Node JS Deployment Pipeline

**Pods:**

```
NAME          TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes    ClusterIP   10.96.0.1    <none>        443/TCP   3d16h
jitendrasinghthakur@jitendras-MacBook-Air ~ % kubectl get pods
NAME                              READY   STATUS    RESTARTS   AGE
node-hello-677747796d-4c8c4   1/1     Running   0          8m10s
node-hello-677747796d-m98lt   1/1     Running   0          8m9s
node-hello-677747796d-pztw9   1/1     Running   0          8m20s
jitendrasinghthakur@jitendras-MacBook-Air ~ %
```

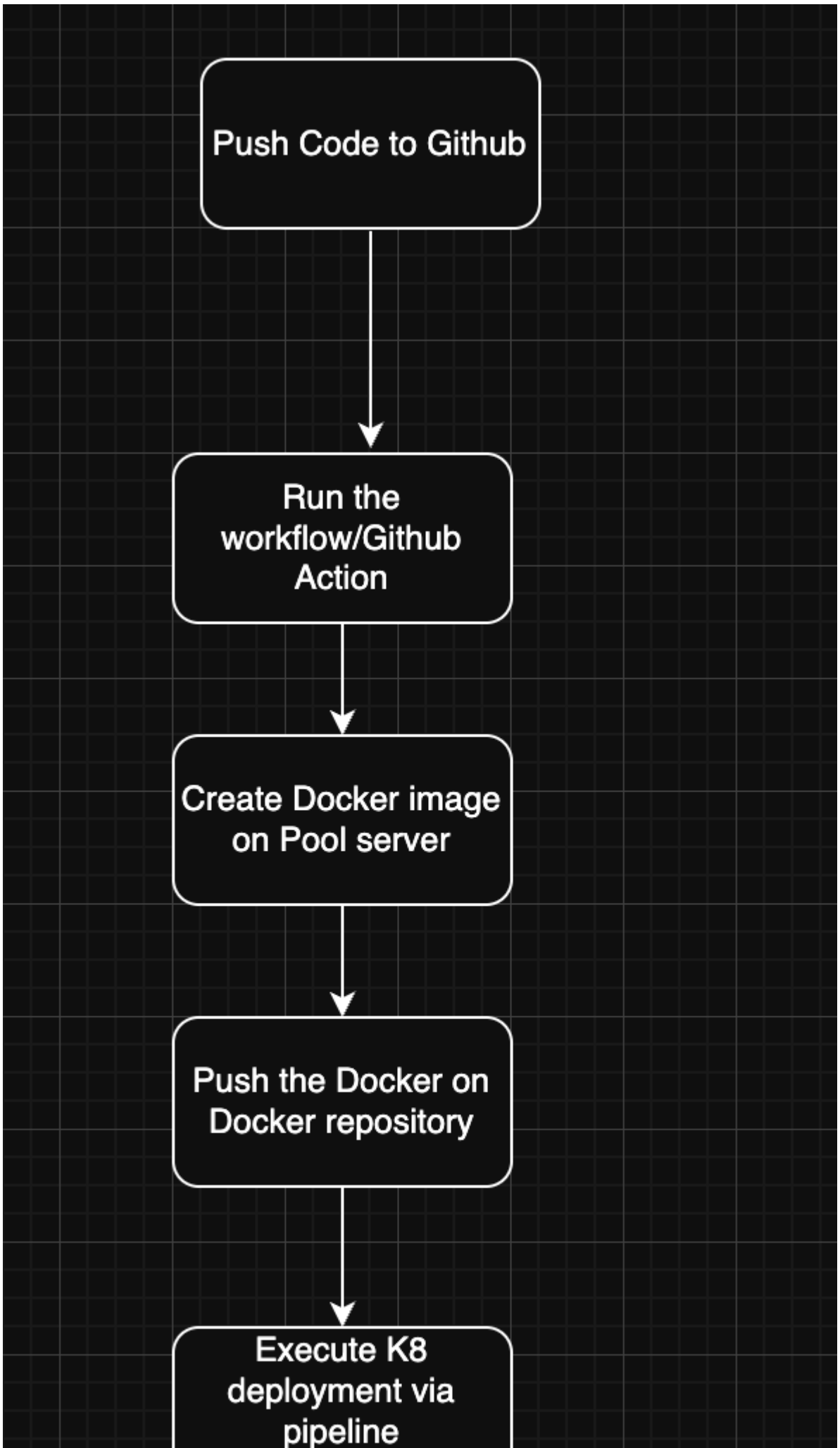
**Deployment Details**

```
jitendrasinghthakur@jitendras-MacBook-Air ~ % kubectl describe deployment node-hello
Name:                   node-hello
Namespace:              default
CreationTimestamp:      Mon, 29 Sep 2025 17:23:18 +0530
Labels:                 app=node-hello
Annotations:            deployment.kubernetes.io/revision: 4
Selector:               app=node-hello
Replicas:               3 desired | 3 updated | 3 total | 3 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=node-hello
  Containers:
   node-hello:
    Image:         jitendrathakur2/node-hello:1759217002
    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:  node-hello-6885cb8d75 (0/0 replicas created), node-hello-86c64c7cf6 (0/0 replicas created), node-hello-5b6bfbd7fc (0/0 replicas created)
NewReplicaSet:   node-hello-677747796d (3/3 replicas created)
Events:
  Type    Reason             Age   From                   Message
  ----    ------             ----  ----                   -------
  Normal  ScalingReplicaSet  11m   deployment-controller  Scaled up replica set node-hello-677747796d from 0 to 1
  Normal  ScalingReplicaSet  11m   deployment-controller  Scaled down replica set node-hello-5b6bfbd7fc from 3 to 2
  Normal  ScalingReplicaSet  11m   deployment-controller  Scaled up replica set node-hello-677747796d from 1 to 2
  Normal  ScalingReplicaSet  11m   deployment-controller  Scaled down replica set node-hello-5b6bfbd7fc from 2 to 1
  Normal  ScalingReplicaSet  11m   deployment-controller  Scaled up replica set node-hello-677747796d from 2 to 3
  Normal  ScalingReplicaSet  11m   deployment-controller  Scaled down replica set node-hello-5b6bfbd7fc from 1 to 0
```

Cluster Used: Local system
Agent Pool: Local, self hosted server
Docker: Docker Desktop
Used MiniKube for K8 setup

**Scope of Improvement:**

- Create Docker Image scan check for security
- use encrypted secured tool to store the tokens and secret
- Integrate Dynamic and Static scan check
- Create a Test.yml file which will trigger as soon as code is merged to main, and provide us code coverage report (SonarQube integration)


Push Code to Github
Run the workflow/Github Action
Create Docker image on Pool server
Push the Docker on Docker repository
Execute K8 deployment via pipeline