Project 01

Project Overview

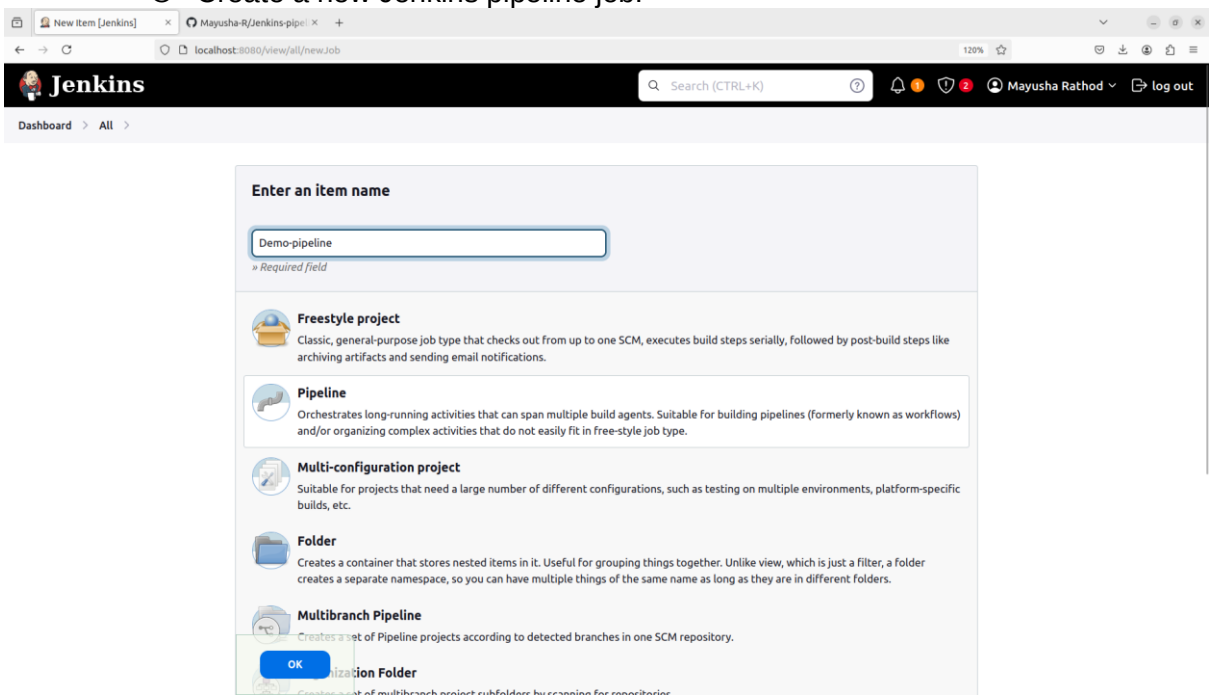
Your organization is implementing continuous integration (CI) practices to streamline the software development lifecycle. As part of this initiative, you will create a Jenkins declarative pipeline for building a simple Maven project hosted on GitHub. This project aims to automate the build process, ensure code quality, and facilitate continuous delivery (CD).

Objectives

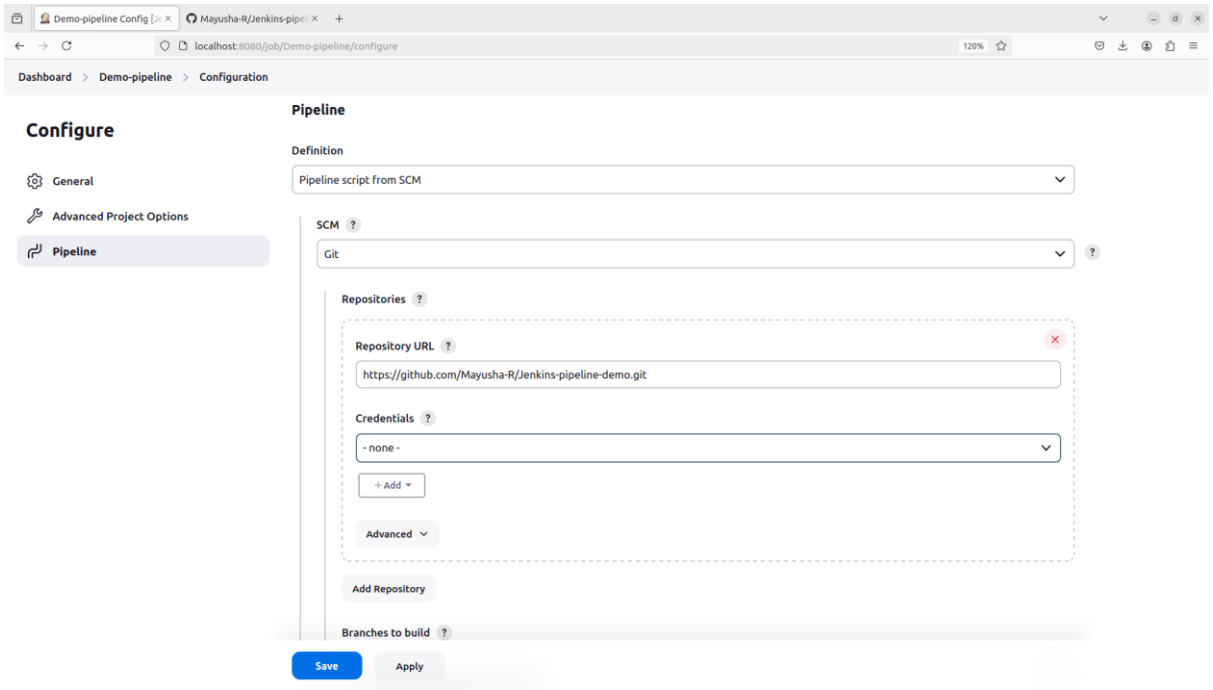
- Create a Jenkins pipeline script using declarative syntax.
- Clone a Maven project from a specified GitHub repository.
- Execute the build process and run unit tests.
- Archive build artifacts.
- Provide clear feedback on build status through Jenkins' UI and console output.

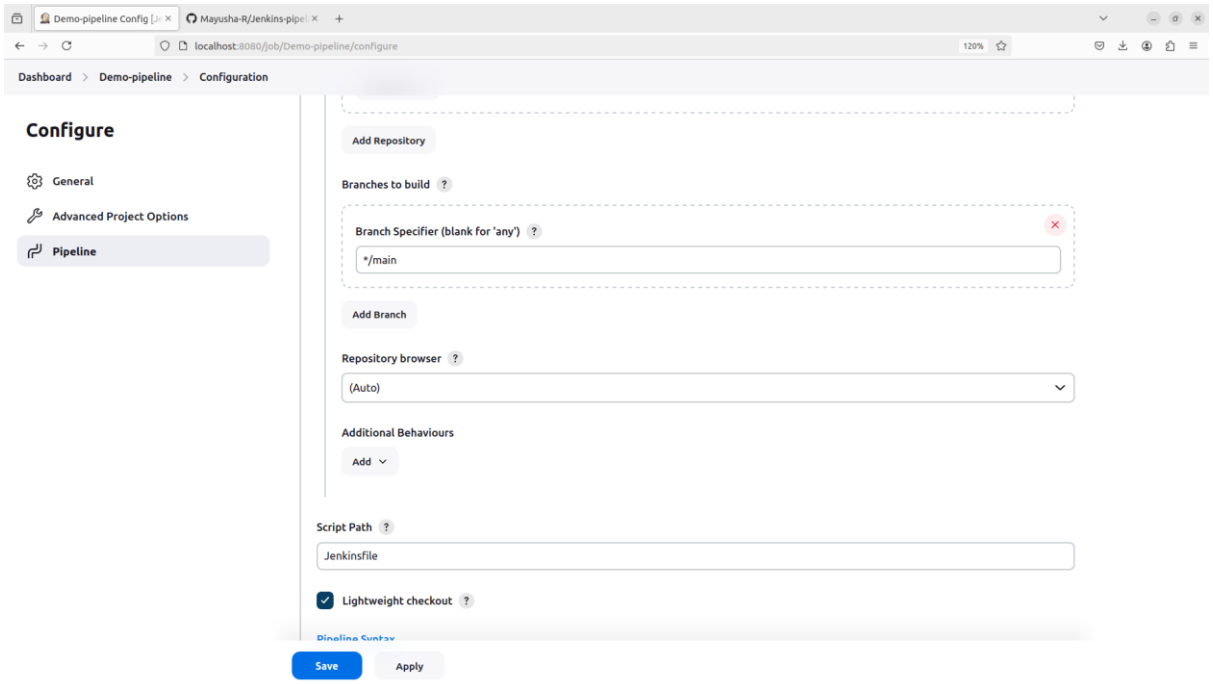
Instructions

1. **Setup Jenkins Job**
    - Create a new Jenkins pipeline job.



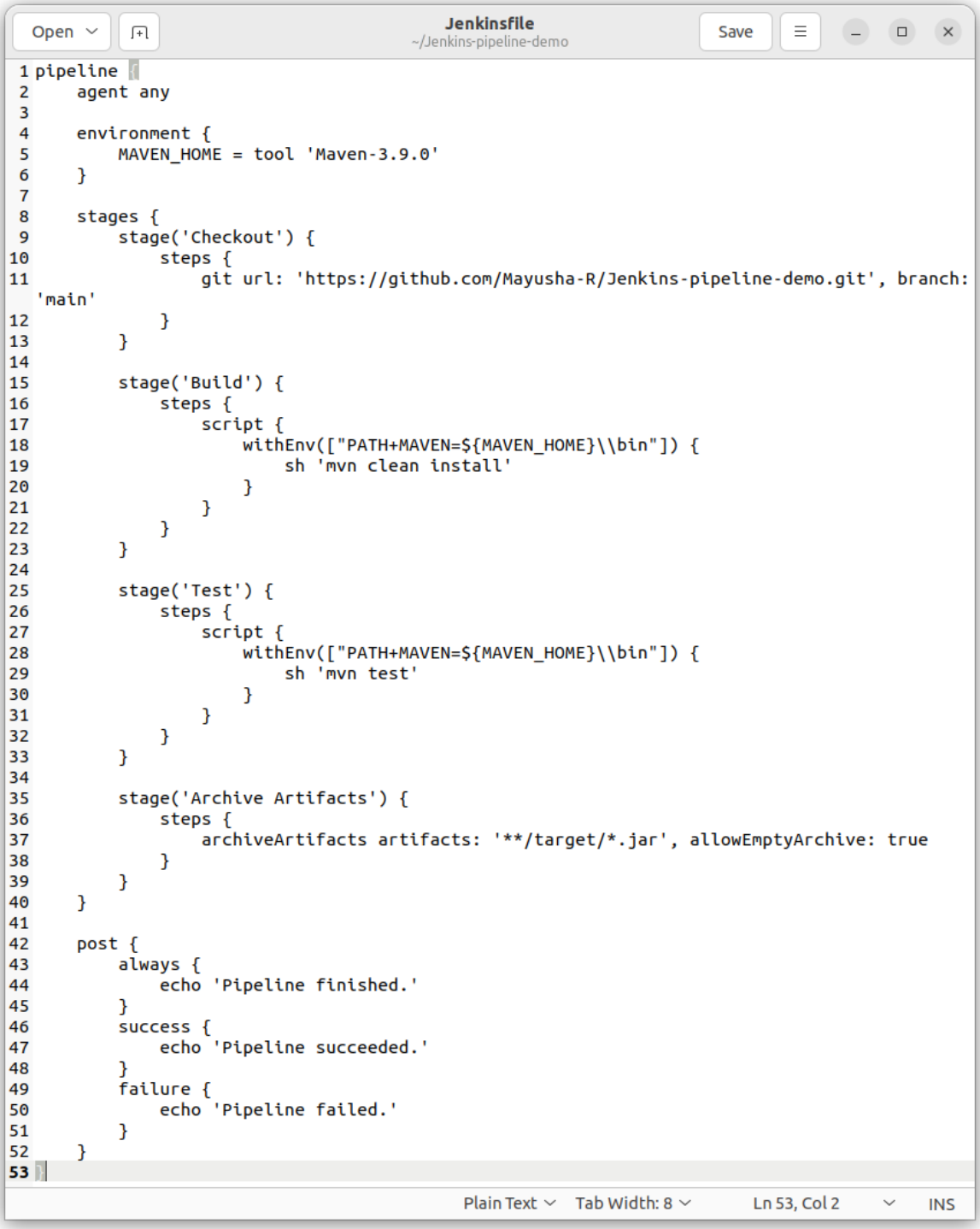
    - Configure the job to pull the Jenkinsfile from the GitHub repository.





2. **Create Jenkinsfile**
   - Write a declarative pipeline script (Jenkinsfile) that includes the following stages:
     - **Clone Repository**: Clone the Maven project from the GitHub repository.
     - **Build**: Execute the Maven build process (mvn clean install).
     - **Test**: Run unit tests as part of the Maven build.
     - **Archive Artifacts**: Archive the build artifacts for future use.

Public Repo:




```
pipeline {
    agent any

    environment {
        MAVEN_HOME = tool 'Maven-3.9.0'
    }

    stages {
        stage('Checkout') {
            steps {
                git url: 'https://github.com/Mayusha-R/Jenkins-pipeline-demo.git', branch: 'main'
            }
        }

        stage('Build') {
            steps {
                script {
                    withEnv(["PATH+MAVEN=${MAVEN_HOME}\\bin"]) {
                        sh 'mvn clean install'
                    }
                }
            }
        }

        stage('Test') {
            steps {
                script {
                    withEnv(["PATH+MAVEN=${MAVEN_HOME}\\bin"]) {
                        sh 'mvn test'
                    }
                }
            }
        }

        stage('Archive Artifacts') {
            steps {
                archiveArtifacts artifacts: '**/target/*.jar', allowEmptyArchive: true
            }
        }
    }

    post {
        always {
            echo 'Pipeline finished.'
        }
        success {
            echo 'Pipeline succeeded.'
        }
        failure {
            echo 'Pipeline failed.'
        }
    }
}
```

Private Repo :

Open

Jenkinsfile
~/Jenkins-pipeline-demo

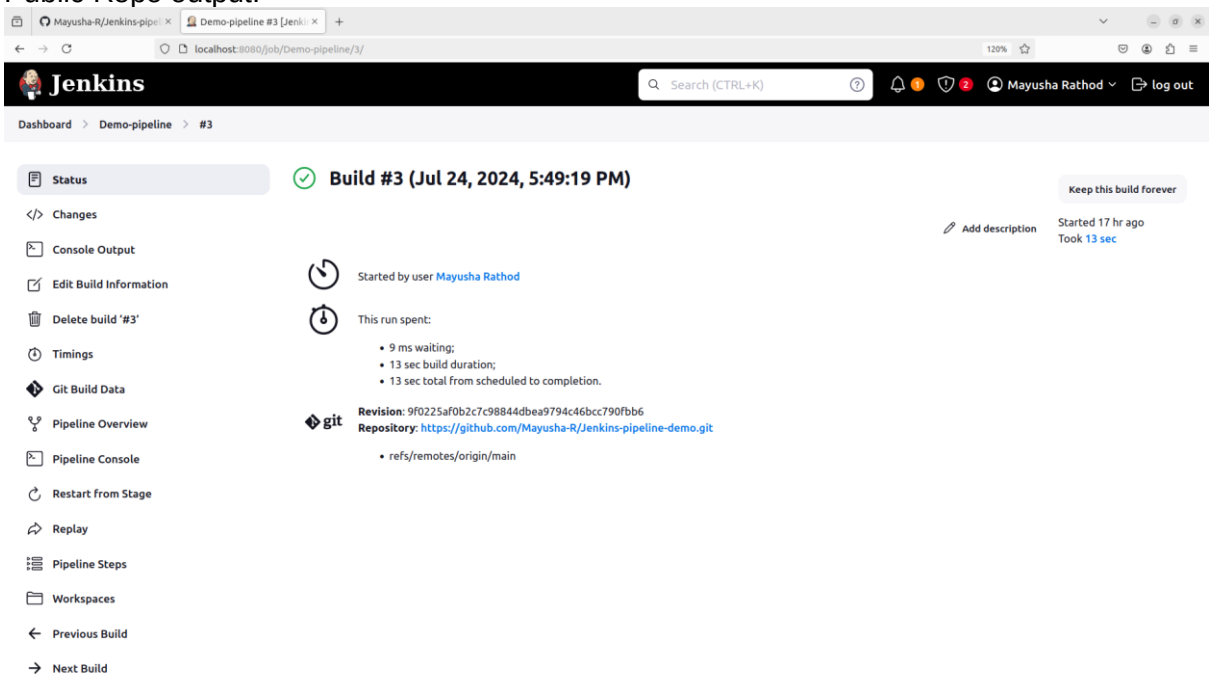
Save

```
pipeline {
    agent any

    environment {
        MAVEN_HOME = tool 'Maven-3.9.0'
    }

    stages {
        stage('Checkout') {
            steps {
                git url: 'https://github.com/Mayusha-R/Jenkins-pipeline-demo.git', branch: 'main', credentialsId: 'private-demo-cred'
            }
        }

        stage('Build') {
            steps {
                script {
                    withEnv(["PATH+MAVEN=${MAVEN_HOME}//bin"]) {
                        sh 'mvn clean install'
                    }
                }
            }
        }

        stage('Test') {
            steps {
                script {
                    withEnv(["PATH+MAVEN=${MAVEN_HOME}//bin"]) {
                        sh 'mvn test'
                    }
                }
            }
        }

        stage('Archive Artifacts') {
            steps {
                archiveArtifacts artifacts: '**/target/*.jar', allowEmptyArchive: true
            }
        }
    }

    post {
        always {
            echo 'Pipeline finished.'
        }
        success {
            echo 'Pipeline succeeded.'
        }
        failure {
            echo 'Pipeline failed.'
        }
    }
}
```

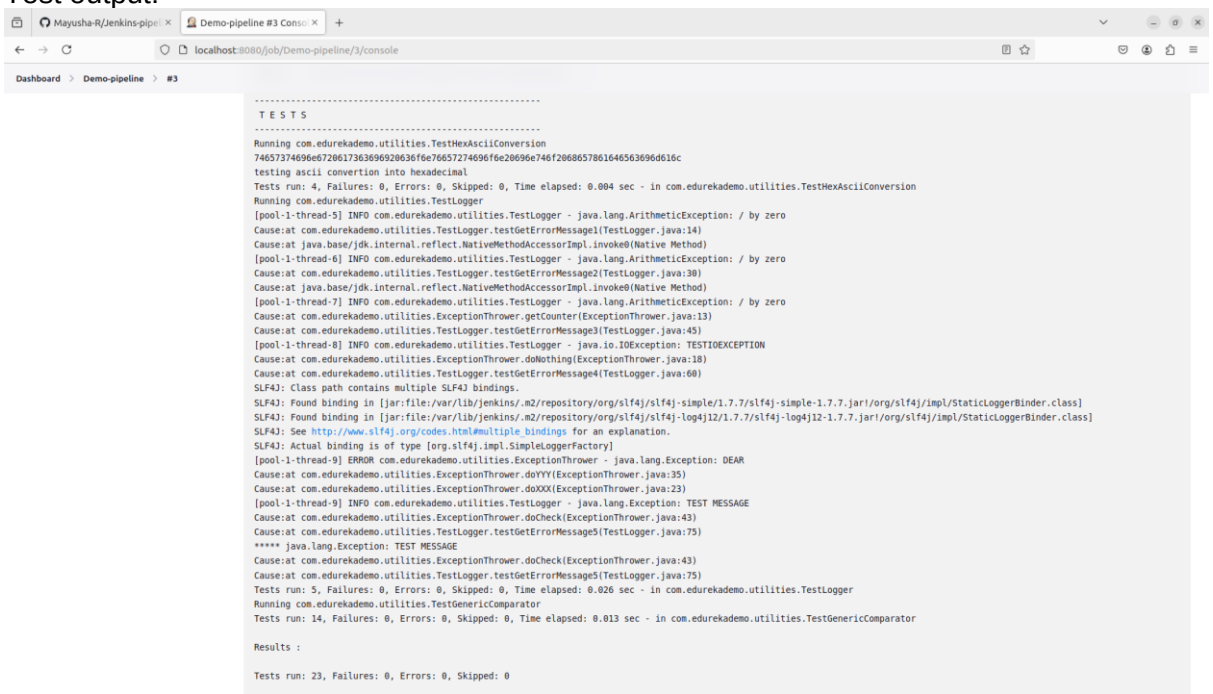
Bracket match found on line: 1 | Plain Text | Tab Width: 8 | Ln 53, Col 2 | INS

3. **Configure Pipeline Parameters**
   - Allow the pipeline to accept parameters such as Maven goals and options for flexibility.
   - Ensure the pipeline can be easily modified for different build configurations.
4. **Run the Pipeline**
   - Trigger the Jenkins pipeline job manually or set up a webhook for automatic triggering on GitHub repository changes.
   - Monitor the build process through Jenkins' UI and console output.

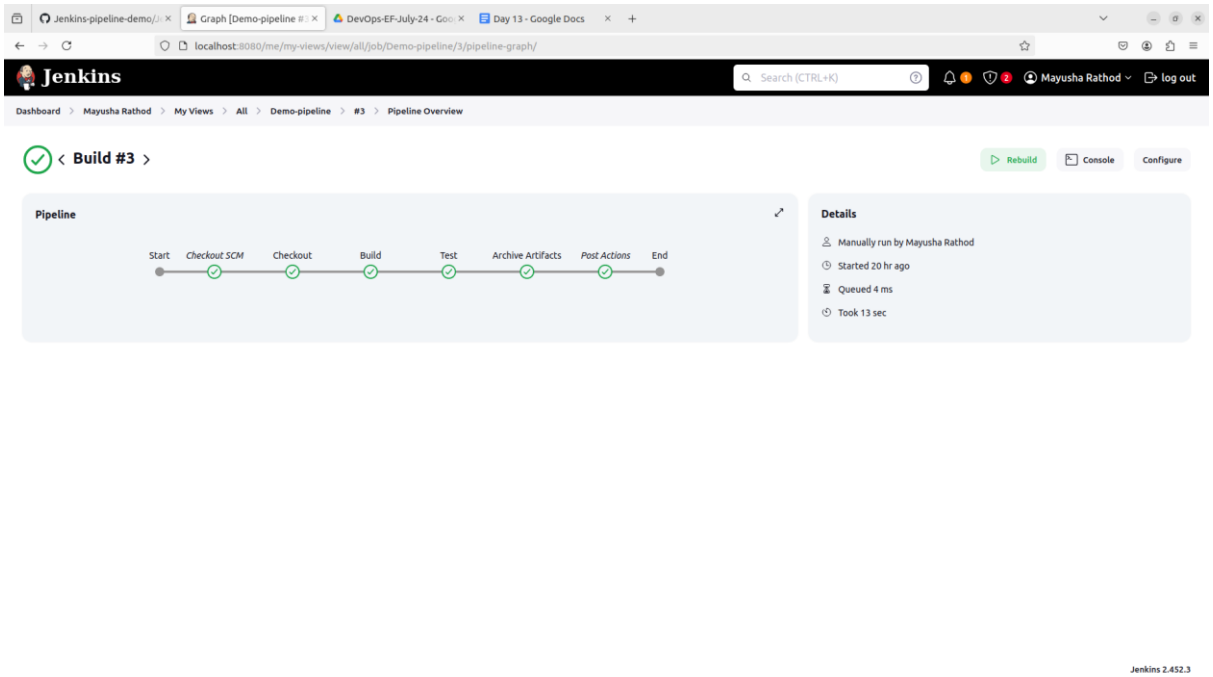
Public Repo output:

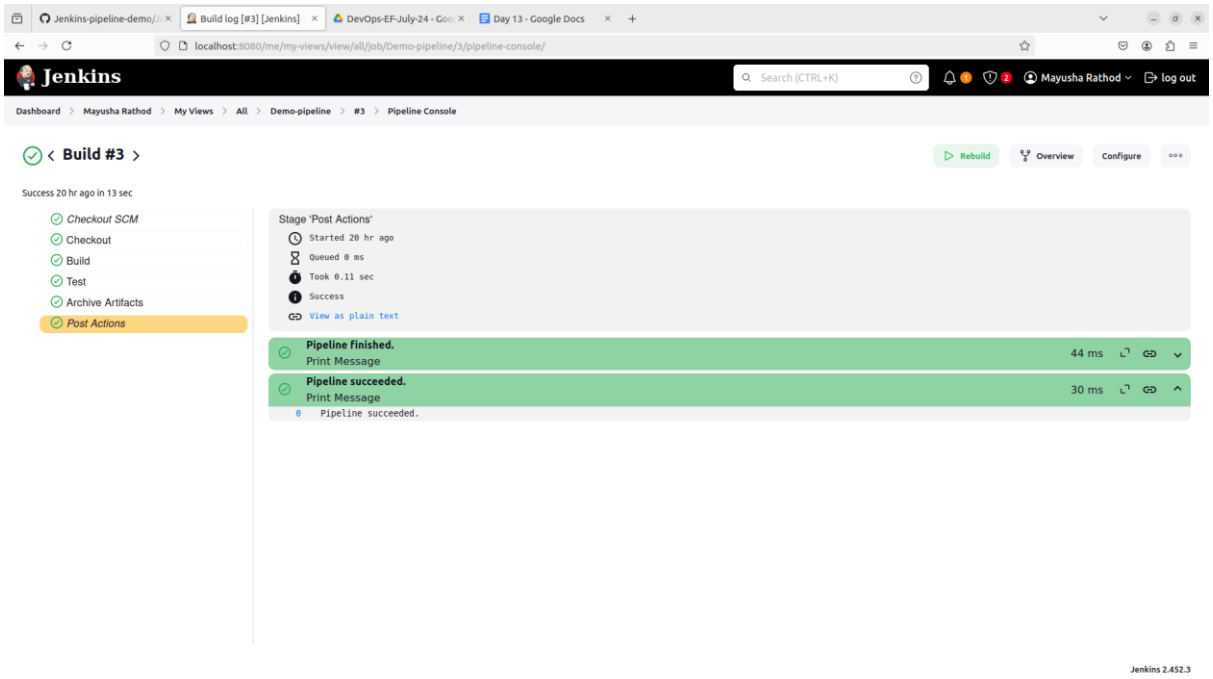


Test output:

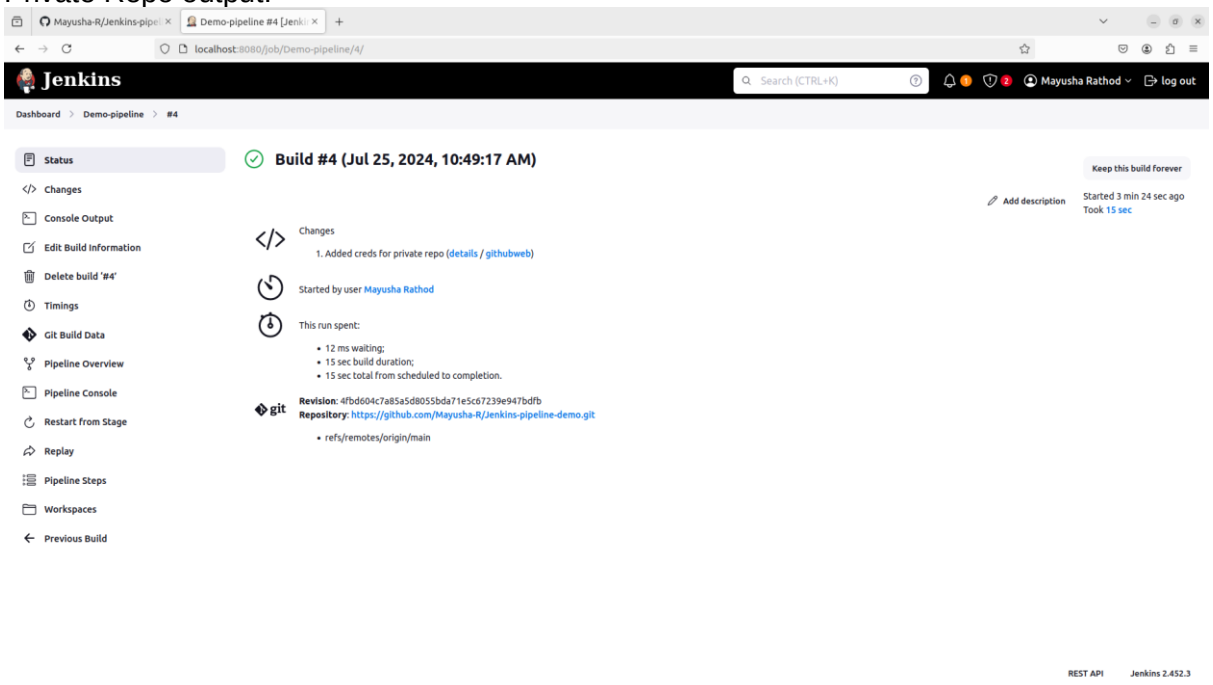


Pipeline overview:

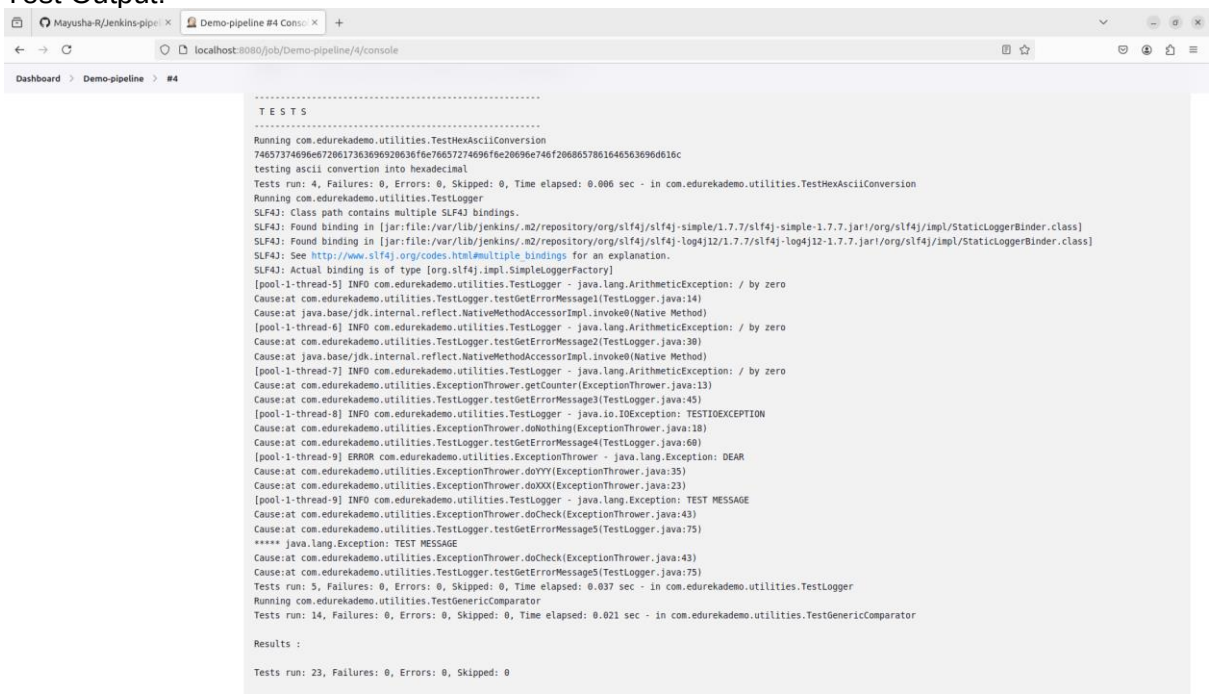

Jenkins-pipeline-demo/
Graph [Demo-pipeline
DevOps-EF-July-24 - Goo
Day 13 - Google Docs
localhost:8080/me/my-views/view/all/job/Demo-pipeline/3/pipeline-graph/
Jenkins
Search (CTRL+K)
Mayusha Rathod
log out
Dashboard
Mayusha Rathod
My Views
All
Demo-pipeline
#3
Pipeline Overview
Build #3
Rebuild
Console
Configure
Pipeline
Start
Checkout SCM
Checkout
Build
Test
Archive Artifacts
Post Actions
End
Details
Manually run by Mayusha Rathod
Started 20 hr ago
Queued 4 ms
Took 13 sec
Jenkins 2.452.3

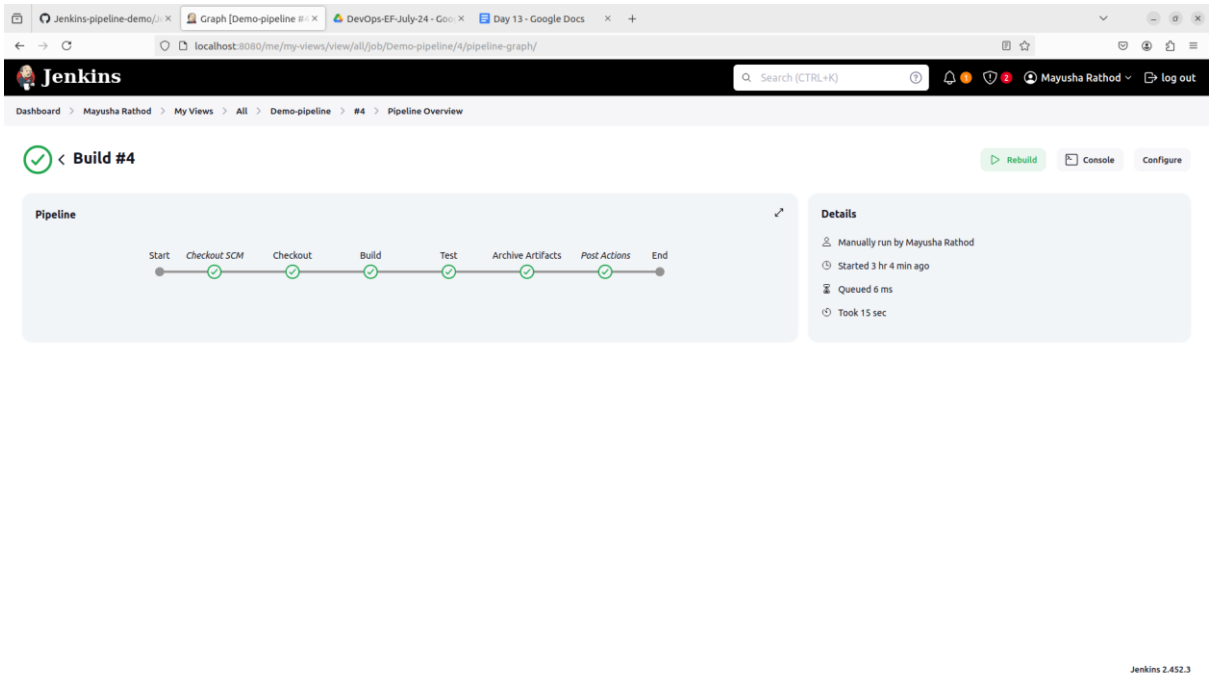

Jenkins-pipeline-demo/
Build log [#3] [Jenkins]
DevOps-EF-July-24 - Goo
Day 13 - Google Docs
localhost:8080/me/my-views/view/all/job/Demo-pipeline/3/pipeline-console/
Jenkins
Search (CTRL+K)
Mayusha Rathod
log out
Dashboard
Mayusha Rathod
My Views
All
Demo-pipeline
#3
Pipeline Console
Build #3
Rebuild
Overview
Configure
Success 20 hr ago in 13 sec
Checkout SCM
Checkout
Build
Test
Archive Artifacts
Post Actions
Stage 'Post Actions'
Started 20 hr ago
Queued 0 ms
Took 0.11 sec
Success
View as plain text
Pipeline finished.
Print Message
44 ms
Pipeline succeeded.
Print Message
30 ms
0 Pipeline succeeded.
Jenkins 2.452.3

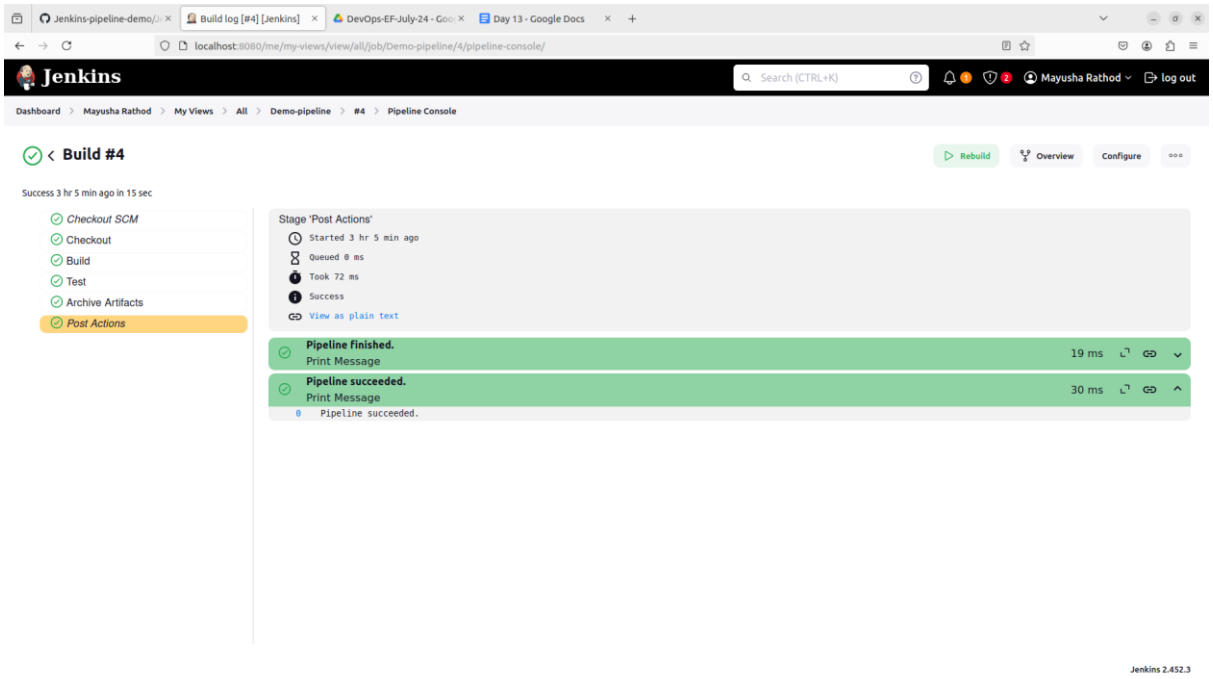
Private Repo output:



Test Output:




Jenkins-pipeline-demo/
Graph [Demo-pipeline
DevOps-EF-July-24 - Goo
Day 13 - Google Docs
localhost:8080/me/my-views/view/all/job/Demo-pipeline/4/pipeline-graph/
Jenkins
Search (CTRL+K)
Mayusha Rathod
log out
Dashboard
Mayusha Rathod
My Views
All
Demo-pipeline
#4
Pipeline Overview
Build #4
Rebuild
Console
Configure
Pipeline
Start
Checkout SCM
Checkout
Build
Test
Archive Artifacts
Post Actions
End
Details
Manually run by Mayusha Rathod
Started 3 hr 4 min ago
Queued 6 ms
Took 15 sec
Jenkins 2.452.3


Jenkins-pipeline-demo/
Build log [#4] [Jenkins]
DevOps-EF-July-24 - Goo
Day 13 - Google Docs
localhost:8080/me/my-views/view/all/job/Demo-pipeline/4/pipeline-console/
Jenkins
Search (CTRL+K)
Mayusha Rathod
log out
Dashboard
Mayusha Rathod
My Views
All
Demo-pipeline
#4
Pipeline Console
Build #4
Rebuild
Overview
Configure
Success 3 hr 5 min ago in 15 sec
Checkout SCM
Checkout
Build
Test
Archive Artifacts
Post Actions
Stage 'Post Actions'
Started 3 hr 5 min ago
Queued 0 ms
Took 72 ms
Success
View as plain text
Pipeline finished.
Print Message
19 ms
Pipeline succeeded.
Print Message
30 ms
0 Pipeline succeeded.
Jenkins 2.452.3