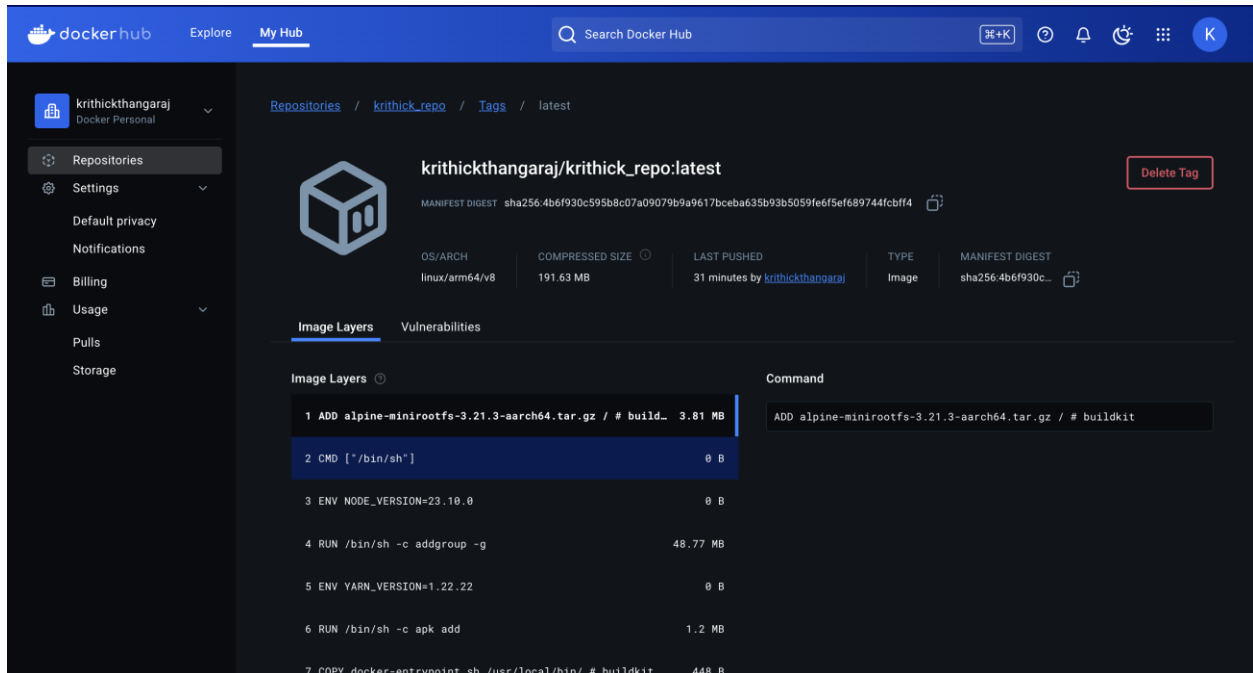


DAY 2



2.PIPELINE SCRIPT

```
pipeline { agent any tools { jdk 'jdk17' nodejs 'node20' }
```

```
environment {  
    DOCKER_CREDENTIALS = 'docker' // Jenkins Docker credentials ID  
    DOCKER_IMAGE = 'krithickthangaraj/krithick_repo'  
    GIT_REPO = 'https://github.com/krithickthangaraj/jenkins_devops.git'  
    CONTAINER_NAME = 'krithick_container'  
}
```

```
stages {  
    stage('Clean Workspace') {  
        steps {  
            script {  
                echo "Cleaning workspace..."  
                deleteDir() // Ensures a fresh start before cloning  
            }  
        }  
    }  
}
```

```
}
```

```
stage('Git Checkout') {  
  steps {  
    script {  
      git branch: 'main',  
        credentialsId: 'github_secured',  
        url: "${GIT_REPO}"  
    }  
  }  
}
```

```
stage('Install Dependencies') {  
  steps {  
    script {  
      sh ""  
      if [ -f package.json ]; then  
        echo "package.json found. Running npm install..."  
        npm install  
      else  
        echo "ERROR: package.json is missing. Stopping pipeline."  
        exit 1  
      fi  
      ""  
    }  
  }  
}
```

```
stage('Docker Build & Push') {  
  steps {  
    script {  
      withDockerRegistry(credentialsId: "${DOCKER_CREDENTIALS}") {  
        def tag = "latest"  
  
        sh "docker build -t ${DOCKER_IMAGE} ."  
        sh "docker tag ${DOCKER_IMAGE} ${DOCKER_IMAGE}:${tag}"  
        sh "docker push ${DOCKER_IMAGE}:${tag}"  
      }  
    }  
  }  
}
```

```

    }
  }
}

stage('Deploy Docker Container') {
  steps {
    script {
      echo "Stopping and removing any existing container..."
      sh "docker stop ${CONTAINER_NAME} || true"
      sh "docker rm ${CONTAINER_NAME} || true"

      echo "Running new container..."
      sh "docker run -d --name ${CONTAINER_NAME} -p 3001:3000
${DOCKER_IMAGE}:latest"
    }
  }
}

post {
  success {
    echo "Pipeline executed successfully! \U0001f680"
  }
  failure {
    echo "Pipeline failed! \u274c Check logs for issues."
  }
}
}

```

Tools and plugins

Java,Nodejs,Docker

Eclipse,pipeline

Dashboard > last-pipe > #7

- Status
- </> Changes
- 📄 Console Output
- 📄 Edit Build Information
- 🗑️ Delete build '#7'
- 🕒 Timings
- 📄 Git Build Data
- 🔗 Pipeline Overview
- 📄 Pipeline Console
- 🔄 Restart from Stage
- 🔄 Replay
- 📄 Pipeline Steps
- 📄 Workspaces
- ⬅️ Previous Build

🟢 #7 (Mar 19, 2025, 3:11:03 AM)

Add description


Keep this build forever

🕒 Started by user [krithick](#)

Started 47 min ago
Took 3 min 25 sec

🕒 This run spent:

- 10 ms waiting;
- 3 min 25 sec build duration;
- 3 min 25 sec total from scheduled to completion.

📄  Revision: 6e7f83e179c31f6af12c1415a697578e026f8492
Repository: https://github.com/krithickthangaraj/jenkins_devops.git

- refs/remotes/origin/main

</> No changes.

localhost:8080/job/last-pipe/7/changes

REST API Jenkins 2.492.2

Dashboard > last-pipe >

- Status
- </> Changes
- 🏗️ Build Now
- ⚙️ Configure
- 🗑️ Delete Pipeline
- 🔍 Full Stage View
- 📄 Stages
- 🔄 Rename
- 🔗 Pipeline Syntax

🟢 last-pipe

Add description

Stage View

Average stage times:
(full run time: ~3min 25s)

	Declarative: Tool Install	Clean Workspace	Git Checkout	Install Dependencies	Docker Build & Push	Deploy Docker Container	Declarative: Post Actions
	108ms	1s	1s	57s	2min 22s	1s	70ms
#7 03:11 No Changes	108ms	1s	1s	57s	2min 22s	1s	70ms

Permalinks

- Last build (#7), 48 min ago
- Last stable build (#7), 48 min ago
- Last successful build (#7), 48 min ago
- Last completed build (#7), 48 min ago

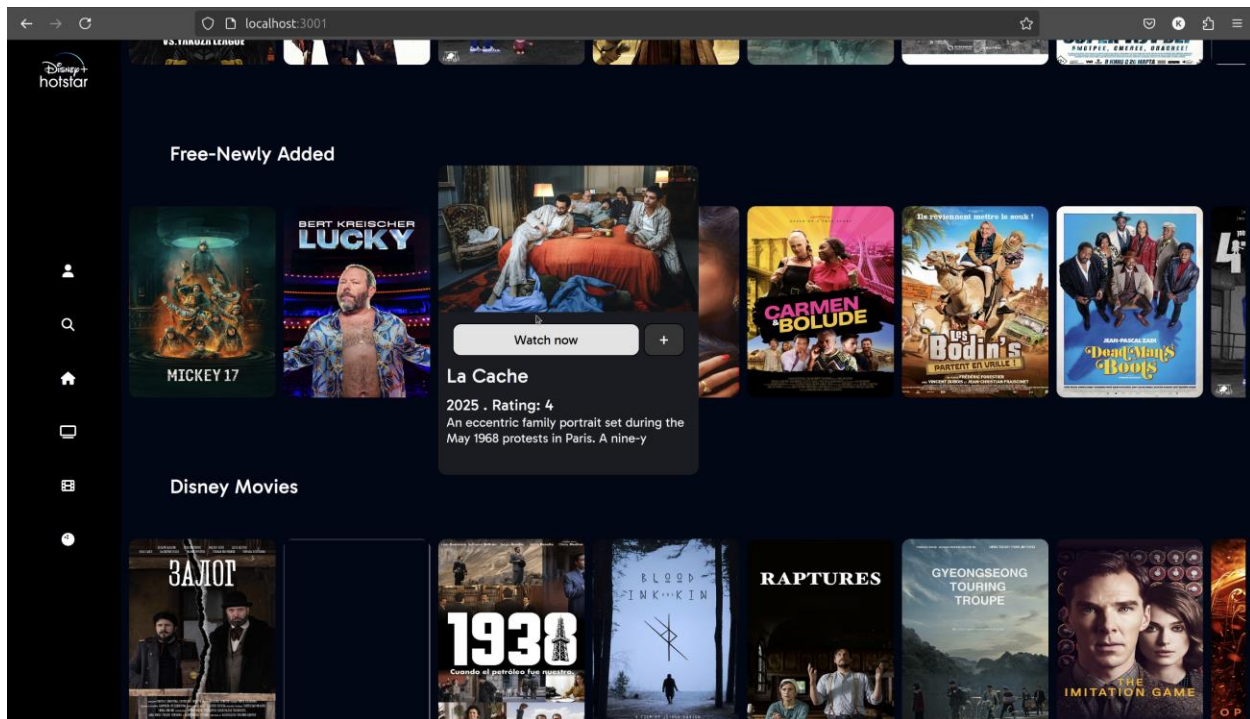
BUILDS

Filter

Today

🟢 #7 3:11 AM

REST API Jenkins 2.492.2



krithick
log out

Dashboard > last-pipe > #7

Status

</> Changes

Console Output

Edit Build Information

Delete build '#7'

Timings

Git Build Data

Pipeline Overview

Pipeline Console

Restart from Stage

Replay

Pipeline Steps

Workspaces

Previous Build

✓ Console Output
Download
Copy
View as plain text

```

Started by user krithick
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /opt/jenkins/workspace/last-pipe
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Tool Install)
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Clean Workspace)
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] withEnv
[Pipeline] {
[Pipeline] script
[Pipeline] {

```