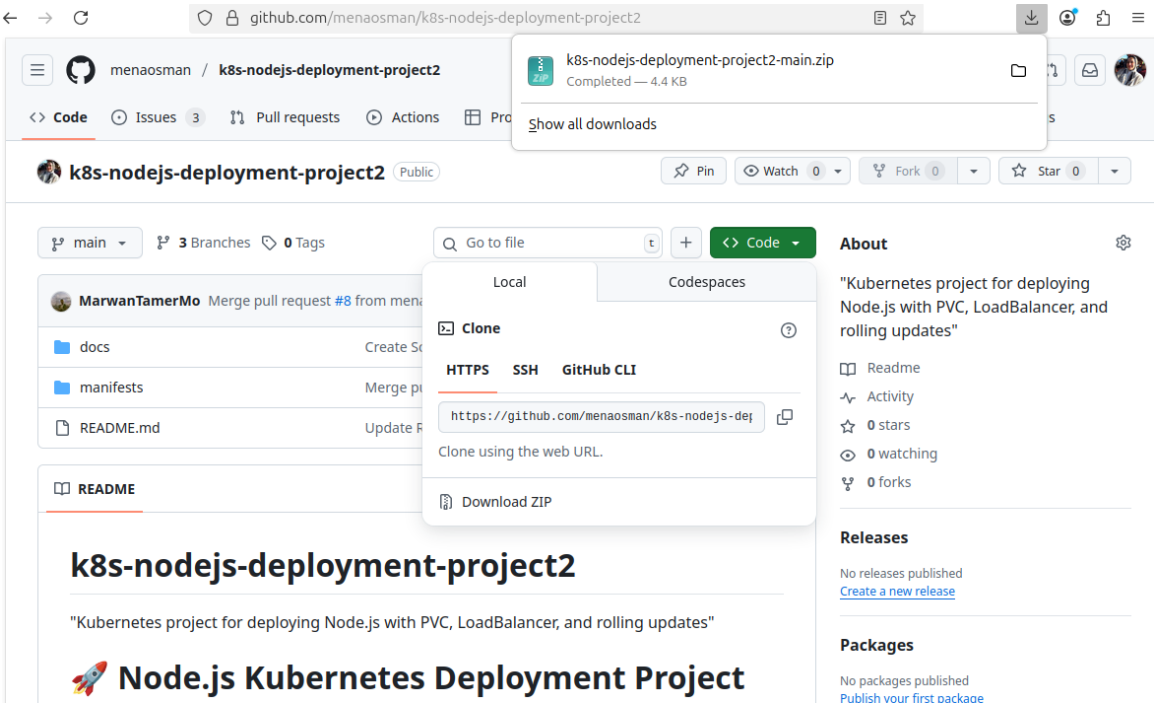
Kubernetes Node.js Screenshots from Testing Process

Below are screenshots captured during various phases of testing the Kubernetes Node.js deployment pipeline:

📸 1.Minikube start and environment check

First, download the folder from github link unzip your folder:

{https://github.com/menaosman/k8s-nodejs-deployment-project2}



```
menna@menna-VMware-Virtual-Platform:~$ cd /home/menna/Downloads
menna@menna-VMware-Virtual-Platform:~/Downloads$ unzip k8s-nodejs-deployment-project2-main.zip
Archive:  k8s-nodejs-deployment-project2-main.zip
1e433b3cd906f9f696f36b31a30fb12f9a36c3ea
   creating: k8s-nodejs-deployment-project2-main/
  inflating: k8s-nodejs-deployment-project2-main/README.md
   creating: k8s-nodejs-deployment-project2-main/docs/
 extracting: k8s-nodejs-deployment-project2-main/docs/Final_Report.md
 extracting: k8s-nodejs-deployment-project2-main/docs/Screenshots
 extracting: k8s-nodejs-deployment-project2-main/docs/tasks.md
   creating: k8s-nodejs-deployment-project2-main/manifests/
  inflating: k8s-nodejs-deployment-project2-main/manifests/node-deployment.yaml
  inflating: k8s-nodejs-deployment-project2-main/manifests/node-pvc.yaml
  inflating: k8s-nodejs-deployment-project2-main/manifests/node-service.yaml
menna@menna-VMware-Virtual-Platform:~/Downloads$ cd k8s-nodejs-deployment-project2-main
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ ls
docs  manifests  README.md
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ ls manifests
```

📸 2. File Check & Structure & start the minikube :

confirms that your folder includes at least these files:

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ ls
docs  manifests  README.md
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ ls manifests
node-deployment.yaml  node-pvc.yaml  node-service.yaml
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ minikube start
😄  minikube v1.36.0 on Ubuntu 24.04
✨  Using the docker driver based on existing profile
👍  Starting "minikube" primary control-plane node in "minikube" cluster
🚜  Pulling base image v0.0.47 ...
🔄  Restarting existing docker container for "minikube" ...

🧯  Docker is nearly out of disk space, which may cause deployments to fail! (94% of capacity). You can pass '--force' to
skip this check.
💡  Suggestion:

    Try one or more of the following to free up space on the device:
```

📸 Step 1: Apply Persistent Volume Claim:

```
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl apply -f manifests/node-pvc.
yaml
persistentvolumeclaim/nodejs-pvc created
```

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl get pvc
NAME         STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTE
SCLASS   AGE
nodejs-pvc   Bound    pvc-7cf656bb-08fa-4bc1-99d4-639f6ebb81e8   1Gi        RWO            standard       <unset>
         44s
```

📸 Step 2: Apply Node.js Deployment:

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl apply -f manifests/node-depl
oyment.yaml
deployment.apps/nodejs-app-deployment created
```

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl get pods
NAME                                    READY   STATUS    RESTARTS       AGE
config-test                             1/1     Running   1 (116s ago)   2d6h
echoserver-85678b6cb4-6qwcq             1/1     Running   1 (116s ago)   2d7h
nginx                                   1/1     Running   3 (116s ago)   4d12h
nginx-764dd87c46-p8kw2                  1/1     Running   3 (116s ago)   4d12h
nginx-764dd87c46-pgm8z                  1/1     Running   3 (116s ago)   4d12h
nginx-764dd87c46-ts2zn                  1/1     Running   3 (116s ago)   4d12h
nginx-deployment-d7b8c5b4b-g8cbc        1/1     Running   1 (116s ago)   2d6h
nginx-deployment-d7b8c5b4b-mqwgz        1/1     Running   1 (116s ago)   2d6h
nodejs-app-deployment-5cbc7b585b-9ztm5  1/1     Running   0              11s
nodejs-app-deployment-5cbc7b585b-jfccs  1/1     Running   0              11s
solar-system-5f8b8dc9c6-bjvqq           1/1     Running   4 (116s ago)   5d17h
```

📸 Step 3: Apply the LoadBalancer Service

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl apply -f manifests/node-se
ice.yaml
service/nodejs-service created
```

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ minikube service nodejs-service
|-----------|---------------|-------------|--------------------------|
| NAMESPACE |      NAME     | TARGET PORT |            URL           |
|-----------|---------------|-------------|--------------------------|
| default   | nodejs-service |          80 | http://192.168.49.2:32217 |
|-----------|---------------|-------------|--------------------------|
🎉  Opening service default/nodejs-service in default browser...
```

📸 Step 4 :Check the PVC Mount:

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl exec -it nodejs-app-deployme
nt-5cbc7b585b-9ztm5 -- sh
/ # cd /app/data
/app/data # touch test.txt
/app/data # ls
test.txt
/app/data # ^C

/app/data # exit
command terminated with exit code 130
```

📸 New pods created after rolling update:

Edit node-deployment.yaml

🔄 Change image from:

image: node:18-alpine

⬇️ To:

image: node:20-alpine

Then reapply:

kubectl apply -f node-deployment.yaml

kubectl rollout status deployment/nodejs-app-deployment

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ nano manifests/node-deployment.yaml
```

```
  GNU nano 7.2                                manifests/node-deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nodejs-app-deployment
  labels:
    app: nodejs-app
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nodejs-app
  template:
    metadata:
      labels:
        app: nodejs-app
    spec:
      containers:
      - name: nodejs-app
        image: node:20-alpine
        command: ["node"]
        args: ["-e", "require('http').createServer((req, res) => { res.writeHead(200); res.end('Hello from Node.js on po>
        ports:
        - containerPort: 3000
        resources:
          requests:
            memory: "64Mi"
            cpu: "250m"
          limits:
```

📸 Node version check inside updated pod (v20.19.4)

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl apply -f manifests/node-depl
oyment.yaml
deployment.apps/nodejs-app-deployment configured
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl rollout status deployment/no
dejs-app-deployment
Waiting for deployment "nodejs-app-deployment" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "nodejs-app-deployment" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "nodejs-app-deployment" rollout to finish: 2 old replicas are pending termination...
Waiting for deployment "nodejs-app-deployment" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "nodejs-app-deployment" rollout to finish: 1 old replicas are pending termination...
deployment "nodejs-app-deployment" successfully rolled out
```

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl get pods -o wide
NAME                                READY   STATUS    RESTARTS      AGE     IP            NODE       NOMINATED NODE
 READINESS GATES
config-test                         1/1     Running   1 (15m ago)   2d6h    10.244.0.41   minikube   <none>
 <none>
echoserver-85678b6cb4-6qwcq         1/1     Running   1 (15m ago)   2d7h    10.244.0.37   minikube   <none>
 <none>
nginx                               1/1     Running   3 (15m ago)   4d12h   10.244.0.36   minikube   <none>
 <none>
nginx-764dd87c46-p8kw2              1/1     Running   3 (15m ago)   4d12h   10.244.0.39   minikube   <none>
 <none>
nginx-764dd87c46-pgm8z              1/1     Running   3 (15m ago)   4d12h   10.244.0.35   minikube   <none>
 <none>
nginx-764dd87c46-ts2zn              1/1     Running   3 (15m ago)   4d12h   10.244.0.44   minikube   <none>
 <none>
nginx-deployment-d7b8c5b4b-g8cbc    1/1     Running   1 (15m ago)   2d6h    10.244.0.38   minikube   <none>
 <none>
nginx-deployment-d7b8c5b4b-mqwgz    1/1     Running   1 (15m ago)   2d6h    10.244.0.40   minikube   <none>
 <none>
nodejs-app-deployment-7fc55959f-95jzs   1/1     Running   0             4m3s    10.244.0.48   minikube   <none>
 <none>
nodejs-app-deployment-7fc55959f-9hc6v   1/1     Running   0             4m24s   10.244.0.47   minikube   <none>
 <none>
solar-system-5f8b8dc9c6-bjvqq       1/1     Running   4 (15m ago)   5d17h   10.244.0.42   minikube   <none>
```

📸 Final successful test of full deployment pipeline

node -v = v20.19.4 in new pod

```
menna@menna-VMware-Virtual-Platform:~/Downloads/k8s-nodejs-deployment-project2-main$ kubectl exec -it nodejs-app-deployme
nt-7fc55959f-95jzs -- node -v
```