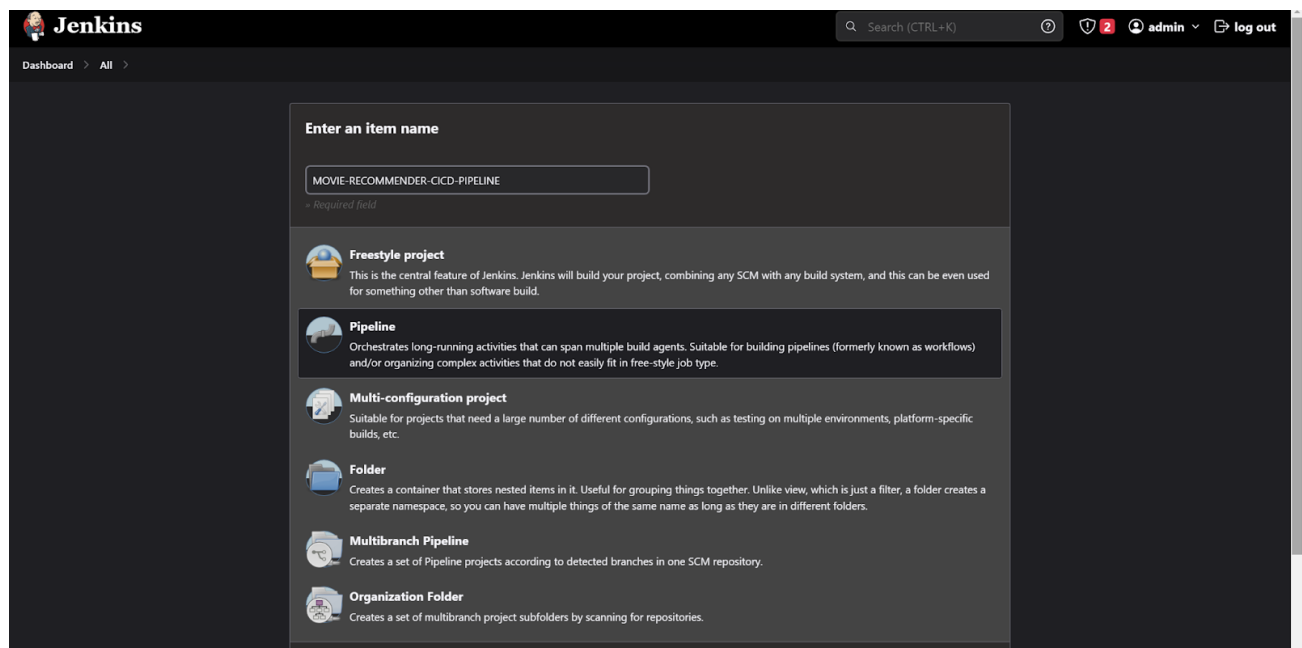
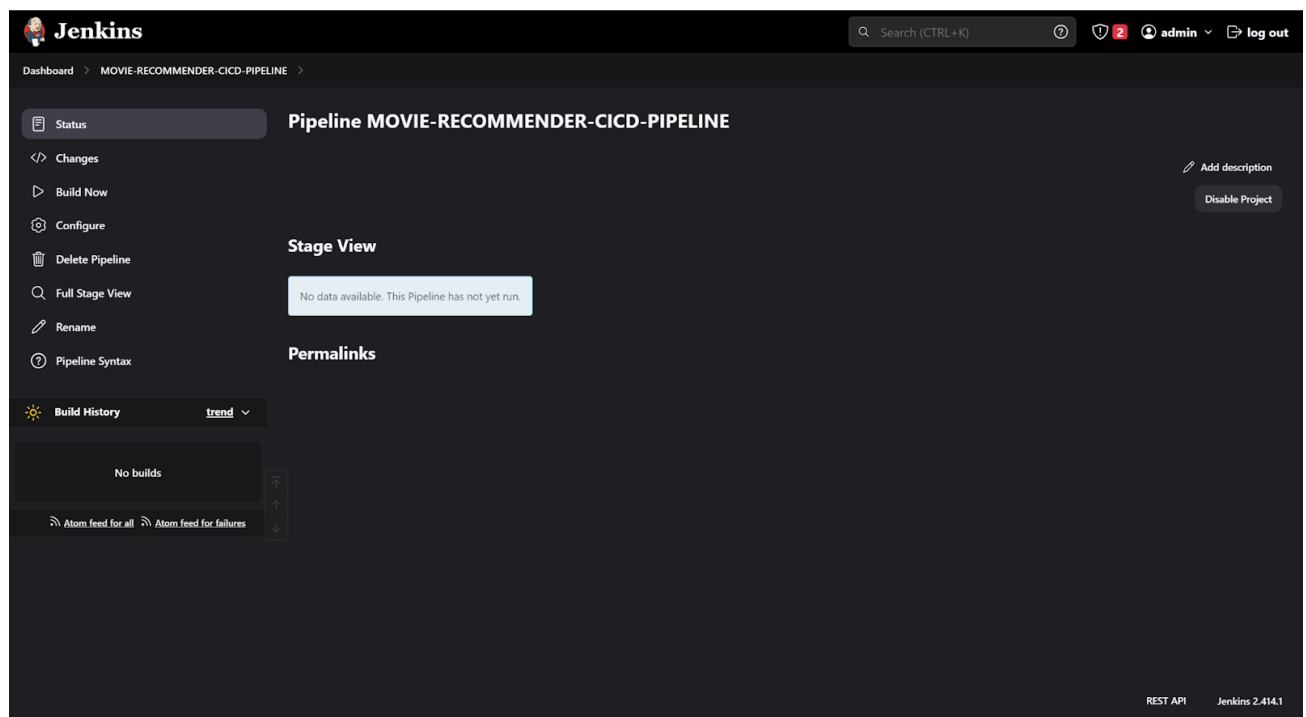


# CICD PIPELINE TO DEPLOY PYTHON APP

1.Created a pipeline having name “MOVIE-RECOMMENDER-CICD-PIPELINE”

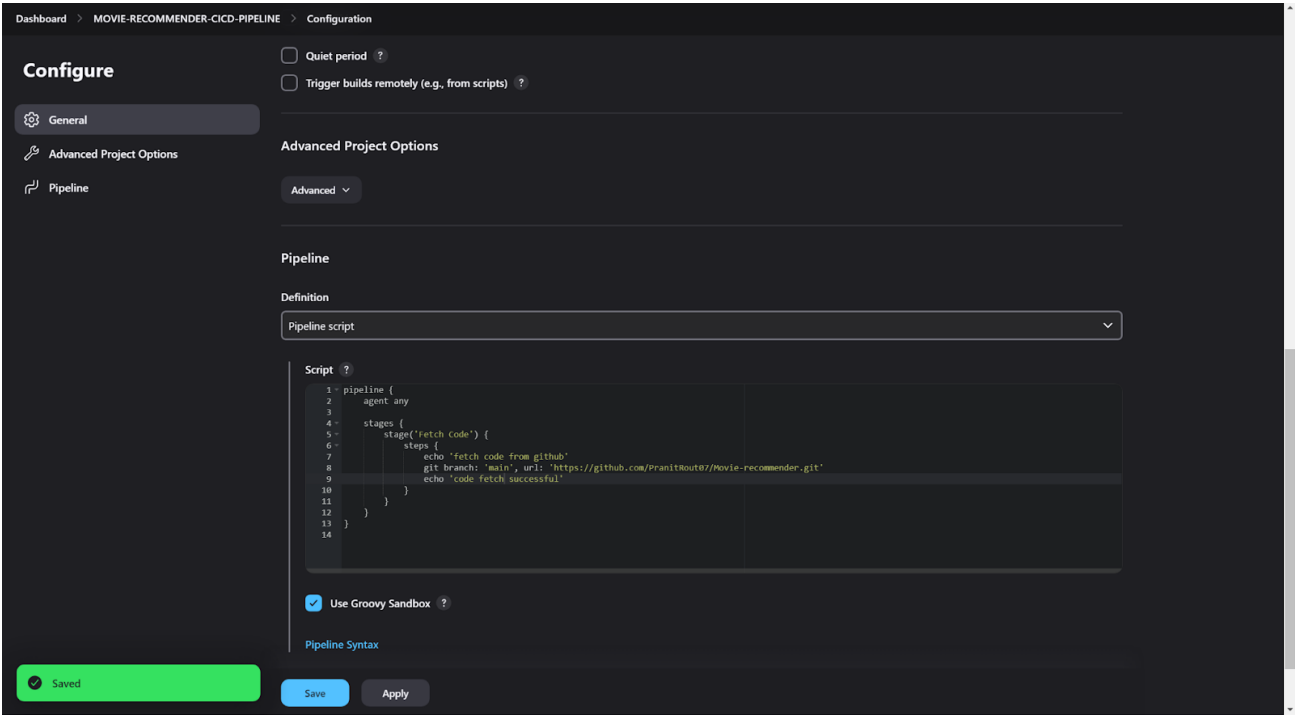


2.Now from this interface i would configure the pipeline and then build it.

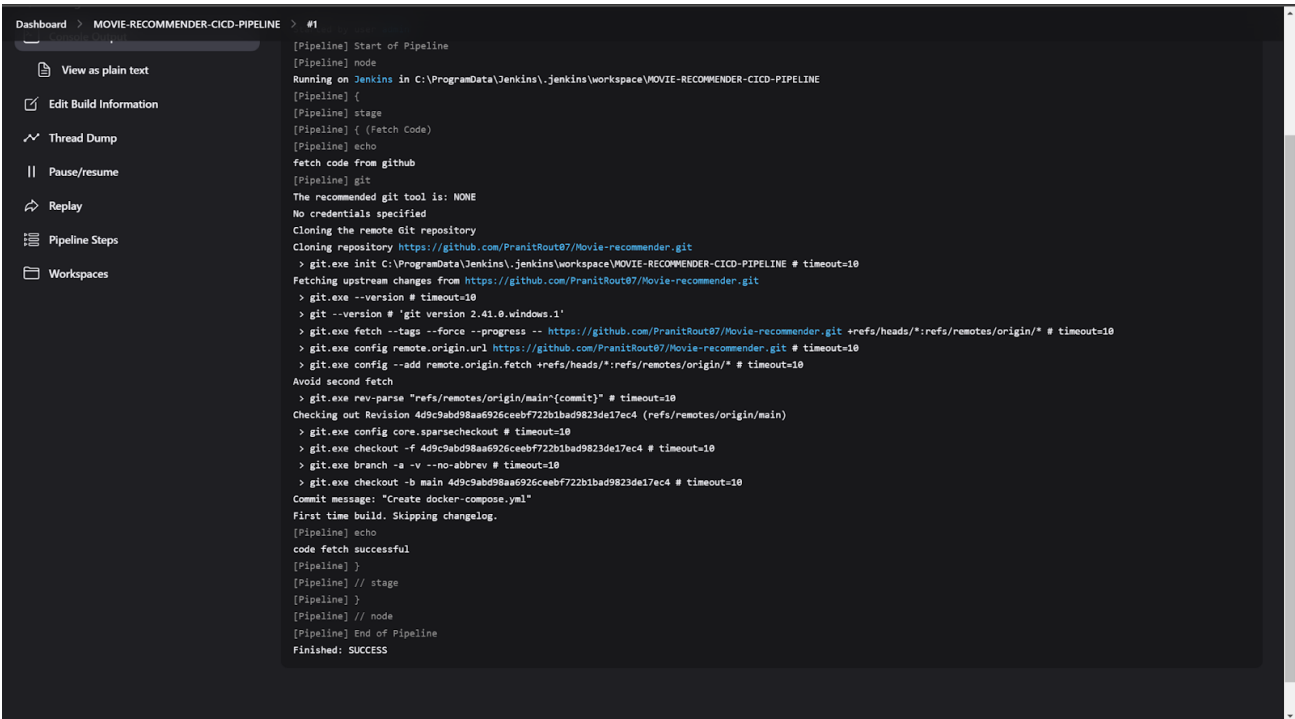


3.To build the entire CI/CD pipeline i have used declarative pipeline. Here in the first stage first code is fetched from the github.

git repository URL:-<https://github.com/PranitRout07/Movie-recommender.git>



4.After writing the pipeline i build the pipeline and observed the following output in the console output. Here the code is fetched successfully from the github.



5. Before writing this pipeline code for the stage 'Python code Analysis', first i run the sonarqube server locally in port 9000 and also installed sonarqube scanner pluggin in the jenkins. After that i have created a token on sonarqube server. Then in the jenkins's global credential pasted the copied token from the sonarqube server. After in manage jenkins>tools i have to give which version of sonarqube is going to install in jenkins. After that sonarqube server needed to be configured on jenkins. To do this task i have to go to the manage jenkins>system>system>sonarqube server. Here i have given the sonarqube url and added the credential through which jenkins is integrated with sonarqube. Also i have to configure the webhooks in the sonarqube server which will send http request to jenkins.

After all the above steps over then i have written the sonarqube pipeline, where i mentioned about project name, project key, project version and also mentioned the directory from where the sonarqube analysis will begin.

Dashboard > MOVIE-RECOMMENDER-CICD-PIPELINE > Configuration

### Configure

- General
- Advanced Project Options**
- Pipeline

#### Advanced Project Options

Advanced

#### Pipeline

Definition

Pipeline script

Script

```
9      echo 'code fetch successful'
10    }
11  }
12  stage('Python Code Analysis'){
13    environment {
14      scannerName = tool 'sonar4.7'
15    }
16    steps {
17      withSonarQubeEnv('sonar') {
18        bat "Zacamer\\tools\\sonar-scanner -Dsonar.projectKey=1 -Dsonar.projectName=MOVIE-RECOMMENDER -Dsonar.projectVersion=1.0 -Dsonar.sources=."
19      }
20    }
21  }
22  }
23  }
24  }
25  }
```

☒ Use Groovy Sandbox

Pipeline Syntax

Save Apply

REST API Jenkins 2.414.1

6.Here python code analysis is also completed successfully.

Jenkins

Search (CTRL+K)

admin

log out

Dashboard

MOVIE-RECOMMENDER-CICD-PIPELINE

Status

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

Rename

Pipeline Syntax

Build History

trend

Filter builds...

#2

Sep 11, 2023, 8:21 PM

#1

Sep 11, 2023, 8:11 PM

Atom feed for all

Atom feed for failures

Pipeline MOVIE-RECOMMENDER-CICD-PIPELINE

Add description

Disable Project

Stage View

Average stage times:  
(Average full run time: ~2min 30s)

Fetch Code

Python Code Analysis

2min 22s

13s

1s

13s

4min 43s

Permalinks

Last build (#1), 10 min ago

Last stable build (#1), 10 min ago

Last successful build (#1), 10 min ago

Last completed build (#1), 10 min ago

7.Here the report generated in sonarqube server. My code is successfully passed the test.

sonarqube

Projects

Issues

Rules

Quality Profiles

Quality Gates

Administration

MOVIE-RECOMMENDER

master

September 11, 2023 at 8:22 PM

Version 1.0

Overview

Issues

Security Hotspots

Measures

Code

Activity

Project Settings

Project Information

QUALITY GATE STATUS

Passed

All conditions passed.

MEASURES

New Code

Overall Code

0 Bugs

Reliability

A

0 Vulnerabilities

Security

A

0 Security Hotspots

Reviewed

Security Review

A

0 Debt

0 Code Smells

Maintainability

A

0.0%

Coverage on 46 Lines to cover

Unit Tests

0.0%

Duplications on 164 Lines

Duplicated Blocks

0

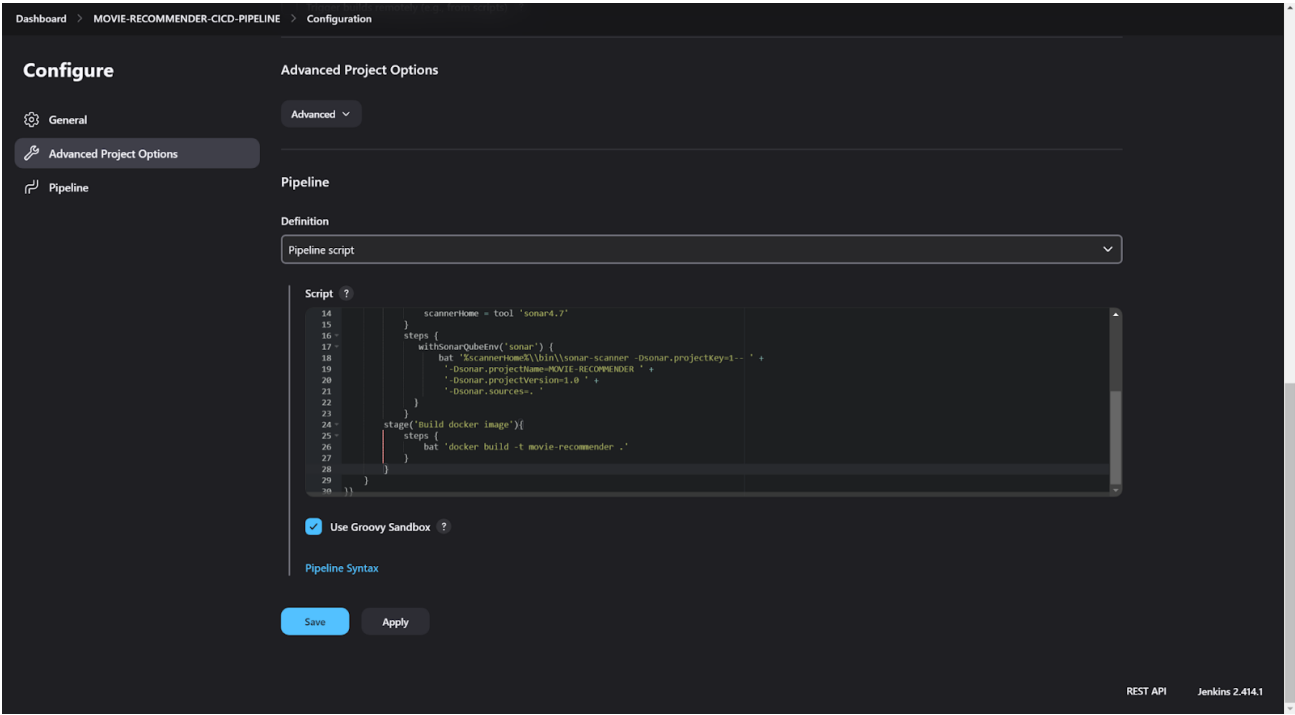
ACTIVITY

Issues

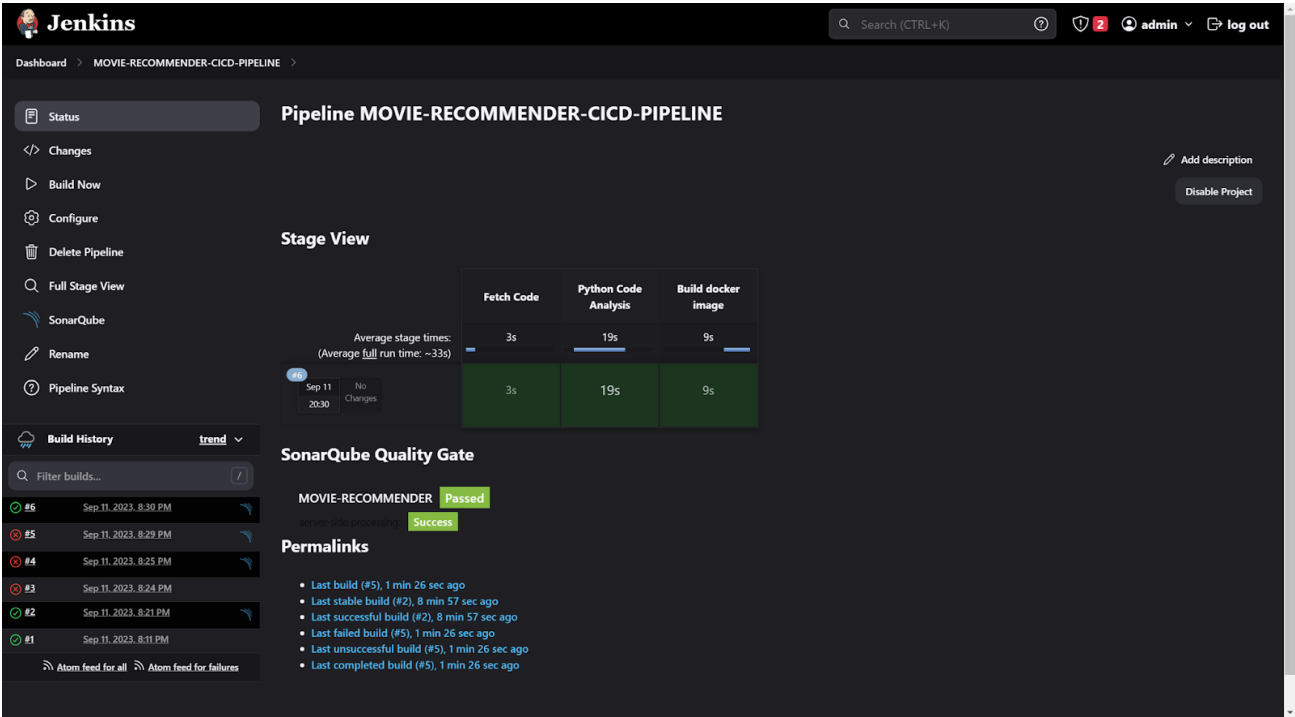
September 11, 2023 at 8:22 PM

1.0

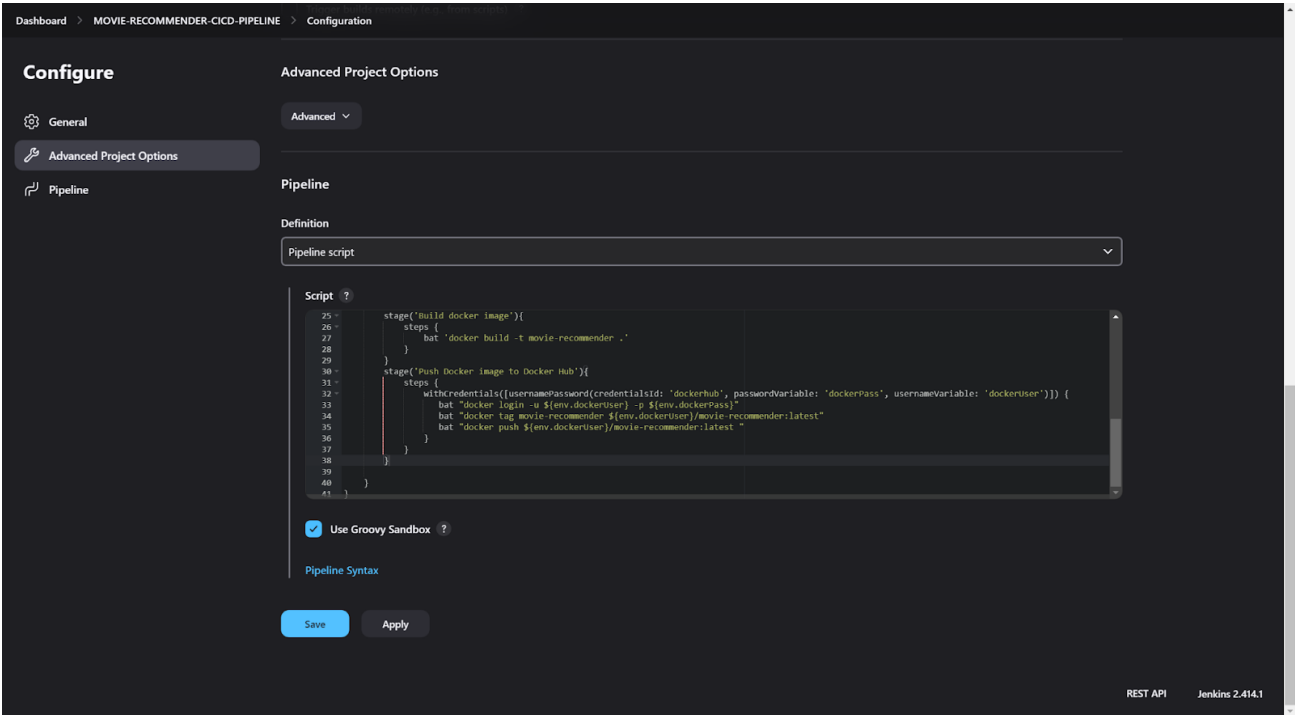
8.Now i have to build the docker image. In the background i run the docker engine then i build this step.



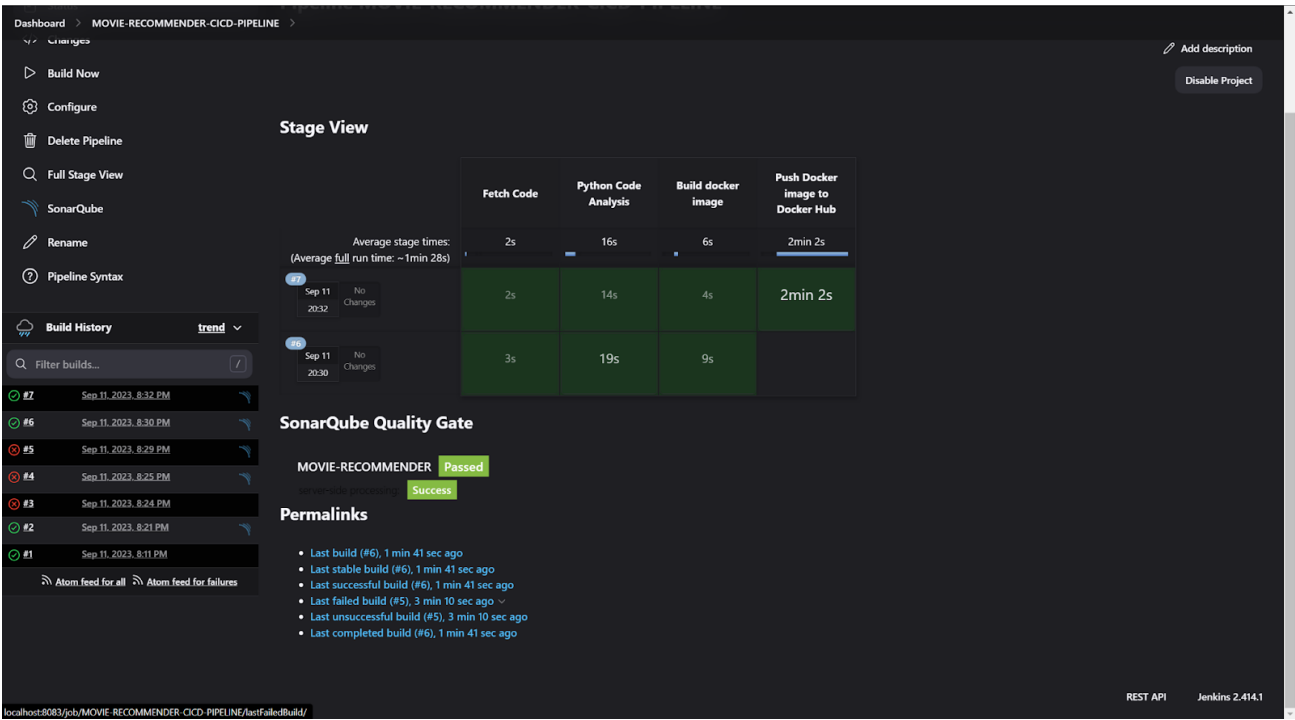
9.Docker image is successfully created.



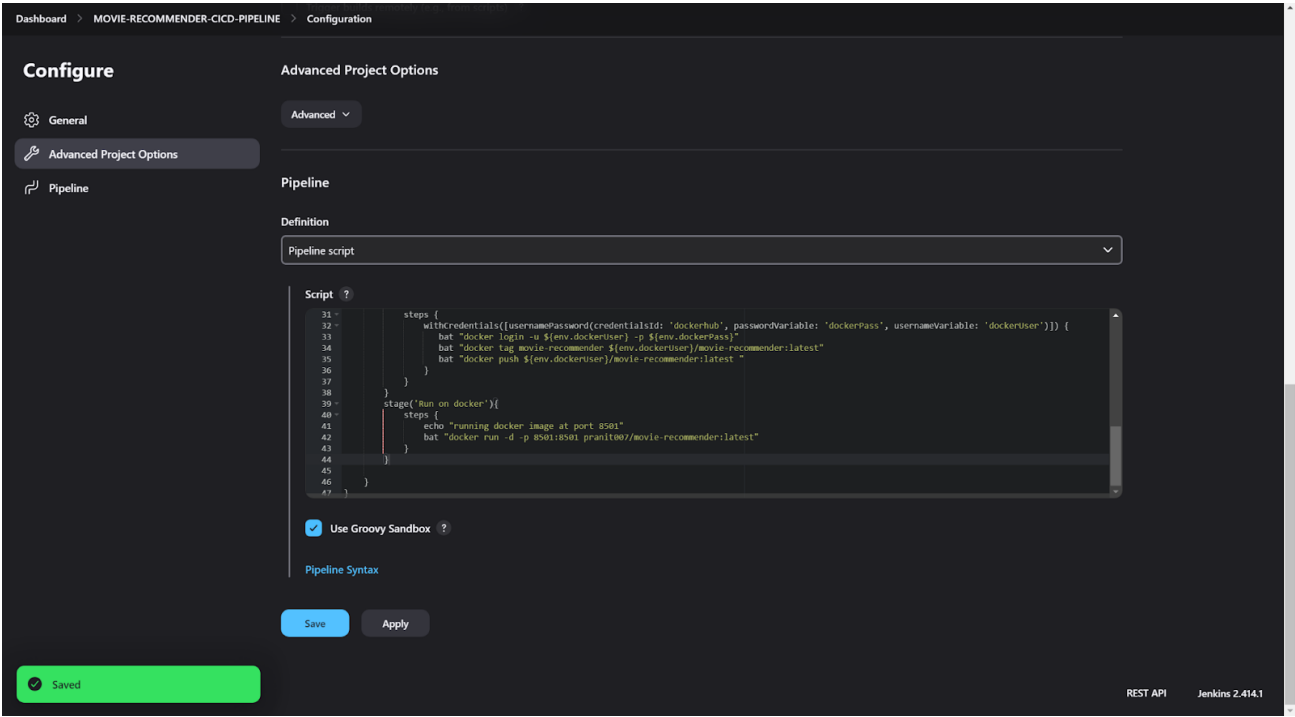
10.Now i have to push the docker image to the docker hub. To do this i have to add credential of docker hub to the global credential of jenkins. Then here i pushed the docker image to my docker hub.



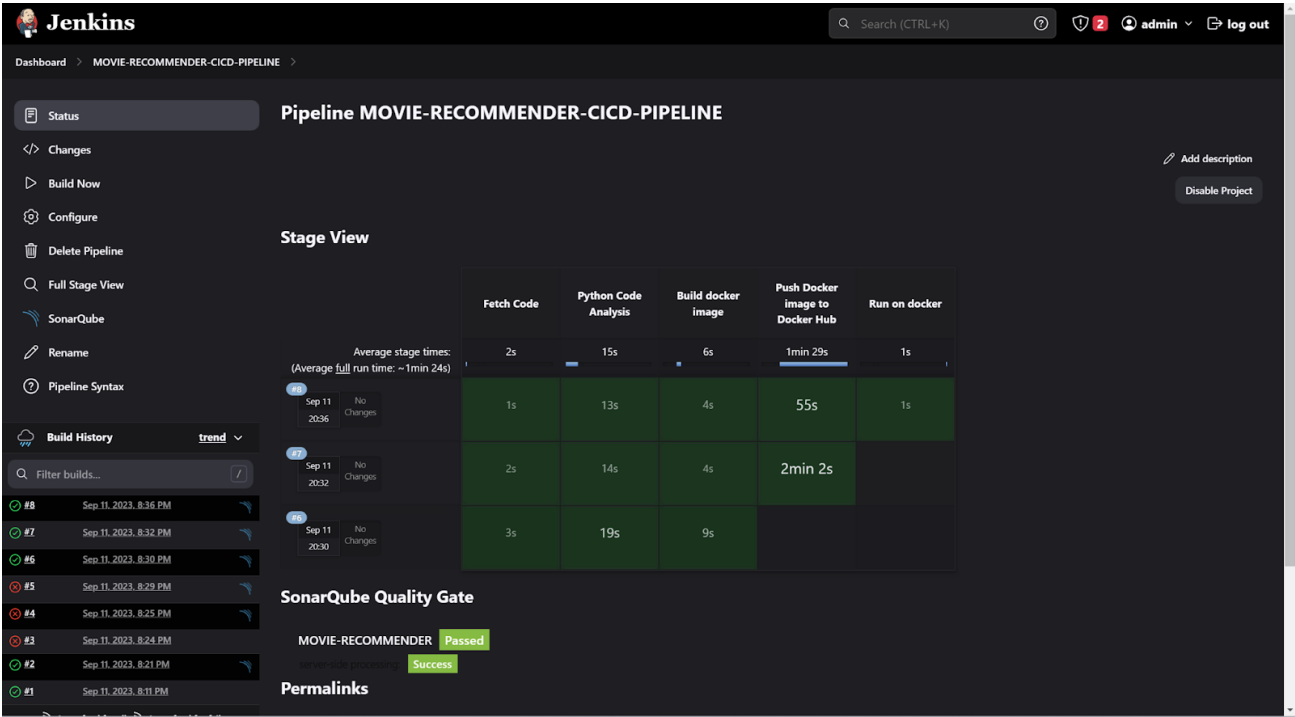
11.Pushing docker image to docker hub is succesful.



12.Now i have to deploy the image.To do this i have used docker run command. My image run on port 8501.



13.Deploying is successful.



14.Now i successfully access the app on port 8501.

