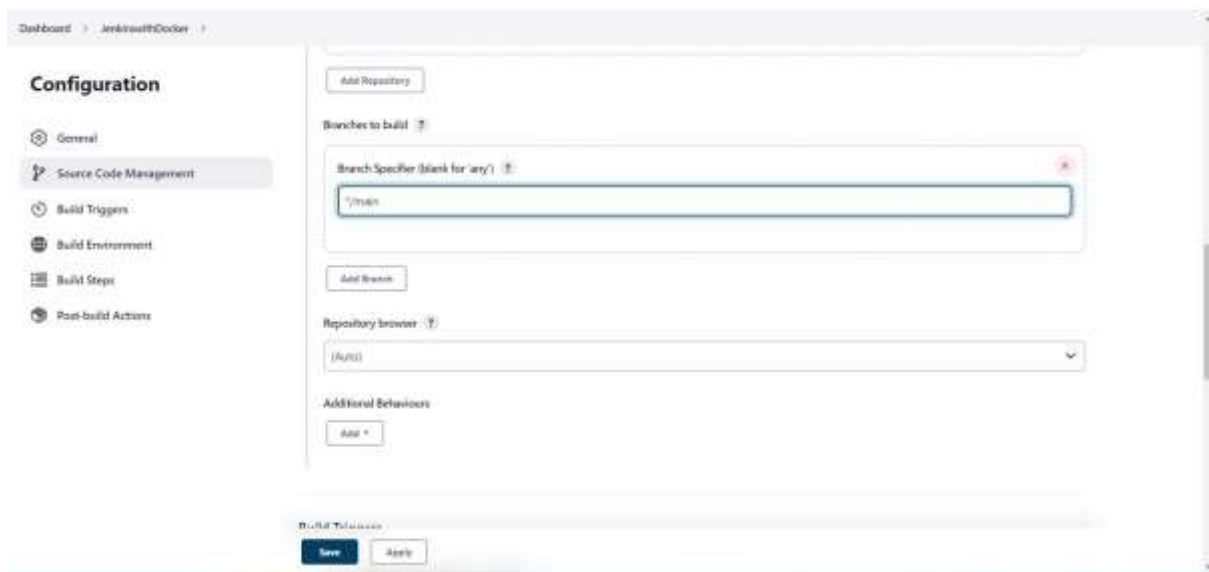


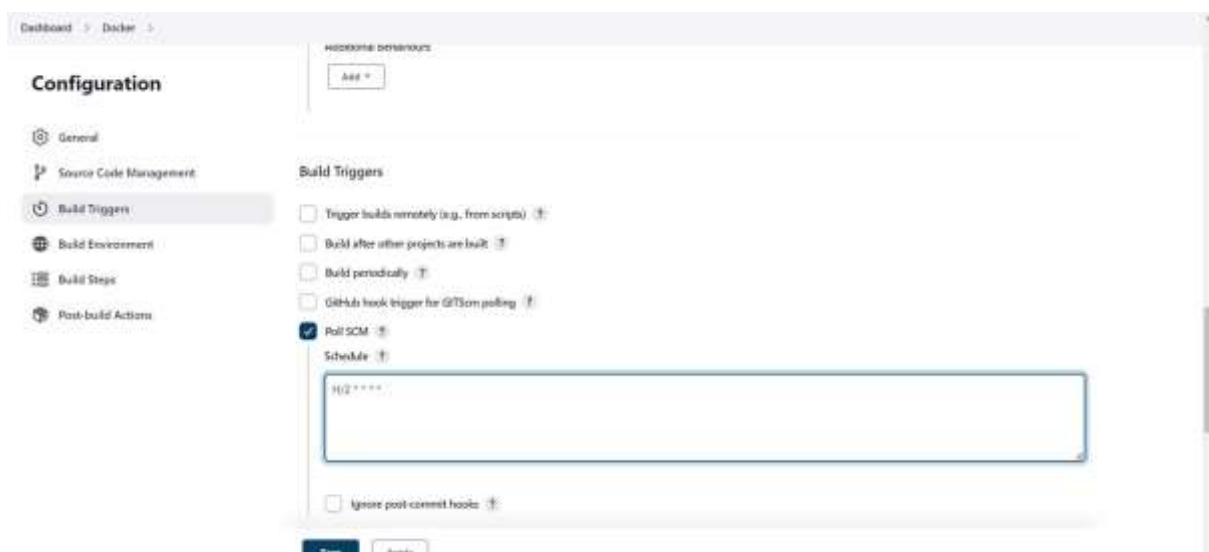
Branch Specifier in GIT



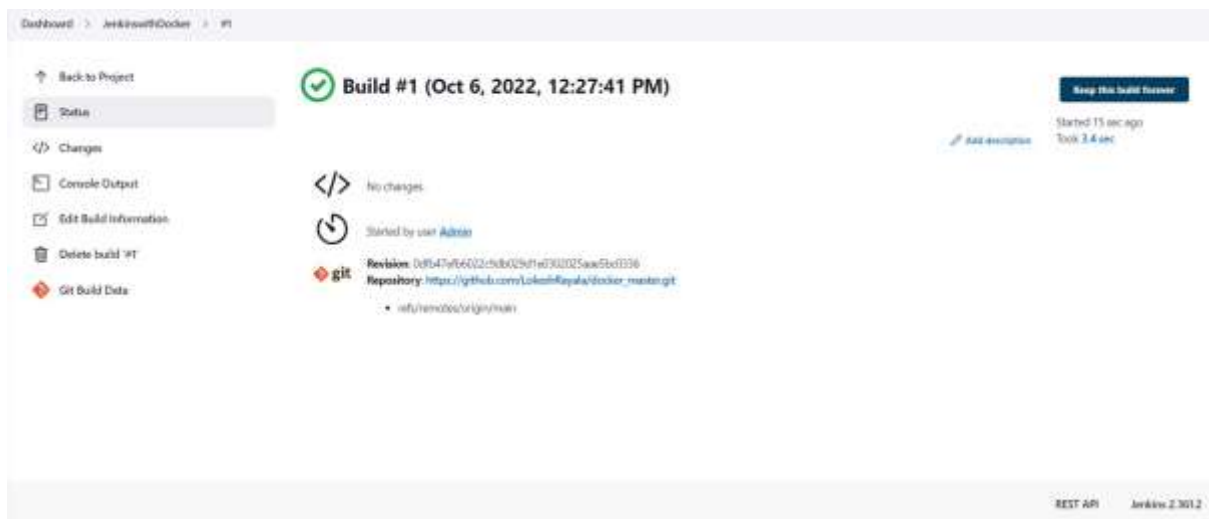
Build History of Jenkins



Build Triggers



Build-1



Dashboard > JenkinswithDocker > #1

Back to Project

Status

Changes

Console Output

Edit Build Information

Delete build #1

Get Build Data

Build #1 (Oct 6, 2022, 12:27:41 PM)

Keep this build forever

Started 15 sec ago
Took 3.4 sec

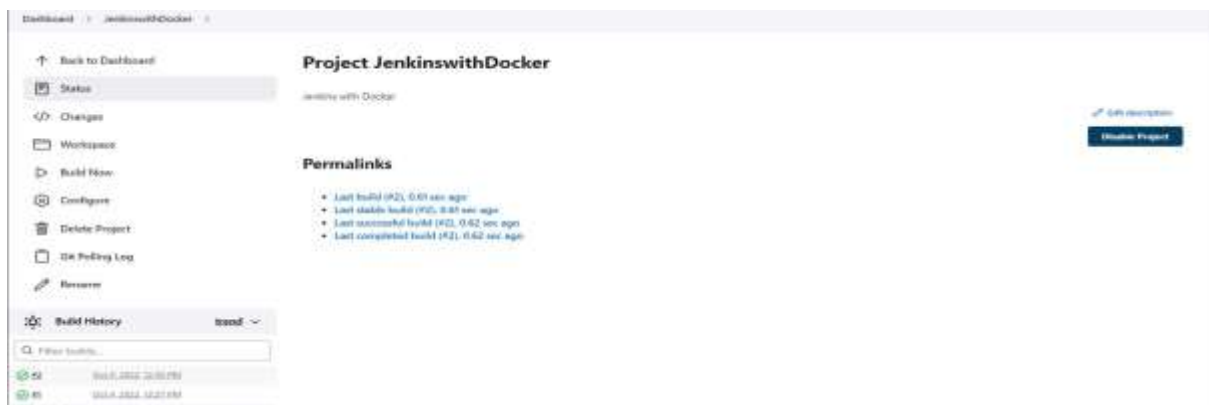
No changes

Started by user Admin

Revision: 0d91479f64023c8b029f1e030025a9e5bd036
Repository: https://github.com/LokeshRayala/docker_master.git
• [info/remotesurl/ign/main](#)

REST API Jenkins 2.361.2

Build-2



Dashboard > JenkinswithDocker >

Back to Dashboard

Status

Changes

Workspace

Build View

Configure

Delete Project

Get Polling Log

Reopen

Build History trend

Filter builds

Project JenkinswithDocker

Jenkins with Docker

Get description

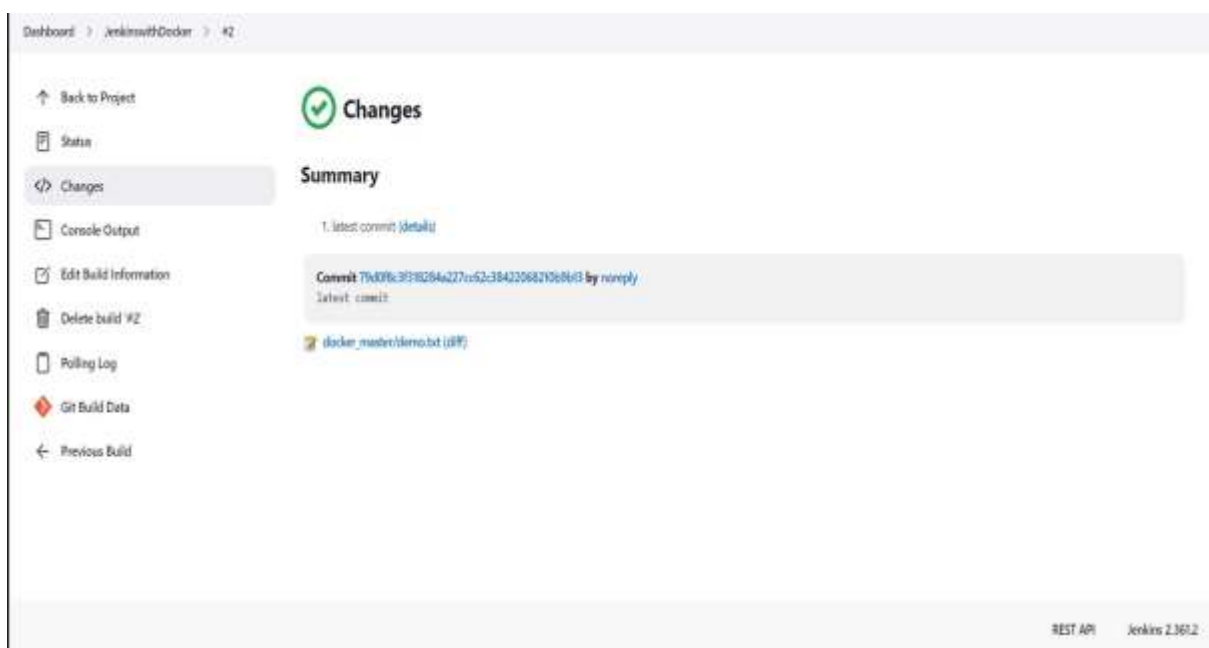
Disable Project

Permalinks

- Last build (#2), 0.61 sec ago
- Last stable build (#2), 0.61 sec ago
- Last successful build (#2), 0.62 sec ago
- Last completed build (#2), 0.62 sec ago

Build	Status	Duration
#1	Success	0:00.000 (20:00:00)
#2	Success	0:00.000 (20:00:00)

Changes Updated In Jenkins



Dashboard > JenkinswithDocker > #2

Back to Project

Status

Changes

Console Output

Edit Build Information

Delete build #2

Rolling Log

Get Build Data

Previous Build

Changes

Summary

1 latest commit (details)

Commit 79d09b3f38264a227cd52c38423068795b643 by nureply
latest commit

docker_master/ignore.txt (diff)

REST API Jenkins 2.361.2

Console Output

The screenshot shows the Jenkins 'Console Output' page for a build. On the left, a sidebar contains navigation links: 'Back to Project', 'Status', 'Changes', 'Console Output' (selected), 'View as plain text', 'Edit Build Information', 'Delete build log', and 'Get Build Data'. The main area displays the build log with a green checkmark icon and the title 'Console Output'. The log text is as follows:

```
Started by user admin
Running as SP5TER
Building in workspace /var/lib/jenkins/workspace/JenkinsWithDocker
The recommended git tool is: NONE
No credentials specified
Cloning the remote git repository
Cloning repository https://github.com:lokanikayala/docker_master.git
> git init /var/lib/jenkins/workspace/JenkinsWithDocker # timeout=10
Fetching upstream changes from https://github.com:lokanikayala/docker_master.git
> git --version # timeout=10
> git --version # "git version 2.34.1"
> git fetch --tags --force --progress -- https://github.com:lokanikayala/docker_master.git --ref=refs/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com:lokanikayala/docker_master.git # timeout=10
> git config --add remote.origin.fetch refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/main^{commit} # timeout=30
Checking out Revision 9d0d4f46dd2c9d6d901a900025ee1cd936 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 9d0d4f46dd2c9d6d901a900025ee1cd936 # timeout=30
Commit message: "Docker Master"
First time build. Skipping changelog.
Finished: SUCCESS
```

Creating a Freestyle Project

The screenshot shows the 'Create new item' dialog in Jenkins. At the top, there is a text input field labeled 'Enter an item name' and a 'Create' button. Below this, a list of project types is shown with icons and descriptions:

- Freestyle project**: This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.
- Maven project**: Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
- Pipeline**: Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Multi-configuration project**: Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds.

A 'OK' button is located at the bottom of the list.

Dashboard

The screenshot shows the Jenkins Dashboard. On the left, a sidebar contains navigation links: '+ New Item', 'People', 'Build History', 'Manage Jenkins', and 'My Views'. The main area features a 'Welcome to Jenkins!' message and a 'Start building your software project' section. The 'Start building your software project' section includes a 'Create a job' button with a right arrow. Below this, the 'Set up a distributed build' section includes a 'Set up an agent' button with a right arrow, a 'Configure a cloud' button with a right arrow, and a 'Learn more about distributed builds' link with a right arrow.

[illegible][illegible]

Dashboard

+ New Item

People

Build History

Manage Jenkins

My Views

Build Queue

No builds in the queue

Build Executor Status

1 idle

2 idle

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job

Set up a distributed build

Set up an agent

Configure a cloud

Learn more about distributed builds

Get started

Jenkins with Docker project

The screenshot shows the Jenkins Dashboard with a table of recent builds. The build 'JenkinswithDocker' is highlighted, showing it was successful, took 1 min 29 sec, and has a last duration of 0.35 sec. Below the table, the 'Build Queue' and 'Build Executor Status' sections are visible, showing no builds in the queue and one executor in use.

#	W	Name	Last Success	Last Failure	Last Duration
1	✓	JenkinswithDocker	1 min 29 sec	N/A	0.35 sec

Latest Commit Details

The screenshot shows the 'Latest Commit Details' page for the 'JenkinswithDocker' project. It displays a table of commits, with the latest commit being 'JenkinswithDocker' (commit 1) with a message 'JenkinswithDocker' and a date of '05/10/2022'.

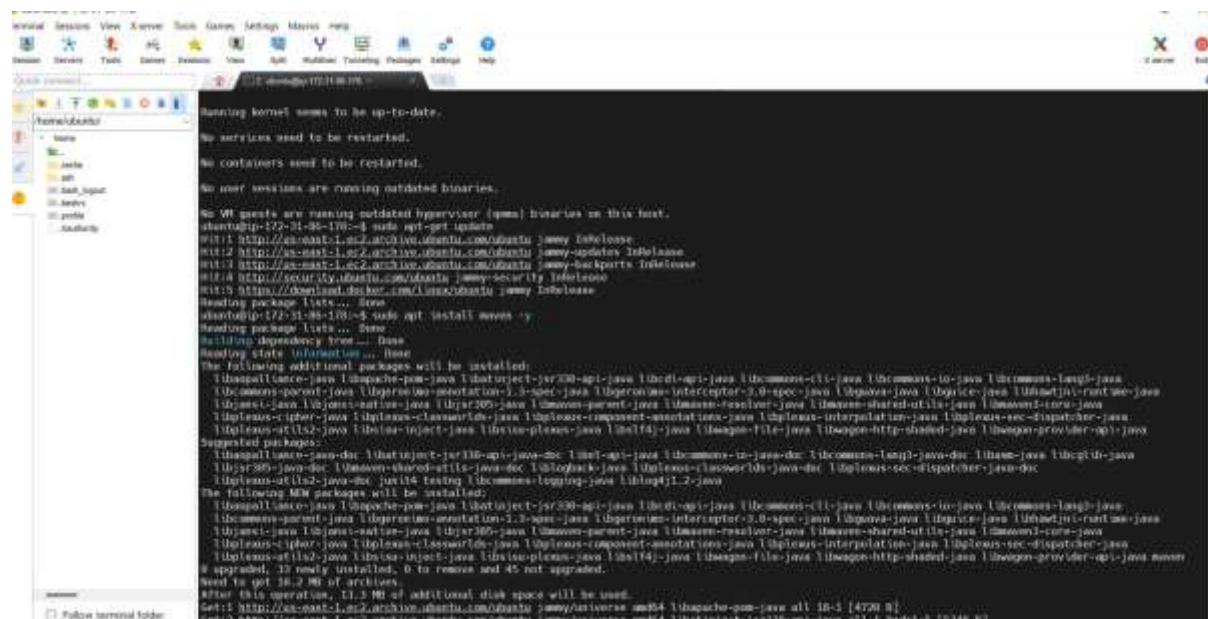
#	Commit	Message	Date
1	JenkinswithDocker	JenkinswithDocker	05/10/2022

Maven Installation

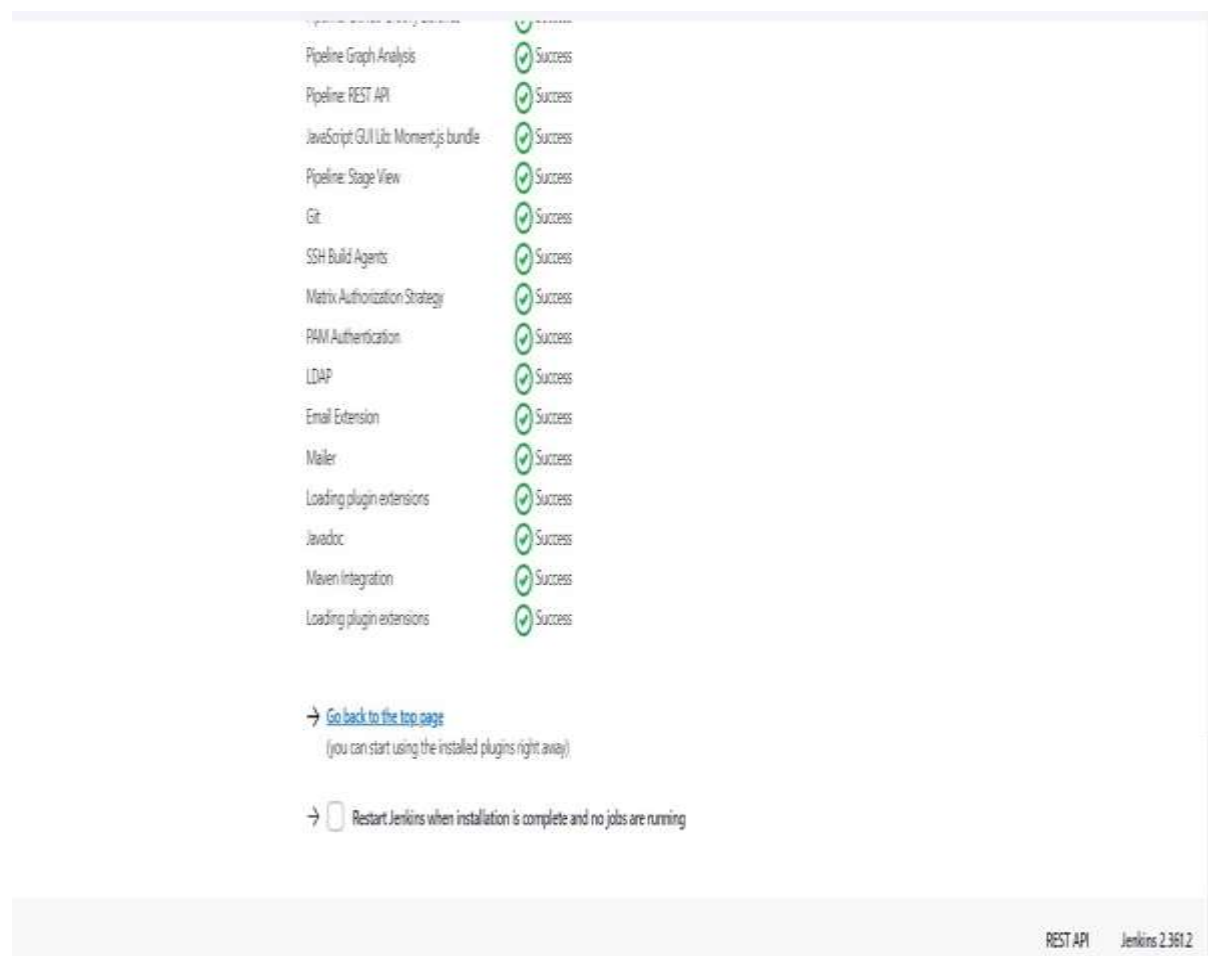
The screenshot shows a terminal window with the command 'mvn -v' being executed. The output shows the Maven version (3.8.6) and the Java version (11.0.15). The terminal also shows the installation of Maven on a Linux system.

```
mvn -v
Apache Maven 3.8.6 (8e597d281041fb38940f732705622e7a9f4e2d05)
Maven home: /usr/share/maven
Java version: 11.0.15, vendor: Oracle Corporation, runtime: /usr/lib/jvm/java-11-openjdk
Default locale: en_US, platform encoding: UTF-8
OS name: 'linux', version: '5.15.0-46-generic', arch: 'amd64', type: 'linux'
```

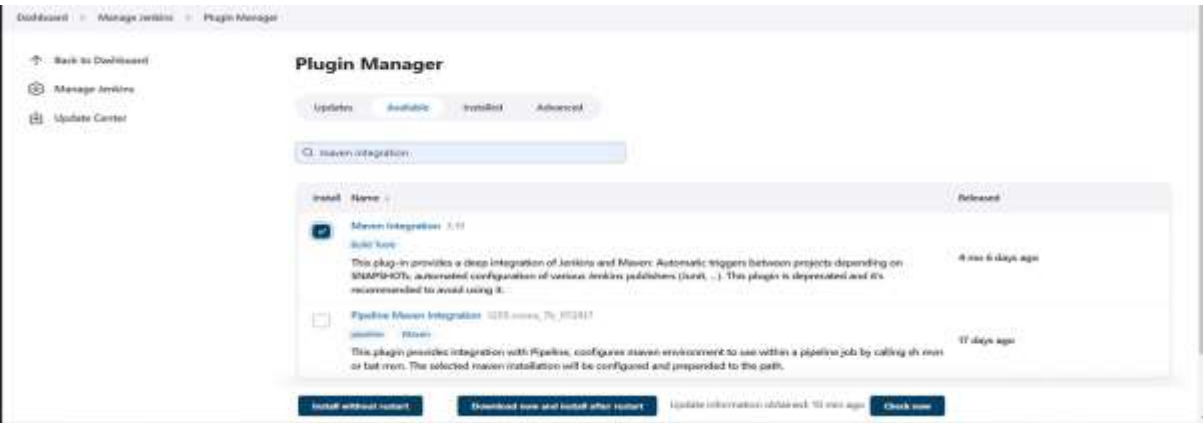
Maven Installation-2



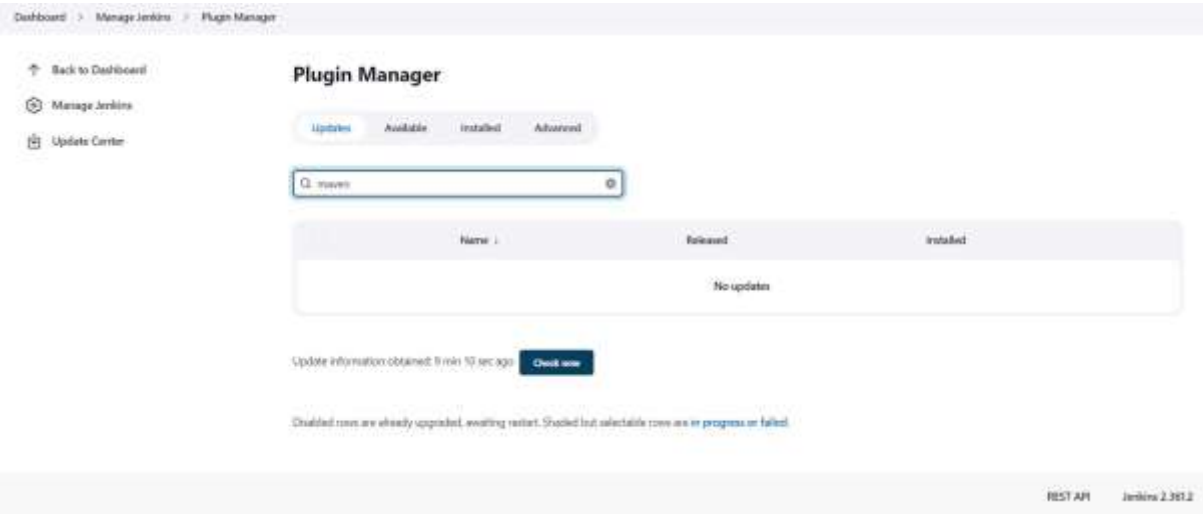
Maven integration installed without restart



Maven Integration plugin installation



Plugin Manager



Project Description

