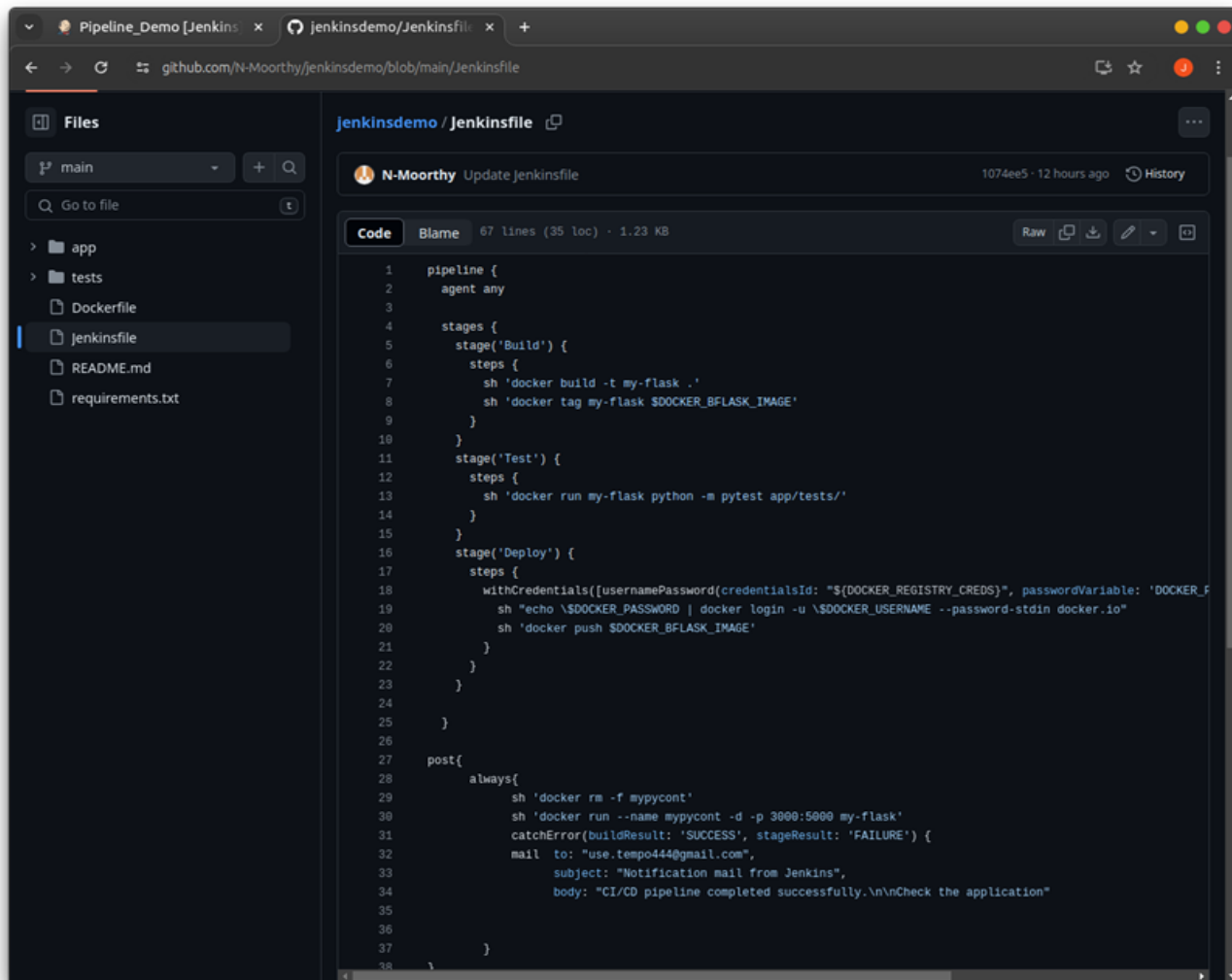
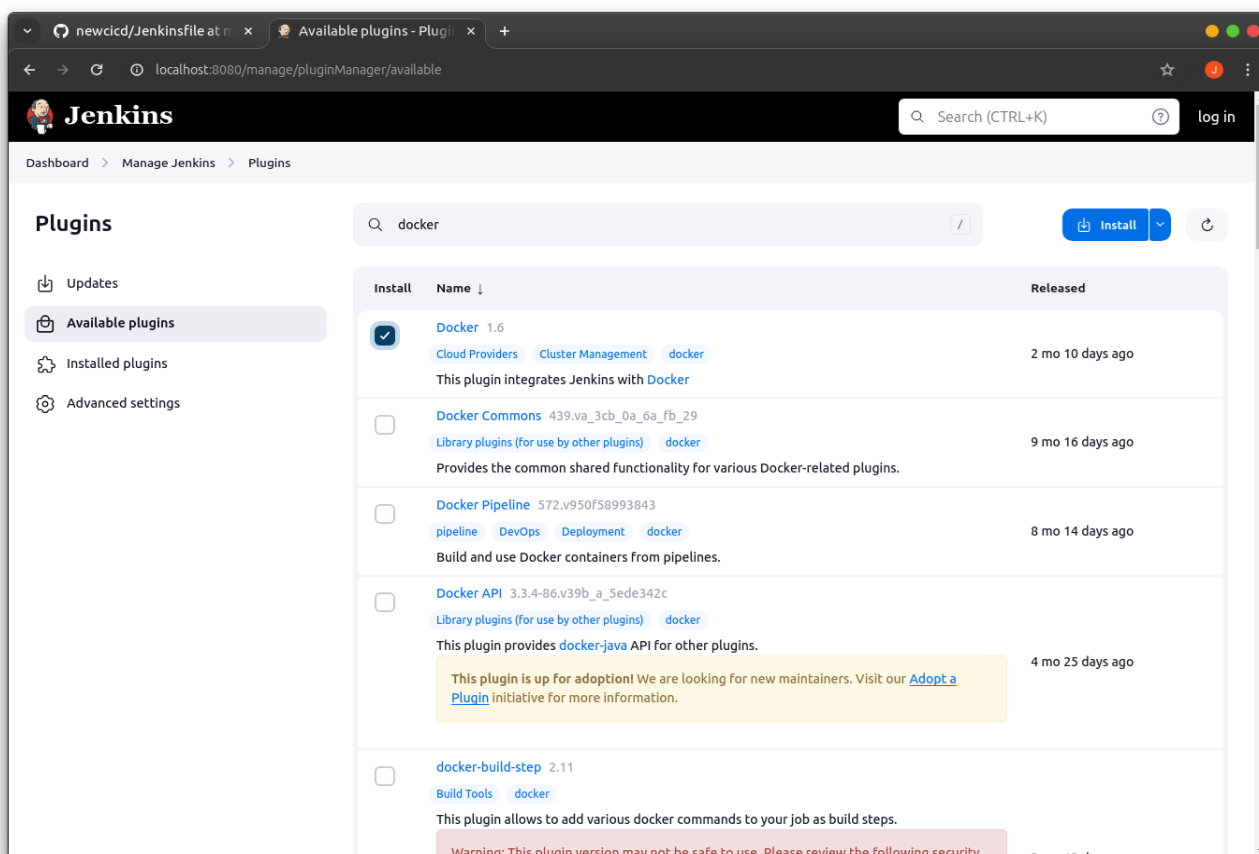


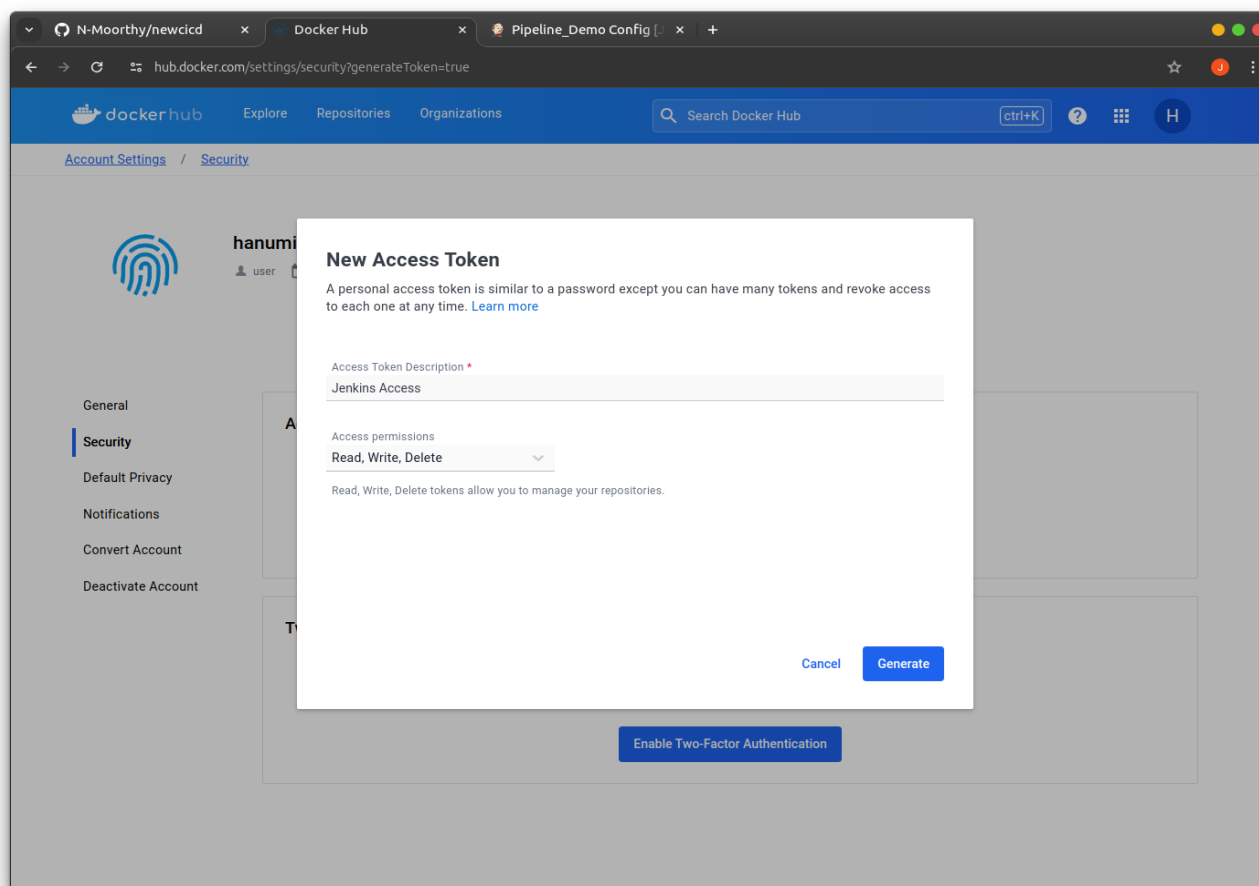
1. Jenkins File



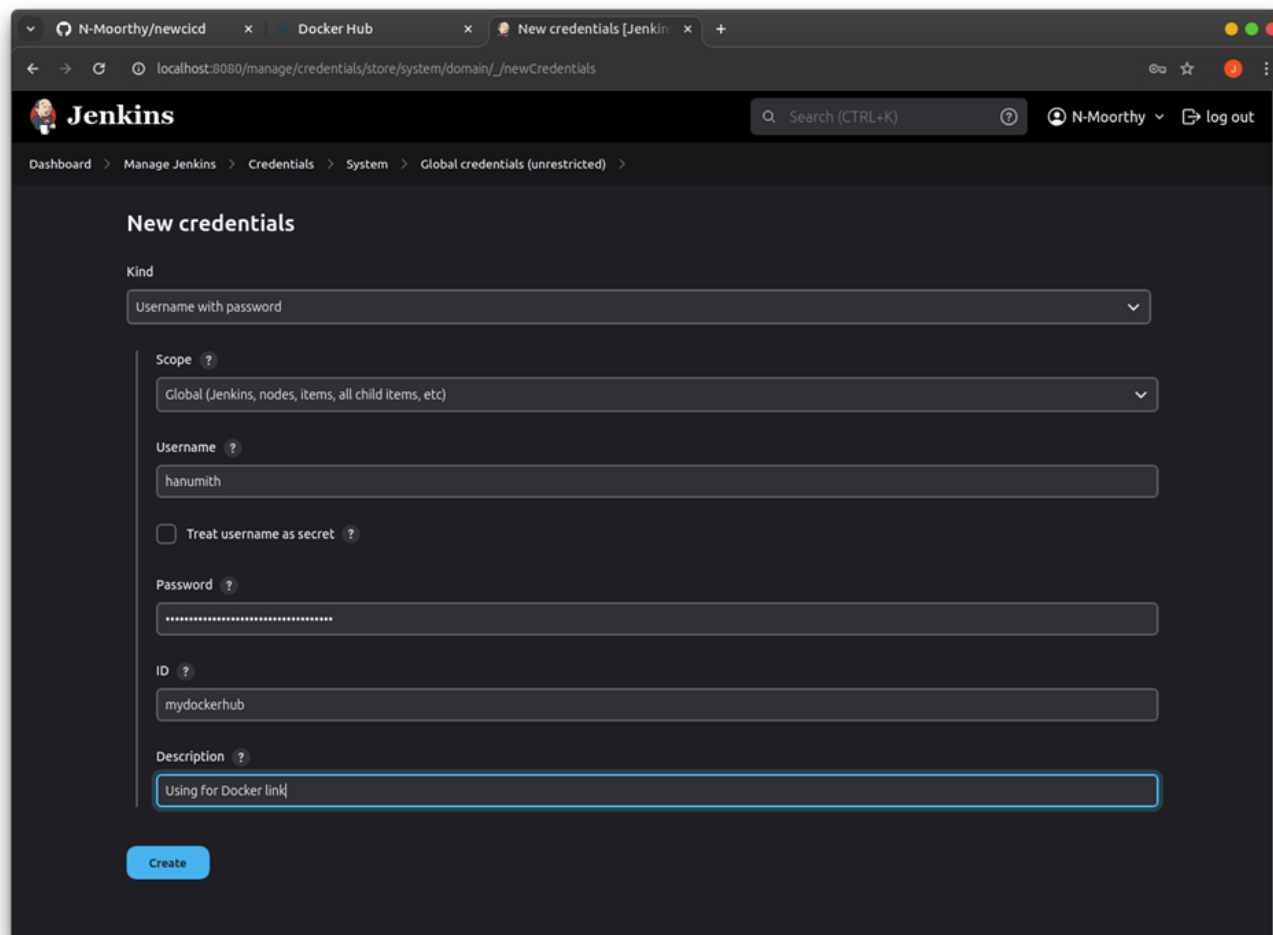
2. Docker Plugin



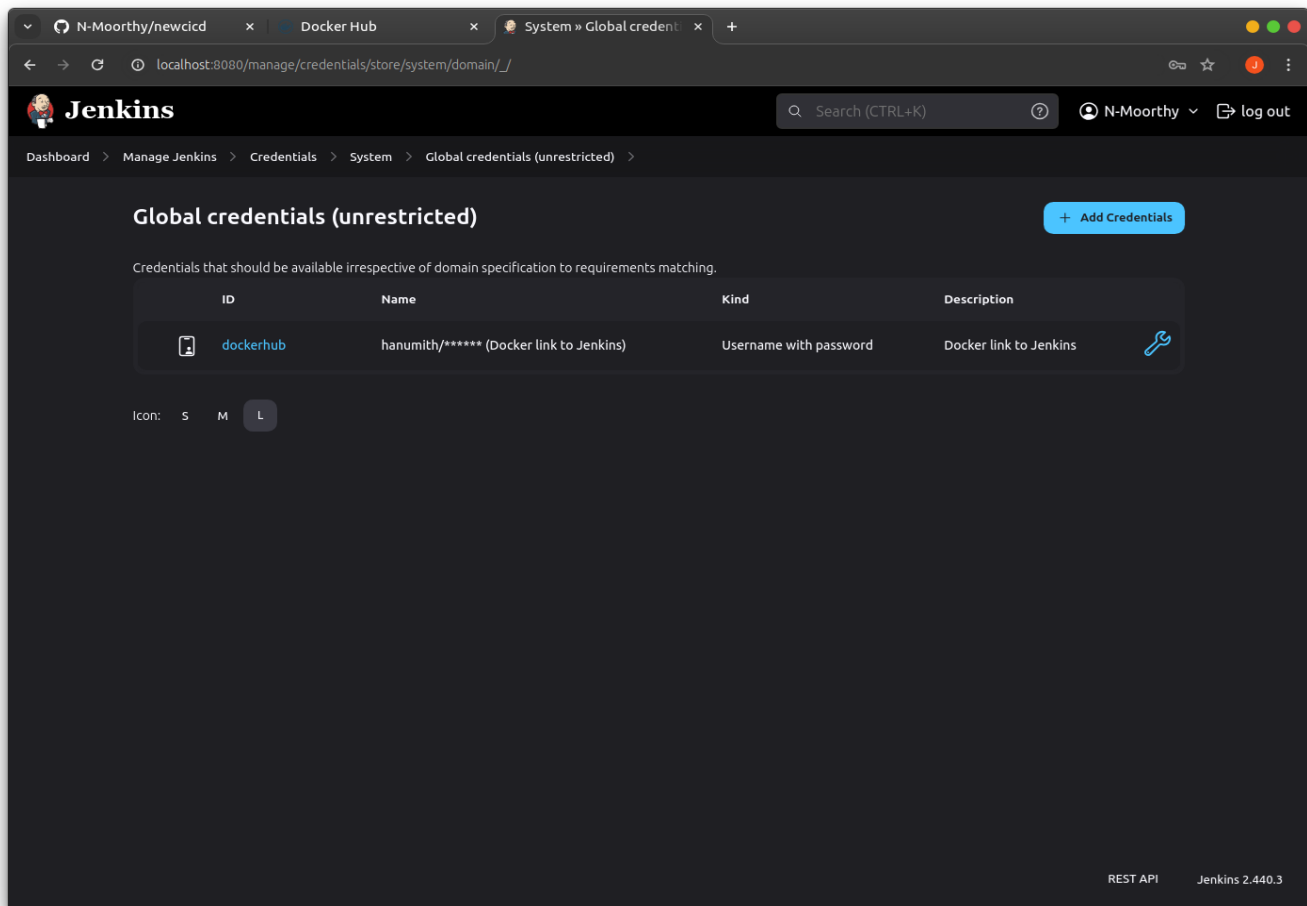
3.Access Token for DockerHub



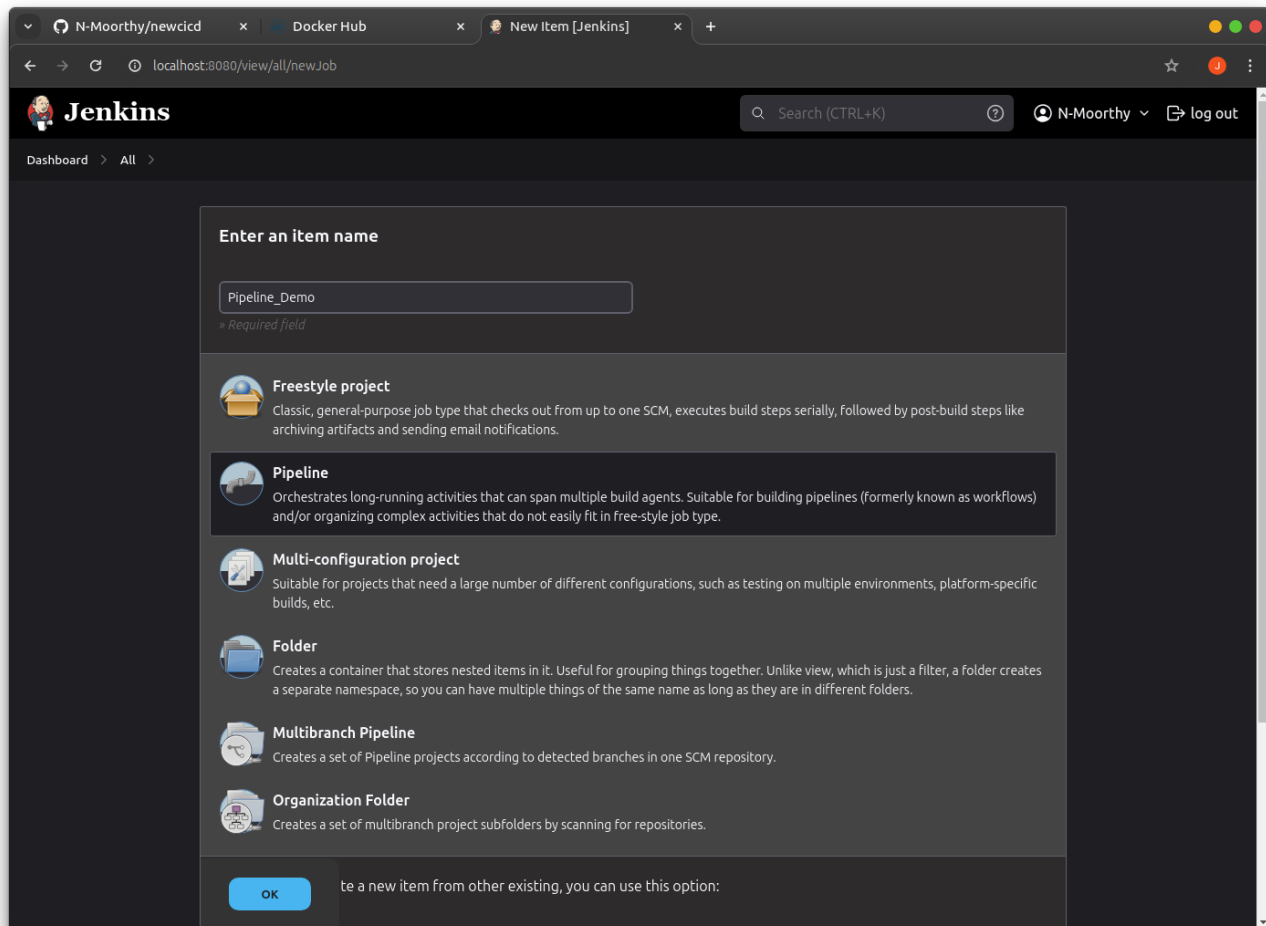
4.Integrating Docker Credentials in Jenkins



5.Docker Creds in Jenkins



6.Creating Pipeline



7. Linking my Git Repo to Pipeline

The screenshot shows the Jenkins Pipeline configuration page for a job named 'Pipeline_Demo'. The 'General' tab is selected. The 'Description' field contains 'Jenkins Pipeline Demo'. The 'GitHub project' checkbox is checked, and the 'Project url' is set to 'https://github.com/N-Moorthy/jenkinsdemo.git/'. The 'Enabled' toggle is turned on. The 'Save' button is highlighted.

Configure

- General
- Advanced Project Options
- Pipeline

General Enabled ☒

Description

Jenkins Pipeline Demo

Plain text: [Preview](#)

☐ Discard old builds ?

☐ Do not allow concurrent builds

☐ Do not allow the pipeline to resume if the controller restarts

☒ GitHub project

Project url ?

https://github.com/N-Moorthy/jenkinsdemo.git/

Advanced ▾

☐ Pipeline speed/durability override ?

☐ Preserve stashes from completed builds ?

☐ This project is parameterised ?

☐ Branches to build ?

Save Apply

8. Pipeline Script with my Git Repo

The screenshot shows the Jenkins Pipeline configuration page for the same job, but the 'Pipeline' tab is selected. The 'Definition' is set to 'Pipeline script from SCM'. The 'SCM' is set to 'Git'. The 'Repository URL' is 'https://github.com/N-Moorthy/jenkinsdemo.git'. The 'Credentials' are set to 'none'. The 'Branches to build' section is expanded, showing the 'Branch Specifier (blank for 'any')' set to '*/main'. The 'Save' button is highlighted.

Configure

- General
- Advanced Project Options
- Pipeline

Pipeline

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://github.com/N-Moorthy/jenkinsdemo.git

Credentials ?

- none -

+ Add ▾

Advanced ▾

Add Repository

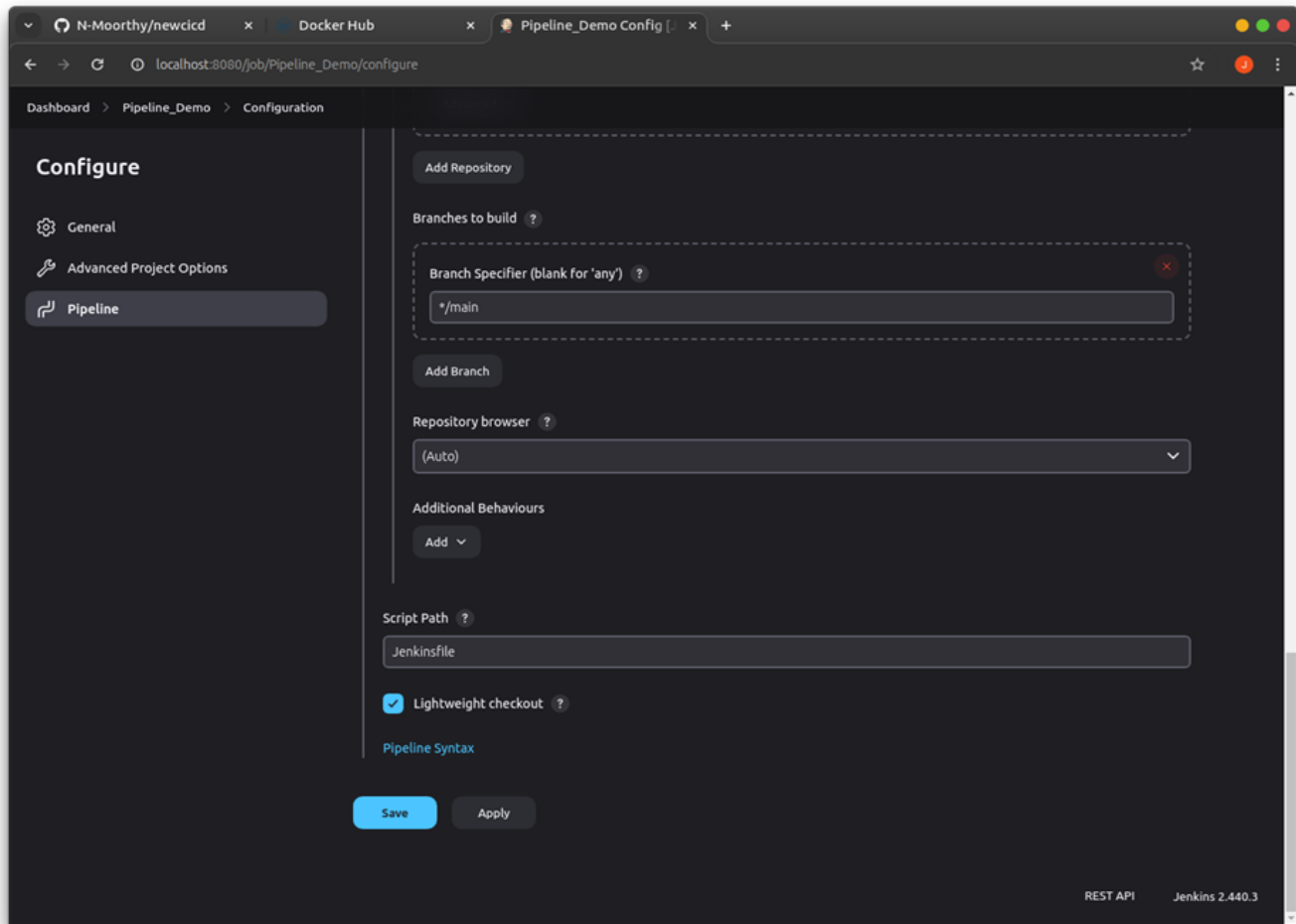
Branches to build ?

Branch Specifier (blank for 'any') ?

*/main

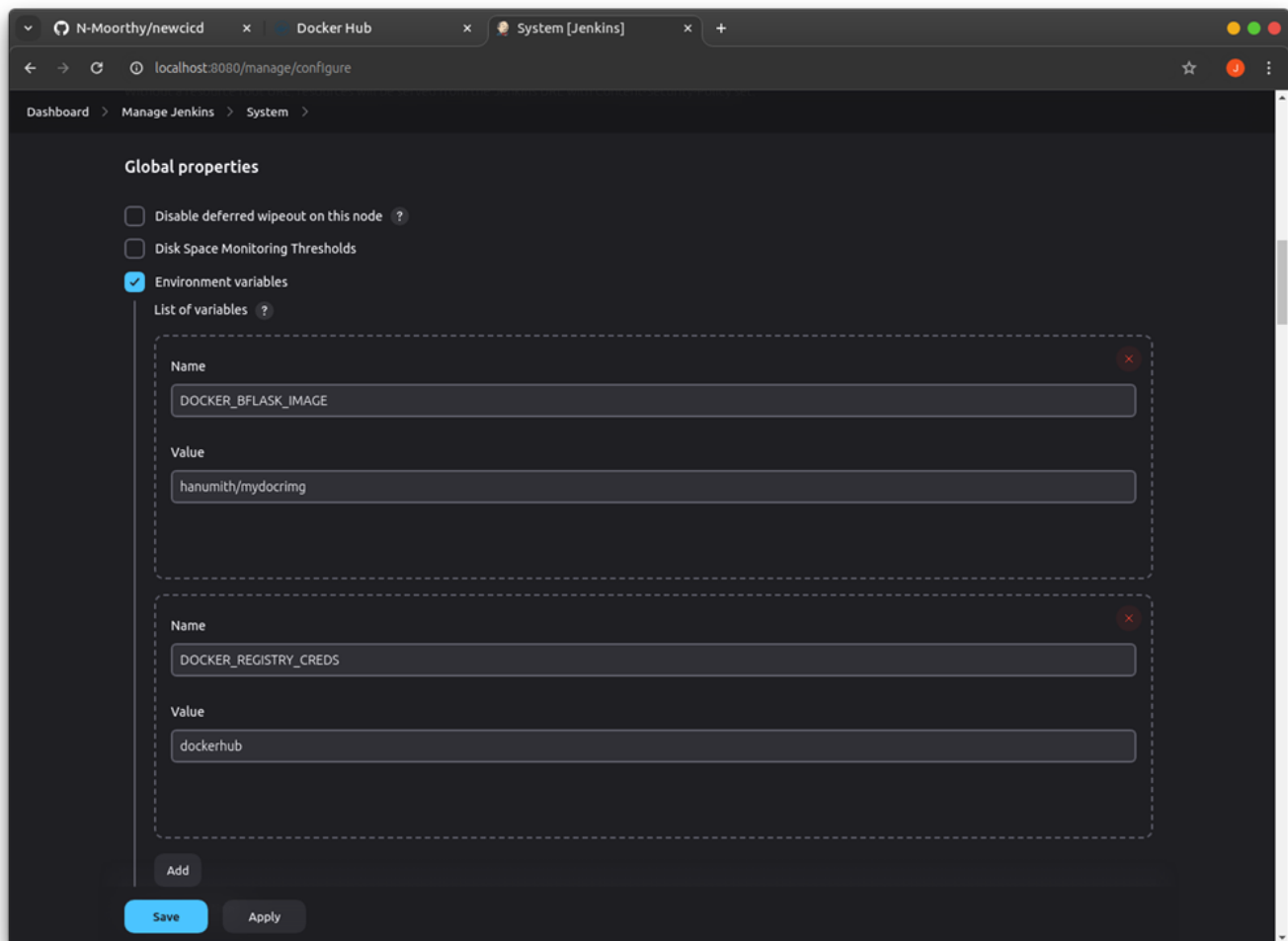
Save Apply

9. Selecting Script File Path



The screenshot shows the Jenkins Pipeline configuration page for a job named 'Pipeline_Demo'. The left sidebar has three tabs: 'General', 'Advanced Project Options', and 'Pipeline', with 'Pipeline' selected. The main area is titled 'Configure' and contains several sections: 'Add Repository' (a button), 'Branches to build' (a section with a 'Branch Specifier (blank for 'any')' field containing '/main' and an 'Add Branch' button), 'Repository browser' (a dropdown menu set to '(Auto)'), 'Additional Behaviours' (an 'Add' button), 'Script Path' (a text field containing 'Jenkinsfile'), and 'Lightweight checkout' (a checked checkbox). At the bottom are 'Save' and 'Apply' buttons. The footer shows 'REST API' and 'Jenkins 2.440.3'.

10. Setting Environment Variable



The screenshot shows the Jenkins System configuration page. The left sidebar has three tabs: 'Manage Jenkins', 'System', and 'Tools', with 'System' selected. The main area is titled 'Global properties' and contains three checkboxes: 'Disable deferred wipeout on this node', 'Disk Space Monitoring Thresholds', and 'Environment variables', with 'Environment variables' checked. Below this is a 'List of variables' section with two variable entries. The first entry has 'Name' 'DOCKER_BFLASK_IMAGE' and 'Value' 'hanumith/mydoctring'. The second entry has 'Name' 'DOCKER_REGISTRY_CREDS' and 'Value' 'dockerhub'. At the bottom are 'Add', 'Save', and 'Apply' buttons.

11. Triggered from Jenkins & Done

The screenshot shows the Jenkins Pipeline Demo dashboard. The left sidebar contains navigation links: Status, Changes, Build Now, Configure, Delete Pipeline, Full Stage View, GitHub, Rename, and Pipeline Syntax. The main area displays the Pipeline_Demo status as 'Jenkins Pipeline Demo'. Below this, the Stage View shows a progress bar for the pipeline stages: Declarative: Checkout SCM (6s), Build (1min 1s), Test (3s), Deploy (32s), and Declarative: Post Actions (7s). The average stage time is 6s, and the average full run time is ~2min 0s. The Build History section shows a single build (#2) from Apr 24, 18:48, with a status of 'No Changes'. The Permalinks section lists four links: Last build (#2), Last stable build (#2), Last successful build (#2), and Last completed build (#2), all with a time of 8 min 43 sec ago. The bottom right corner shows 'REST API' and 'Jenkins 2.440.3'.

Declarative: Checkout SCM	Build	Test	Deploy	Declarative: Post Actions
6s	1min 1s	3s	32s	7s

Average stage times: 6s
(Average full run time: ~2min 0s)

Build History: #2, 24 Apr 2024, 18:48, No Changes

Permalinks:

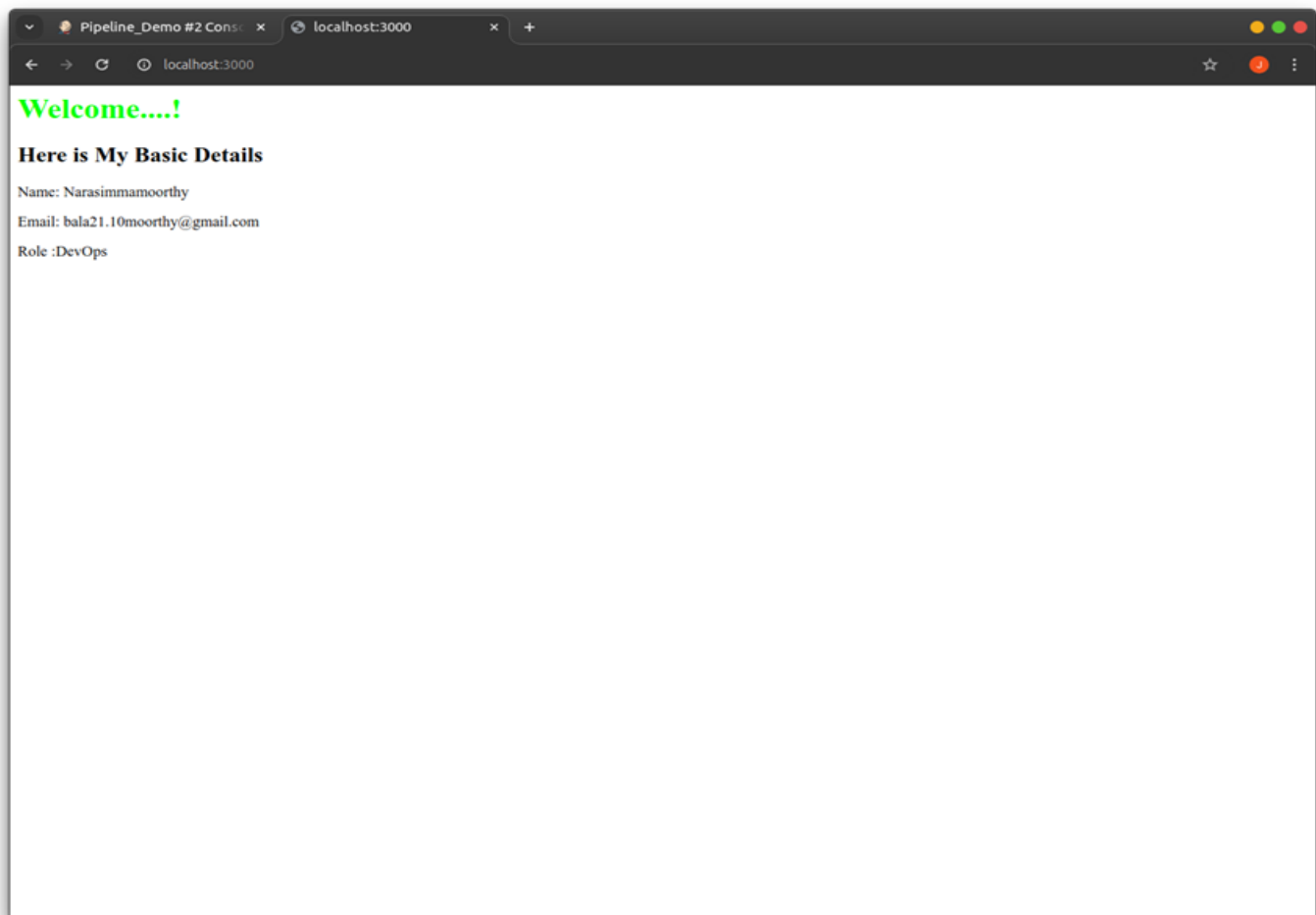
- Last build (#2), 8 min 43 sec ago
- Last stable build (#2), 8 min 43 sec ago
- Last successful build (#2), 8 min 43 sec ago
- Last completed build (#2), 8 min 43 sec ago

12-Build Console Output

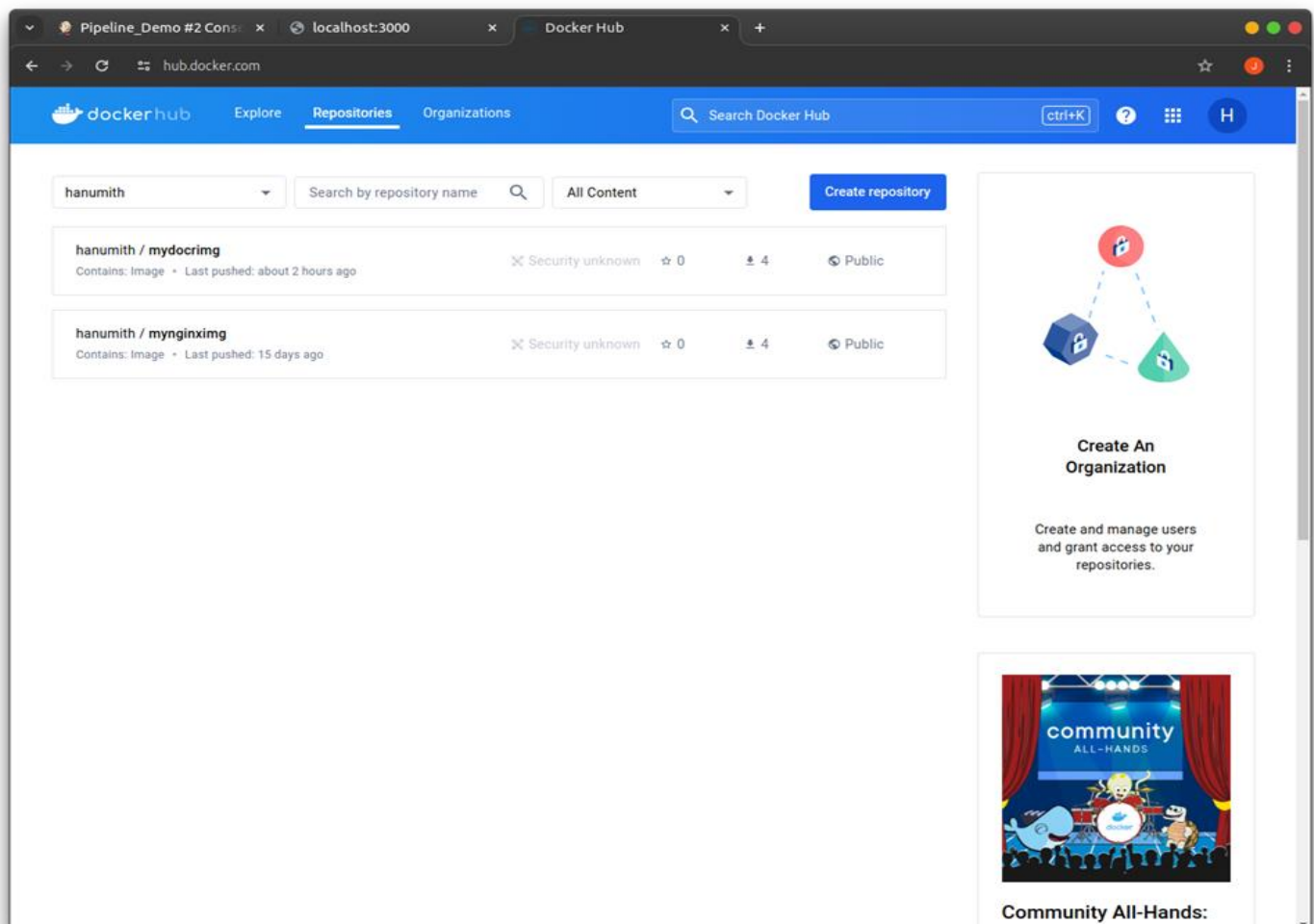
The screenshot shows the Jenkins Pipeline Demo console output for build #2. The left sidebar contains navigation links: Status, Changes, Console Output, View as plain text, Edit Build Information, Delete build '#2', Git Build Data, Restart from Stage, Replay, Pipeline Steps, and Workspaces. The main area displays the Console Output for build #2, showing the following text:

```
Started by user N-Moorthy
Obtained Jenkinsfile from git https://github.com/N-Moorthy/jenkinsdemo.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Pipeline_Demo
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Pipeline_Demo/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/N-Moorthy/jenkinsdemo.git # timeout=10
Fetching upstream changes from https://github.com/N-Moorthy/jenkinsdemo.git
> git --version # timeout=10
> git --version # 'git version 2.40.1'
> git fetch --tags --force --progress -- https://github.com/N-Moorthy/jenkinsdemo.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision c0f3980853a53613fcd6c4392f47150a6468f139 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f c0f3980853a53613fcd6c4392f47150a6468f139 # timeout=10
Commit message: "Jen"
First time build. Skipping changelog.
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] sh
+ docker build -t my-flask .
```

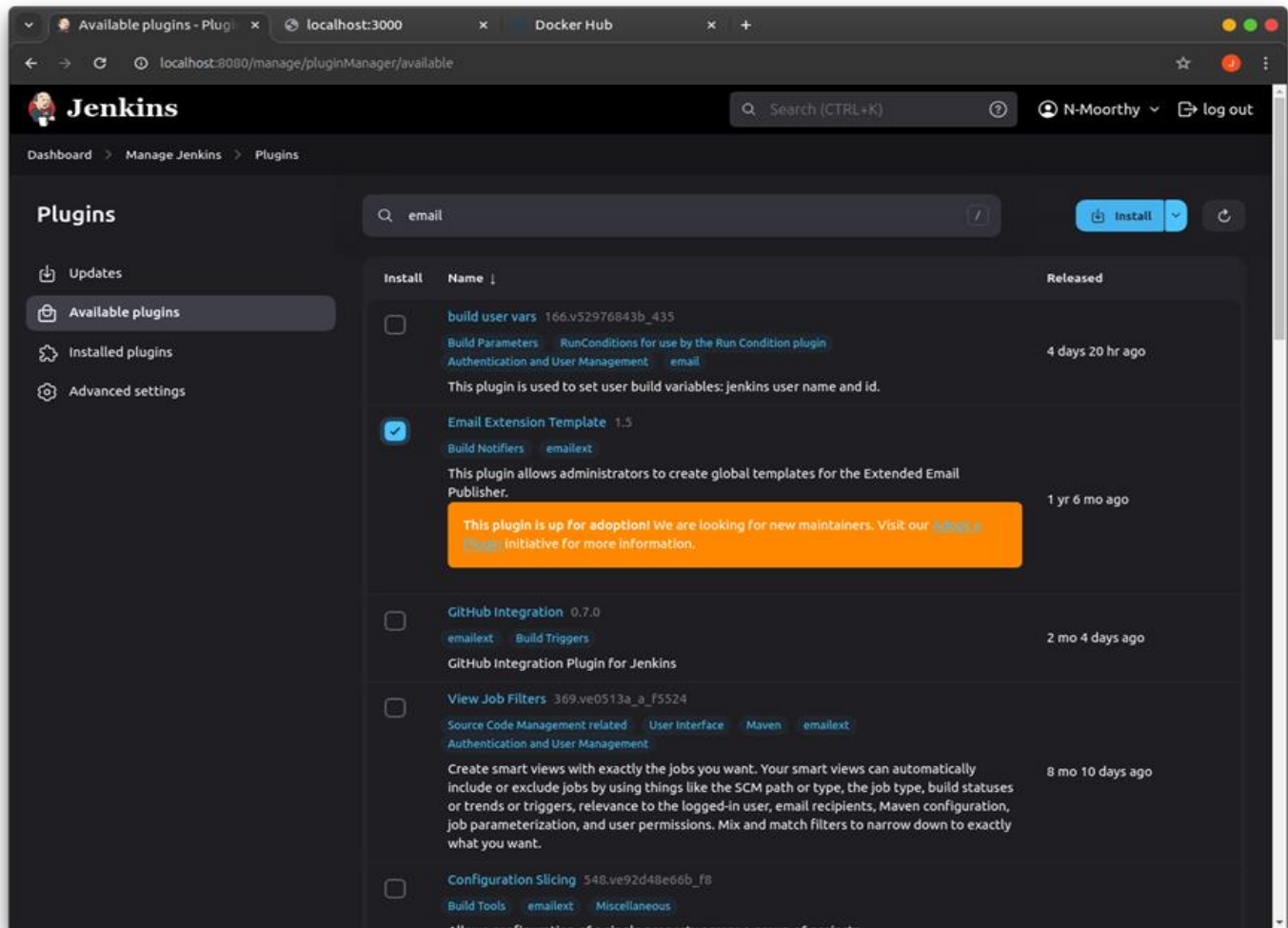
13.App Output



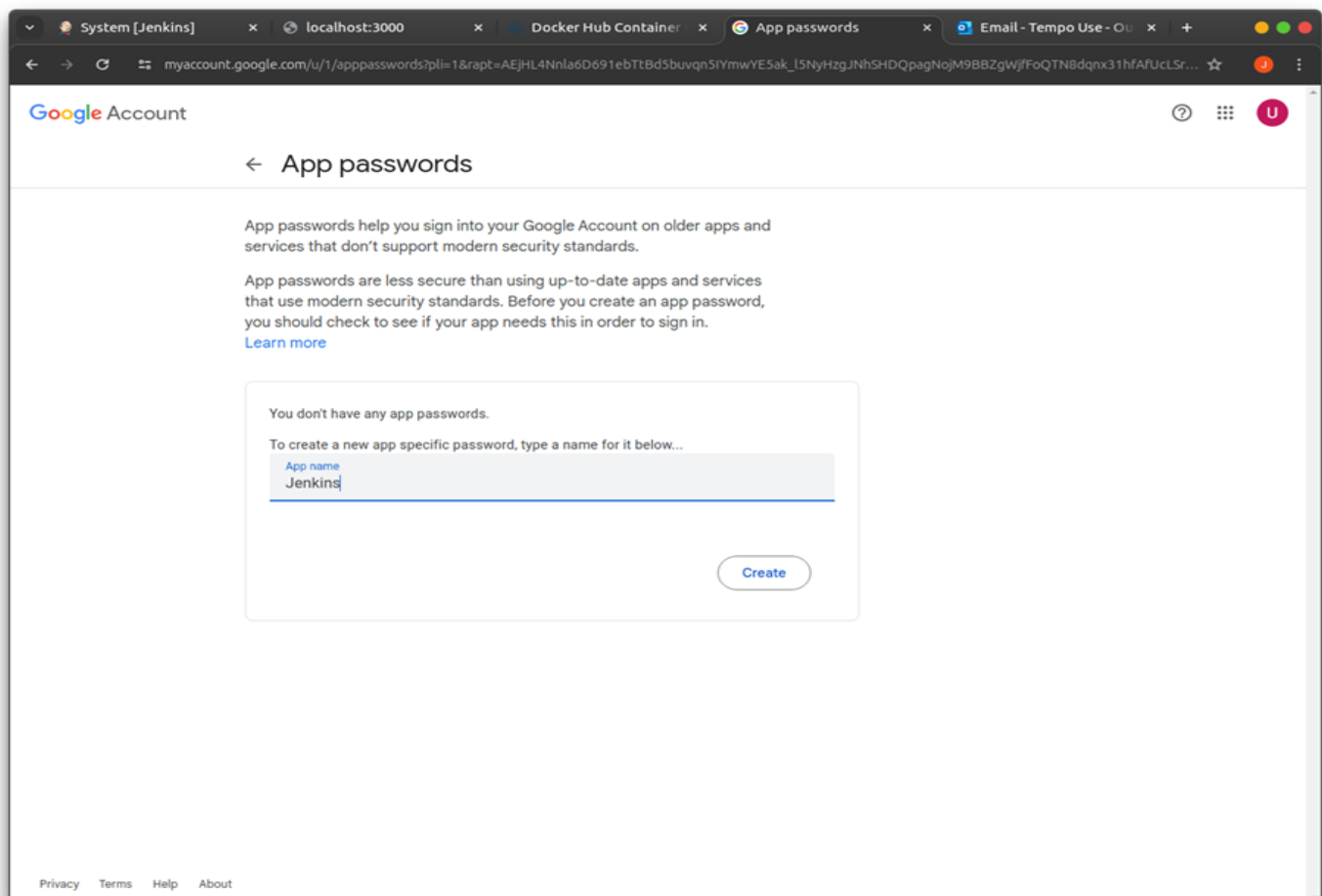
14-Docker image pushed to My Docker Repo



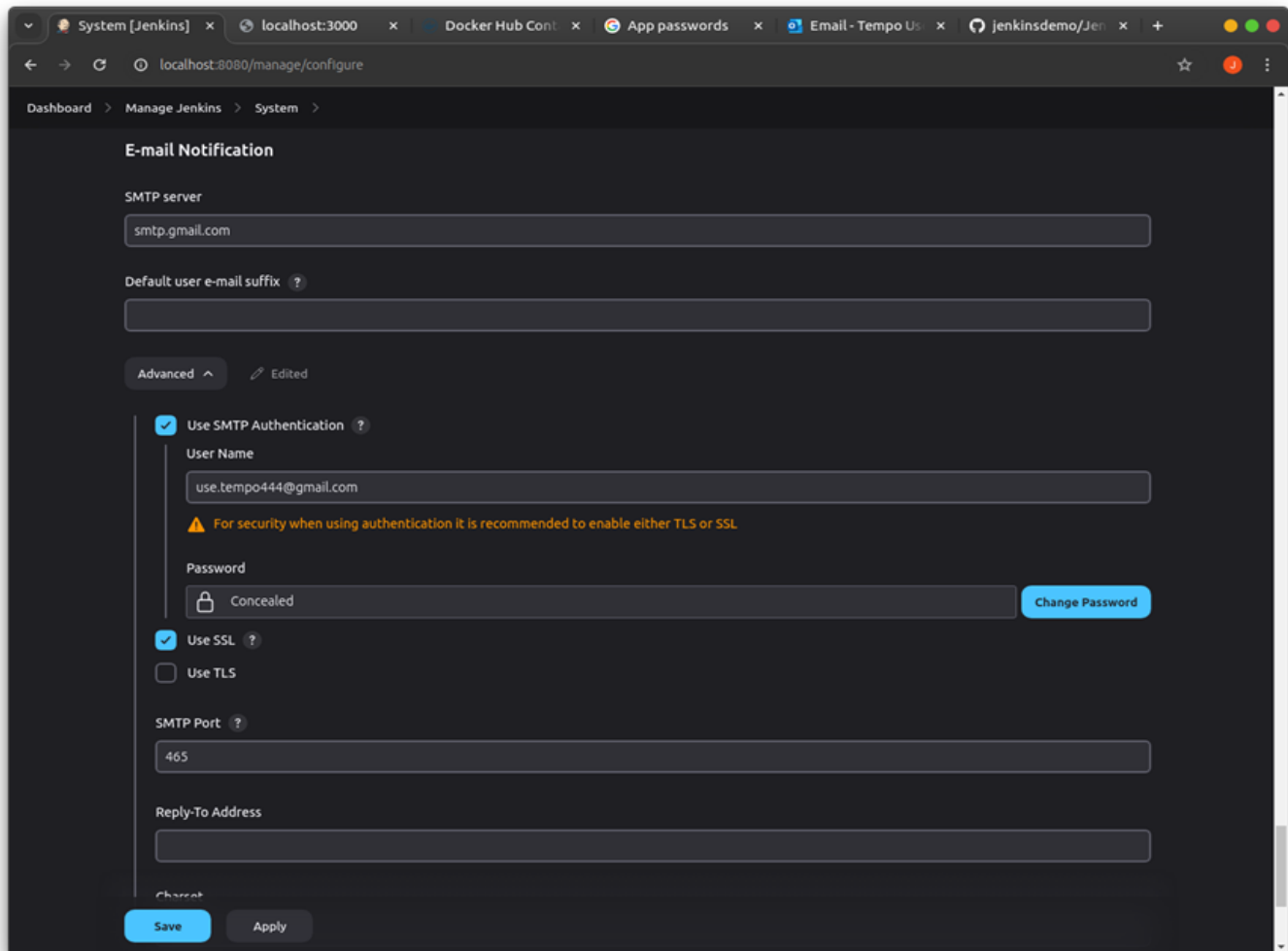
15-Installing Email Plugin



16-Creating App Password



17-Email Notifi Configuration



The screenshot shows the Jenkins 'E-mail Notification' configuration page. The browser tabs include 'System [Jenkins]', 'localhost:3000', 'Docker Hub Cont...', 'App passwords', 'Email - Tempo Us...', and 'jenkinsdemo/Jen...'. The address bar shows 'localhost:8080/manage/configure'. The breadcrumb trail is 'Dashboard > Manage Jenkins > System >'. The page title is 'E-mail Notification'. The 'SMTP server' field contains 'smtp.gmail.com'. The 'Default user e-mail suffix' field is empty. Below these fields is an 'Advanced' section with an 'Edited' icon. In this section, 'Use SMTP Authentication' is checked, and 'Use SSL' is also checked. The 'User Name' field contains 'use.temp444@gmail.com'. A warning message states: 'For security when using authentication it is recommended to enable either TLS or SSL'. The 'Password' field is 'Concealed' with a 'Change Password' button. The 'SMTP Port' field contains '465'. The 'Reply-To Address' field is empty. At the bottom of the configuration section are 'Save' and 'Apply' buttons.

System [Jenkins] x localhost:3000 x Docker Hub Cont... x App passwords x Email - Tempo Us... x jenkinsdemo/Jen... x

localhost:8080/manage/configure

Dashboard > Manage Jenkins > System >

E-mail Notification

SMTP server

smtp.gmail.com

Default user e-mail suffix ?

Advanced ^ Edited

☒ Use SMTP Authentication ?

User Name

use.temp444@gmail.com

⚠ For security when using authentication it is recommended to enable either TLS or SSL

Password

Concealed Change Password

☒ Use SSL ?

☐ Use TLS

SMTP Port ?

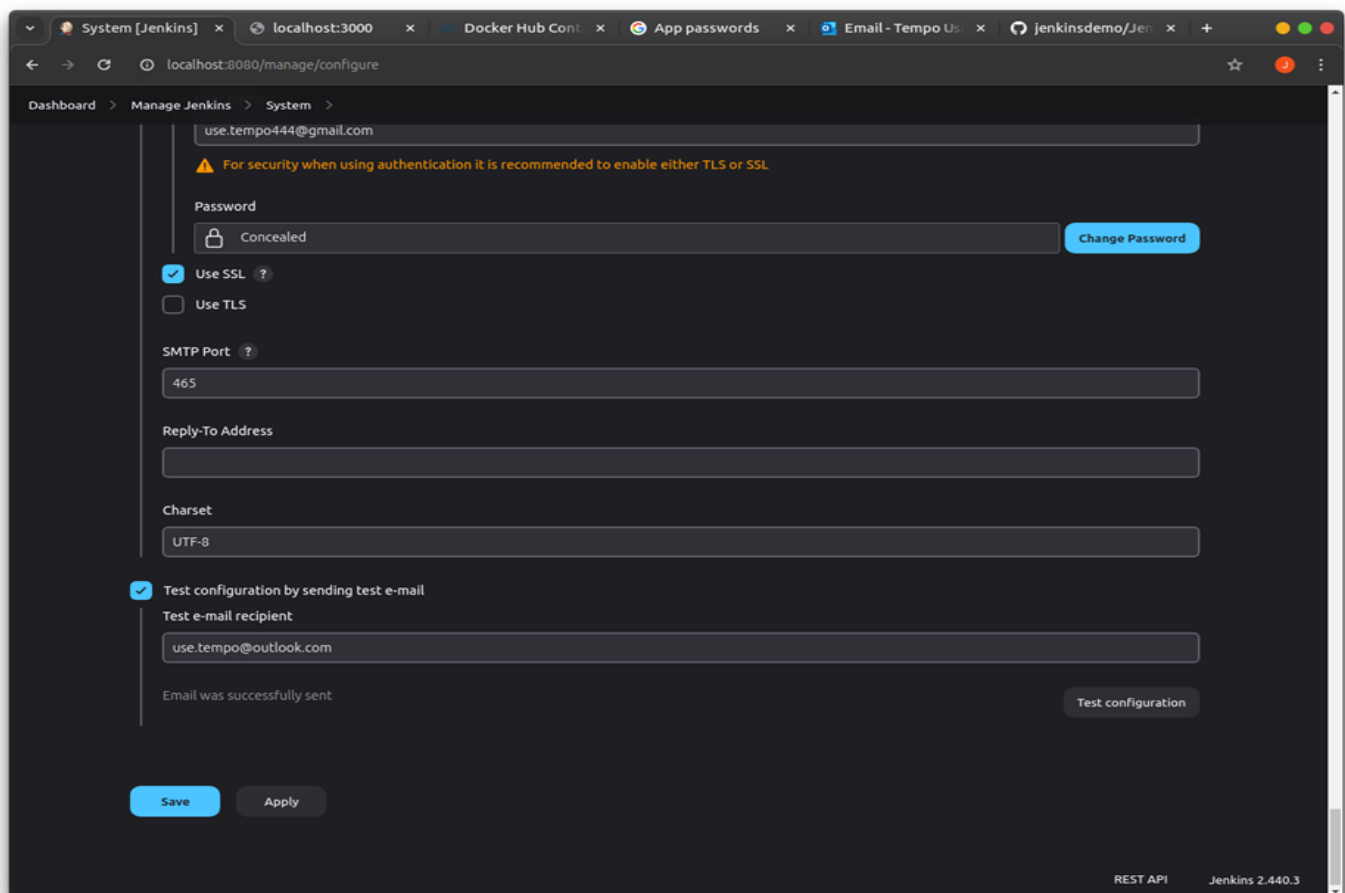
465

Reply-To Address

Charset

Save Apply

18-Test Mail Sent Successfully



This screenshot shows the same Jenkins 'E-mail Notification' configuration page, but with the 'Test configuration' section expanded. The 'Test e-mail recipient' field now contains 'use.temp444@outlook.com'. A message at the bottom of the section states 'Email was successfully sent'. The 'Test configuration' button is visible. The 'Save' and 'Apply' buttons remain at the bottom. The footer of the page shows 'REST API' and 'Jenkins 2.440.3'.

System [Jenkins] x localhost:3000 x Docker Hub Cont... x App passwords x Email - Tempo Us... x jenkinsdemo/Jen... x

localhost:8080/manage/configure

Dashboard > Manage Jenkins > System >

use.temp444@gmail.com

⚠ For security when using authentication it is recommended to enable either TLS or SSL

Password

Concealed Change Password

☒ Use SSL ?

☐ Use TLS

SMTP Port ?

465

Reply-To Address

Charset

UTF-8

☒ Test configuration by sending test e-mail

Test e-mail recipient

use.temp444@outlook.com

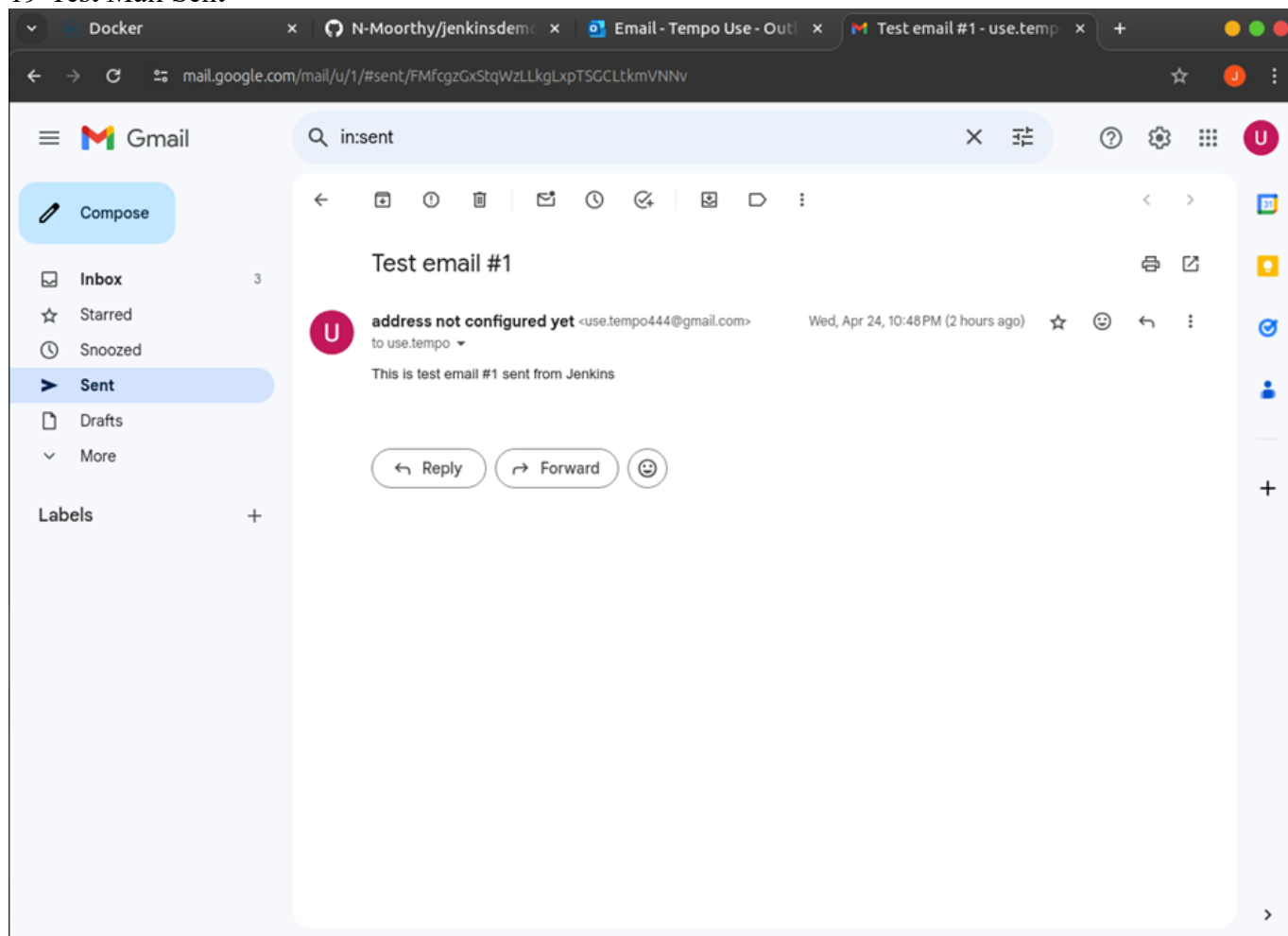
Email was successfully sent

Test configuration

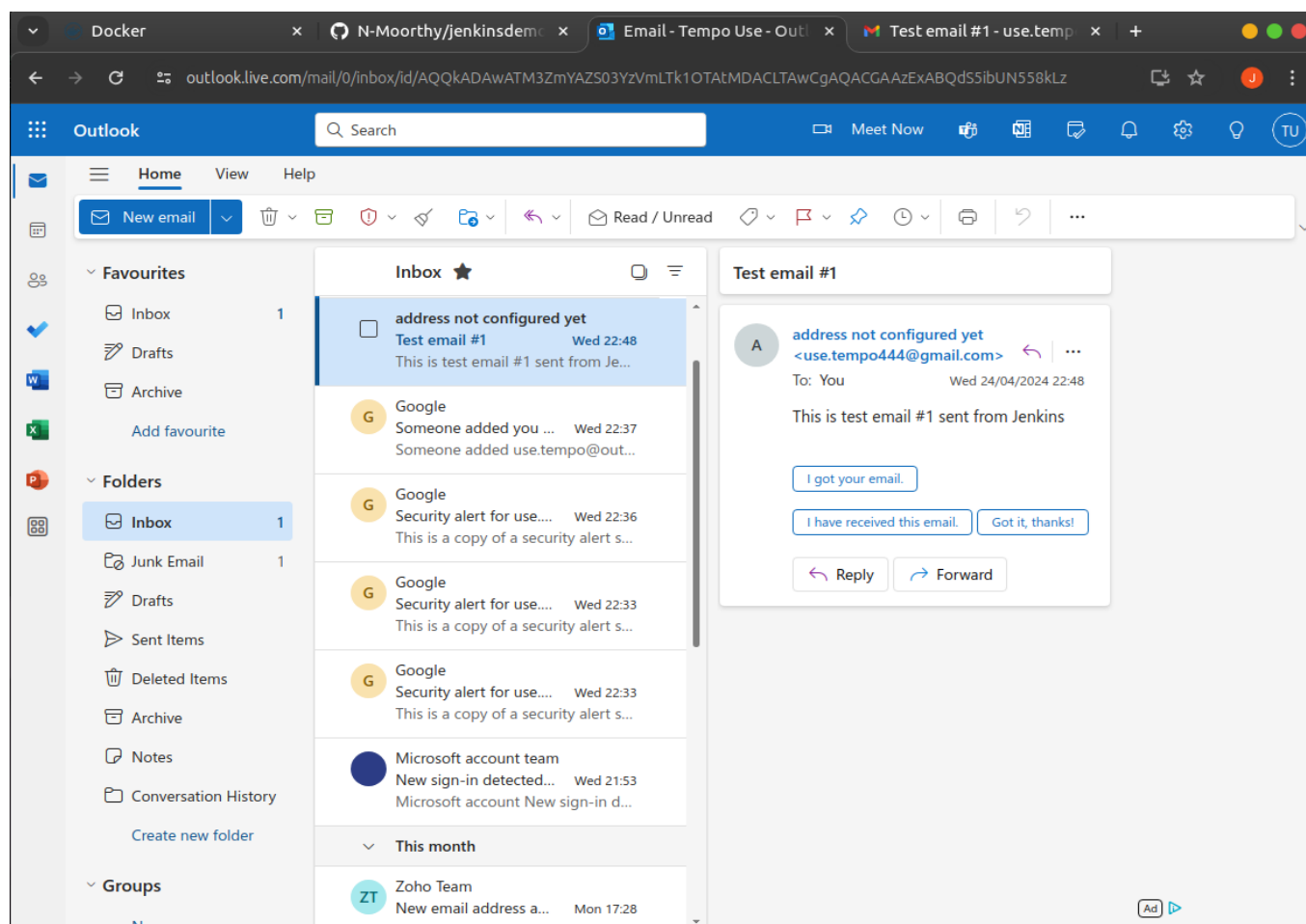
Save Apply

REST API Jenkins 2.440.3

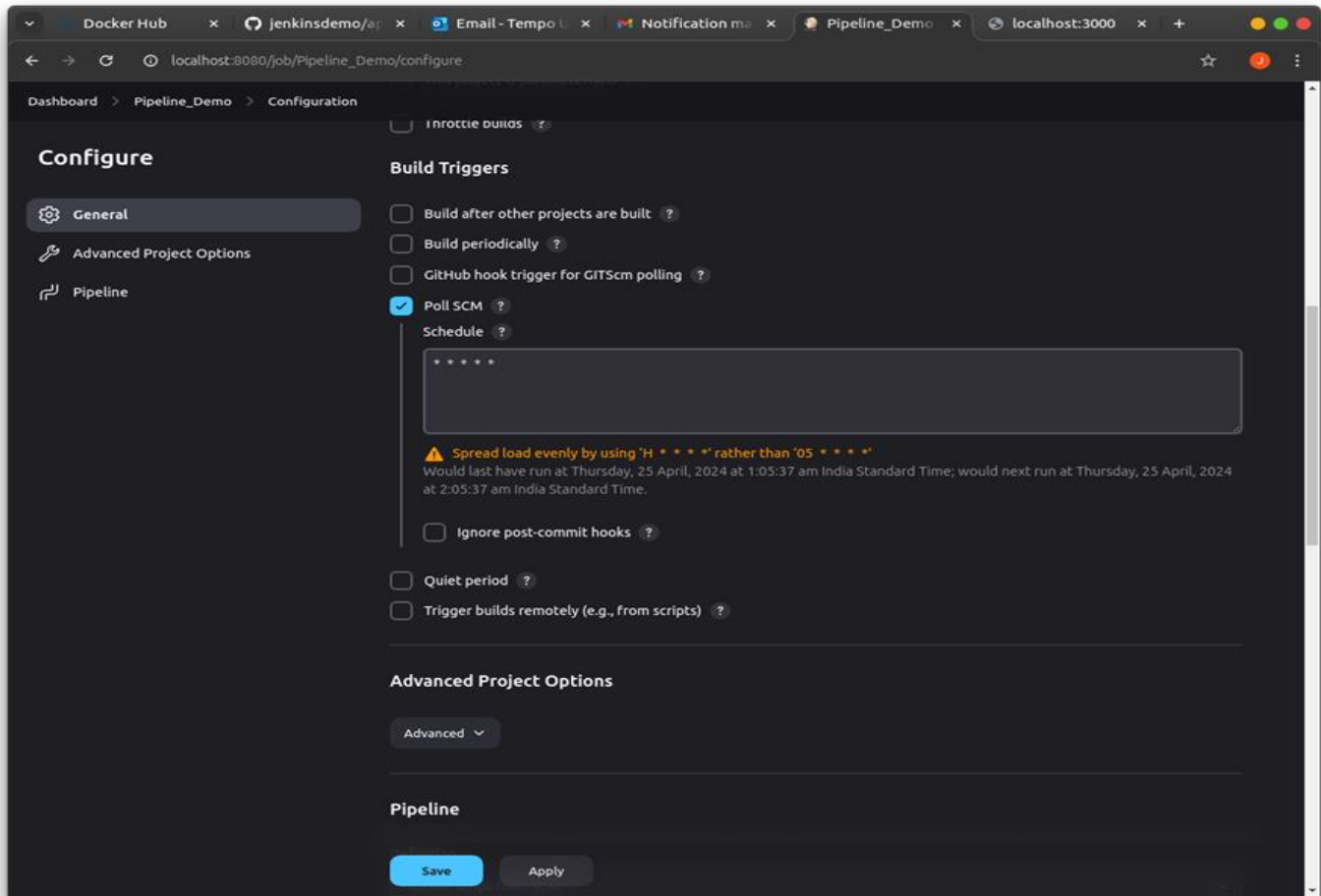
19-Test Mail Sent



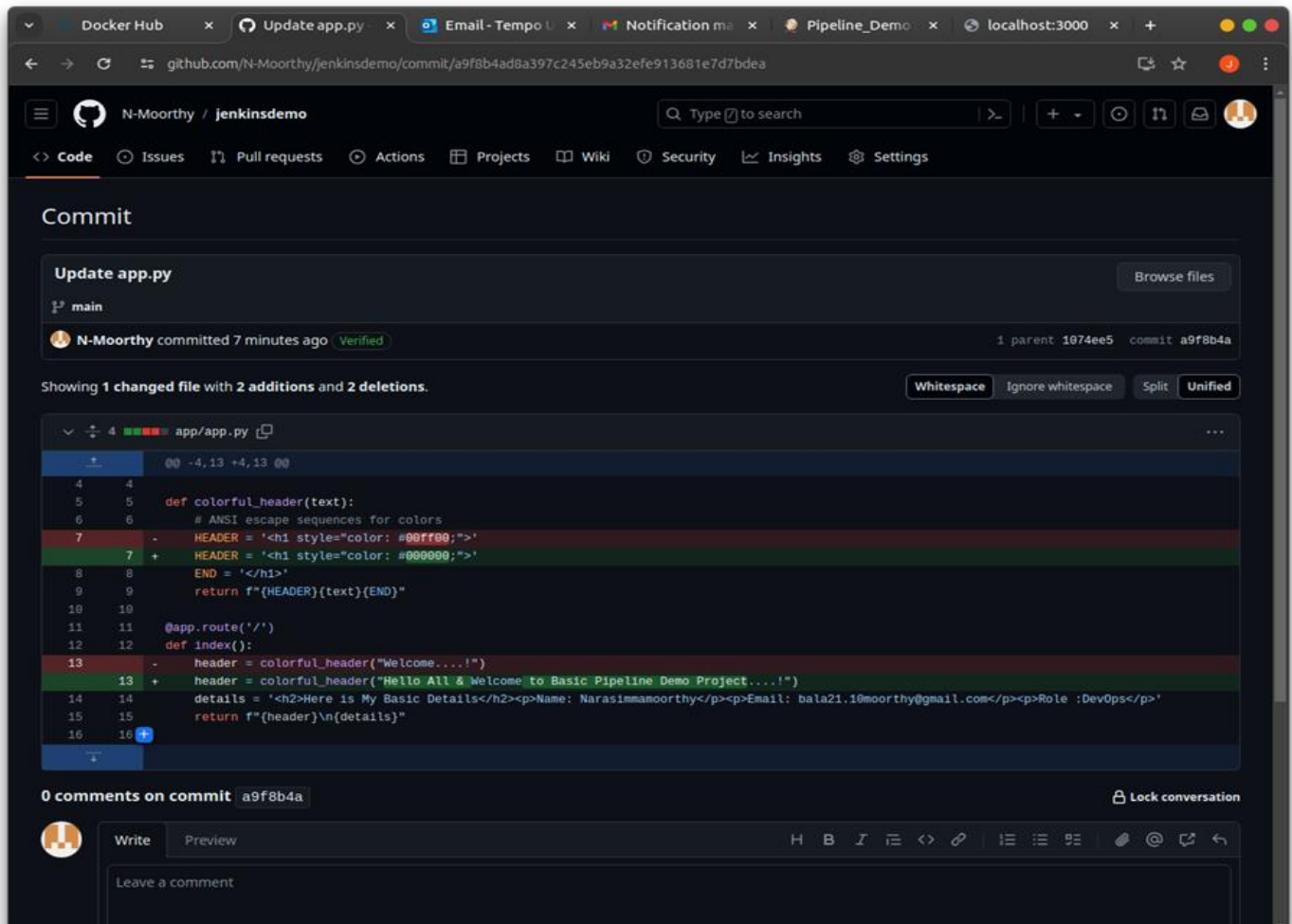
20-Test Mail Received



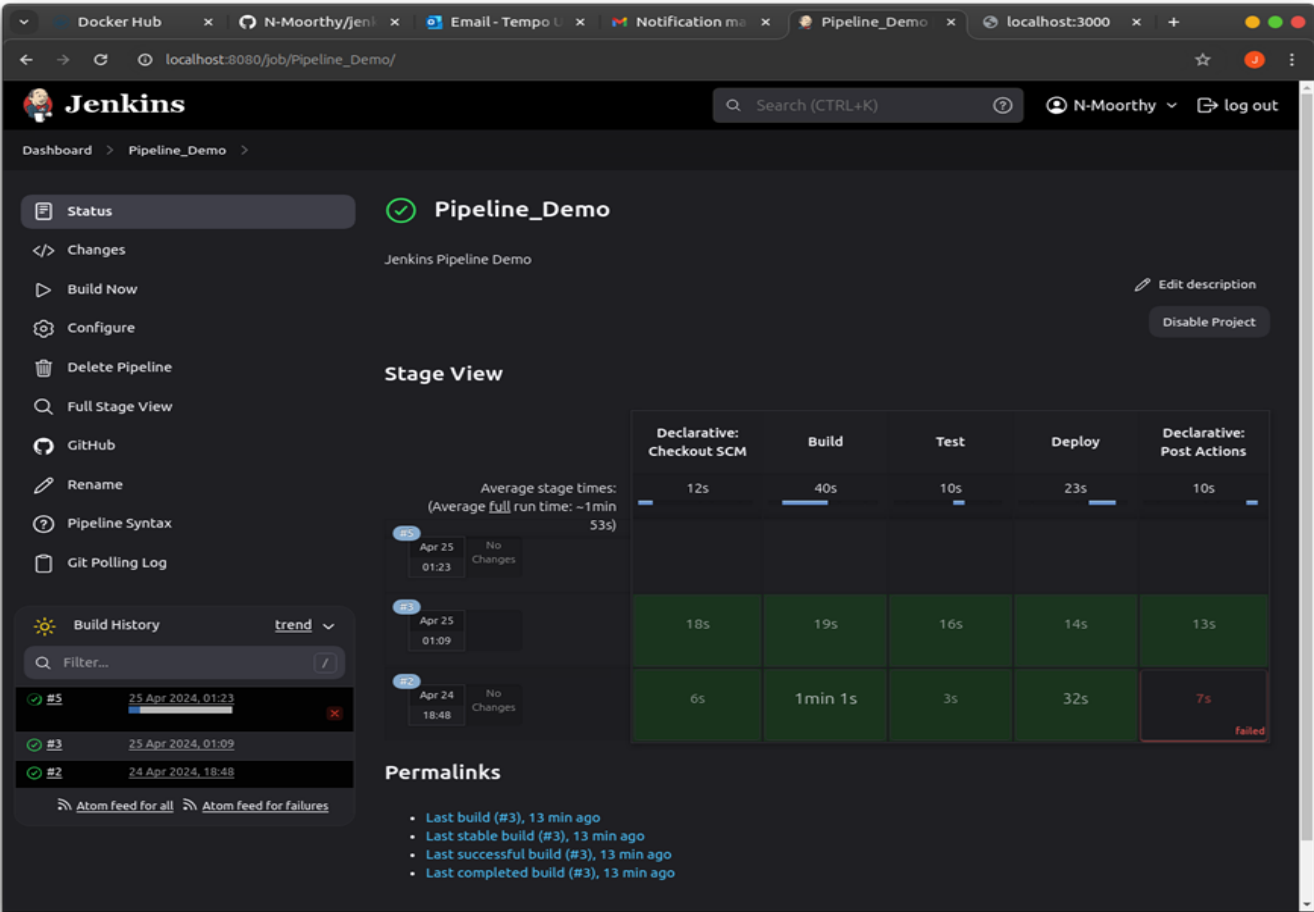
21-Configuring Poll SCM for AutoTrigger Build



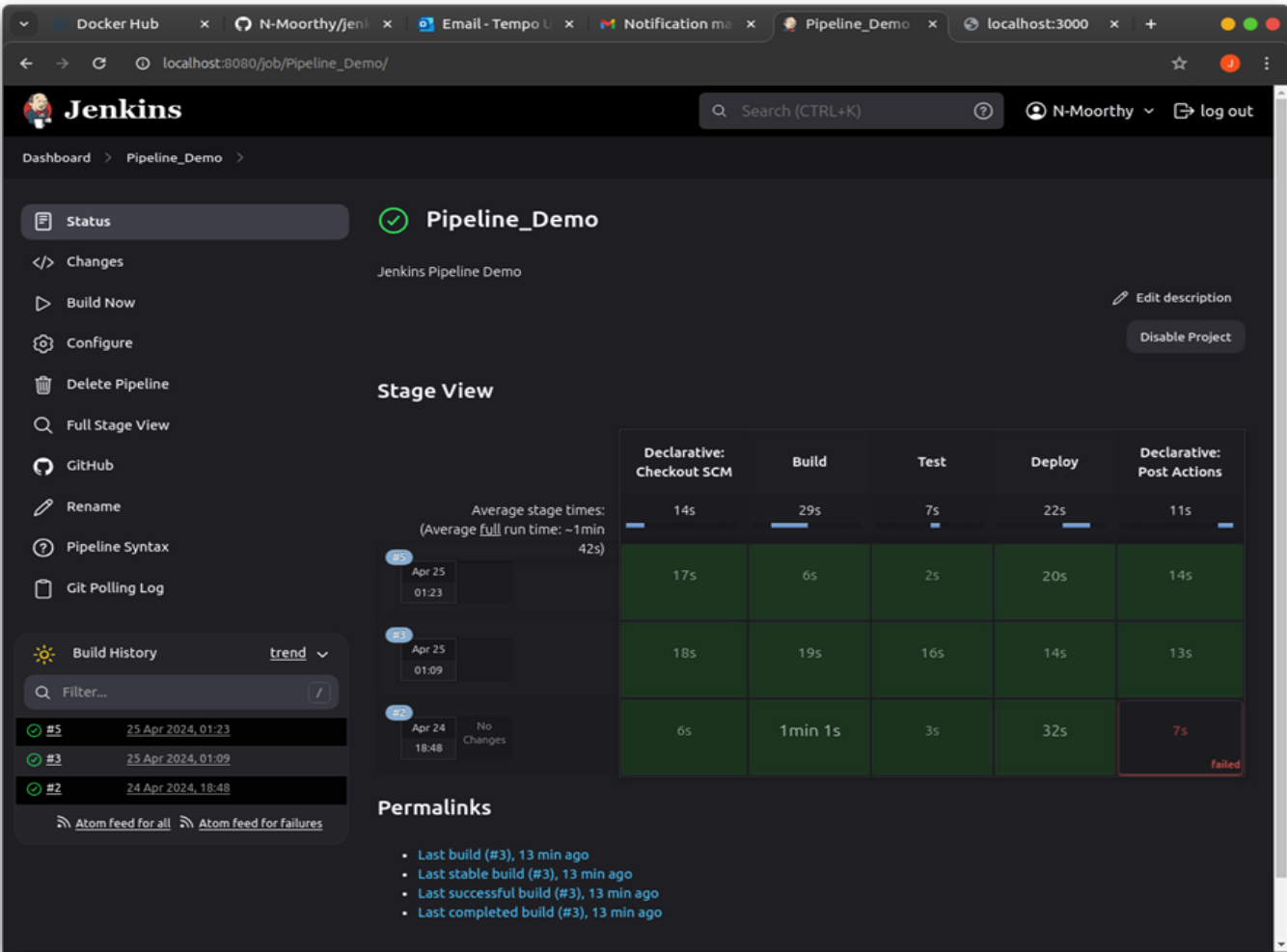
22-Commits made in App



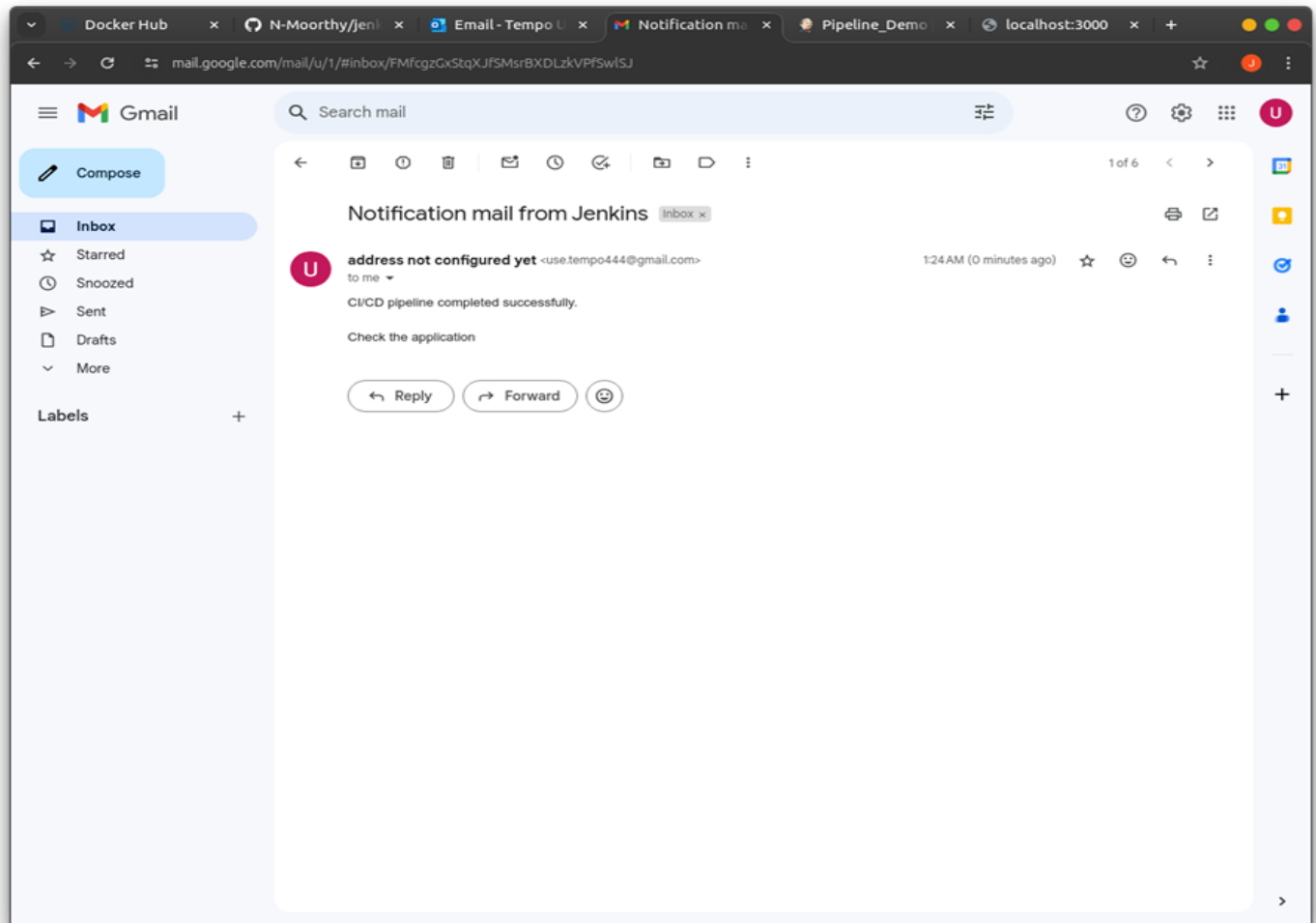
23-Build Auto Triggered for Commit Made



24-Build Done with Post Build Action



25-Email Notification Received for Build Done



26-New Commit made in App

