JENKINS ASSIGNMENT SUBMISSION

Name: Vikram
Assignment: Jenkins – Trigger Pipeline Using Git (Develop Branch) & Pull Code into Folder

---

Problem Statement

You are required to configure a Jenkins Pipeline that automatically triggers whenever code is pushed to the develop branch in GitHub. The pipeline must also pull the Git repository content into a specific folder in the Jenkins workspace.

Environment Used

All Jenkins tasks were performed on a single Ubuntu EC2 instance, with:

- Jenkins installed on port 8080
- Git installed for repository cloning
- Java 11 installed as a prerequisite for Jenkins
- GitHub repository configured with a webhook
- Jenkins Pipeline job configured to trigger automatically on develop branch updates

---

Tasks Performed

Task 1: Install Jenkins on Ubuntu EC2 Instance

Steps Taken

1. Updated Ubuntu Packages

sudo apt update -y

sudo apt upgrade -y

```
ubuntu@ip-10-0-9-71:~$ sudo apt update -y
sudo apt upgrade -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1625 kB]
Get:14 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [1340 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [304 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [175 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [15.7 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1500 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [304 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [378 kB]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [31.4 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [2307 kB]
Get:23 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [219 kB]
Get:24 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [21.5 kB]
Get:25 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [9448 B]
```

2. Installed Java (required for Jenkins)

sudo apt install fontconfig openjdk-17-jre -y

java -version

```
ubuntu@ip-10-0-9-71:~$ sudo apt install openjdk-17-jre -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  adwaita-icon-theme gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme libcairo-gobject2 libcairo2 libdatrie1 libdeflate0 libgail-common libgail18t64
  libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libjbig0 liblerc4 libpango-1.0-0 libpangocairo-1.0-0
  libpangoft2-1.0-0 libpixman-1-0 librsvg2-2 librsvg2-common libsharpyuv0 libthai-data libthai0 libtiff6 libwebp7 libxcb-render0 libxcursor1 libxdamage1 libxfixes3
  openjdk-17-jre-headless ubuntu-mono
Suggested packages:
  gvfs librsvg2-bin libnss-mdns fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei fonts-indic
The following NEW packages will be installed:
  adwaita-icon-theme gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme libcairo-gobject2 libcairo2 libdatrie1 libdeflate0 libgail-common libgail18t64
  libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libjbig0 liblerc4 libpango-1.0-0 libpangocairo-1.0-0
  libpangoft2-1.0-0 libpixman-1-0 librsvg2-2 librsvg2-common libsharpyuv0 libthai-data libthai0 libtiff6 libwebp7 libxcb-render0 libxcursor1 libxdamage1 libxfixes3
  openjdk-17-jre openjdk-17-jre-headless ubuntu-mono
0 upgraded, 36 newly installed, 0 to remove and 0 not upgraded.
Need to get 57.3 MB of archives.
After this operation, 254 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libgdk-pixbuf2.0-common all 2.42.10+dfsg-3ubuntu3.2 [8192 B]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libdeflate0 amd64 1.19-1build1.1 [43.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libjbig0 amd64 2.1-6.1ubuntu2 [29.7 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblerc4 amd64 4.0.0+ds-4ubuntu2 [179 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libsharpyuv0 amd64 1.3.2-0.4build3 [15.8 kB]
```

3. Installed Git

sudo apt install git -y

```
ubuntu@ip-10-0-9-71:~$ sudo apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.43.0-1ubuntu7.3).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-10-0-9-71:~$
```

4. Added Jenkins Repository

curl -fsSL https://pkg.jenkins.io/debian/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 binary/ | sudo tee \

/etc/apt/sources.list.d/jenkins.list > /dev/null

```
ubuntu@ip-10-0-9-71:~$ curl -fsSL https://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee \
/usr/share/keyrings/jenkins-keyring.asc > /dev/null
ubuntu@ip-10-0-9-71:~$ echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
ubuntu@ip-10-0-9-71:~$
```

5. Installed Jenkins

sudo apt update

sudo apt install jenkins -y

```
ubuntu@ip-10-0-9-71:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Ign:5 https://pkg.jenkins.io/debian binary/ InRelease
Get:6 https://pkg.jenkins.io/debian binary/ Release [2044 B]
Get:7 https://pkg.jenkins.io/debian binary/ Release.gpg [833 B]
Get:8 https://pkg.jenkins.io/debian binary/ Packages [74.3 kB]
Fetched 77.2 kB in 1s (143 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
ubuntu@ip-10-0-9-71:~$ sudo apt install jenkins -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 96.0 MB of archives.
After this operation, 97.1 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 net-tools amd64 2.10-0.1ubuntu4.4 [204 kB]
Get:2 https://pkg.jenkins.io/debian binary/ jenkins 2.540 [95.8 MB]
```

6. Started Jenkins Service

sudo systemctl start jenkins

sudo systemctl enable jenkins

sudo systemctl status Jenkins

```
ubuntu@ip-10-0-9-71:~$ sudo systemctl start jenkins
sudo systemctl enable jenkins
sudo systemctl status jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
     Active: active (running) since Wed 2025-12-03 02:52:40 UTC; 1s ago
   Main PID: 2813 (java)
      Tasks: 47 (limit: 2213)
     Memory: 666.2M (peak: 678.9M)
        CPU: 23.421s
     CGroup: /system.slice/jenkins.service
             └─2813 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

Dec 03 02:52:35 ip-10-0-9-71 jenkins[2813]: [LF]> This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Dec 03 02:52:35 ip-10-0-9-71 jenkins[2813]: [LF]>
Dec 03 02:52:35 ip-10-0-9-71 jenkins[2813]: [LF]> ************************************************************
Dec 03 02:52:35 ip-10-0-9-71 jenkins[2813]: [LF]> ************************************************************
Dec 03 02:52:35 ip-10-0-9-71 jenkins[2813]: [LF]> ************************************************************
Dec 03 02:52:40 ip-10-0-9-71 jenkins[2813]: 2025-12-03 02:52:40.350+0000 [id=32]        INFO        jenkins.InitReactorRunner$1#onAttained: Completed initialization
Dec 03 02:52:40 ip-10-0-9-71 jenkins[2813]: 2025-12-03 02:52:40.379+0000 [id=23]        INFO        hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up and running
Dec 03 02:52:40 ip-10-0-9-71 systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
Dec 03 02:52:40 ip-10-0-9-71 jenkins[2813]: 2025-12-03 02:52:40.723+0000 [id=49]        INFO        h.m.DownloadService$Downloadable#load: Obtained the updated data fil
Dec 03 02:52:40 ip-10-0-9-71 jenkins[2813]: 2025-12-03 02:52:40.723+0000 [id=49]        INFO        hudson.util.Retrier#start: Performed the action check updates server
lines 1-20/20 (END)
```

7. Opened Port 8080 in EC2 Security Group

Allowed:

Custom TCP → Port 8080 → Source: 0.0.0.0/0

Accessed Jenkins using:

http://<EC2_PUBLIC_IP>:8080

Task 2: Configure Jenkins and Required Plugins

Steps Taken

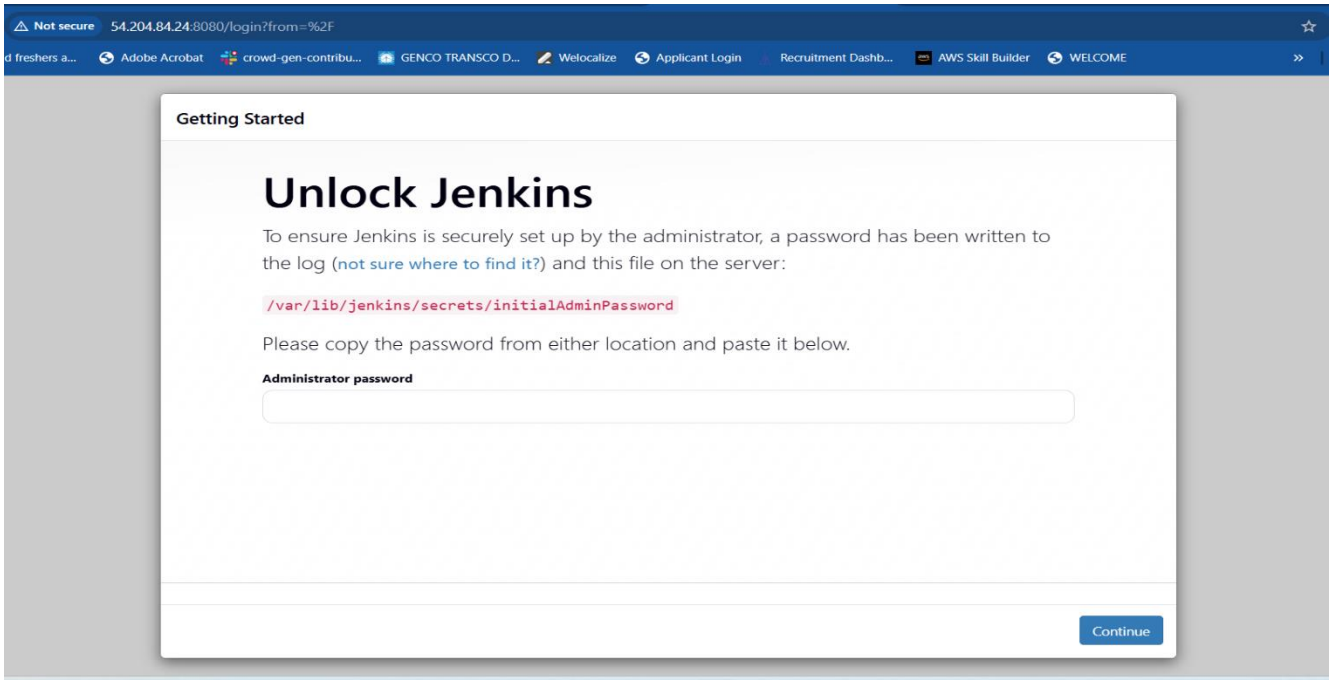
1. Retrieved Jenkins Admin Password

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

2. Installed Suggested Plugins

This included:

- Git Plugin
- Pipeline Plugin
- GitHub Integration Plugin



Task 3: Create Jenkins Pipeline Job Triggered by Git

Steps Taken

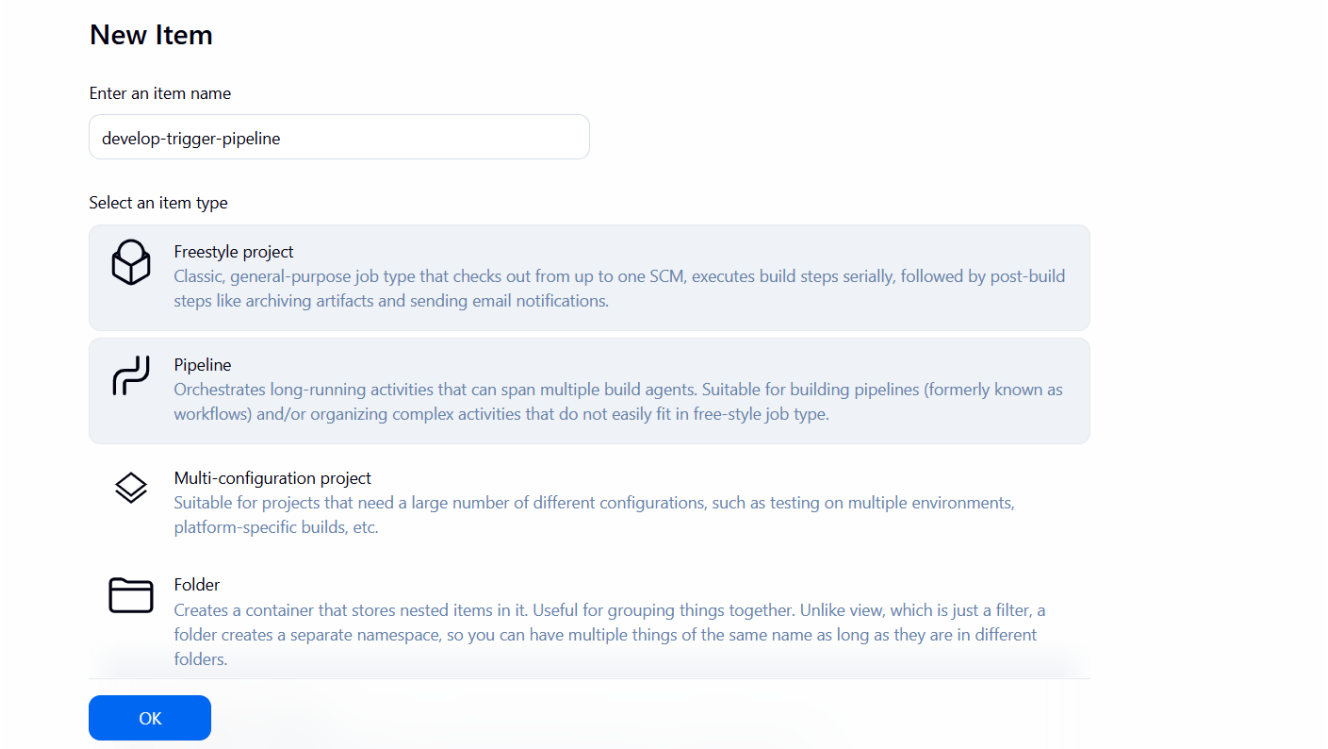
1. Created New Pipeline Job

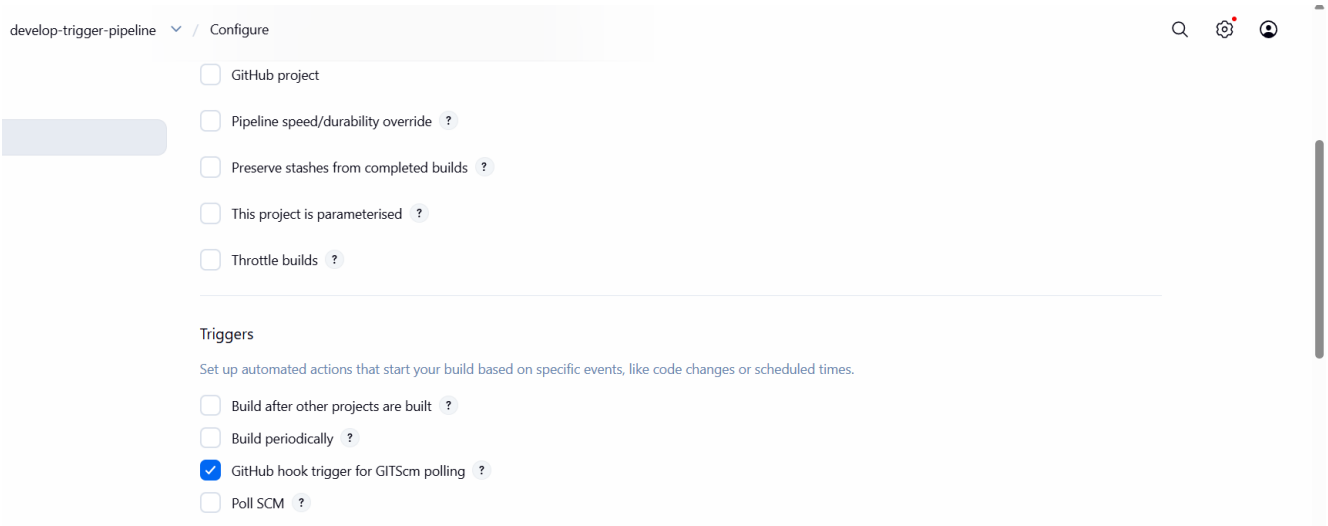
- Name: develop-trigger-pipeline
- Type: Pipeline

2. Enabled GitHub Trigger

Under Build Triggers, enabled:

GitHub hook trigger for GITScm polling





Task 4: Add Pipeline Script to Pull Code into a Folder

Steps Taken

Added the following script:

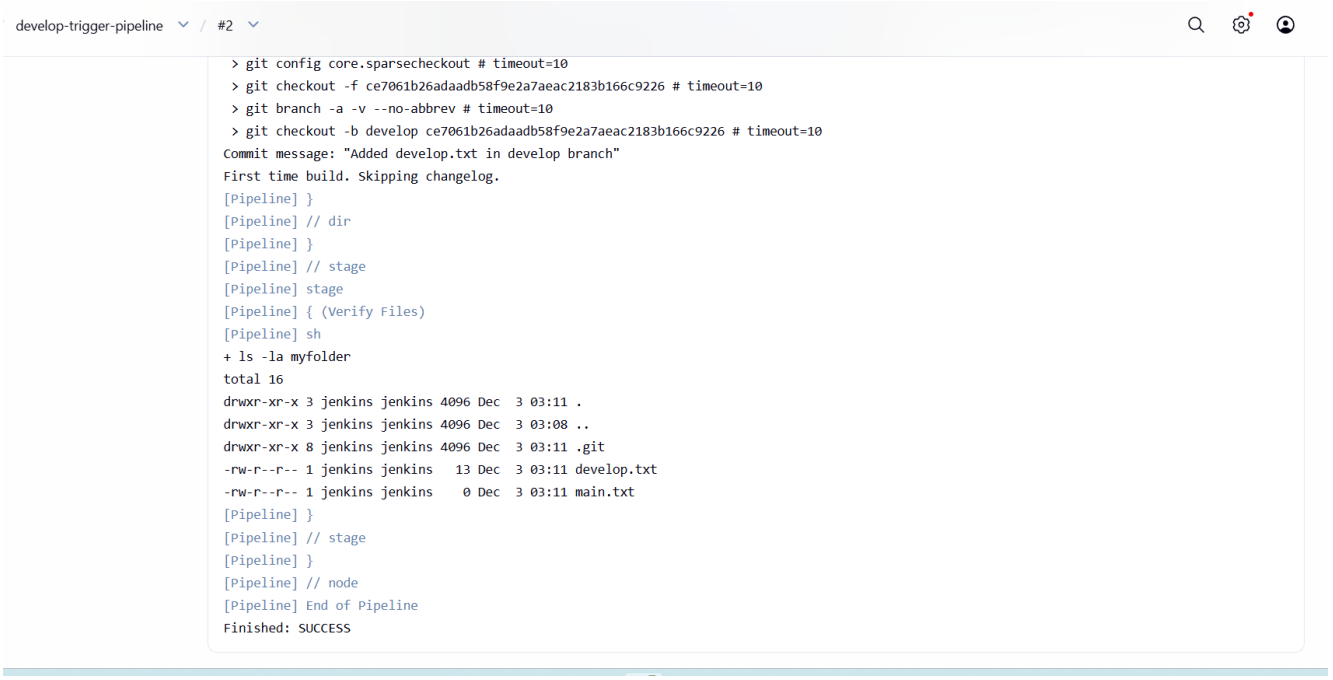
```
pipeline {
    agent any

    stages {
        stage('Pull Code to Folder') {
            steps {
                dir('myfolder') {
                    git branch: 'develop',
                        url: 'https://github.com/<your-username>/<your-repo>.git'
                }
            }
        }

        stage('Verify Files') {
            steps {
                sh 'ls -la myfolder'
            }
        }
    }
}
```

Apply and save.

Click on build now.

This script performs:

- Automatic checkout of the develop branch
- Cloning into a folder named myfolder
- Listing the files for confirmation



Task 5: Create GitHub Webhook to Trigger Jenkins Pipeline

Steps Taken

Go to:

GitHub → Repository → Settings → Webhooks → Add Webhook

Fill:

Payload URL

http://<EC2_PUBLIC_IP>:8080/github-webhook/

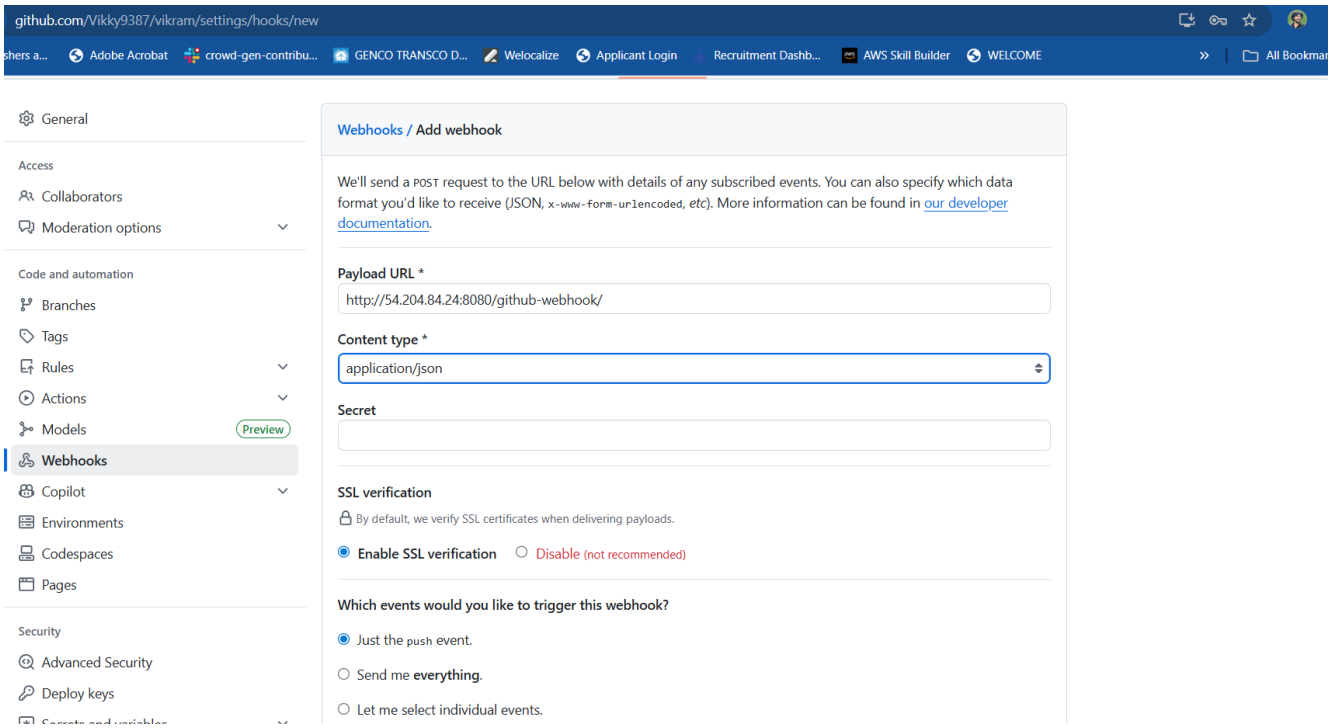
Content type

application/json

Event

✓ Just the push event

Webhook was successfully created.



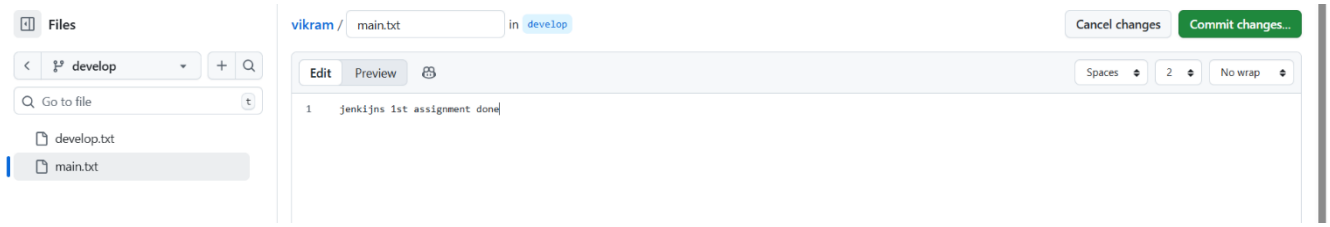
✓ Task 6: Test Automatic Pipeline Trigger (UPDATED)

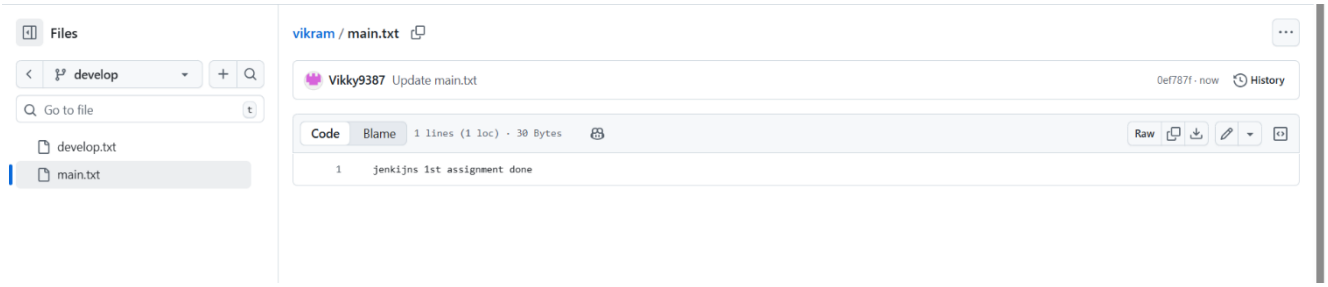
Steps Taken

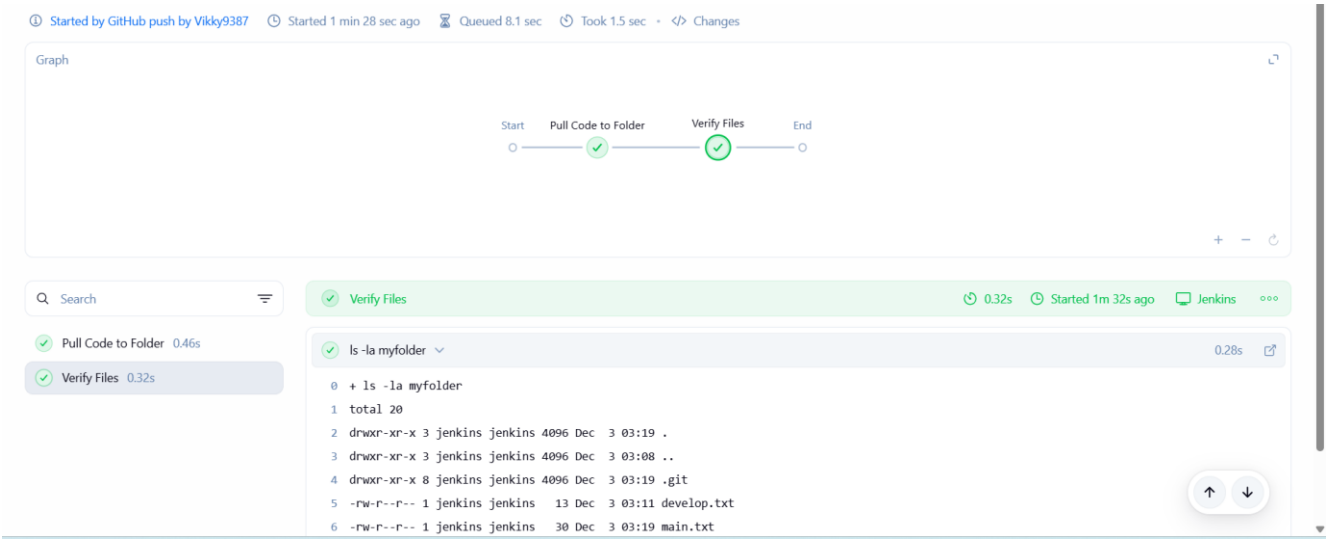
Performed a commit directly in GitHub:

1. Opened the repository on GitHub
2. Selected the develop branch
3. Opened or created any file
4. Clicked Edit
5. Made a small change
6. Committed directly to the develop branch

This action automatically triggered the Jenkins pipeline.







Conclusion

Successfully installed Jenkins on Ubuntu EC2, configured required plugins, created a pipeline job, set up GitHub webhook triggers, and implemented a Jenkins Pipeline that pulls code from the develop branch into a specific folder when a push is made. This completes all assignment requirements and demonstrates the ability to configure CI/CD automation using GitHub and Jenkins.