

CSS426 Machine Learning Operations(MLOps) Batch1

Lab 6

Automated Model Deployment Using Jenkins (CI/CD)

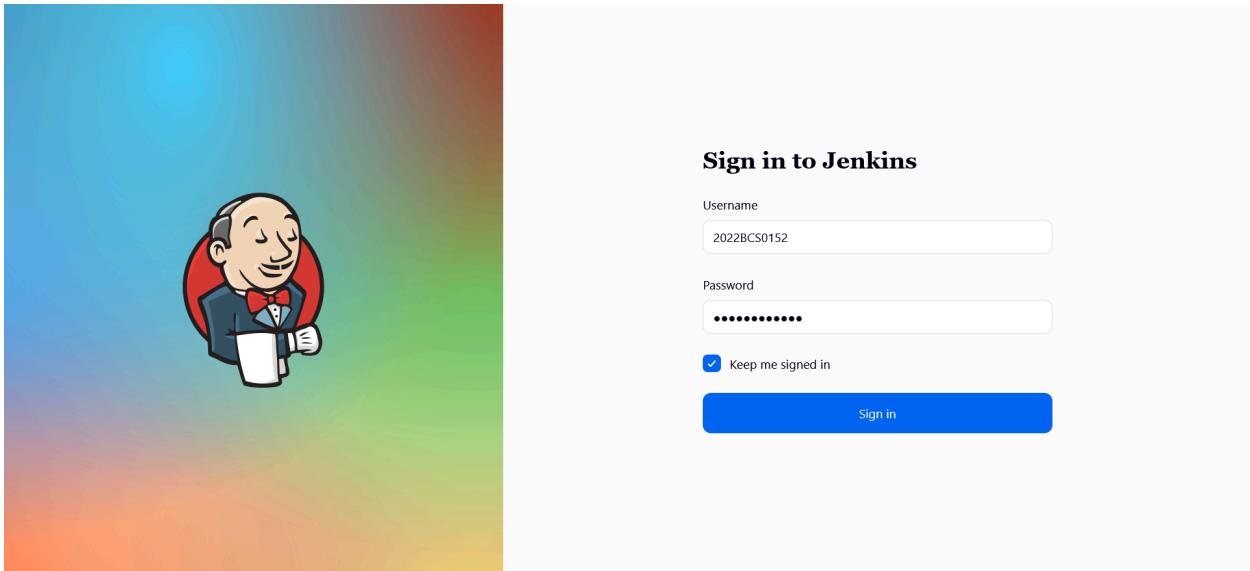
Name: S Varun Balaji
Roll No: 2022BCS0152

-
- > Github Repo Link: <https://github.com/2022bcs0152-varunbalaji/Lab-4>
 - > DockerHub Repo Link : <https://hub.docker.com/r/svarunbalaji2022bcs0152/wine-api>

- > Start Docker Image:

```
PS C:\Users\VARUN BALAJI> docker start 2022bcs0152_jenkins
2022bcs0152_jenkins
PS C:\Users\VARUN BALAJI> docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
NAMES
9140338e7707 2022bcs0152-jenkins "/usr/bin/tnt -- /u..." 7 days ago Up 14 seconds 0.0.0.0:8080->8080/tcp, [::]:8080->8080/tcp, 0.0.0.0:50000->50000/tcp, [::].50
000->50000/tcp 2022bcs0152_jenkins
PS C:\Users\VARUN BALAJI>
```

- > SignIn Jenkins:



-> Create Jenkinsfile :

```
pipeline {
    agent any

    environment {
        DOCKER_IMAGE = "svarunbalaji2022bcs0152/wine-api:latest"
    }

    stages {
        stage('Install Dependencies') {
            steps {
                sh '''
                    python3 -m venv venv
                    . venv/bin/activate
                    pip install -r requirements.txt
                '''
            }
        }

        stage('Train Model') {
            steps {
                sh '''
                    . venv/bin/activate
                    python scripts/train.py
                    echo "Name: Varun Balaji"
                    echo "Roll No: 2022BCS0152"
                '''
            }
        }

        stage('Build Docker Image') {
            steps {
                sh 'docker build -t $DOCKER_IMAGE .'
            }
        }

        stage('Push to DockerHub') {
```

```

        steps {
            withCredentials([string(credentialsId: 'dockerhub-token',
variable: 'DOCKER_PASS')]) {
                sh '''
                    echo $DOCKER_PASS | docker login -u
svarunbalaji2022bcs0152 --password-stdin
                    docker push $DOCKER_IMAGE
                '''
            }
        }
    }
}

```

-> Add the Jenkinsfile to git :

```

e0cc25d..1e40cf7  main -> main
● PS C:\Users\VARUN BALAJI\OneDrive\Desktop\Semester 8\MLops\Lab_4> git add Jenkinsfile
● PS C:\Users\VARUN BALAJI\OneDrive\Desktop\Semester 8\MLops\Lab_4> git commit -m "Fixed Jenkinsfile for SCM pipeline"
[main 3655f04] Fixed Jenkinsfile for SCM pipeline
 1 file changed, 1 insertion(+), 7 deletions(-)
● PS C:\Users\VARUN BALAJI\OneDrive\Desktop\Semester 8\MLops\Lab_4> git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 320 bytes | 320.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:2022bcs0152-varunbalaji/Lab-4.git
  1e40cf7..3655f04  main -> main
* [new branch]      main -> main
PS C:\Users\VARUN BALAJI\OneDrive\Desktop\Semester 8\MLops\Lab_4>

```

-> Install Dependencies/Plugins:

The screenshot shows the Jenkins 'Manage Jenkins' interface under the 'Plugins' section. The left sidebar has links for 'Updates', 'Available plugins', 'Installed plugins' (which is selected), and 'Advanced settings'. A search bar at the top right says 'Search installed plugins'. The main area lists several installed plugins with their names, versions, descriptions, and health status (green circles with numbers like 96, 92, 100). To the right of the list is a user profile for 'Varun Balaji' with options for 'Theme' (set to Light), 'My Views', 'Account', 'Appearance', 'Preferences', 'Security', 'Experiments', 'Credentials', and 'Sign out'. At the bottom of the page, the URL is http://localhost:8080/user/2022bc0152, and the system status bar shows various icons and the date 13-02-2026.

-> Create New Pipeline:

The screenshot shows the Jenkins 'New Item' dialog. The title is 'New Item' and it asks 'Enter an item name' with the input field containing '2022BCS0152-VarunBalaji'. Below it, 'Select an item type' is shown with a list of options. The 'Pipeline' option is highlighted with a blue background and a description: 'Build, test, and deploy using pipelines. Supports stages, parallel work, and running on multiple agents.' Other options include 'Freestyle project', 'Multi-configuration project', 'Folder', and 'Organization Folder'. At the bottom left is a blue 'OK' button, and at the bottom right are links for 'REST API' and 'Jenkins 2.541.1'.

Screenshot of Jenkins Pipeline configuration page:

The pipeline script is defined as "Pipeline script from SCM".

The SCM configuration shows a Git repository with the URL `https://github.com/2022bcs0152-varunbalaji/Lab-4.git`. There are no credentials specified.

Buttons at the bottom: Save and Apply.

Screenshot of Jenkins Pipeline configuration page:

The branches to build are specified as `*/main`.

The repository browser is set to (Auto).

Additional Behaviours: + Add

The Script Path is set to `Jenkinsfile`.

Buttons at the bottom: Save and Apply.

-> Create Access Token in Docker :

The screenshot shows the Docker web interface. On the left, there's a sidebar with a user profile for 'Varun Balaji S' and a 'Personal access tokens' section highlighted. The main content area is titled 'Create access token' and contains fields for 'Access token description' (set to 'jenkins-pipeline'), 'Expiration date' (set to '90 days'), and 'Access permissions' (set to 'Read, Write, Delete'). At the bottom are 'Cancel' and 'Generate' buttons.

-> Add new Global Credentials :

The screenshot shows the Jenkins 'New credentials' configuration page. It has sections for 'Kind' (set to 'Secret text'), 'Scope' (set to 'Global (Jenkins, nodes, items, all child items, etc)'), 'Secret' (containing a redacted password), 'ID' (set to 'dockerhub-token'), and 'Description' (set to 'DockerHub Access Token'). A 'Create' button is at the bottom. The footer indicates 'REST API' and 'Jenkins 2.541.1'.

-> Build and Run Jenkins Pipeline:

The screenshot shows the Jenkins interface for a pipeline named "2022BCS0152-VarunBalaji" run #4. The pipeline consists of the following stages: Start, Checkout SCM, Install Dependencies, Train Model, Build Docker Image, Push to DockerHub, and End. All stages have been completed successfully, indicated by green checkmarks. The "Checkout SCM" stage took 2 seconds and was started 15 minutes ago. The "Push to DockerHub" stage took 11 minutes and was started 15 minutes ago. The Jenkins version shown is 2.541.1.

```
graph LR; Start((Start)) --> CheckoutSCM[Checkout SCM]; CheckoutSCM --> InstallDependencies[Install Dependencies]; InstallDependencies --> TrainModel[Train Model]; TrainModel --> BuildDockerImage[Build Docker Image]; BuildDockerImage --> PushDockerHub[Push to DockerHub]; PushDockerHub --> End((End))
```

Graph

Start Checkout SCM Install Dependencies Train Model Build Docker Image Push to DockerHub End

Checkout SCM 2s Started 15m ago Jenkins

Install Dependencies 1s

Train Model 3s

Build Docker Image 4s

Push to DockerHub 11m

Jenkins 2.541.1

This screenshot shows the same Jenkins pipeline run #4. The "Install Dependencies" stage is currently highlighted, showing its command: "python3 -m venv venv .venv/bin/activate pip install -r requirements.txt". This stage completed successfully in 1 second. The Jenkins version is 2.541.1.

Graph

Start Checkout SCM Install Dependencies Train Model Build Docker Image Push to DockerHub End

Checkout SCM 2s

Install Dependencies 1s

Train Model 3s

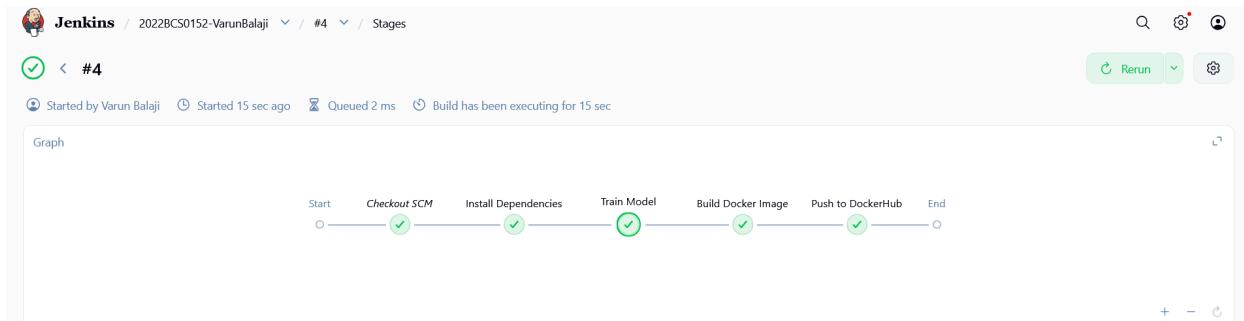
Build Docker Image 4s

Push to DockerHub 11m

Install Dependencies 1s Started 15m ago Jenkins

python3 -m venv venv .venv/bin/activate pip install -r requirements.txt

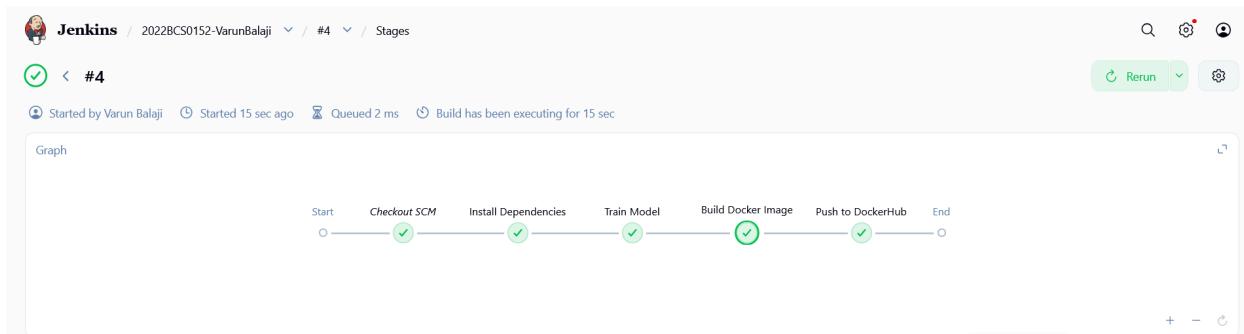
Jenkins 2.541.1



Search Train Model

- Checkout SCM 2s
- Install Dependencies 1s
- Train Model 3s**
- Build Docker Image 4s
- Push to DockerHub 11m

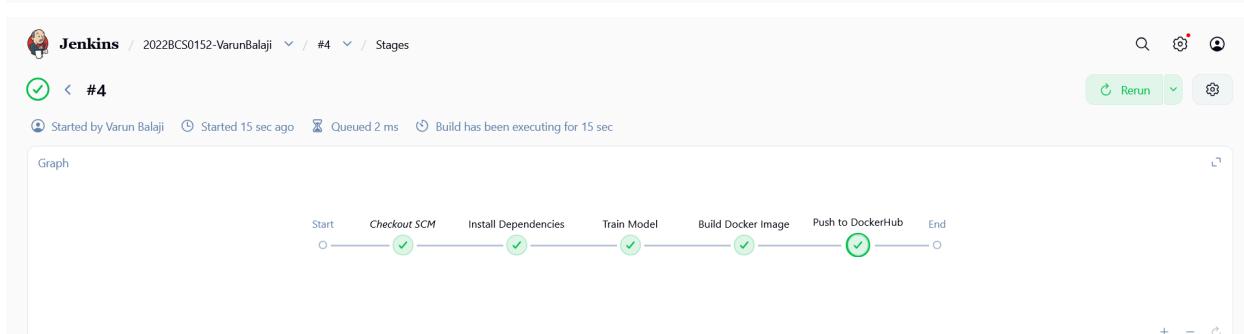
3s Started 16m ago Jenkins ⌂



Search Build Docker Image

- Checkout SCM 2s
- Install Dependencies 1s
- Train Model 3s
- Build Docker Image 4s**
- Push to DockerHub 11m

4s Started 16m ago Jenkins ⌂



Search Push to DockerHub

- Checkout SCM 2s
- Install Dependencies 1s
- Train Model 3s
- Build Docker Image 4s
- Push to DockerHub 11m**

11m Started 16m ago Jenkins ⌂

```

echo $DOCKER_PASS | docker login -u svarunbalaji2022bc0152 --password-stdin docker push $DOCKER_IMAGE
+ echo ****
+ docker login -u svarunbalaji2022bc0152 --password-stdin
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

```

5 ⌂