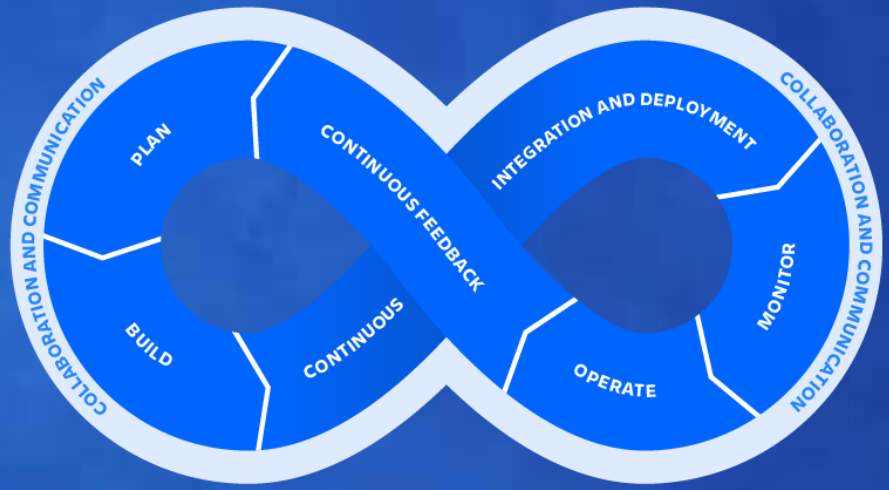


Session 6. Jenkins CI – Introduction and Installation

Ram N Sangwan

Agenda

- Understanding Continuous Integration
- Introduction to Jenkins
- Build Cycle
- Jenkins Architecture
- Obtaining and installing Jenkins
- Installing and configuring Jenkins using WAR and RPM
- Java installation and configuration
- Maven Plugin Installation on Jenkins
- Exploring Jenkins Dashboard

Continuous Integration?

- Continuous Integration (CI) is a development practice that requires developers to integrate code into a shared repository several times a day.
- Each check-in is then verified by an automated build, allowing teams to detect problems early.

What is Jenkins?

- Jenkins is an open source continuous integration tool written in Java.
- The project was forked from Hudson after a dispute with Oracle.
- Jenkins provides continuous integration services for software development.
- It is a server-based system running in a servlet container such as Apache Tomcat.
- Jenkins is an award-winning application that monitors executions of repeated jobs, such as building a software project or jobs run by cron.
- Continuous integration server – **detects** changes in Subversion, **performs** tasks, **repeatedly**. (Build, Test, Deploy, Test, Package)

Jenkins Features

- Easy Installation - Jenkins is a platform-agnostic, ready to run with packages for Windows, Mac OS, and Unix-like operating systems.
- Easy Configuration - Jenkins is easily set up and configured using its web interface, featuring error checks and a built-in help function.
- Available Plugins - There are hundreds of plugins available in the Update Center, integrating with every tool in the CI and CD toolchain.
- Extensible - Jenkins can be extended by means of its plugin architecture, providing nearly endless possibilities for what it can do.
- Easy Distribution - Jenkins can easily distribute work across multiple machines for faster builds, tests, and deployments across multiple platforms.
- Free Open Source

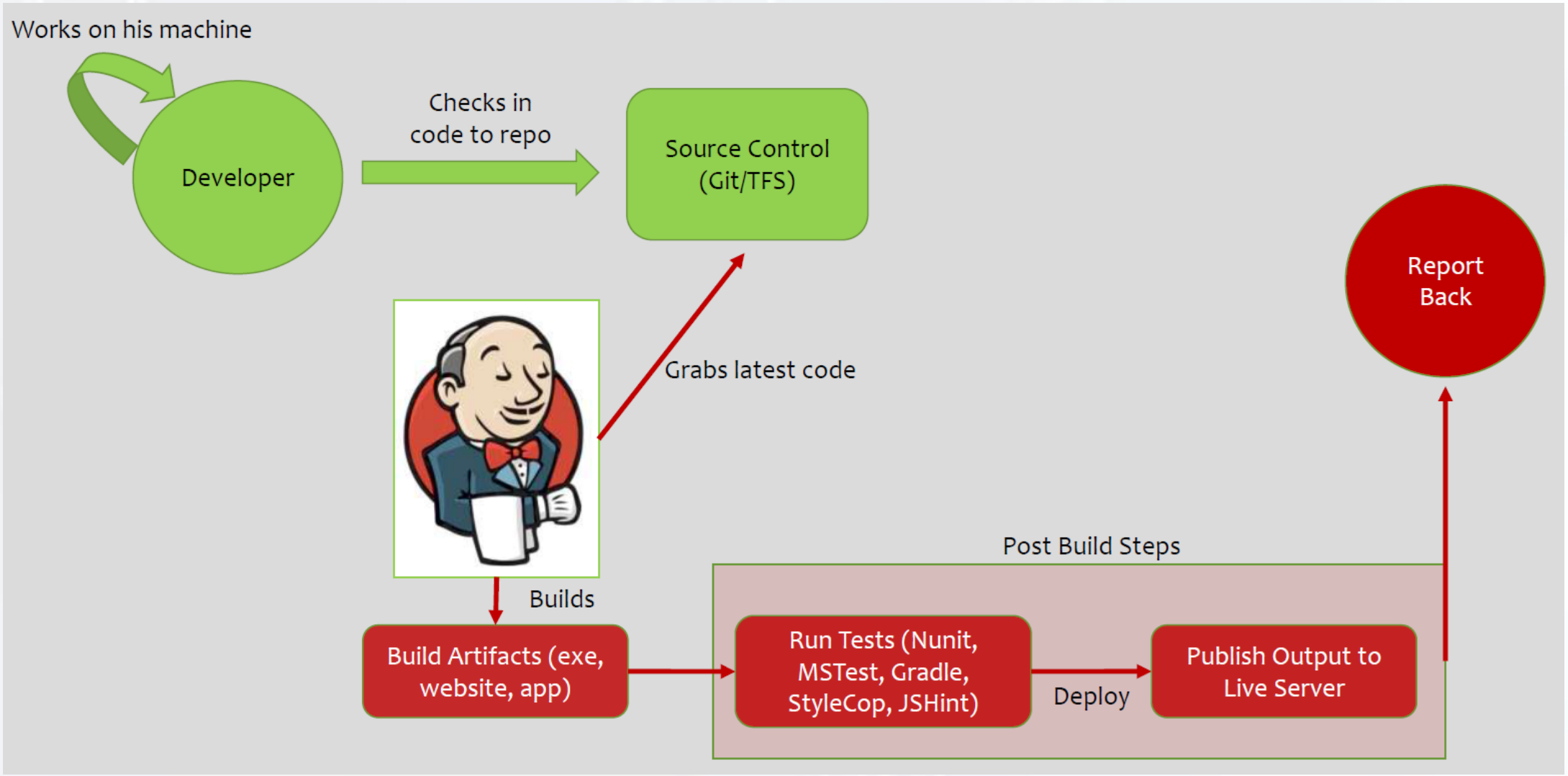
Why do we need it?

- To integrate more frequently, detect errors quicker, improve quality and reduce cost
- Co-ordinate the running of tasks as part of workflows.
- Compile, test and package, deploy, script, verify, build vm's

Terminology

- **Job** - a unit of work for a project
- **View** - user defined collection of jobs or a workflow
- **Master** - the central Jenkins master, does job scheduling
- **Slave** - executes one or more jobs within slots (executors)
- **Workspace** - the working area where a job is carried out
- **Plugin** - user defined collection of jobs or a workflow

Build Cycle


Works on his machine
Developer
Checks in
code to repo
Source Control
(Git/TFS)
Report
Back
Grabs latest code
Post Build Steps
Builds
Build Artifacts (exe,
website, app)
Run Tests (Nunit,
MSTest, Gradle,
StyleCop, JSHint)
Deploy
Publish Output to
Live Server

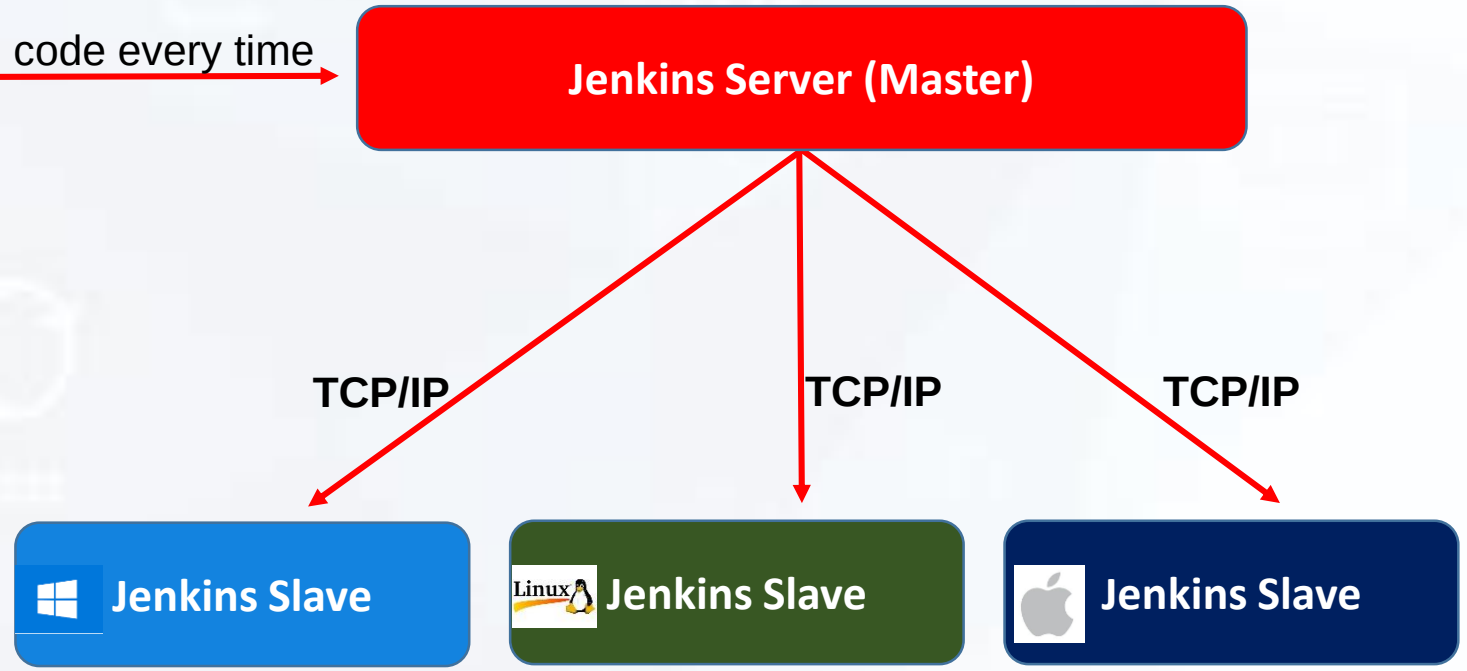
Jenkins Architecture

- Developers commit changes to the source code, found in the repository.
- The Jenkins CI server checks the repository at regular intervals and pulls any newly available code.
- The Build Server builds the code into an executable file. In case the build fails, feedback is sent to the developers.
- Jenkins deploys the build application to the test server. If the test fails, the developers are alerted.
- If the code is error-free, the tested application is deployed on the production server.

Jenkins Architecture

Remote Source Code Repository

Jenkins master pulls the code every time there is a commit.



- Jenkins master distributes its workload to all slaves.
- On request from Jenkins master, the slaves carry out builds and tests and produce test reports.

Installing Java

- Jenkins is a Java application, so the first step is to install Java.
- Run the following command to install the OpenJDK 8 package.

  # yum install -y java-1.8.0-openjdk-devel

  The current version of Jenkins may not support Java 10 (and Java 11).
- If you have multiple versions of Java installed on your machine, make sure Java 8 is the default Java version.

Import GPG Key and Create Repo

- The next step is to enable the Jenkins repository.
- To do that, import the GPG key using the following curl command.
  ***# curl --silent --location http://pkg.jenkins-ci.org/redhat-stable/jenkins.repo | ***
  sudo tee /etc/yum.repos.d/jenkins.repo
- And add the repository to your system with:
  # rpm --import https://pkg.jenkins.io/redhat/jenkins.io.key

  This Command will not give any output.

```
[root@server ~]# curl --silent --location http://pkg.jenkins-ci.org/redhat-stable/jenkins.repo | sudo tee /etc
um.repos.d/jenkins.repo
[jenkins]
name=Jenkins-stable
baseurl=http://pkg.jenkins.io/redhat-stable
gpgcheck=1
[root@server ~]#
```

Install Jenkins

- Once the repository is enabled, install the latest stable version of Jenkins by typing:

  # yum install -y Jenkins

```
================================================================================
 Package                 Arch                Version                  Repository                Size
================================================================================
Installing:
 jenkins                 noarch              2.263.2-1.1              jenkins                   64 M

Transaction Summary
================================================================================
Install  1 Package

Total download size: 64 M
Installed size: 64 M
Downloading packages:
jenkins-2.263.2-1.1.noarch.rpm                                     |  64 MB  00:01:26
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : jenkins-2.263.2-1.1.noarch                                       1/1
  Verifying  : jenkins-2.263.2-1.1.noarch                                       1/1

Installed:
  jenkins.noarch 0:2.263.2-1.1

Complete!
[root@server ~]#
```

Start and Enable Jenkins

- After the installation process is completed, start and enable the Jenkins service to start on system boot

  # systemctl enable --now jenkins

- To check whether it started successfully run:

  # systemctl status jenkins

```
[root@server ~]# systemctl start jenkins && systemctl enable jenkins
jenkins.service is not a native service, redirecting to /sbin/chkconfig.
Executing /sbin/chkconfig jenkins on
[root@server ~]# systemctl status jenkins
● jenkins.service - LSB: Jenkins Automation Server
   Loaded: loaded (/etc/rc.d/init.d/jenkins; bad; vendor preset: disabled)
   Active: active (running) since Sun 2021-01-17 05:15:47 EST; 10s ago
     Docs: man:systemd-sysv-generator(8)
   CGroup: /system.slice/jenkins.service
           └─11856 /etc/alternatives/java -Dcom.sun.akuma.Daemon=daemonized -Djava.awt.headless=true -DJENKIN...

Jan 17 05:15:45 server.example.com systemd[1]: Starting LSB: Jenkins Automation Server...
Jan 17 05:15:45 server.example.com runuser[11842]: pam_unix(runuser:session): session opened for user jen...d=0)
Jan 17 05:15:47 server.example.com jenkins[11837]: Starting Jenkins [  OK  ]
Jan 17 05:15:47 server.example.com systemd[1]: Started LSB: Jenkins Automation Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@server ~]#
```

Open Firewall Ports

- If you are installing Jenkins on a remote CentOS server that is protected by a firewall you need to port 8080. Use the following commands to open the necessary port:

  # firewall-cmd --permanent --zone=public --add-port=8080/tcp

  # firewall-cmd –reload

Installing with WAR File

Jenkins is running from the previous steps. Stop Jenkins if running and disable it. Alternatively you can remove Jenkins with ***yum erase Jenkins***.

- Download latest Jenkins WAR file from here.

```
[root@server ~]# wget http://mirrors.jenkins.io/war-stable/latest/jenkins.war
--2021-01-17 05:19:59--  http://mirrors.jenkins.io/war-stable/latest/jenkins.war
Resolving mirrors.jenkins.io (mirrors.jenkins.io)... 52.202.51.185
Connecting to mirrors.jenkins.io (mirrors.jenkins.io)|52.202.51.185|:80... connected.
HTTP request sent, awaiting response... 302 Found
Location: http://ftp-chi.osuosl.org/pub/jenkins/war-stable/2.263.2/jenkins.war [following]
--2021-01-17 05:20:00--  http://ftp-chi.osuosl.org/pub/jenkins/war-stable/2.263.2/jenkins.war
Resolving ftp-chi.osuosl.org (ftp-chi.osuosl.org)... 64.50.236.52, 2600:3402:200:227::2
Connecting to ftp-chi.osuosl.org (ftp-chi.osuosl.org)|64.50.236.52|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 67285886 (64M) [application/x-java-archive]
Saving to: ‘jenkins.war’

12% [=======>                                                    ] 8,329,897    233KB/s
```

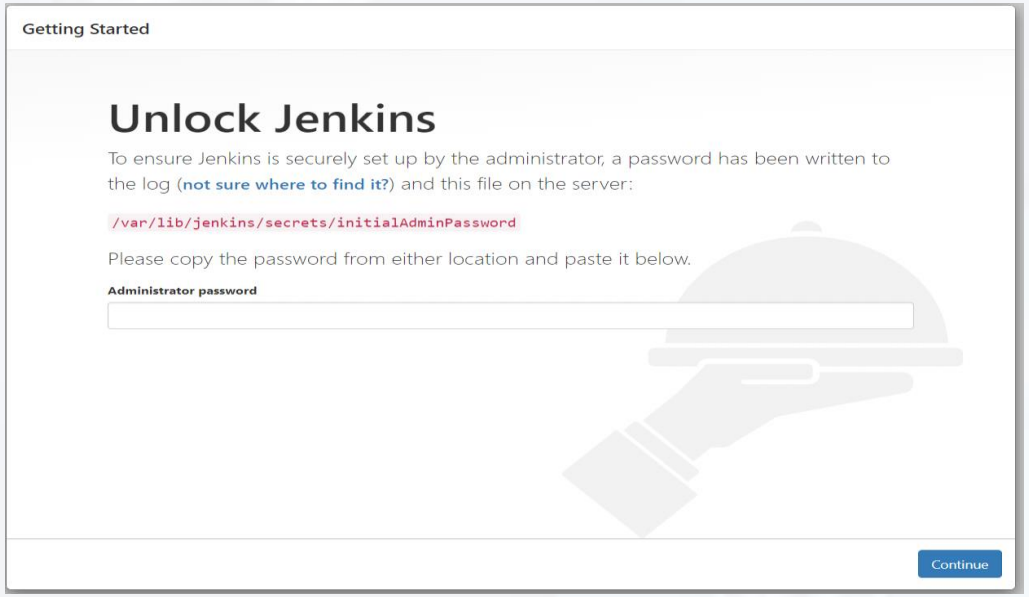
Run the war file to install Jenkins

- Run the command **java -jar jenkins.war**
  # java -jar jenkins.war
- Take note of the Password displayed during installation.
- We will use this password for the initial login later.

```
Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:

fd65449d86d74d29ac0547c5574036a6

This may also be found at: /root/.jenkins/secrets/initialAdminPassword

*************************************************************
*************************************************************
*************************************************************
```

Setup Jenkins

- To set up your new Jenkins installation, open your browser and type your domain or IP address followed by port 8080:
  http://10.10.0.100:8080
- A screen similar to the following will appear, prompting you to enter the Administrator password that is created during the installation:



Take note of the Password

- If you have installed Jenkins using ***yum***, use the following command to print the password on your terminal:

  # cat /var/lib/jenkins/secrets/initialAdminPassword

  You should see a 32-character long alphanumeric password as shown below:

  2115173b548f4e99a203ee99a8732a32

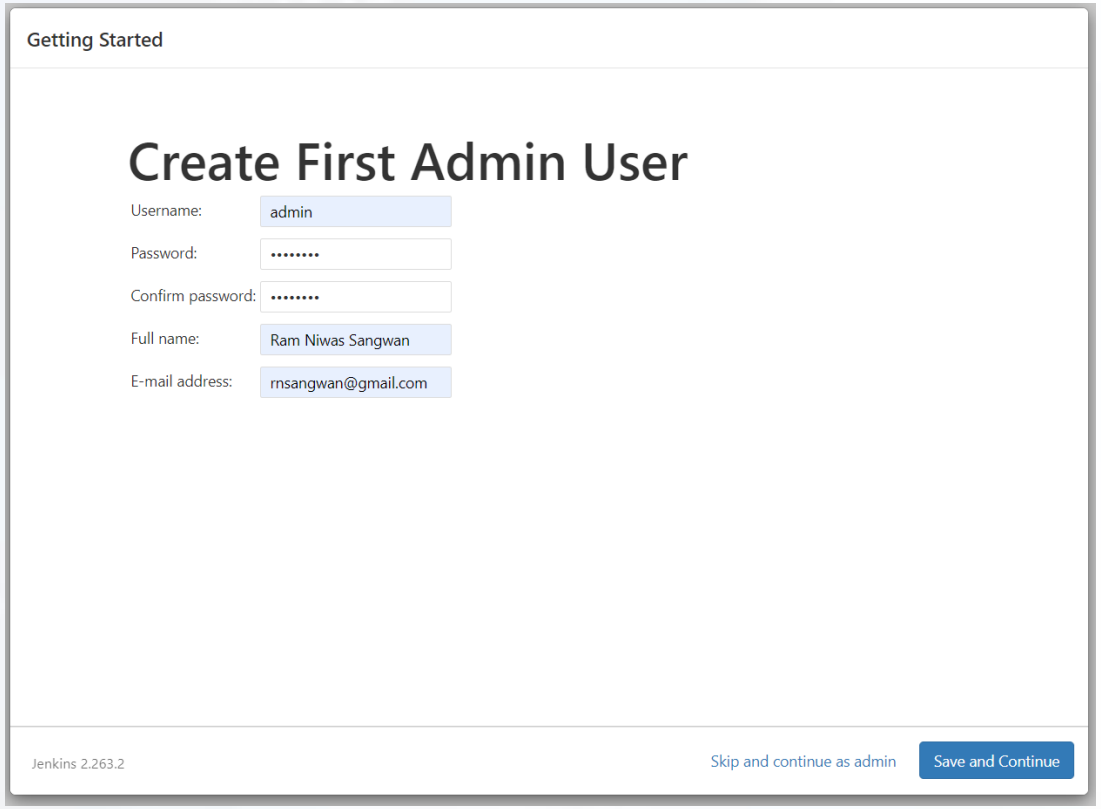
- Copy the password from your terminal, paste it into the Administrator password field and click ***Continue***.

Customize Jenkins

On the next screen, you will be asked whether you want to install the suggested plugins or to select specific plugins.
Click on the Install suggested plugins box, and the installation process will start immediately

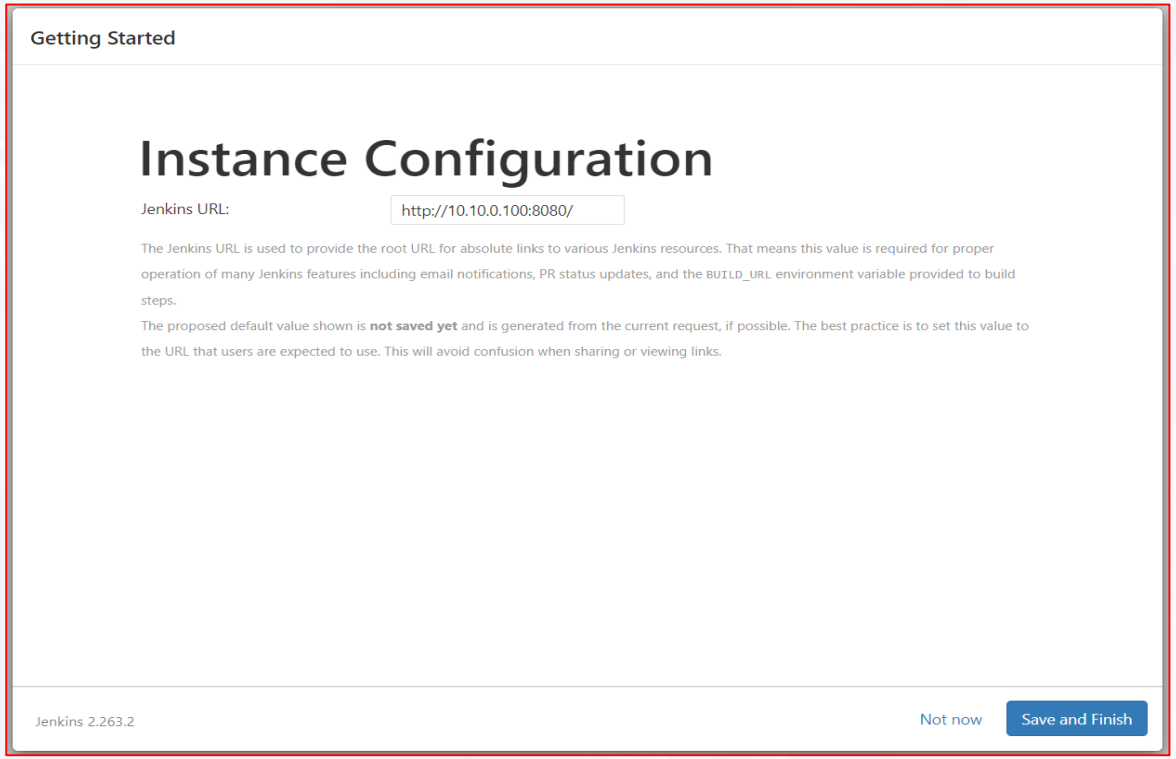
Create First Admin User

- Once the installation is complete, you will be prompted to set up the first administrative user.
- Fill out all required information and click Save and Continue



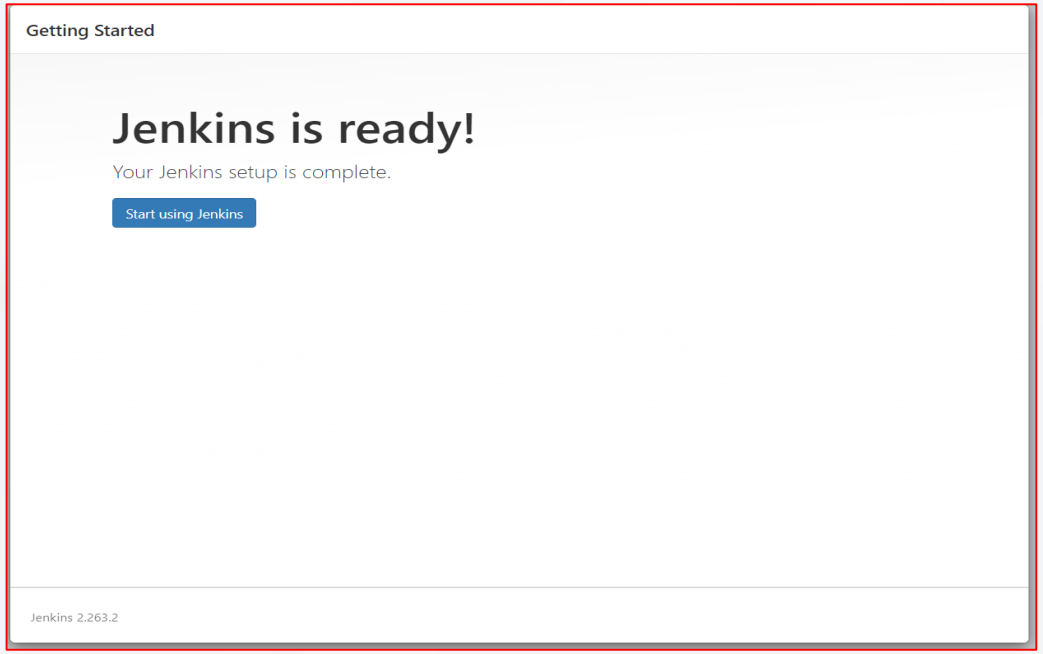
Instance Configuration

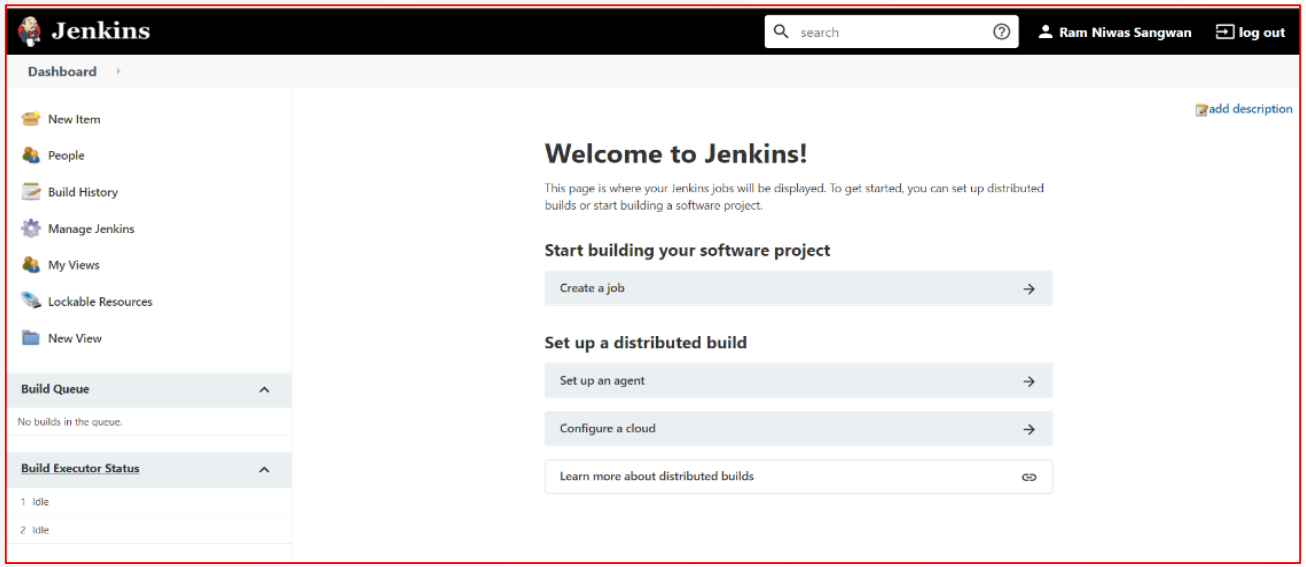
- On the next page, you will be asked to set the URL for the Jenkins instance.
- The URL field will be populated with an automatically generated URL.
- To complete the setup confirm the URL by clicking on the ***Save and Finish*** button



Jenkins is Ready

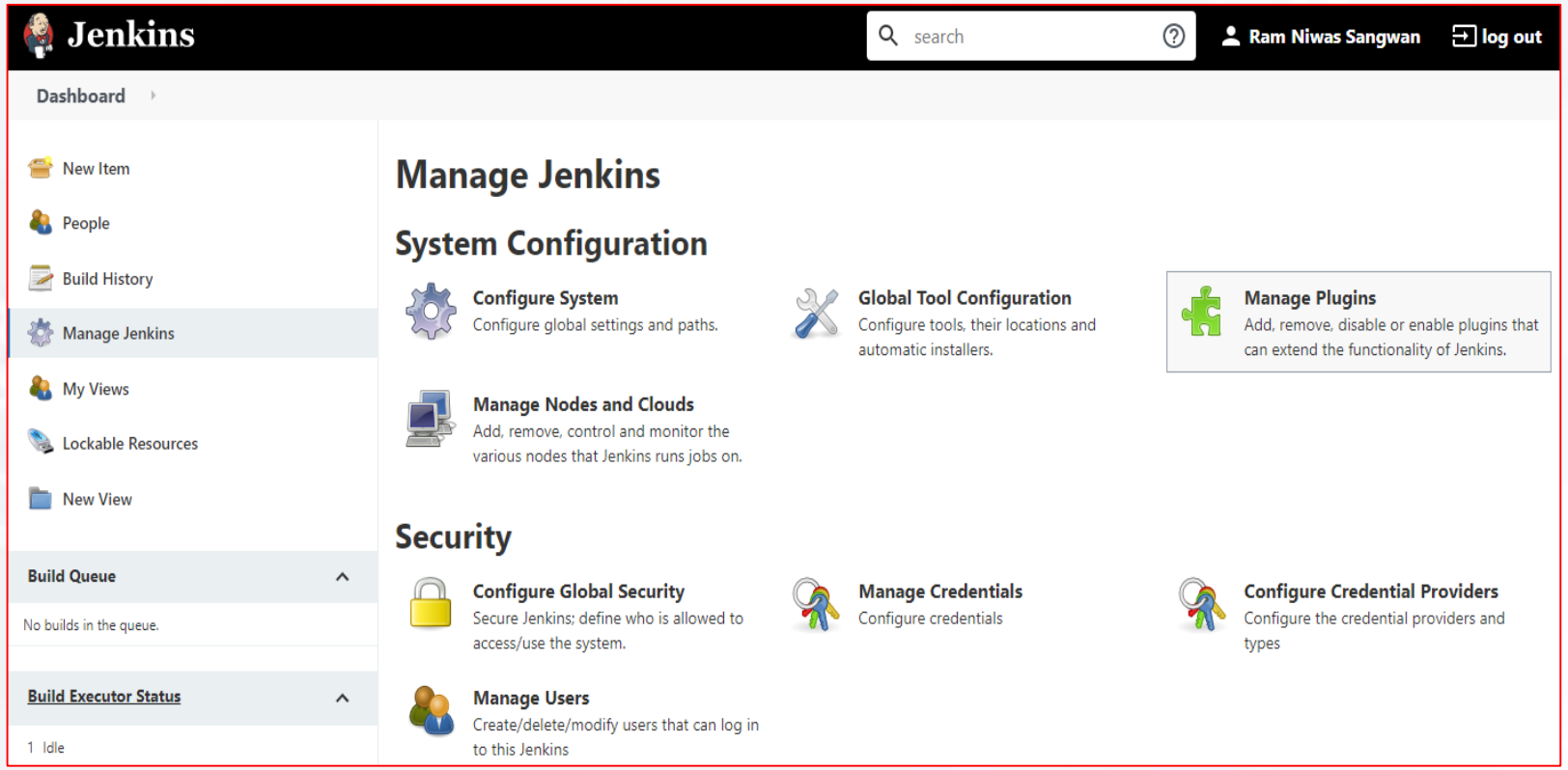
- Finally, click on the Start using Jenkins button and you will be redirected to the Jenkins dashboard logged in as the admin user you have created in one of the previous steps.





Setting up Jenkins and Maven

- In the Jenkins dashboard (Home screen), click **Manage Jenkins** from the left-hand side menu.

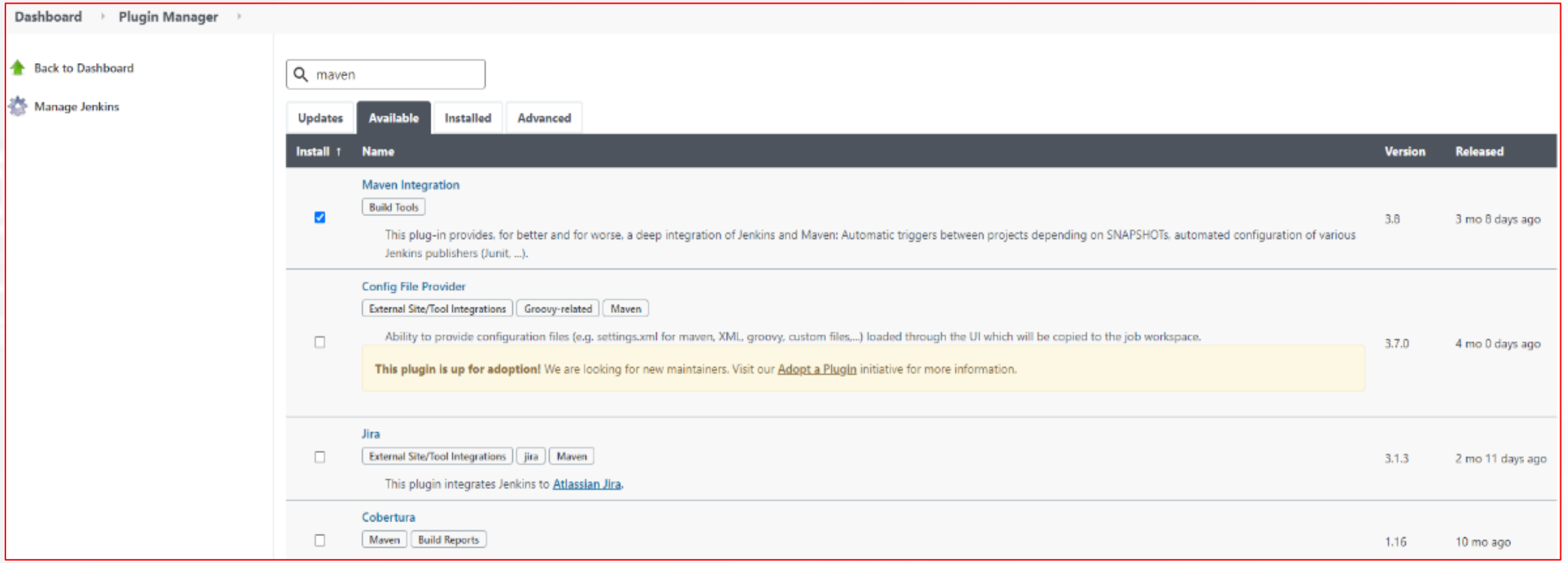


Maven for Jenkins

- Visit http://10.10.0.100:8080/pluginManager/available directly.

OR

- In the Configure system screen, click on **Manage Plugins.**

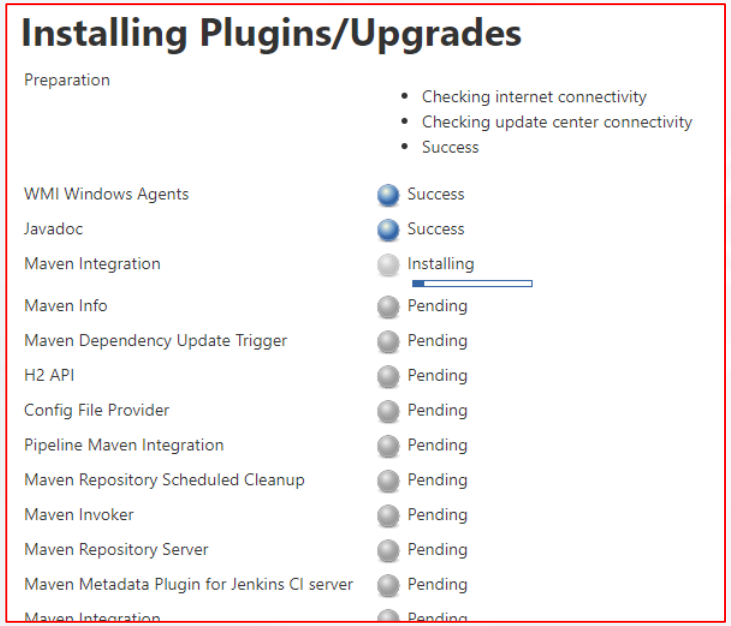


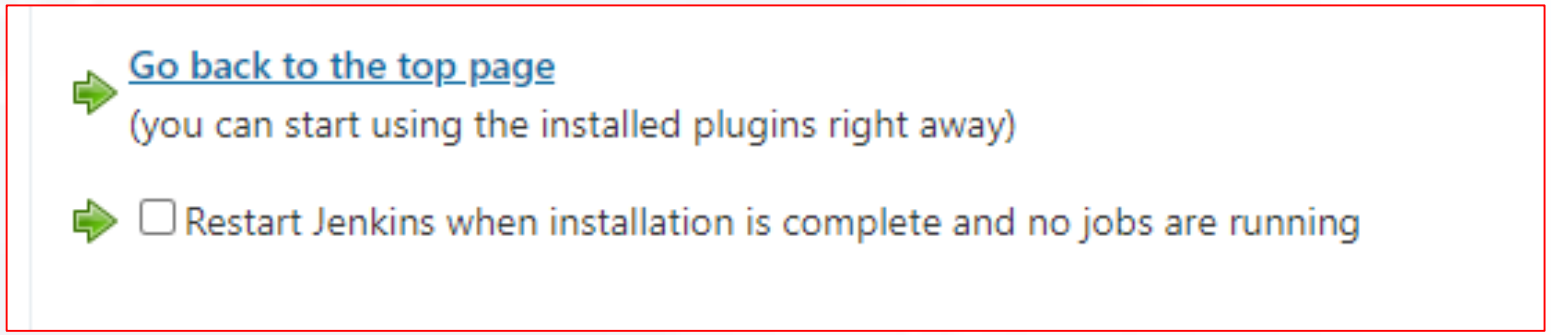
Search Maven for Jenkins

- In the search box type **Maven**.
- You will see all plugins for Maven.
- Select the checkbox in front of the plugins and Click on the button **Install without restart** given at the bottom.

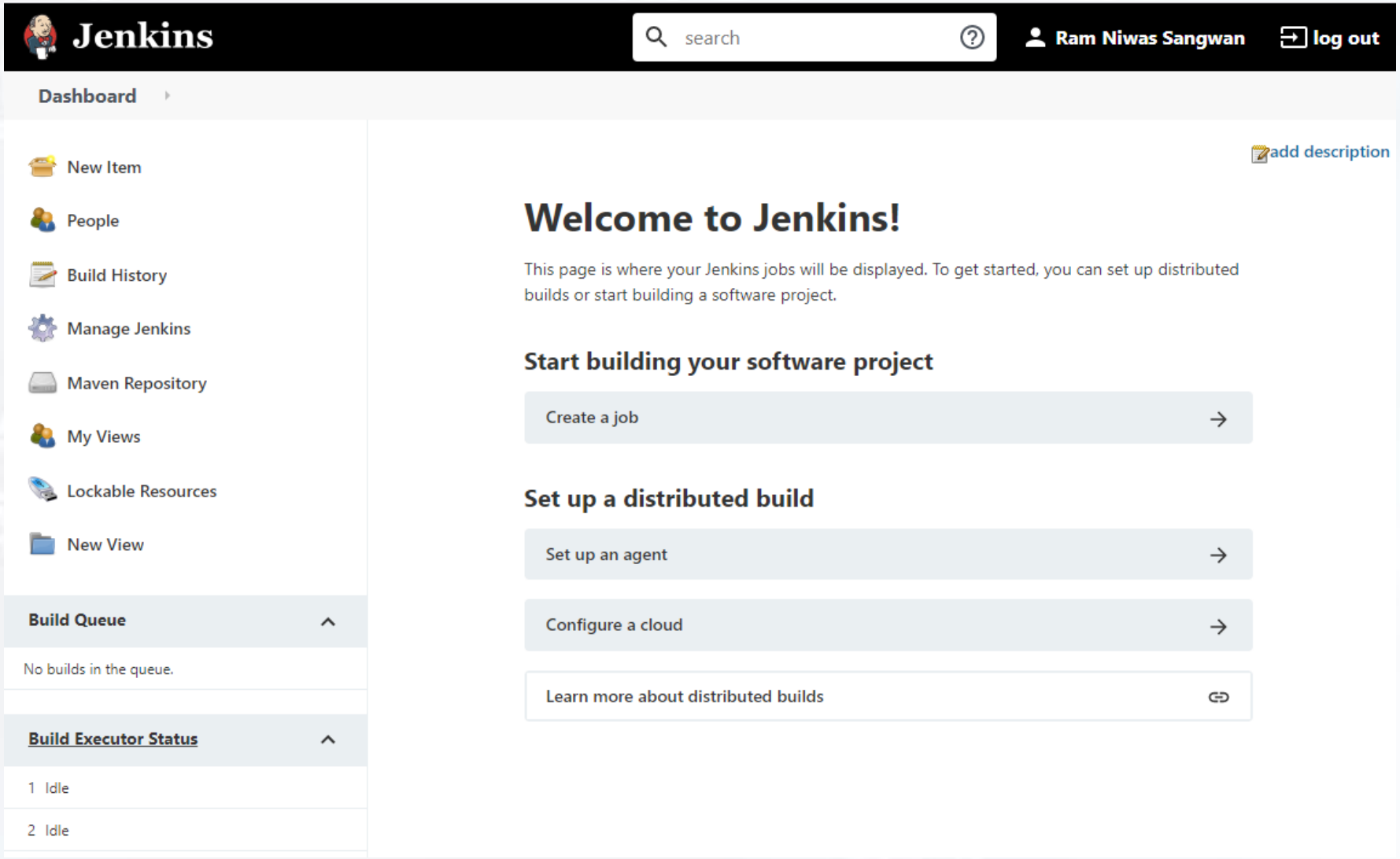
Install Maven and Restart Jenkins

- Plugin Installation will start. After installation is done click on the checkbox “**Restart Jenkins when Installation is complete and no jobs running**.





Explore the Dashboard



Thank You