



Sri Eshwar
College of Engineering
Coimbatore | Tamilnadu
An Autonomous Institution
Affiliated to Anna University, Chennai



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Academic Year: 2025–2026

DEVOPS INTERNSHIP REPORT

Submitted by:

Name: Ravena S

Roll Number: 23CS122

Class: III CSE-B

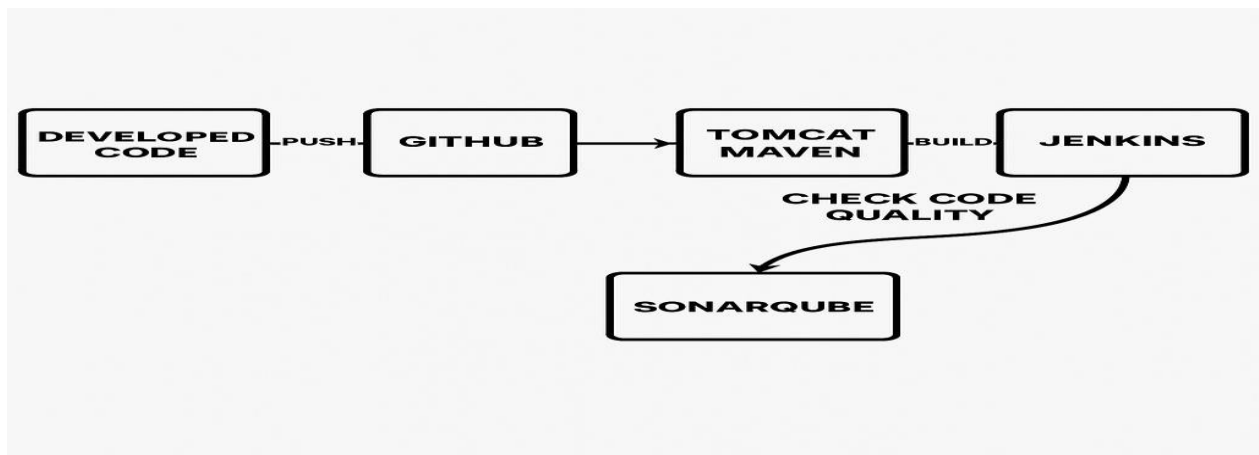
AIM:

To deploy a Java-based Maven application using a CI/CD pipeline that automates code integration, build, testing, code quality analysis, and deployment, ensuring faster and more reliable delivery.

TOOLS REQUIRED:

VS Code, Git, GitHub, Maven, Apache Tomcat, Jenkins, SonarQube, AWS (EC2)

FLOWCHART:



PROCEDURE:

Step 1:

Develop a Maven Project and run it locally

The screenshot shows the Visual Studio Code interface with a Maven project named 'todolist'. The Explorer view on the left shows the project structure, including 'src/main/java/com/example/todolist'. The main editor displays the 'TodolistApplication.java' file, which is a Spring Boot application. The terminal at the bottom shows the output of the 'mvnw install' command, indicating a successful build and installation of the application.

```
dependencies in BOOT-INF/.
[INFO] The original artifact has been renamed to C:\Users\Ravena S\Downloads\todolist\todolist\target\todolist-0.0.1-SNAPSHOT.war.original
[INFO] --- install:3.1.4:install (default-install) @ todolist ---
[INFO] Installing C:\Users\Ravena S\Downloads\todolist\todolist\pom.xml to C:\Users\Ravena S\.m2\repository\com\example\todolist\0.0.1-SNAPSHOT\todolist-0.0.1-SNAPSHOT.pom
[INFO] Installing C:\Users\Ravena S\Downloads\todolist\todolist\target\todolist-0.0.1-SNAPSHOT.war to C:\Users\Ravena S\.m2\repository\com\example\todolist\0.0.1-SNAPSHOT\todolist-0.0.1-SNAPSHOT.war
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 51.705 s
[INFO] Finished at: 2025-07-17T09:45:33+05:30
[INFO] -----
```

Step 2:

Push the Project to a git repository

The screenshot shows the GitHub 'Create a new repository' page. The browser address bar shows 'github.com/new'. The page title is 'New repository'. Below the title, there is a search bar and a 'Type' dropdown. The main content area is titled 'Create a new repository' with a link to 'Try the new experience'. A sub-header states: 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)'. A note says: 'Required fields are marked with an asterisk (*)'. The 'Owner' field is set to 'ravena04'. The 'Repository name' field is 'devops-project', with a green checkmark indicating it is available. Below this, a message says: 'Great repository names are short and memorable. Need inspiration? How about [silver-telegram](#) ?'. The 'Description (optional)' field is empty. There are two radio buttons for visibility: 'Public' (selected) and 'Private'. Below these is the 'Initialize this repository with:' section, with an option to 'Add a README file'. At the bottom, there is an 'Add .gitignore' section with a dropdown set to 'None'.

Create a new repository [Try the new experience](#)

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * ravena04 / Repository name * devops-project
devops-project is available.

Great repository names are short and memorable. Need inspiration? How about [silver-telegram](#) ?

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore
.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

The screenshot shows the GitHub repository page for 'devops-project' by user 'ravena04'. The browser address bar shows 'github.com/ravena04/devops-project'. The page has a navigation bar with links for 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. Below the navigation bar, there are buttons for 'Pin', 'Watch', 'Fork', and 'Star'. The main content area shows the repository structure with a file list: '.mvn/wrapper' (first commit), 'src' (Final), '.gitattributes' (first commit), '.gitignore' (first commit), 'mvnw' (first commit), 'mvnw.cmd' (first commit), and 'pom.xml' (Final, 2 days ago). Below the file list is a 'README' section. On the right side, there is an 'About' section with a description, 'Activity' (0 stars, 0 watching, 0 forks), 'Releases' (No releases published, [Create a new release](#)), 'Packages' (No packages published, [Publish your first package](#)), and a 'Languages' section showing a bar chart for Java (39.8%), CSS (34.4%), and HTML (25.8%).

devops-project Public

main 1 Branch 0 Tags

Go to file Add file <> Code

ravena04 Final

File	Commit
.mvn/wrapper	first commit
src	Final
.gitattributes	first commit
.gitignore	first commit
mvnw	first commit
mvnw.cmd	first commit
pom.xml	Final

2 days ago

README

About

No description, website, or topics provided.

Activity

0 stars

0 watching

0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

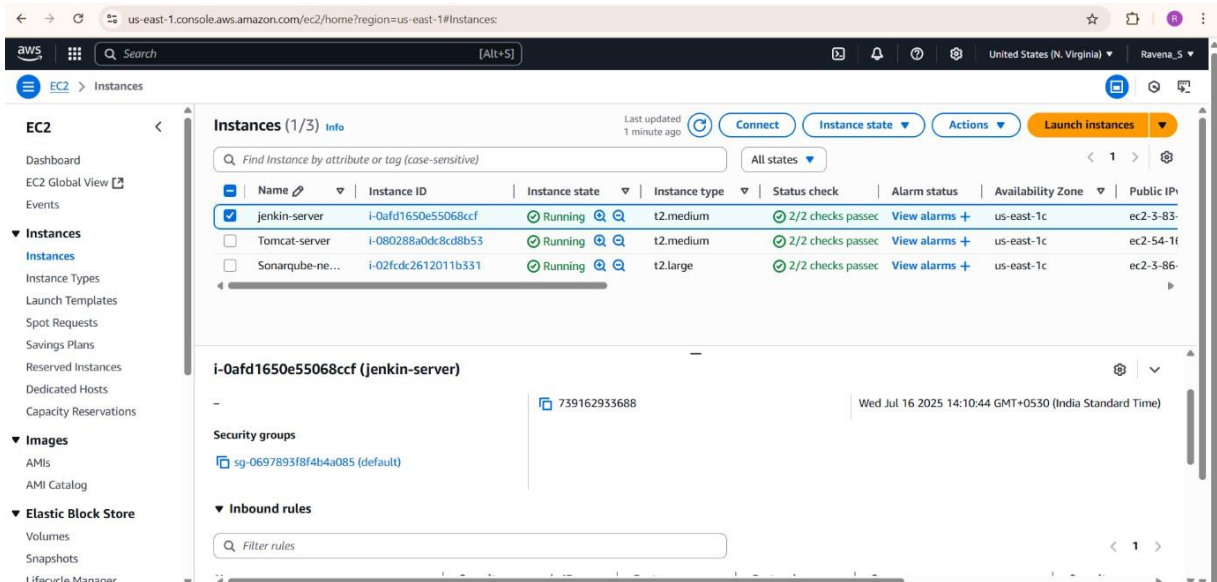
Languages

Java 39.8% CSS 34.4% HTML 25.8%

Step 3:

Open AWS Console-> Navigate to EC2->Create 3 Instances:

- Jenkins server(t2.medium)
- Tomcat-maven server(t2.medium)
- Sonarqube server(t2.large)



In Tomcat-Maven Server:

Step 4:

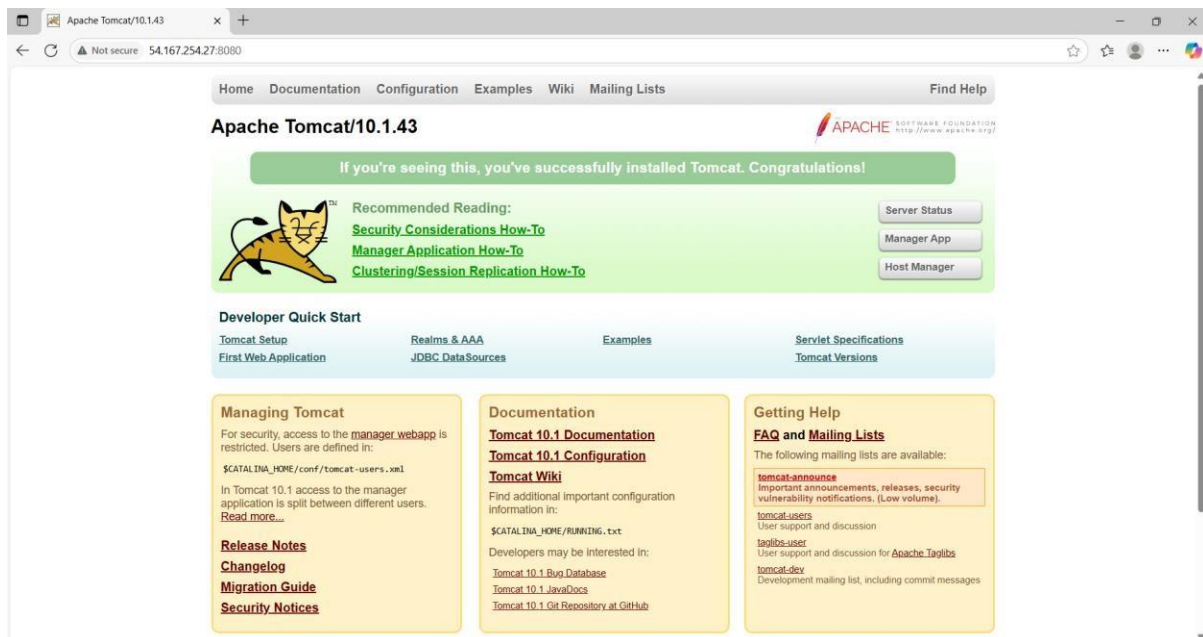
Install Maven and clone the Maven project from the repository and run the project

```
root@ip-172-31-87-63: /usr/share/maven/devops-project#
715 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/sisu/sisu-inject-bean/1.4.2/sisu-inject-bean-1.4.2.jar (153 kB at 543
kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-component-annotations/1.5.5/plexus-component-annotatio
ns-1.5.5.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-classworlds/2.2.3/plexus-classworlds-2.2.3.jar (46 kB a
t 162 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/sonatype/plexus/plexus-cipher/1.4/plexus-cipher-1.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/plexus/plexus-sec-dispatcher/1.3/plexus-sec-dispatcher-1.3.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-component-annotations/1.5.5/plexus-component-annotation
s-1.5.5.jar (4.2 kB at 15 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/plexus/plexus-cipher/1.4/plexus-cipher-1.4.jar (13 kB at 46 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/plexus/plexus-sec-dispatcher/1.3/plexus-sec-dispatcher-1.3.jar (29 kB
at 94 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/sisu/sisu-guice/2.1.7/sisu-guice-2.1.7-noaop.jar (472 kB at 1.5 MB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/github/luben/zstd-jni/1.5.5-2/zstd-jni-1.5.5-2.jar (5.9 MB at 17 MB/s)
[INFO] Packaging webapp
[INFO] Assembling webapp [todolist] in [/usr/share/maven/devops-project/target/todolist-0.0.1-SNAPSHOT]
[INFO] Processing war project
[INFO] Building war: /usr/share/maven/devops-project/target/todolist-0.0.1-SNAPSHOT.war
[INFO] --- spring-boot-maven-plugin:3.5.3:repackage (repackage) @ todolist ---
[INFO] Replacing main artifact /usr/share/maven/devops-project/target/todolist-0.0.1-SNAPSHOT.war with repackaged archive, adding nested depende
ncies in BOOT-INF/.
[INFO] The original artifact has been renamed to /usr/share/maven/devops-project/target/todolist-0.0.1-SNAPSHOT.war.original
[INFO] --- maven-install-plugin:3.1.4:install (default-install) @ todolist ---
[INFO] Installing /usr/share/maven/devops-project/pom.xml to /root/.m2/repository/com/example/todolist/0.0.1-SNAPSHOT/todolist-0.0.1-SNAPSHOT.po
m
[INFO] Installing /usr/share/maven/devops-project/target/todolist-0.0.1-SNAPSHOT.war to /root/.m2/repository/com/example/todolist/0.0.1-SNAPSHOT
/todolist-0.0.1-SNAPSHOT.war
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 9.112 s
[INFO] Finished at: 2025-07-16T10:26:52Z
[INFO] -----
root@ip-172-31-87-63: /usr/share/maven/devops-project#
```

Step 5:

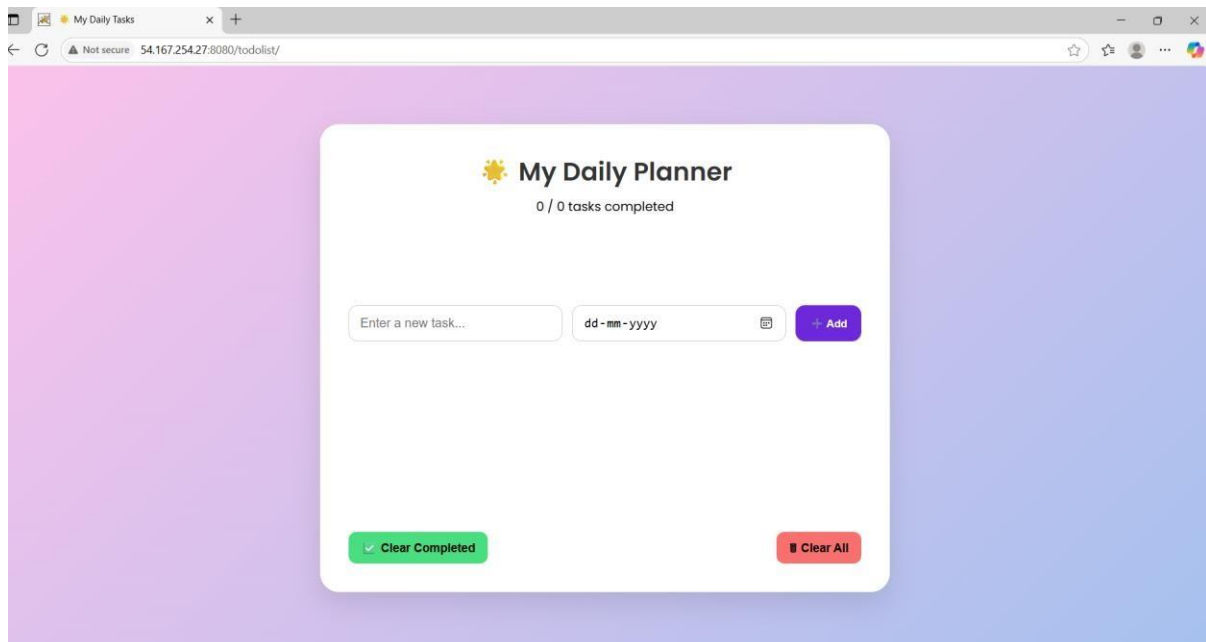
Install Tomcat and start the server

```
root@ip-172-31-87-63: ~/apache
42 cd devops-project
43 ls
44 mvn clean install
45 ls
46 cd target/
47 ls
48 cd /usr/share/maven/devops-project/target
49 ls
50 cd
51 apt update
52 wget https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.43/bin/apache-tomcat-10.1.43.tar.gz
53 ls
54 tar -zxvf apache-tomcat-10.1.43.tar.gz
55 apt update
56 cd apache-tomcat-10.1.43/
57 tomcat --version
58 cd
59 tomcat --version
60 history
root@ip-172-31-87-63:~# cd apache-tomcat-10.1.43/
root@ip-172-31-87-63:~/apache-tomcat-10.1.43# cd bin
ls
bootstrap.jar      ciphers.sh         daemon.sh           migrate.bat         shutdown.sh         tool-wrapper.bat
catalina-tasks.xml commons-daemon-native.tar.gz digest.bat          migrate.sh          startup.bat         tool-wrapper.sh
catalina.bat       commons-daemon.jar digest.sh            setclasspath.bat   startup.sh          version.bat
catalina.sh        configtest.bat     makebase.bat        setclasspath.sh    tomcat-juli.jar    version.sh
ciphers.bat        configtest.sh      shutdown.bat        tomcat-native.tar.gz
root@ip-172-31-87-63:~/apache-tomcat-10.1.43/bin# ./ startup.sh
-bash: ./: Is a directory
root@ip-172-31-87-63:~/apache-tomcat-10.1.43/bin# ./startup.sh
Using CATALINA_BASE:   /root/apache-tomcat-10.1.43
Using CATALINA_HOME:   /root/apache-tomcat-10.1.43
Using CATALINA_TMPDIR: /root/apache-tomcat-10.1.43/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /root/apache-tomcat-10.1.43/bin/bootstrap.jar:/root/apache-tomcat-10.1.43/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
root@ip-172-31-87-63:~/apache-tomcat-10.1.43/bin#
```



Step 6:

Copy the path of war file of the project from maven to tomcat(webapps) and restart the tomcat server so that the project application is deployed using Tomcat



In Jenkins Server:

Step 7:

Install Jenkins and check the status

```
root@jenkin: ~
Unpacking jenkins (2.504.3) ...
Setting up net-tools (2.10-0.1ubuntu4.4) ...
Setting up jenkins (2.504.3) ...
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

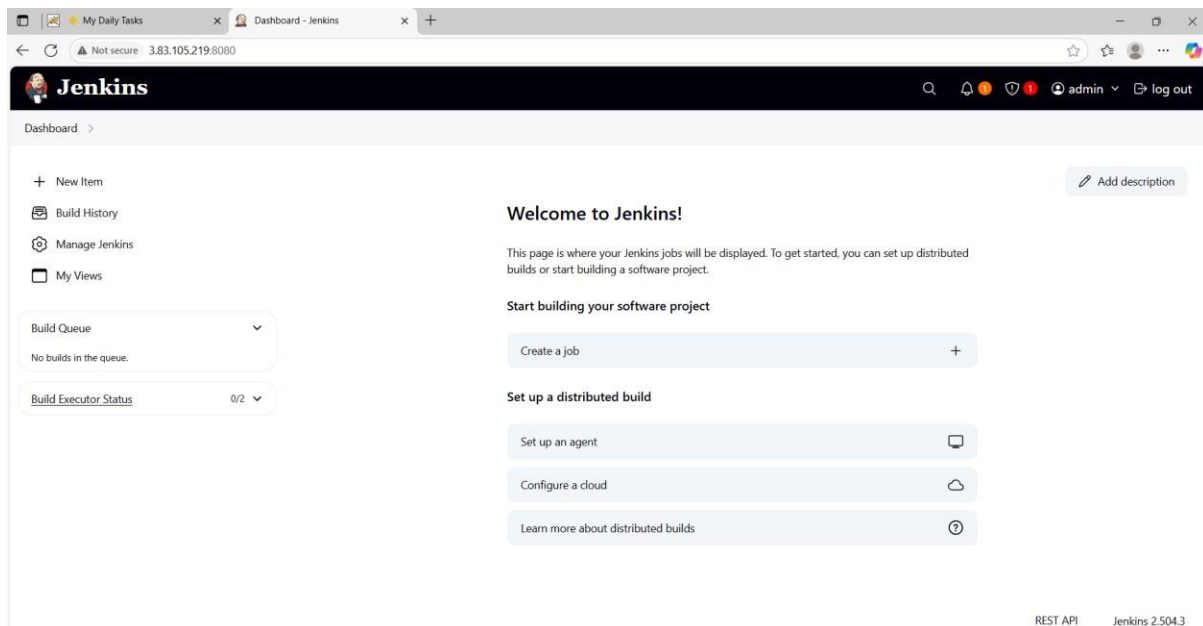
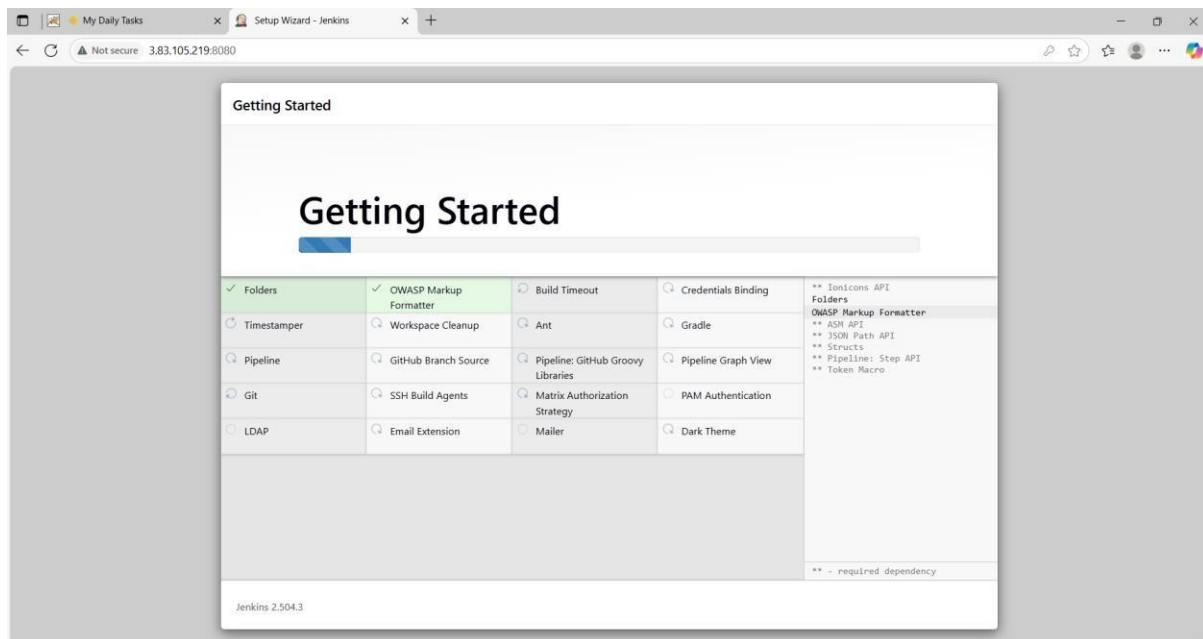
No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

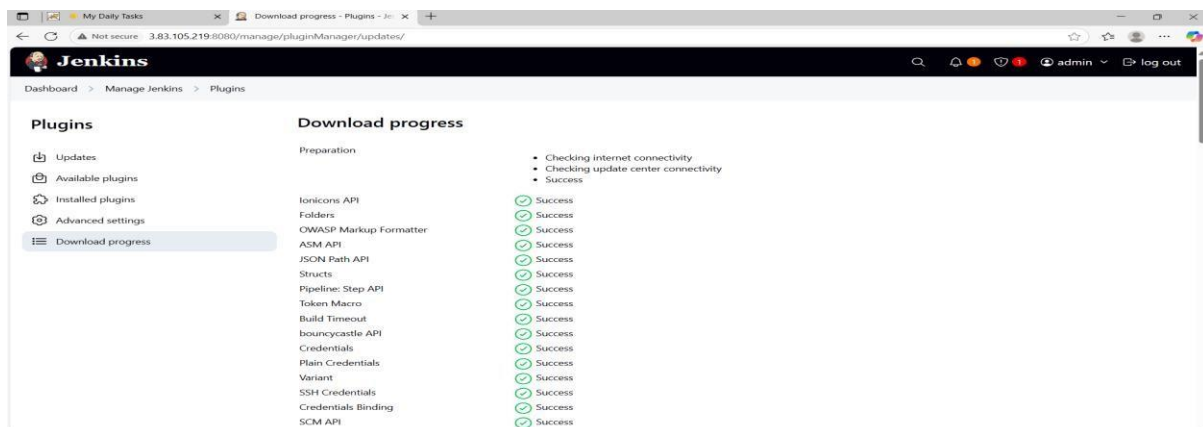
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@jenkin:~# jenkins --version
jenkins: command not found
root@jenkin:~# systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-07-16 13:40:25 UTC; 1min 16s ago
     Main PID: 5379 (java)
       Tasks: 45 (limit: 4670)
      Memory: 675.9M (peak: 677.1M)
         CPU: 17.586s
    CGroup: /system.slice/jenkins.service
            └─5379 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

Jul 16 13:40:21 jenkins jenkins[5379]: b77ff90d90a64ccab705569235aac08f
Jul 16 13:40:21 jenkins jenkins[5379]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Jul 16 13:40:21 jenkins jenkins[5379]: *****
Jul 16 13:40:21 jenkins jenkins[5379]: *****
Jul 16 13:40:21 jenkins jenkins[5379]: *****
Jul 16 13:40:25 jenkins jenkins[5379]: 2025-07-16 13:40:25.505+0000 [id=31] INFO jenkins.InitReactorRunner$1onAttained: Completed initializat
Jul 16 13:40:25 jenkins jenkins[5379]: 2025-07-16 13:40:25.534+0000 [id=23] INFO hudson.lifecycle.Lifecycle$onReady: Jenkins is fully up and r
Jul 16 13:40:25 jenkins systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
Jul 16 13:40:25 jenkins jenkins[5379]: 2025-07-16 13:40:25.788+0000 [id=49] INFO h.m.DownloadService$Downloadable$Load: Obtained the updated d
Jul 16 13:40:25 jenkins jenkins[5379]: 2025-07-16 13:40:25.789+0000 [id=49] INFO hudson.util.Retrier$start: Performed the action check updates
lines 1-20/20 (END)
```

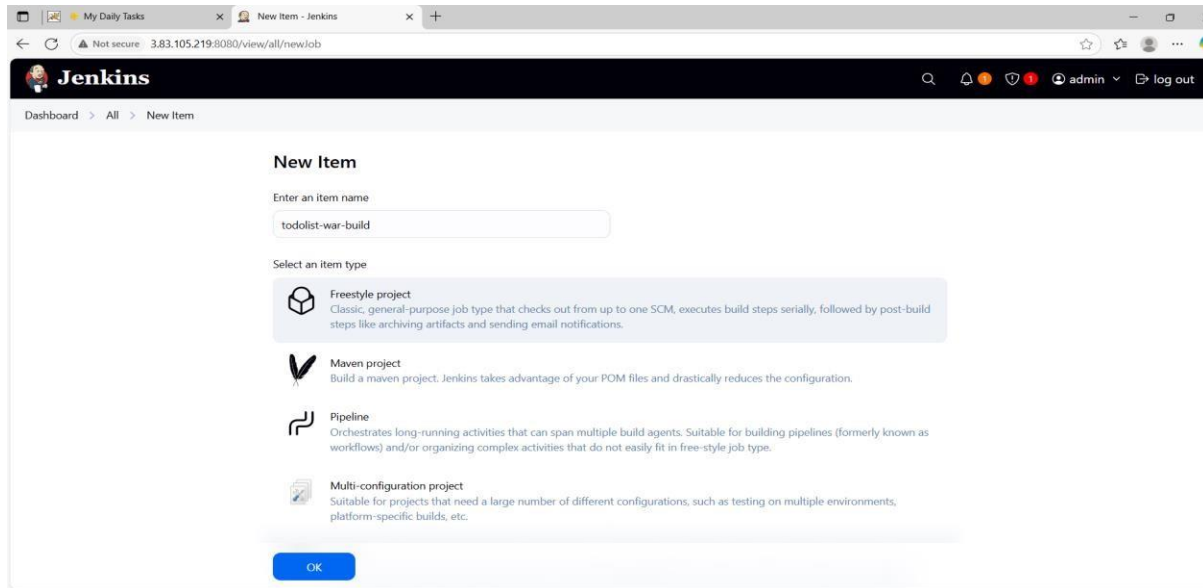
Step 8:

Install Plugins like Maven, Git and Pipeline and add JDK and Maven Credentials



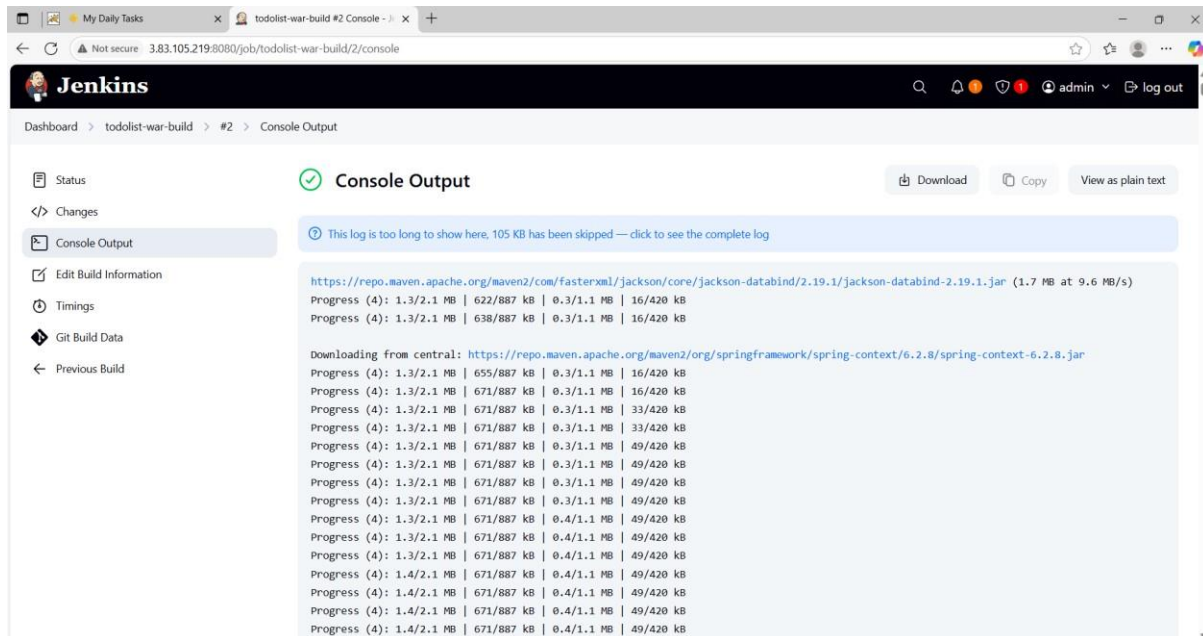
Step 9:

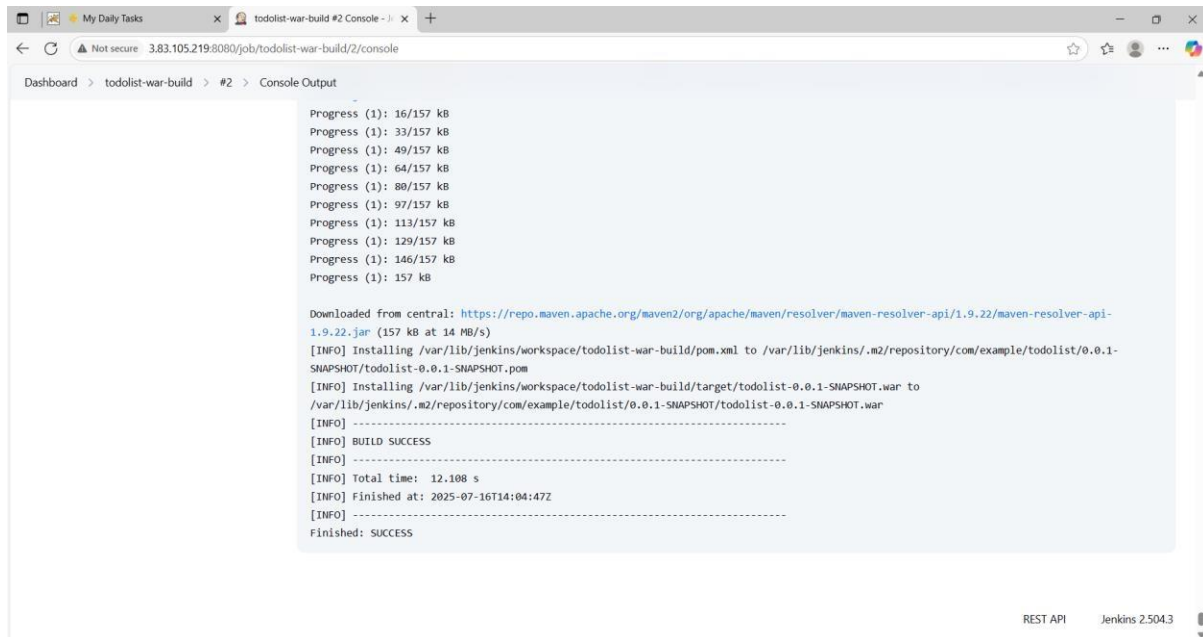
Create a new job and select freestyle project



Step 10:

Include the git hub repository link and build the project

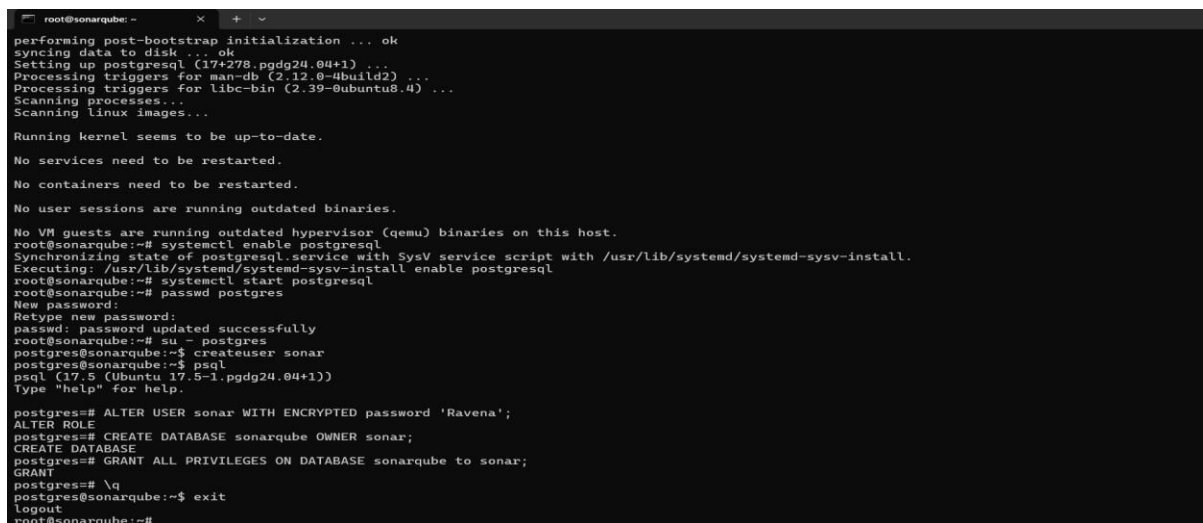




In SonarQube Server:

Step 11:

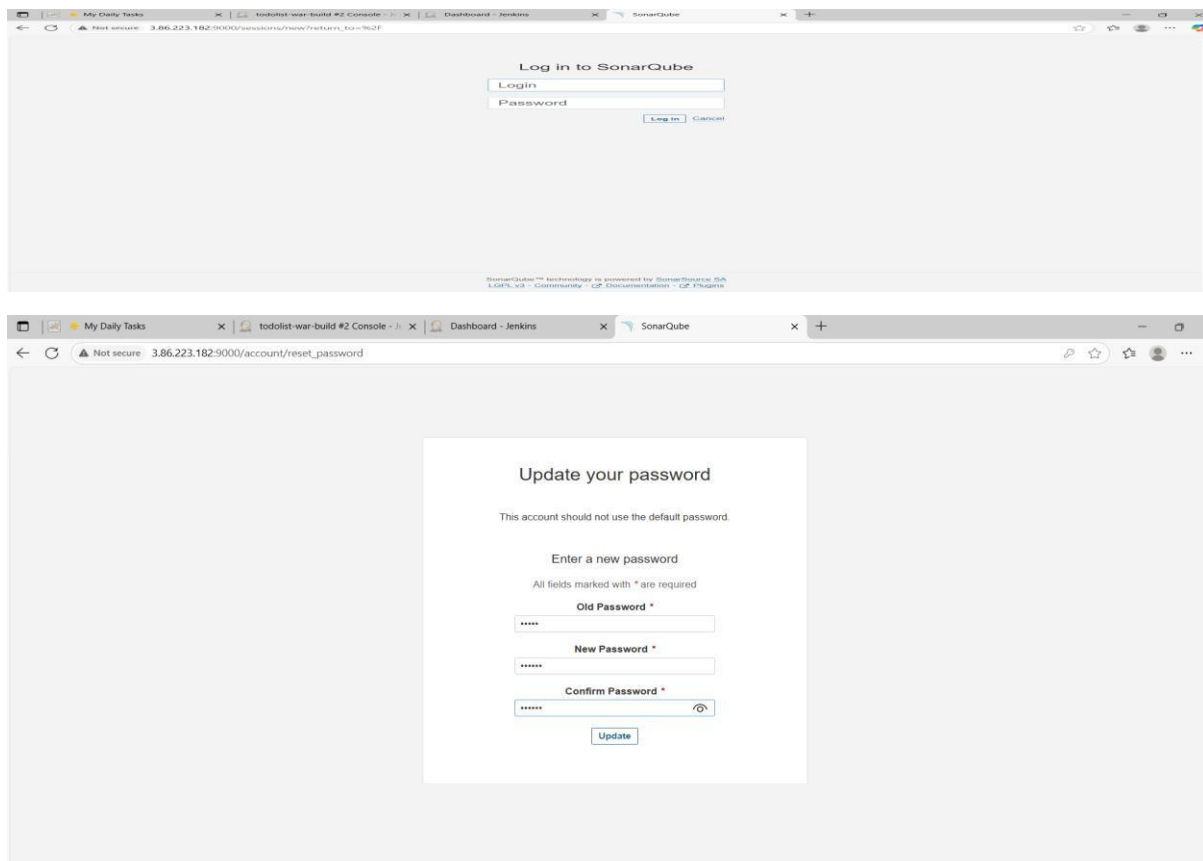
Install SonarQube, Create a database and user credentials and check the status



```
root@sonarqube: /opt
inflating: sonarqube-9.9.3.79811/lib/sonar-shutdowner-9.9.3.79811.jar
creating: sonarqube-9.9.3.79811/elasticsearch/plugins/
root@sonarqube:~# ls
snap sonarqube-9.9.3.79811 sonarqube-9.9.3.79811.zip
root@sonarqube:~# mv sonarqube-9.9.3.79811 sonarqube
root@sonarqube:~# ls
snap sonarqube sonarqube-9.9.3.79811.zip
root@sonarqube:~# mv sonarqube /opt
root@sonarqube:~# cd /opt/
root@sonarqube:~# ls
sonarqube
root@sonarqube:/opt# sudo groupadd sonar
root@sonarqube:/opt# sudo useradd -d /opt/sonarqube -g sonar sonar
root@sonarqube:/opt# sudo chown sonar:sonar /opt/sonarqube -R
root@sonarqube:/opt# sudo vi /opt/sonarqube/conf/sonar.properties
root@sonarqube:/opt# 427L, 21215B written
root@sonarqube:/opt# vi /opt/sonarqube/bin/linux-x86-64/sonar.sh
317L, 7156B written
root@sonarqube:/opt# sudo vi /etc/systemd/system/sonar.service
root@sonarqube:/opt# sudo systemctl enable sonar
Created symlink /etc/systemd/system/multi-user.target.wants/sonar.service → /etc/systemd/system/sonar.service.
root@sonarqube:/opt# sudo systemctl start sonar
root@sonarqube:/opt# sudo systemctl status sonar
● sonar.service - SonarQube service
   Loaded: loaded (/etc/systemd/system/sonar.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-07-16 15:39:11 UTC; 8s ago
     Process: 8098 ExecStart=/opt/sonarqube/bin/linux-x86-64/sonar.sh start (code=exited, status=0/SUCCESS)
    Main PID: 8121 (java)
      Tasks: 62 (limit: 9501)
     Memory: 768.3M (peak: 768.5M)
        CPU: 16.302s
    CGroup: /system.slice/sonar.service
            └─8121 java -Xms8m -Xmx32m --add-exports=java.base/jdk.internal.ref=ALL-UNNAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.b
              └─8147 /usr/lib/jvm/java-17-openjdk-amd64/bin/java -XX:+UseG1GC -Djava.io.tmpdir=/opt/sonarqube/temp -XX:ErrorFile=/opt/sonarqube/logs/es_hs_e
Jul 16 15:39:11 sonarqube systemd[1]: Starting sonar.service - SonarQube service...
Jul 16 15:39:11 sonarqube sonar.sh[8098]: /usr/bin/java
Jul 16 15:39:11 sonarqube sonar.sh[8098]: Starting SonarQube...
Jul 16 15:39:11 sonarqube sonar.sh[8098]: Started SonarQube.
Jul 16 15:39:11 sonarqube systemd[1]: Started sonar.service - SonarQube service.
lines 1-17/17 (END)
```

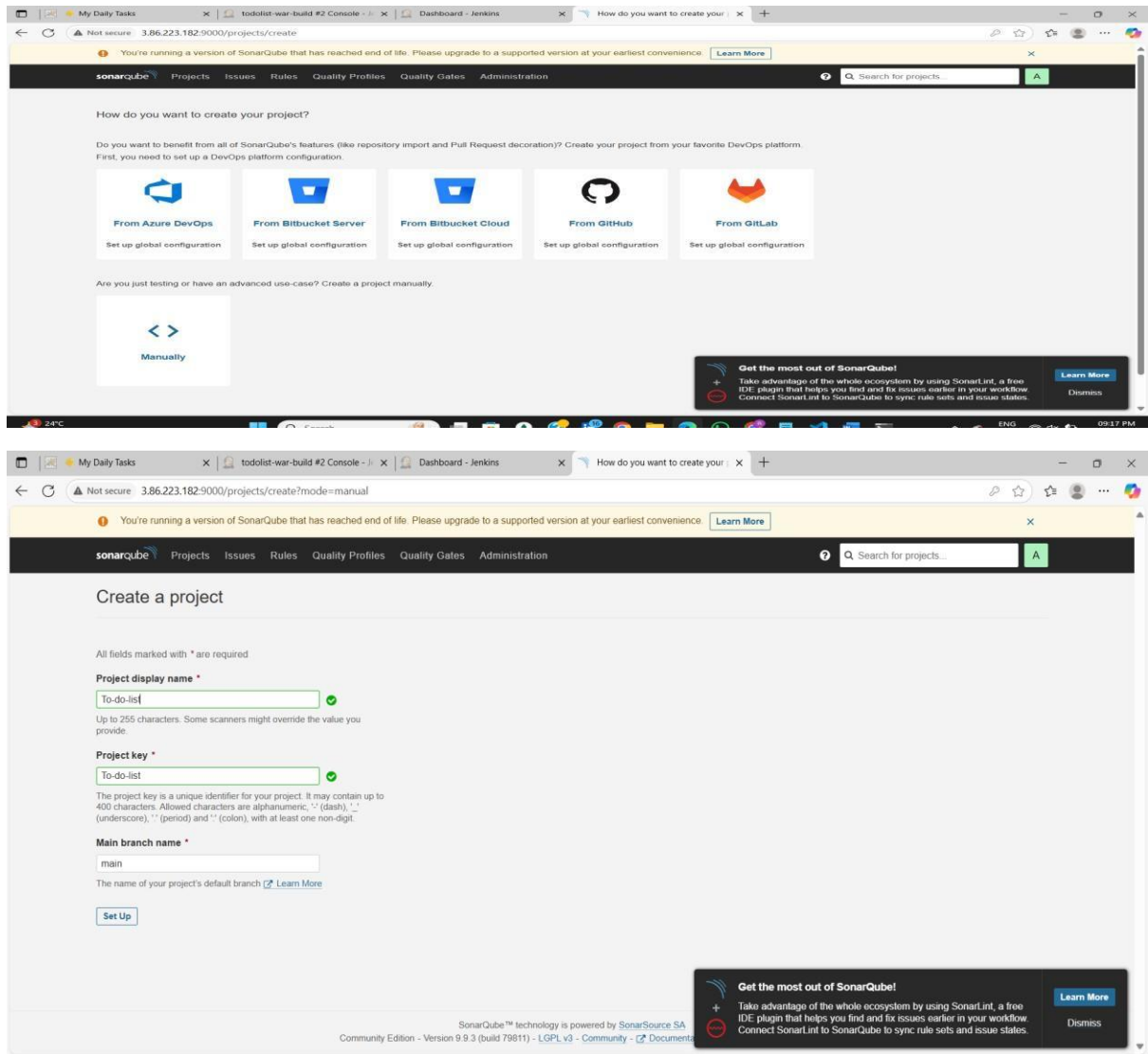
Step 12:

Ping public ip:9000 in the browser and login to the official sonarqube website using the default username and password(admin) and update the password



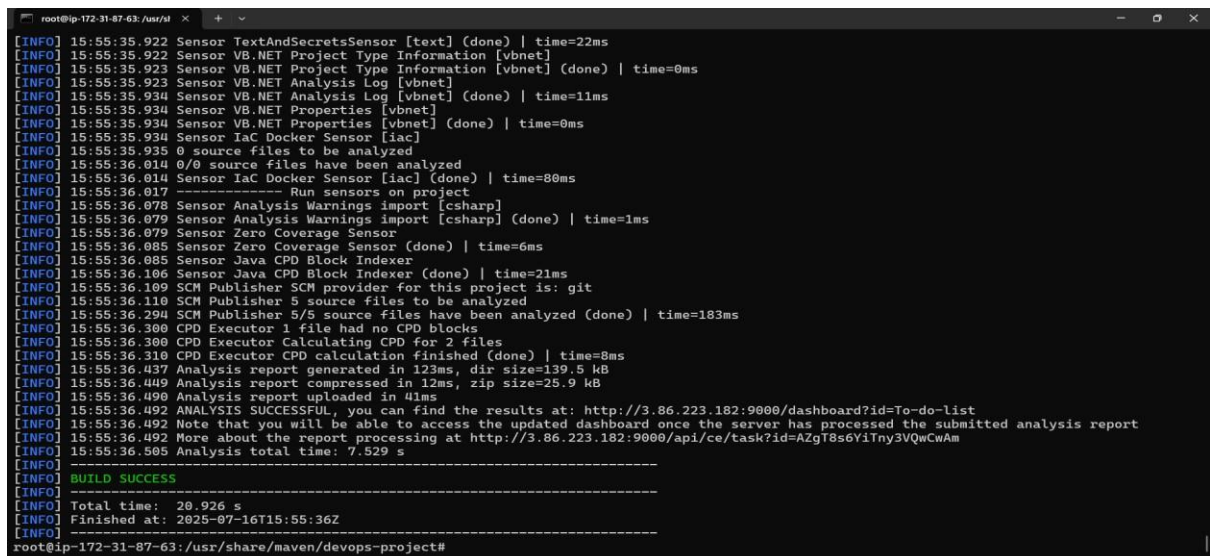
Step 13:

Select the Manual project creation mode and Create a project and select the project type as maven and copy the token generated .



Step 14:

Run the generated token in the **tomcat-maven** server



Step 15:

After Build Success open the sonarqube website to check the code quality

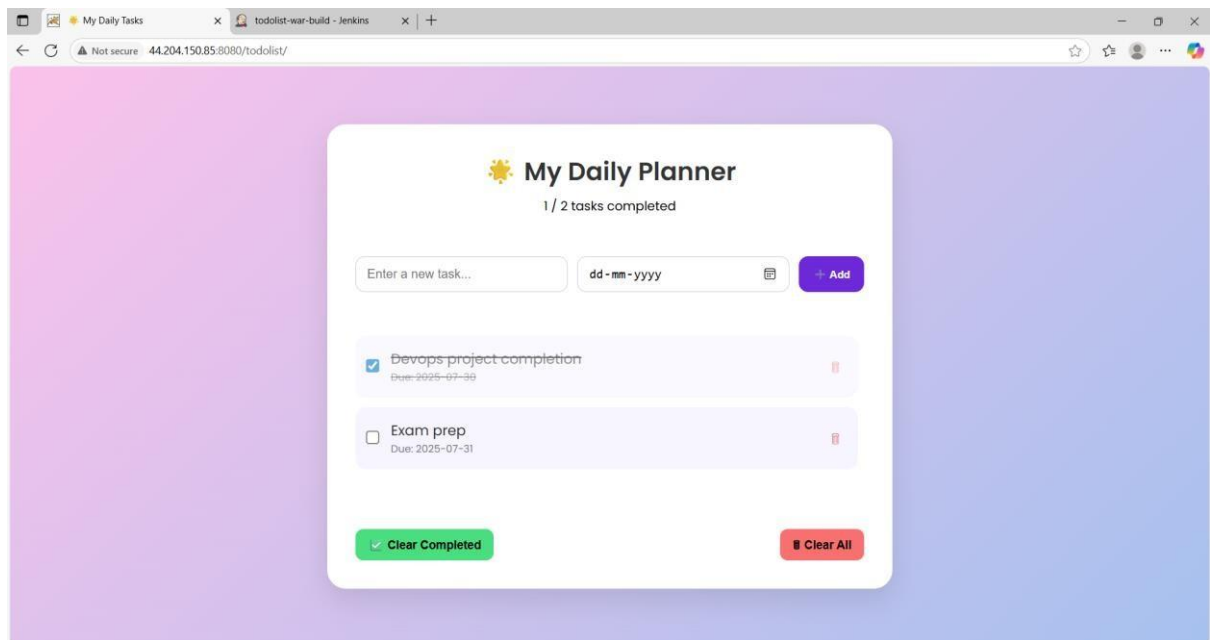
The first screenshot shows the SonarQube dashboard with the following details:

- Quality Gate Status:** Passed (All conditions passed).
- Measures:**
 - New Code:**
 - Bugs: 0
 - Vulnerabilities: 0
 - Security Hotspots: 0
 - Overall Code:**
 - Reliability: A
 - Security: A
 - Security Review: A
 - Debt: 22min
 - Code Smells: 2
 - Maintainability: A

The second screenshot shows the SonarQube dashboard with the following details:

- Measures:**
 - Code:**
 - Unit Tests: 1
 - Code Quality:**
 - Coverage: 0.0% (Coverage on 32 Lines to cover)
 - Duplications: 0.0% (Duplications on 139 Lines)
 - Duplicated Blocks: 0
- ACTIVITY:**
 - Choose graph type: Issues
 - There isn't enough data to generate an activity graph.
 - July 16, 2025 at 9:25 PM
 - 0.0.1-SNAPSHOT

OUTPUT:



RESULT:

The maven project is deployed successfully in the CI/CD pipeline