Continuous Integration with Jenkins

In the previous chapter we have seen how development process is followed by developers for creating softwares and even maintain that software. So as the developers write code to create softwares, they also merge all that code into a centralised repository or Version Control System like Github.

This code is pushed into repositories several times a day and over the period of time all of the code gets merged. Traditional software development methods don't dictate how frequently or regularly you integrate all of the source on a project. Programmers can work separately for hours, days, or even weeks on the same source without realizing how many conflicts (and perhaps bugs) they are generating.

1. Integration is painful

Agile teams produce workable and robust code in each iteration. All that code if built and evaluated returns lot of conflicts, bugs and errors. Developers needs to solve those conflicts and issues before moving to next iteration. The more programmers are sharing the code, the more problematic this is.

For these reasons, agile teams often therefore choose to use Continuous Integration.

2. Some Terminologies before we begin.

Source Code All the code that developers writes to create the software is called as Source Code.

The build process It is process by which source code is converted into a stand-alone form that can be run on a computer. For example, a source code written to develop a windows software when built will create a .exe or .msi file. Another example if a Java Source code is built it may create a .jar, .war or .ear file. This deployable piece of software is called as Artefact.

The source code can be built, packaged and deployed manually. But there are some build tools that make developers life easy when it comes to building artefact or even deploying it. These are called as build automation tools.

Some build tools

- Ant
- Maven
- Gradle
- Msbuild
- Nant

Unit Testing :

Unit testing simply verifies the individual unit of code(mostly functions) works as expected. Developer along with writing the code will write the test cases that can be executed at the build time. Some test cases can be automatically generated.

The objective of unit testing is to isolate a section of code(unit) and verify its correctness.

3. What is Continuous Integration

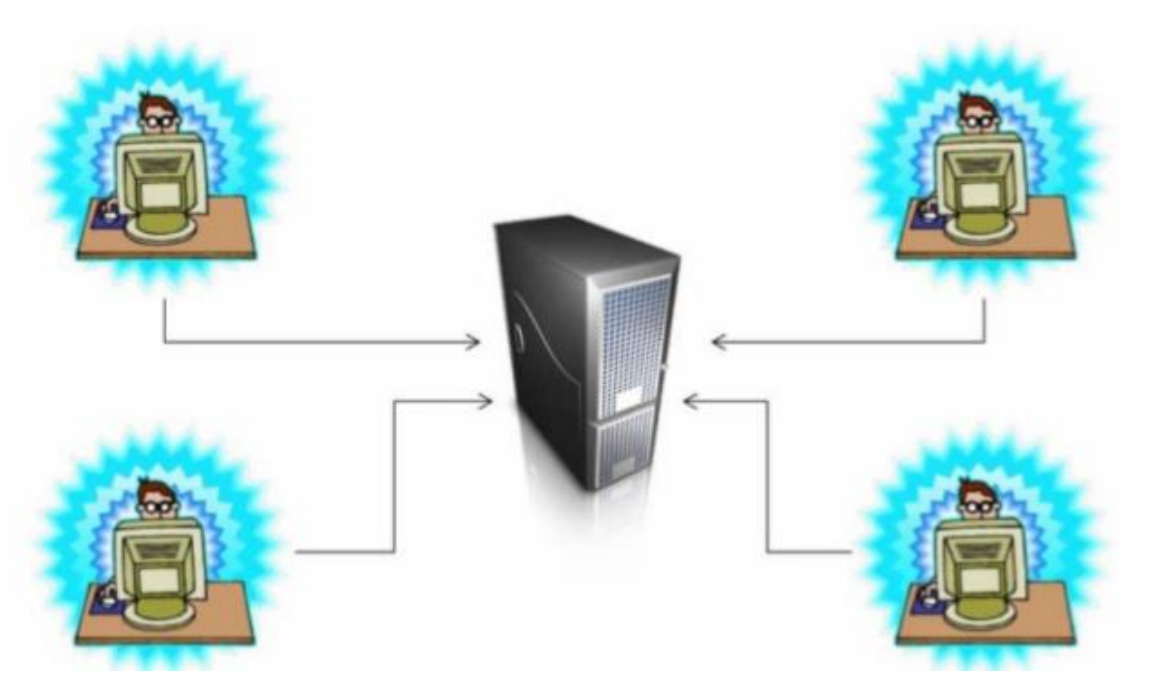
Continuous Integration (CI) is the process of automating the build and testing of code every time a team member commits changes to version control. CI encourages developers to share their code and unit tests by merging their changes into a shared version control repository after every small task completion. Committing code triggers an automated build system to grab the latest code from the shared repository and to build, test, and validate the full master branch (also known as the trunk or main).

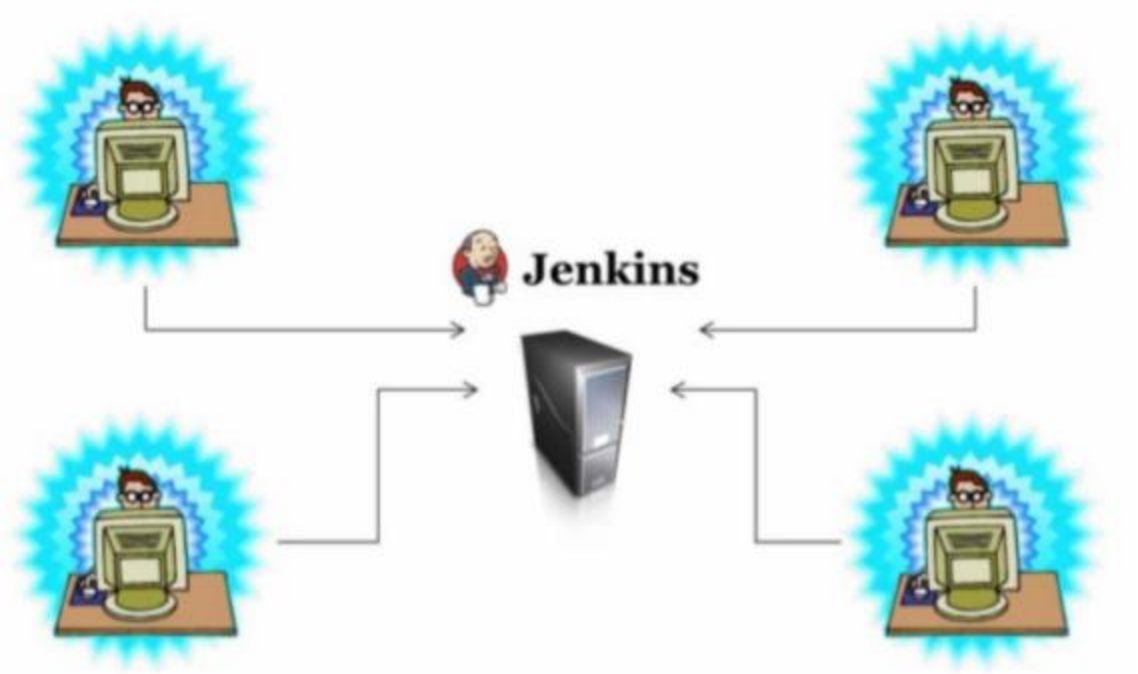
The Problem

Developers will write the code and BUILD it in their local system. Once developers test the code and verify locally they push it to the centralised repository like github. Similarly, all the developers would be pushing their code to VCS several times a day. Developers would be working in their own silos or caves and keep writing the code until they finish a particular task or the project. Now all the code which developers have pushed into the VCS, if built and tested will return lots & lots of conflicts, error due to which build will fail.

The Solution

To get around this very problem whenever the developer push the code to the VCS it should be fetched, built & tested by a build server at the same time.



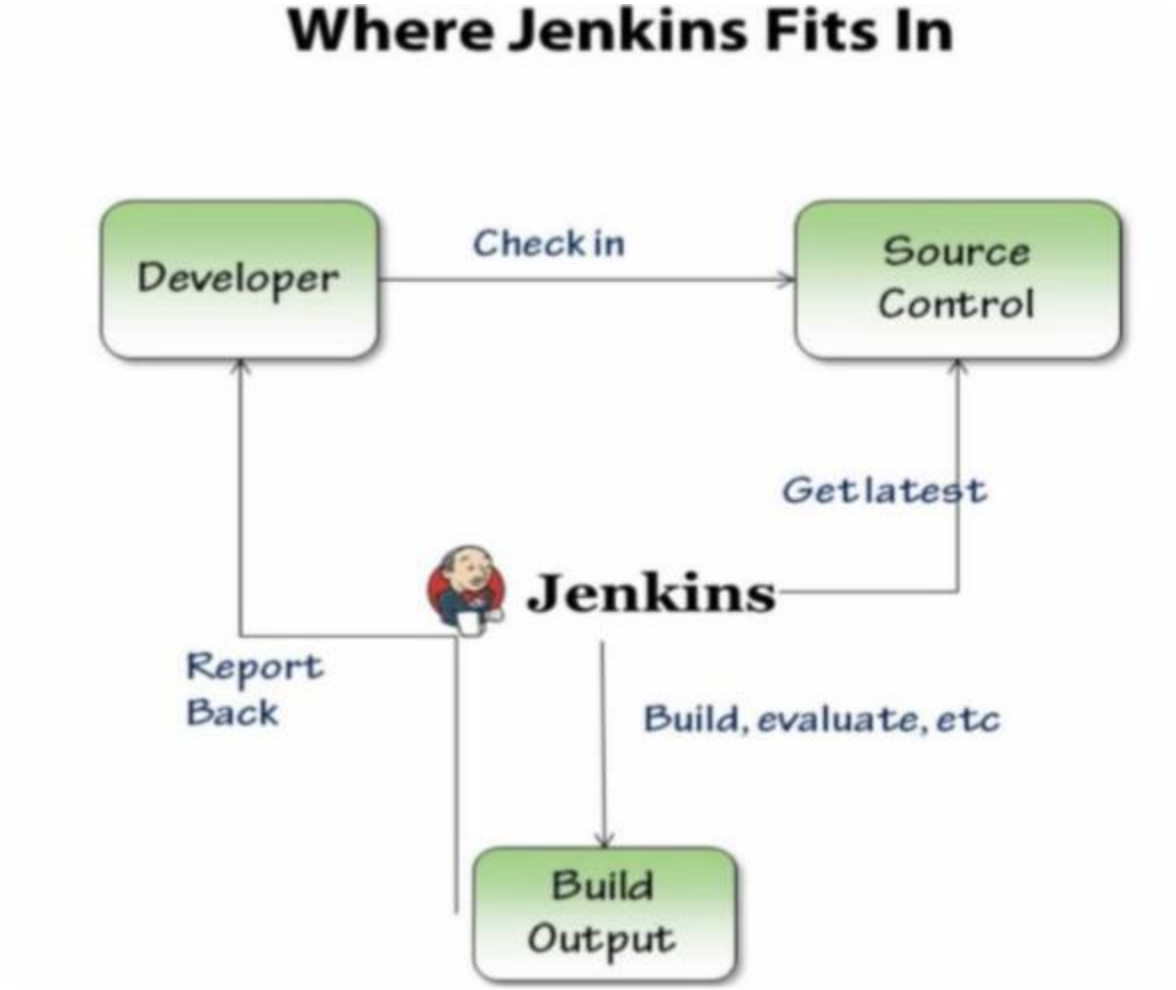


4. What is Jenkins

Jenkins is a continuous integration server which can fetch the latest code from VCS, build it, test it and notify it to the developers. Jenkins can do many more things apart from just being a CI server. It was originally known a Hudson, Oracle inc owns Hudson now. Jenkins is an open source project written by Kohsuke Kawaguchi.

Jenkins is a java based web application server. As a prerequisite, we need to setup first Java on the machine to run Jenkins server.

Where Jenkins fits in



5. Features of Jenkins

OpenSorce

As jenkins is opensource there is lot contribution all around the world to the jenkins software. It has all the latest and greatest feature that developers integrating into it regularly.

Extensible

Jenkins comes with lot of goodies but its just not limited by that, Jenkins main power is its extensibility that can be achieved by installing plugins into it.

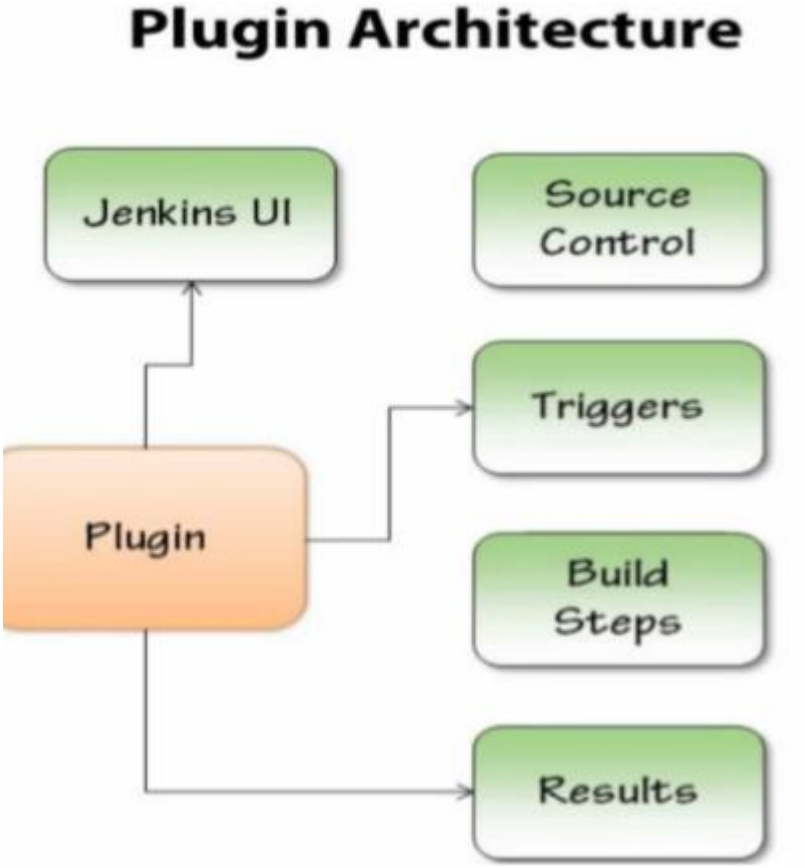
Jenkins opensource community has written tons of plugins, these plugins can do variety of tasks, like integration with external tools or servers.

- VCS plugins – git, svn, subversion etc
- Build plugins – Maven, ANT, Msbuild etc
- Notification plugins – Email, chat, sms etc

- Cloud plugins – Create cloud instances, deploy code to cloud services etc
- Testing plugins – Code analysis, Unit test case, Static code analysis etc

The list of plugins is very long, whenever we want Jenkins to do some tasks just search for that plugin and most of the time you will find something.

For example, if you want jenkins to deploy java artefact to tomcat server, search for the plugin named “deploy to container”.



6. Jenkins Setup On Ubuntu Server :

Jenkins can be installed on windows, Linux or Mac OS. Jenkins just needs java software to run.

In this tutorial, we will install jenkins on a ubuntu server. You can setup a vm or a cloud instance. Prereqs Java runtime environment/ JRE can be installed on the system but we will install JDK as we will setup maven moving along and build some java code. To Build the java code we will need JDK.

```
sudo add-apt-repository ppa:openjdk-r/ppa
```

```
sudo apt-get update
```

sudo apt-get install openjdk-8-jdk

Installing Jenkins on Ubuntu :

```
wget -q -O - https://pkg.jenkins.io/debian/jenkins-ci.org.key | sudo apt-key add -
sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'
sudo apt-get update
sudo apt-get install jenkins
```

Install git client and maven in jenkins server.

We will integrate Jenkins with github to download the source code.

We are testing the java source code which will be built by Maven, so we also need to install Maven on Jenkins server. This is not a mandatory requirment to run Jenkins if you are not using git and maven.

sudo apt-get install git

sudo apt-get install maven

7. Jenkins Setup On Redhat/centos/fedora Server :

- sudo -i
- yum search java
- Here we can identify all the versions of java.
- yum install -y java-1.8.0-openjdk.x86_64
- yum install –y wget unzip

- sudo wget -O /etc/yum.repos.d/jenkins.repo http://pkg.jenkins-ci.org/redhat/jenkins.repo
- sudo rpm --import https://jenkins-ci.org/redhat/jenkins-ci.org.key
- sudo yum install jenkins
- sudo yum install java
- sudo service Jenkins start
- sudo yum install git
- sudo yum install maven

Accessing Jenkins Jenkins runs on port 8080 by default

Open up a browser and use below url.

http://jenkinsIP:8080 (or) http://localhost:8080

Jenkins will set a random password to unlock jenkins setup.

The password would be stored in /var/lib/jenkins/secrets/initialAdminPassword file. Read that file and get the password. Use that password to unlock Jenkins

cd /var/lib/jenkins/secrets/initialAdminPassword

```
vagrant@vagrant-ubuntu-trusty-64:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
f05498523c3547f89754376def663c6e
vagrant@vagrant-ubuntu-trusty-64:~$
```

Jenkins gives you an option to install some suggested plugins at the time of setup. You can select individual plugins or install suggested group of plugins. Select suggested plugin for now.

Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

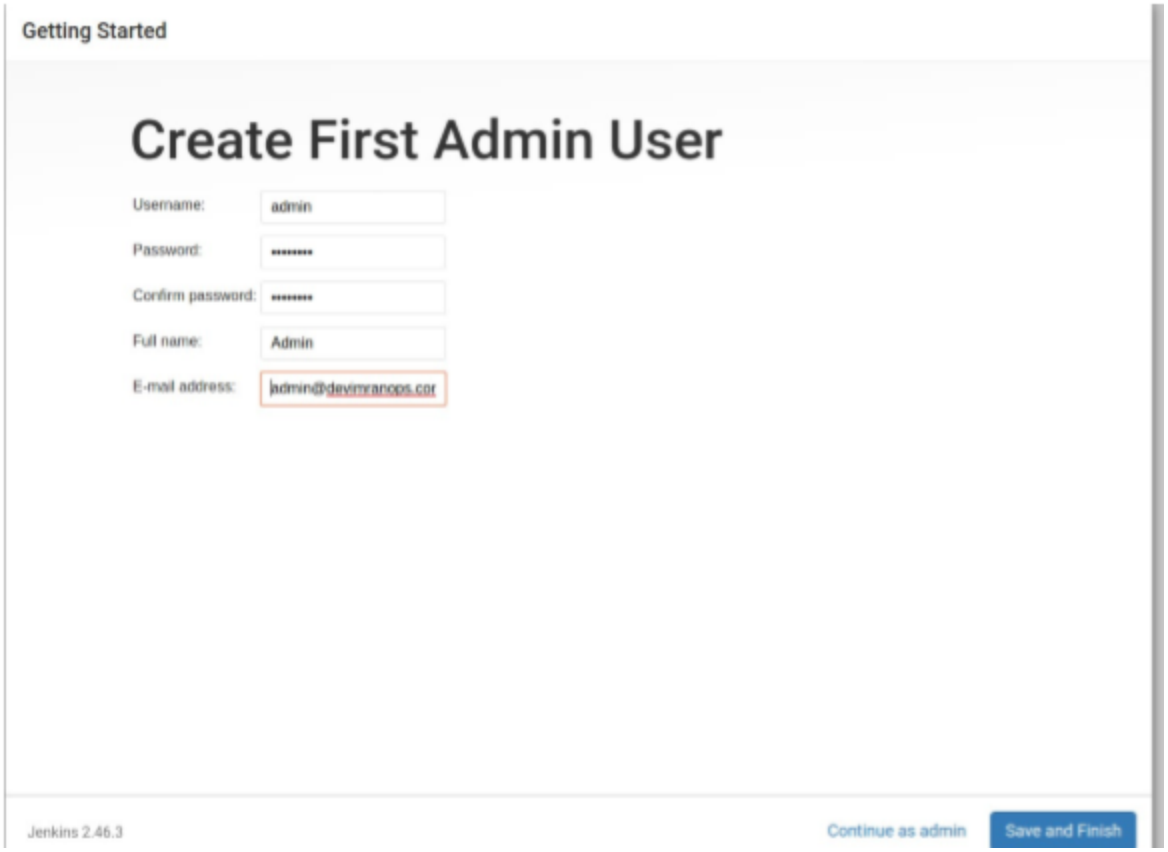
Select plugins to install

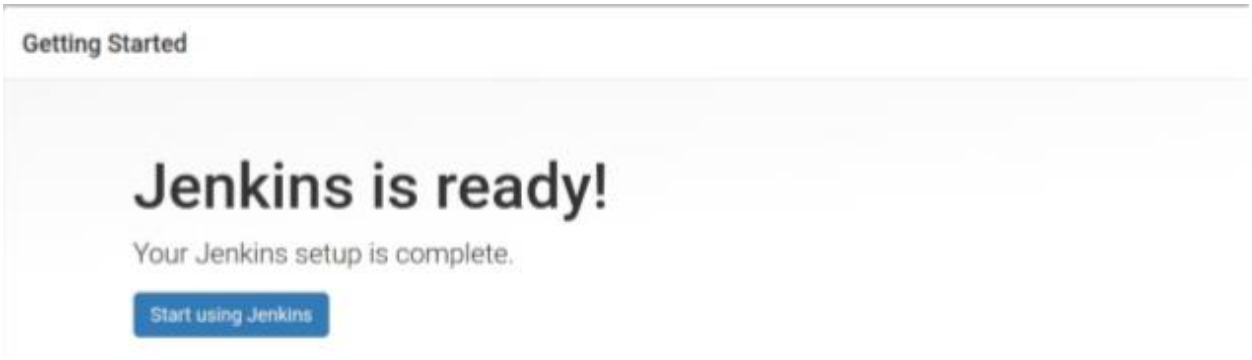
Select and install plugins most suitable for your needs.

Getting Started

✔ Folders Plugin	✔ OWASP Markup Formatter Plugin	✔ build timeout plugin	✔ Credentials Binding Plugin
✔ Timestamper	✔ Workspace Cleanup Plugin	✔ Ant Plugin	✔ Gradle Plugin
✔ Pipeline	GitHub Organization Folder Plugin	✔ Pipeline: Stage View Plugin	Git plugin
Subversion Plug-in	SSH Slaves plugin	✔ Matrix Authorization Strategy Plugin	✔ PAM Authentication plugin
✔ LDAP Plugin	Email Extension Plugin	✔ Mailer Plugin	

```
** Pipeline: Job
** Pipeline Graph Analysis Plugin
** Pipeline: REST API Plugin
** JavaScript GUI Lib: Handlebars bundle plugin
** JavaScript GUI Lib: Moment.js bundle plugin
Pipeline: Stage View Plugin
** Pipeline: Model API
** Pipeline: Basic Steps
** Pipeline: Stage Tags Metadata
** Authentication Tokens API Plugin
** Docker Commons Plugin
** Docker Pipeline
** Pipeline: Declarative Extension Points API
** Pipeline: Declarative Agent
```





Main dashboard of jenkins.

Jenkins setup done Thanks

How to install Jenkins as stand alone in windows :

1. Download Jenkins.

- Go to google.com and type Jenkins and click on jenkins official website first link of search.
- Just download .war filefrom LTS version not from the release version why because LTS is more stable than release version.

2. Place the var file into any location on your system.

So now what you have to do is, if you want to start jenkins you have to follow the below steps

3. go to command promt in windows.

Goto folder where jenkins.war

4. type "Java -jar jenkins.war "

5. store the admin password which was generated while starting up the jenkins.

6. goto browser and type : http:/localhost:8080
And it will ask for initial password, we have to give which was saved in previous step.

7. Install required plug ins.

8. Get started with Jenkins.

9. how to change port Number 8080

java -jar jenkins.war –httpPort=9090

http:/localhost:9090 - Jenkins should run here.

How to set up Jenkins on Tomcat :

Q : Why should we deploy Jenkins on Tomcat ?

Because we can start Jenkins on his own stand alone server and Jenkins comes bydeafult with his own servlet containers which is **Jetty** and **winstone.** And we can start deploy Jenkins on other servlets like tomcat.

So why exactly do we need it. ?

So what Is the difference between running Jenkins standalone (Jetty/winstone) Vs Running Jenkins on Tomcat.

So one of the benefit of running jenkins on tomcat servlet is that you can start all your applications on a single server. So mostly we have all web aplications only so if we want to keep jenkins as well it is good to start at a time.

We have many other containers also available

Glassfish
Tomcat
Jboss
IBM Websphere
Jetty
Jonas
Weblogic
Apache Geronimo 3.0
Liberty Profile

so we can deploy our Jenkins on these kind of all containers but here we are using Tomcat here why because Tomcat is one of the most commonly used container.

Pre Requisites for Jenkins:

- Tomcat 5 or above
- Java on your system (Java 7 or above should be available)

1. Download Tomcat 8

- goto google.com and type Tomcat and click on first link and download tomcat 8 from

downloads tab

2. Unzip and place Tomcat folder at any location on your system.

3. Copy/Place the Jankins.war file inside Tomcat/Webapps folder.

To start Tomcat server:

4. Go to command promt

go to cmd - Tomcat/bin - startup.bat or startup.sh (or)

go to cmd - Tomcat/bin - catalina.out start

go to tocat/bin directory – Click on startup.bat or startup.sh

go to browser type : http:/localhost --Then tomcat is up and running

To stop Tomcat server :

go to cmd - Tomcat/bin - shotdown.bat or shotdown.sh (or)

go to cmd - Tomcat/bin - catalina.out stop

go to tocat/bin directory – Click on shotdown.bat or shotdown.sh

How to change default port number 8080 :

Goto Tomcat – conf
Edit server.xml file

we have to change number at connector port.

Save and restart tomcat server.

Now Tomcat container up and running fine

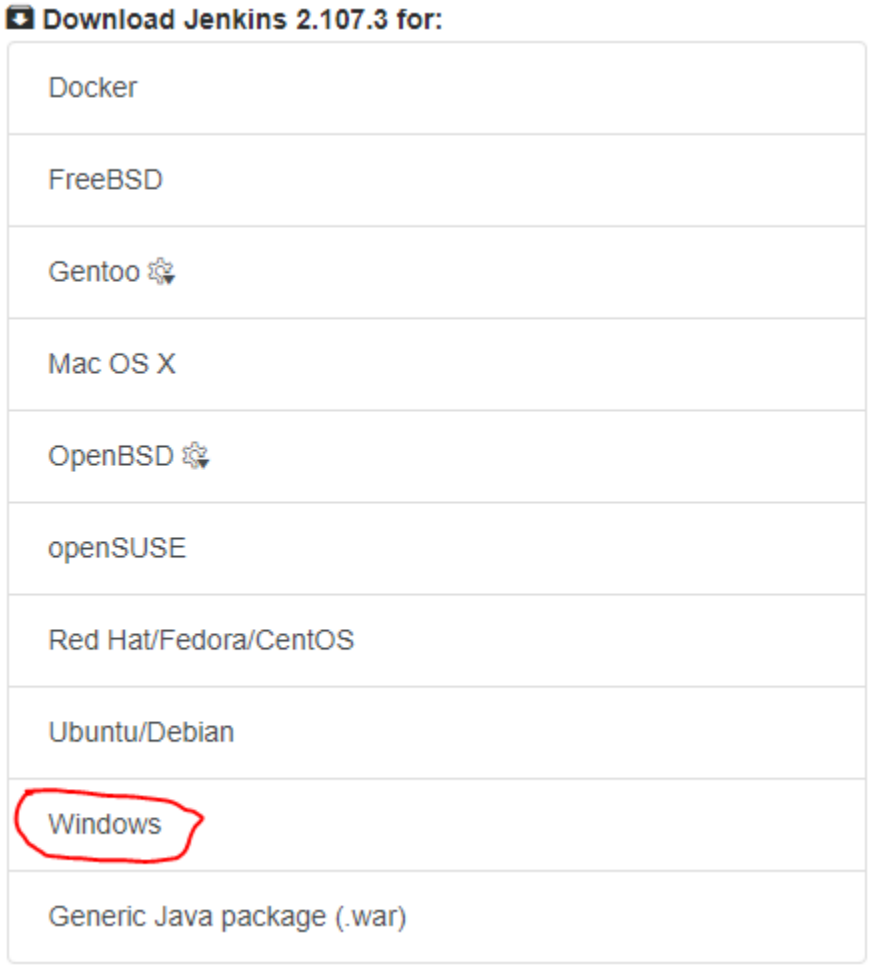
Now we have to start our Jenkins on Tomcat

goto browser and type : http:/localhost:8080/Jenkins

How to install Jenkins as a service (LTS – Long Term support) in windows :

1. Download Jenkins.

- Go to google.com and type Jenkins and click on jenkins official website first link of search.
- Just click on windows tab from LTS version not from the release version why because LTS is more stable than release version.



We have to run the downloaded file. And says next next and again you are going to get same above screens and we have to give same options but only thing password can be available on this directory.
C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword

After doing this native package installation we are able to Jenkins as a service in our system, so when ever you reboot our system it will get automatic restart but that wont be happen in standalone installation.