04/02/2025

CI/CD pipeline flow

Docker

Docker is an open-source platform enabling developers to build, package, and run applications in isolated containers. It allows applications to run consistently across different environments, eliminating the "it works on my machine" problem.

Docker Image

*A **Docker image** is a **lightweight, standalone, and executable package** that includes everything needed to run an application—code, runtime, libraries, and dependencies.*

Docker Container

A Docker container is a running instance of a Docker image. It is an isolated environment that runs the application.

Jenkins

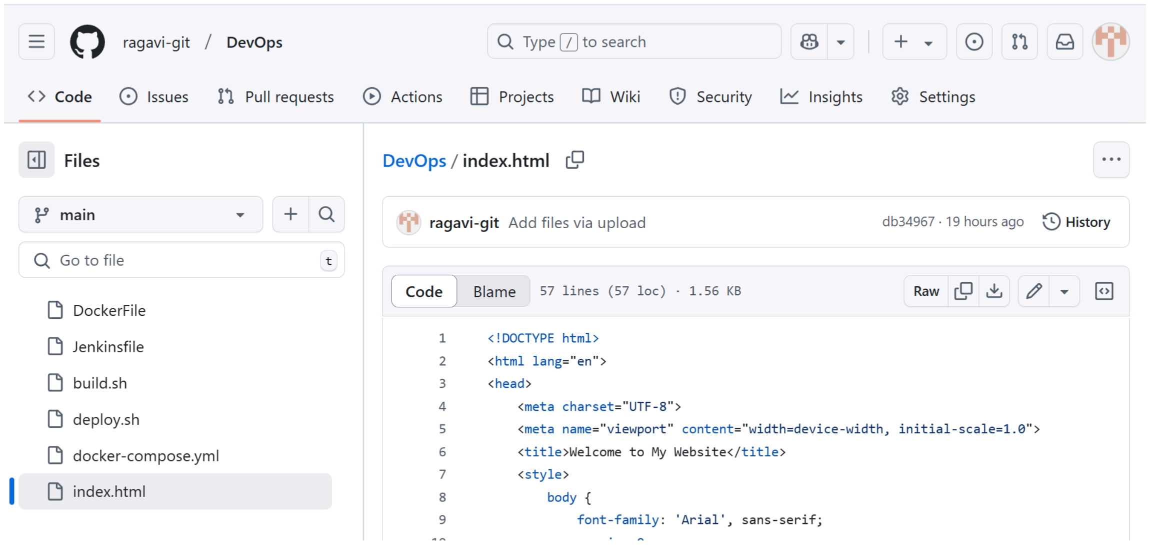
Jenkins is an automation server used for continuous integration and continuous deployment (CI/CD).

- ➢ *Pulling code from repositories (GitHub, GitLab).*
- ➢ *Building Docker images.*
- ➢ *Running tests.*
- ➢ *Deploying applications.*

1. Code Repository Setup

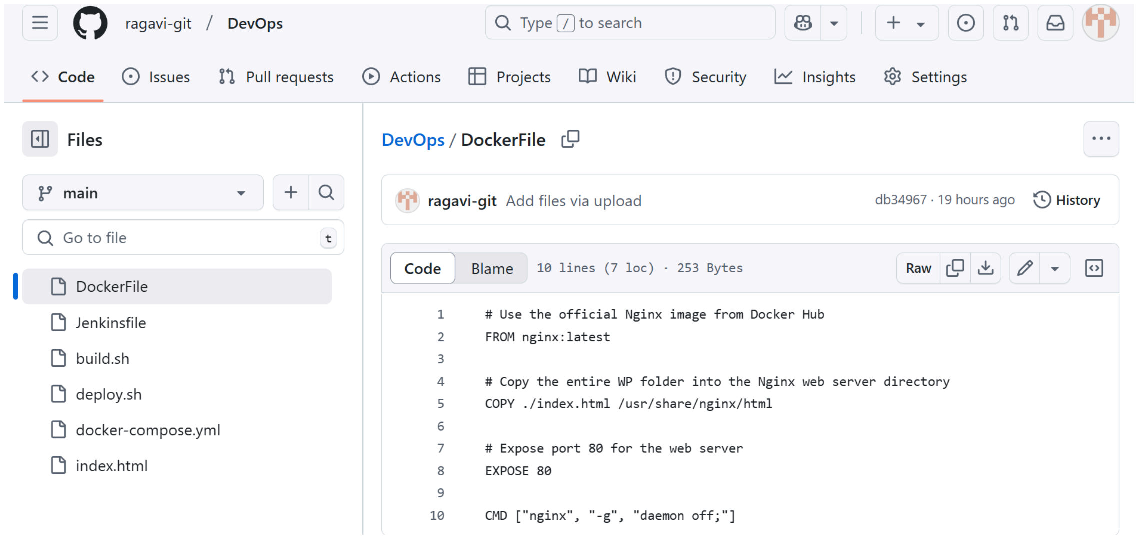
04/02/2025

- *Store project source code in a GitHub repository.*



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Welcome to My Website</title>
    <style>
        body {
            font-family: 'Arial', sans-serif;
```

- *Create a **Dockerfile** in the repository to define the application environment.*

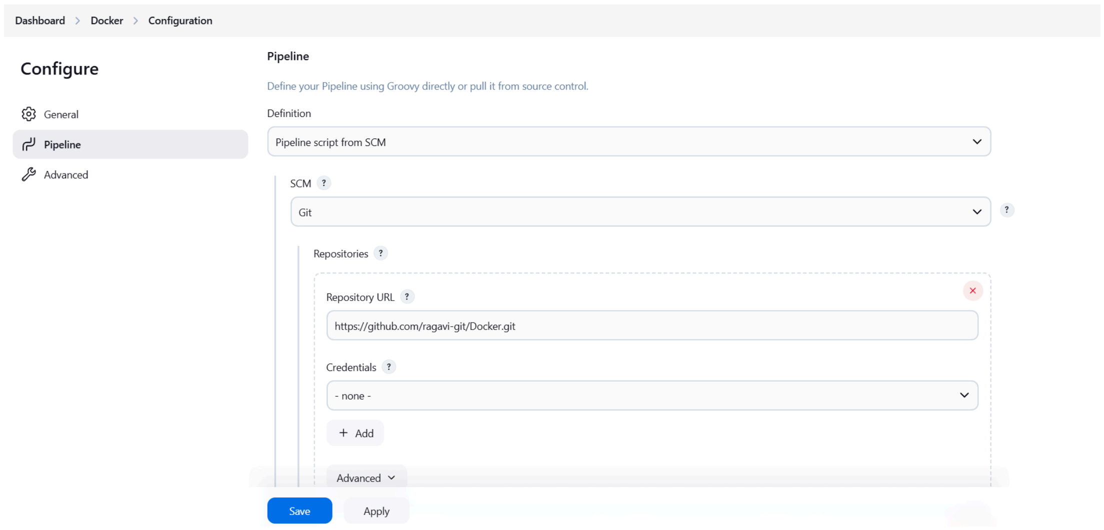


```
# Use the official Nginx image from Docker Hub
FROM nginx:latest

# Copy the entire WP folder into the Nginx web server directory
COPY ./index.html /usr/share/nginx/html

# Expose port 80 for the web server
EXPOSE 80

CMD ["nginx", "-g", "daemon off;"]
```

2. Build Docker Image

- *Use GitHub Actions or Jenkins to pull the code from GitHub.*

04/02/2025


Dashboard > Docker > Configuration

Configure

General
Pipeline
Advanced

Pipeline

Define your Pipeline using Groovy directly or pull it from source control.

Definition: Pipeline script from SCM

SCM: Git

Repositories

Repository URL: https://github.com/ragavi-git/Docker.git

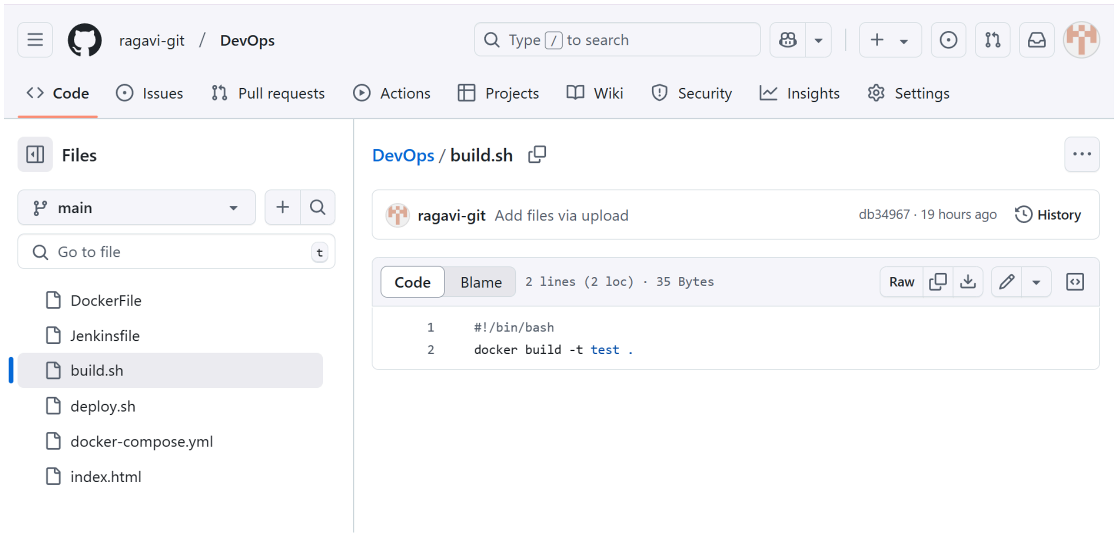
Credentials: - none -

+ Add

Advanced

Save Apply

- *Build a **Docker image** using the **Dockerfile**.*


ragavi-git / DevOps

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Files

main

DockerFile
Jenkinsfile
build.sh
deploy.sh
docker-compose.yml
index.html

DevOps / build.sh

ragavi-git Add files via upload db34967 · 19 hours ago History

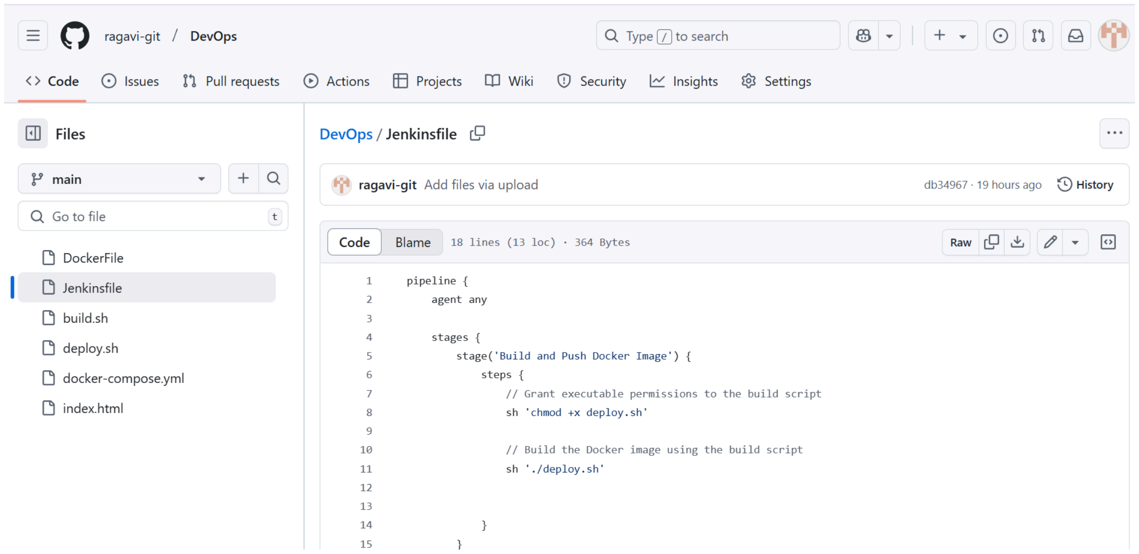
Code Blame 2 lines (2 loc) · 35 Bytes

```
#!/bin/bash
docker build -t test .
```

3. Push to Docker Hub

- *Tag the Docker image appropriately.*

04/02/2025

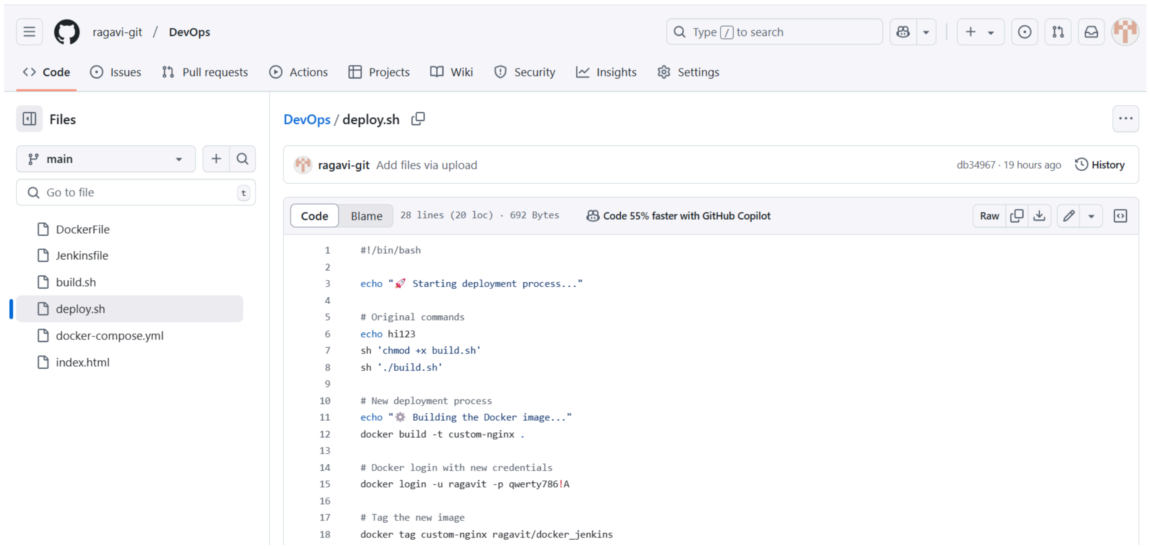

DevOps / Jenkinsfile

ragavi-git Add files via upload · db34967 · 19 hours ago

18 lines (13 loc) · 364 Bytes

```
pipeline {
    agent any

    stages {
        stage('Build and Push Docker Image') {
            steps {
                // Grant executable permissions to the build script
                sh 'chmod +x deploy.sh'

                // Build the Docker image using the build script
                sh './deploy.sh'


            }
        }
```

- *Push the image to* ***Docker Hub*** *for storage and deployment.*


DevOps / deploy.sh

ragavi-git Add files via upload · db34967 · 19 hours ago

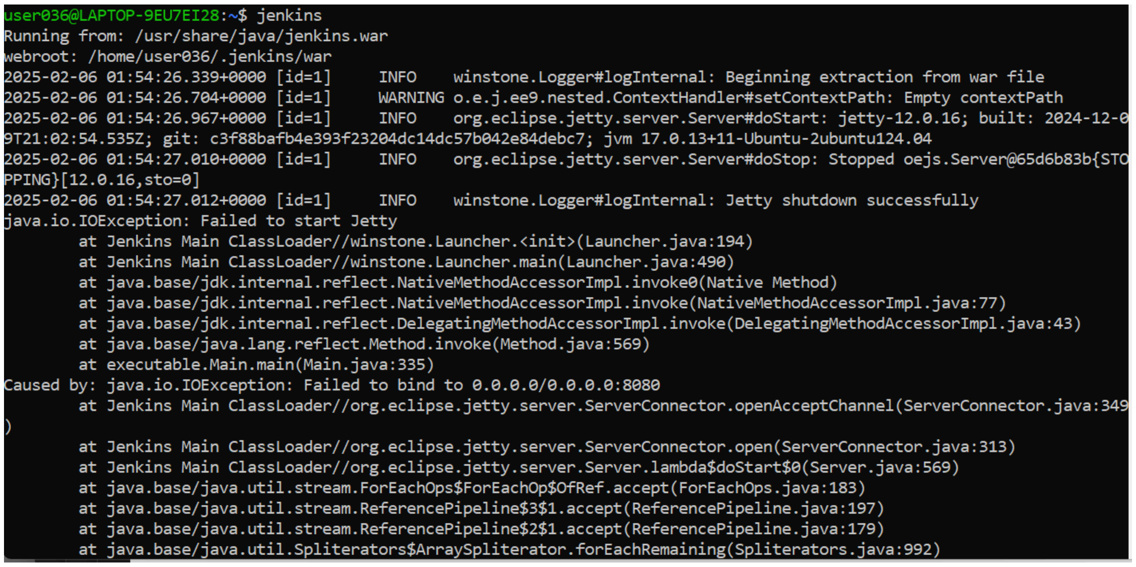
28 lines (20 loc) · 692 Bytes

```
#!/bin/bash

echo "🚀 Starting deployment process..."

# Original commands
echo hi123
sh 'chmod +x build.sh'
sh './build.sh'

# New deployment process
echo "⚙️ Building the Docker image..."
docker build -t custom-nginx .

# Docker login with new credentials
docker login -u ragavit -p qwerty786!A

# Tag the new image
docker tag custom-nginx ragavit/docker_jenkins
```

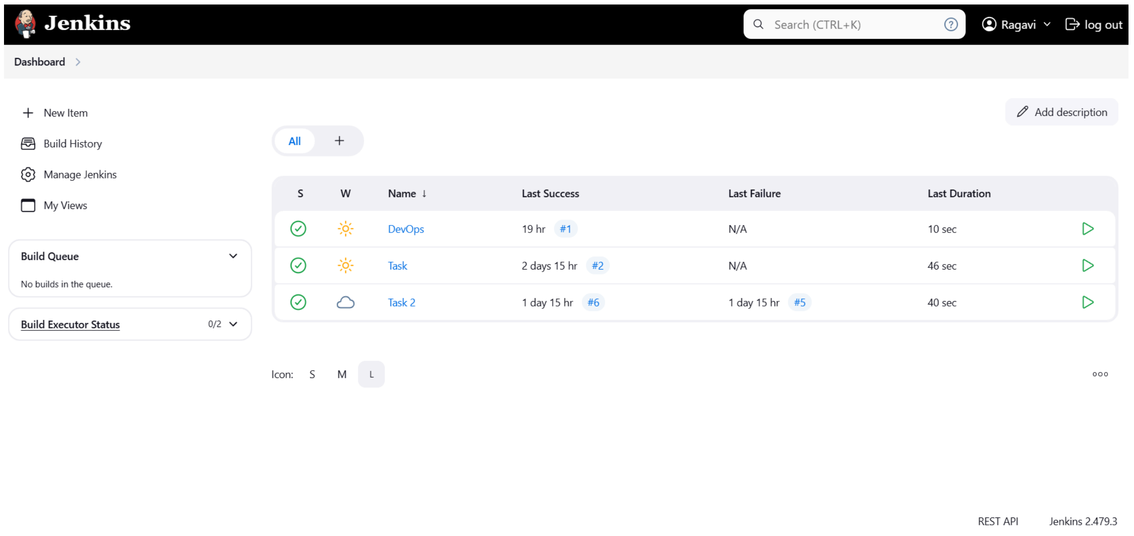
4. Jenkins Setup

- *Install and configure* ***Jenkins****.*

04/02/2025



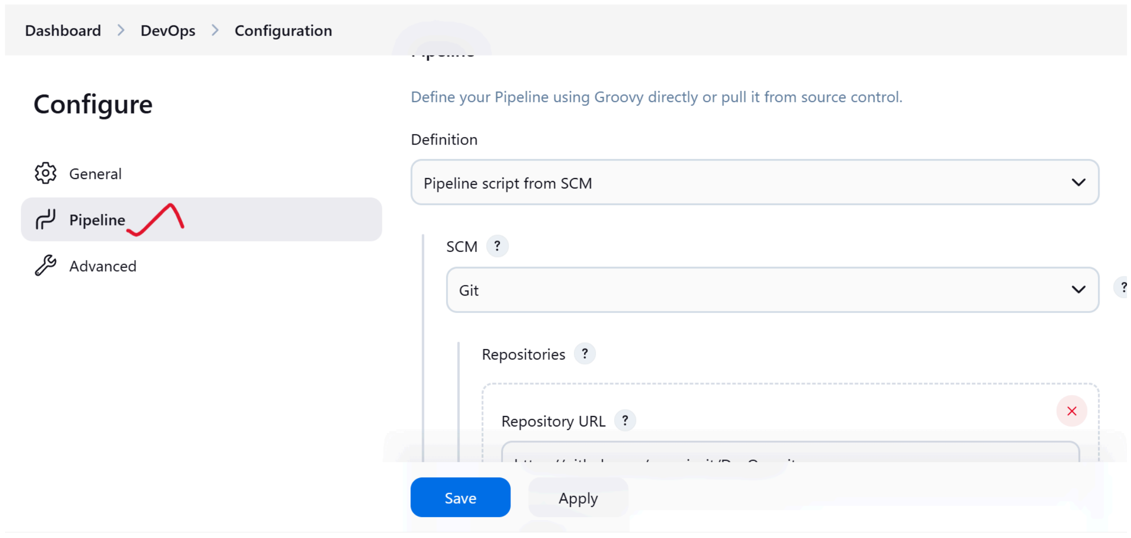
```
user036@LAPTOP-9EU7EI28:~$ jenkins
Running from: /usr/share/java/jenkins.war
webroot: /home/user036/.jenkins/war
2025-02-06 01:54:26.339+0000 [id=1]     INFO    winstone.Logger#logInternal: Beginning extraction from war file
2025-02-06 01:54:26.704+0000 [id=1]     WARNING o.e.j.ee9.nested.ContextHandler#setContextPath: Empty contextPath
2025-02-06 01:54:26.967+0000 [id=1]     INFO    org.eclipse.jetty.server.Server#doStart: jetty-12.0.16; built: 2024-12-0
9T21:02:54.535Z; git: c3f88bafb4e393f23204dc14dc57b042e84debc7; jvm 17.0.13+11-Ubuntu-2ubuntu124.04
2025-02-06 01:54:27.010+0000 [id=1]     INFO    org.eclipse.jetty.server.Server#doStop: Stopped oejs.Server@65d6b83b{STO
PPING}[12.0.16,sto=0]
2025-02-06 01:54:27.012+0000 [id=1]     INFO    winstone.Logger#logInternal: Jetty shutdown successfully
java.io.IOException: Failed to start Jetty
        at Jenkins Main ClassLoader//winstone.Launcher.<init>(Launcher.java:194)
        at Jenkins Main ClassLoader//winstone.Launcher.main(Launcher.java:490)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:77)
        at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.base/java.lang.reflect.Method.invoke(Method.java:569)
        at executable.Main.main(Main.java:335)
Caused by: java.io.IOException: Failed to bind to 0.0.0.0/0.0.0.0:8080
        at Jenkins Main ClassLoader//org.eclipse.jetty.server.ServerConnector.openAcceptChannel(ServerConnector.java:349
)

        at Jenkins Main ClassLoader//org.eclipse.jetty.server.ServerConnector.open(ServerConnector.java:313)
        at Jenkins Main ClassLoader//org.eclipse.jetty.server.Server.lambda$doStart$0(Server.java:569)
        at java.base/java.util.stream.ForEachOps$ForEachOp$OfRef.accept(ForEachOps.java:183)
        at java.base/java.util.stream.ReferencePipeline$3$1.accept(ReferencePipeline.java:197)
        at java.base/java.util.stream.ReferencePipeline$2$1.accept(ReferencePipeline.java:179)
        at java.base/java.util.Spliterators$ArraySpliterator.forEachRemaining(Spliterators.java:992)
```



S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☀	DevOps	19 hr #1	N/A	10 sec
✓	☀	Task	2 days 15 hr #2	N/A	46 sec
✓	☁	Task 2	1 day 15 hr #6	1 day 15 hr #5	40 sec

- *Set up a **Jenkins pipeline** to automate the **CI/CD** process.*

04/02/2025


Dashboard > DevOps > Configuration

Configure

General

Pipeline

Advanced

Define your Pipeline using Groovy directly or pull it from source control.

Definition: Pipeline script from SCM

SCM: Git

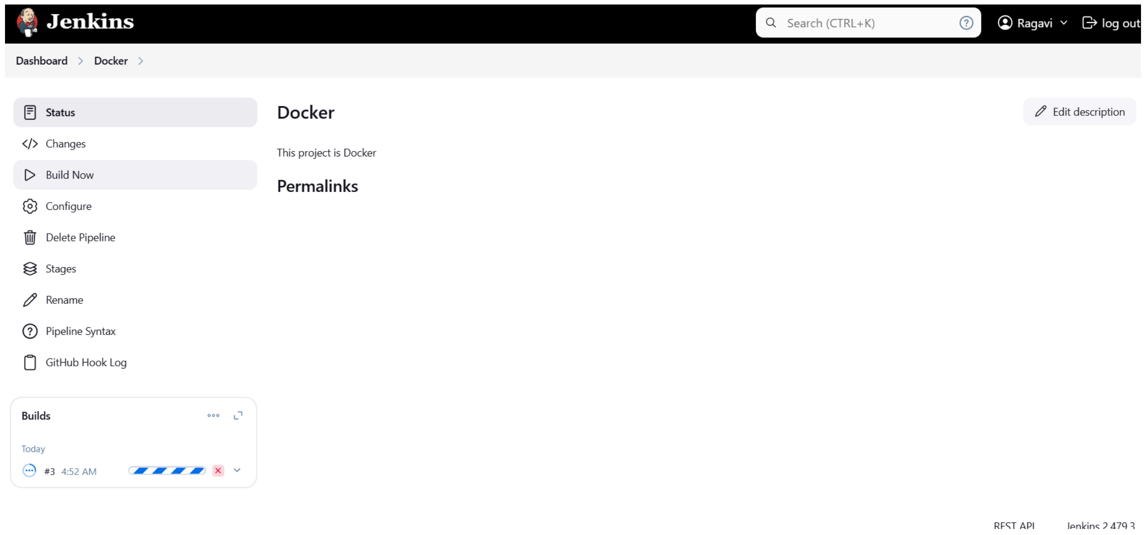
Repositories

Repository URL

Save Apply

5. CI/CD Pipeline Execution (Jenkins)

- *Jenkins pulls the latest code from GitHub.*
- *Builds the Docker image.*


Jenkins

Dashboard > Docker >

Status
Changes
Build Now
Configure
Delete Pipeline
Stages
Rename
Pipeline Syntax
GitHub Hook Log

Docker

This project is Docker

Permalinks

Edit description

Builds

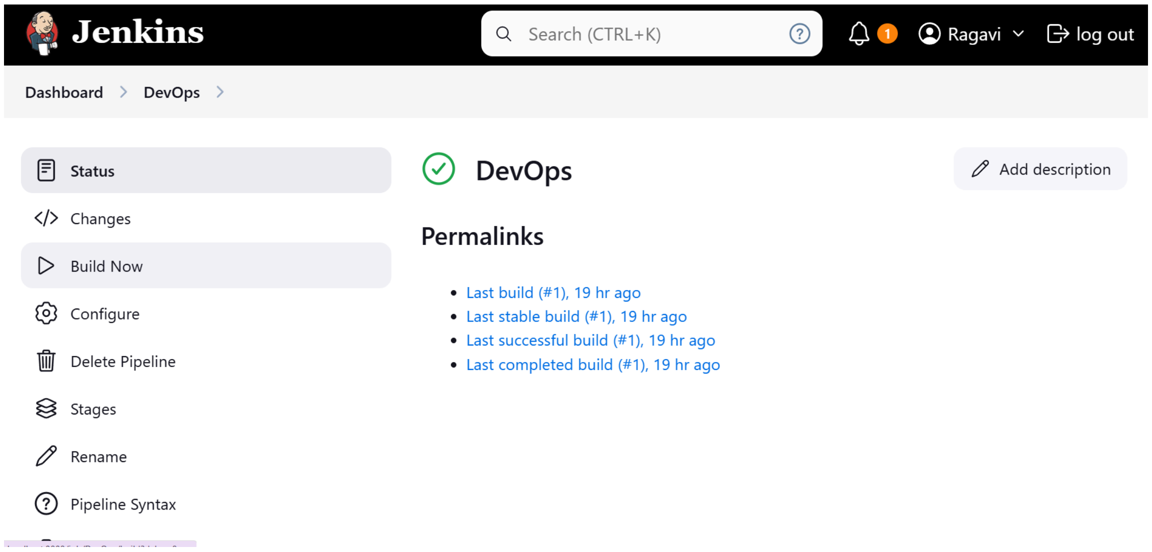
Today

#3 4:52 AM

REST API Jenkins 2.479.3

- *Pushes the built image to **Docker Hub**.*
- *Deploys the application using **Docker containers**.*

04/02/2025



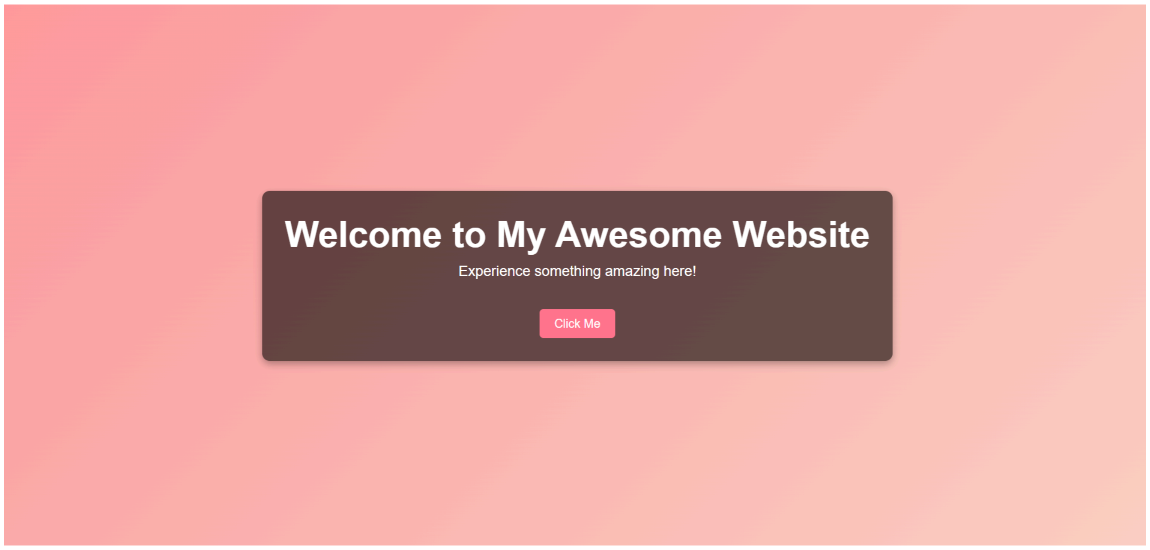
DevOps

Permalinks

- Last build (#1), 19 hr ago
- Last stable build (#1), 19 hr ago
- Last successful build (#1), 19 hr ago
- Last completed build (#1), 19 hr ago

6. Deployment

- *Jenkins pulls the latest Docker image from Docker Hub.*
- *Deploys the containerized application on a server (e.g. local machine).*


Welcome to My Awesome Website
Experience something amazing here!
Click Me

7. Continuous Integration and Deployment

- *Whenever there's a code change in GitHub, Jenkins automatically triggers the pipeline.*
- *The latest version of the application is built, pushed, and deployed seamlessly.*