

Mr Cloud Book

Home Blog DevSecOps Contact About Me Testimonials

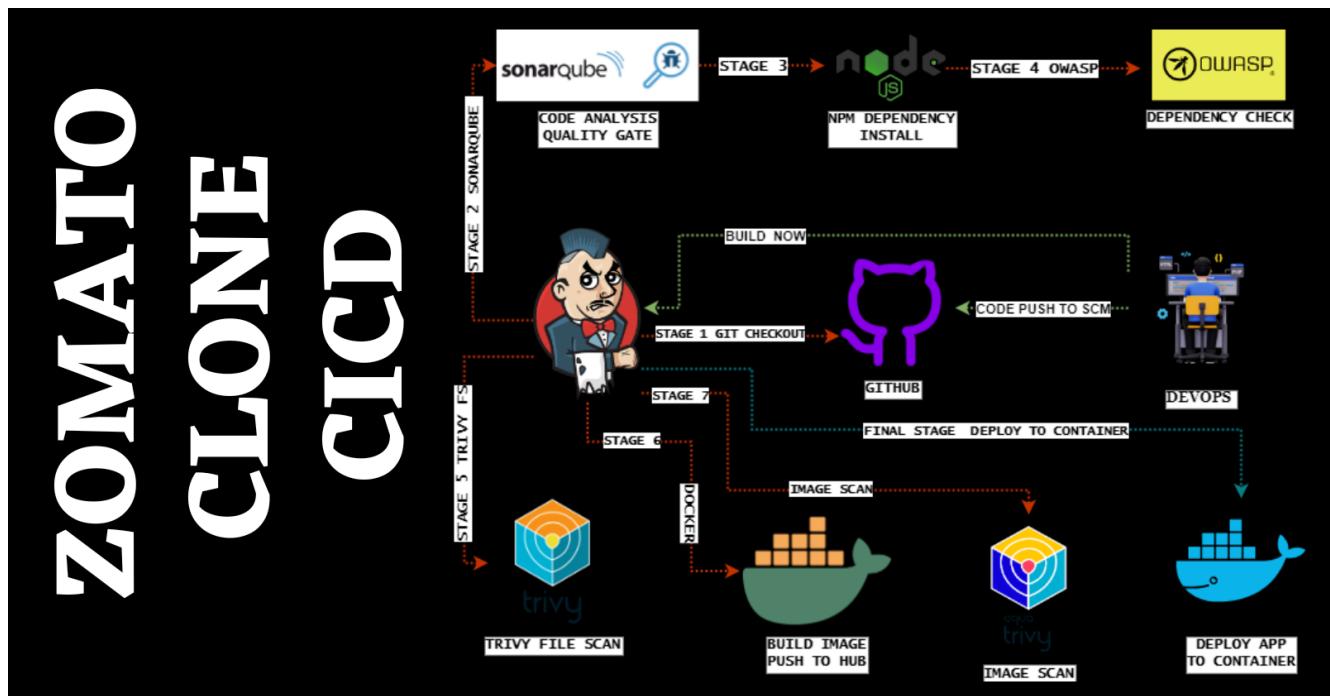
[Search Blogs](#)

DevOps

Zomato Clone App with DevSecOps CI/CD

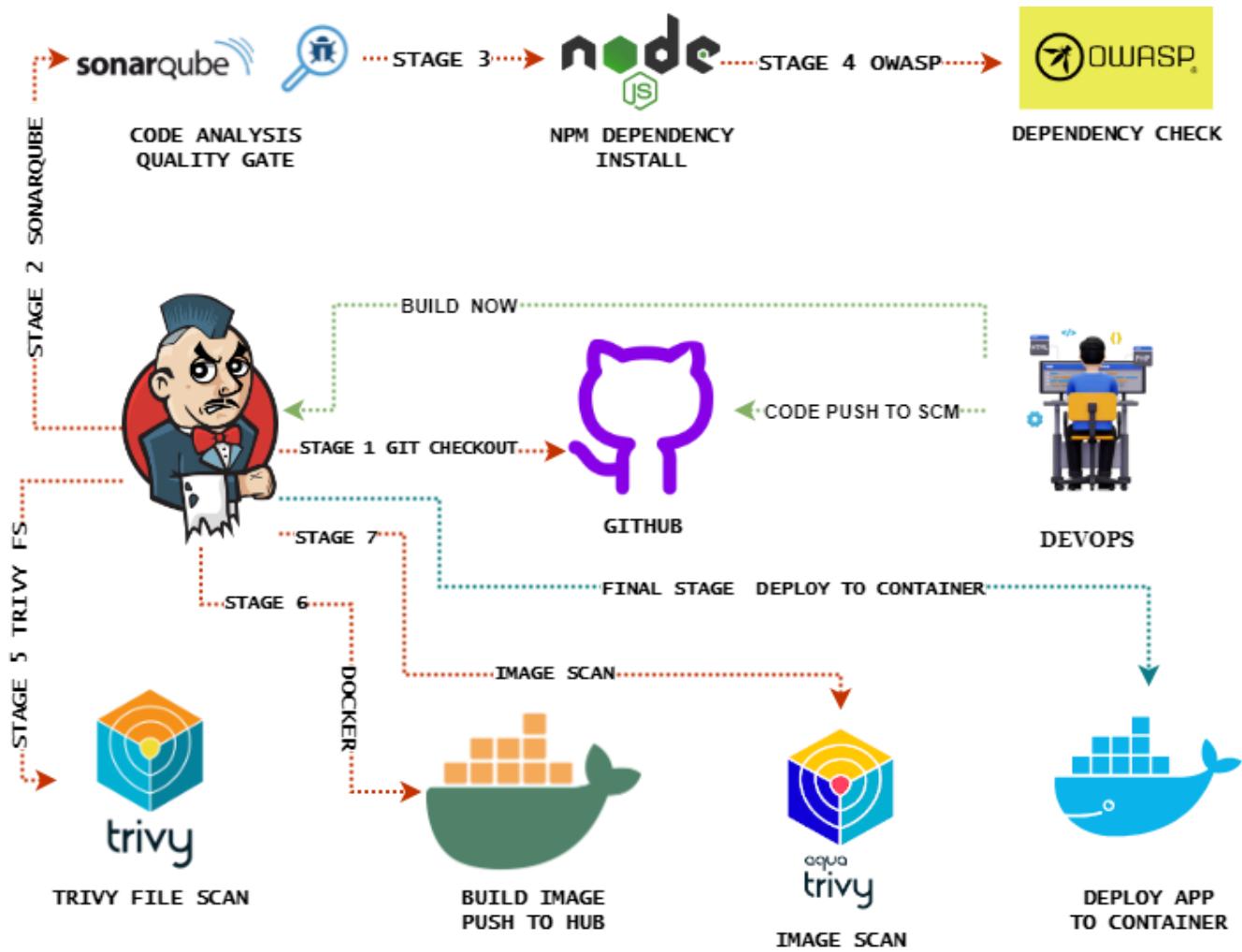


mrcloudbook.com · 8 January 2024



Hello friends, we will be deploying a React Js Zomato-clone. We will be using Jenkins as a CICD tool and deploying our application on a Docker container. I Hope this detailed blog is useful.

[CLICK HERE FOR GITHUB REPOSITORY](#)



Steps:-

Step 1 – Launch 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, Nodejs, 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 8 – Terminate the AWS EC2 Instances.

Now, let's get started and dig deeper into each of these steps:-

Contents [hide]

- [STEP1:Launch an Ubuntu\(22.04\) T2 Large Instance](#)
- [Step 2 – Install Jenkins, Docker and Trivy](#)
- [2A – To Install Jenkins](#)
- [2B – Install Docker](#)
- [2C – Install Trivy](#)
- [Step 3 – Install Plugins like JDK, Sonarqube Scanner, NodeJs, OWASP Dependency Check](#)
- [3A – Install Plugin](#)
- [3B – Configure Java and Nodejs in Global Tool Configuration](#)
- [3C – Create a Job](#)
- [Step 4 – Configure Sonar Server in Manage Jenkins](#)
- [Step 5 – Install OWASP Dependency Check Plugins](#)
- [Step 6 – Docker Image Build and Push](#)
- [Step 8: Terminate instances.](#)
- [Complete Pipeline](#)

STEP1:Launch an Ubuntu(22.04) T2 Large Instance

Launch an AWS T2 Large Instance. Use the image as Ubuntu. You can create a new key pair or use an existing one. Enable HTTP and HTTPS settings in the Security Group and open all ports (not best case to open all ports but just for learning purposes it's okay).

Instances (1) Info										
C Connect Instance state ▾ Actions ▾ Launch instances ▾										
<input type="text"/> Find instance by attribute or tag (case-sensitive)										
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 Dl			
CI-CD	i-065c10200537a1eee	Running	t2.large	2/2 checks passed	No alarms	ap-south-1a	ec2-52-66-14			

Step 2 – Install Jenkins, Docker and Trivy

2A – To Install Jenkins

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



```
vi jenkins.sh
```



```
#!/bin/bash
sudo apt update -y
#sudo apt upgrade -y
wget -O - https://packages.adoptium.net/artifactory/api/gpg/key/public
echo "deb [signed-by=/etc/apt/keyrings/adoptium.asc] https://packages.adoptium.net/artifactory/api/debian stable main" | sudo tee /etc/apt/sources.list.d/adoptium.list > /dev/null
sudo apt update -y
sudo apt install temurin-17-jdk -y
/usr/bin/java --version
curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /usr/share/keyrings/jenkins-keyring.asc > /dev/null
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-get update -y
sudo apt-get install jenkins -y
sudo systemctl start jenkins
sudo systemctl status jenkins
```



```
sudo chmod 777 jenkins.sh
./jenkins.sh
```



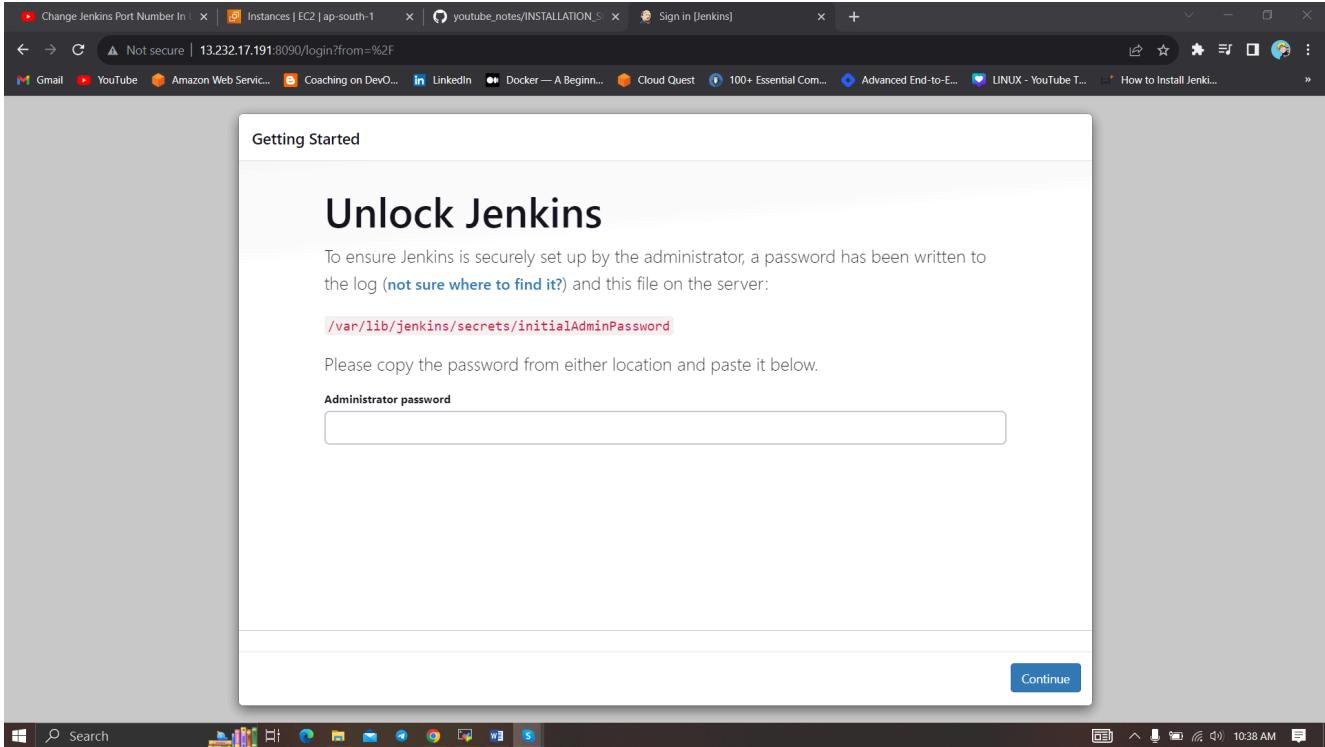
Once Jenkins is installed, you will need to go to your AWS EC2 Security Group and open Inbound Port 8080, since Jenkins works on Port 8080.

Now, grab your Public IP Address

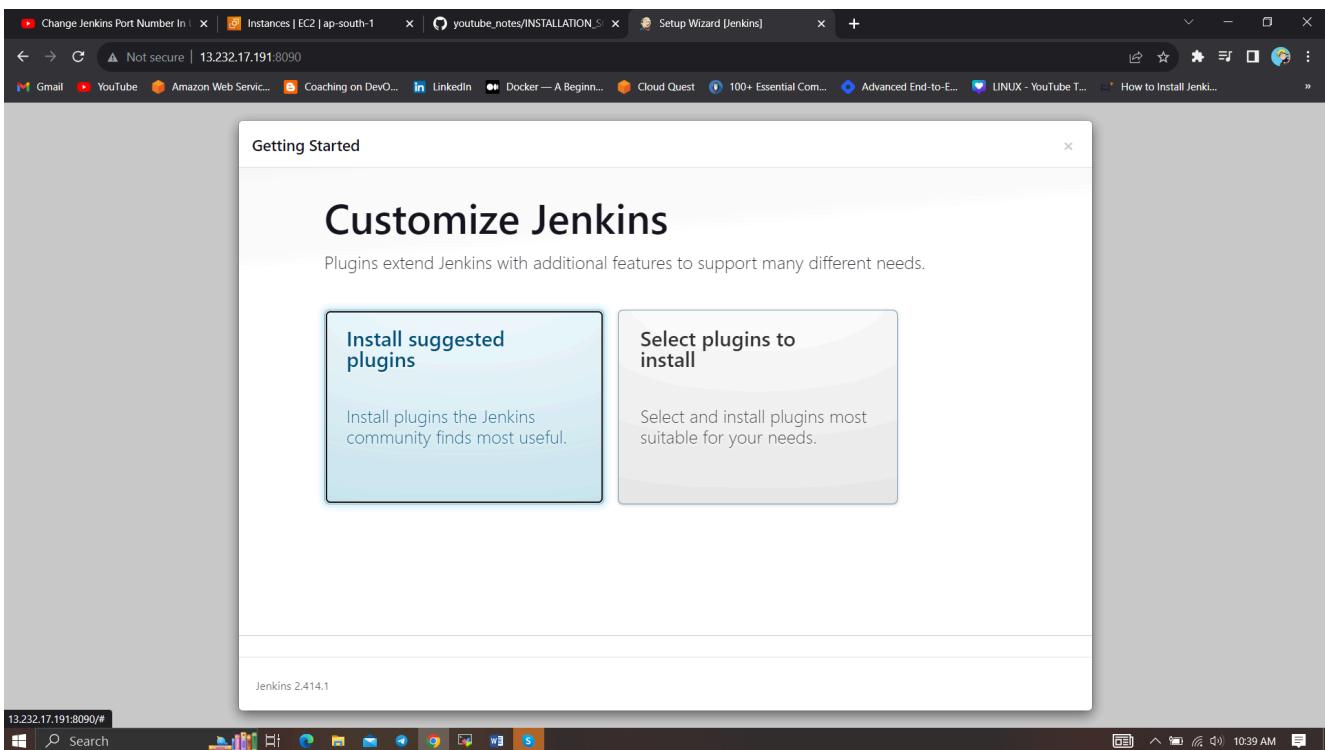


EC2 Public IP Address:8080

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



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



Jenkins will now get installed and install all the libraries.

Getting Started

Create First Admin User

Username

Password

Confirm password

Full name

E-mail address

Jenkins 2.414.1

Skip and continue as admin Save and Continue

Create a user click on save and continue.

Jenkins Getting Started Screen.

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

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

Create a job

Set up an agent

Configure a cloud

Learn more about distributed builds

2B – Install Docker

```
sudo apt-get update
sudo apt-get install docker.io -y
sudo usermod -aG docker $USER
newgrp docker
sudo chmod 777 /var/run/docker.sock
```



After the docker installation, we create a sonarqube container (Remember to add 9000 ports in the security group).

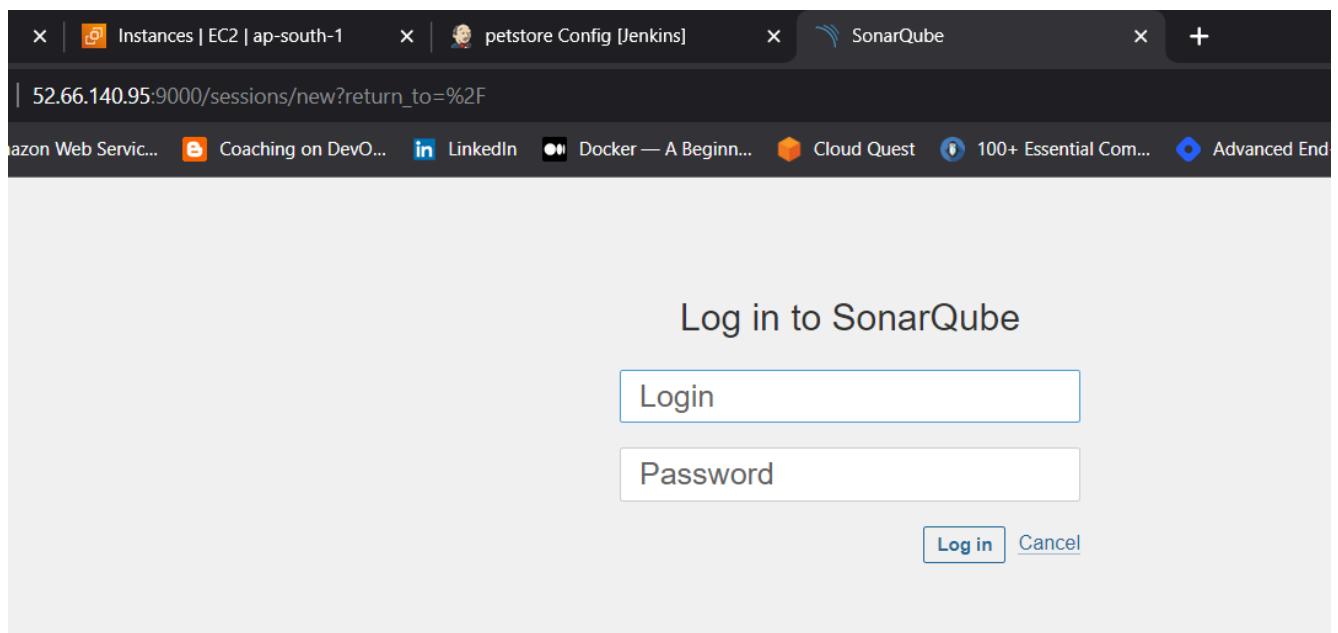


```
docker run -d --name sonar -p 9000:9000 sonarqube:lts-community
```



```
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
44ba2882f9eb: Pull complete
2cabecc57fa36: Pull complete
c20481384b6a: Pull complete
bf7b17ee74fb: Pull complete
38617faac714: Pull complete
706f20f58f5e: Pull complete
65a29566c257: Pull complete
Digest: sha256:1a118f8ab960d6c3d4ea8b4455a5a6560654511c88a6816f1603f764d5dcc77c
Status: Downloaded newer image for sonarqube:lts-community
4b66c96bf9ad3d62289436af/f752fdb04993092d0ca3065e2f2e32301b50139
ubuntu@ip-172-31-42-253:~$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
4b66c96bf9ad 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



Enter username and password, click on login and change password



```
username admin  
password admin
```



The screenshot shows a web browser window with the following details:

- Tab Bar:** Instances | EC2 | ap-south-1, petstore Config [Jenkins], SonarQube.
- Address Bar:** 56.140.95:9000/account/reset_password
- Toolbar:** Web Servic..., Coaching on DevO..., LinkedIn, Docker — A Beginn..., Cloud Quest, 100+ Essential Com..., Advanced End-to-E...
- Content Area:**
 - Title:** Update your password
 - Text:** This account should not use the default password.
 - Section:** Enter a new password
 - Text:** All fields marked with * are required
 - Form Fields:**
 - Old Password *
 - New Password *
 - Confirm Password *
 - Buttons:** Update

Update New password, This is Sonar Dashboard.

The screenshot shows the SonarQube interface for creating a new project. At the top, there's a navigation bar with links to Gmail, YouTube, Amazon Web Services, Coaching on DevOps, LinkedIn, Docker — A Beginner's Guide, Cloud Quest, 100+ Essential Commands, Advanced End-to-End, LINUX - YouTube Tutorials, and How to Install Jenkins. Below the navigation bar, the main content area asks "How do you want to create your project?". It provides five options with icons and links: "From Azure DevOps" (Azure icon), "From Bitbucket Server" (Bitbucket icon), "From Bitbucket Cloud" (Bitbucket icon), "From GitHub" (GitHub icon), and "From GitLab" (GitLab icon). Each option includes a "Set up global configuration" link.

2C – 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 > /usr/share/keyrings/trivy.gpg
echo "deb [signed-by=/usr/share/keyrings/trivy.gpg] https://aquasecurity.net/trivy-repo stable main" | tee /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

Step 3 – Install Plugins like JDK, Sonarqube Scanner, NodeJs, OWASP Dependency Check

3A – Install Plugin

Goto Manage Jenkins → Plugins → Available Plugins →

Install below plugins

1 → Eclipse Temurin Installer (Install without restart)

2 → SonarQube Scanner (Install without restart)

3 → NodeJs Plugin (Install Without restart)

The screenshot shows the Jenkins Plugins page. On the left, there's a sidebar with links for Updates, Available plugins (which is selected), Installed plugins, Advanced settings, and Download progress. The main area has a search bar at the top right. Below it, there are two sections: one for the Eclipse Temurin installer (version 1.5) and another for the SonarQube Scanner (version 2.15). Both sections show a green checkmark indicating they are installed. The SonarQube Scanner section also includes links for External Site/Tool Integrations and Build Reports.

This screenshot shows the continuation of the Jenkins Plugins page. It displays the NodeJS plugin (version 1.6.1) which is also installed, indicated by a green checkmark. A note below the plugin says "NodeJS Plugin executes NodeJS script as a build step."

3B – Configure Java and Nodejs in Global Tool Configuration

Goto Manage Jenkins → Tools → Install JDK(17) and NodeJs(16)→ Click on Apply and Save

The screenshot shows the Jenkins Global Tool Configuration page under the Tools section. It's set up to add a new JDK named "jdk17". The "Install automatically" checkbox is checked. Below it, there's a dropdown menu for "Install from adoptium.net" with "jdk-17.0.8.1+1" selected. There's also a "Add Installer" button.

3C – Create a Job

create a job as Zomato Name, select pipeline and click on ok.

Step 4 – Configure Sonar Server in Manage Jenkins

Grab the Public IP Address of your EC2 Instance, Sonarqube works on Port 9000, so <Public IP>:9000. Goto your Sonarqube Server. Click on Administration → Security → Users → Click on Tokens and Update Token → Give it a name → and click on Generate Token

click on update Token

Create a token with a name and generate

Tokens of Administrator**Generate Tokens**

Name	Expires in
<input type="text" value="Enter Token Name"/>	30 days
<input type="button" value="Generate"/>	

! 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	<input type="button" value="Revoke"/>

copy Token

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

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

New credentials

Kind: Secret text

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

Secret: **POST THE TOKEN HERE**

ID: Sonar-token

Description: Sonar-token

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	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	<input type="text" value="sonar-server"/>	X
Server URL	Default is http://localhost:9000 <input type="text" value="http://13.232.17.191:9000"/>	
Server authentication token	SonarQube authentication token. Mandatory when anonymous access is disabled. <input type="text" value="Sonar-token"/>	
<input style="margin-right: 10px;" type="button" value="Add"/> <input style="background-color: #0072bc; color: white; border-radius: 5px; padding: 5px; margin-right: 10px;" type="button" value="Save"/> <input type="button" value="Apply"/>		

Click on Apply and Save

The Configure System option is used in Jenkins to configure different server

Global Tool Configuration is used to configure different tools that we install using Plugins

We will install a sonar scanner in the tools.

SonarQube Scanner installations**SonarQube Scanner****Name**

Install automatically ?

Install from Maven Central**Version**

In the Sonarqube Dashboard add a quality gate also

Administration-> Configuration->Webhooks

The screenshot shows the SonarQube administration interface. The top navigation bar has tabs for Gmail, YouTube, Amazon Web Servic..., Coaching on DevO..., LinkedIn, Docker — A Beginn..., Cloud Quest, 100+ Essential Com..., Advanced End-to-E..., LINUX - YouTube T..., How to Install Jenki..., sonarqube, Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, and a search bar. The Administration tab is highlighted with a red box.

The main content area is titled "Administration" and "Configuration". Under "Configuration", there are links for General Settings, Encryption, and Webhooks, with the Webhooks link highlighted by a red box. Below this is a search bar with placeholder text "Search by login or name...".

A table lists users with columns for SCM Accounts, Last connection, Groups, and Tokens. One user, "Administrator admin", is listed with a green profile icon. The "Last connection" column shows "< 1 hour ago". The "Groups" column lists "sonar-administrators" and "sonar-users". The "Tokens" column shows "1" and a gear icon. At the bottom of the table, it says "1 of 1 shown".

Click on Create

The screenshot shows the SonarQube configuration page for Webhooks. The top navigation bar is identical to the previous screenshot. The main content area is titled "Webhooks". It contains a brief description: "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](#)". To the right of this text is a large "Create" button highlighted with a black box.

Below the description, a message states "No webhook defined.".

Add details

```
#in url section of quality gate
http://jenkins-public-ip:8080/sonarqube-webhook/
```



The screenshot shows the SonarQube administration interface for webhooks. The main navigation bar includes links for Gmail, YouTube, Amazon Web Servic..., Coaching on DevO..., LinkedIn, Docker — A Beginn..., Cloud Quest, 100+ Essential Com..., Advanced End-to-E..., LINUX - YouTube T..., and How to Install Jenk... . The top navigation bar also has tabs for sonarqube, Projects, Issues, Rules, Quality Profiles, Quality Gates, and Administration. A search bar is present at the top right. The main content area is titled "Administration" and "Webhooks". It explains what webhooks are used for and provides a link to the documentation. Below this, it says "No webhook defined." and shows the "Create Webhook" dialog box. The dialog box has fields for "Name" (jenkins), "URL" (http://43.204.36.242:8090/sonarqube-webhook/), and "Secret" (empty). A note about using a secret for HMAC hex digest generation is present. At the bottom of the dialog is a "Create" button.

Administration

Configuration ▾ Security ▾ Projects ▾ Rules ▾ Quality Profiles ▾ Quality Gates ▾ Administration

Search for projects...

Webhooks

Webhooks are used to notify external services when a project analysis is completed. You can define one or more URLs to each of the provided URLs. Learn more in the [Webhooks documentation](#).

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.

Create Cancel

Let's go to our Pipeline and add the script in our Pipeline Script.

```

-Dsonar.projectKey=zomato ''
}

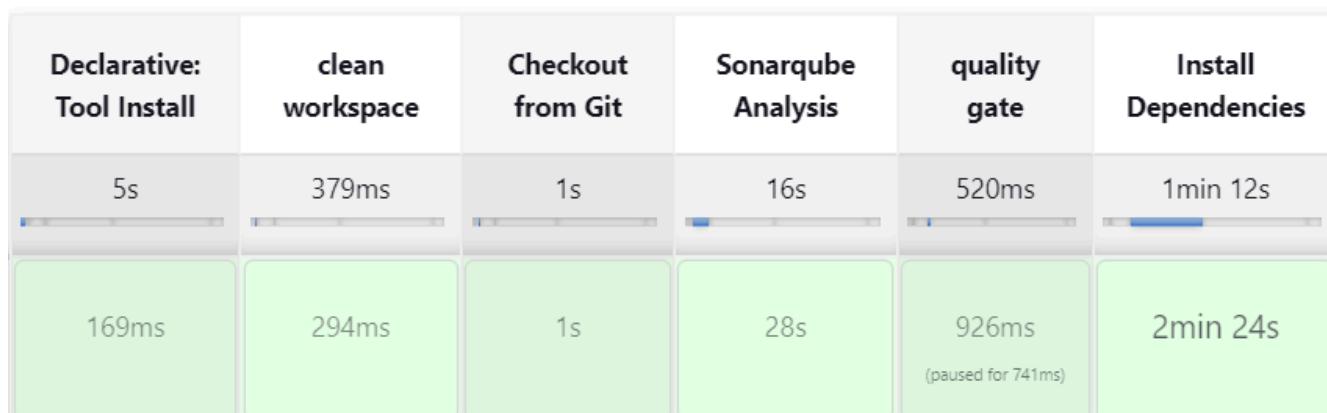
}

stage("quality gate"){
    steps {
        script {
            waitForQualityGate abortPipeline: false, credential:
        }
    }
}

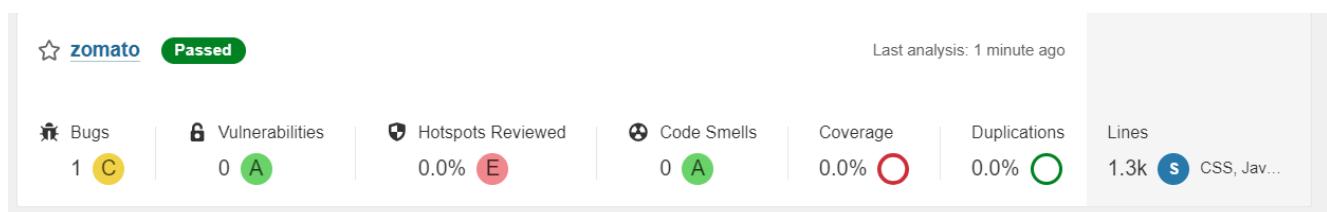
stage('Install Dependencies') {
    steps {
        sh "npm install"
    }
}
}
}

```

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



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



You can see that there are 1.3k lines. To see a detailed report, you can go to issues.

Step 5 – Install OWASP Dependency Check Plugins

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

The screenshot shows the Jenkins 'Plugins' page. On the left, there's a sidebar with links: 'Updates', 'Available plugins' (which is selected and highlighted in grey), 'Installed plugins', 'Advanced settings', and 'Download progress'. The main area has a search bar with placeholder text 'Search available plugins' and a button with a magnifying glass icon. Below the search bar, there are two tabs: 'Install' and 'Name ↴'. A list of plugins is displayed, with 'OWASP Dependency-Check 5.4.2' selected. This item has a blue checkmark icon, the name 'OWASP Dependency-Check 5.4.2', the category 'Security', and sub-categories 'DevOps', 'Build Tools', and 'Build Reports'. A small description below says: '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.' To the right of the description, it says 'Released 8 days 17 hr ago'. At the top right of the main area, there are 'Install' and 'Cancel' buttons.

First, we configured the Plugin and next, we had to configure the Tool

Goto Dashboard → Manage Jenkins → Tools →

The screenshot shows the Jenkins 'Tools' page. In the top navigation bar, under 'Manage Jenkins', there are links: 'Dashboard', 'Manage Jenkins', and 'Tools'. The main content area is titled 'Dependency-Check installations'. It features a 'Add Dependency-Check' button. Below it, there's a section for 'Dependency-Check' with a 'Name' field containing 'DP-Check' and an 'Install automatically' checkbox which is checked. Underneath this, there's a dashed-line box labeled 'Install from github.com' with a 'Version' field containing 'dependency-check 6.5.1' and a 'Add Installer' button.

Click on Apply and Save here.

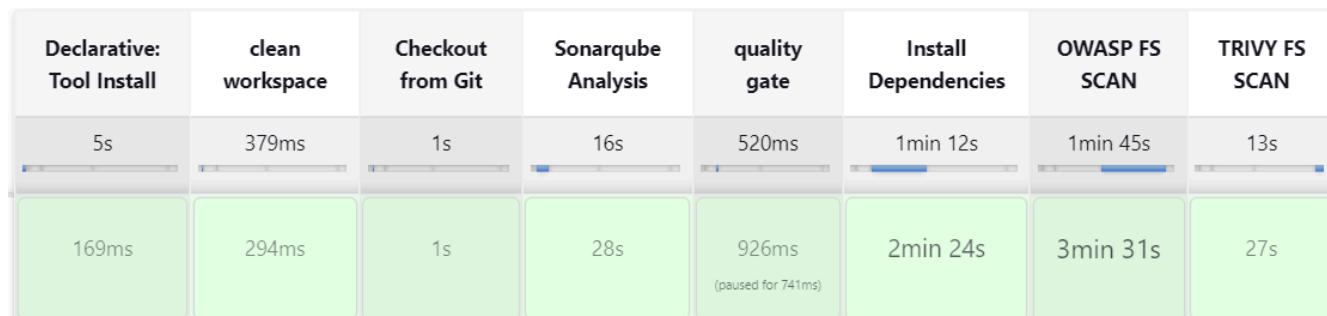
Now go configure → Pipeline and add this stage to your pipeline and build.



```
stage('OWASP FS SCAN') {
    steps {
        dependencyCheck additionalArguments: '--scan ./ --disable-scan-exclude-pattern=src/main/resources'
        dependencyCheckPublisher pattern: '**/dependency-check-report.xml'
    }
}
stage('TRIVY FS SCAN') {
    steps {
        sh "trivy fs . > trivyfs.txt"
    }
}
```



The stage view would look like this,



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

Dependency-Check Results

SEVERITY DISTRIBUTION

5

8

3

Search



File Name	Vulnerability	Severity	Weakness
+ css-what:3.4.2	OSSINDEX CVE-2022-21222	High	CWE-1333
+ ejs:3.1.8	OSSINDEX CVE-2023-29827	High	CWE-74
+ json5:1.0.1	NVD CVE-2022-46175	High	CWE-1321
+ jsonpointer:5.0.1	NVD CVE-2022-4742	Critical	CWE-1321
+ nth-check:1.0.2	NVD CVE-2021-3803	High	CWE-1333
+ parseurl:1.3.3	NVD CVE-2022-0722	High	CWE-200
+ parseurl:1.3.3	NVD CVE-2022-2216	Critical	CWE-918
+ parseurl:1.3.3	NVD CVE-2022-2217	Medium	CWE-79
+ parseurl:1.3.3	NVD CVE-2022-2218	Medium	CWE-79
+ parseurl:1.3.3	NVD CVE-2022-2900	Critical	CWE-918

Step 6 – 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

The screenshot shows the Jenkins Plugins page with a search bar containing "docker". Three results are listed:

- Docker 1.5**: This plugin integrates Jenkins with Docker. It is up for adoption and was released 3 days 15 hr ago.
- Docker Commons 439.va_3cb_0a_6a_fb_29**: Provides the common shared functionality for various Docker-related plugins. It was released 1 mo 29 days ago.
- Docker Pipeline 572.v950f58993843**: Build and use Docker containers from pipelines. It is up for adoption and was released 27 days ago.
- Docker API 3.3.1-79.v20b_53427e041**: This plugin provides docker-JAVA API for other plugins. It was released 3 mo 4 days ago.

Now, goto Dashboard → Manage Jenkins → Tools →

Docker installations

Add Docker

Docker

Name: docker

Install automatically ?

Download from docker.com

Docker version: latest

Add Installer ▾

Add DockerHub Username and Password under Global Credentials

Kind

Username with password

Scope

Global (Jenkins, nodes, items, all child items, etc)

Username

sevenajay

 Treat username as secret

Password

ID

docker

Description

docker

Create

Add this stage to Pipeline Script



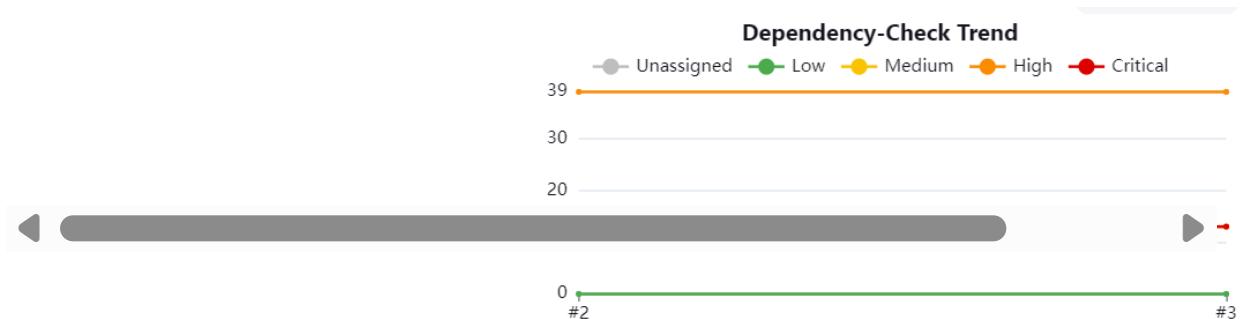
```
stage("Docker Build & Push"){
    steps{
        script{
            withDockerRegistry(credentialsId: 'docker', toolName
                sh "docker build -t zomato ."
                sh "docker tag zomato sevenajay/zomato:latest"
                sh "docker push sevenajay/zomato:latest"
            }
        }
    }
}
```

```

    }
    stage("TRIVY"){
        steps{
            sh "trivy image sevenajay/zomato:latest > trivy.txt"
        }
    }
}

```

You will see the output below, with a dependency trend.



Declarative: Tool Install	clean workspace	Checkout from Git	Sonarqube Analysis	quality gate	Install Dependencies	OWASP FS SCAN	TRIVY FS SCAN	Docker Build & Push	TRIVY
3s	366ms	1s	19s	451ms	1min 20s	2min 1s	16s	3min 9s	4s
154ms	341ms	1s	25s	315ms	1min 36s	2min 31s	23s	3min 9s	4s

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

Now Run the container to see if the app coming up or not by adding the below stage

```

stage('Deploy to container'){
    steps{
        sh 'docker run -d --name zomato -p 3000:3000 sevenajay/zoma1
    }
}

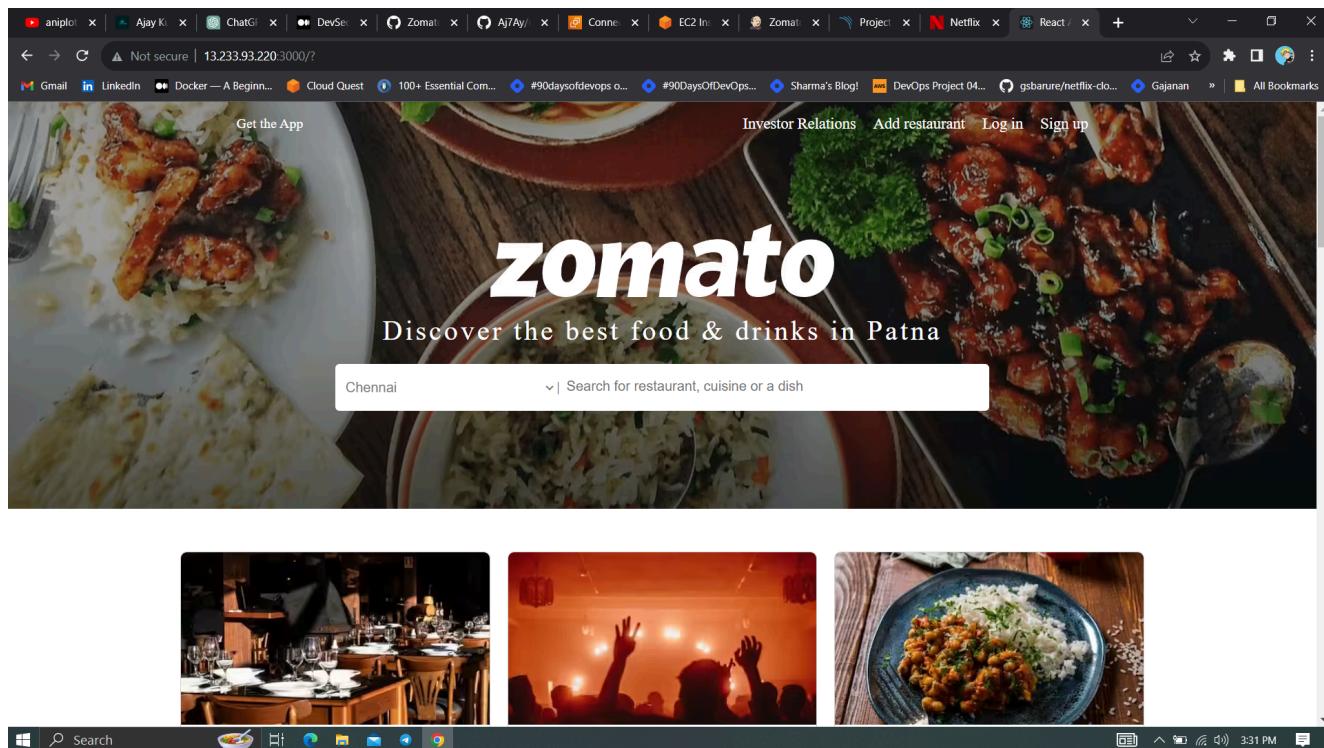
```

stage view

Declarative: Tool Install	clean workspace	Checkout from Git	Sonarqube Analysis	quality gate	Install Dependencies	OWASP FS SCAN	TRIVY FS SCAN	Docker Build & Push	TRIVY	Deploy to container
144ms	284ms	1s	25s	410ms	1min 47s	2min 43s	23s	2min 7s	36s	789ms
146ms	251ms	1s	26s	305ms	1min 36s	2min 35s	23s	1min 50s	2min 8s	1s

<Jenkins-public-ip:3000>

You will get this output



Step 8: Terminate instances.

Complete Pipeline



```
pipeline{
    agent any
    tools{
        jdk 'jdk17'
```

```

        nodejs 'node16'
    }
environment {
    SCANNER_HOME=tool 'sonar-scanner'
}
stages {
    stage('clean workspace'){
        steps{
            cleanWs()
        }
    }
    stage('Checkout from Git'){
        steps{
            git branch: 'main', url: 'https://github.com/Aj7Ay/Zoma1
        }
    }
    stage("Sonarqube Analysis"){
        steps{
            withSonarQubeEnv('sonar-server') {
                sh ''' $SCANNER_HOME/bin/sonar-scanner -Dsonar.proje
-Dsonar.projectKey=zomato '''
            }
        }
    }
    stage("quality gate"){
        steps {
            script {
                waitForQualityGate abortPipeline: false, credential:
            }
        }
    }
    stage('Install Dependencies') {
        steps {
            sh "npm install"
        }
    }
    stage('OWASP FS SCAN') {
        steps {
            dependencyCheck additionalArguments: '--scan ./ --disab
            dependencyCheckPublisher pattern: '**/dependency-check-i
        }
    }
    stage('TRIVY FS SCAN') {
        steps {
            sh "trivy fs . > trivyfs.txt"
        }
    }
}

```

```
        }
    }
    stage("Docker Build & Push"){
        steps{
            script{
                withDockerRegistry(credentialsId: 'docker', toolName
                    sh "docker build -t zomato ."
                    sh "docker tag zomato sevenajay/zomato:latest "
                    sh "docker push sevenajay/zomato:latest "
                }
            }
        }
    }
    stage("TRIVY"){
        steps{
            sh "trivy image sevenajay/zomato:latest > trivy.txt"
        }
    }
    stage('Deploy to container'){
        steps{
            sh 'docker run -d --name zomato -p 3000:3000 sevenajay/zomato'
        }
    }
}
}
```



Ajay Kumar Yegireddi is a DevSecOps Engineer and System Administrator, with a passion for sharing real-world DevSecOps projects and tasks. **Mr. Cloud Book**, provides hands-on tutorials and practical insights to help others master DevSecOps tools and workflows. Content is designed to bridge the gap between development, security, and operations, making complex concepts easy to understand for both beginners and professionals.

Comments

3 responses to “Zomato Clone App with DevSecOps CI/CD”



sarvadnya
19 June 2024

Ajay really amazing blog buddy, with detailed explaination on each step, i successfully implemented this project, soon i will share about this on linkedin, you are doing really hard work buddy really i appreaciate this. one last thing i want to ask you i want to learn to write those pipeline syntax, how can write pipelines such as this one. any tips ?

[Reply](#)



Abhinandan Vishwakarma
27 November 2024

Can Your deploye with database type application which is connect with database and for user suppose logged in application so that related deploy session store pod and any thing which is required so real time deploy the app in CICD process

Can u do this for real world examples

[Reply](#)



Sathvika Iella
4 December 2024

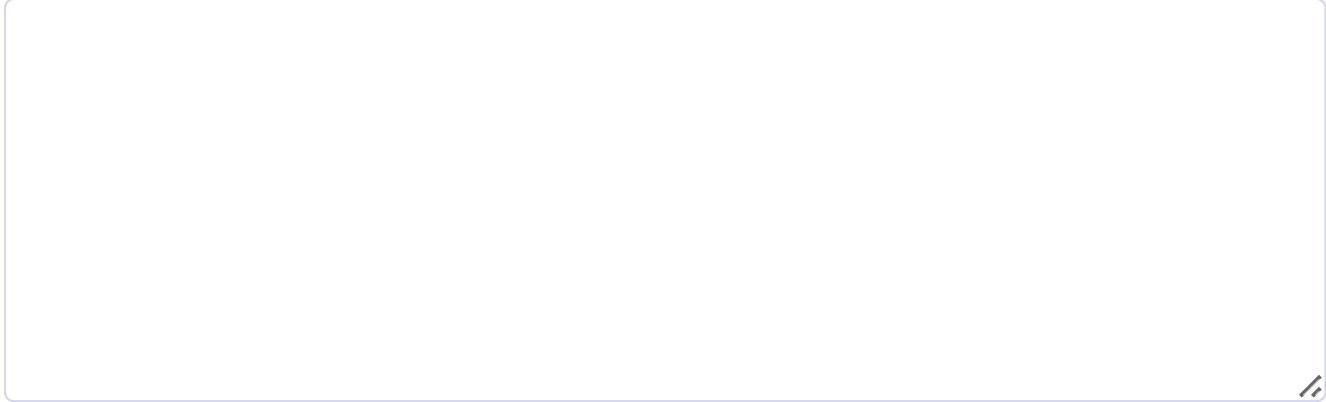
This is very simple project any one can do with this steps

[Reply](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment *



Name *

Email *

Website

Save my name, email, and website in this browser for the next time I comment.



I'm not a robot

reCAPTCHA
[Privacy](#) - [Terms](#)

Post Comment

Uncategorized

How to Automate Incident Response : How Q Developer Helped Me Automate a Daily Pain Point

22 July 2025

AI

How to Run Docker Model Runner on Ubuntu 24.04

11 July 2025

AI, DevOps

How to Install docker-ai on Ubuntu 24.04

15 June 2025

Upskill with Ajay: DevSecOps Mastery

Join Mr Cloud book to master DevSecOps through real-world projects. Learn CI/CD, security integration, automation, and more, gaining hands-on skills for industry-level challenges.



Important Links

[Privacy Policy](#)

[Terms & Conditions](#)

[Contact](#)

Resources

[Blog](#)

[YouTube Channel](#)

© 2024 · Powered by [Mr Cloud Book](#)

[Follow Us on YouTube](#)