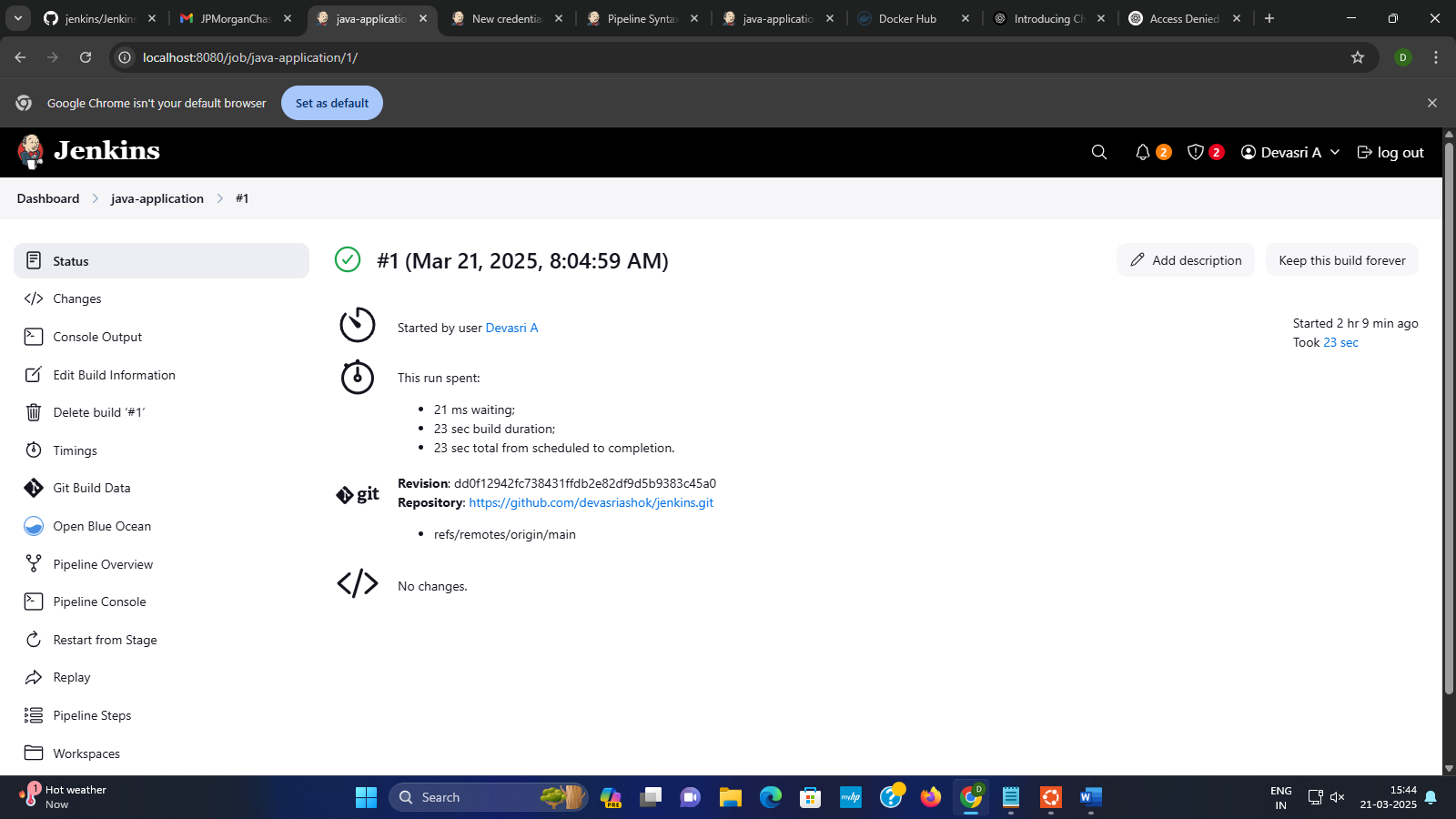
Day 5:

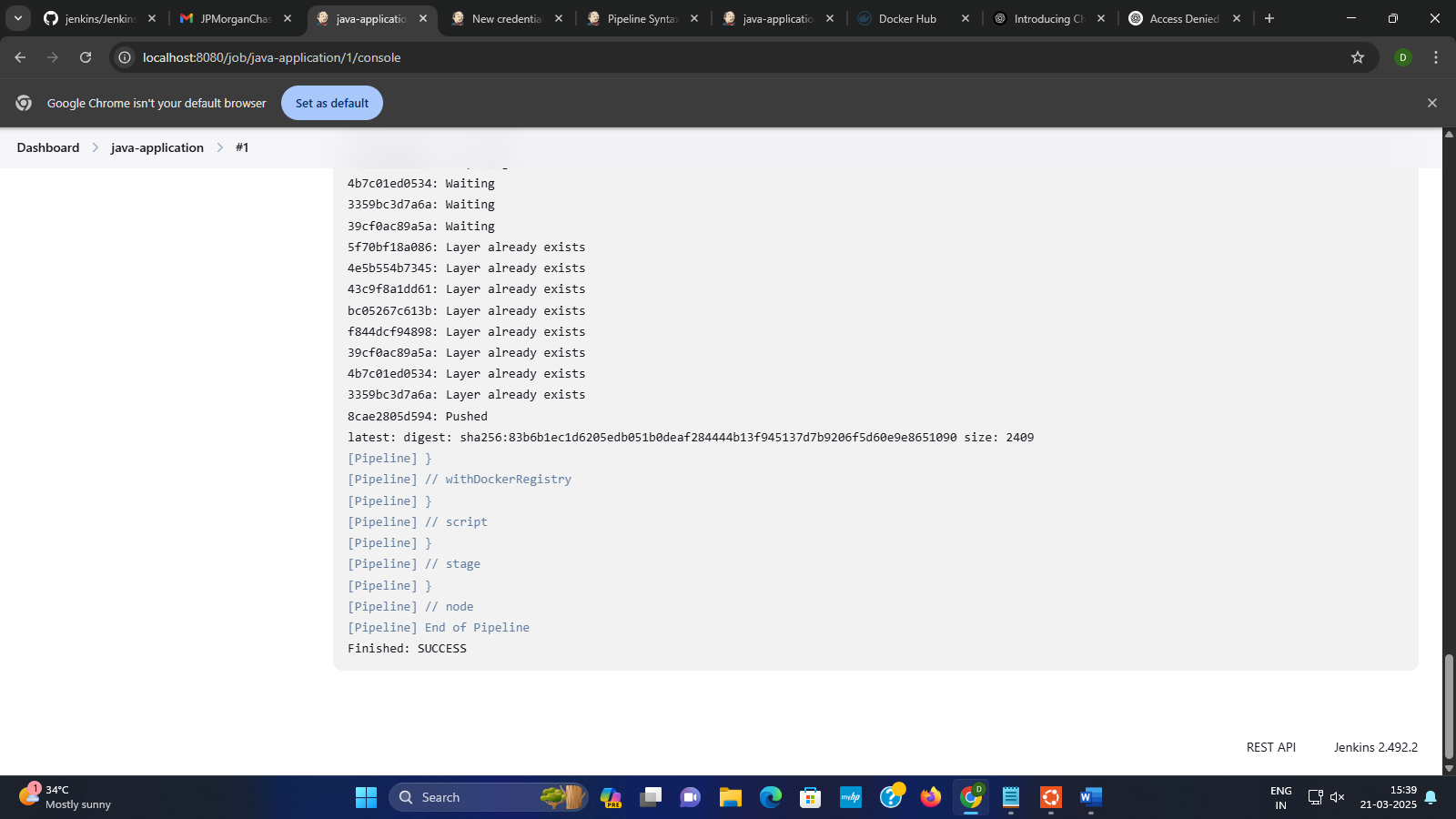
**PIPELINE CODE FOR DEPLOY:**

pipeline {  
    agent any  
  
    stages {  
        stage('scm') {  
            steps {  
        git branch: 'main', url: '<https://github.com/devasriashok/jenkins.git>'  
            }  
        }  
        stage('build') {  
            steps {  
               sh "mvn clean"  
               sh "mvn install"  
}  
}  
stage('build to images') {  
            steps {  
               script{  
                  sh 'docker build -t devasria/mysimplewebapplication .'  
               }  
    }  
}  
stage('push to hub') {  
            steps {  
               script{  
                withDockerRegistry(credentialsId: 'docker\_cre', url: '<https://index.docker.io/v1/>') {  
                  sh 'docker push devasria/mysimplewebapplication'  
               }  
            }  
            }  
}  
stage('deploy') {  
            steps {  
               withKubeConfig(caCertificate: '', clusterName: 'minikube', contextName: 'minikube', credentialsId: 'minikube-cred', namespace: '', restrictKubeConfigAccess: false, serverUrl: '[https://192.168.39.226:8443](https://192.168.39.226:8443/)') {  
                 sh 'kubectl apply -f deployment\_NP.yml--validate=false'  
}  
}  
}    
           
}  
}

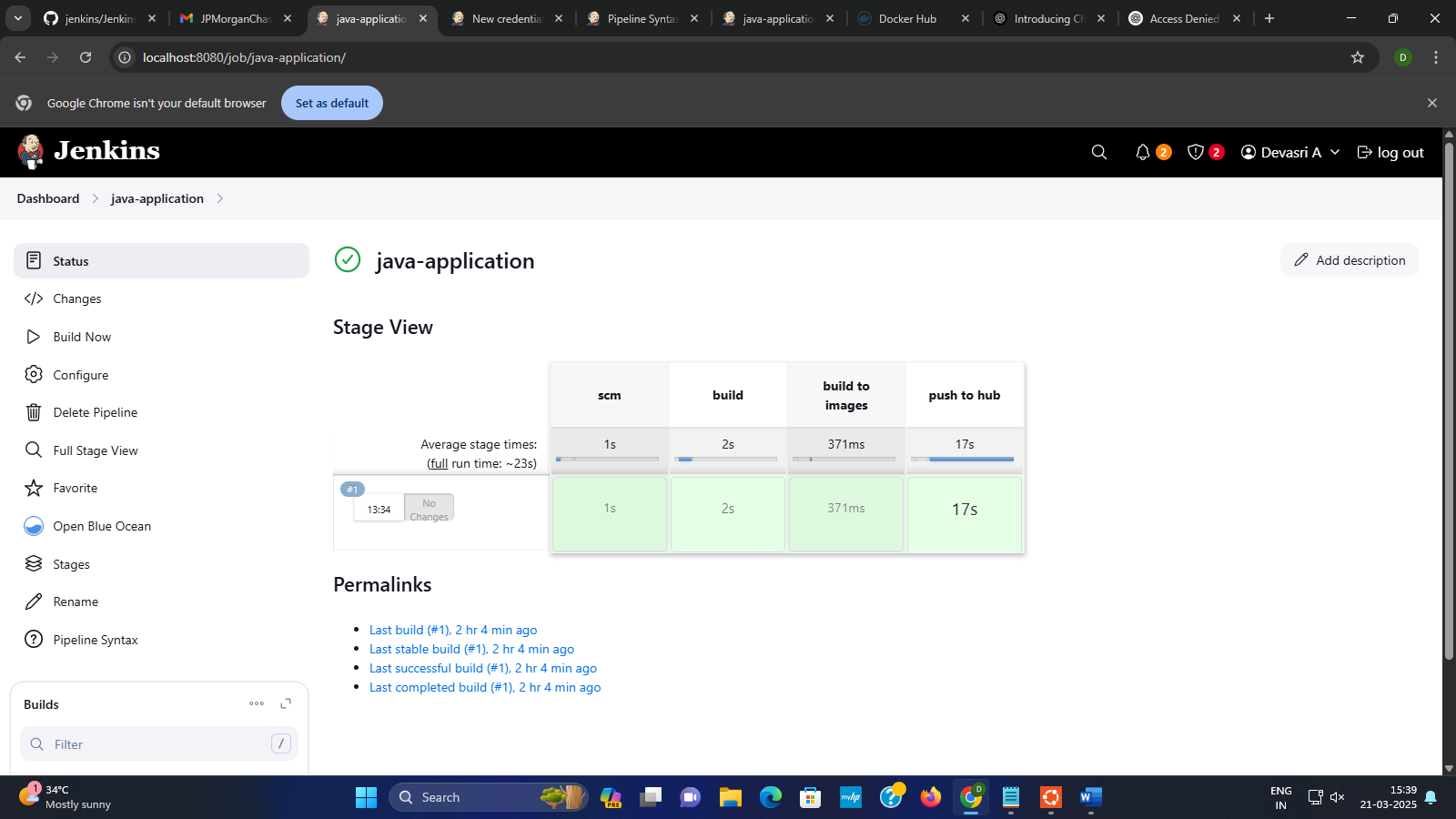
**BUILD OUTPUT: MINIKUBE DEPLOYMENT**



BUILD STATUS



CONSOLE OUTPUT



**DEPLOYOMENT.YML**

**Visudo:**

1. sudo visudo

2. Add jenkins ALL=(ALL) NOPASSWD: ALL

**ssh installtion:**

1. sudo systemctl restart ssh.service

2. sudo systemctl restart sshd.service

3. sudo apt update

4. sudo apt install openssh-server

5. sudo systemctl restart ssh

6. sudo systemctl status ssh

7. ls /etc/systemd/system/sshd.service or ls /usr/lib/systemd/system/sshd.service

8. sudo systemctl daemon-reload

9. sudo systemctl status ssh

**Deployment:**

1. cd ~/.kube

2. ls

3. cat config

4. sudo vi config

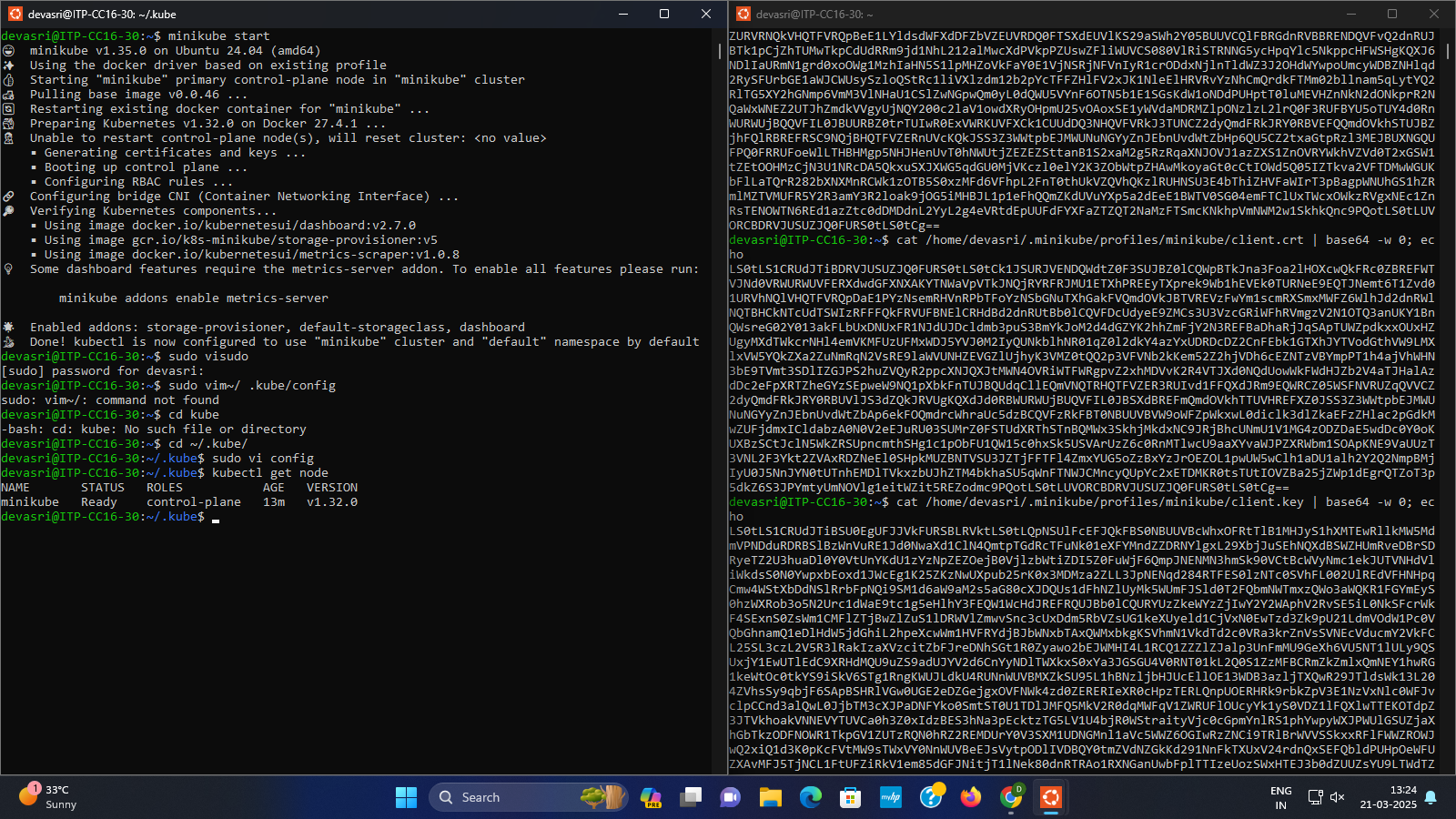
5. i

6. -data

7. cat url | base64 -w 0; echo

8. minikube start

9. kubectl get node



**OUTPUT**