

Name: Ruthik Jadhav
Roll No: 321087
GR No.: 22120243
Division and Batch: A3

Assignment No. 8

Aim: Deploy a web app on Kubernetes Cluster.

Theory:

What is Kubernetes?

With the widespread adoption of containers among organizations, Kubernetes, the container-centric management software, has become the de facto standard to deploy and operate containerized applications. Google Cloud is the birthplace of Kubernetes—originally developed at Google and released as open source in 2014. Kubernetes builds on 15 years of running Google's containerized workloads and the valuable contributions from the open source community. Inspired by Google's internal cluster management system, Borg, Kubernetes makes everything associated with deploying and managing your application easier. Providing automated container orchestration, Kubernetes improves your reliability and reduces the time and resources attributed to daily operations.

What does Kubernetes do?

Kubernetes schedules and automates container-related tasks throughout the application lifecycle, including:

- Deployment: Deploy a specified number of containers to a specified host and keep them running in a desired state.
- Rollouts: A rollout is a change to a deployment. Kubernetes lets you initiate, pause, resume, or roll back rollouts.
- Service discovery: Kubernetes can automatically expose a container to the internet or to other containers using a DNS name or IP address.
- Storage provisioning: Set Kubernetes to mount persistent local or cloud storage for your containers as needed.
- Load balancing: Based on CPU utilization or custom metrics, Kubernetes load balancing can distribute the workload across the network to maintain performance and stability.

- Autoscaling: When traffic spikes, Kubernetes autoscaling can spin up new clusters as needed to handle the additional workload.
- Self-healing for high availability: When a container fails, Kubernetes can restart or replace it automatically to prevent downtime. It can also take down containers that don't meet your health-check requirements.

What are the benefits of Kubernetes?

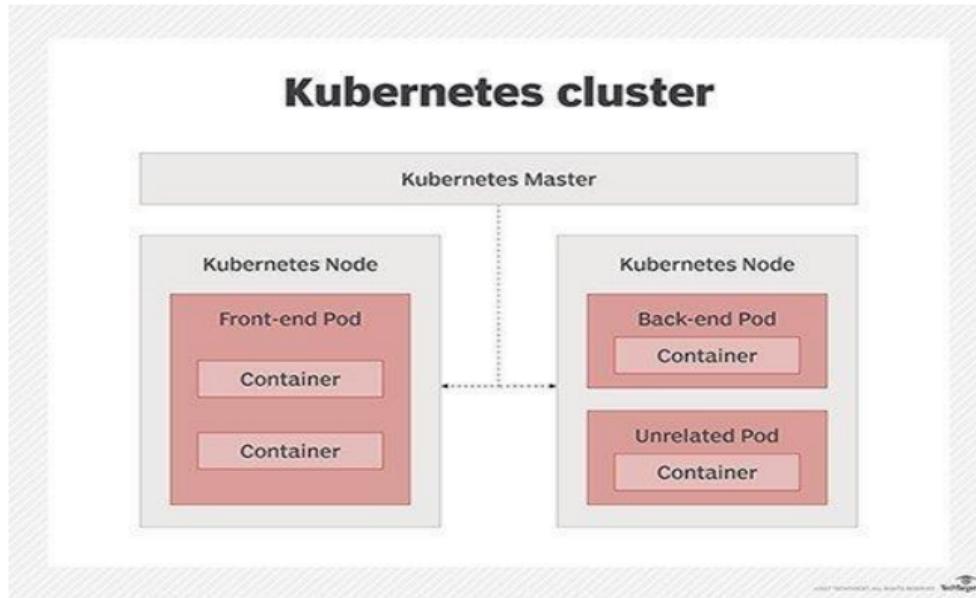
- Automated operations: Kubernetes has built-in commands to handle a lot of the heavy lifting that goes into application management, allowing you to automate day-to-day operations. You can make sure applications are always running the way you intended them to run.
- Infrastructure abstraction: When you install Kubernetes, it handles the compute, networking, and storage on behalf of your workloads. This allows developers to focus on applications and not worry about the underlying environment.
- Service health monitoring: Kubernetes continuously runs health checks against your services, restarting containers that fail, or have stalled, and only making available services to users when it has confirmed they are running.
- Kubernetes architecture: The chief components of Kubernetes architecture include the following:

Clusters and nodes (compute)

Clusters are the building blocks of Kubernetes architecture. The clusters are made up of nodes, each of which represents a single compute host (virtual or physical machine).

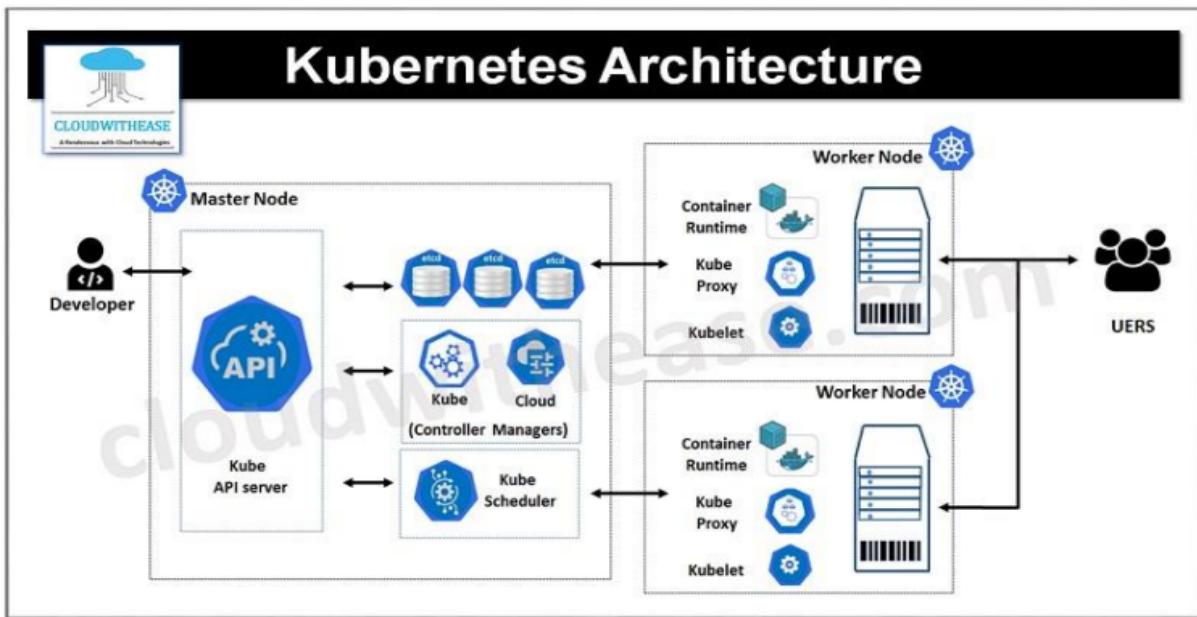
Each cluster consists of a master node that serves as the control plan for the cluster, and multiple worker nodes that deploy, run, and manage containerized applications. The master node runs a scheduler service that automates when and where the containers are deployed based on developer set deployment requirements and available computing capacity. Each worker node includes the tool that is being used to manage the containers — such as Docker — and a software agent called a Kubelet that receives and executes orders from the master node.

Developers manage cluster operations using kubectl, a command-line interface (cli) that communicates directly with the Kubernetes API.



Pods and deployments (software)

Pods are groups of containers that share the same compute resources and the same network. They are also the unit of scalability in Kubernetes: if a container in a pod is getting more traffic than it can handle, Kubernetes will replicate the pod to other nodes in the cluster. For this reason, it's a good practice to keep pods compact so that they contain only containers that must share resources. The deployment controls the creation and state of the containerized application and keeps it running. It specifies how many replicas of a pod should run on the cluster. If a pod fails, the deployment will create a new one.



IMPLEMENTATION

Step 1 – Install minikube on your local system.

Go to <https://minikube.sigs.k8s.io/docs/start/> and click on the latest release to download the installer or you can copy the below commands and paste it into your powershell to install minikube.

The screenshot shows a browser window with the URL <https://minikube.sigs.k8s.io/docs/start/>. The page displays the 'Installation' section, which includes configuration options for operating system (Windows selected), architecture (x86-64 selected), release type (Stable selected), and installer type (.exe download selected). Below these options, instructions are provided for installing the latest minikube stable release on x86-64 Windows using .exe download. It includes a PowerShell command for creating a directory and downloading the executable, and a note to add the binary to the PATH and run PowerShell as Administrator. The browser's sidebar shows a navigation menu with links like Documentation, Get Started!, Handbook, Addons, Configuration, Dashboard, Pushing images, Proxies and VPNs, Registries, Certificates, Offline usage, Host access, Network Policy, Persistent Volumes, and more.

The screenshot also shows a Windows taskbar at the bottom with the date and time (27-04-2023, 13:16). Below the taskbar is a Windows PowerShell window titled 'Administrator: Windows Pow'. The PowerShell session shows the user running a command to download the minikube executable to the C:\ directory. The command is:

```
New-Item -Path 'c:' -Name 'minikube' -ItemType Directory -Force  
Invoke-WebRequest -OutFile 'c:\minikube\minikube.exe' -Uri 'https://github.com/kubernetes/minikube/releases/latest/dc'
```

After running the command, the PowerShell window shows the directory contents:

Mode	LastWriteTime	Length	Name
d----	27-04-2023 22:02		minikube

Finally, the PowerShell command to invoke the minikube executable is shown:

```
PS C:\Users\Ruthik> Invoke-WebRequest -OutFile 'c:\minikube\minikube.exe' -Uri 'https://github.com/kubernetes/minikube/releases/latest/download/minikube-windows-amd64.exe' -UseBasicParsing
```



```

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

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

PS C:\WINDOWS\system32> $oldPath = [Environment]::GetEnvironmentVariable('Path', [EnvironmentVariableTarget]::Machine)
>> if ($oldPath.Split(';') -inotcontains 'C:\minikube'){
>>   [Environment]::SetEnvironmentVariable('Path', ${'${0};C:\minikube'} -f $oldPath), [EnvironmentVariableTarget]::Machine)
>> }
>>
PS C:\WINDOWS\system32>

```

Step 2 – After downloading minikube run the following commands on your terminal.

Login onto your docker



```

Administrator: Windows PowerShell
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\Ruthik> 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\Ruthik> |

```

and run the docker image and Check for docker images.



```

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

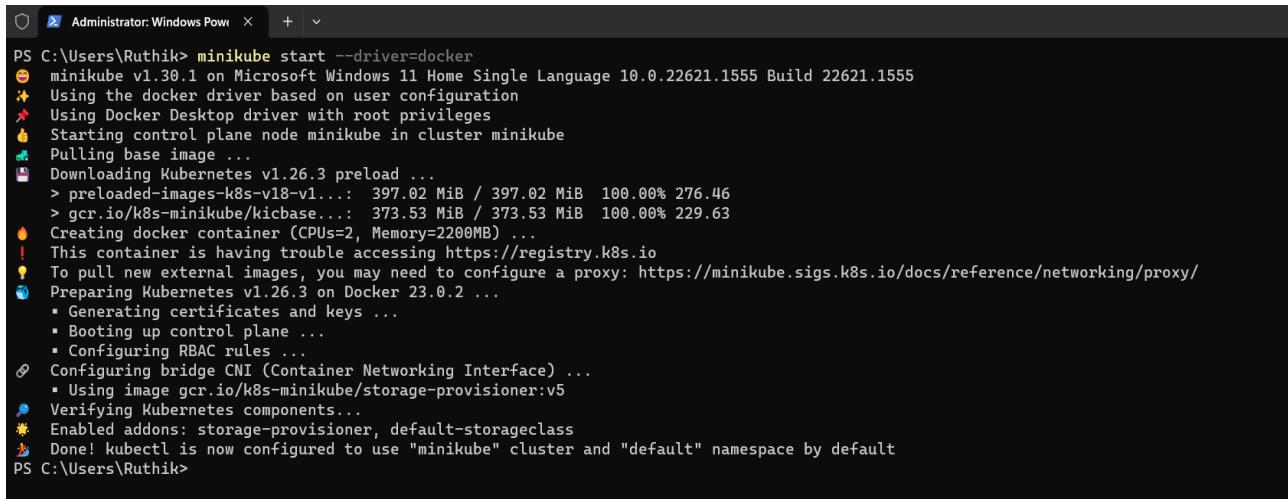
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Ruthik> docker run -d -p 84:80 myweb
acd8e304a3fb52fffb2ba553ecae51305daebb1348fffc99bdd0b95e013d14b4
PS C:\Users\Ruthik> |

PS C:\Users\Ruthik> docker tag myweb ruthikjadhav/myweb
PS C:\Users\Ruthik> docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
myweb              latest   0772b387edc5  22 hours ago  42.5MB
ruthikjadhav/myweb    latest   0772b387edc5  22 hours ago  42.5MB
food-image         v3       65f79b1d6474  23 hours ago  41.3MB
myapp              latest   65f79b1d6474  23 hours ago  41.3MB
pro-image          v2       407cc32fa1d9  23 hours ago  43.6MB
webserver-image     v1       0f571064eac1  36 hours ago  41MB
<none>            <none>  e98adcabd11  36 hours ago  41MB
<none>            <none>  af9075828cae  36 hours ago  41MB
gcr.io/k8s-minikube/kicbase  v0.0.39  67a4b1138d2d  3 weeks ago  1.05GB
PS C:\Users\Ruthik> docker push ruthikjadhav/myweb
Using default tag: latest
The push refers to repository [docker.io/ruthikjadhav/myweb]
470cb1d28511: Pushed
31531248c7cb: Mounted from library/nginx
f9cb3f1f1d3d: Mounted from library/nginx
f0fb842dea41: Mounted from library/nginx
c1cd5c8c68ef: Mounted from library/nginx
1d54586a1706: Mounted from library/nginx
1003ff723696: Mounted from library/nginx
f1417ff83b31: Mounted from library/nginx
latest: digest: sha256:83a9582e4cf3b3e9665080f88527c9095c9ec8aa48faf9c74fd559c6b534c3e2 size: 1992

```

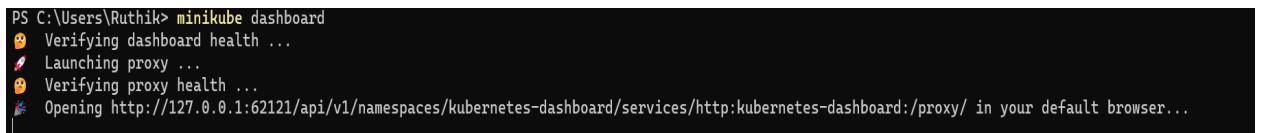
Then run minikube start --driver=docker command



```
Administrator: Windows PowerShell C:\Users\Ruthik> minikube start --driver=docker
minikube v1.30.1 on Microsoft Windows 11 Home Single Language 10.0.22621.1555 Build 22621.1555
* Using the docker driver based on user configuration
* Using Docker Desktop driver with root privileges
Starting control plane node minikube in cluster minikube
Pulling base image ...
Downloading Kubernetes v1.26.3 preload ...
> preloaded-images-k8s-v18-v1...: 397.02 MiB / 397.02 MiB 100.00% 276.46
> gcr.io/k8s-minikube/kicbase...: 373.53 MiB / 373.53 MiB 100.00% 229.63
Creating docker container (CPUs=2, Memory=2200MB) ...
This container is having trouble accessing https://registry.k8s.io
To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking/proxy/
Preparing Kubernetes v1.26.3 on Docker 23.0.2 ...
  • Generating certificates and keys ...
  • Booting up control plane ...
  • Configuring RBAC rules ...
Configuring bridge CNI (Container Networking Interface) ...
  • Using image gcr.io/k8s-minikube/storage-provisioner:v5
Verifying Kubernetes components...
Enabled addons: storage-provisioner, default-storageclass
Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\Users\Ruthik>
```

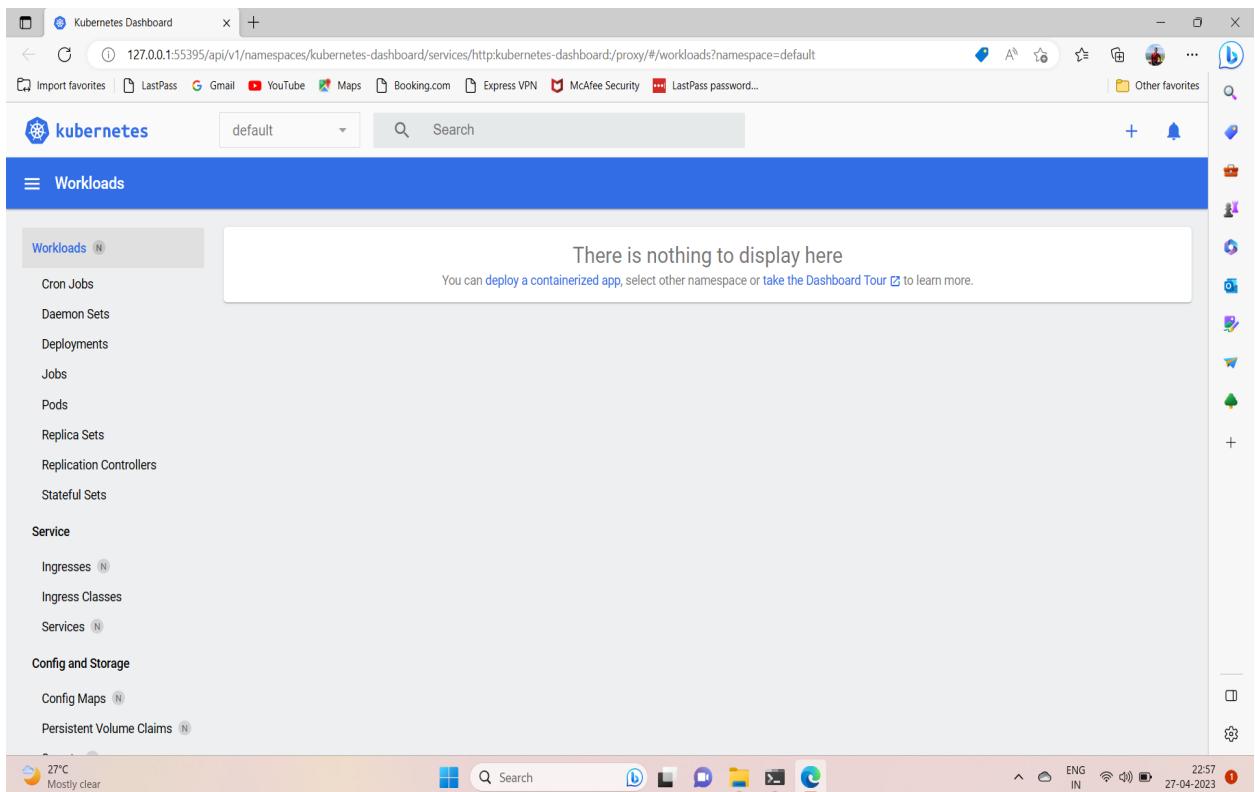
Once the minikube has started, start the Kubernetes cluster dashboard by running the following command.

>> minikube dashboard



```
Administrator: Windows PowerShell C:\Users\Ruthik> minikube dashboard
Verifying dashboard health ...
Launching proxy ...
Verifying proxy health ...
Opening http://127.0.0.1:62121/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/ in your default browser...
```

A prompt will open on Browser.



Now, click on the + icon on the top right corner of the window.

Now, click on ‘create from form’ give app name of your choice and specify the docker container image name with username and specify the port number and click on deploy. After that your website will be hosted successfully.

The screenshot shows the Kubernetes Dashboard interface. On the left, a sidebar lists various workload types: Cron Jobs, Daemon Sets, Deployments, Jobs, Pods, Replica Sets, Replication Controllers, and Stateful Sets. Below that, under Service, are Ingresses, Ingress Classes, and Services. Under Config and Storage, there are Config Maps and Persistent Volume Claims. The main area is titled 'Workloads' and contains two sections: 'Workload Status' and 'Deployments'. The 'Workload Status' section has three green circles representing 'Running' counts: 2 for Deployments, 6 for Pods, and 4 for Replica Sets. The 'Deployments' section contains a table with two rows:

Name	Images	Labels	Pods	Created
ruthikrj-myweb	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb	3 / 3	11.hours.ago
ruthikrj-myweb	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb	3 / 3	11.hours.ago

More information about the cluster.

The screenshot shows the Kubernetes Dashboard interface, similar to the previous one but with a different focus. The sidebar includes all the same categories as before. The main area is titled 'Workloads' and contains a 'Pods' section. This section displays a table of five pods, each with a green circular icon and a tooltip showing its name and labels. The columns are: Name, Images, Labels, Node, Status, Restarts, CPU Usage (cores), Memory Usage (bytes), and Created. The pods listed are:

Name	Images	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created
ruthikrj-myweb-6b8954d4bf-cnbg5	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-has-h: 6b8954d4bf	minikube	Running	0	-	-	4.minutes.ago
ruthikrj-myweb-6b8954d4bf-8w79r	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-has-h: 6b8954d4bf	minikube	Running	0	-	-	4.minutes.ago
ruthikrj-myweb-6b8954d4bf-cn6cn	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-has-h: 6b8954d4bf	minikube	Running	0	-	-	4.minutes.ago
ruthikrj-myweb-7777b85f88-t5xdg	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-has-h: 7777b85f88	minikube	Running	0	-	-	5.minutes.ago
ruthikrj-myweb-7777b85f88-4xqjp	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-has-h: 7777b85f88	minikube	Running	0	-	-	5.minutes.ago

Kubernetes Dashboard

127.0.0.1:62121/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/#/workloads?namespace=default

Import favorites | LastPass | Gmail | YouTube | Maps | Booking.com | Express VPN | McAfee Security | LastPass password...

kubernetes

default

Search

Workloads

Workloads (2)

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Service

Ingresses (1)

Ingress Classes

Services (1)

Config and Storage

Config Maps (1)

Persistent Volume Claims (1)

ruthikrj-myweb-7777b85f88-4xqjp

ruthikjadhab/myweb

k8s-app: ruthikrj-myweb
pod-template-hash: 7777b85f88

minikube

Running 0 - - 6.minutes ago

ruthikrj-myweb-7777b85f88-mvjqz

ruthikjadhab/myweb

k8s-app: ruthikrj-myweb
pod-template-hash: 7777b85f88

minikube

Running 0 - - 6.minutes ago

Replica Sets

Name	Images	Labels	Pods	Created
ruthikrj-myweb-6b8954d4bf	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-hash: 6b8954d4bf	3 / 3	5.minutes.ago
ruthikrj-myweb-7777b85f88	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-hash: 7777b85f88	3 / 3	6.minutes.ago
ruthikrj-myweb-c97c65948	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-hash: c97c65948	0 / 0	11.hours.ago
ruthikrj-myweb-75bcd56cf8	ruthikjadhab/myweb	k8s-app: ruthikrj-myweb pod-template-hash: 75bcd56cf8	0 / 0	11.hours.ago

30°C Cloudy

Search

ENG IN

11:09 28-04-2023

This is our deployment on the specified port.

Kubernetes Dashboard

localhost:84

Import favorites | LastPass | Gmail | YouTube | Maps | Booking.com | Express VPN | McAfee Security | LastPass password...

FREGG

HOME ABOUT SERVICE CONTACT US

WELCOME TO CONTENT WRITING SERVICES

It is a long established fact that a reader will be distracted by the readable content of a page when looking

Contact Us

01

02

30°C Cloudy

Search

ENG IN

11:10 28-04-2023

A screenshot of a Microsoft Edge browser window. The address bar shows 'localhost:84'. The main content area displays an 'About Us' page with the following text:

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their

[Get Started](#)

To the right of the text is a photograph of a person's hands typing on a laptop keyboard. The laptop screen shows a slide presentation with the title 'About Us' and some text.

A screenshot of a Microsoft Edge browser window. The address bar shows 'localhost:84'. The main content area displays a landing page with the following text:

At Your Service

Written with Love

Fast Turnaround

It is a long established fact that a reader will be distracted

The page includes a small red heart icon and a weather widget showing '30°C Cloudy'. The bottom of the screen shows the Windows taskbar with various pinned icons and system status indicators.

Conclusion:

Thus, I've learnt how we can deploy web applications using the Kubernetes cluster.