

Project Development Phase
Delivery of Sprint-4

Date	24 November 2022
Team ID	PNT2022TMID25192
Project Name	Inventory Management System For Retailer

Deploying it on our Kubernetes services:

Step 1:

Open command prompt in the location where the project file is in.

Step 2:

Login to docker, ibmcloud and ibmcloud cr by using [docker login], [ibmcloud login] and [ibmcloud cr login] commands.

Step 3:

Build the image to docker hub.

Step 4:

Tag the docker image with our IBM container register namespaces.

Step 5:

Push to the namespaces of IBM container registry.

Step 6:

View the pushed image on IBM container registry.

Step 5:

Launch Kubernetes service and connect via CLI

Step 5:

Apply the yaml files using command prompt.

Step 5:

View the public IP and port number.

Step 5:

Go to the Public IP with the respected port number.

Output:

Login to the docker, ibmcloud & ibmcloud cr in Command Prompt:

```
Windows PowerShell
Email> mdyaseen.246@gmail.com

Password:
Authenticating...
OK

Targeted account Mohamed Yaseen's Account (b16c6751300c4cf5b9c374e8dd38ae18)
Targeted resource group Default
Targeted region eu-de

API endpoint: https://cloud.ibm.com
Region: eu-de
User: mdyaseen.246@gmail.com
Account: Mohamed Yaseen's Account (b16c6751300c4cf5b9c374e8dd38ae18)
Resource group: Default
CF API endpoint:
Org:
Space:
PS C:\Users\Yaseen\Desktop\Final Code> ibmcloud cr login
Logging 'docker' in to 'jp.icr.io'...
Logged in to 'jp.icr.io'.

OK
PS C:\Users\Yaseen\Desktop\Final Code> docker login
Authenticating with existing credentials...
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/access-tokens/
PS C:\Users\Yaseen\Desktop\Final Code>
```

Building, Tagging and Pushing the Image to Container Registry:

```
Windows PowerShell
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Yaseen\Desktop\Final Code> docker login
Authenticating with existing credentials...
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/access-tokens/
PS C:\Users\Yaseen\Desktop\Final Code> docker build -t yaseen-project .
[+] Building 78.2s (12/12) FINISHED
=> [internal] load build definition from Dockerfile 0.1s
=> => transferring dockerfile: 280B 0.0s
=> [internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> [internal] load metadata for docker.io/library/python:3.10.6 2.9s
=> [auth] library/python:pull token for registry-1.docker.io 0.0s
=> [1/6] FROM docker.io/library/python:3.10.6@sha256:745efd7b7e4aac9a8422bd8c62d8bc35a693e8979a240d29677cb03e6aa 0.0s
=> [internal] load build context 0.3s
=> => transferring context: 293.40kB 0.2s
=> CACHED [2/6] RUN apt-get update 0.0s
=> CACHED [3/6] RUN mkdir /app 0.0s
=> CACHED [4/6] WORKDIR /app 0.0s
=> [5/6] COPY . /app 1.0s
=> [6/6] RUN pip install -r requirements.txt 72.8s
=> exporting to image 0.9s
=> => exporting layers 0.9s
=> => writing image sha256:d3c7f8c2db08e4ef89eeab61cc3b946ac152e11036b5075e24835fc828a157b6 0.0s
=> => naming to docker.io/library/yaseen-project 0.0s
PS C:\Users\Yaseen\Desktop\Final Code> docker tag yaseen-project jp.icr.io/ns-ibm-project/yaseen-project
```

```
Windows PowerShell
PS C:\Users\Yaseen\Desktop\Final Code> docker push jp.icr.io/ns-ibm-project/yaseen-project
Using default tag: latest
The push refers to repository [jp.icr.io/ns-ibm-project/yaseen-project]
7392f00cc04c: Pushed
2184e185e727: Pushed
5f70bf18a086: Layer already exists
ad02e4ab118b: Layer already exists
8488b33b6249: Layer already exists
bfc1deb8136e: Layer already exists
1f123186824c: Layer already exists
3d6eb1152931: Layer already exists
100796cdf3b1: Layer already exists
54acb5a6fa0b: Layer already exists
8d51c618126f: Layer already exists
9ff6e4d46744: Layer already exists
a89d1d47b5a1: Layer already exists
655ed1b7a428: Layer already exists
latest: digest: sha256:3c076e146cc01276444570c16fac22939f5027ae2e4c8346edeaf27298bec7eb size: 3266
PS C:\Users\Yaseen\Desktop\Final Code> |
```

Viewing the Pushed Image on Container Registry:

IBM Cloud

Search resources and products...

Catalog

Manage

Mohamed Yaseen's Account

Container Registry

Quick start

Namespaces 1

Repositories 2

Images 2

Trash 7

Settings

Images

Location

Tokyo

View by: Digest

Search

Create

Repository@digest	Tags	Manifest type	Created	Size	Security status
ns-ibm-project/yaseen-project@sha256:3c076e146cc0...	latest	Docker	10 minutes ago	466 MB	26 Issues
ns-ibm-project/ibmproject@sha256:2adfde7c4b7a...	1-master-d8603b42-20221122120836	Docker	1 day ago	376 MB	9 Issues

Items per page: 25 1-2 of 2 items

1 1 of 1 page

Launch Kubernetes Service and Connect Via CLI:

The screenshot displays the IBM Cloud console interface for a Kubernetes cluster named 'mycluster-free'. The cluster is in a 'Normal' state and is scheduled to expire in 29 days. The overview section shows the following details:

- Node status:** 1 of 1 nodes are Normal.
- Add-on status:** 0 of 0 add-ons are Normal.
- Master status:** Normal.
- Ingress status:** Healthy.

The details section provides the following information:

- Cluster ID:** cdub908f0um7dbgfn50
- Version:** 1.24.8_1544
- Infrastructure:** Classic
- Zones:** Milan 01
- Created:** 11/22/2022, 5:07 PM
- Resource group:** Default
- Image security enforcement:** Enable

The node health section shows 1 total node, with a 100% Normal status (0% Critical, 0% Warning, 0% Pending).

The screenshot shows the 'Connect via CLI' dialog box overlaid on the cluster overview. The dialog provides instructions for connecting to the cluster via the command line interface (CLI).

Cluster status: Normal

If this is your first time connecting to an IBM Cloud cluster, see the [full setup directions](#).

- Log in to your IBM Cloud account. Include the `--sso` option if using a federated ID.
`ibmcloud login -a cloud.ibm.com -x eu-de -g Default`
- Set the Kubernetes context to your cluster for this terminal session. For more information about this command, see the docs.
`ibmcloud ks cluster config --cluster cdub908f0um7dbgfn50`
- Verify that you can connect to your cluster.
`kubectl config current-context`

Now, you can run `kubectl` commands to manage your cluster workloads in IBM Cloud! For a full list of commands, see the [Kubernetes docs](#).

Tip: Plan to use multiple clusters? Repeat these steps for each cluster. Then, you can use the `kubectl config use-context` command to switch your context to a different cluster.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Vaseen\Desktop\Final Code> ibmcloud login -a cloud.ibm.com -r eu-de -g Default
API endpoint: https://cloud.ibm.com

Email> mdyaseen.246@gmail.com

Password>
Authenticating...
OK

Targeted account Mohamed Yaseen's Account (b16c6751390c4cf5b9c374e8dd38ae18)

Targeted resource group Default

Targeted region eu-de

API endpoint: https://cloud.ibm.com
Region: eu-de
User: mdyaseen.246@gmail.com
Account: Mohamed Yaseen's Account (b16c6751390c4cf5b9c374e8dd38ae18)
Resource group: Default
CF API endpoint:

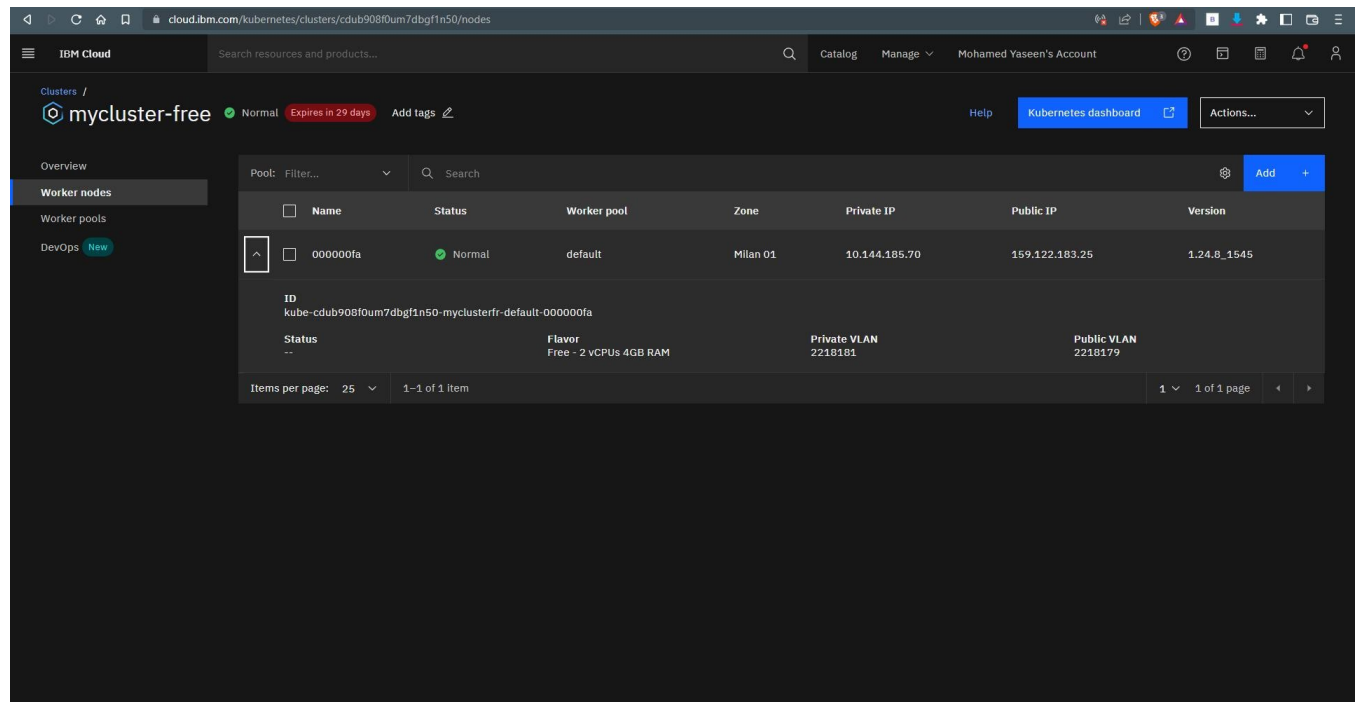
Org:
Space:
PS C:\Users\Vaseen\Desktop\Final Code> ibmcloud ks cluster config --cluster cduub908f0um7dbgf1n50
OK
The configuration for cduub908f0um7dbgf1n50 was downloaded successfully.

Added context for cduub908f0um7dbgf1n50 to the current kubeconfig file.
You can now execute 'kubectl' commands against your cluster. For example, run 'kubectl get nodes'.
If you are accessing the cluster for the first time, 'kubectl' commands might fail for a few seconds while RBAC synchronizes.
PS C:\Users\Vaseen\Desktop\Final Code> kubectl config current-context
mycluster-free/cduub908f0um7dbgf1n50
PS C:\Users\Vaseen\Desktop\Final Code> |
```

Apply the yaml Files Using Command Prompt:

```
Windows PowerShell X + ~  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code>  
PS C:\Users\Yaseen\Desktop\Final Code> kubectl run yaseen-project --image=jp.icr.io/ns-ibm-project/yaseen-project  
pod/yaseen-project created  
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\deployment.yaml  
deployment.apps/yaseen-project created  
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\service.yaml  
service/project-service unchanged  
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\kubernetes\flask_ingress.yaml  
ingress.networking.k8s.io/flask-app-ingress unchanged  
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\kubernetes\flask_service.yaml  
service/mycluster-free-service unchanged  
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\kubernetes\ibm_deployment.yaml  
deployment.apps/mycluster-free created  
PS C:\Users\Yaseen\Desktop\Final Code>
```

View the Public IP and Port Number:



The screenshot shows the IBM Cloud Kubernetes dashboard for a cluster named 'mycluster-free'. The 'Worker nodes' tab is active, displaying a table with one node. The node's details are expanded, showing its ID, status, flavor, and IP addresses.

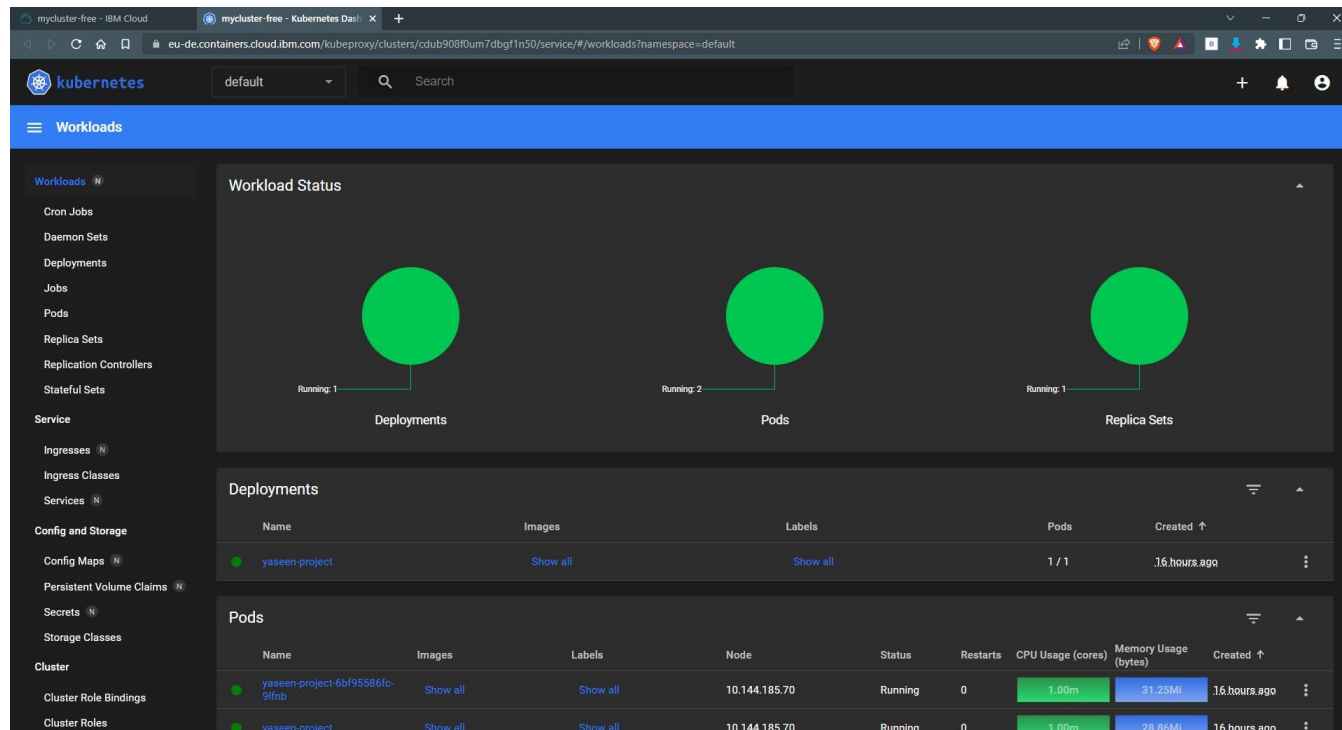
Name	Status	Worker pool	Zone	Private IP	Public IP	Version
000000fa	Normal	default	Milan 01	10.144.185.70	159.122.183.25	1.24.8_1545

Node details:

- ID: kube-cdub908f0um7dbg1n50-myclusterfr-default-000000fa
- Status: --
- Flavor: Free - 2 vCPUs 4GB RAM
- Private VLAN: 2218181
- Public VLAN: 2218179

```
Windows PowerShell
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code>
PS C:\Users\Yaseen\Desktop\Final Code> kubectl run yaseen-project --image=jp.icr.io/ns-ibm-project/yaseen-project
pod/yaseen-project created
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\deployment.yaml
deployment.apps/yaseen-project created
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\service.yaml
service/project-service unchanged
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\kubernetes\flask_ingress.yaml
ingress.networking.k8s.io/flask-app-ingress unchanged
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\kubernetes\flask_service.yaml
service/mycluster-free-service unchanged
PS C:\Users\Yaseen\Desktop\Final Code> kubectl apply -f .\kubernetes\ibm_deployment.yaml
deployment.apps/mycluster-free created
PS C:\Users\Yaseen\Desktop\Final Code> kubectl get svc
NAME                                TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
flask-app-service                   NodePort      172.21.169.184  <none>           5000:32672/TCP   24h
flask-service                       NodePort      172.21.250.81   <none>           5000:32015/TCP   22h
ibmproject                          NodePort      172.21.91.42    <none>           3000:30960/TCP   26h
kubernetes                          ClusterIP     172.21.0.1      <none>           443/TCP          26h
mycluster-free-service              NodePort      172.21.163.161  <none>           5000:31134/TCP   24h
project-service                     ClusterIP     172.21.55.51    <none>           5000/TCP          7h31m
yaseen-project-service              ClusterIP     172.21.65.168   <none>           5000/TCP          24h
PS C:\Users\Yaseen\Desktop\Final Code> |
```


Kubernetes Dashboard:



Visit the Public IP and Port Number of Project:

The screenshot shows the 'Inventory' website. The header includes the 'Inventory' logo and a 'signin' button. The main content area is titled 'Inventory management for Retailers' and contains the text: 'Manage orders. Track inventory. Handle GST billing. One inventory management software to run all your inventory operations.' Below this text is a 'Get started' button. On the left side, there is an illustration of a person sitting at a desk with a laptop, looking at a smartphone. On the right side, there is a chat bubble with the text: 'Hi! I'm a virtual assistant. How can I help you today?'.