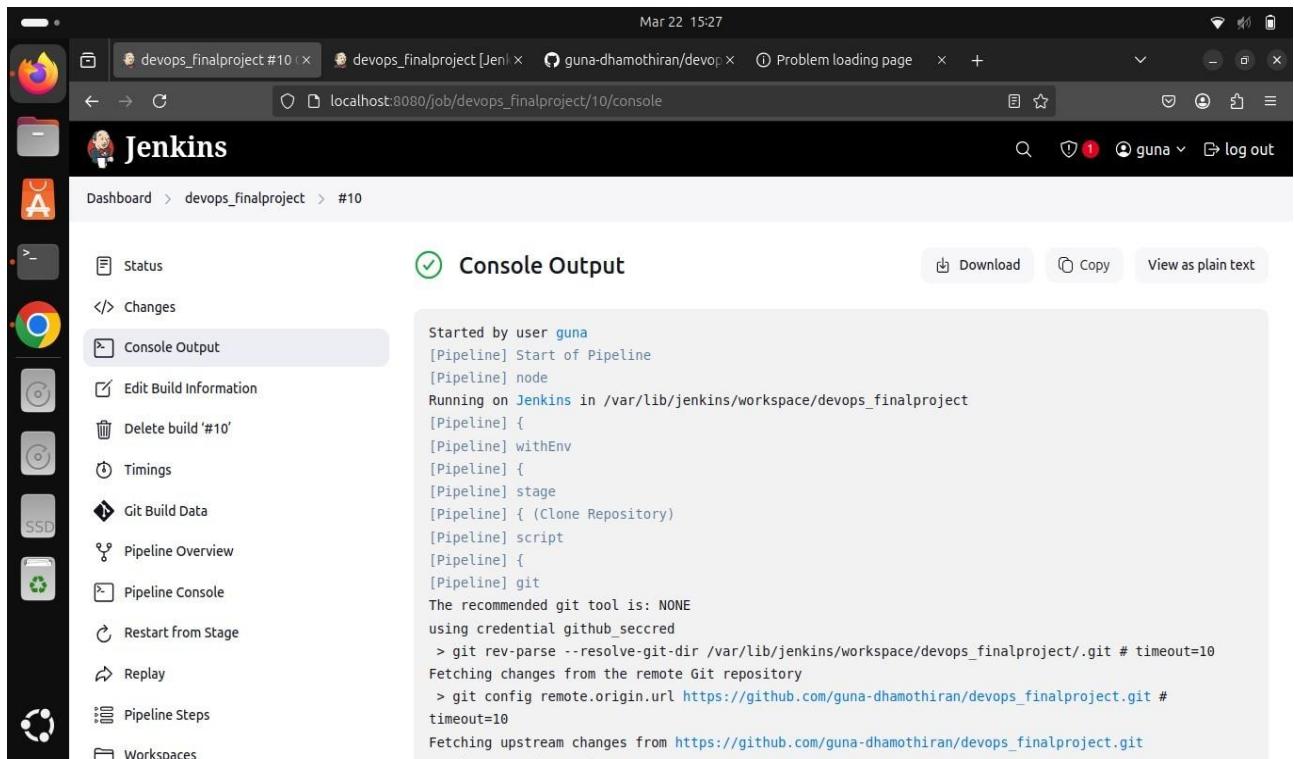


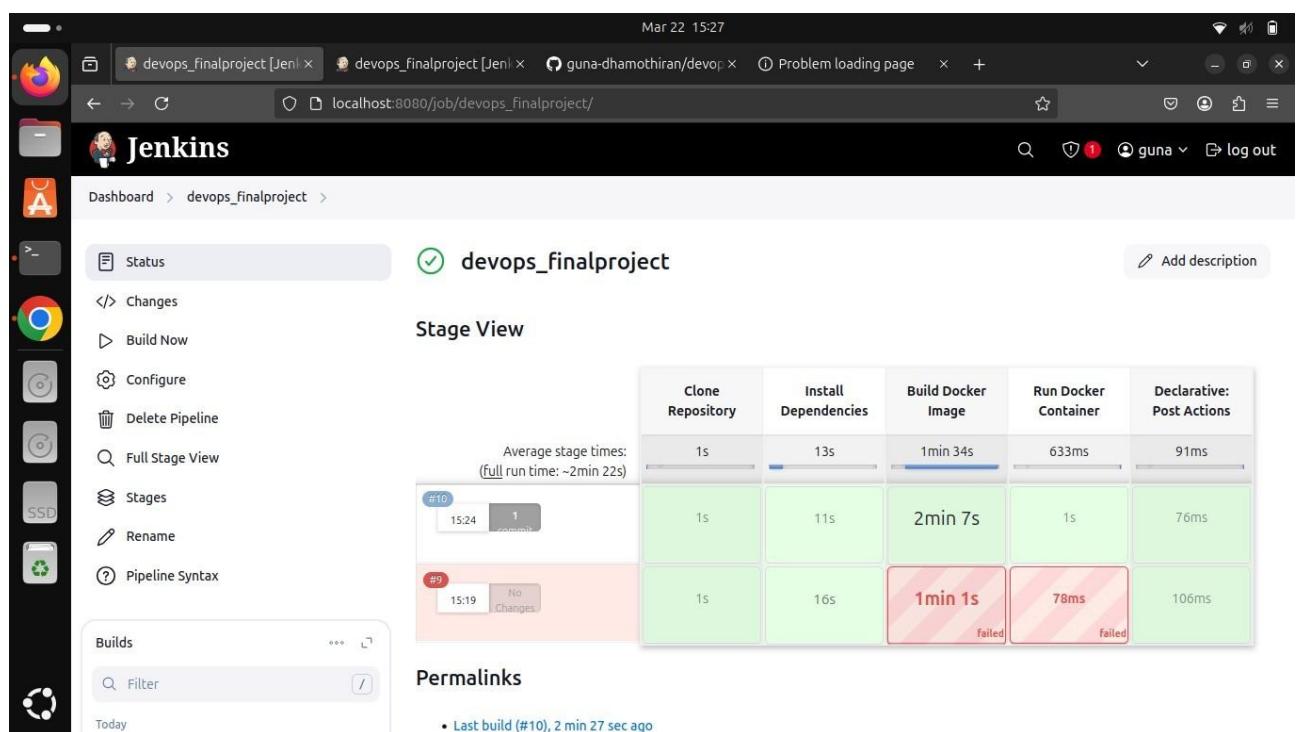
Devops_Final_project

Github link: <https://github.com/arasu-murugan/devopsproject>



The screenshot shows the Jenkins interface for the 'devops_finalproject' job. The left sidebar has a 'Console Output' item selected. The main content area is titled 'Console Output' and displays the log for build #10. The log output is as follows:

```
Started by user guna
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/devops_finalproject
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Clone Repository)
[Pipeline] script
[Pipeline] {
[Pipeline] git
The recommended git tool is: NONE
using credential github_seccred
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/devops_finalproject/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/guna-dhamothiran/devops_finalproject.git #
timeout=10
Fetching upstream changes from https://github.com/guna-dhamothiran/devops_finalproject.git
```



The screenshot shows the Jenkins interface for the 'devops_finalproject' job. The left sidebar has a 'Stage View' item selected. The main content area is titled 'devops_finalproject' and shows the 'Stage View' for build #10. The stage view table has the following columns: Clone Repository, Install Dependencies, Build Docker Image, Run Docker Container, and Declarative: Post Actions. The data for build #10 is as follows:

Clone Repository	Install Dependencies	Build Docker Image	Run Docker Container	Declarative: Post Actions
1s	13s	1min 34s	633ms	91ms
1s	11s	2min 7s	1s	76ms
1s	16s	1min 1s	78ms	106ms

A tooltip indicates an average stage time of approximately 2 minutes and 22 seconds. Below the table, there is a summary for build #9: 'No Changes.' The bottom section shows 'Permalinks' for the last build.

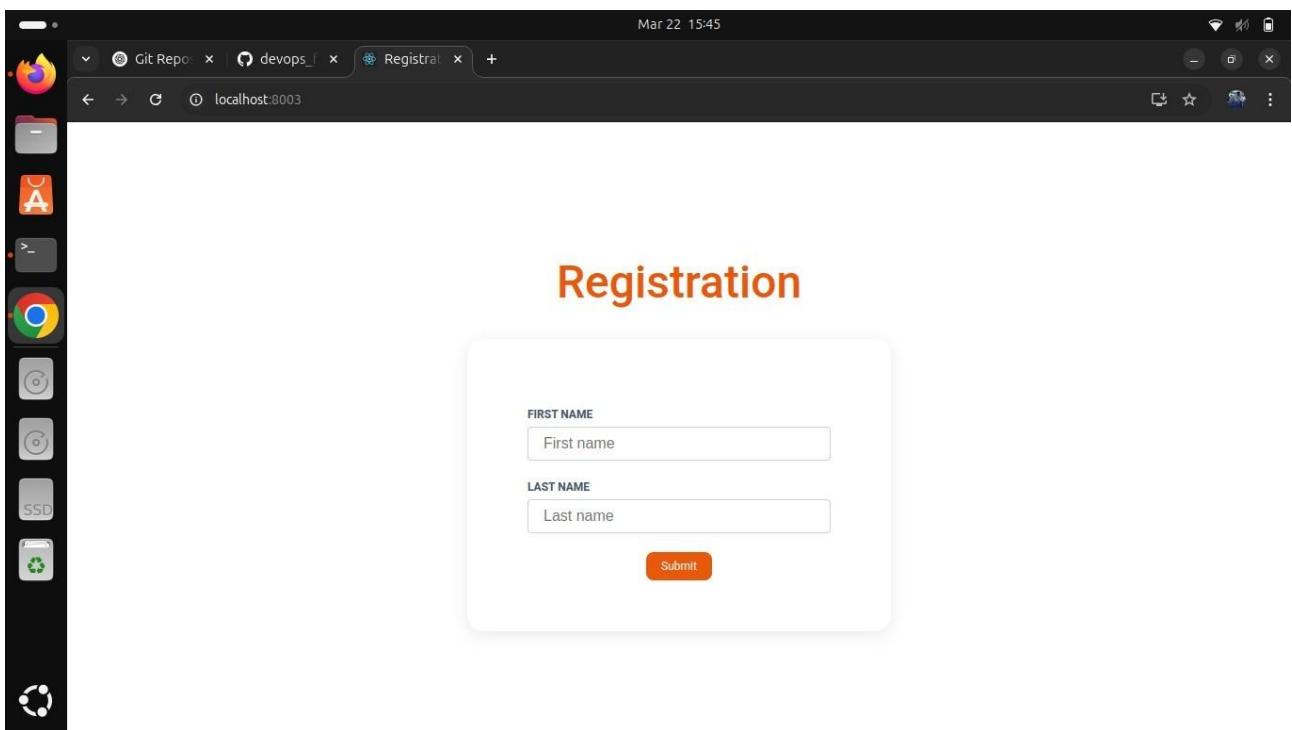
Mar 22 15:59

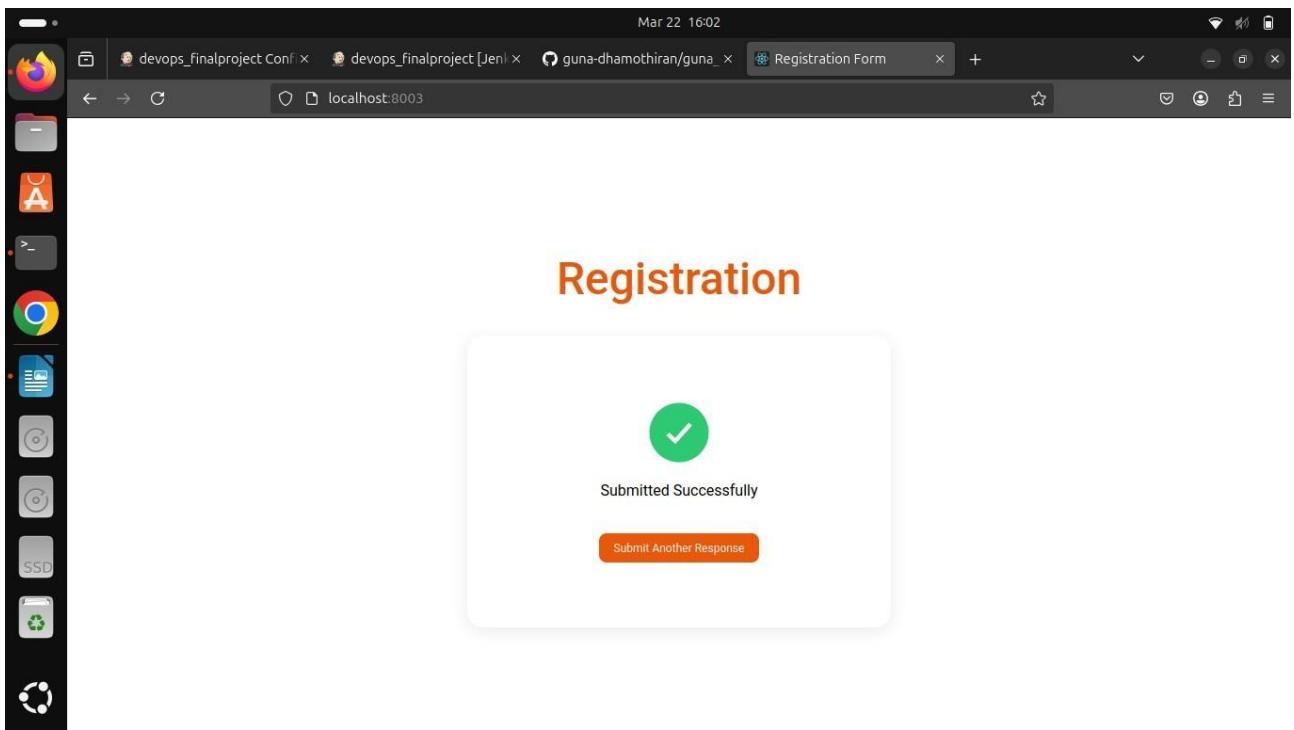
Untitled 1 — LibreOffice Writer

```
guna@guna-VivoBook-ASUSLaptop-X515EA-X515EA:~$ docker run -d -p 8003:3000 --name devops-final-app guna1311/devfinal
docker: Error response from daemon: Conflict. The container name "/devops-final-app" is already in use by container "ce98ddfb0dce4e8e487392ac7614a105cd045cdc2208a043234f26ab0b278d5c". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
guna@guna-VivoBook-ASUSLaptop-X515EA-X515EA:~$ # Stop the container (if running)
docker stop devops-final-app
# Remove the container
docker rm devops-final-app
devops-final-app
guna@guna-VivoBook-ASUSLaptop-X515EA-X515EA:~$ docker run -d -p 8003:3000 --name devops-final-app-v2 guna1311/devfinal
1688312be079fc0bc55c92c50390f9333b5b296b17941998bd8494d9f7a9a510
guna@guna-VivoBook-ASUSLaptop-X515EA-X515EA:~$
```

Registration

Pages 1 and 2 of 2 6 words, 146 characters Default Page Style English (USA) 100%





Dockerfile:

```
# Dockerfile
FROM node:18

# Set working directory
WORKDIR /app

# Copy package.json and package-lock.json
COPY package*.json .

# Install dependencies
RUN npm install

# Copy all project files COPY .

.

#  Use OpenSSL Legacy Provider to fix build issues
ENV NODE_OPTIONS=--openssl-legacy-provider

# Build the application RUN
npm run build

# Expose port and run application
EXPOSE 8003
CMD ["npm", "start"]
```

jenkins file: pipeline

```
{ agent any

environment {
    IMAGE_NAME = "arasumurugan/mydocker"
    CONTAINER_NAME = "devops-react-app"
    PORT = "5003"
}

stages {
    stage('Clone Repository') {
        steps {
            git 'https://github.com/arasu-murugan/devopsproject'
        }
    }

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

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

    stage('Run Docker Container') {
        steps {
            script {
                sh 'docker stop $CONTAINER_NAME || true'
                sh 'docker rm $CONTAINER_NAME || true'
                sh 'docker run -d --name $CONTAINER_NAME -p $PORT:80 $IMAGE_NAME'
            }
        }
    }
}
```

```
post {
success {
    echo '🎉 Deployment successful! Visit http://localhost:5003'
}
failure {
    echo '❌ Deployment failed. Check logs.'
}
}
```