📄 Task 4: Implement Rolling Updates

🎯 Objective

Configure the Deployment to use **RollingUpdate** strategy with controlled surge and availability, then trigger an image version change and observe the rollout.

🛠️ Steps Taken

1. Edited the Deployment manifest to include the following rolling update strategy:

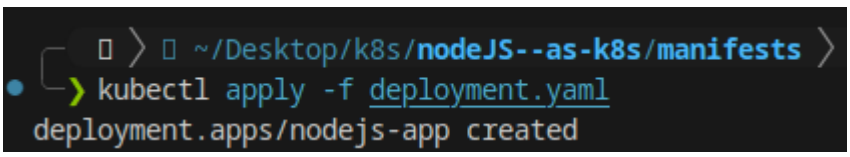
```
strategy:
  type: RollingUpdate
  rollingUpdate:
    maxSurge: 1
    maxUnavailable: 0
```

2. Applied the updated Deployment using:
   `kubectl apply -f deployment.yaml`
3. Triggered a new rollout by updating the container image:
   `kubectl set image deployment/nodejs-app nodejs=aw033/nodejs-hello:v1`
4. Monitored the rollout progress:
   `kubectl rollout status deployment/my-app`
5. Verified updated pods and running image version:
   `kubectl get pods -o wide`
6. Described the Deployment to inspect rollout details:
   `kubectl describe deploy nodejs-app`
7. Performed a rollback to the previous version:
   `kubectl rollout undo deployment/nodejs-app`
8. Verified the rollback via Deployment description:
   `kubectl describe deploy nodejs-app`

📸 Screenshots

• **Screenshot 1:**

`kubectl apply -f deployment.yaml`



- **Screenshot 2:**

`kubectl set image deployment/nodejs-app nodejs=aw033/nodejs-hello:v1`

```
PS C:\Users\user\Desktop\Banque Misr -Sprints\NodeJS-as-K8s> kubectl set image deployment/nodejs-app nodejs=aw033/nodejs-hello:v1
deployment.apps/nodejs-app image updated
```

- **Screenshot 3:**

`kubectl rollout status deployment/my-app`

```
🔄 Updating image to aw033/nodejs-hello:v1...
deployment.apps/nodejs-app image updated
Waiting for deployment "nodejs-app" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "nodejs-app" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "nodejs-app" rollout to finish: 1 out of 2 new replicas have been updated...
Waiting for deployment "nodejs-app" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "nodejs-app" rollout to finish: 1 old replicas are pending termination...
deployment "nodejs-app" successfully rolled out
```

- **Screenshot 4:**

`kubectl get pods -o wide`

```
 ~/Desktop/k8s/nodeJS--as-k8s/app  main !4
❯ kubectl get pods -o wide
NAME                          READY   STATUS    RESTARTS   AGE     IP            NODE       NOMINATED NODE   READINESS GATES
nodejs-app-77dcb56dbc-g972r   1/1     Running   0          9m59s   10.244.0.33   minikube   <none>           <none>
nodejs-app-77dcb56dbc-k88g2   1/1     Running   0          9m59s   10.244.0.34   minikube   <none>           <none>
```

- **Screenshot 5:**

kubectl describe deploy nodejs-app

```
PS C:\Users\user\Desktop\Banque Misr -Sprints\NodeJS-as-K8s> kubectl describe deploy nodejs-app
Name:                   nodejs-app
Namespace:              default
CreationTimestamp:      Fri, 01 Aug 2025 03:22:21 +0300
Labels:                 <none>
Annotations:            deployment.kubernetes.io/revision: 3
Selector:               app=nodejs
Replicas:               2 desired | 2 updated | 2 total | 2 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nodejs
  Containers:
   nodejs:
    Image:         aw033/nodejs-hello:v1
    Port:          3000/TCP
    Host Port:     0/TCP
    Environment:   <none>
    Mounts:        <none>
  Volumes:         <none>
  Node-Selectors:  <none>
  Tolerations:     <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  nodejs-app-dbd4566bb (0/0 replicas created)
NewReplicaSet:   nodejs-app-7677dd445c (2/2 replicas created)
Events:
  Type    Reason             Age   From                   Message
  ----    ------             ----  ----                   -------
  Normal  ScalingReplicaSet  34h   deployment-controller  Scaled up replica set nodejs-app-7677dd445c from 0 to 2
  Normal  ScalingReplicaSet  20s   deployment-controller  Scaled up replica set nodejs-app-dbd4566bb from 0 to 1
  Normal  ScalingReplicaSet  9s    deployment-controller  Scaled down replica set nodejs-app-dbd4566bb from 1 to 0
PS C:\Users\user\Desktop\Banque Misr -Sprints\NodeJS-as-K8s>
```

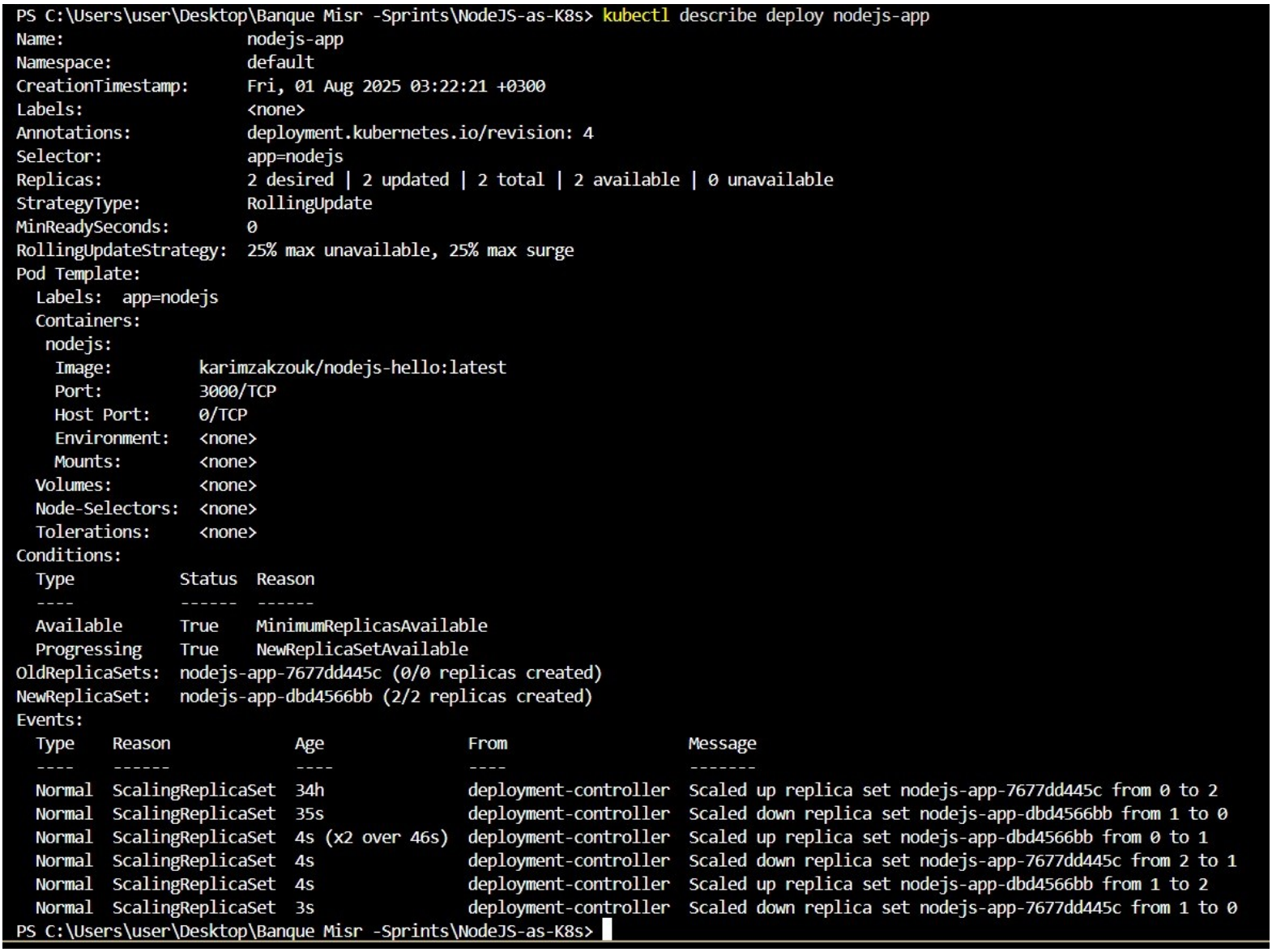
- **Screenshot 6:**

kubectl rollout undo deployment/nodejs-app

```
PS C:\Users\user\Desktop\Banque Misr -Sprints\NodeJS-as-K8s> kubectl rollout undo deployment/nodejs-app
deployment.apps/nodejs-app rolled back
```

- **Screenshot 7:**

`kubectl describe deploy nodejs-app`



```
PS C:\Users\user\Desktop\Banque Misr -Sprints\NodeJS-as-K8s> kubectl describe deploy nodejs-app
Name:                   nodejs-app
Namespace:              default
CreationTimestamp:      Fri, 01 Aug 2025 03:22:21 +0300
Labels:                 <none>
Annotations:            deployment.kubernetes.io/revision: 4
Selector:               app=nodejs
Replicas:               2 desired | 2 updated | 2 total | 2 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nodejs
  Containers:
   nodejs:
    Image:         karimzakzouk/nodejs-hello:latest
    Port:          3000/TCP
    Host Port:     0/TCP
    Environment:   <none>
    Mounts:        <none>
  Volumes:         <none>
  Node-Selectors:  <none>
  Tolerations:     <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  nodejs-app-7677dd445c (0/0 replicas created)
NewReplicaSet:   nodejs-app-dbd4566bb (2/2 replicas created)
Events:
  Type    Reason             Age                From                   Message
  ----    ------             ----               ----                   -------
  Normal  ScalingReplicaSet  34h                deployment-controller  Scaled up replica set nodejs-app-7677dd445c from 0 to 2
  Normal  ScalingReplicaSet  35s                deployment-controller  Scaled down replica set nodejs-app-dbd4566bb from 1 to 0
  Normal  ScalingReplicaSet  4s (x2 over 46s)   deployment-controller  Scaled up replica set nodejs-app-dbd4566bb from 0 to 1
  Normal  ScalingReplicaSet  4s                 deployment-controller  Scaled down replica set nodejs-app-7677dd445c from 2 to 1
  Normal  ScalingReplicaSet  4s                 deployment-controller  Scaled up replica set nodejs-app-dbd4566bb from 1 to 2
  Normal  ScalingReplicaSet  3s                 deployment-controller  Scaled down replica set nodejs-app-7677dd445c from 1 to 0
PS C:\Users\user\Desktop\Banque Misr -Sprints\NodeJS-as-K8s>
```

✅ Outcome

The Deployment used a **RollingUpdate** strategy successfully. The application was updated incrementally with zero downtime, and all pods transitioned to the new version while maintaining availability.