Assignment - 4

Aim: To understand Continuous integration, Jenkins installation.

LO Mapped: LO1, LO3

Theory:

Jenkins is an open-source automation server. With Jenkins, organizations can accelerate the software development process by automating it. Jenkins manages and controls software delivery processes throughout the entire lifecycle, including build, document, test, package, stage, deployment, static code analysis and much more.
You can set up Jenkins to watch for any code changes in places like GitHub, Bitbucket or GitLab and automatically do a build a with tools like Maven and Gradle. You can utilize container technology such as Docker and Kubernetes, initiate tests and then take actions like rolling back or rolling forward in production.

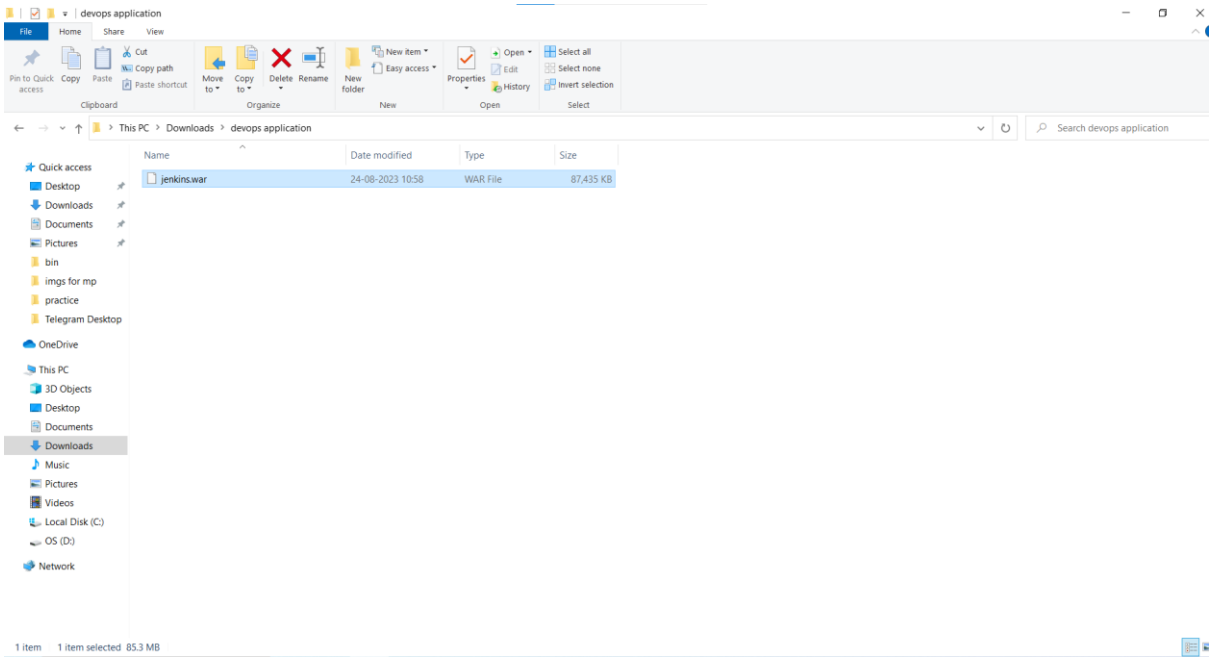
Jenkins History

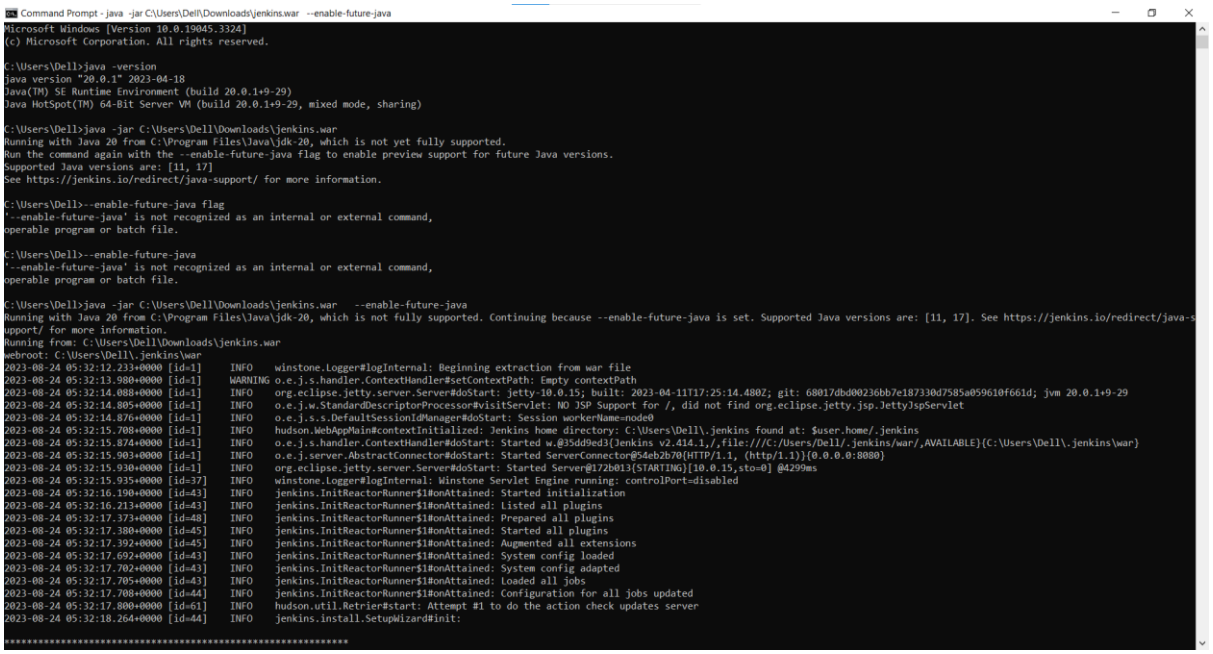
The Jenkins project was started in 2004 (originally called Hudson) by Kohsuke Kawaguchi, while he worked for Sun Microsystems. Kohsuke was a developer at Sun and got tired of incurring the wrath of his team every time his code broke the build. He created Jenkins as a way to perform continuous integration – that is, to test his code before he did an actual commit to the repository, to be sure all was well. Once his teammates saw what he was doing, they all wanted to use Jenkins. Kohsuke open sourced it, creating the Jenkins project, and soon Jenkins usage had spread around the world.

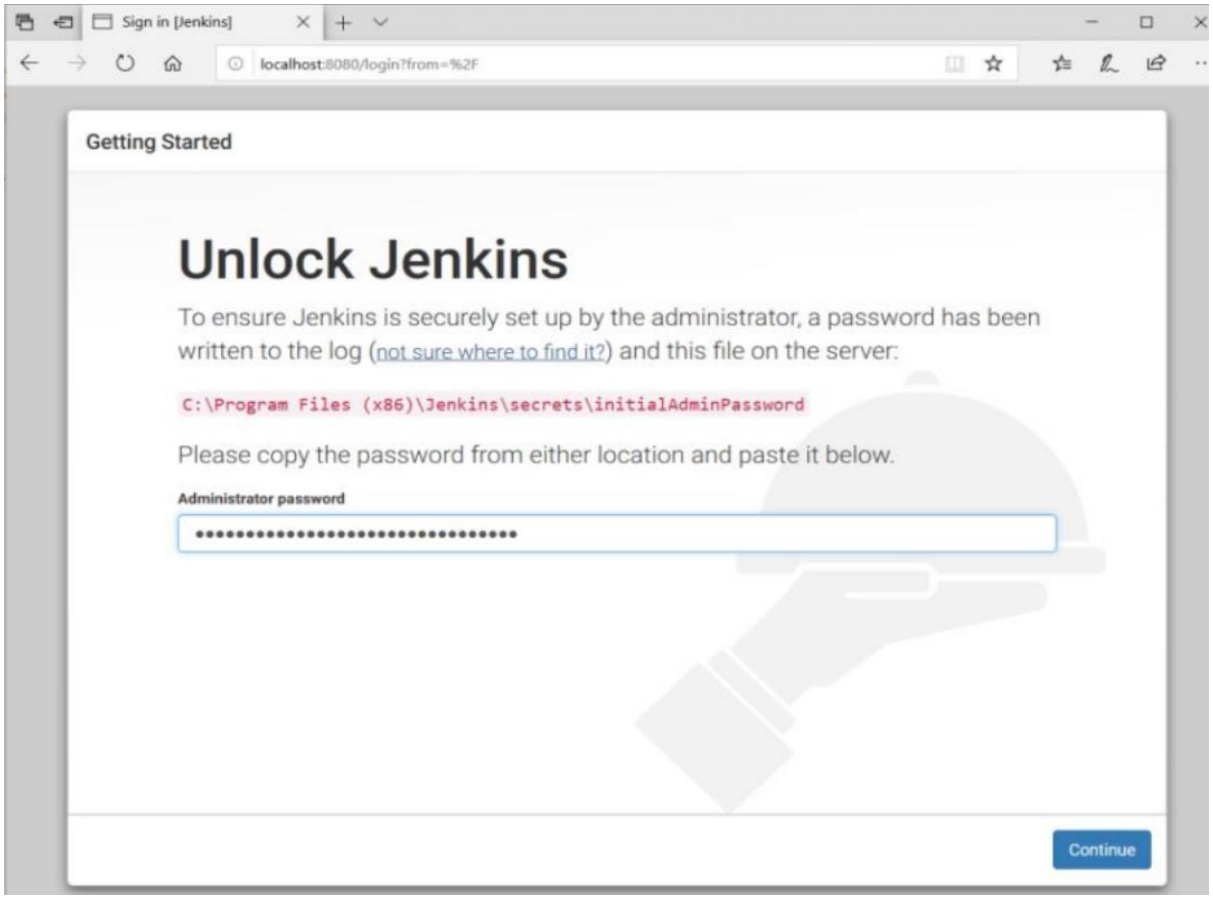
Jenkins Today

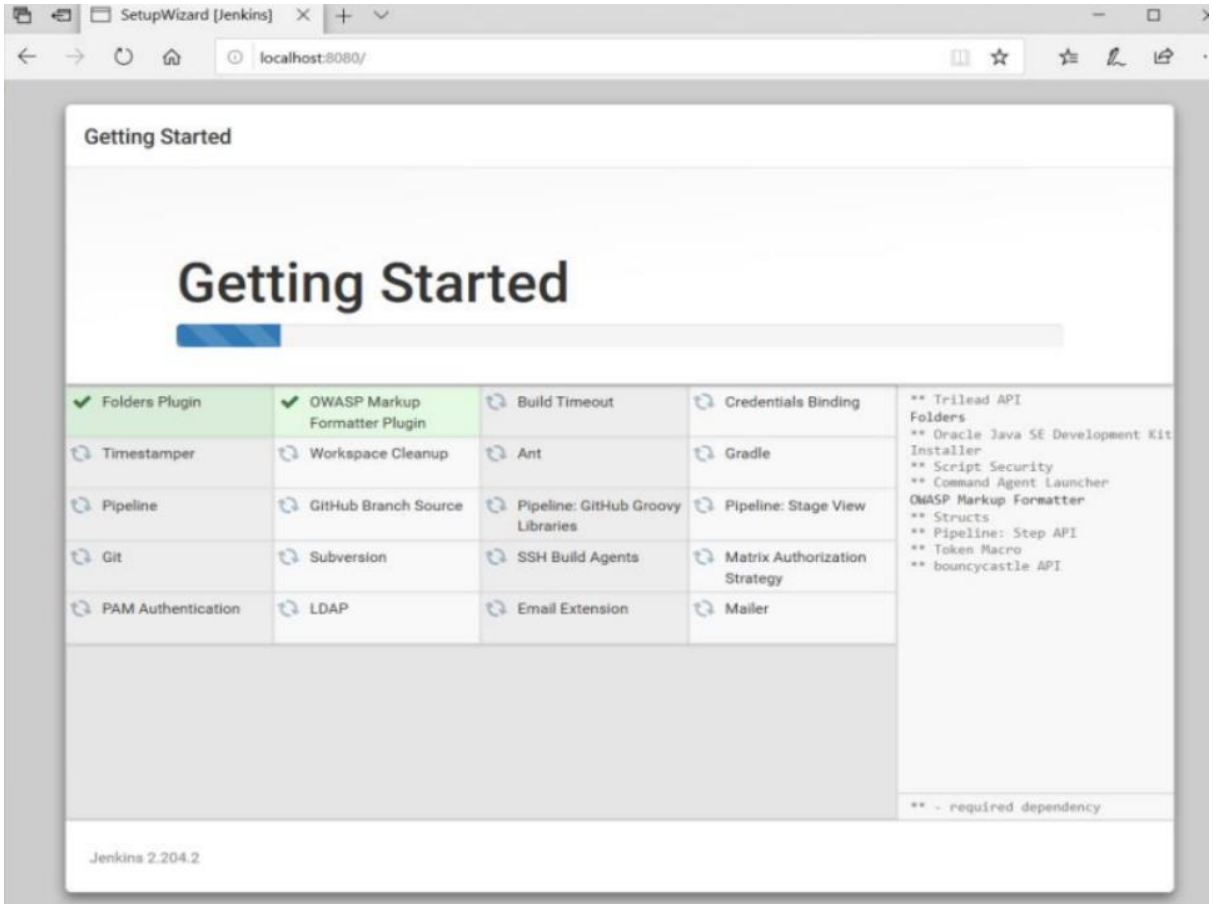
Originally developed by Kohsuke for continuous integration (CI), today Jenkins orchestrates the entire software delivery pipeline – called continuous delivery. For some organizations automation extends even further, to continuous deployment. Continuous delivery (CD), coupled with a DevOps culture, dramatically accelerates the delivery of software.
Jenkins is the most widely adopted solution for continuous delivery, thanks to its extensibility and a vibrant, active community. The Jenkins community offers more than 1,700 plugins that enable Jenkins to integrate with virtually any tool, including all of the best-of-breed solutions used throughout the continuous delivery process. Jenkins continues to grow as the dominant solution for software process automation, continuous integration and continuous delivery and, as of February 2018, there are more than 165,000 active installations and an estimated 1.65 million users around the world. Jenkins offers a simple way to set up a continuous integration or continuous delivery (CI/CD) environment for almost any combination of languages and source code repositories using pipelines, as well as automating other routine development tasks. While Jenkins doesn't eliminate the need to create scripts for individual steps, it does give you a faster and more robust way to integrate your entire chain of build, test, and deployment tools than you can easily build yourself.

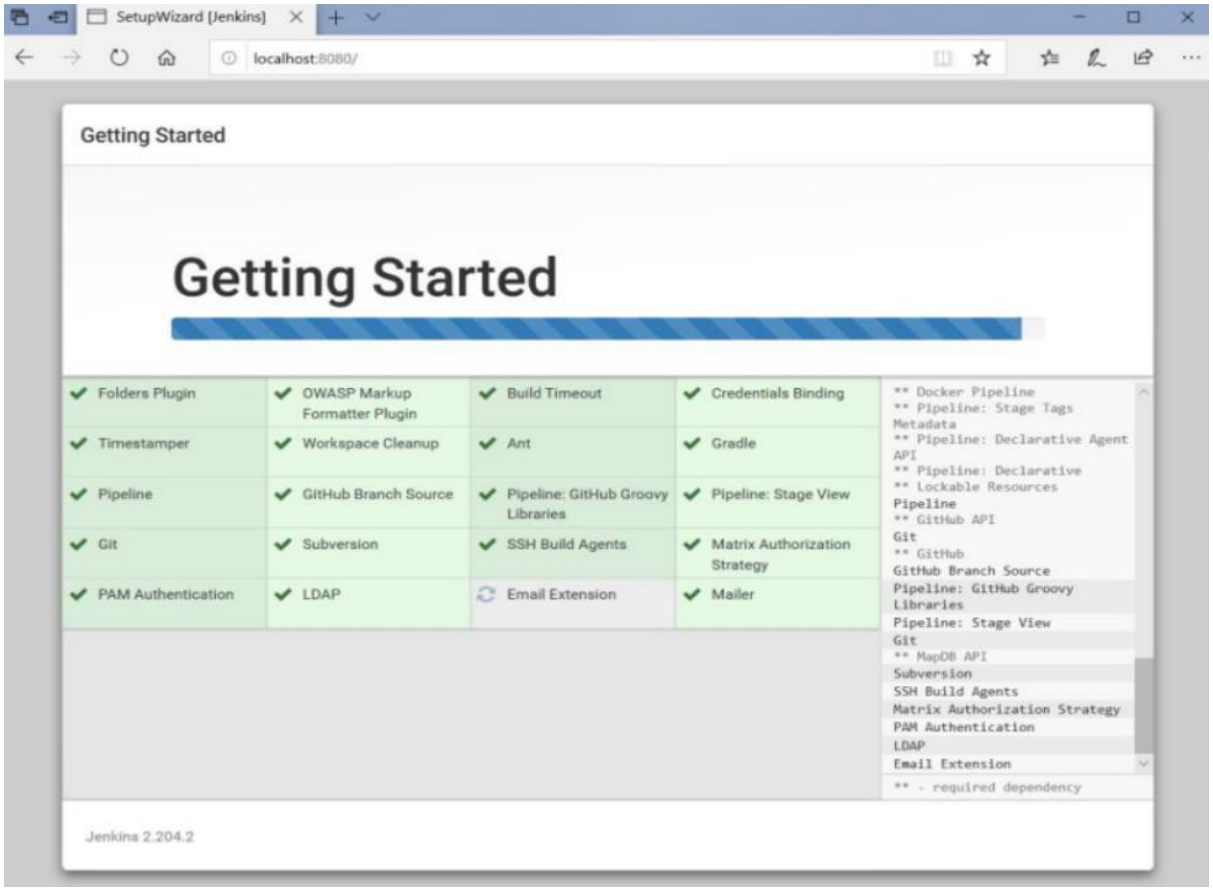
Screenshots:

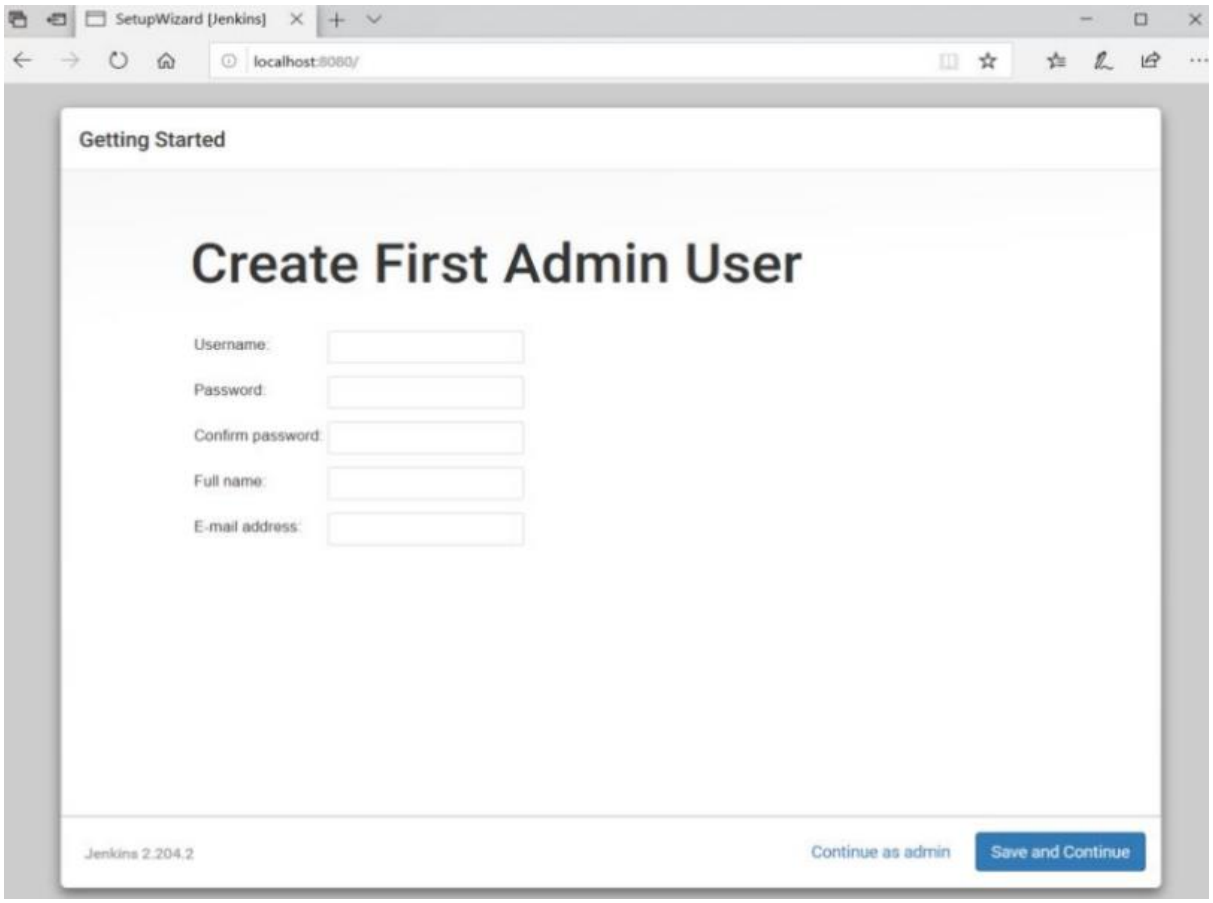


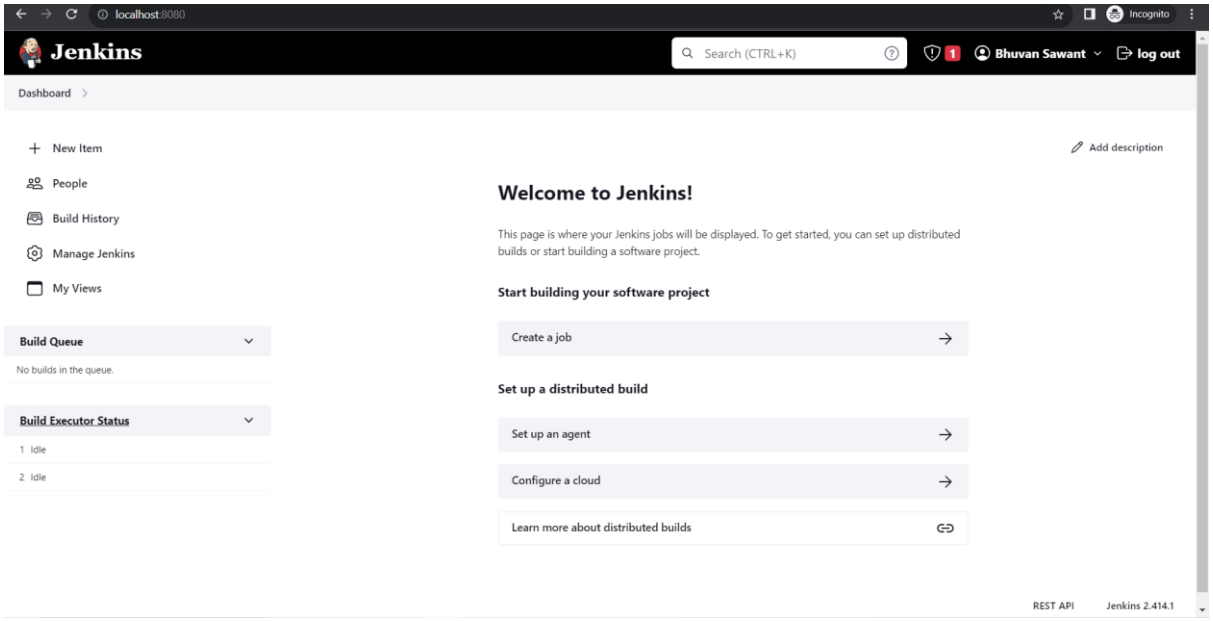



Sign in [Jenkins]
localhost:8080/login?from=%2F
Getting Started
Unlock Jenkins
To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:
C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword
Please copy the password from either location and paste it below.
Administrator password
Continue


SetupWizard [Jenkins]
localhost:8080/
Getting Started
Getting Started
Folders Plugin
OWASP Markup Formatter Plugin
Build Timeout
Credentials Binding
Timestamper
Workspace Cleanup
Ant
Gradle
Pipeline
GitHub Branch Source
Pipeline: GitHub Groovy Libraries
Pipeline: Stage View
Git
Subversion
SSH Build Agents
Matrix Authorization Strategy
PAM Authentication
LDAP
Email Extension
Mailer
** Trilead API
Folders
** Oracle Java SE Development Kit Installer
** Script Security
** Command Agent Launcher
OWASP Markup Formatter
** Structs
** Pipeline: Step API
** Token Macro
** bouncycastle API
** - required dependency
Jenkins 2.204.2


SetupWizard [Jenkins]
localhost:8080/
Getting Started
Getting Started
Folders Plugin
OWASP Markup Formatter Plugin
Build Timeout
Credentials Binding
Timestamper
Workspace Cleanup
Ant
Gradle
Pipeline
GitHub Branch Source
Pipeline: GitHub Groovy Libraries
Pipeline: Stage View
Git
Subversion
SSH Build Agents
Matrix Authorization Strategy
PAM Authentication
LDAP
Email Extension
Mailer
** Docker Pipeline
** Pipeline: Stage Tags Metadata
** Pipeline: Declarative Agent API
** Pipeline: Declarative
** Lockable Resources
Pipeline
** GitHub API
Git
** GitHub
GitHub Branch Source
Pipeline: GitHub Groovy Libraries
Pipeline: Stage View
Git
** MapDB API
Subversion
SSH Build Agents
Matrix Authorization Strategy
PAM Authentication
LDAP
Email Extension
** - required dependency
Jenkins 2.204.2


SetupWizard [Jenkins]
localhost:8080/
Getting Started
Create First Admin User
Username:
Password:
Confirm password:
Full name:
E-mail address:
Jenkins 2.204.2
Continue as admin
Save and Continue



Conclusion: In this experiment we had understood Jenkins tool and installation of Jenkins.