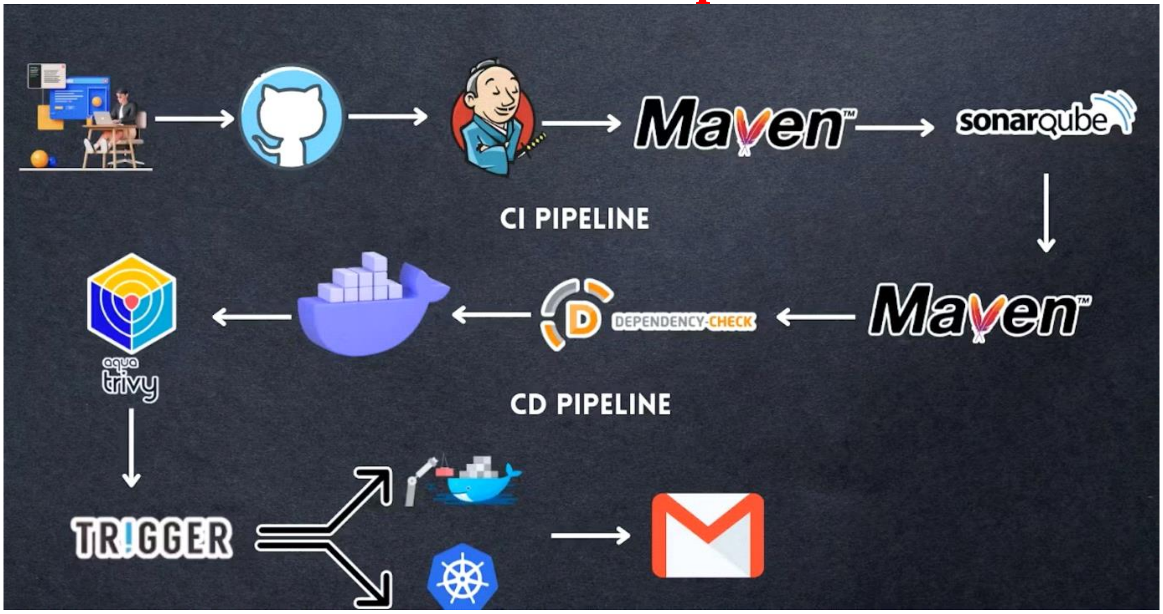
DEVSECOPS Project : Complete CI-CD (3 tier app)- Pet shop



we will be deploying a Pet shop Java Based Application. This is an everyday use case scenario used by several organizations. We will be using Jenkins as a CICD tool and deploying our application on a Docker container. Hope this detailed blog is useful.

.

Project Repo: https://github.com/Aj7Ay/jpetstore-6.git

Steps:-

Step 1 — Create an Ubuntu(22.04) T2 Large Instance

Step 2 — Install Jenkins, Docker and Trivy. Create a SonarQube Container using Docker.

Step 3 — Install Plugins like JDK, SonarQube Scanner, Maven, and OWASP Dependency Check.

Step 4 — Create a Pipeline Project in Jenkins using a Declarative Pipeline

Step 5 — Install OWASP Dependency Check Plugins

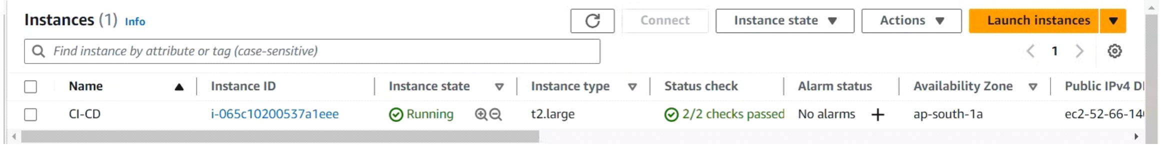
Step 6 — Docker Image Build and Push

Step 7 — Deploy the image using Docker

Step 9 — Access the Real-World Application

Step 10 — Terminate the AWS EC2 Instances.

Create an Ubuntu (22.04) T2 Large Instance



□	Name ▲	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DI
□	CI-CD	i-065c10200537a1eee	✓ Running	t2.large	✓ 2/2 checks passed	No alarms +	ap-south-1a	ec2-52-66-14

Install Jenkins, Docker and Trivy

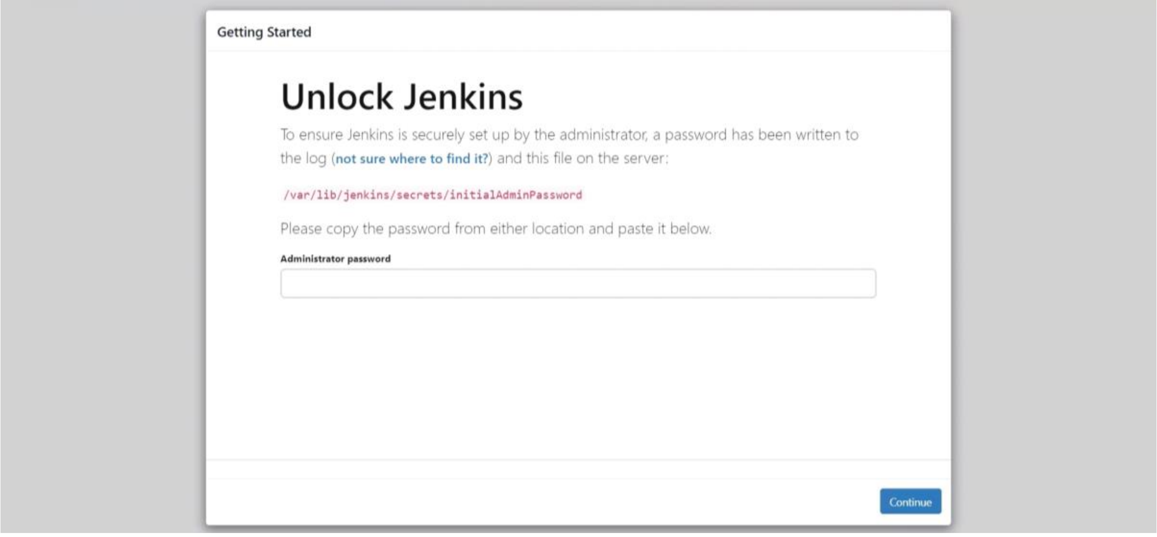
To Install Jenkins

Connect to your console, and enter these commands to Install Jenkins

```
apt update -y
apt install default-jdk
apt install maven
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \https://pkg.jenkins.io/debian-stable binary/ | sudo tee \ /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt install Jenkins -y
```

<EC2 Public IP Address:8080>

sudo cat /var/lib/jenkins/secrets/initialAdminPassword



Getting Started

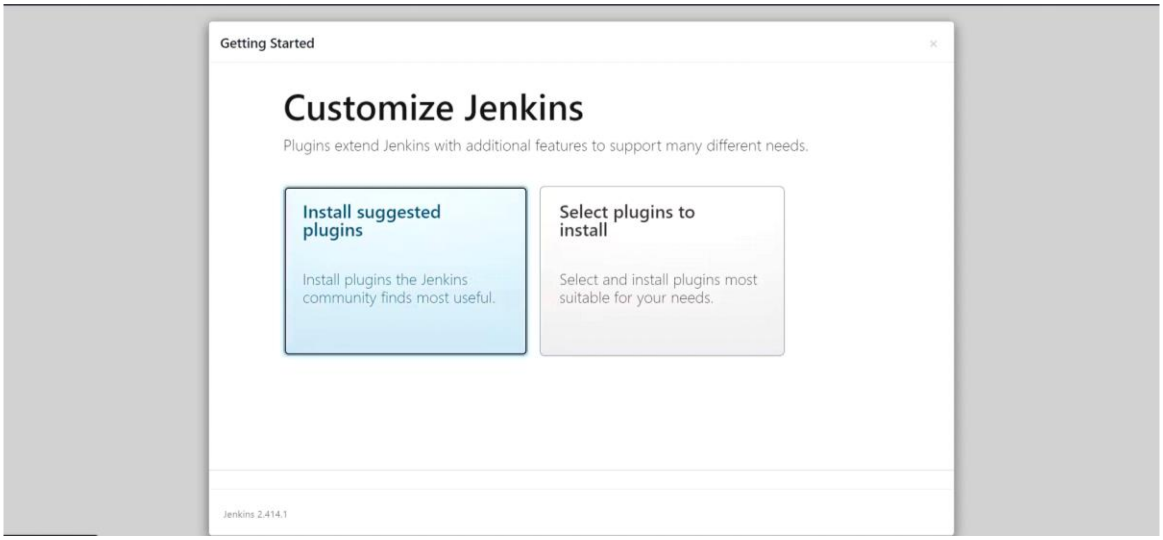
Unlock Jenkins

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:

```
/var/lib/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Unlock Jenkins using an administrative password and install the suggested plugins.



Getting Started

Customize Jenkins

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

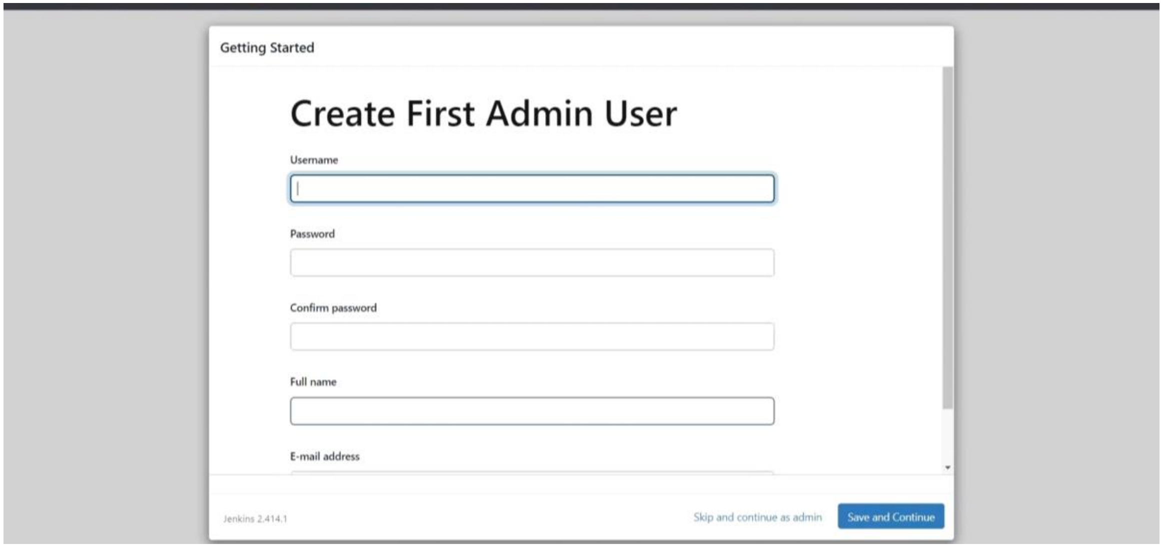
Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

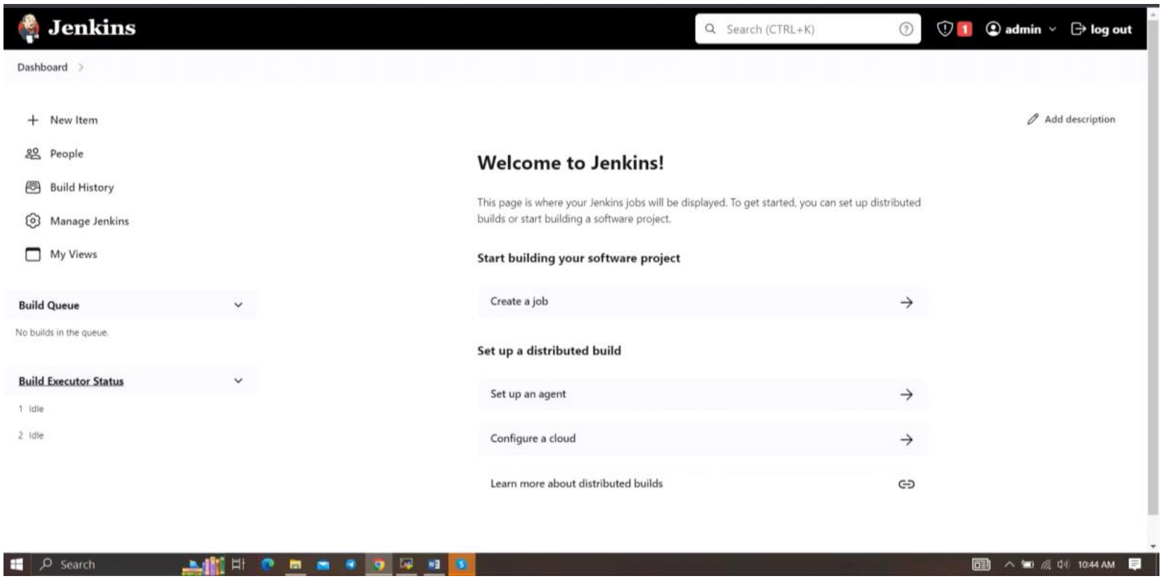
Select and install plugins most suitable for your needs.

Jenkins will now get installed and install all the libraries.



Create a user click on save and continue.

Jenkins Getting Started Screen.



No builds in the queue.

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

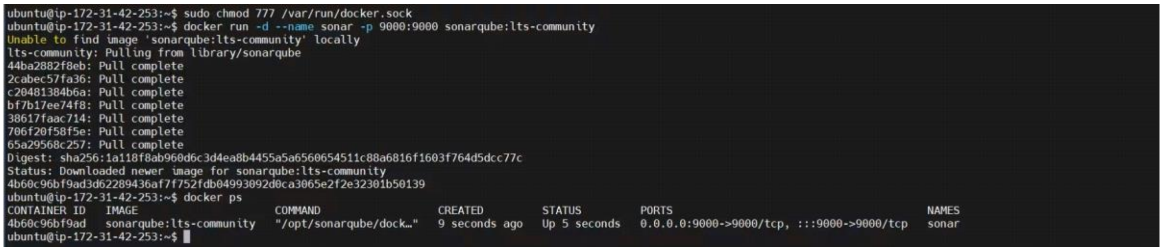
Set up a distributed build

Install Docker

```
sudo apt-get update

sudo apt-get install docker.io -y

sudo docker pull sonarqube:latest
```

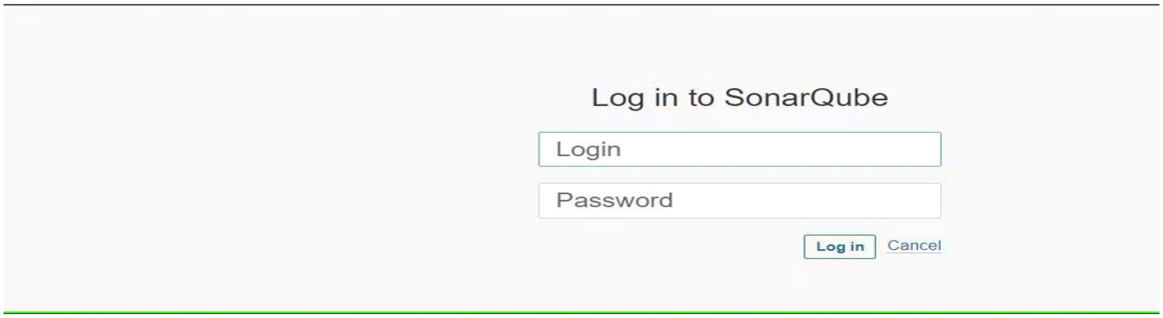
After the docker installation, we create a sonarqube container (Remember added 9000 ports in the security group

docker run -d --name sonar -p 9000:9000 sonarqube:latest



```
ubuntu@ip-172-31-42-253:~$ sudo chmod 777 /var/run/docker.sock
ubuntu@ip-172-31-42-253:~$ docker run -d --name sonar -p 9000:9000 sonarqube:lts-community
Unable to find image 'sonarqube:lts-community' locally
lts-community: Pulling from library/sonarqube
44ba2882f8eb: Pull complete
2cabec57fa36: Pull complete
c20481384b6a: Pull complete
bf7b17ee74f8: Pull complete
38617faac714: Pull complete
706f20f58f5e: Pull complete
65a29568c257: Pull complete
Digest: sha256:1a118f8ab960d6c3d4ea8b4455a5a6560654511c88a6816f1603f764d5dcc77c
Status: Downloaded newer image for sonarqube:lts-community
4b60c96bf9ad3d62289436af7f752fdb04993092d0ca3065e2f2e32301b50139
ubuntu@ip-172-31-42-253:~$ docker ps
CONTAINER ID   IMAGE                     COMMAND                  CREATED         STATUS         PORTS                                       NAMES
4b60c96bf9ad   sonarqube:lts-community   "/opt/sonarqube/dock…"   9 seconds ago   Up 5 seconds   0.0.0.0:9000->9000/tcp, :::9000->9000/tcp   sonar
ubuntu@ip-172-31-42-253:~$ 
```

Now our SonarQube is up and running

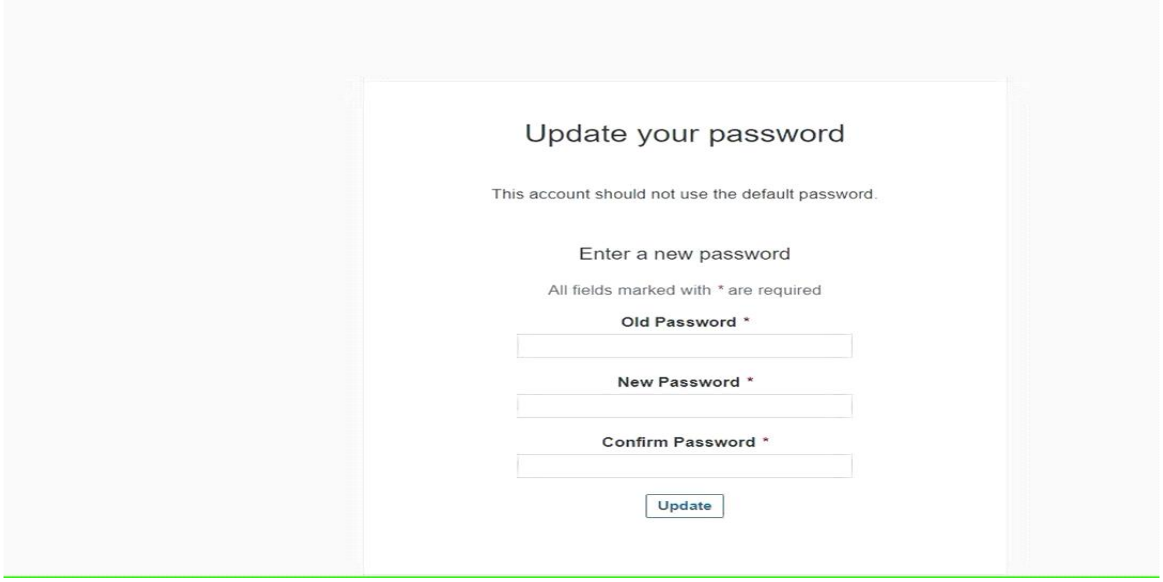


Log in to SonarQube

Enter username and password, click on login and change password

username admin

password admin



Update your password

This account should not use the default password.

Enter a new password

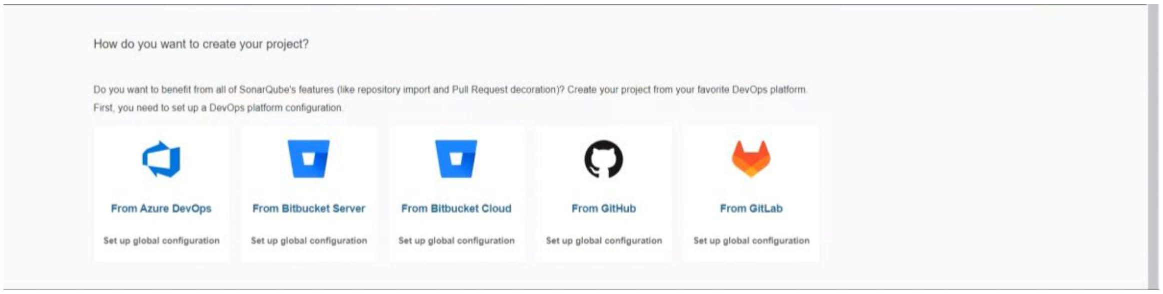
All fields marked with * are required

Old Password *

New Password *

Confirm Password *

Update New password, This is Sonar Dashboard.



How do you want to create your project?

Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform.
First, you need to set up a DevOps platform configuration.

From Azure DevOps
Set up global configuration
From Bitbucket Server
Set up global configuration
From Bitbucket Cloud
Set up global configuration
From GitHub
Set up global configuration
From GitLab
Set up global configuration

Install Trivy

```
vi trivy.sh

sudo apt-get install wget apt-transport-https gnupg lsb-release -y

wget -qO - https://aquasecurity.github.io/trivy-repo/deb/public.key | gpg --dearmor | sudo tee /usr/share/keyrings/trivy.gpg > /dev/null

echo "deb [signed-by=/usr/share/keyrings/trivy.gpg] https://aquasecurity.github.io/trivy-repo/deb $(lsb_release -sc) main" | sudo tee -a /etc/apt/sources.list.d/trivy.list

sudo apt-get update

sudo apt-get install trivy -y
```

Next, we will log in to Jenkins and start to configure our Pipeline in Jenkins

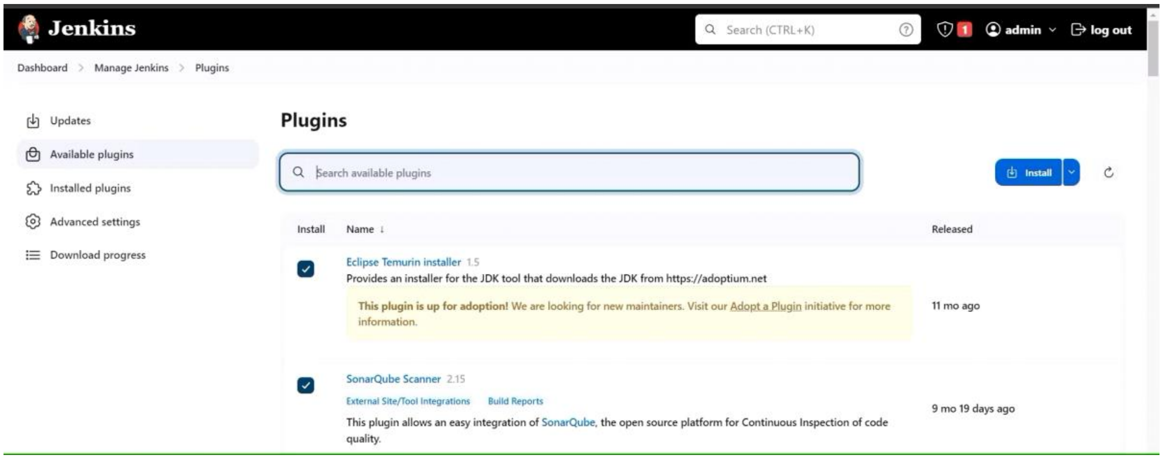
Install Plugins like JDK, Sonarqube Scanner, Maven, OWASP Dependency Check

Install Plugin

Goto Manage Jenkins →Plugins → Available Plugins →Install below plugins

Eclipse Temurin Installer (Install without restart)

SonarQube Scanner (Install without restart)

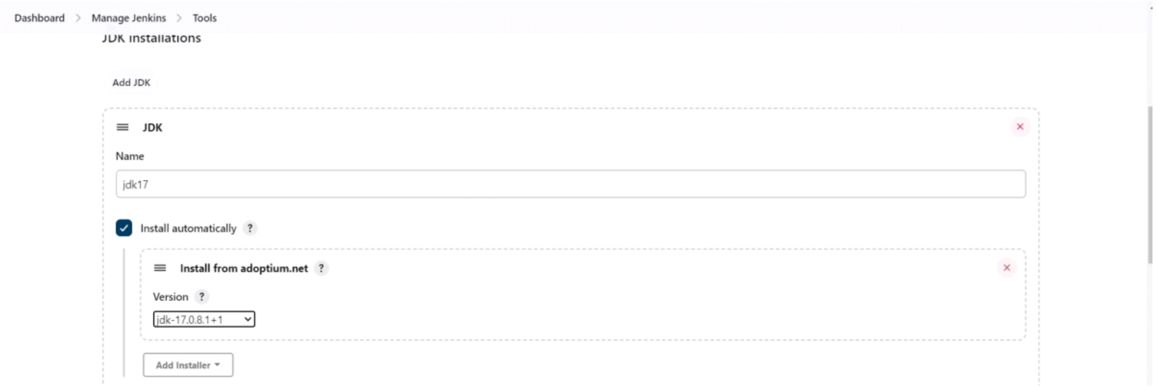


Plugins

Install	Name ↓	Released
☑	Eclipse Temurin installer 1.5 Provides an installer for the JDK tool that downloads the JDK from https://adoptium.net **This plugin is up for adoption!** We are looking for new maintainers. Visit our Adopt a Plugin initiative for more information.	11 mo ago
☑	SonarQube Scanner 2.15 External Site/Tool Integrations Build Reports This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.	9 mo 19 days ago

Configure Java and Maven in Global Tool Configuration

Goto Manage Jenkins → Tools → Install JDK(17) and Maven3(3.6.0) → Click on Apply and Save



JDK installations

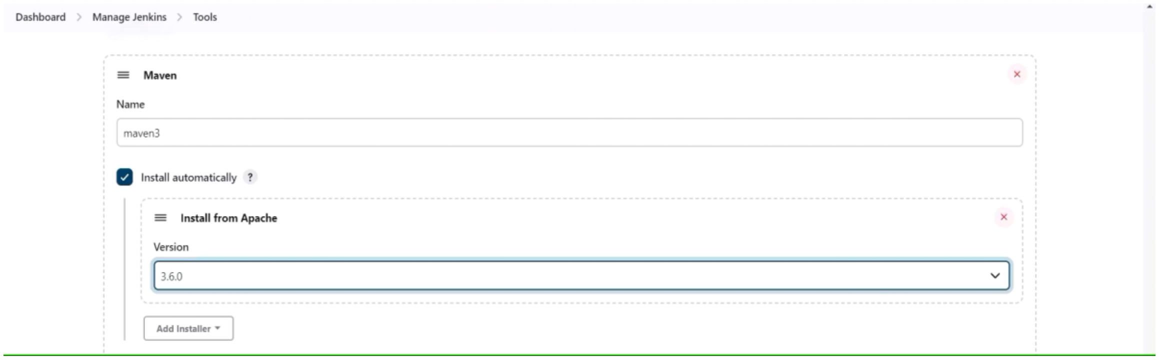
JDK

Name: jdk17

☑ Install automatically

Install from adoptium.net

Version: jdk-17.0.8.1+1



Maven

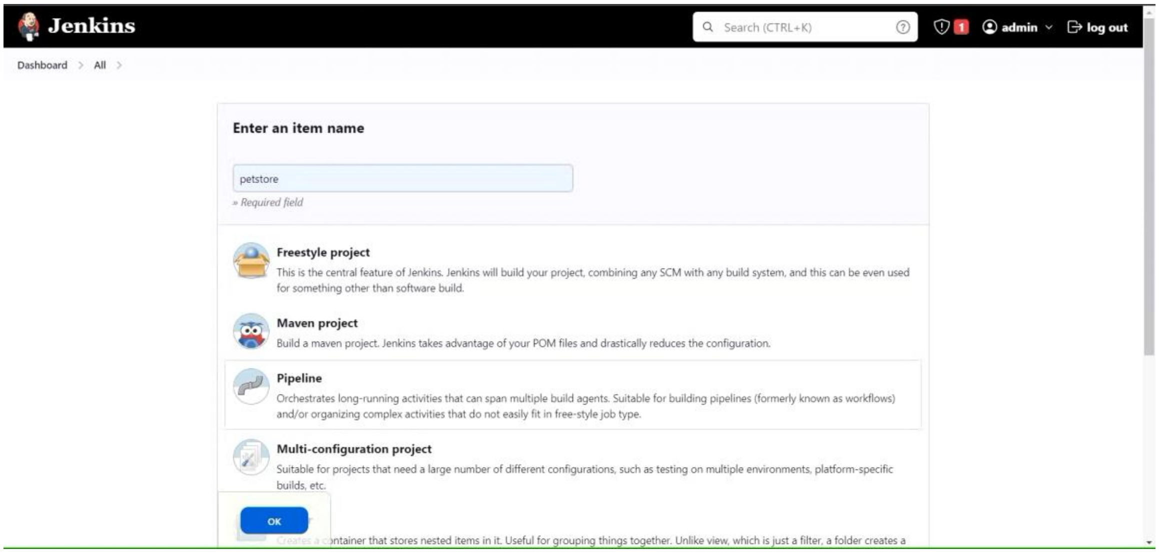
Name: maven3

☑ Install automatically

Install from Apache

Version: 3.6.0

Create a Job in pipeline Script



Enter an item name

petstore

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

Pipeline
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.

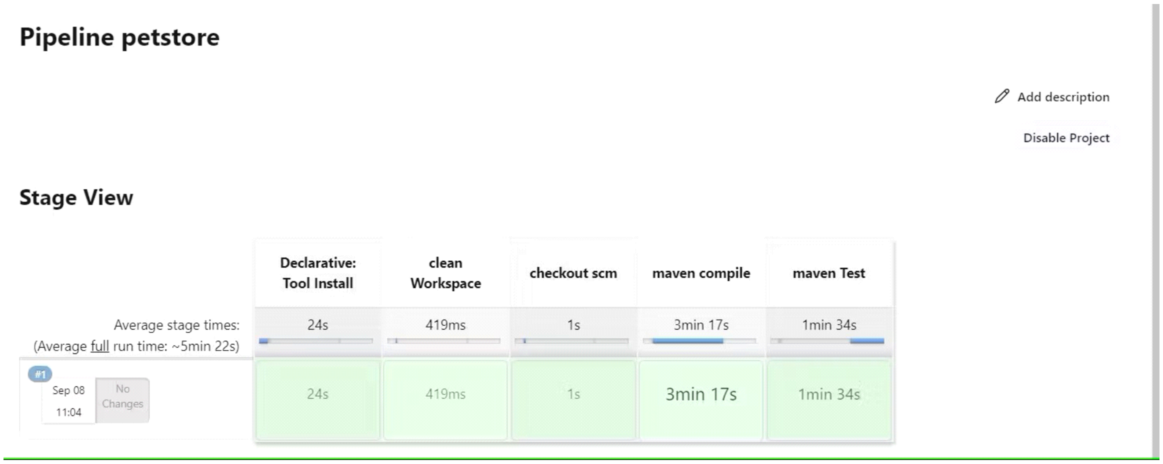
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a

Enter this in Pipeline Script,

```
pipeline{
   agent any
   tools {
      jdk 'jdk17'
      maven 'maven3'
   }
   stages{
      stage ('clean Workspace'){
         steps{
            cleanWs()
         }
      }
      stage ('checkout scm') {
         steps {
            git ' '
         }
      }
```

```
    stage ('maven compile') {
      steps {
        sh 'mvn clean compile'
      }
    }
    stage ('maven Test') {
      steps {
        sh 'mvn test'
      }
    }
  }
}
```

The stage view would look like this,

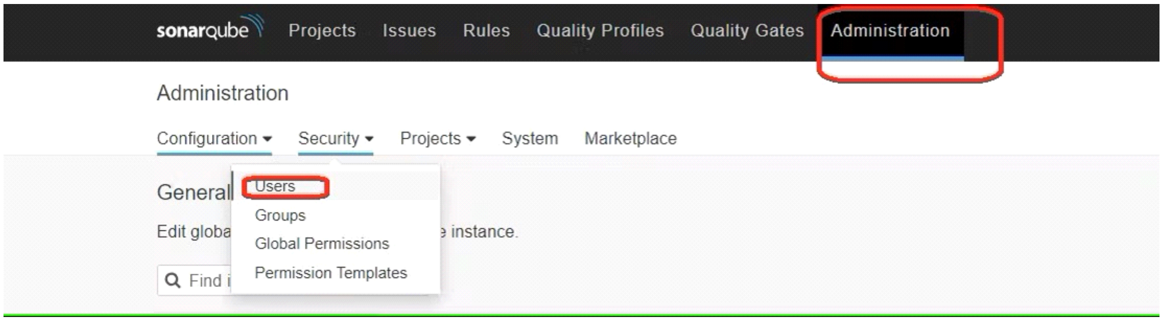


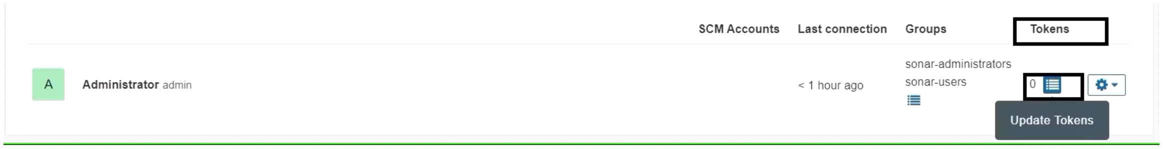
Pipeline petstore

Stage View

	Declarative: Tool Install	clean Workspace	checkout scm	maven compile	maven Test
Average stage times: (Average full run time: ~5min 22s)	24s	419ms	1s	3min 17s	1min 34s
#1 Sep 08 11:04 No Changes	24s	419ms	1s	3min 17s	1min 34s

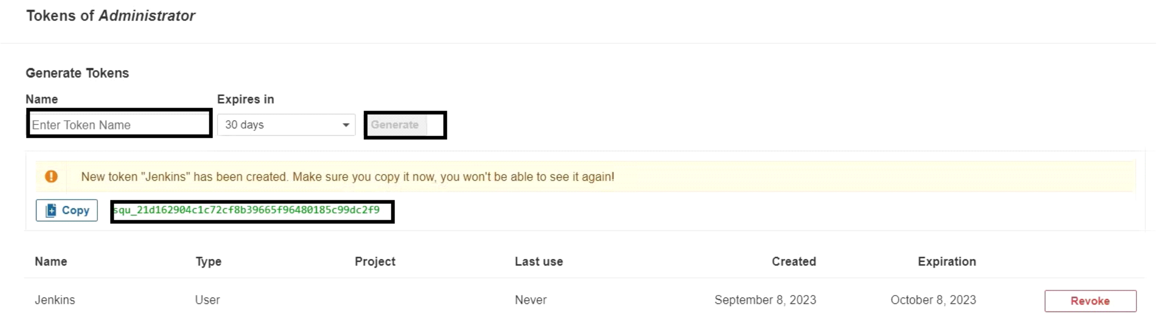
Configure Sonar Server in Manage Jenkins





	SCM Accounts	Last connection	Groups	Tokens
A Administrator admin		< 1 hour ago	sonar-administrators sonar-users	0

Create a token with a name and generate



Tokens of *Administrator*

Generate Tokens

Name: Enter Token Name

Expires in: 30 days

New token "Jenkins" has been created. Make sure you copy it now, you won't be able to see it again!

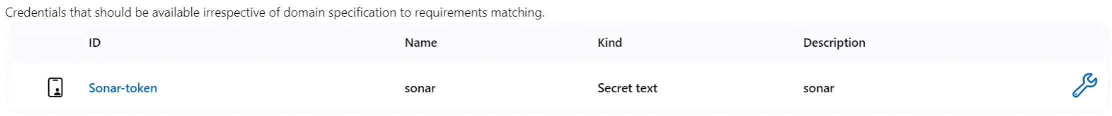
```
squ_21d162904c1c72cf8b39665f96480185c99dc2f9
```

Name	Type	Project	Last use	Created	Expiration	
Jenkins	User		Never	September 8, 2023	October 8, 2023	Revoke

copy Token

Goto Jenkins Dashboard → Manage Jenkins → Credentials → Add Secret Text. It should look like this

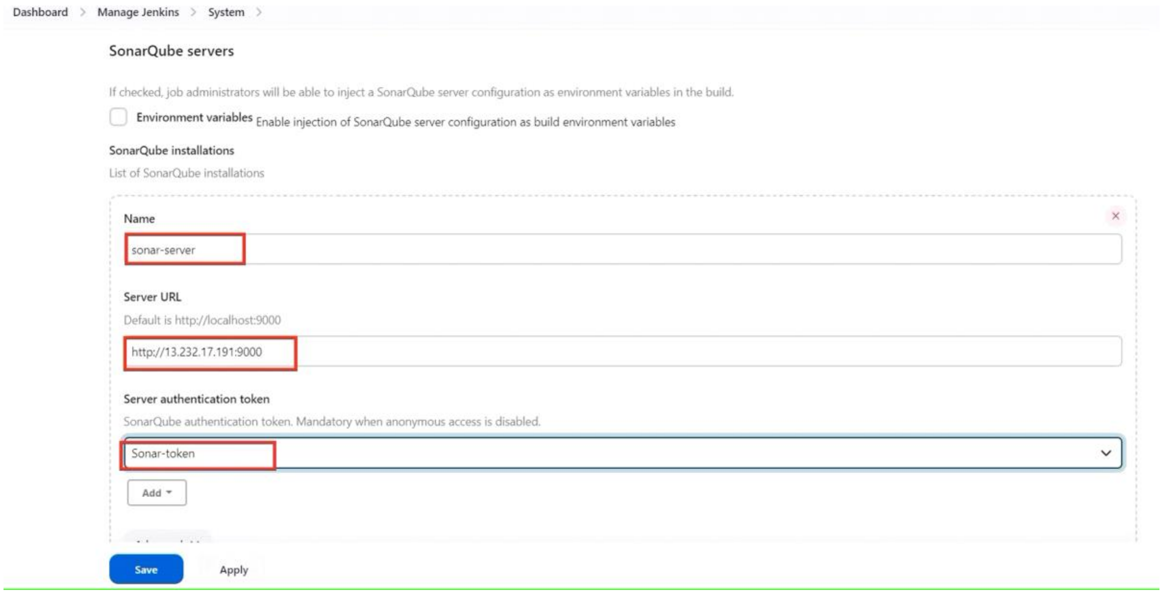
You will this page once you click on create



Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description
Sonar-token	sonar	Secret text	sonar

Now, go to Dashboard → Manage Jenkins → System and Add like the below image.



SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☐ Environment variables Enable injection of SonarQube server configuration as build environment variables

SonarQube installations

List of SonarQube installations

Name: sonar-server

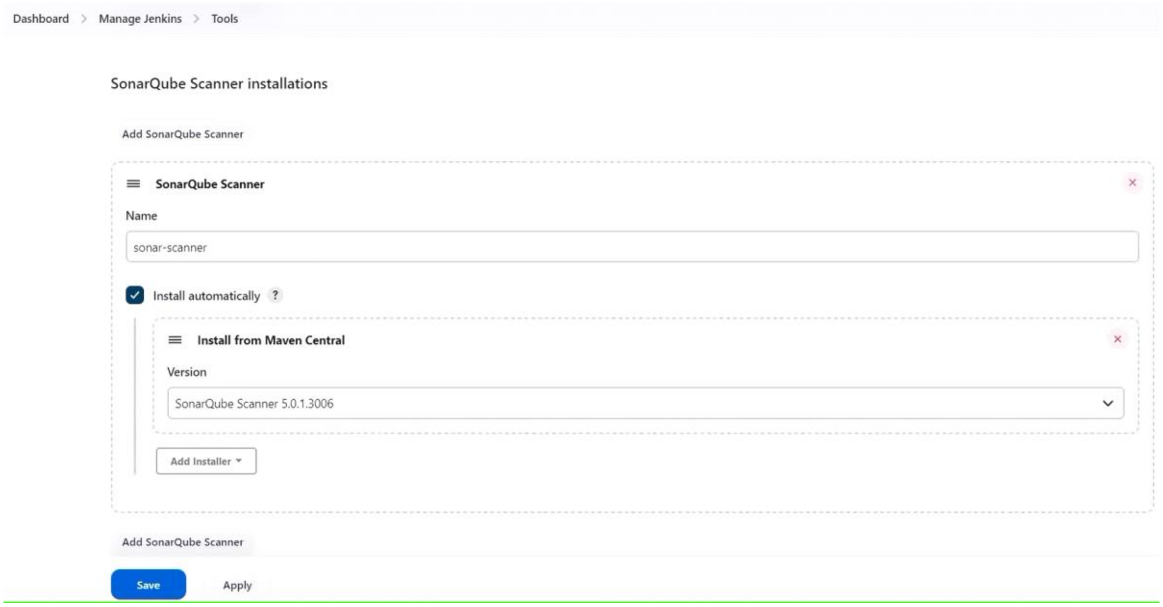
Server URL: Default is http://localhost:9000

http://13.232.17.191:9000

Server authentication token: SonarQube authentication token. Mandatory when anonymous access is disabled.

Sonar-token

Click on Apply and Save



SonarQube Scanner installations

SonarQube Scanner

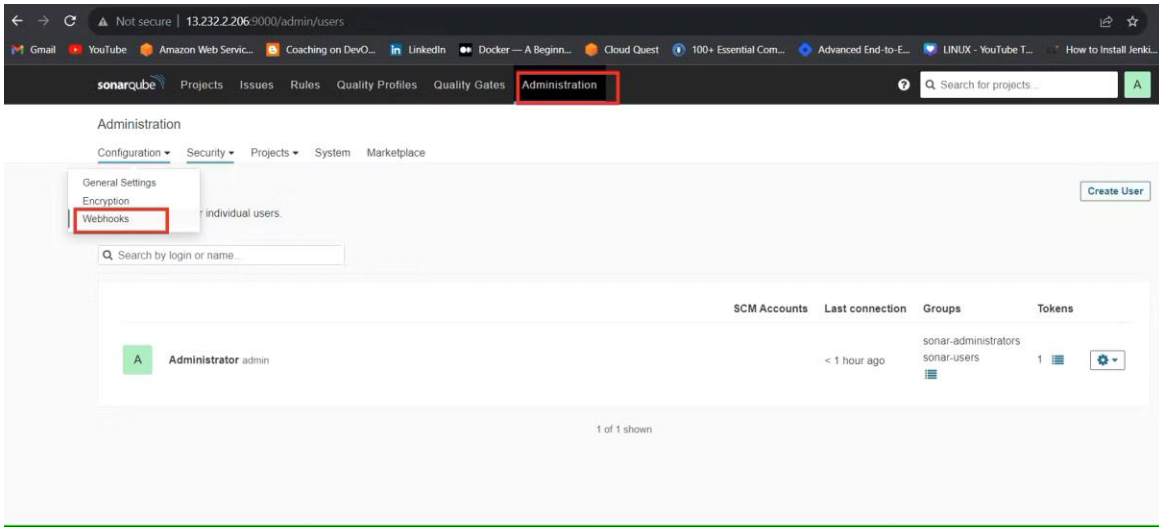
Name: sonar-scanner

☑ Install automatically ?

Install from Maven Central

Version: SonarQube Scanner 5.0.1.3006

In the Sonarqube Dashboard add a quality gate also-->Administration--> Configuration-->Webhooks

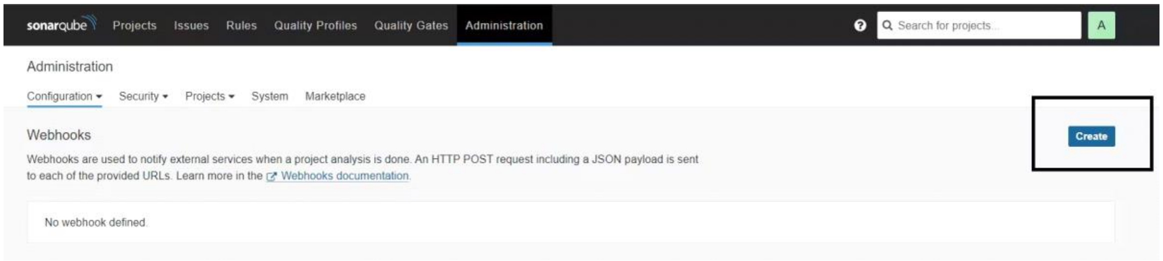


r individual users.

	SCM Accounts	Last connection	Groups	Tokens
A Administrator admin		< 1 hour ago	sonar-administrators sonar-users	1

1 of 1 shown

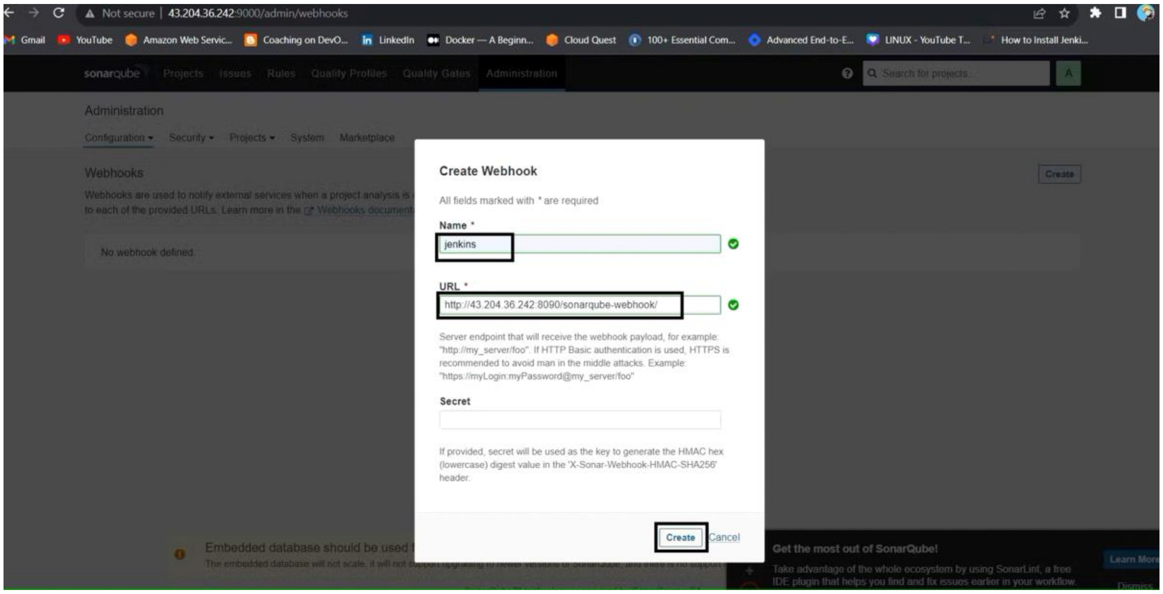
Click on Create



Webhooks

Webhooks are used to notify external services when a project analysis is done. An HTTP POST request including a JSON payload is sent to each of the provided URLs. Learn more in the Webhooks documentation.

No webhook defined.



Webhooks

Webhooks are used to notify external services when a project analysis is
to each of the provided URLs. Learn more in the Webhooks document

No webhook defined.

Create Webhook

All fields marked with * are required

Name *: jenkins

URL *: http://43.204.36.242:8090/sonarqube-webhook/

Server endpoint that will receive the webhook payload, for example: "http://my_server/foo". If HTTP Basic authentication is used, HTTPS is recommended to avoid man in the middle attacks. Example: "https://myLogin:myPassword@my_server/foo"

Secret

If provided, secret will be used as the key to generate the HMAC hex (lowercase) digest value in the 'X-Sonar-Webhook-HMAC-SHA256' header.

Embedded database should be used f

The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support

Get the most out of SonarQube!

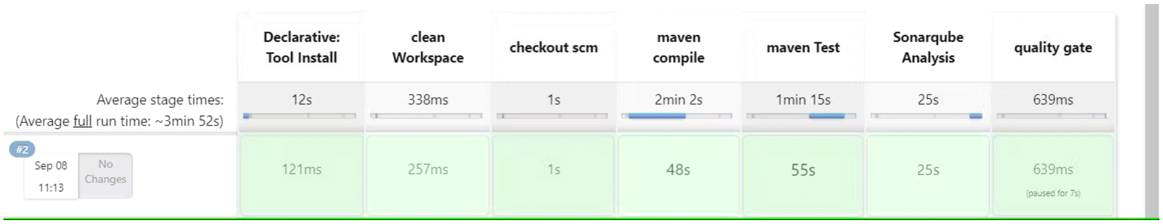
Take advantage of the whole ecosystem by using SonarLint, a free IDE plugin that helps you find and fix issues earlier in your workflow.

Let's write our Pipeline and add Sonarqube Stage in our Pipeline Script.

```
#under tools section add this environment

environment {

    SCANNER_HOME=tool 'sonar-scanner'

  }

# in stages add this

stage("Sonarqube Analysis "){

      steps{

        withSonarQubeEnv('sonar-server') {

          sh ''' $SCANNER_HOME/bin/sonar-scanner -Dsonar.projectName=Petshop \

          -Dsonar.java.binaries=. \

          -Dsonar.projectKey=Petshop '''

        }

      }

    }

    stage("quality gate"){

      steps {

        script {

         waitForQualityGate abortPipeline: false, credentialsId: 'sonar-token'
```

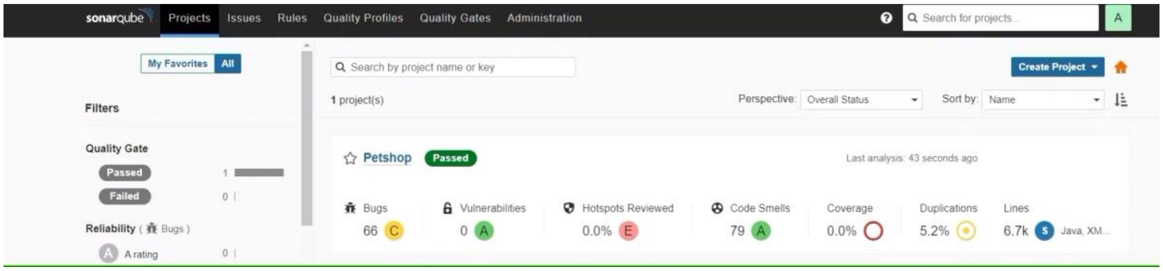
```
            }
        }
      }
```

Click on Build now, you will see the stage view like this



	Declarative: Tool Install	clean Workspace	checkout scm	maven compile	maven Test	Sonarqube Analysis	quality gate
Average stage times: (Average full run time: ~3min 52s)	12s	338ms	1s	2min 2s	1min 15s	25s	639ms
#2 Sep 08 11:13 No Changes	121ms	257ms	1s	48s	55s	25s	639ms (paused for 7s)

To see the report, you can go to Sonarqube Server and go to Projects.



1 project(s)

Quality Gate: Passed 1; Failed 0

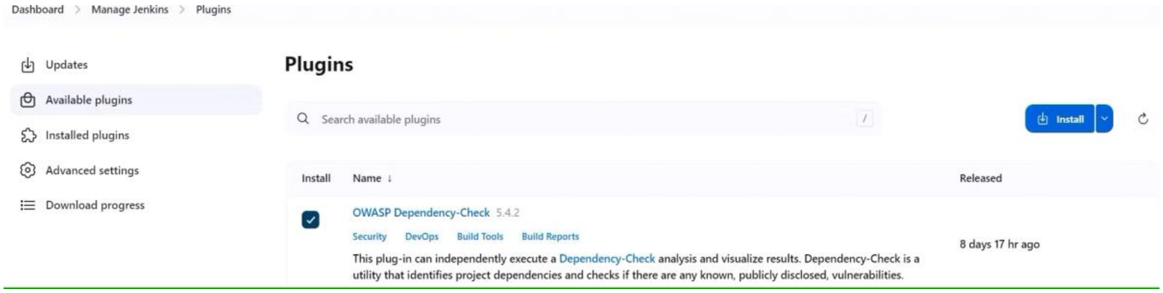
Reliability (Bugs): A rating 0

Petshop Passed — Last analysis: 43 seconds ago

Bugs	Vulnerabilities	Hotspots Reviewed	Code Smells	Coverage	Duplications	Lines
66 C	0 A	0.0% E	79 A	0.0%	5.2%	6.7k S Java, XM...

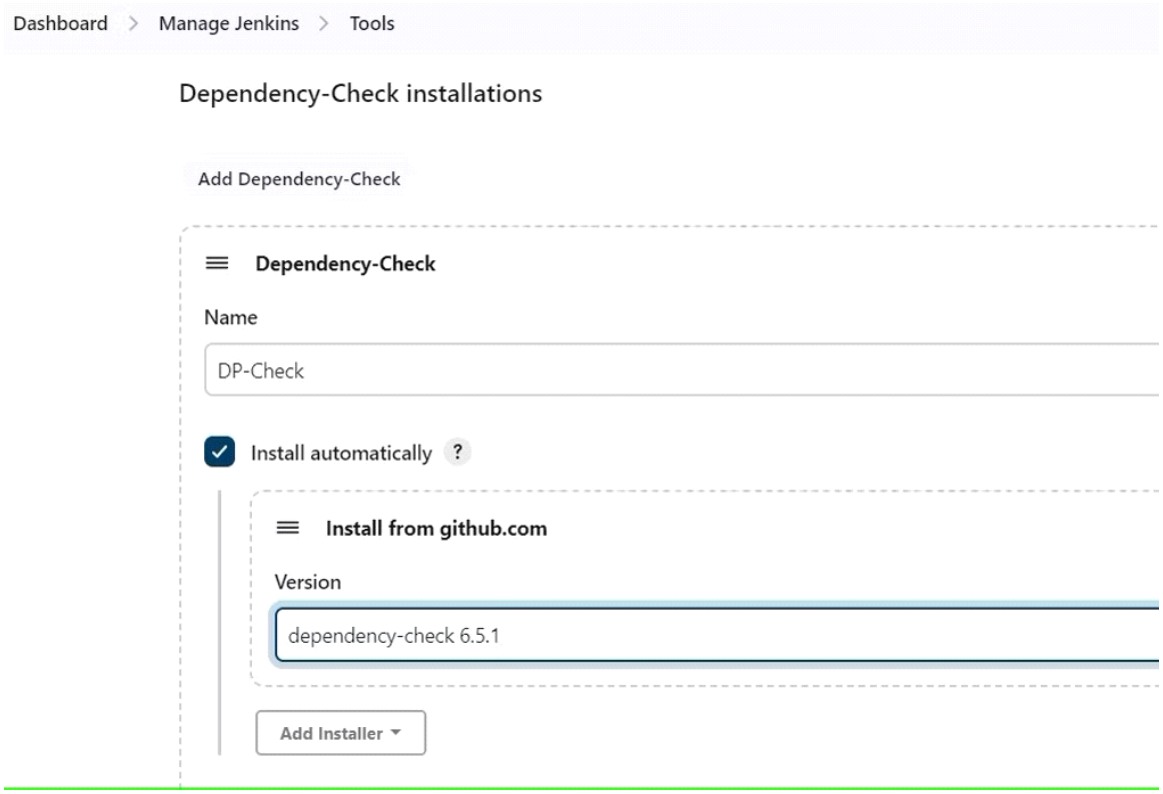
.Install OWASP Dependency Check Plugins

GotoDashboard → Manage Jenkins → Plugins → OWASP Dependency-Check. Click on it and install it without restart.



Install	Name ↓	Released
☑	OWASP Dependency-Check 5.4.2 Security DevOps Build Tools Build Reports This plug-in can independently execute a Dependency-Check analysis and visualize results. Dependency-Check is a utility that identifies project dependencies and checks if there are any known, publicly disclosed, vulnerabilities.	8 days 17 hr ago

Goto Dashboard → Manage Jenkins → Tools →add Dependency-check



Dashboard > Manage Jenkins > Tools

Dependency-Check installations

Add Dependency-Check

Dependency-Check

Name: DP-Check

☑ Install automatically ?

Install from github.com

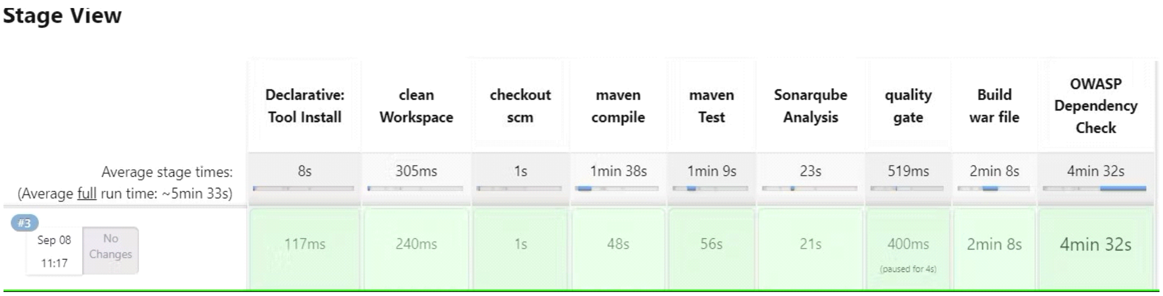
Version: dependency-check 6.5.1

Add Installer ▾

Click on Apply and Save .

Now go configure → Pipeline and add this stage to your pipeline and build.

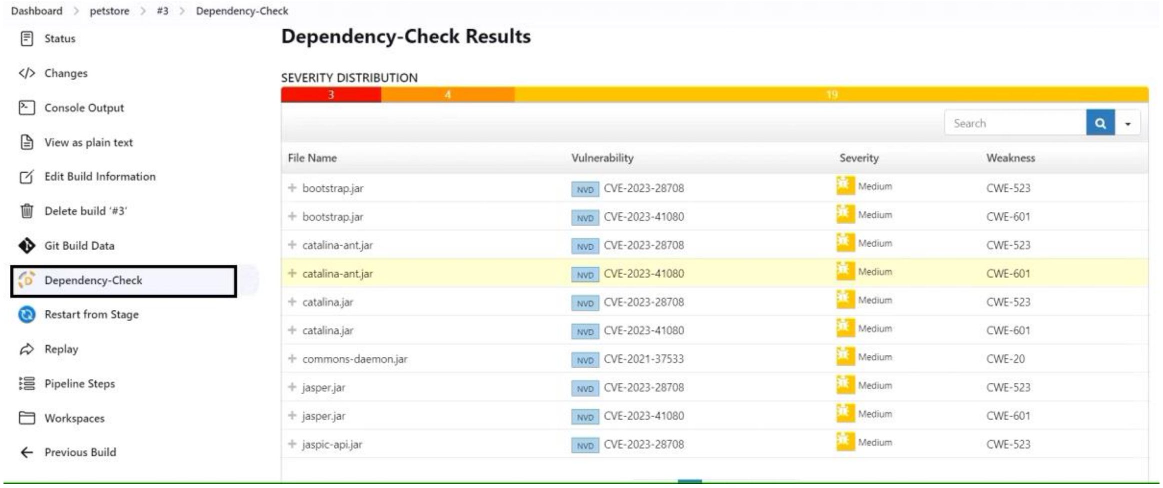
```
stage ('Build war file'){
      steps{
        sh 'mvn clean install -DskipTests=true'
      }
    }
    stage("OWASP Dependency Check"){
      steps{
        dependencyCheck additionalArguments: '--scan ./ --format XML ', odcInstallation: 'DP-Check'
        dependencyCheckPublisher pattern: '**/dependency-check-report.xml'
      }
    }
```



Stage View

	Declarative: Tool Install	clean Workspace	checkout scm	maven compile	maven Test	Sonarqube Analysis	quality gate	Build war file	OWASP Dependency Check
Average stage times: (Average full run time: ~5min 33s)	8s	305ms	1s	1min 38s	1min 9s	23s	519ms	2min 8s	4min 32s
#3 Sep 08 11:17 No Changes	117ms	240ms	1s	48s	56s	21s	400ms (paused for 4s)	2min 8s	4min 32s

You will see that in status, a graph will also be generated and Vulnerabilities.



Dependency-Check Results

SEVERITY DISTRIBUTION 3 4 19

File Name	Vulnerability	Severity	Weakness
+ bootstrap.jar	NVD CVE-2023-28708	Medium	CWE-523
+ bootstrap.jar	NVD CVE-2023-41080	Medium	CWE-601
+ catalina-ant.jar	NVD CVE-2023-28708	Medium	CWE-523
+ catalina-ant.jar	NVD CVE-2023-41080	Medium	CWE-601
+ catalina.jar	NVD CVE-2023-28708	Medium	CWE-523
+ catalina.jar	NVD CVE-2023-41080	Medium	CWE-601
+ commons-daemon.jar	NVD CVE-2021-37533	Medium	CWE-20
+ jasper.jar	NVD CVE-2023-28708	Medium	CWE-523
+ jasper.jar	NVD CVE-2023-41080	Medium	CWE-601
+ jaspic-api.jar	NVD CVE-2023-28708	Medium	CWE-523

Docker Image Build and Push

We need to install the Docker tool in our system, Goto Dashboard → Manage Plugins → Available plugins → Search for Docker and install these plugins

`Docker`

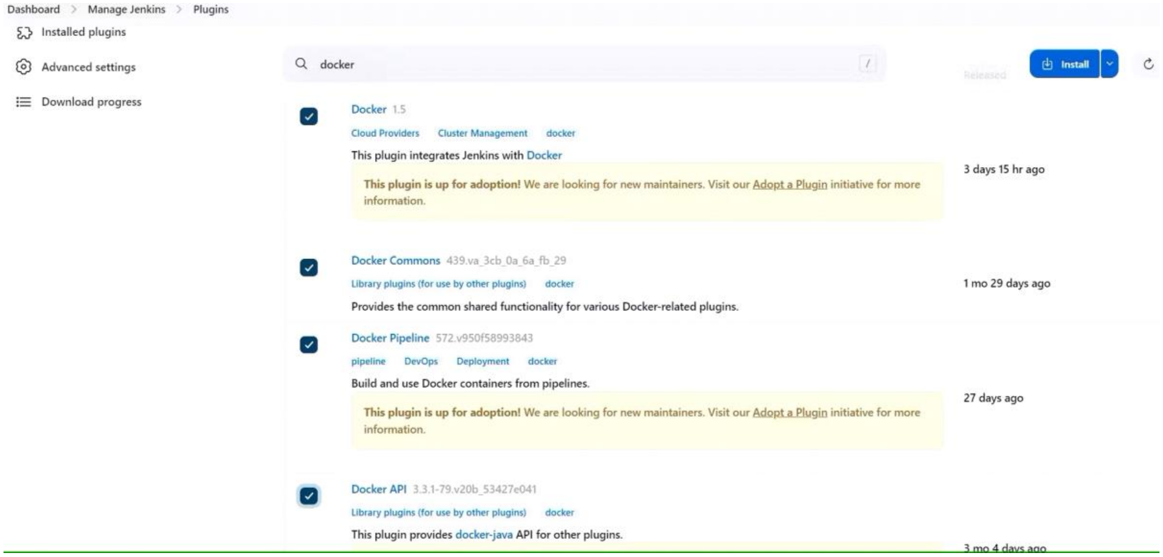
`Docker Commons`

`Docker Pipeline`

`Docker API`

`docker-build-step`

and click on install without restart



Dashboard > Manage Jenkins > Plugins

docker

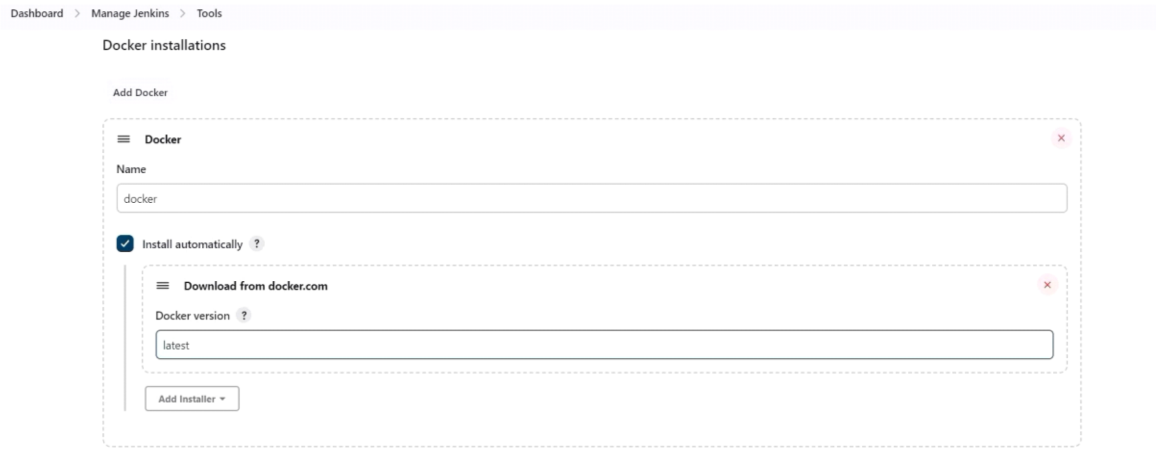
☑ Docker 1.5
Cloud Providers Cluster Management docker
This plugin integrates Jenkins with Docker
This plugin is up for adoption! We are looking for new maintainers. Visit our Adopt a Plugin initiative for more information.
3 days 15 hr ago

☑ Docker Commons 439.va_3cb_0a_6a_fb_29
Library plugins (for use by other plugins) docker
Provides the common shared functionality for various Docker-related plugins.
1 mo 29 days ago

☑ Docker Pipeline 572.v950f58993843
pipeline DevOps Deployment docker
Build and use Docker containers from pipelines.
This plugin is up for adoption! We are looking for new maintainers. Visit our Adopt a Plugin initiative for more information.
27 days ago

☑ Docker API 3.3.1-79.v20b_53427e041
Library plugins (for use by other plugins) docker
This plugin provides docker-java API for other plugins.
3 mo 4 days ago

Now, goto Dashboard → Manage Jenkins → Tools →



Dashboard > Manage Jenkins > Tools

Docker installations

Docker

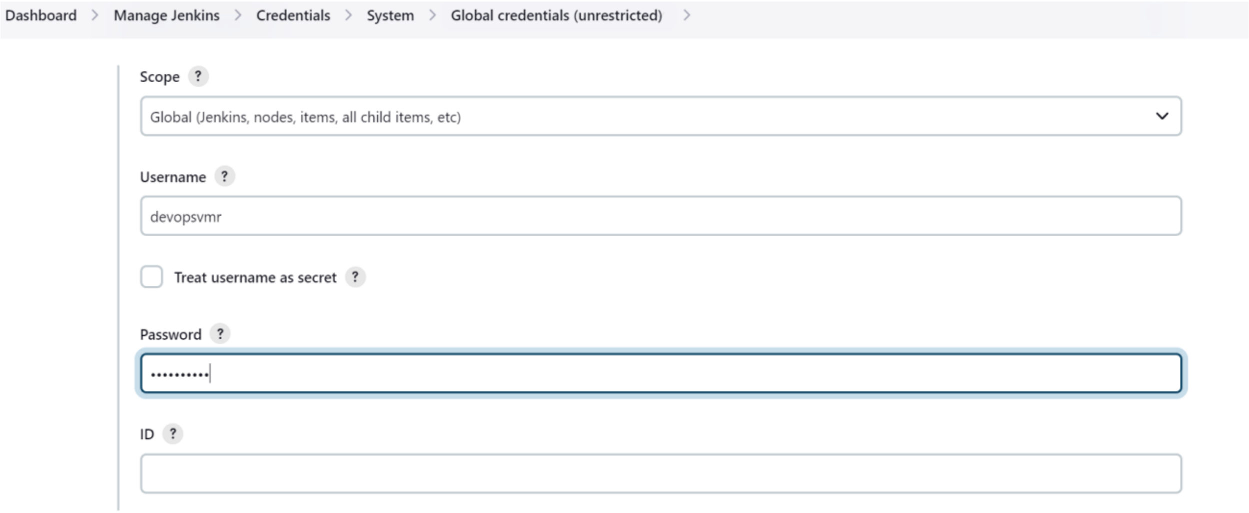
Name: docker

☑ Install automatically ?

Download from docker.com

Docker version ?: latest

Add DockerHub Username and Password under Global Credentials



Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

Scope ?: Global (Jenkins, nodes, items, all child items, etc)

Username ?: devopsvmr

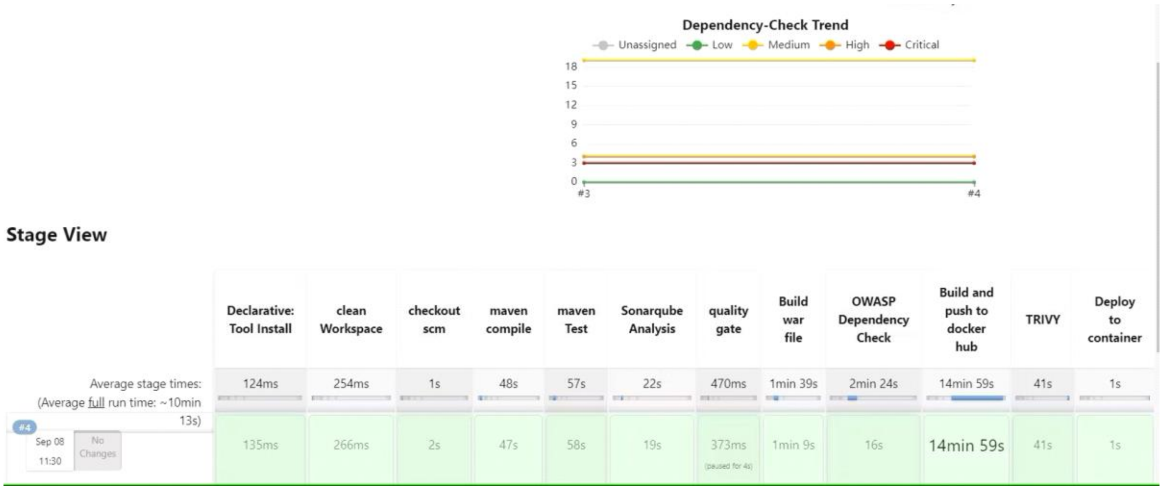
☐ Treat username as secret ?

Password ?: ••••••••••

ID ?:

Add this stage to Pipeline Script

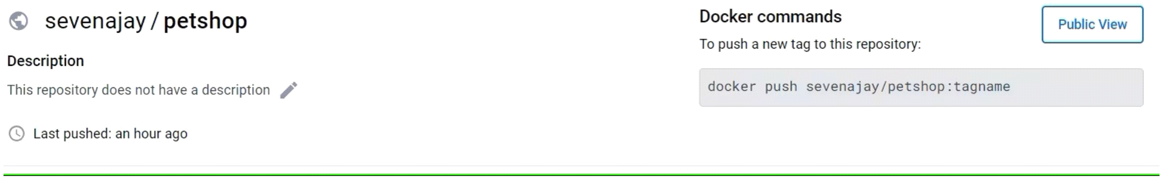
```
stage ('Build and push to docker hub'){
      steps{
        script{
          withDockerRegistry(credentialsId: 'docker', toolName: 'docker') {
            sh "docker build -t petshop ."
            sh "docker tag petshop devopsvmr/petshop:latest"
            sh "docker push devopsvmr/petshop:latest"
          }
        }
      }
    }
    stage("TRIVY"){
      steps{
        sh "trivy image devopsvmr/petshop:latest > trivy.txt"
      }
    }
    stage ('Deploy to container'){
      steps{
        sh 'docker run -d --name pet1 -p 8080:8080 devopsvmr/petshop:latest'
      }
    }
```



Stage View

	Declarative: Tool Install	clean Workspace	checkout scm	maven compile	maven Test	Sonarqube Analysis	quality gate	Build war file	OWASP Dependency Check	Build and push to docker hub	TRIVY	Deploy to container
Average stage times: (Average full run time: ~10min 13s)	124ms	254ms	1s	48s	57s	22s	470ms	1min 39s	2min 24s	14min 59s	41s	1s
#4 Sep 08 11:30 No Changes	135ms	266ms	2s	47s	58s	19s	373ms (paused for 4s)	1min 9s	16s	14min 59s	41s	1s

When you log in to Dockerhub, you will see a new image is created



sevenajay / **petshop**

Description

This repository does not have a description

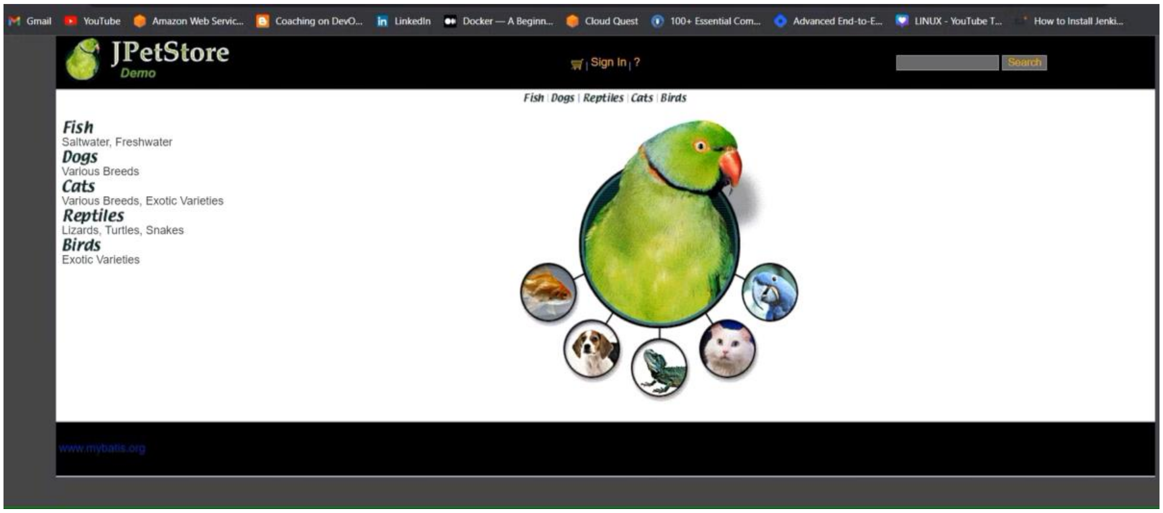
Last pushed: an hour ago

Docker commands

To push a new tag to this repository:

```
docker push sevenajay/petshop:tagname
```

this output :



Fish
Saltwater, Freshwater
Dogs
Various Breeds
Cats
Various Breeds, Exotic Varieties
Reptiles
Lizards, Turtles, Snakes
Birds
Exotic Varieties