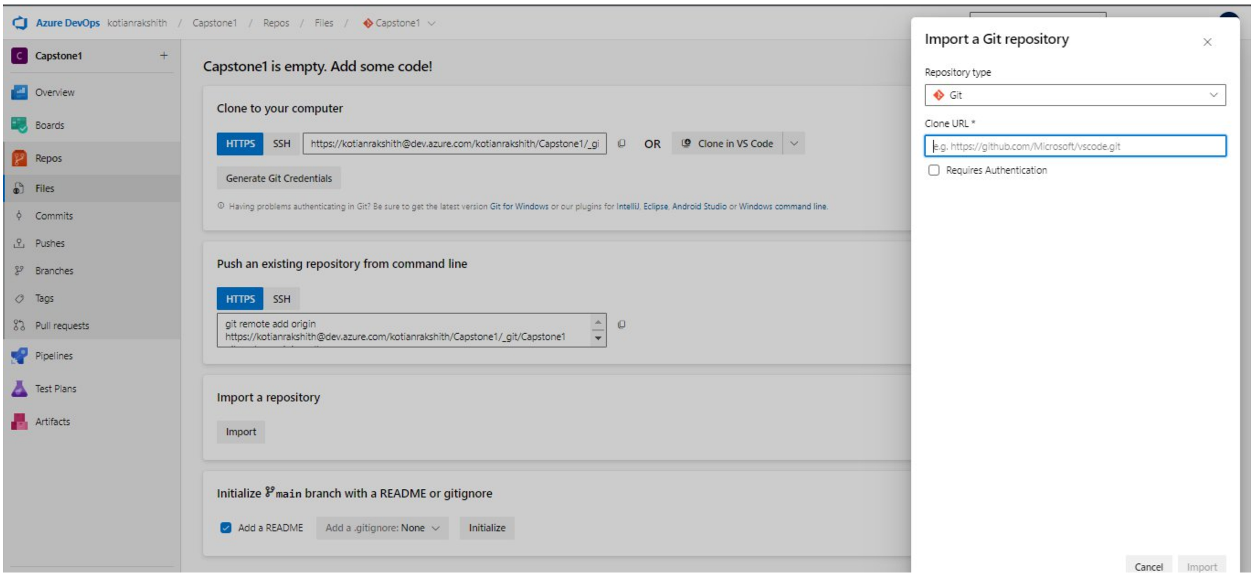
Deployment of Java application through Azure Devops Pipeline, Azure Repos, ACR, Azure Kubentes Service

We have deployed banking application in capstone project1 using jenkins, github, dockerhub, ansible, kubernetes, we will take the same application and deploy through Azure Devops, Azure Repos, Azure container registry, Azure Kubentes Service.

1. **Import the repository:**

   We will create a new project in Azure DevOps and then go to Repo and click on Import a Repository



Capstone1 is empty. Add some code!

Clone to your computer

https://kotianrakshith@dev.azure.com/kotianrakshith/Capstone1/_gi

Having problems authenticating in Git? Be sure to get the latest version Git for Windows or our plugins for IntelliJ, Eclipse, Android Studio or Windows command line.

Push an existing repository from command line

```
git remote add origin
https://kotianrakshith@dev.azure.com/kotianrakshith/Capstone1/_git/Capstone1
```

Import a repository

Initialize main branch with a README or gitignore

☑ Add a README   Add a .gitignore: None

Import a Git repository

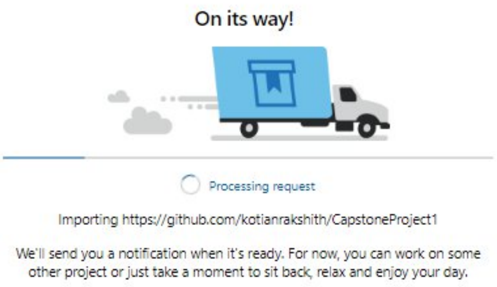
Repository type: Git

Clone URL *: e.g. https://github.com/Microsoft/vscode.git

□ Requires Authentication

We have already created gihub repo for the capstone1 project: https://github.com/kotianrakshith/CapstoneProject1

We will import this:



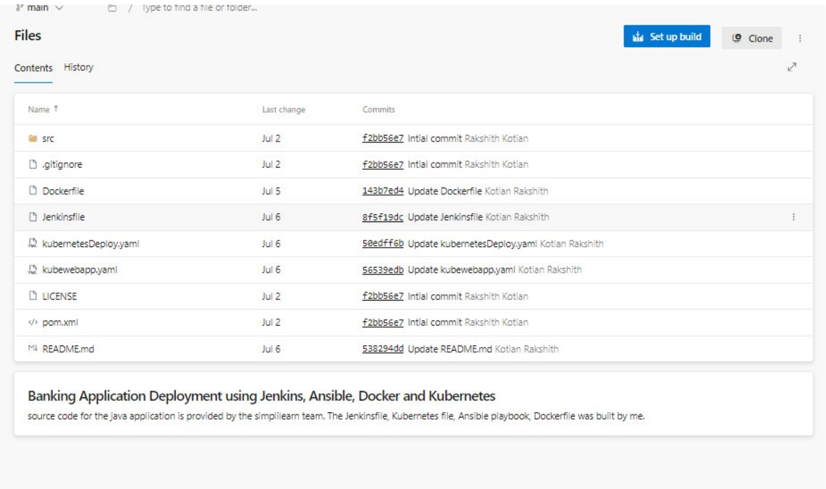
On its way!

Processing request

Importing https://github.com/kotianrakshith/CapstoneProject1

We'll send you a notification when it's ready. For now, you can work on some other project or just take a moment to sit back, relax and enjoy your day.

We can see after the import all the files are imported:



Files

Name ↑	Last change	Commits
src	Jul 2	f2bb56e7 Intial commit Rakshith Kotian
.gitignore	Jul 2	f2bb56e7 Intial commit Rakshith Kotian
Dockerfile	Jul 5	143b7ed4 Update Dockerfile Kotian Rakshith
Jenkinsfile	Jul 6	8f5f19dc Update Jenkinsfile Kotian Rakshith
kubernetesDeploy.yaml	Jul 6	50edff6b Update kubernetesDeploy.yaml Kotian Rakshith
kubewebapp.yaml	Jul 6	56539edb Update kubewebapp.yaml Kotian Rakshith
LICENSE	Jul 2	f2bb56e7 Intial commit Rakshith Kotian
pom.xml	Jul 2	f2bb56e7 Intial commit Rakshith Kotian
README.md	Jul 6	538294dd Update README.md Kotian Rakshith

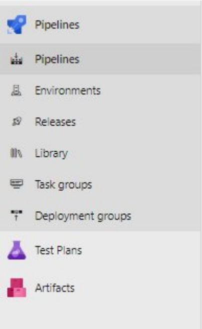
Banking Application Deployment using Jenkins, Ansible, Docker and Kubernetes

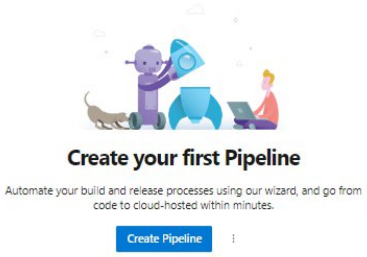
source code for the java application is provided by the simplilearn team. The Jenkinsfile, Kubernetes file, Ansible playbook, Dockerfile was built by me.

(Later we will delete all the files not needed ex. Jenkinsfile.)

2. Build the Java application

We wiill go to pipeline and create a new pipeline:

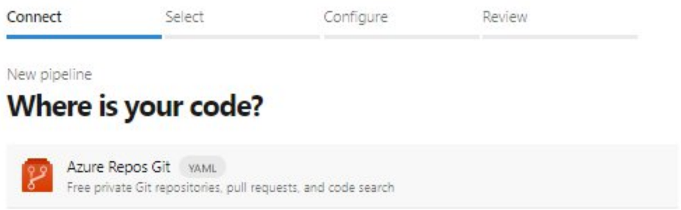




Create your first Pipeline

Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.

In the first step we will chose repo as Azure Repos Git and then our repo:

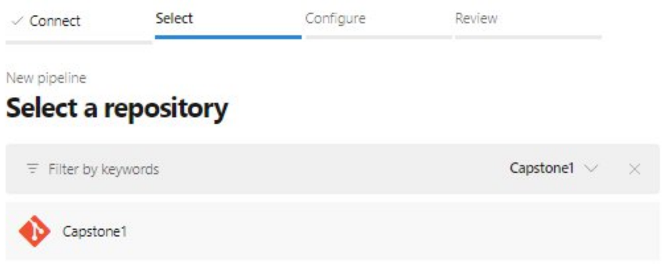


New pipeline

Where is your code?

Azure Repos Git YAML

Free private Git repositories, pull requests, and code search

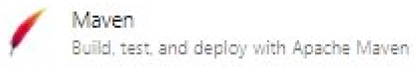


New pipeline

Select a repository

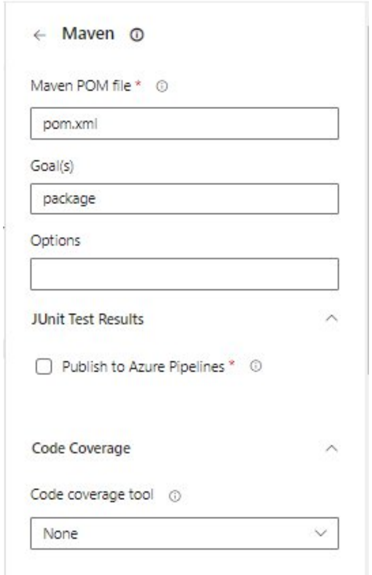
Capstone1

Then we will chose maven and only chose package:



Maven

Build, test, and deploy with Apache Maven



Maven POM file * ⓘ

pom.xml

Goal(s)

package

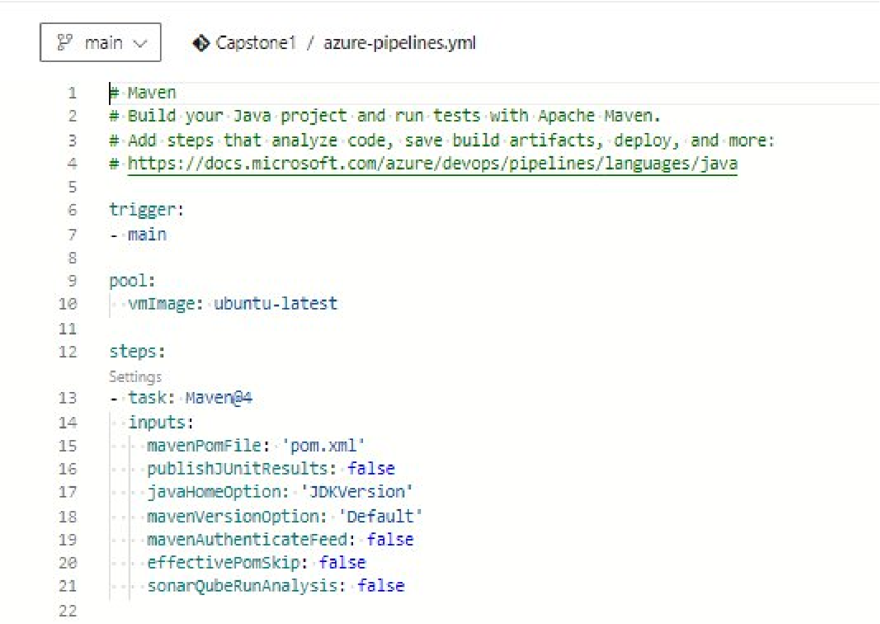
Options

☐ Publish to Azure Pipelines * ⓘ

Code coverage tool ⓘ

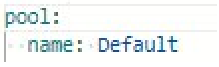
None

Then we will have our pipeline code till the build .



```yaml
# Maven
# Build your Java project and run tests with Apache Maven.
# Add steps that analyze code, save build artifacts, deploy, and more:
# https://docs.microsoft.com/azure/devops/pipelines/languages/java

trigger:
- main

pool:
  vmImage: ubuntu-latest

steps:
- task: Maven@4
  inputs:
    mavenPomFile: 'pom.xml'
    publishJUnitResults: false
    javaHomeOption: 'JDKVersion'
    mavenVersionOption: 'Default'
    mavenAuthenticateFeed: false
    effectivePomSkip: false
    sonarQubeRunAnalysis: false

```

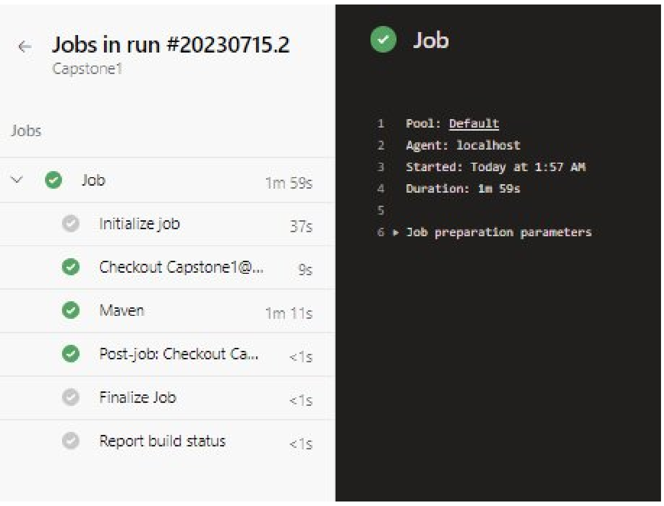
We will change our pool to our default pool as we are using our local machine to build(not yet authorized to use microsoft hosted agent)



```yaml
pool:
  name: Default
```

Then we will save and run.This will be saved in the azure repo.

We see that run is completed sucessfully



Jobs in run #20230715.2
Capstone1

Jobs

Job	1m 59s
Initialize job	37s
Checkout Capstone1@...	9s
Maven	1m 11s
Post-job: Checkout Ca...	<1s
Finalize Job	<1s
Report build status	<1s

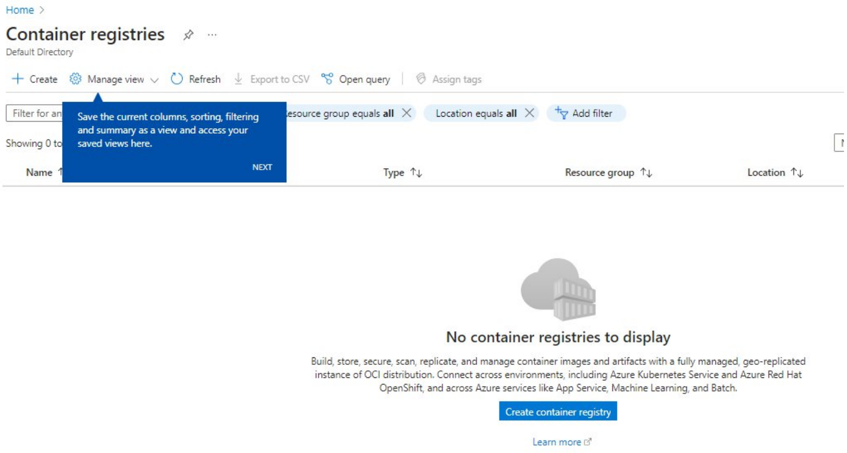
Job

```
Pool: Default
Agent: localhost
Started: Today at 1:57 AM
Duration: 1m 59s

▸ Job preparation parameters
```

3. **Build and push the image to azure container registry**

First let us create a container registry:

Go to container registries in azure portal:



Home >

Container registries

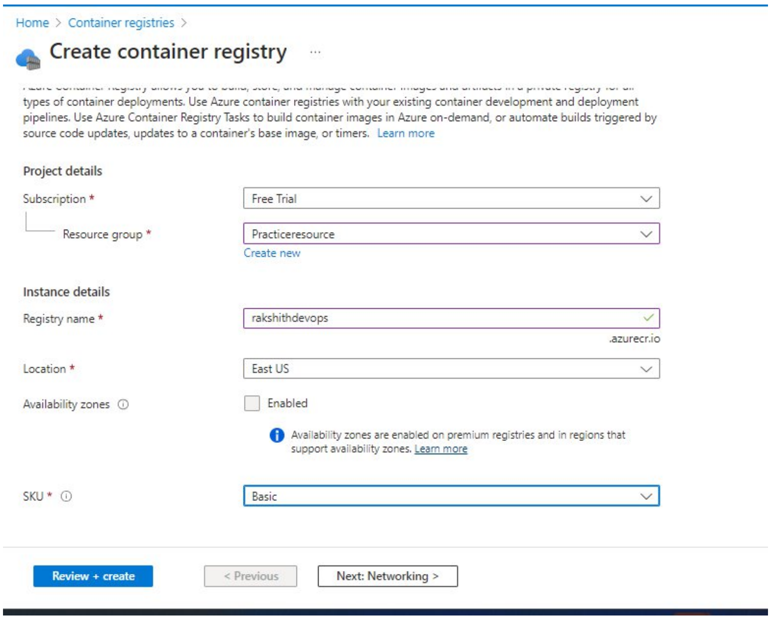
Default Directory

Save the current columns, sorting, filtering and summary as a view and access your saved views here.

No container registries to display

Build, store, secure, scan, replicate, and manage container images and artifacts with a fully managed, geo-replicated instance of OCI distribution. Connect across environments, including Azure Kubernetes Service and Azure Red Hat OpenShift, and across Azure services like App Service, Machine Learning, and Batch.

Click create.Give a name and other details and click review and create:



Home > Container registries >

Create container registry

types of container deployments. Use Azure container registries with your existing container development and deployment pipelines. Use Azure Container Registry Tasks to build container images in Azure on-demand, or automate builds triggered by source code updates, updates to a container's base image, or timers. Learn more

Project details

Subscription *: Free Trial

Resource group *: Practiceresource

Create new

Instance details

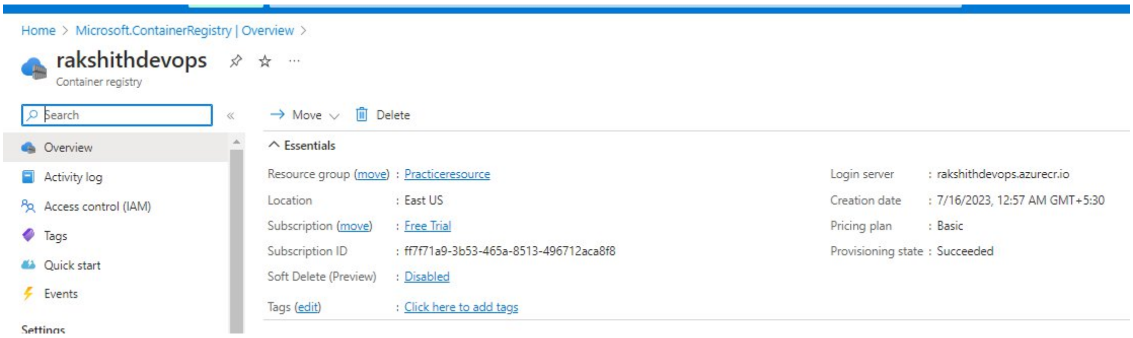
Registry name *: rakshithdevops .azurecr.io

Location *: East US

Availability zones: □ Enabled

Availability zones are enabled on premium registries and in regions that support availability zones. Learn more

SKU *: Basic



rakshithdevops
Container registry

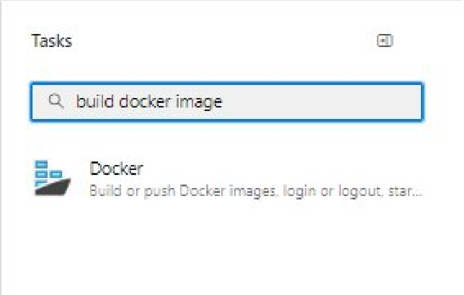
Essentials

Resource group (move) : Practiceresource
Location : East US
Subscription (move) : Free Trial
Subscription ID : ff7f71a9-3b53-465a-8513-496712aca8f8
Soft Delete (Preview) : Disabled
Tags (edit) : Click here to add tags

Login server : rakshithdevops.azurecr.io
Creation date : 7/16/2023, 12:57 AM GMT+5:30
Pricing plan : Basic
Provisioning state : Succeeded

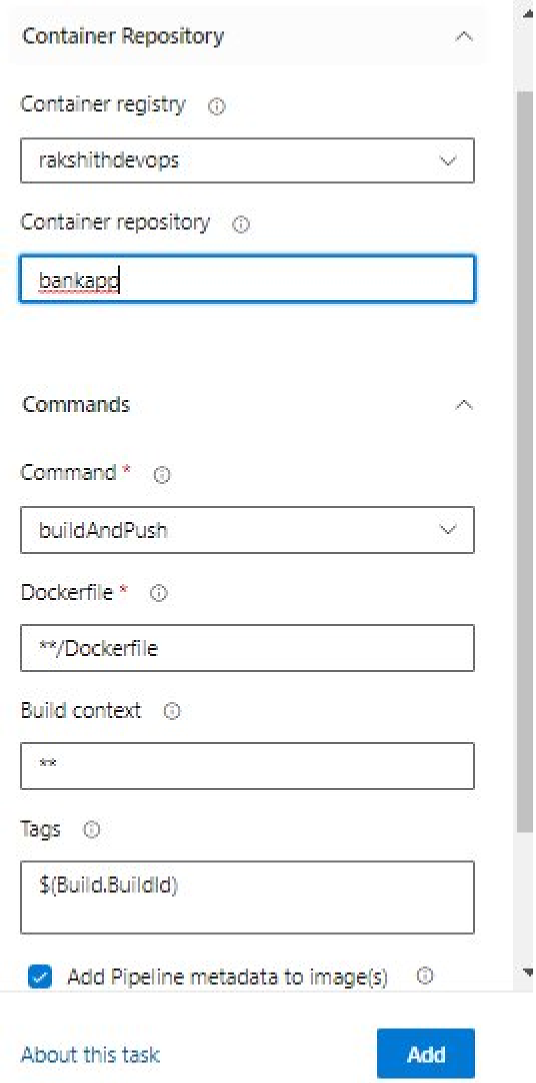
Now we will go the pipeline and add the build and push to containe registry steps:

Edit the pipeline and add the task:



build docker image

Docker
Build or push Docker images, login or logout, star...



Container registry: rakshithdevops

Container repository: bankapp

Command *: buildAndPush

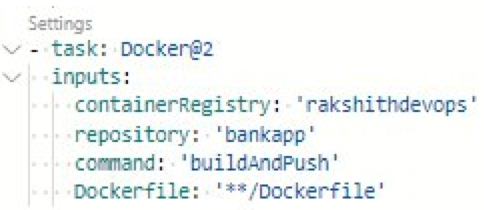
Dockerfile *: **/Dockerfile

Build context: **

Tags: $(Build.BuildId)

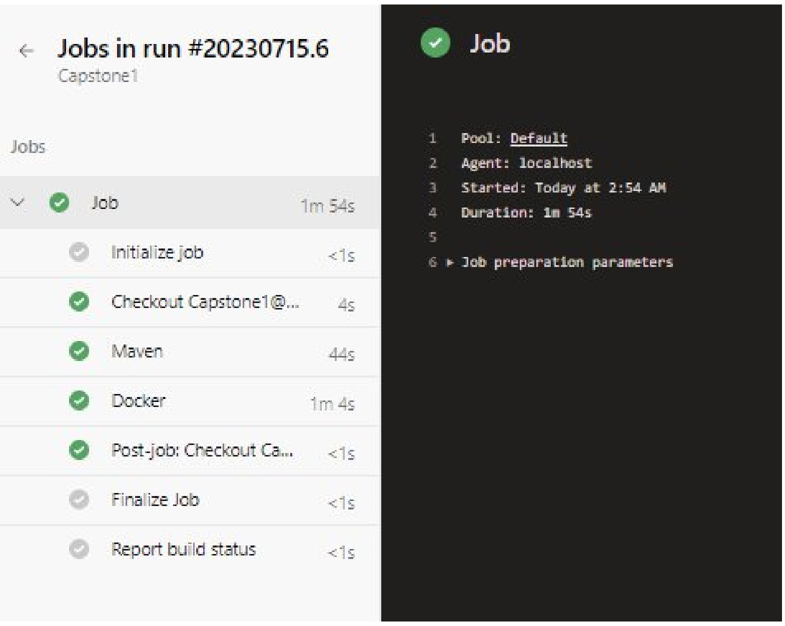
☑ Add Pipeline metadata to image(s)

We will get below code to add:



```
- task: Docker@2
  inputs:
    containerRegistry: 'rakshithdevops'
    repository: 'bankapp'
    command: 'buildAndPush'
    Dockerfile: '**/Dockerfile'
```

We had to install docker in the local system to run this, but once done then it runs successfully:



Jobs in run #20230715.6

Capstone1

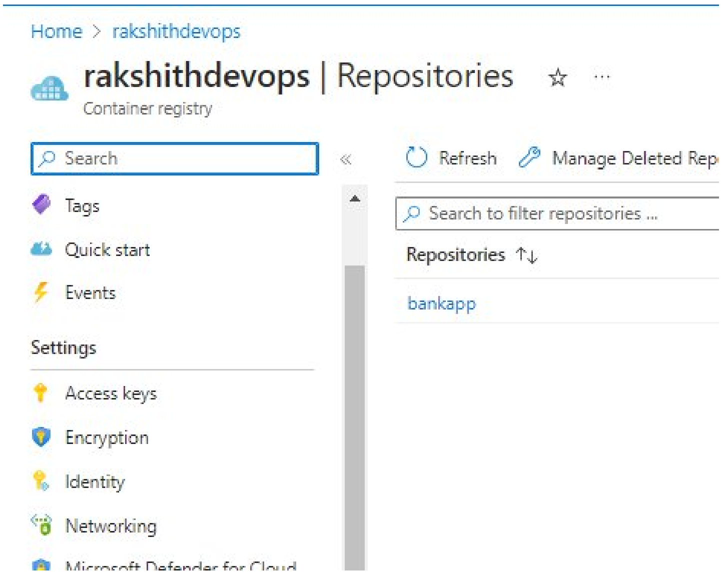
Jobs

Job	1m 54s
Initialize job	<1s
Checkout Capstone1@...	4s
Maven	44s
Docker	1m 4s
Post-job: Checkout Ca...	<1s
Finalize Job	<1s
Report build status	<1s

Job

```
Pool: Default
Agent: localhost
Started: Today at 2:54 AM
Duration: 1m 54s

▸ Job preparation parameters
```

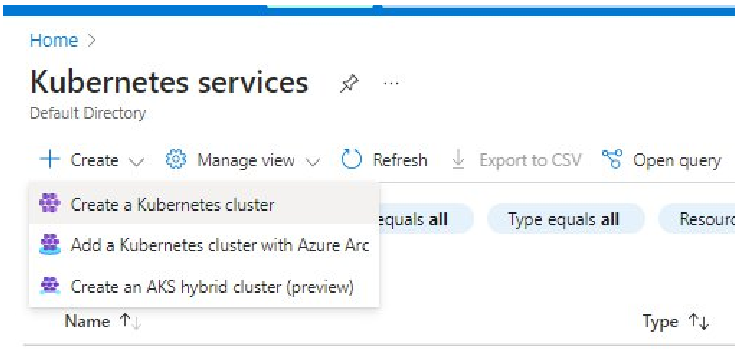
Then we can see in the azure container registries that docker image is pushed:



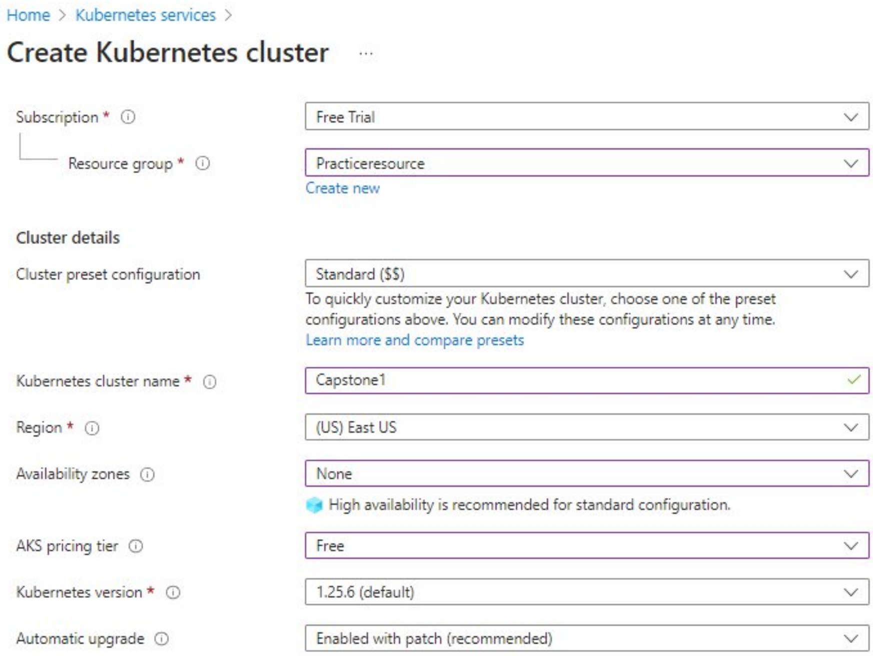
Repositories ↑↓
bankapp

4. Create Kubernetes cluster:

We will go to kubernetes services in Azure DevOps and click create kubernetes cluster:

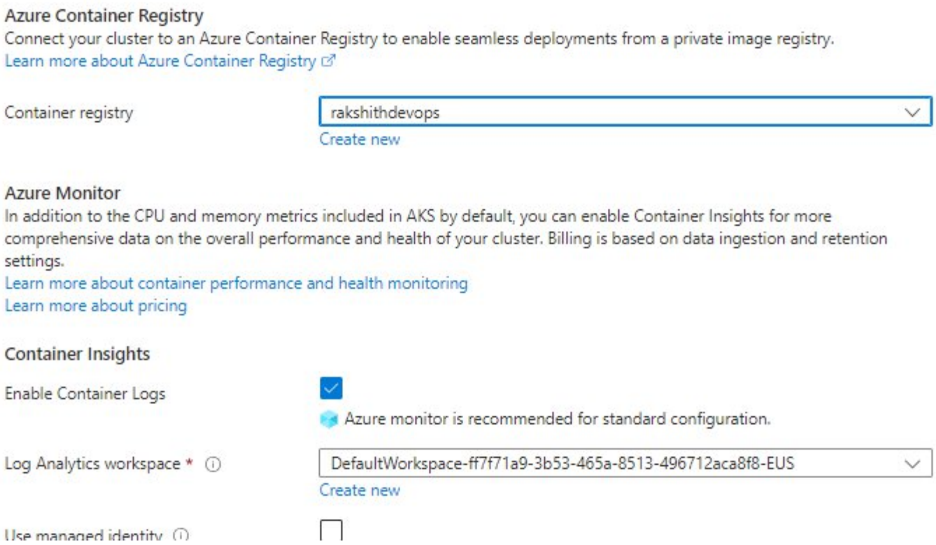


We will give the required details:



Subscription *	Free Trial
Resource group *	Practiceresource Create new
Cluster details	
Cluster preset configuration	Standard ($$) To quickly customize your Kubernetes cluster, choose one of the preset configurations above. You can modify these configurations at any time. Learn more and compare presets
Kubernetes cluster name *	Capstone1 ✓
Region *	(US) East US
Availability zones	None High availability is recommended for standard configuration.
AKS pricing tier	Free
Kubernetes version *	1.25.6 (default)
Automatic upgrade	Enabled with patch (recommended)

In integration settings we will integrate container registry and default log analytics workspace:



Azure Container Registry

Connect your cluster to an Azure Container Registry to enable seamless deployments from a private image registry.
Learn more about Azure Container Registry

Container registry	rakshithdevops Create new

Azure Monitor

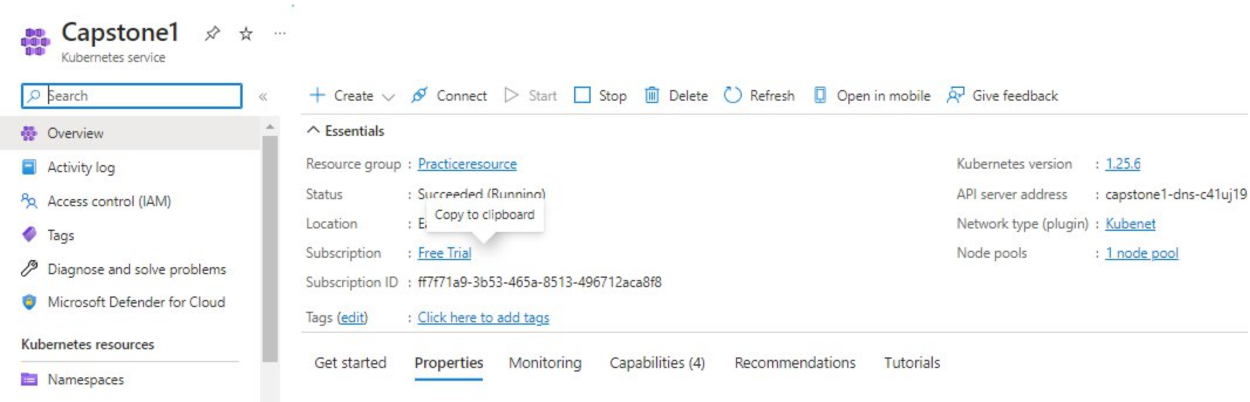
In addition to the CPU and memory metrics included in AKS by default, you can enable Container Insights for more comprehensive data on the overall performance and health of your cluster. Billing is based on data ingestion and retention settings.
Learn more about container performance and health monitoring
Learn more about pricing

Container Insights

Enable Container Logs	☑ Azure monitor is recommended for standard configuration.
Log Analytics workspace *	DefaultWorkspace-ff7f71a9-3b53-465a-8513-496712aca8f8-EUS Create new
Use managed identity	□

You can then review and create.

Once deployment is successful you should be able to see the cluster created



Resource group : Practiceresource

Status : Succeeded (Running)

Location : E

Subscription : Free Trial

Subscription ID : ff7f71a9-3b53-465a-8513-496712aca8f8

Tags (edit) : Click here to add tags

Kubernetes version : 1.25.6

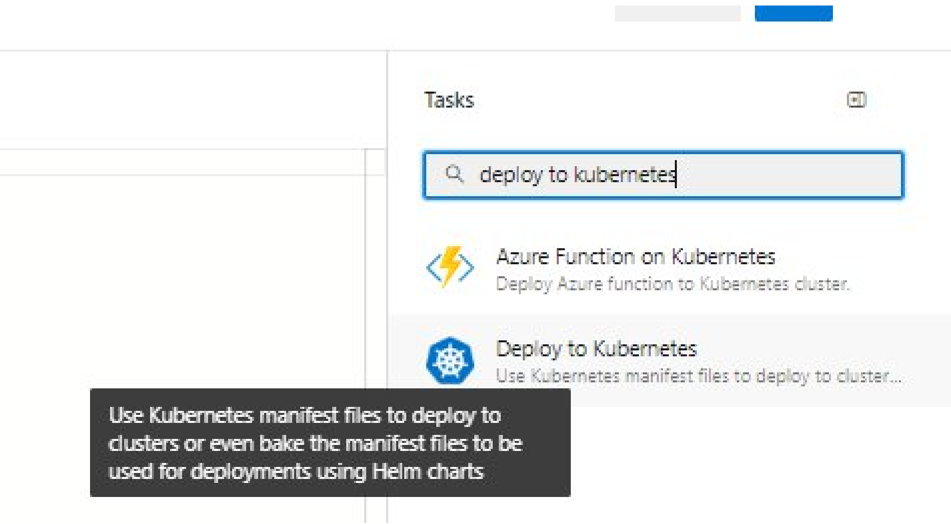
API server address : capstone1-dns-c41uj19

Network type (plugin) : Kubenet

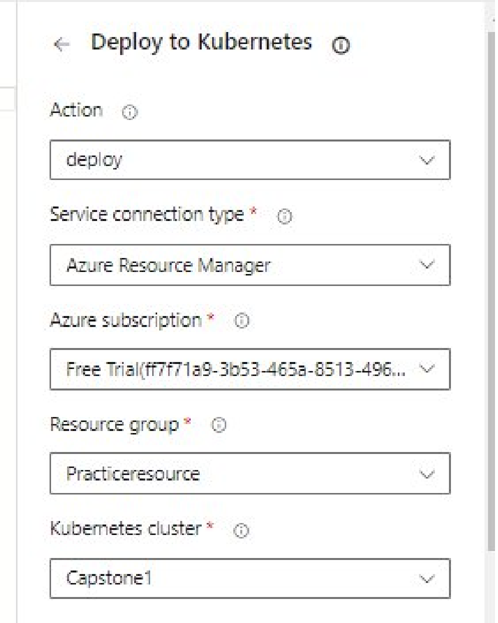
Node pools : 1 node pool

5. **Deploy the app to kubernetes:**

We will now edit the pipeline to add the kubernetes task:


Tasks
deploy to kubernetes
Azure Function on Kubernetes
Deploy Azure function to Kubernetes cluster.
Deploy to Kubernetes
Use Kubernetes manifest files to deploy to cluster...

Use Kubernetes manifest files to deploy to clusters or even bake the manifest files to be used for deployments using Helm charts



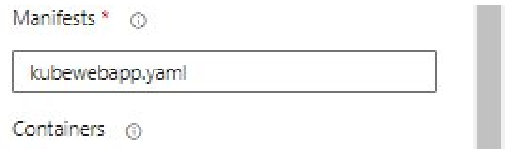
Action: deploy

Service connection type *: Azure Resource Manager

Azure subscription *: Free Trial(ff7f71a9-3b53-465a-8513-496...

Resource group *: Practiceresource

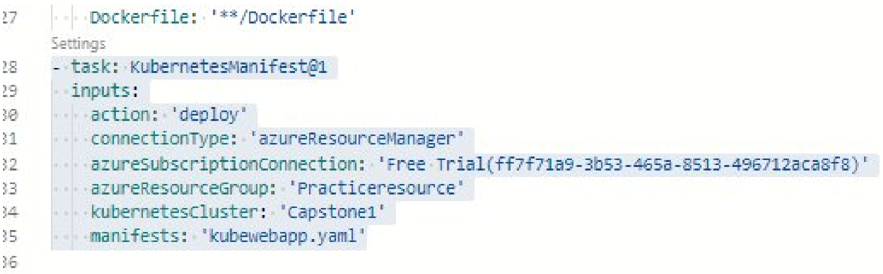
Kubernetes cluster *: Capstone1



Manifests *

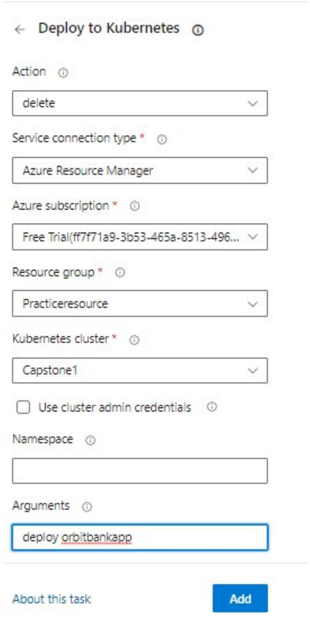
kubewebapp.yaml

Containers



```
    Dockerfile: '**/Dockerfile'
Settings
- task: KubernetesManifest@1
  inputs:
    action: 'deploy'
    connectionType: 'azureResourceManager'
    azureSubscriptionConnection: 'Free Trial(ff7f71a9-3b53-465a-8513-496712aca8f8)'
    azureResourceGroup: 'Practiceresource'
    kubernetesCluster: 'Capstone1'
    manifests: 'kubewebapp.yaml'
```

To make the pipeline reusable we will add delete action below this code:



Action: delete

Service connection type *: Azure Resource Manager

Azure subscription *: Free Trial(ff7f71a9-3b53-465a-8513-496...

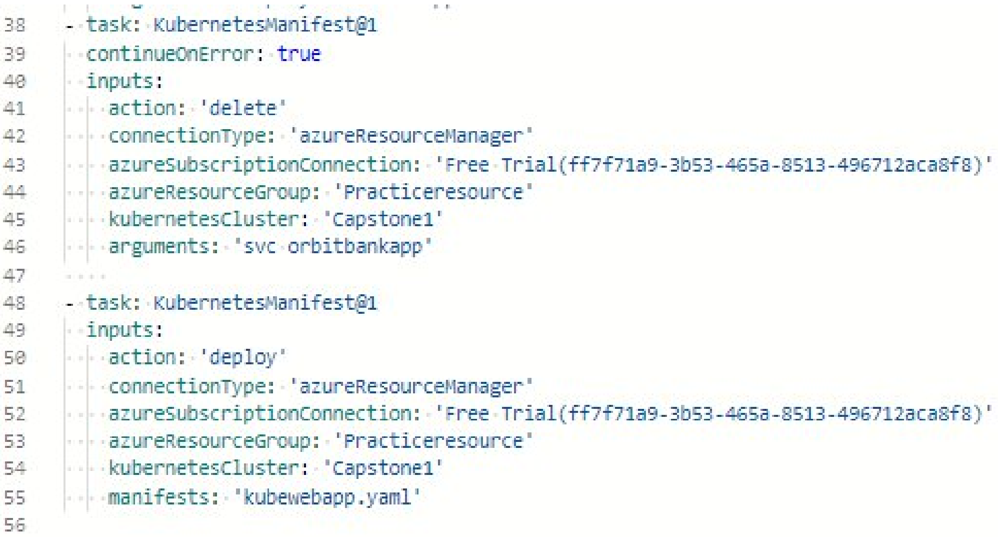
Resource group *: Practiceresource

Kubernetes cluster *: Capstone1

☐ Use cluster admin credentials

Namespace:

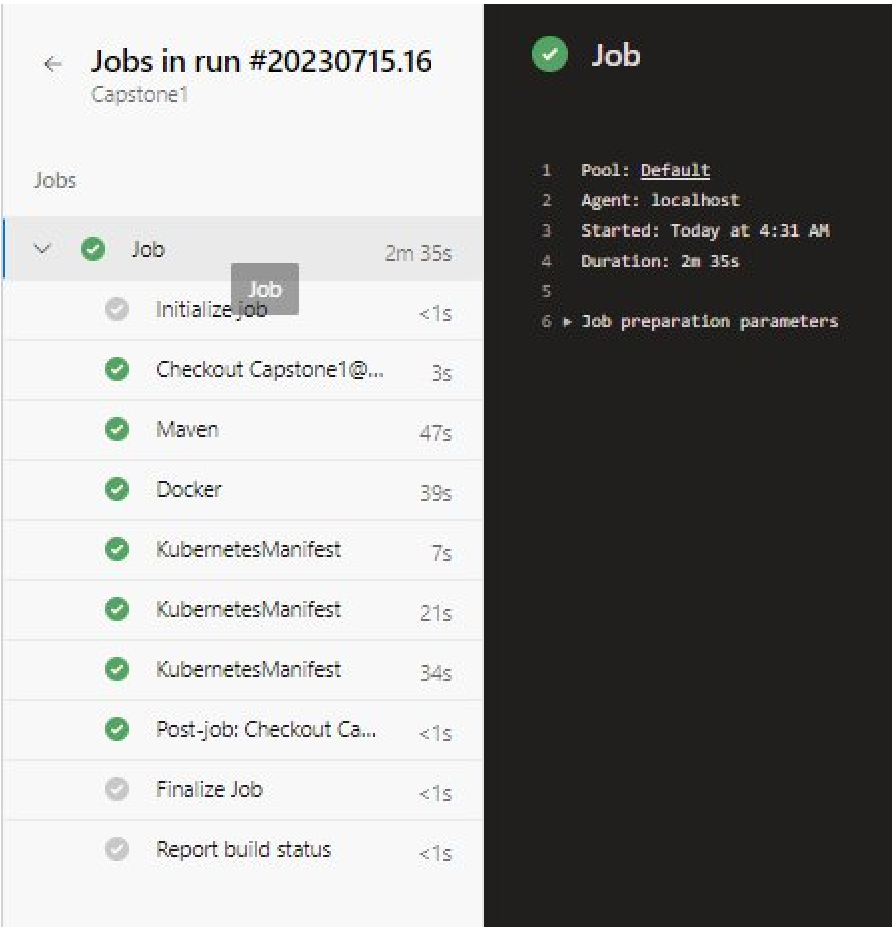
Arguments: deploy orbitbankapp



```
- task: KubernetesManifest@1
  continueOnError: true
  inputs:
    action: 'delete'
    connectionType: 'azureResourceManager'
    azureSubscriptionConnection: 'Free Trial(ff7f71a9-3b53-465a-8513-496712aca8f8)'
    azureResourceGroup: 'Practiceresource'
    kubernetesCluster: 'Capstone1'
    arguments: 'svc orbitbankapp'

- task: KubernetesManifest@1
  inputs:
    action: 'deploy'
    connectionType: 'azureResourceManager'
    azureSubscriptionConnection: 'Free Trial(ff7f71a9-3b53-465a-8513-496712aca8f8)'
    azureResourceGroup: 'Practiceresource'
    kubernetesCluster: 'Capstone1'
    manifests: 'kubewebapp.yaml'
```

We have added **continueOnError: true** so that it build can proceed even if this fails.(if there is no deployment/service it will fail)

After making few small changes in previous manifest file and the pipeline, our run was successful:



Jobs in run #20230715.16

Capstone1

Jobs

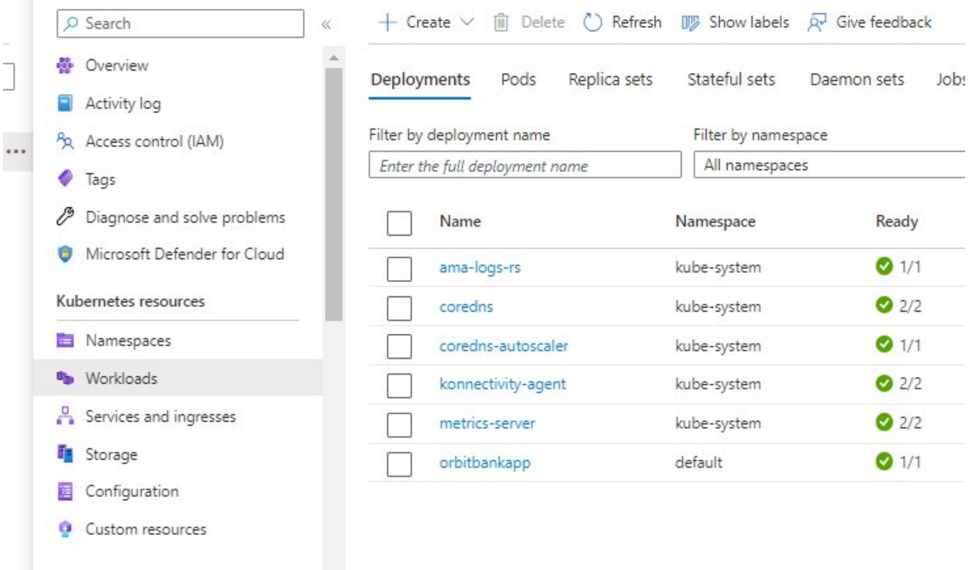
Job	2m 35s
Initialize job	<1s
Checkout Capstone1@...	3s
Maven	47s
Docker	39s
KubernetesManifest	7s
KubernetesManifest	21s
KubernetesManifest	34s
Post-job: Checkout Ca...	<1s
Finalize Job	<1s
Report build status	<1s

Job

```
Pool: Default
Agent: localhost
Started: Today at 4:31 AM
Duration: 2m 35s

▸ Job preparation parameters
```

6. Check the deployment and access the app:

In AKS, we can go to the workloads to see our deployment:



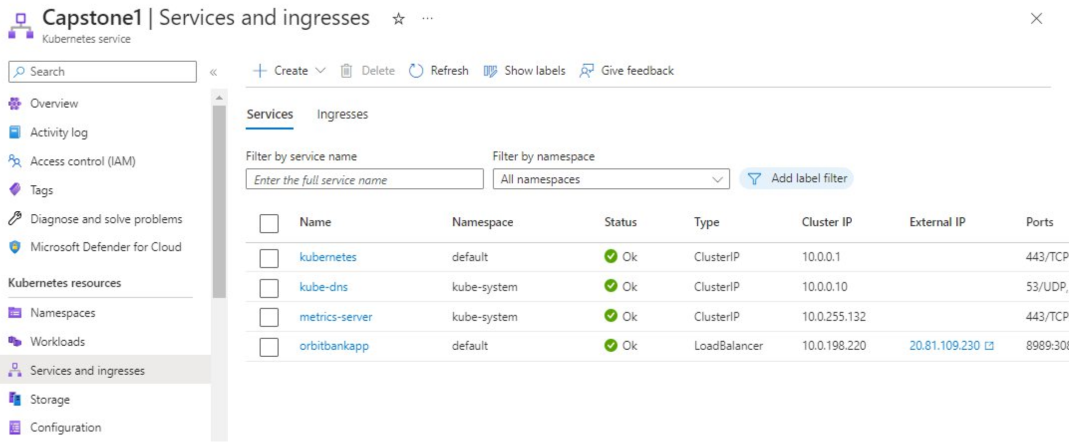
Filter by deployment name: Enter the full deployment name

Filter by namespace: All namespaces

□	Name	Namespace	Ready
□	ama-logs-rs	kube-system	✓ 1/1
□	coredns	kube-system	✓ 2/2
□	coredns-autoscaler	kube-system	✓ 1/1
□	konnectivity-agent	kube-system	✓ 2/2
□	metrics-server	kube-system	✓ 2/2
□	orbitbankapp	default	✓ 1/1

We can see that orbitbankapp is deployed.

In Services and ingresses, we can see that service is also deployed:



Filter by service name: Enter the full service name

Filter by namespace: All namespaces

	Name	Namespace	Status	Type	Cluster IP	External IP	Ports
□	kubernetes	default	✓ Ok	ClusterIP	10.0.0.1		443/TCP
□	kube-dns	kube-system	✓ Ok	ClusterIP	10.0.0.10		53/UDP,
□	metrics-server	kube-system	✓ Ok	ClusterIP	10.0.255.132		443/TCP
□	orbitbankapp	default	✓ Ok	LoadBalancer	10.0.198.220	20.81.109.230	8989:30

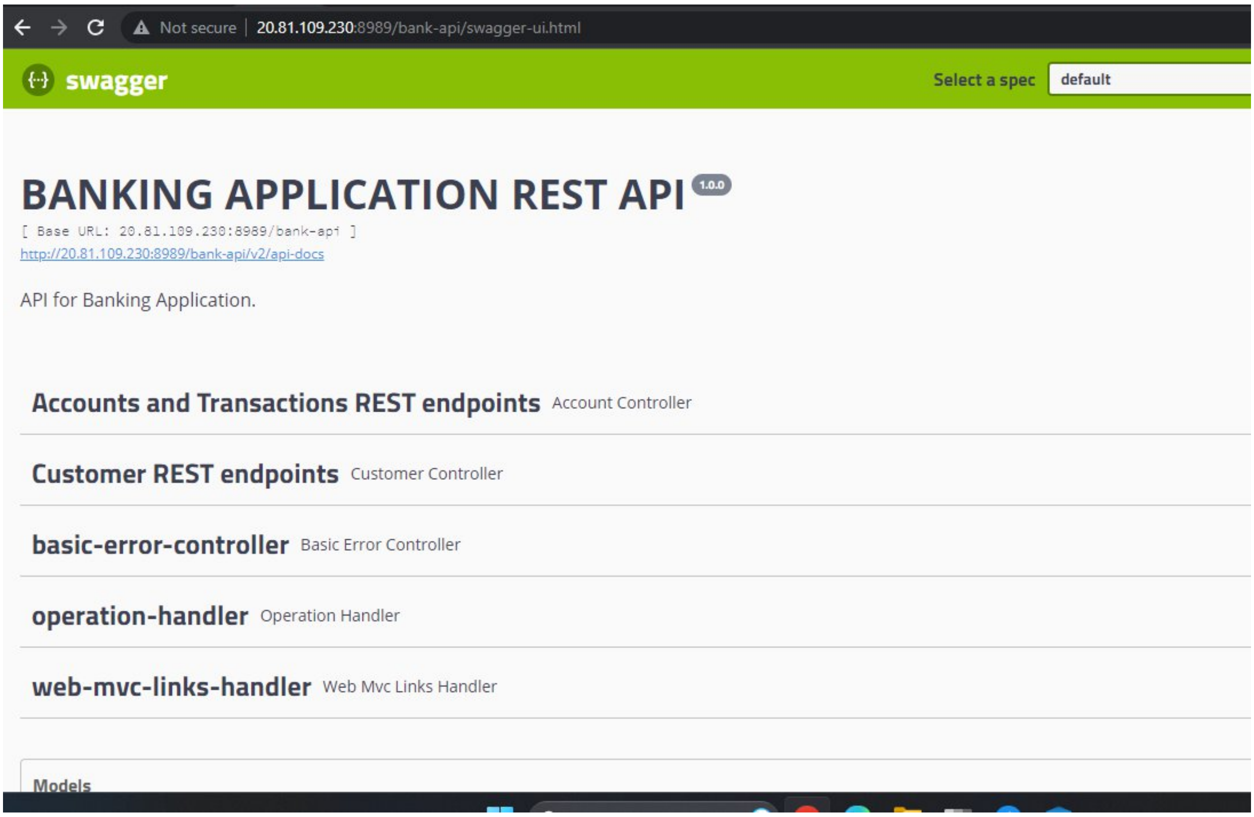
In this page we can see externalIP given.

We can access to required app page in

http://20.81.109.230:8989/bank-api/swagger-ui.html

(externalip:port/pagelink)

As you can see we are able to access:



20.81.109.230:8989/bank-api/swagger-ui.html

BANKING APPLICATION REST API 1.0.0

[Base URL: 20.81.109.230:8989/bank-api]

http://20.81.109.230:8989/bank-api/v2/api-docs

API for Banking Application.

Accounts and Transactions REST endpoints Account Controller

Customer REST endpoints Customer Controller

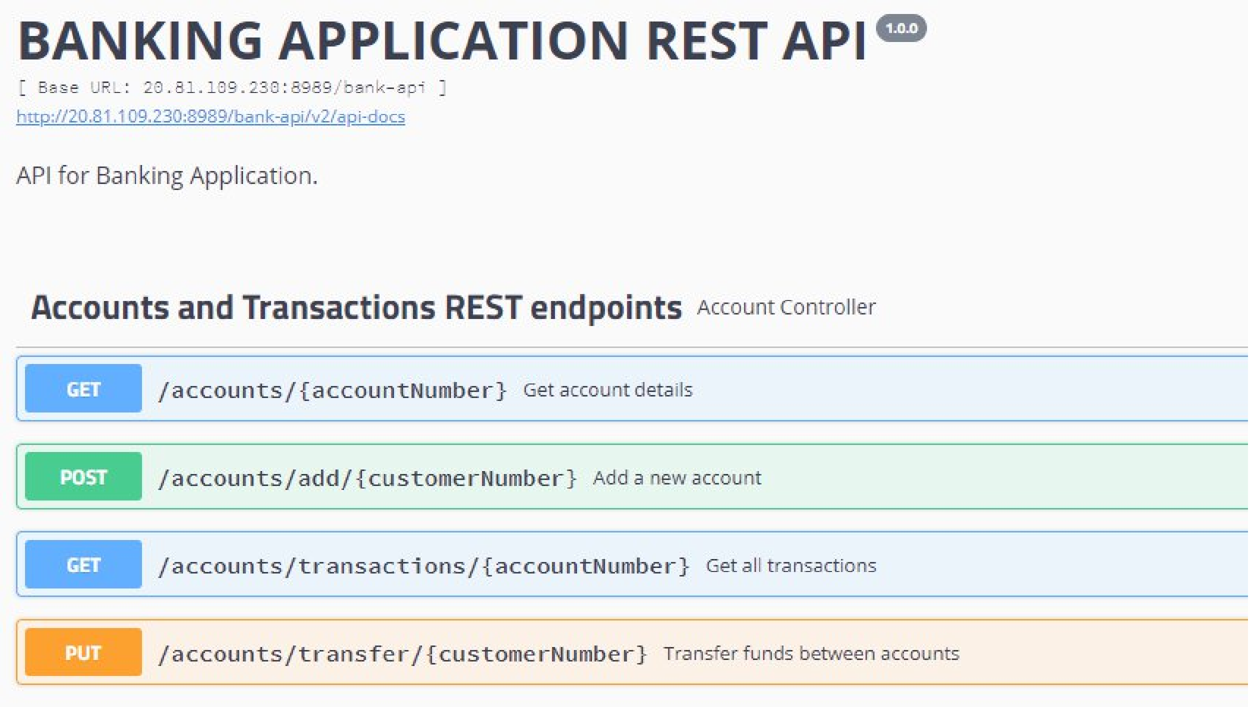
basic-error-controller Basic Error Controller

operation-handler Operation Handler

web-mvc-links-handler Web Mvc Links Handler

Models

And also navigate:



BANKING APPLICATION REST API 1.0.0

[Base URL: 20.81.109.230:8989/bank-api]

http://20.81.109.230:8989/bank-api/v2/api-docs

API for Banking Application.

Accounts and Transactions REST endpoints Account Controller

GET /accounts/{accountNumber} Get account details

POST /accounts/add/{customerNumber} Add a new account

GET /accounts/transactions/{accountNumber} Get all transactions

PUT /accounts/transfer/{customerNumber} Transfer funds between accounts

That completes this project. We have used the same code from capstone1 but we have used azure tools like Azure Devops, Azure Repos, Azure container registry, Azure Kubentes Service to achive the same results much easily.