Jenkins

Continuous Integration :-

Importance of Continuous Integration :-

1.**Improves Quality :-**

- Improves the quality by running multiple unit tests and analysing various static code

2.**Increases Productivity :-**

- Automating build of code saves a lot of time, there by increasing productivity

3.**Reduces Risk :-**

- Elimanate the risk of Potential human errors by automating test

Introduction to Jenkins

Features of Jenkins :-

- Easy Installation process
- Provides advance security
- Optimized performance
- Upgrades are easily available
- Light weight containers support
- Distributed team management

What is continuous Integration ?

- It is the process of automating the building and testing of code , each time one of the team member commits changes to version control

CI&CD → Continuous Deployment & also Delivery
Continous Integration

Popular Continuous Integration Tools :-

- Gitlab CI
- Code ship
- Bamboo
- Jenkins
- Team city
- Travis CI

In AWS Cloud perform CI&CD there are various services like code commit , code display , code build , code pipeline , code deploy & code guru

From AZURE Cloud we have AZURE DevOps :-

Azure Boards , Azure Repos , Azure Piplines , Azure testplans , Azure artifacts

What is Jenkins ?

- A Continous Integration Server which manages and control process such as plan , code , build , test , deploy , operate and monitor in DevOps Environment .

Jenkins :-

Plan , Code , Build , Test , Deploy , Operate , Monitor

Why Jenkins is So Popular ?

Jenkins :-

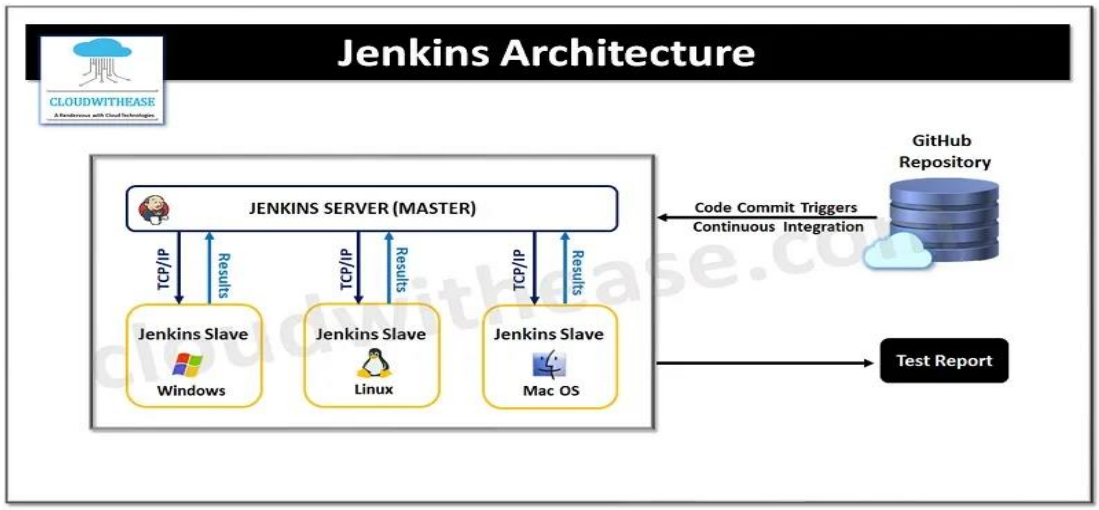
- Open Source
- Good Plugin support
- Good community support
- Fast and Reliable
- Good OS support
- Scripted Builds

Topics :-

- Jenkins Architecture
- Plugin Management in Jenkins
- Jenkins security Management
- Notification in Jenkins
- Jenkins Master slave Architecture
- Jenkins Delivery Pipeline
- Jenkins Declarative Pipeline

Jenkins Architecture :-

Source Control Management :-



Tabs :-

Update :-

- Shows updates to plugins already installed

Available :-

- Shows Plugins that are available for installation

Installed :-

- Displays Plugins Installed that have no updates

Advanced :-

- Lists Configuration of Http Proxy , allows manual upload of plugin and URL of Plugin site

In real time we can't install the plugins as simple as reason being the might be Challengs with proxy settings as well as vpn in order to avoid this we should configuration , http proxiy in Jankins , Plugin of advance tab

How To Install Jenkins On Windows ?

- **Installing Jenkins**
    - Docker
    - Kubernetes
    - Linux
    - macOS
    - **Windows**
    - Other Systems
    - WAR file

- Other Servlet Containers
- Offline Installations
- Initial Settings

Prerequisites

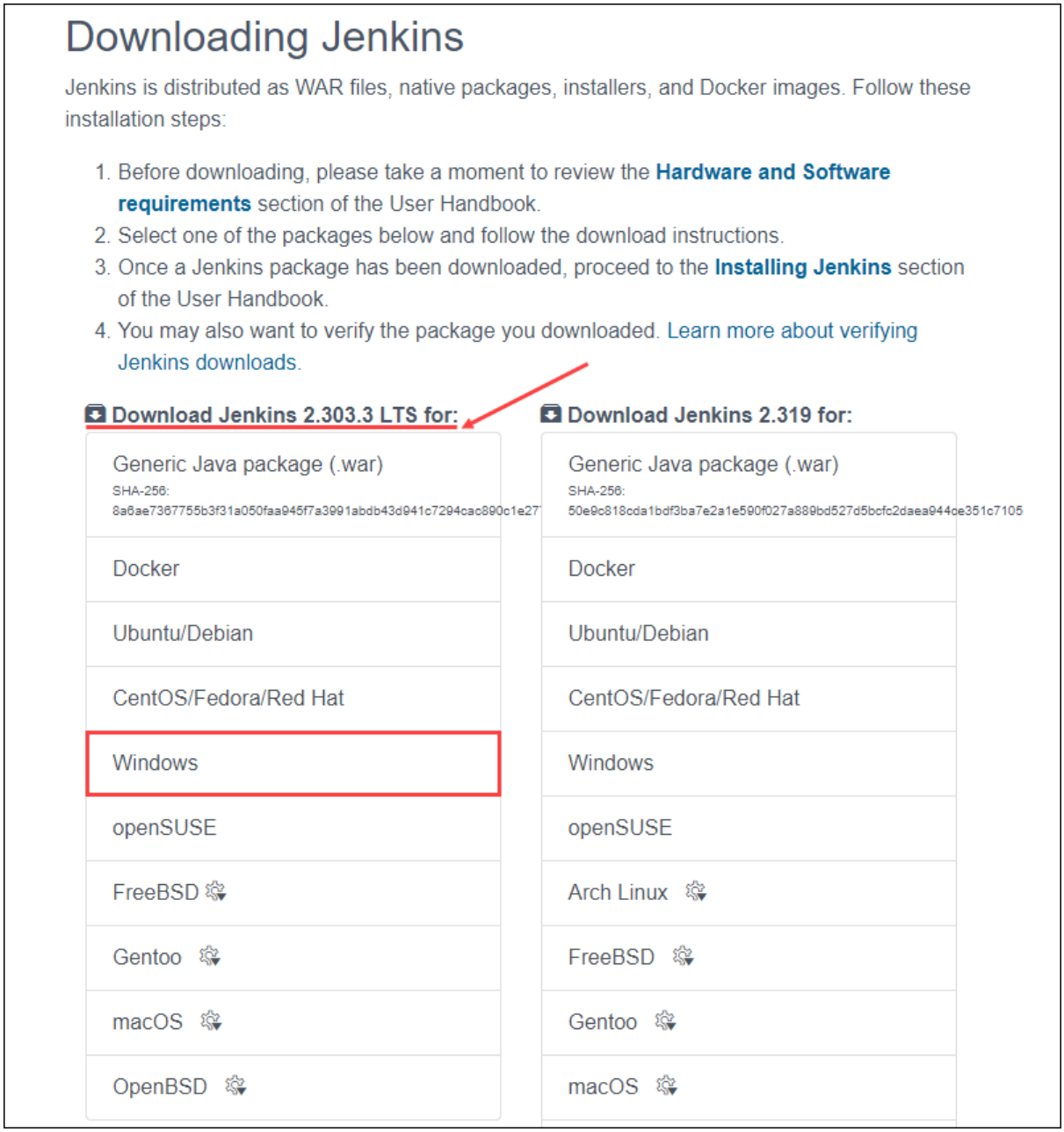
- 256 MB of RAM
- 1 GB of drive space (although 10 GB is a recommended minimum if running Jenkins as a Docker container)
- 4 GB+ of RAM
- 50 GB+ of drive space

Jenkins Tool is developed on Javacode

Prerequisites

- A system running Windows 10
- The latest copy of Java Development Kit or Java Runtime Environment installed
- Access to an account with administrator privileges

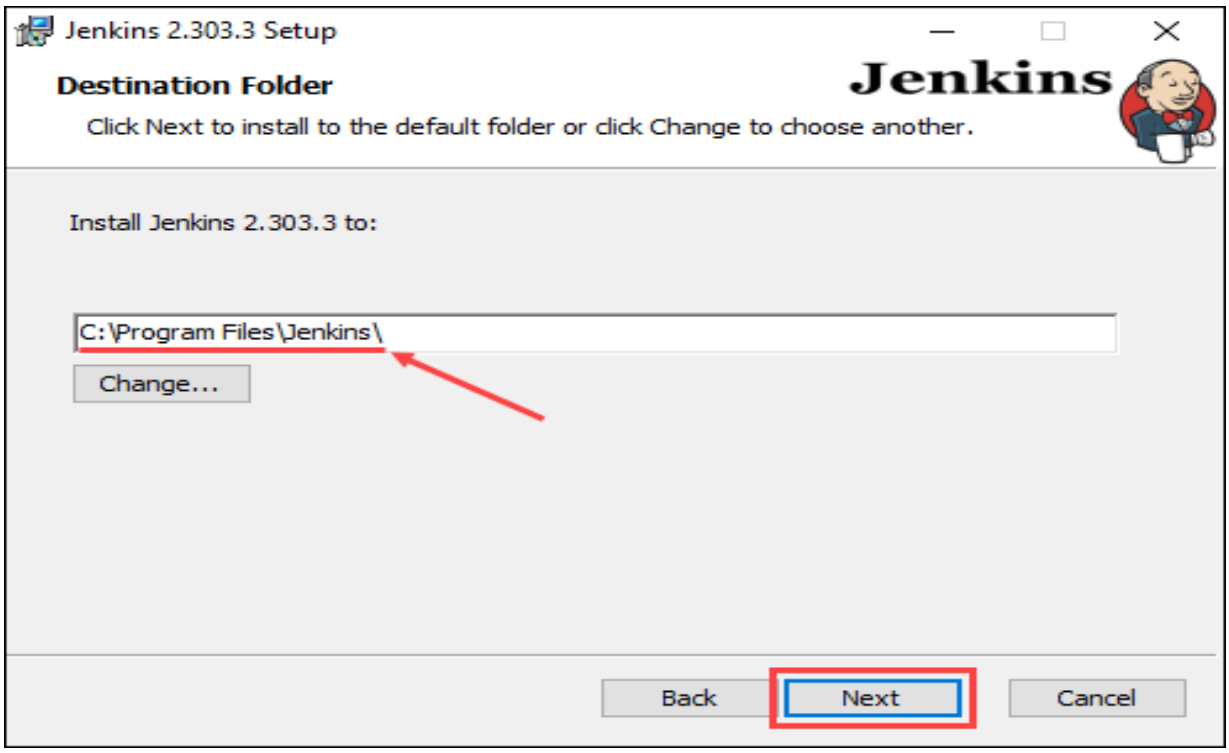
Browse to the **official Jenkins download page**. Under the **Downloading Jenkins** section is a list of installers for the long-term support (LTS) version of Jenkins. Click the **Windows** link to begin the download.



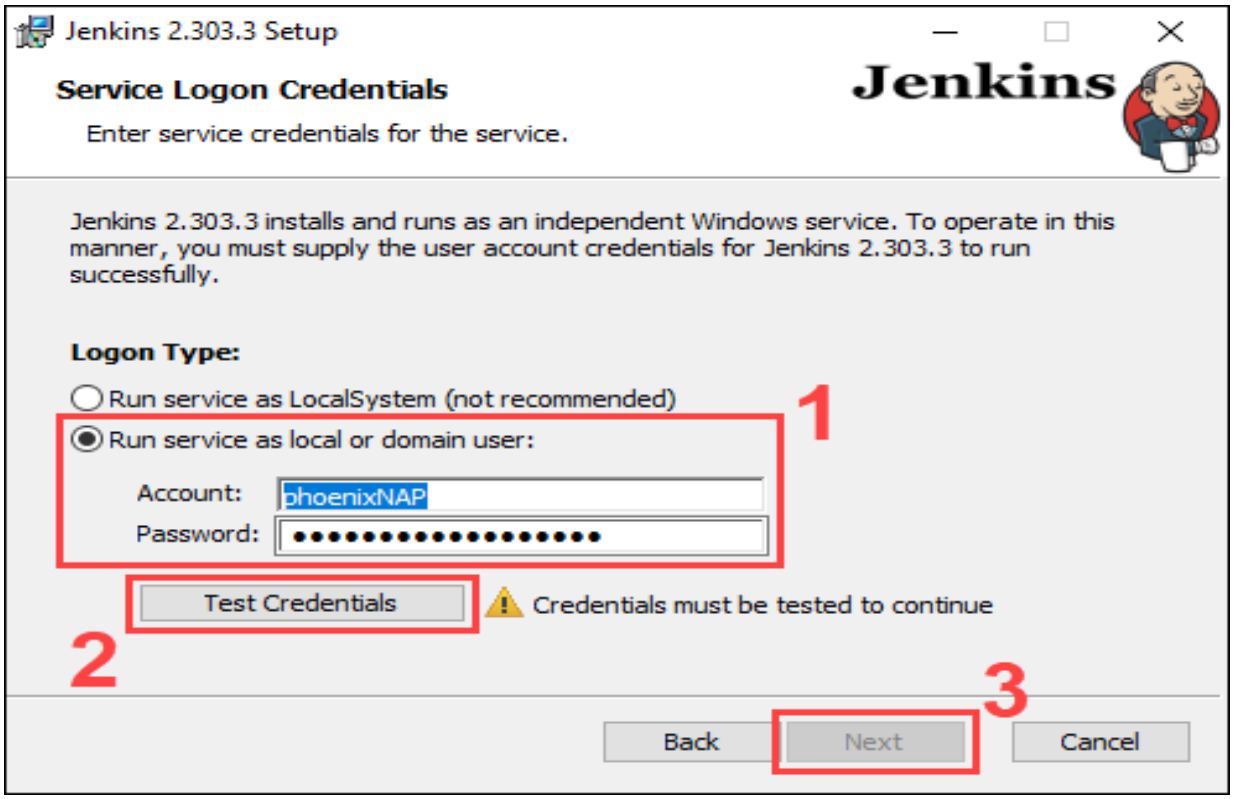
2. Once the download is complete, run the **jenkins.msi** installation file.

3. The setup wizard starts. Click **Next** to proceed.

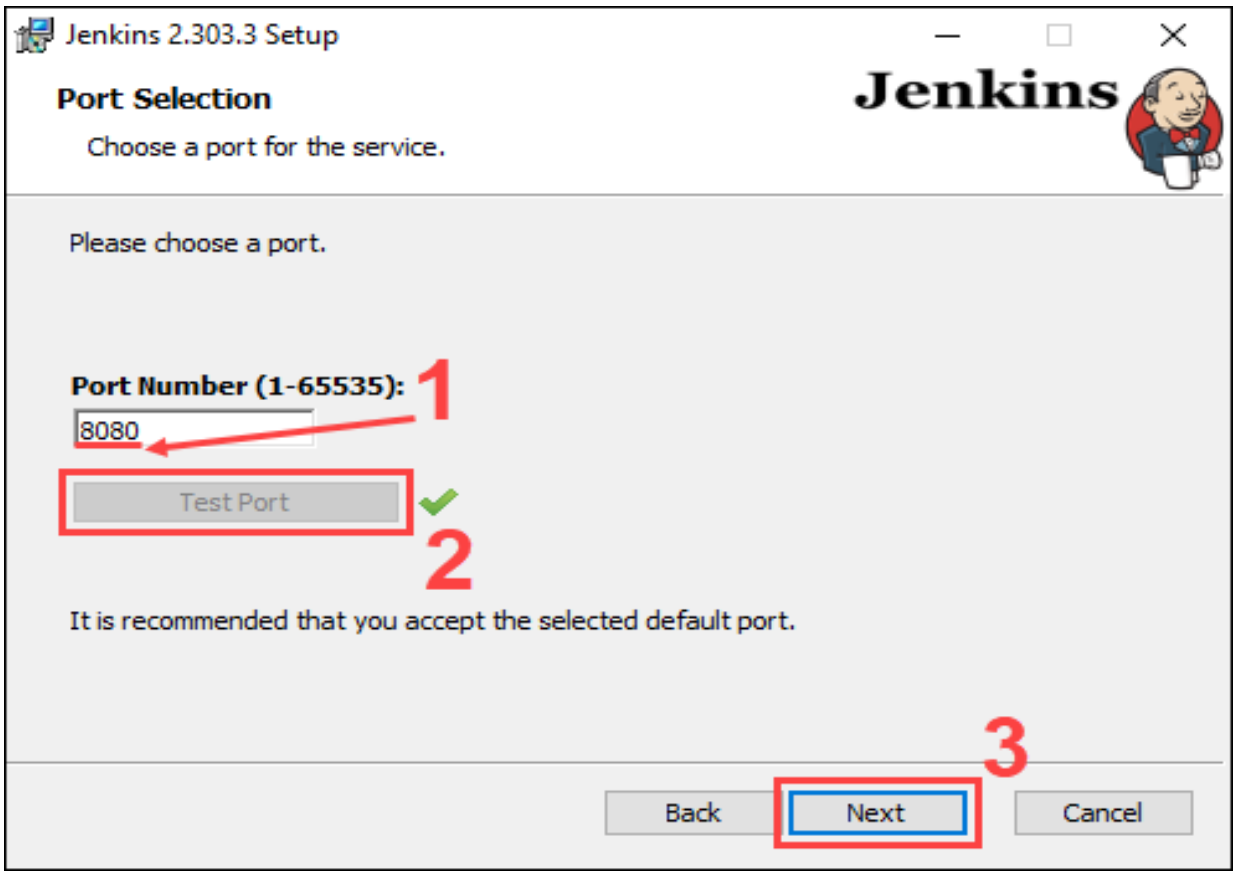
4. Select the install destination folder and click **Next** to continue.



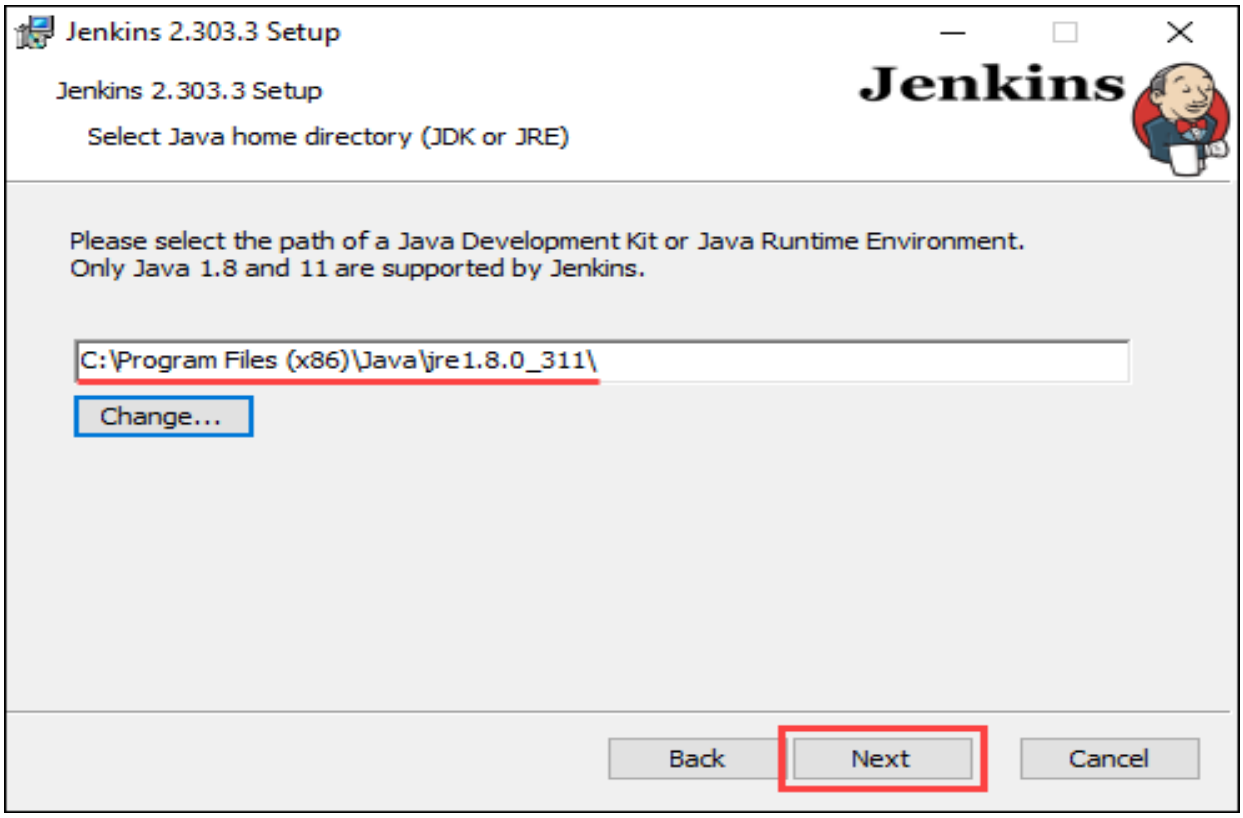
5. Under the **Run service as a local or domain user** option, enter the **domain** username and password for the user account you want to run Jenkins with. Click **Test Credentials** to verify the login data, then click **Next** to proceed.



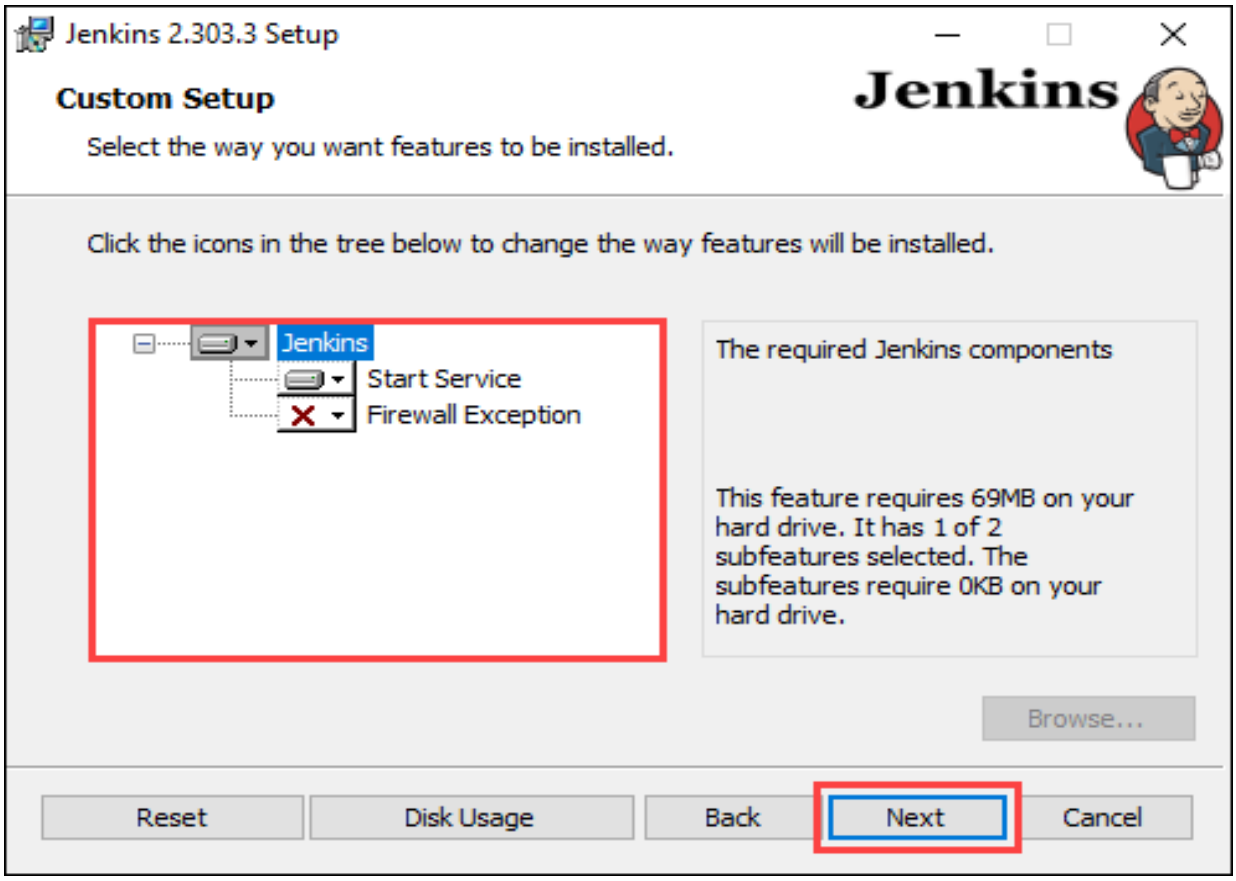
6. Enter the port number you want Jenkins to run on.
Click **Test Port** to check if the selected port is available, then click **Next** to continue.



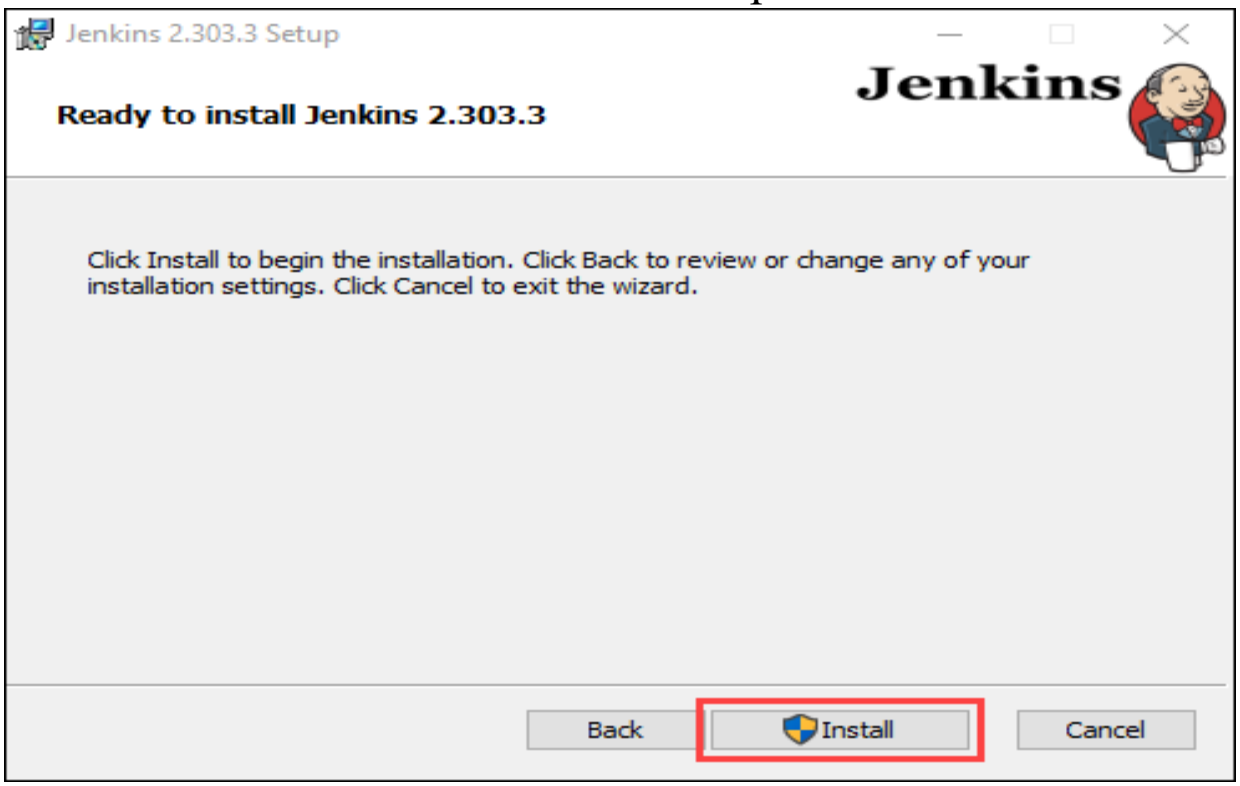
7. Select the directory where **Java is installed** on your system and click **Next** to proceed.



8. Select the features you want to install with Jenkins and click **Next** to continue.



9. Click **Install** to start the installation process.



10. Once the installation is complete, click **Finish** to exit the install wizard.

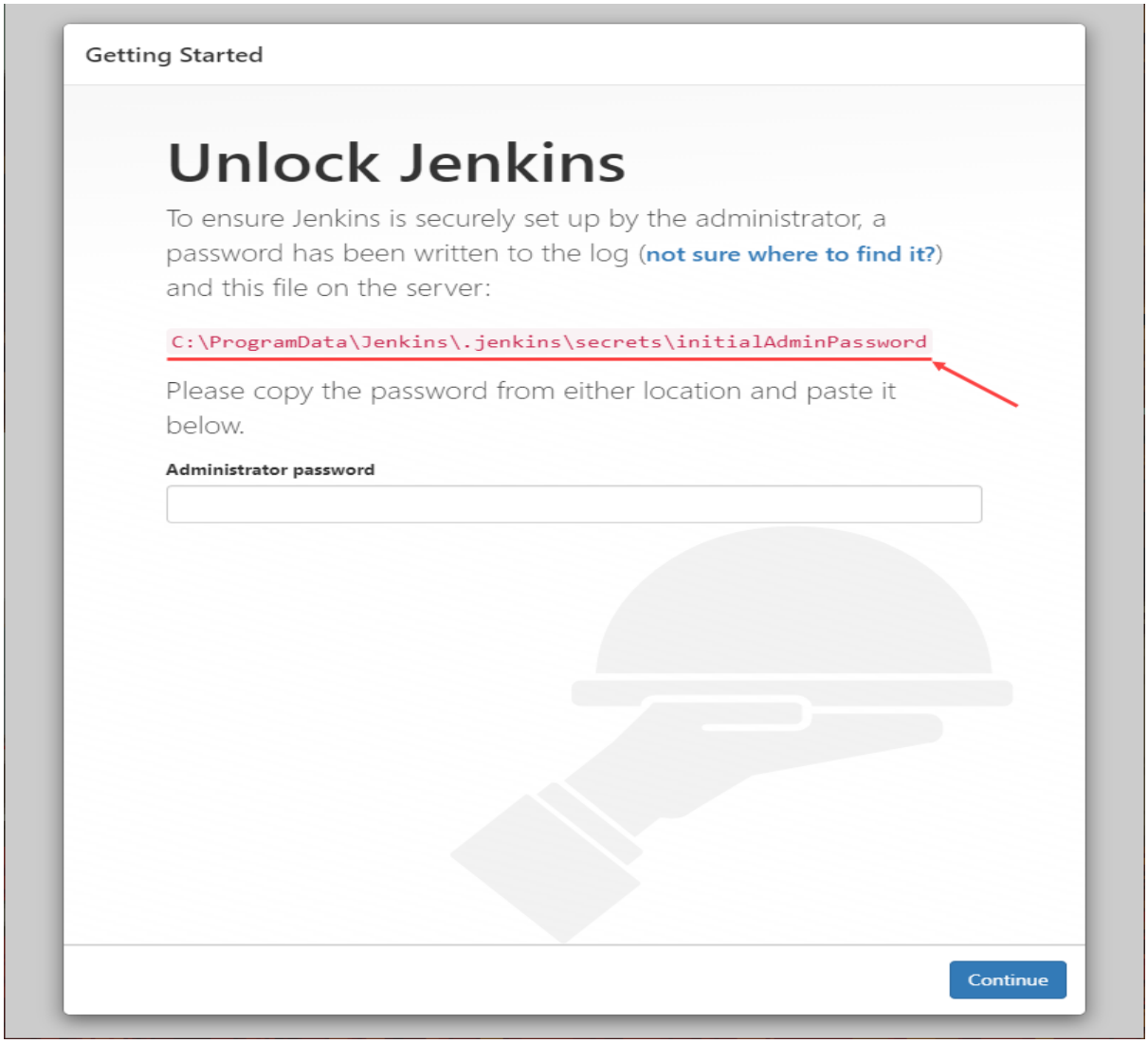
How to Configure Jenkins :-

- After completing the installation process, you have to unblock Jenkins before you can customize and start using it.

1.Browser, navigate to the port number you selected during the installation using the following address:

- http://localhost:[port number]
- ex :- http://localhost:[8080]
- 127.0.0.1:8080

2.Navigate to the location on your system specified by the Unblock Jenkins page.

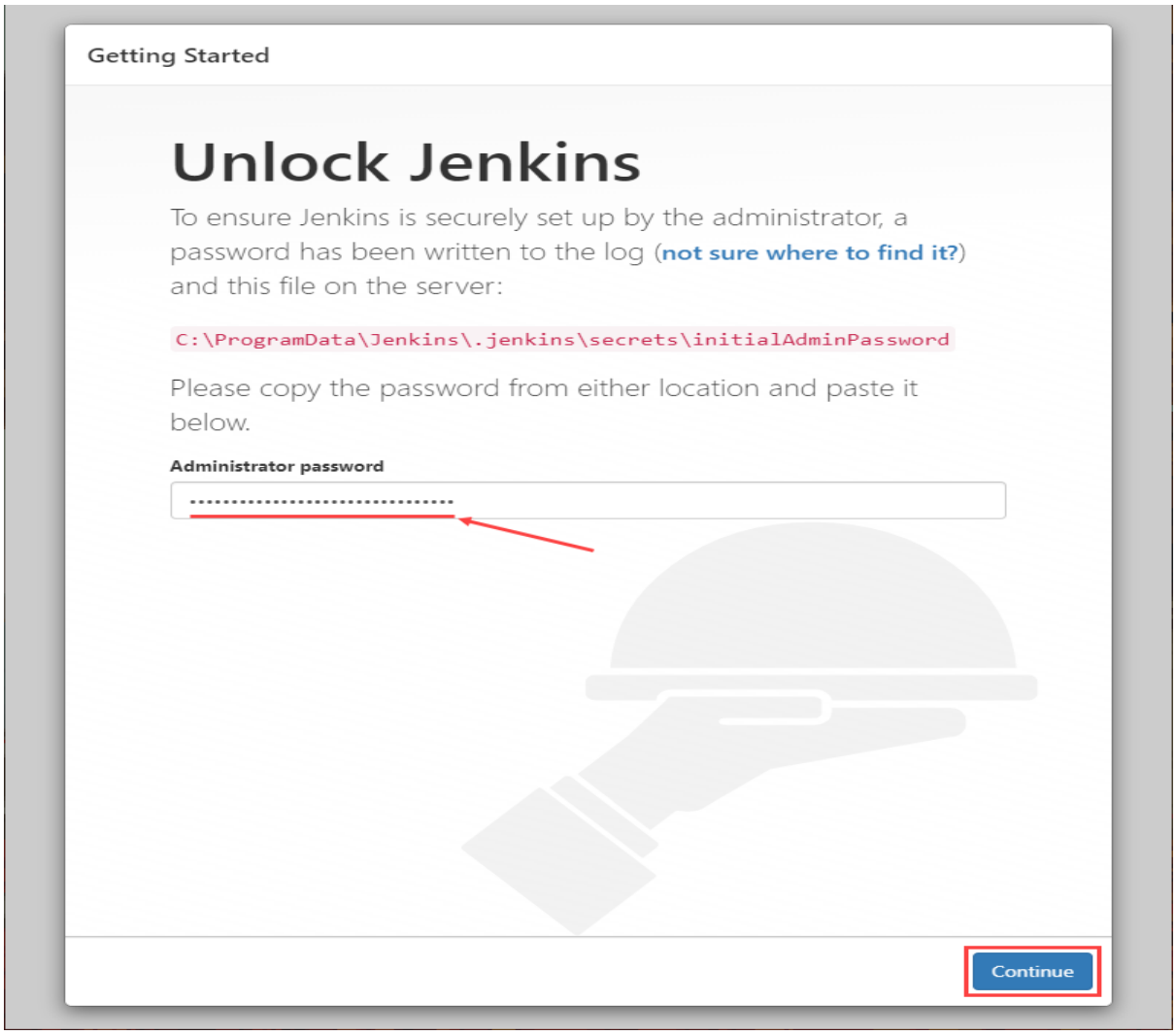


3.Open the **initialAdminPassword** file using a text editor such

as Notepad.

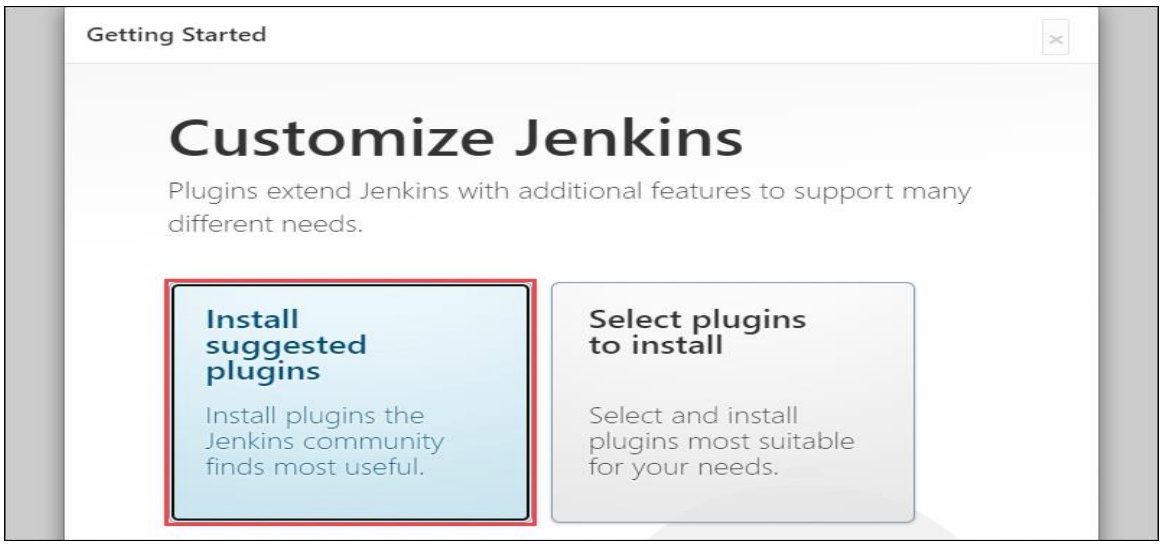
4. Copy the password from the **initialAdminPassword** file.

5. Paste the password in the **Administrator password** field on the Unblock Jenkins page and click **Continue** to proceed.

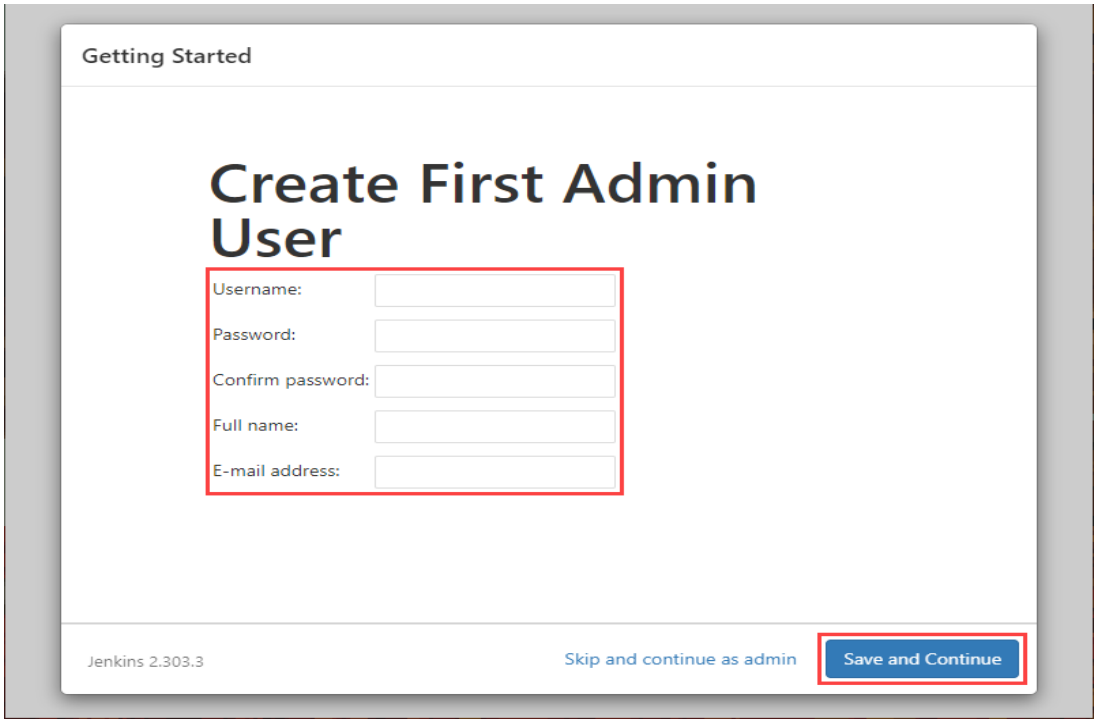


Customize Jenkins :-

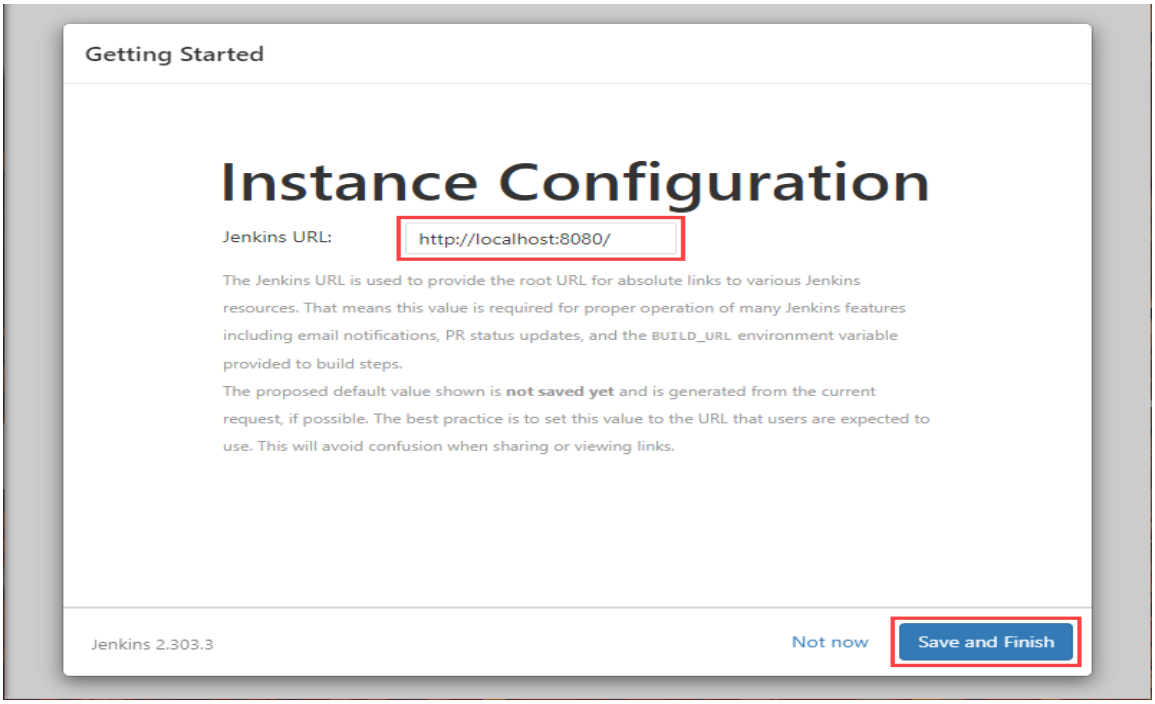
1. Click the Install suggested plugins button to have Jenkins automatically install the most frequently used plugins.



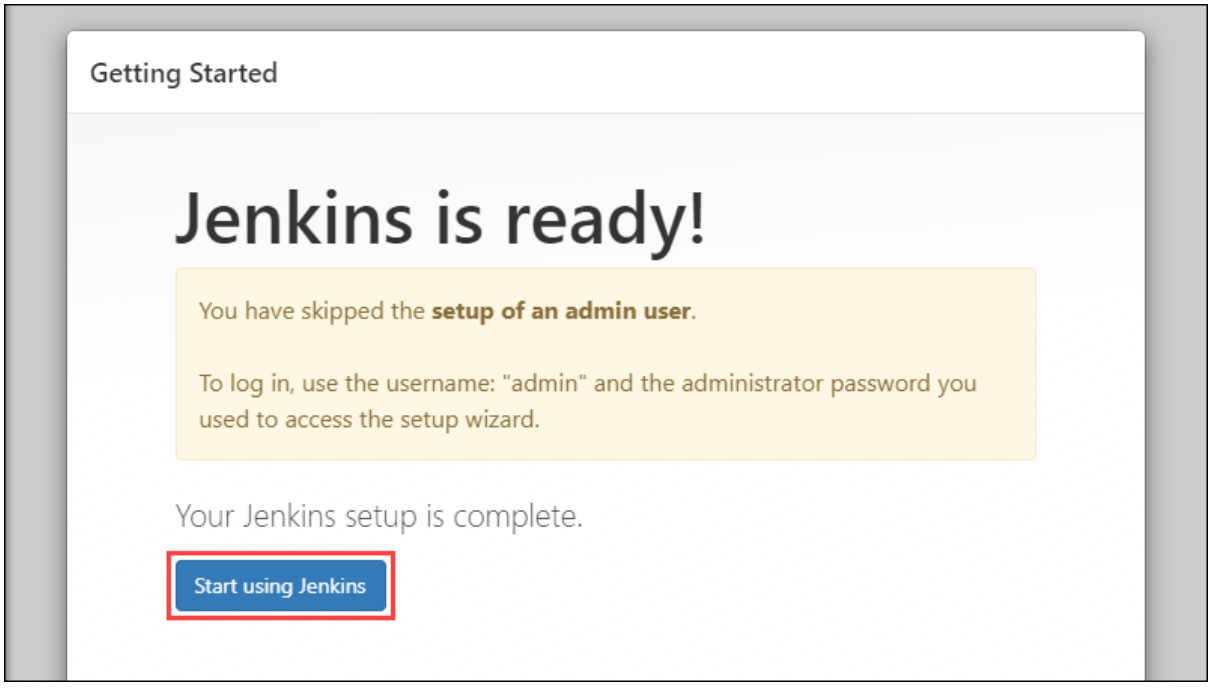
2. After Jenkins finishes installing the plugins, enter the required information on the **Create First Admin User** page. Click **Save and Continue** to proceed.



3. On the **Instance Configuration** page, confirm the port number you want Jenkins to use and click **Save and Finish** to finish the initial customization.



4. Click the **Start using Jenkins** button to move to the Jenkins dashboard.



How to Stop Jenkins server in windows :-

- In windows search for services & select (or) search for J
- Click on this stop (or) restart button

How to Restart the Jenkins :-

- If we go to URL http://localhost:8080/safeRestart
- Jenkins will try to pause jobs and restart once all running jobs are either finished or paused

Other way :-

- Localhost:8080/restart (don't use)

(06/02/2025)

Plugin Installation :-

There are 2 ways :-

  - Automatically
  - Manualy

- Once we login to Jenkins GUI , left side you will see the Option of manage Jenkins
- Click On manage Jenkins In first tab i.e., system configuration , under that we can see plugins
- Go to available Junkins , search for the plugin which is required , then select the plugin under the list
- Once we select the plugin then Install utton will be highlighted
- The Click on Install button

2. Manually :-

- Under Plugins tab , go for available plugins & search for it , Click on the plugin which is required
- Once we click on the plugin , click on Releases

- Under releases we can see the various versions of the Plugins
- Go for the Plugins which is required Specific version
- Under installation options we can get the direct link , click on direct link plugin will get downloaded to our local machine
- When we download Plugin in manual way again we need to deploy to Jenkins
- In Jenkins GUI Plugins tab , click on advance settings
- Their you can search for the choose file upload
- Which is download to local machine (or) local system

Plugins :-

- Add , remove , disable (or) enable Plugins that can extend the functionality of Jenkins
- When we install Plugin in automatic way Plugins extention is JPI (Jenkins Plugins)
- When we Install Plugin in Mannual way is HPI (Hudson Plugin)

How to Uninstall the Plugin ?

- Manage Jenkins → Plugins then install Plugin , then search for Plugin name which we want to uninstall
- Select the plugin and then uninstall button will be highlighted once clicked on uninstall , plugins will get uninstalled

How to Update the Plugin ?

- Manage Jenkins →Plugins →Updates
- Select the Plugin and then Update button will be Highlighted

- Once clicked on update, Plugin will get Updated

❖ **In real time we shouldn't update the plugin directly (because there maybe challenges (or) encounter)**

- It will recommended to in your local machines where other team member willn't get effect if anythink goes wrong

How to Create the Users ?

- Manage Jenkins → Security → Users (Create / Delete / Modify users that can log in to Jenkins)
- Click on the users we can see the create user option after clicking on the users option we can see the username , password , confirm password , full name , email address the click on create users

How to give permissions to the Users ?

- Manage Jenkins → security (secure Jenkins , define who is allowed to access / use the system)
- Add a user button , click on it
- It will ask userID , give user ID & click on OK button
- You can see user is add then give the required permission to the user by checking the box
- The minimum access to us is over all read
- By default what ever we create users for Genkins own user Database
- This is suitable for smaller set up where you have no existing user database
- The other option is LADP (Lightweight Directory Access Protocol)

Authorization strategy :-

- By default we will be “ Project – based Matrix Authorizate Strategy ” , with the we can manage the authorization baced the option available under Authorization strategy
- Cradentials , agent , Job etc

Matrix Authorization Strategy :-

Use Cases :-

- Matrix , Authorization , allows Configuring the lowest level permission , such as starting new builds iteams or deleting configuration items

Jenkins Projects (or) jobs :-

Free style project :-

- Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

Maven Project

Pipeline (workflows) :-

- 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, etc.

Folder :-

- Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

Multi branch pipeline :-

- Creates a set of Pipeline projects according to detected branches in one SCM repository.

Organization folder :-

- Creates a set of multibranch project subfolders by scanning for repositories.

RBAC (Roll Base Access control) , where we can secure the Jenkins

How to Create Job (or) Project ?

- In Jenkins GUI , you have “ + New item ” , click on the new item , Enter item name (or) job name
- Then select Item type , click on ok button it will be highlighted

- Job configuration is consists of “ General ” (Where we can give information (or) description about the project)
- Source code management where we will be passing URL’s of the repositories , build triggers (In what way we wanted to trigger (or) build the Job) “ Build environment ” , “ Build steps ” (where we can add tasks) , post build actions

How to Create Free style Job ?

- New item → Job name → Select click on Ok
- Go to Build steps configuration of Job
- Click on “ Add Build step ” drop down button
- Select “ Execute windows batch command ”
- Select DIR , click on save
- Click on Buildnow , click for the console output

Blue Ocean :-

- Blue Ocean re thinks the Jankins user experience Designed from the ground up for Jenkins Pipeline and Compatible with freestyle jobs