**Step to deploy simple web application created using React JS with Kubernetes with Minikube cluster environment**

1. **Create react js project**
2. **Build the react js application**
3. **Then create docker file**
4. **Publish the image.**
5. **Then start minikube (one the of the cluster environment)**
6. **minikube start (start minikube cluster)**
7. **minikube dashboard (open Minikube GUI base dashboard)**
8. **create the deployment file with anyname.yml : this file contains pods configuration details. One or many pods link with pre defined or user defined image**

**deployment.yml**

**apiVersion: apps/v1**

**kind: Deployment**

**metadata:**

**name: my-server**

**labels:**

**app: my-server**

**spec:**

**replicas: 3**

**selector:**

**matchLabels:**

**app: my-server**

**template:**

**metadata:**

**labels:**

**app: my-server**

**spec:**

**containers:**

**- name: web-server-container**

**image: akashkale/my-reactjs:pro1**

1. **kubectl apply -f deployment.yml**
2. **kubectl get deployments**
3. **kubectl get pods**
4. **if we want access container part of pods we need to expose as service. Then only we can access container.**
5. **Now we need to create the service.yml. which contains deployment details and expose as service.**

**service.yml**

**apiVersion: v1**

**kind: Service**

**metadata:**

**name: simple-app-service**

**labels:**

**app: my-server**

**spec:**

**type: LoadBalancer**

**selector:**

**app: my-server**

**ports:**

**- port: 80**

**nodePort: 30001**

**targetPort: 80**

**protocol: TCP**

1. **kubectl apply -f service.yml**
2. **kubectl get service**
3. **now to view application output we need to take the help of minikube command**
4. **minikube service servicename**
5. **you can view the application on browser.**
6. **Once task done do the clean up activity**
7. **Minikube stop**
8. **Using docker desktop stop as well as remove container**
9. **Then remove images**

**CI and CD:** Continuous integration and Continuous deployment or delivery

Dev1

Login page Shared

Dev2 remote

Resapi repository 🡨----------🡪 CI / CD --🡪 actual/testing server

Dev3 (git hub)

database design **Manager**

Dev4

Testing code

**Jenkin :** Jenkin is an open source CI and CD tools base upon Java technologies.

Jenkin is plugin base CI and CD tools. Jenkin is a GUI base CI and CD tools.

Jenkin provide environment to build the different type of projects.

Download Jenkin war file from below URL

<https://www.jenkins.io/download/>

open the command prompt and run the below

**java -jar jenkins.war by default 8080**

or

**java -jar jenkins.war --httpPort=9090**

<http://localhost:8080>

<http://localhost:9090>

first time it will ask you password. Your can see your random password on console.

Please select installed suggested plugin

After plugin install it will ask you to create the account. Please create the account and login to Jenkin dashboard.