the registry using the tag “latest”

5) Create a Kubernetes Cluster with 3 nodes

6) Run the created image as a container in the cluster created above

Note:

● Other required values can be set as per your discretion.

● Submission of this assessment shall be done in the form of a pdf document containing the labeled screenshots as outlined in the marks distribution section.

**Solution:**

**Steps:**

* Clone the given git repo:

<https://github.com/Azure-Samples/azure-voting-app-redis>

* Using command: git clone <https://github.com/Azure-Samples/azure-voting-app-redis.git>
* Go inside the folder azure-voting-app-redis
* Run command to create the docker file

docker-compose up –d

* Check the docker image created by running below command

docker images

* Create Azure Container Registry

az acr create --resource-group Regroup\_4bfj --name acrContainerRegistryGl --sku Basic

az acr create --resource-group Regroup\_4bfj --name acrcontainerregistrygl1 --sku Basic

* Run below command to start docker

systemctl enable docker.service

* Log in to the container registry

Connect-AzContainerRegistry -Name acrContainerRegistryGl

Connect-AzContainerRegistry -Name acrcontainerregistrygl1

* To use the azure-vote-front container image with ACR,

(Get-AzContainerRegistry -ResourceGroupName Regroup\_4bfj -Name acrContainerRegistryGl).LoginServer

(Get-AzContainerRegistry -ResourceGroupName Regroup\_4bfj -Name acrcontainerregistrygl1).LoginServer

* Enable Admin Login for ACR in the portal
* Login to portal with credentials

docker login --username acrContainerRegistryGl acrcontainerregistrygl.azurecr.io

docker login --username acrcontainerregistrygl1 acrcontainerregistrygl.azurecr.io

* Tag the images

docker tag mcr.microsoft.com/azuredocs/azure-vote-front:v1 acrContainerRegistryGl.azurecr.io/azure-vote-front:v1

docker tag mcr.microsoft.com/azuredocs/azure-vote-front:v1 acrcontainerregistrygl1.azurecr.io/azure-vote-front:v1

* Push the image to registry

docker push acrContainerRegistryGl.azurecr.io/azure-vote-front:v1

docker push acrcontainerregistrygl1.azurecr.io/azure-vote-front:v1

* Check repositories under ACR to see new repository pushed
* To check health of registry

az acr check-health --name acrContainerRegistryGl

az acr repository show-tags --name acrcontainerregistrygl1 --repository azure-vote-front --output table

az acr repository list --name acrContainerRegistryGl --output table

* To install Cli

az aks install-cli

* Creation of Kubernetes cluster with below command or we can do it on portal

az aks create --resource-group Regroup\_4bfj --name myAKSCluster --node-count 2 --generate-ssh-keys --attach-acr acrcontainerregistrygl1

* Get credentials from power shell with below command

az aks get-credentials --resource-group Regroup\_4bfj --name myAKSCluster

* To verify the connection to our cluster run below command

kubectl get nodes

* Create yaml file for configuration with below command

nano azure-vote.yaml

* Copy the content from git repo

https://github.com/Azure-Samples/azure-voting-app-redis/blob/master/azure-vote-all-in-one-redis.yaml

We need to provide our own ACR login server name so that your manifest file looks like the following example:

YAML

Copy

containers:

- name: azure-vote-front

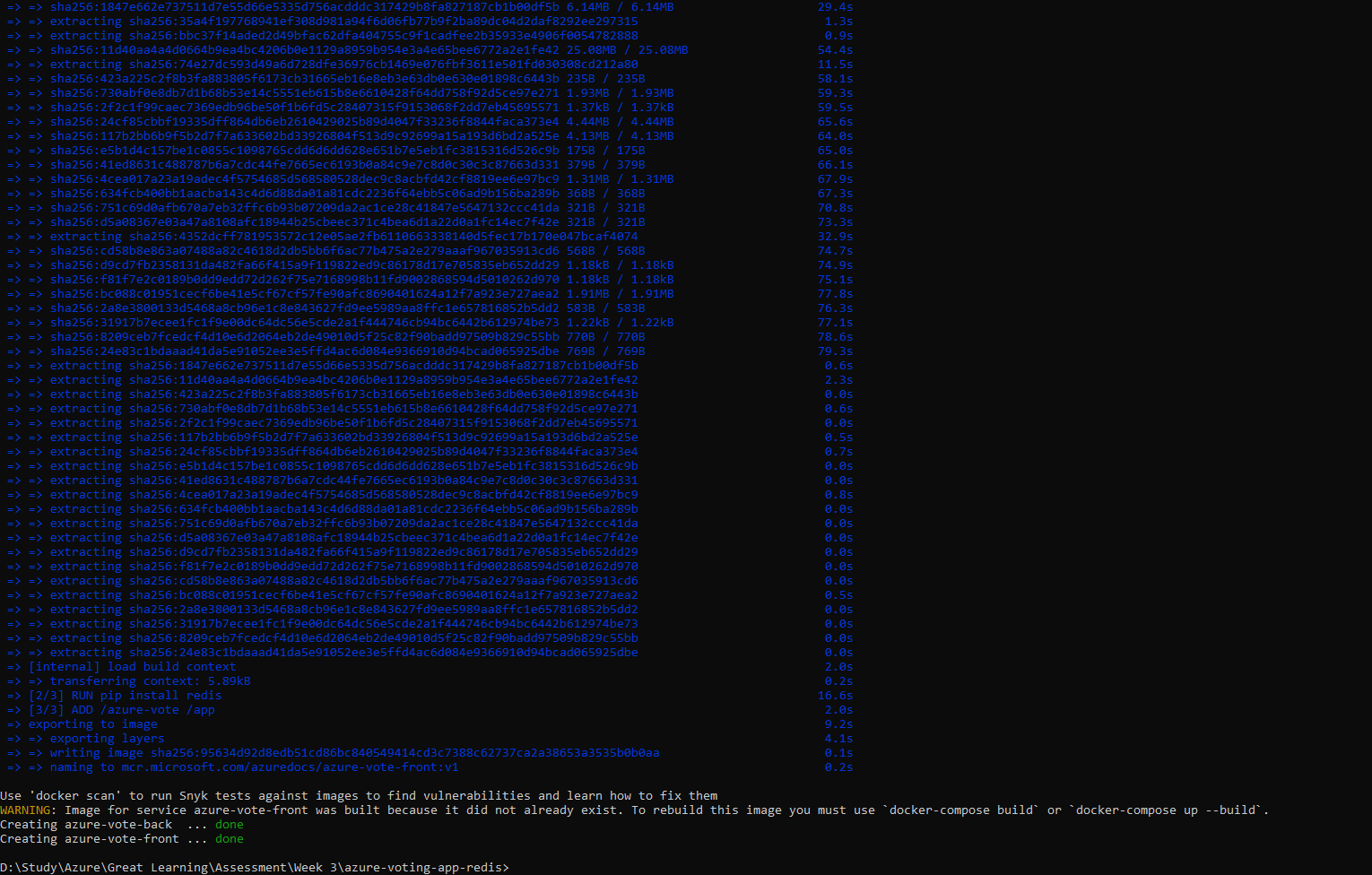
image: acrcontainerregistrygl1.azurecr.io/azure-vote-front:v1

* Run Kubectl command: kubectl apply -f azure-vote.yaml
* Run Command: kubectl get service azure-vote-front --watch
* App is deployed on IP 13.87.200.134

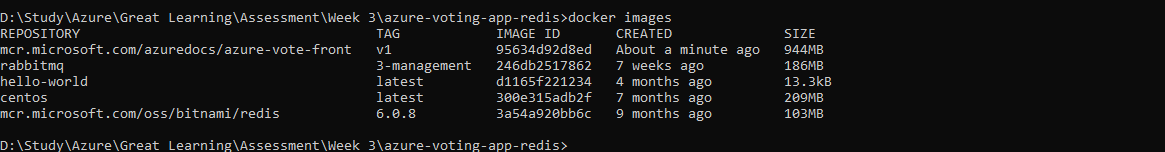
If we launch the IP, the voting application will be launched ☺

**Screenshots:**

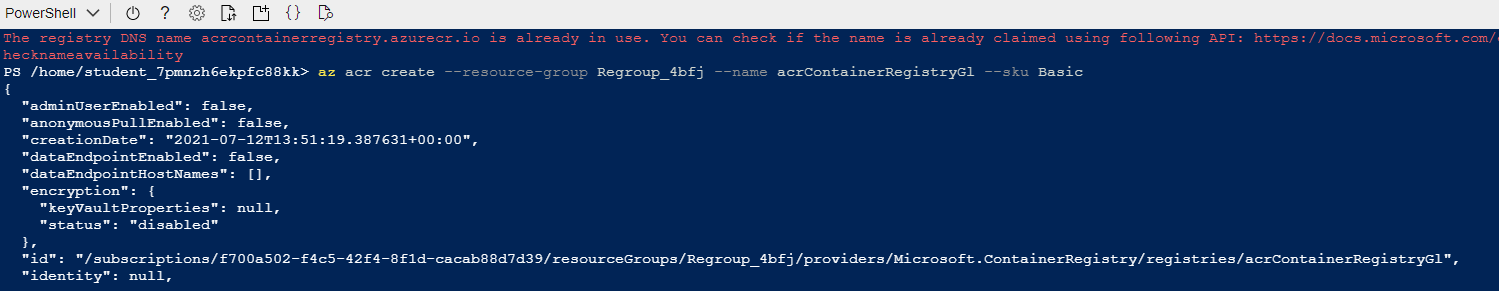
Creation of docker image



Check new docker image created:



Create Azure Container Registry



az acr login -n acrContainerRegistryGl --expose-token

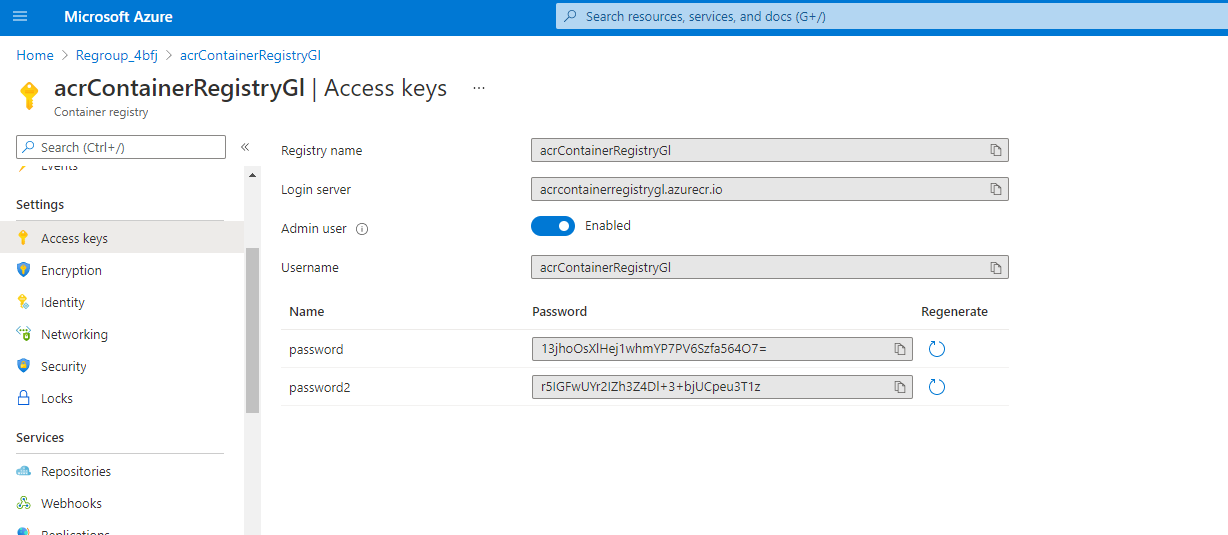


Connect-AzContainerRegistry -Name acrContainerRegistryGl

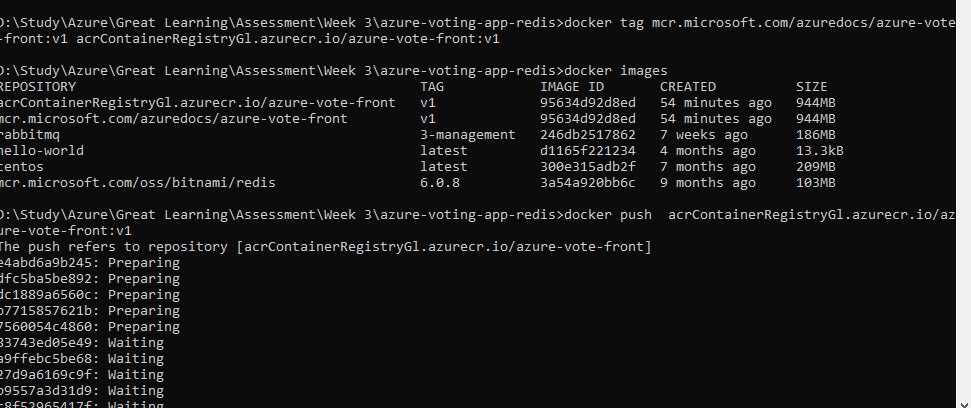


docker tag mcr.microsoft.com/azuredocs/azure-vote-front:v1 acrContainerRegistryGl.azurecr.io/azure-vote-front:v1

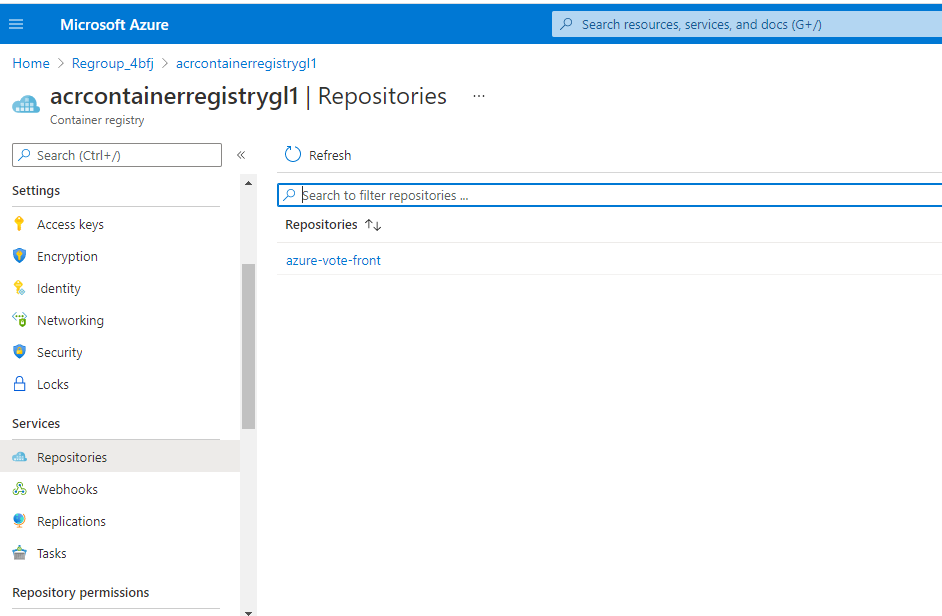
Enable Admin Login on portal to get credentials



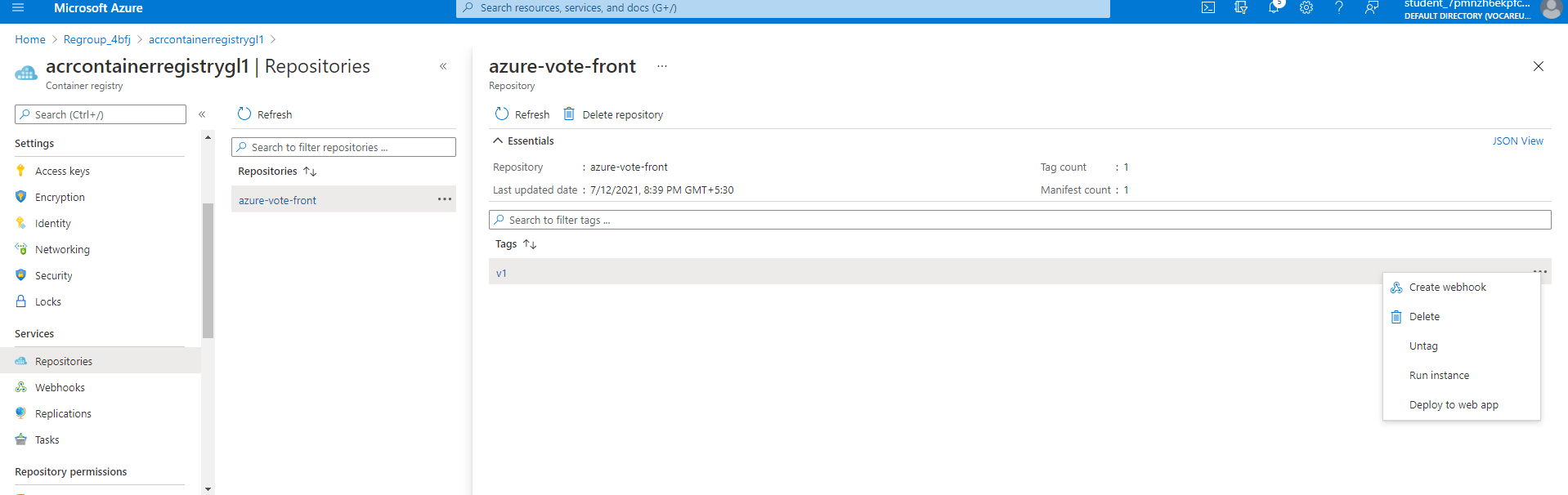
docker push acrContainerRegistryGl.azurecr.io/azure-vote-front:v1



Repository is deployed:

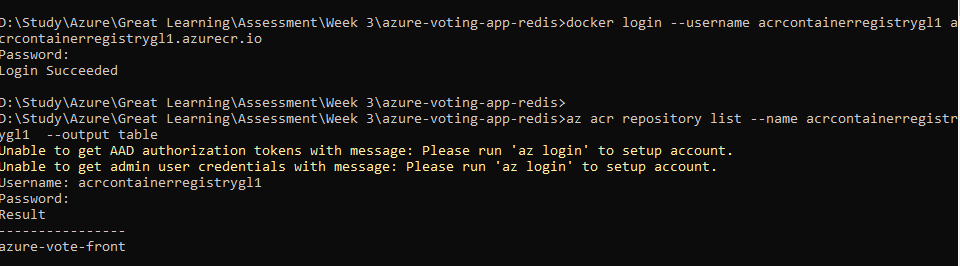


To run instance directly: Run instance

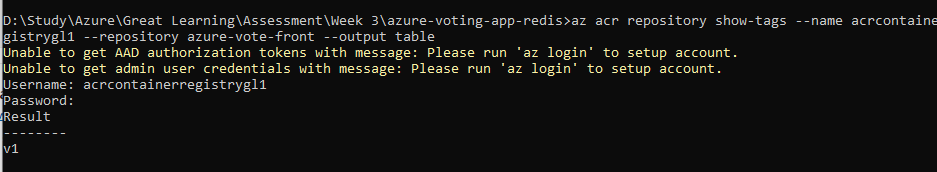




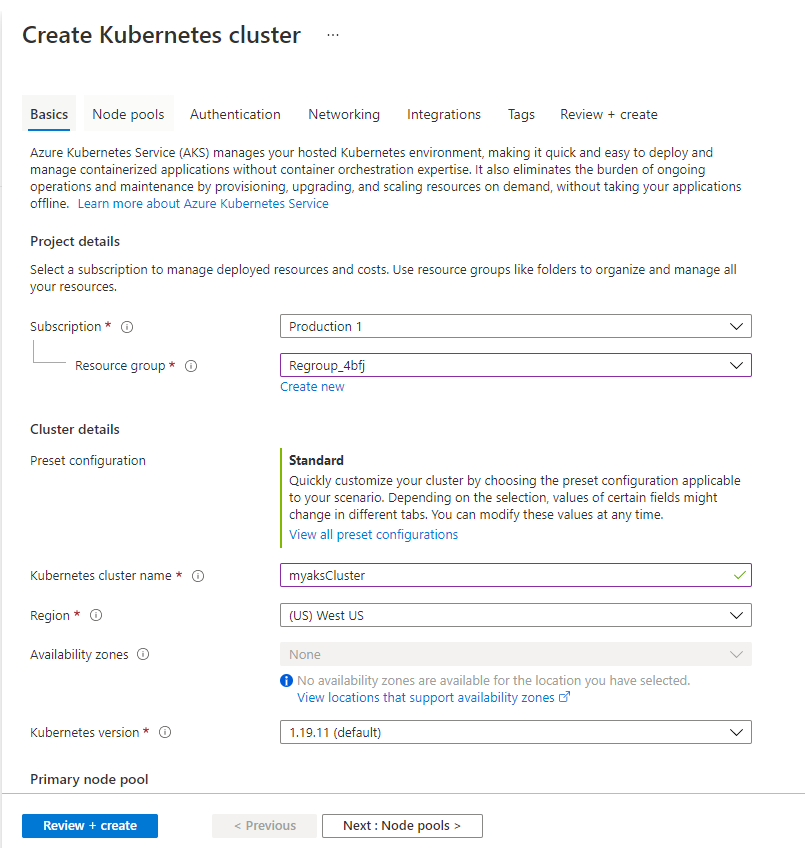
az acr repository list --name acrcontainerregistrygl1 --output table

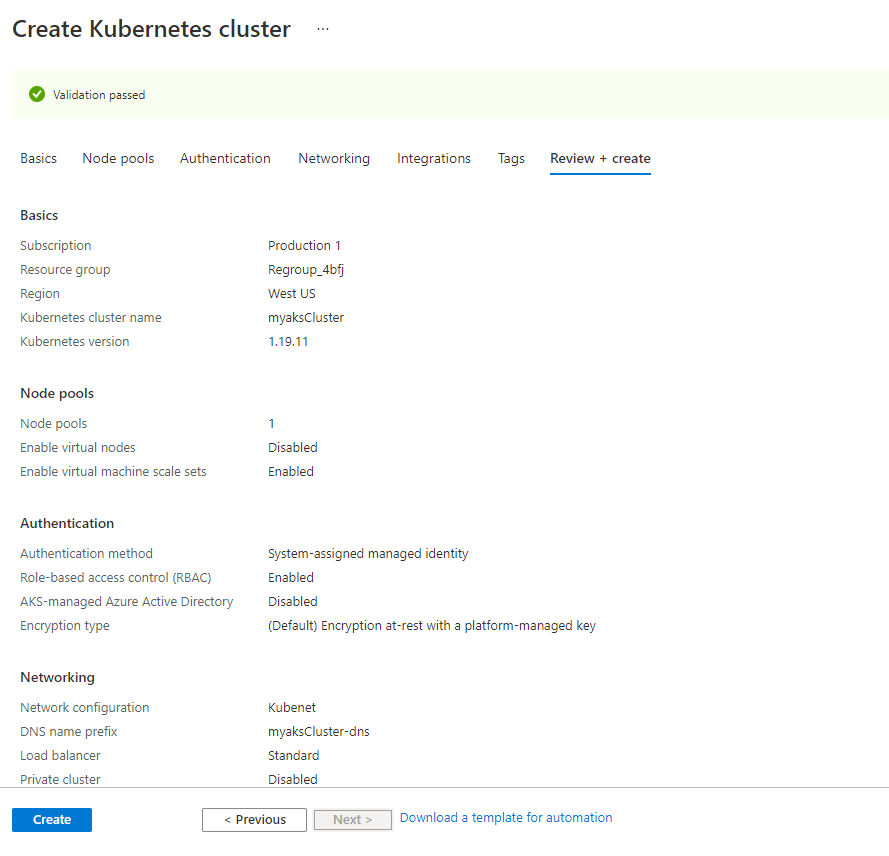


az acr repository show-tags --name acrcontainerregistrygl1 --repository azure-vote-front --output table

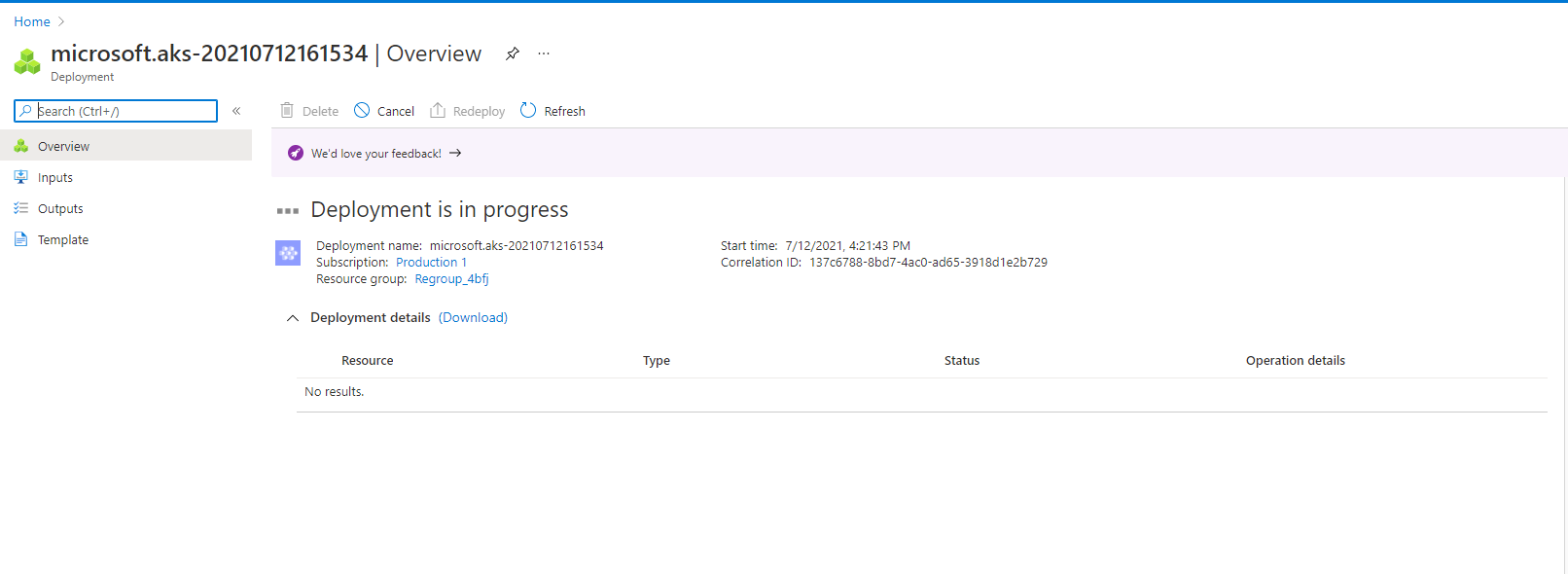


**Create Kubernetes cluster:**

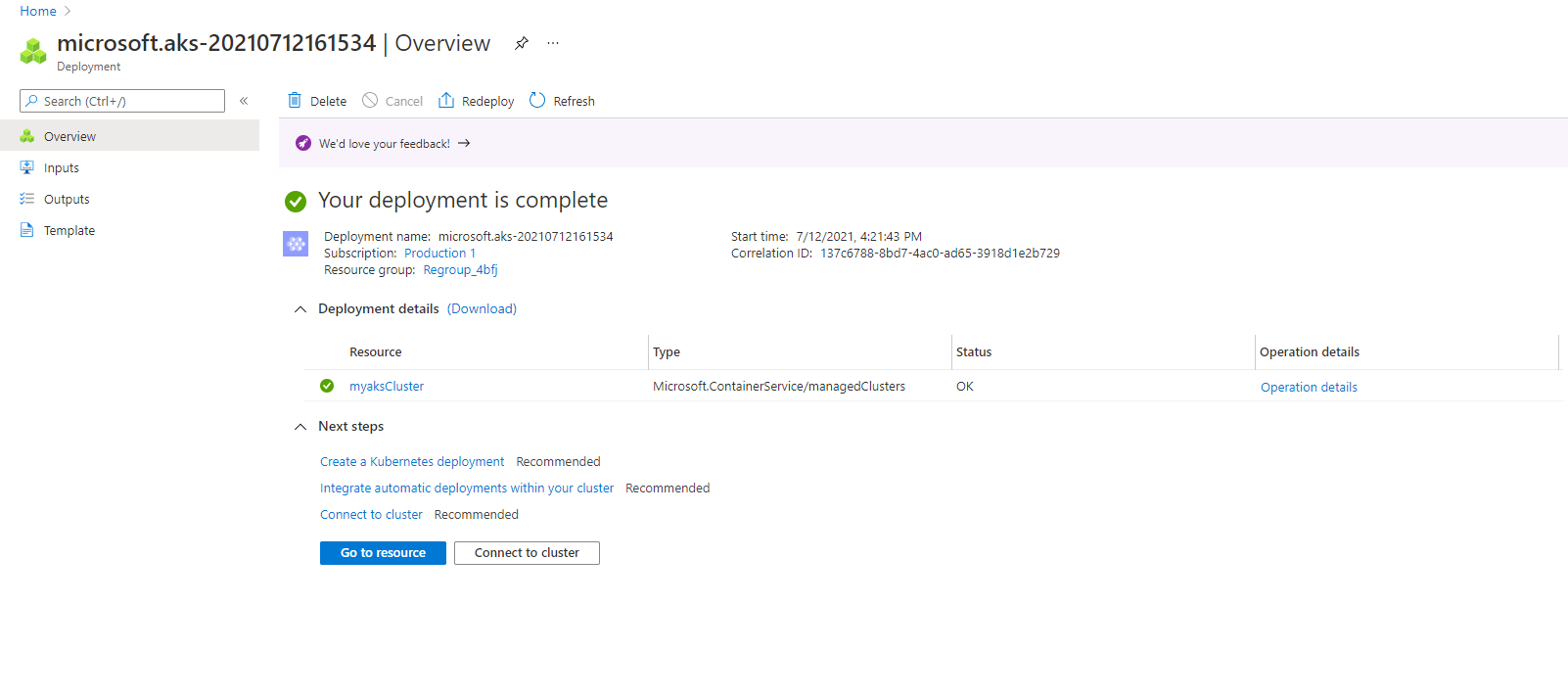




**Deployment in progress:**

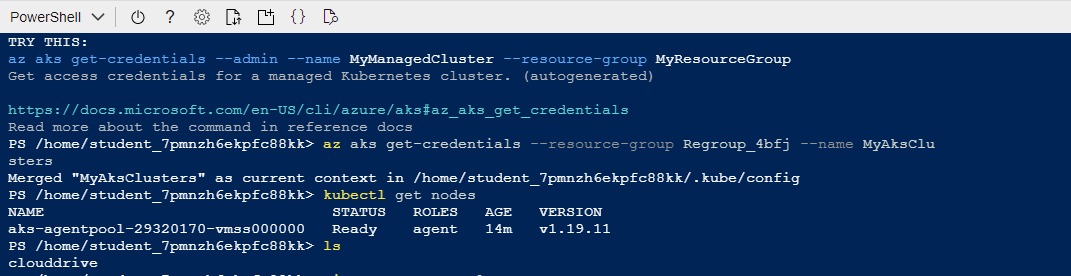


**Deployment Completed:**



**Open PowerShell and run commands:**

* az aks get-credentials --resource-group Regroup\_4bfj --name MyAksClusters
* kubectl get nodes



* Create script: with file name azure-vote.yaml and copy the content from file shared

<https://github.com/Azure-Samples/azure-voting-app-redis/blob/master/azure-vote-all-in-one-redis.yaml>

We need to provide our own ACR login server name so that your manifest file looks like the following example:

YAML

Copy

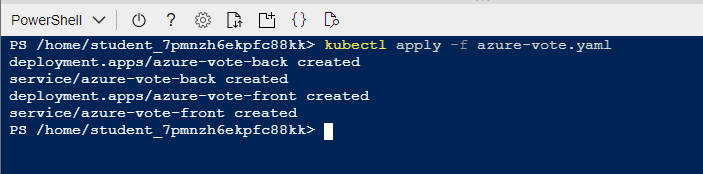
containers:

- name: azure-vote-front

image: acrcontainerregistrygl1.azurecr.io/azure-vote-front:v1



* Run Kubectl command: kubectl apply -f azure-vote.yaml



* Run Command: kubectl get service azure-vote-front --watch



* App is deployed on IP 13.87.200.134

If we launch the IP, we get the below app

