St. Francis Institute of Technology, Mumbai-400 103
Department Of Information Technology

A.Y. 2024-2025
Class: TE-ITA/B, Semester: V

Subject: **DevOps Lab**

Experiment – 7: a. To build pipeline of jobs in Jenkins, create a pipeline script to test and deploy an application. b. To automatically build a job in Jenkins using webhooks (Topic Beyond Syllabus)

1. **Aim:** To build pipeline of jobs in Jenkins, create a pipeline script to test and deploy an application
2. **Objectives:** Aim of this experiment is that, the students will be able
   - To build pipeline of jobs in Jenkins, create a pipeline script to test and deploy an application
3. **Outcomes:** After study of this experiment, the students will be able
   - To understand the importance of Jenkins to Build and deploy Software Applications on server environment.
4. **Prerequisite:** Knowledge of software engineering concept of integration and deployment
5. **Requirements:** Jenkins, JDK, python, Personal Computer, Windows operating system, browser, Internet Connection, Microsoft Word.
6. **Pre-Experiment Exercise:**
   Brief Theory: Refer shared material
7. **Laboratory Exercise**
   A. **Procedure:**
      a. **Answer the following:**
         - What is Jenkins pipeline?
         - What are the different ways to write a Jenkins pipeline?
      b. **Execute following (Refer the shared material) and attach screenshots:**
         - Create and build pipeline project with Git
         - Create and build pipeline project with pipeline script
         - Create and automatically build a pipeline project using webhooks
8. **Post-Experiments Exercise**
   A. **Extended Theory:**
      Nil
   B. **Questions:**
      - Explain the types of agents in a Jenkinsfile?
      - What are webhooks?
   C. **Conclusion:**
      - Write what was performed in the experiment.
      - Write the significance of the topic studied in the experiment.
   D. **References:**

https://jenkins.io/doc/
https://www.jenkins.io/doc/book/pipeline/syntax/
https://www.edureka.co/blog/jenkins-pipeline-tutorial-continuous-delivery
https://www.slideshare.net/abediaz/introduction-to-jenkins
https://www.slideshare.net/jph98/jenkins-ci-presentation

Name: Vishal Rajesh Mahajan Devops EXP07
Class: TE IT A Roll No: 62

7A. Answer the following:

1. What is the Jenkins pipeline?

=> A Jenkins Pipeline is a suite of plugins that supports implementing and integrating continuous delivery pipelines into Jenkins. A pipeline is a sequence of events or jobs that are linked together to perform a task. Jenkins pipelines are used to define the series of steps that your Jenkins server will execute on any given project.

Key Features:

1. Code as Configuration: Jenkins pipelines are typically defined using a DSL (Domain-Specific Language) that models a series of steps in code. This allows the pipelines to be version-controlled as part of the project.
2. Durable: Pipelines can survive Jenkins master restarts and can handle failures in a robust manner.
3. Versatile: They support complex real-world processes and can model any combination of tasks.
4. Extensible: Through the use of plugins, Jenkins pipelines can be extended and customized to fit the needs of the project.

Types of Pipelines:

1. Declarative Pipeline: A more structured and opinionated approach to defining pipelines using a predefined syntax. It provides a simple and easy-to-read way of creating pipelines.
2. Scripted Pipeline: Uses a more powerful Groovy-based DSL that provides greater flexibility and control over the pipeline execution but requires more knowledge of Groovy scripting.

2. What are the different ways to write a Jenkins pipeline?

=> There are mainly two ways to write Jenkins pipelines:

1. Declarative Pipeline
2. Scripted Pipeline

Declarative pipelines provide a simplified and more readable syntax for defining Jenkins pipelines. The syntax is more structured and easier for beginners to understand. The pipeline is defined within a pipeline block.

Example:

```
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                echo 'Building...'
            }
        }
        stage('Test') {
            steps {
                echo 'Testing...'
            }
        }
        stage('Deploy') {
            steps {
                echo 'Deploying...'
            }
        }
    }
}
```

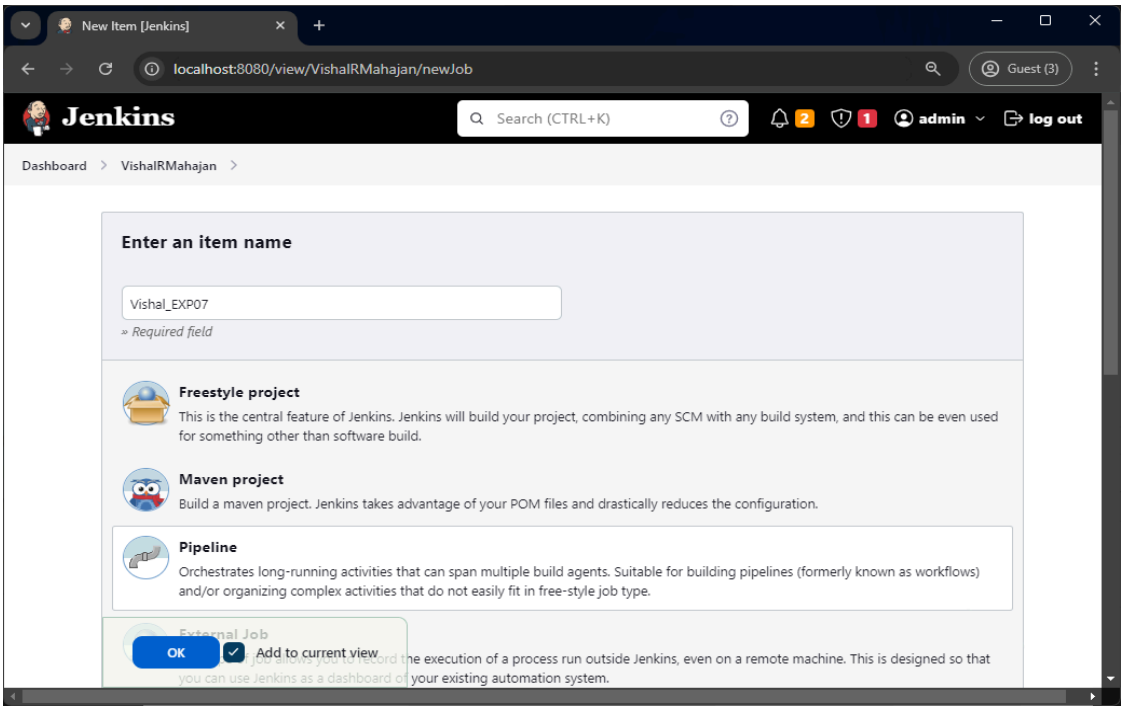
Scripted pipelines use a more flexible and powerful Groovy-based syntax. They provide more control and are suitable for more complex pipeline requirements. The pipeline script is defined in a node block.

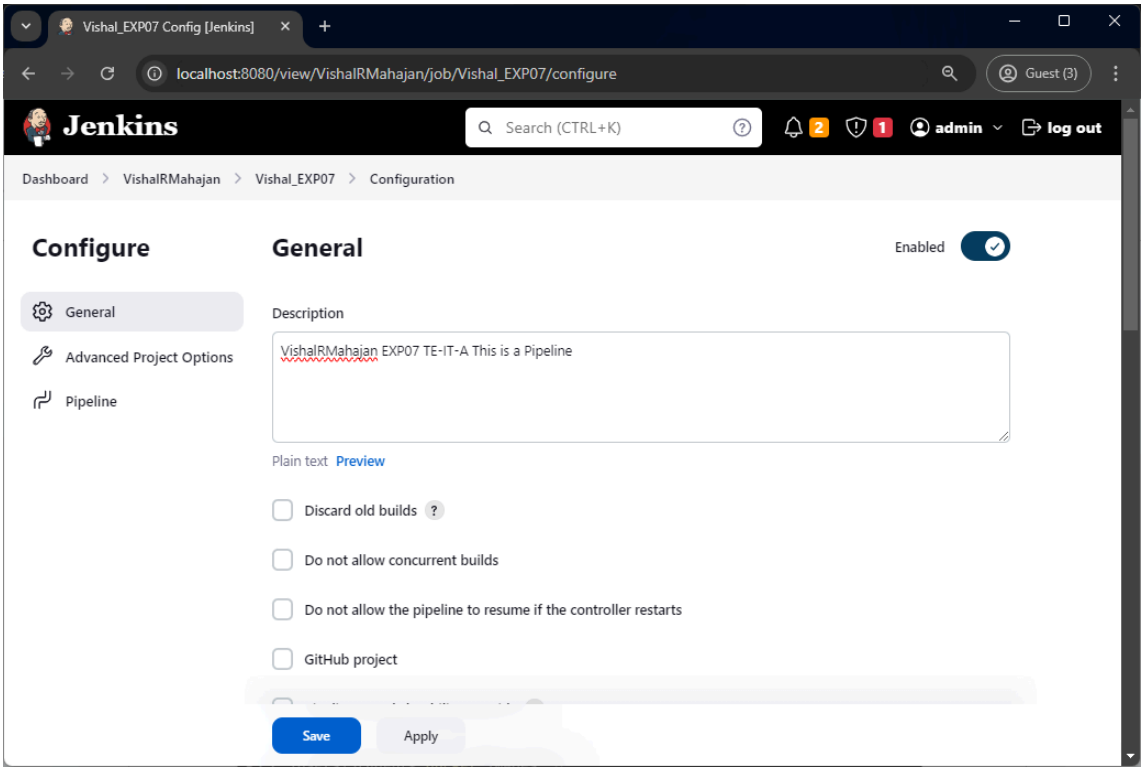
Example:

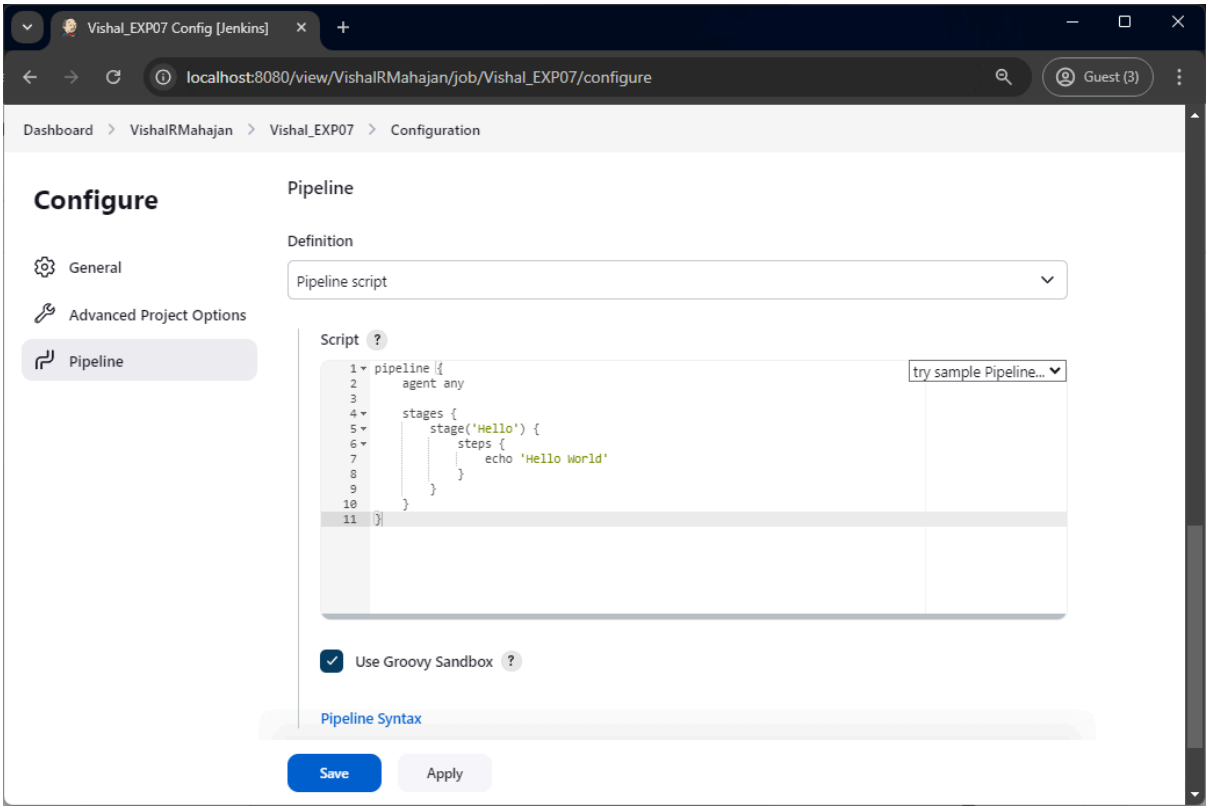
```
node {
    stage('Build') {
        echo 'Building...'
    }
    stage('Test') {
        echo 'Testing...'
    }
    stage('Deploy') {
        echo 'Deploying...'
    }
}
```

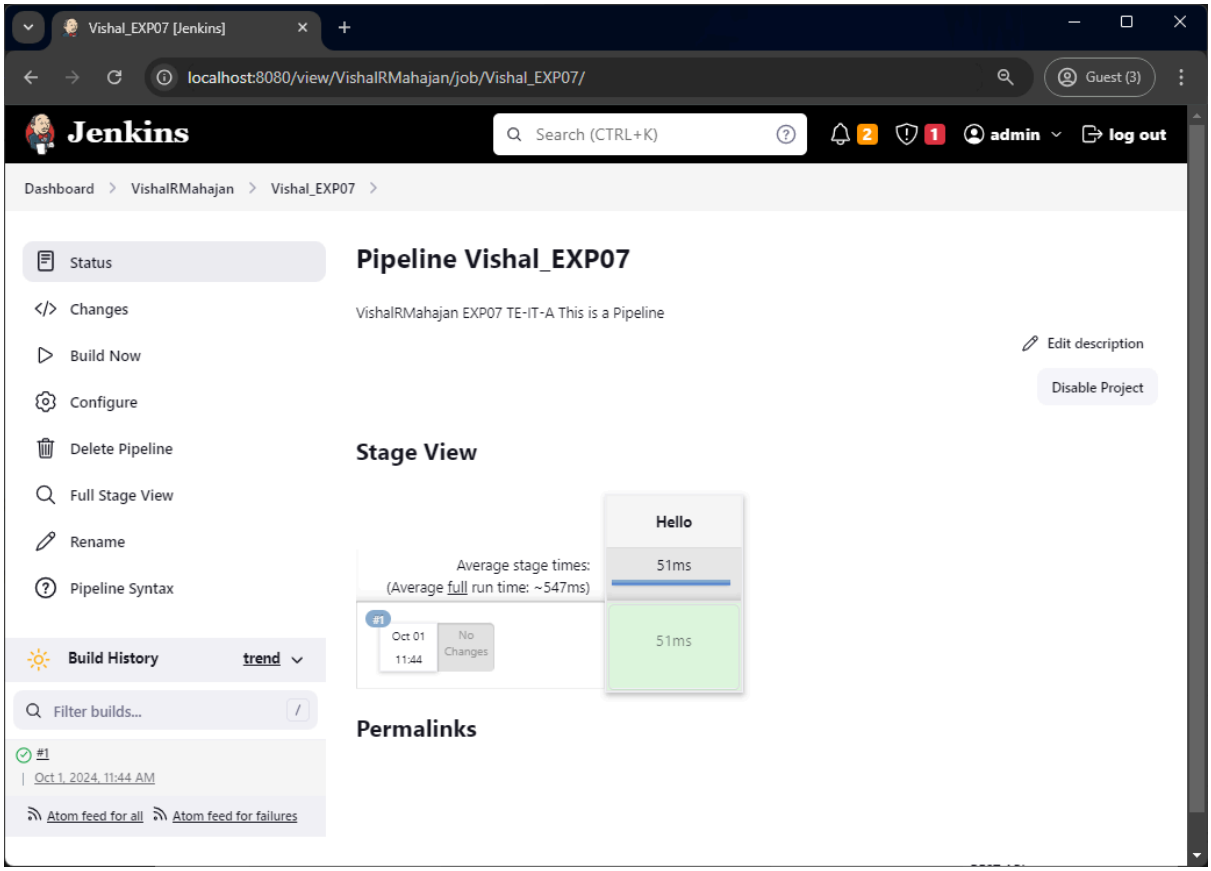
7b. Execute following (Refer the shared material) and attach screenshots:

1. **Create and build pipeline project with pipeline script**
   - **Project with Hello World pipeline script**





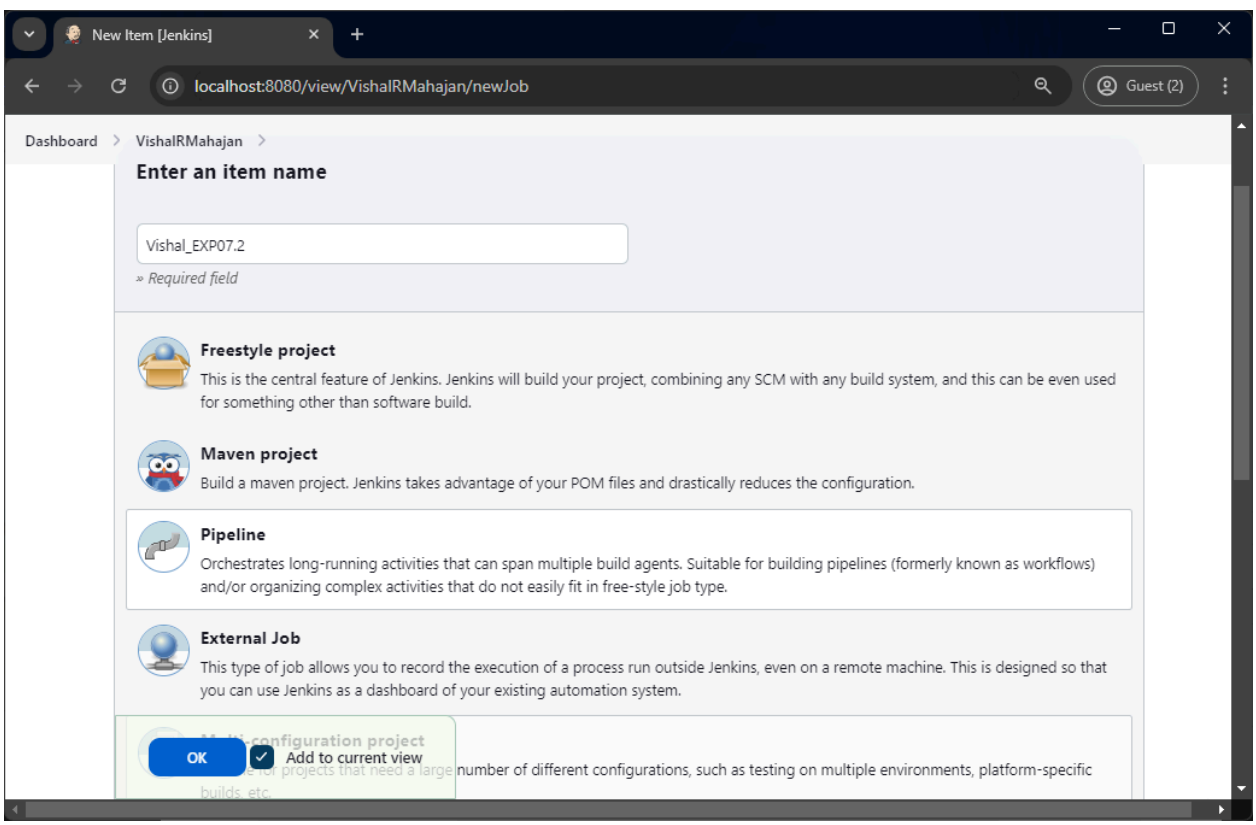

Vishal_EXP07 Config [Jenkins]
localhost:8080/view/VishalRMahajan/job/Vishal_EXP07/configure
Guest (3)
Dashboard > VishalRMahajan > Vishal_EXP07 > Configuration
Configure
General
Advanced Project Options
Pipeline
Pipeline
Definition
Pipeline script
Script
try sample Pipeline...
pipeline {
agent any
stages {
stage('Hello') {
steps {
echo 'Hello World'
}
}
}
}
Use Groovy Sandbox
Pipeline Syntax
Save
Apply


Vishal_EXP07 [Jenkins]
localhost:8080/view/VishalRMahajan/job/Vishal_EXP07/
Guest (3)
Jenkins
Search (CTRL+K)
admin
log out
Dashboard > VishalRMahajan > Vishal_EXP07 >
Status
Changes
Build Now
Configure
Delete Pipeline
Full Stage View
Rename
Pipeline Syntax
Build History
trend
Filter builds...
#1
Oct 1, 2024, 11:44 AM
Atom feed for all
Atom feed for failures
Pipeline Vishal_EXP07
VishalRMahajan EXP07 TE-IT-A This is a Pipeline
Edit description
Disable Project
Stage View
Hello
Average stage times:
(Average full run time: ~547ms)
51ms
#1
Oct 01
11:44
No Changes
51ms
Permalinks

Console Output

```
Started by user admin
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\ProgramData\Jenkins\.jenkins\workspace\Vishal_EXP07
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Hello)
[Pipeline] echo
Hello World
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

- **Project with your own pipeline script**



Vishal_EXP07.2 Config [Jenkins]

localhost:8080/view/VishalRMahajan/job/Vishal_EXP07.2/configure

Jenkins

Search (CTRL+K)

admin log out

Dashboard > VishalRMahajan > Vishal_EXP07.2 > Configuration

Configure

- General
- Advanced Project Options
- Pipeline

General

Enabled

Description

This is VishalRMahajan TEITA4 62 Second Pipeline

Plain text Preview

- [ ] Discard old builds
- [ ] Do not allow concurrent builds
- [ ] Do not allow the pipeline to resume if the controller restarts
- [ ] GitHub project

Save Apply

Vishal_EXP07.2 Config [Jenkins]

localhost:8080/view/VishalRMahajan/job/Vishal_EXP07.2/configure

Dashboard > VishalRMahajan > Vishal_EXP07.2 > Configuration

Configure

- General
- Advanced Project Options
- Pipeline

Pipeline

Definition

Pipeline script

Script

try sample Pipeline...

```
pipeline {
    agent any
    stages {
        stage('Example') {
            input {
                message "Should we continue?"
                ok "Yes, we should."
                submitter "alice,bob"
                parameters {
                    string(name: 'PERSON', defaultValue: 'Mr Jenkins', description: 'Who should I
                }
            }
            steps {
                echo "Hello, ${PERSON}, nice to meet you."
            }
        }
```

- [x] Use Groovy Sandbox

Pipeline Syntax

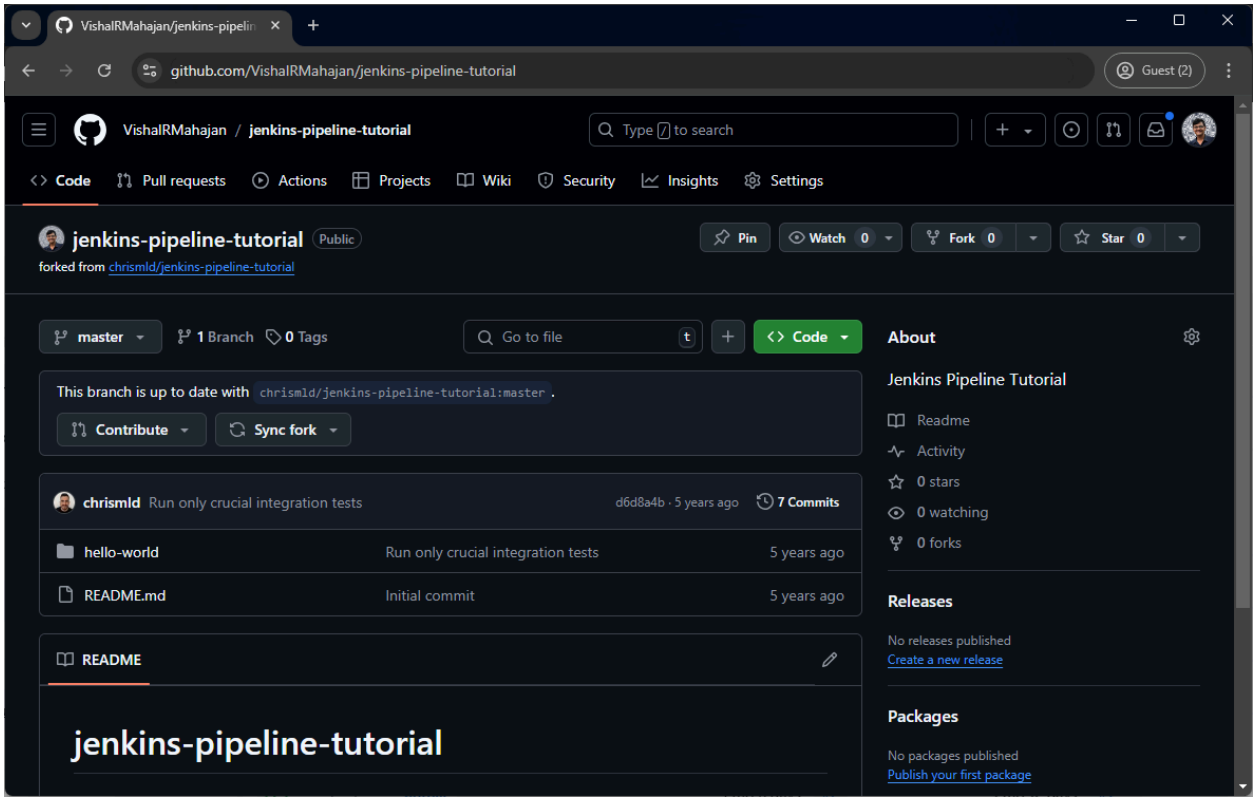
Save Apply

Console Output

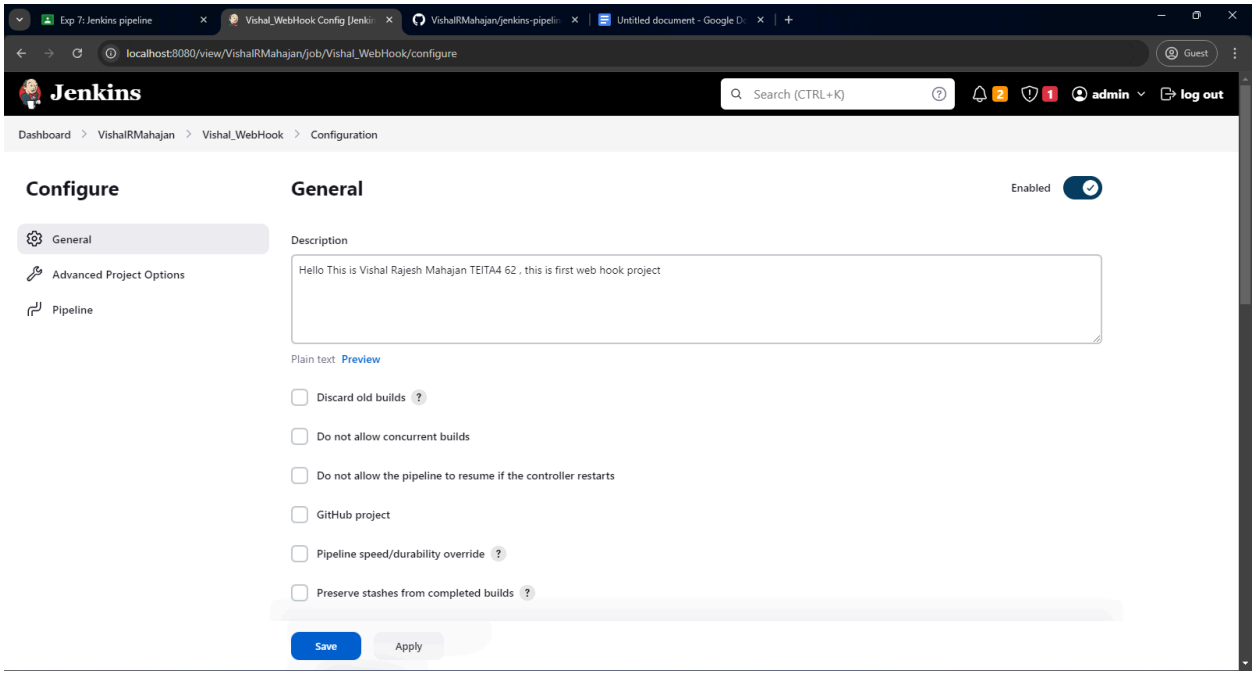
```
Started by user admin
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\ProgramData\Jenkins\.jenkins\workspace\Vishal_EXP07.2
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Example)
[Pipeline] input
Input requested
Approved by admin
[Pipeline] withEnv
[Pipeline] {
[Pipeline] echo
Hello, Mr Jenkins, nice to meet you.
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

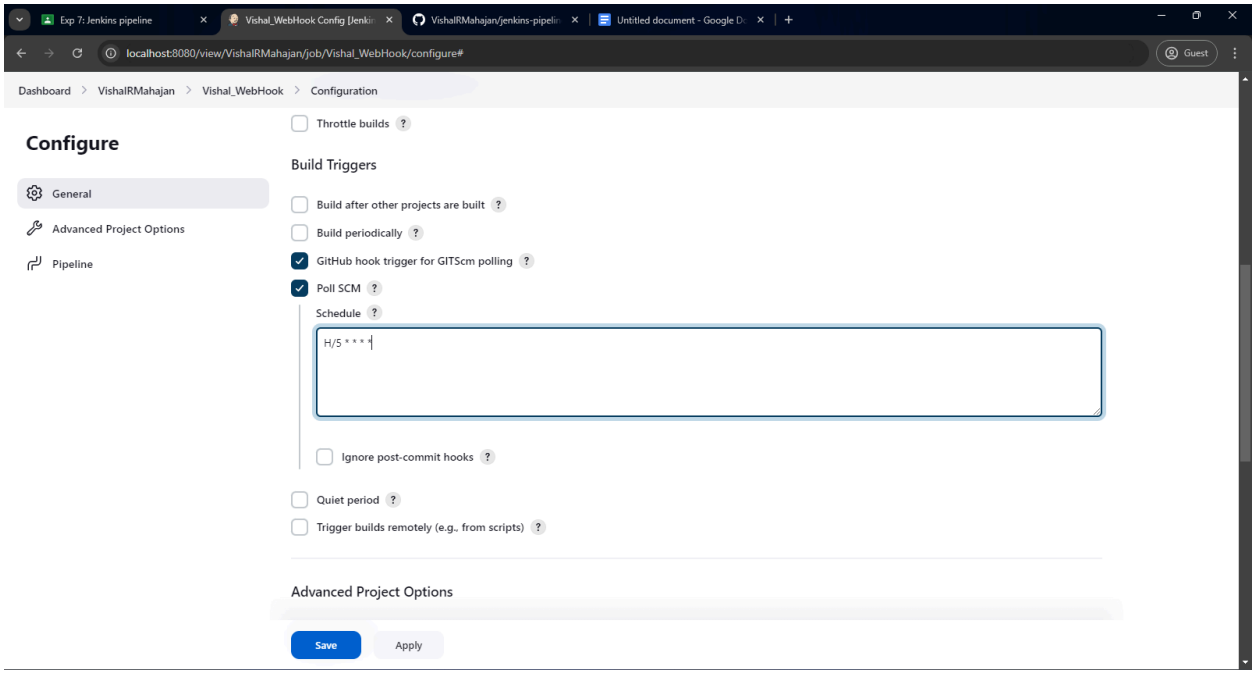
2. Create and build pipeline project with Git

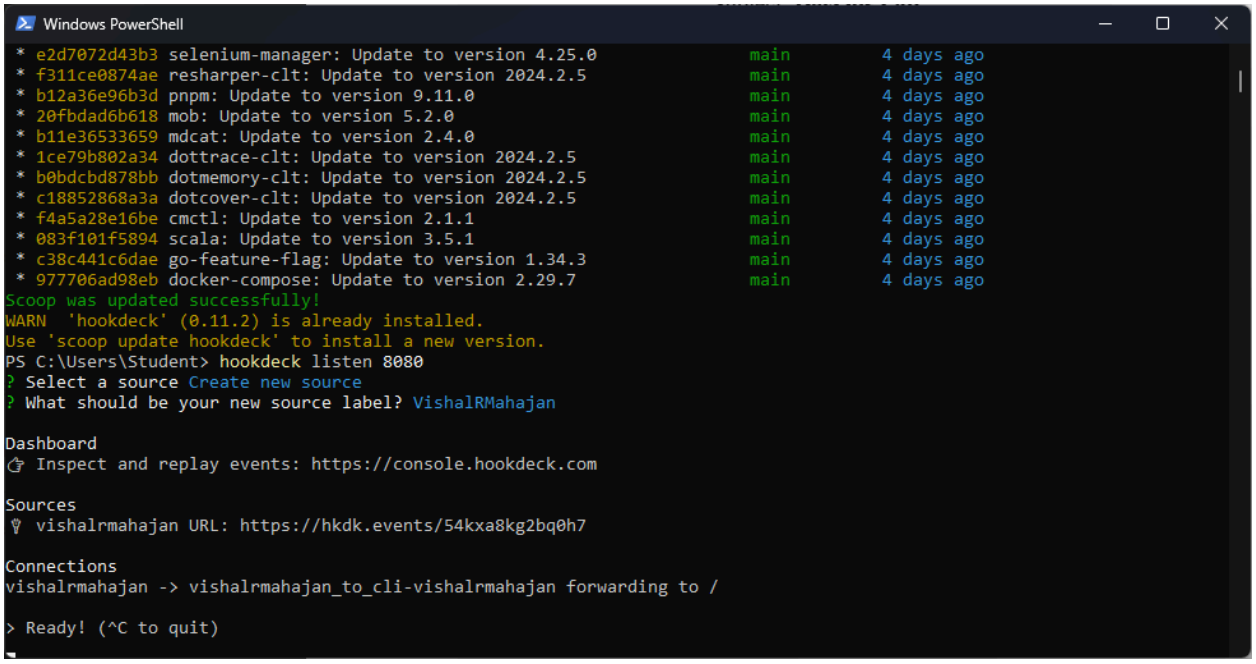
- **Fork repository on GitHub**



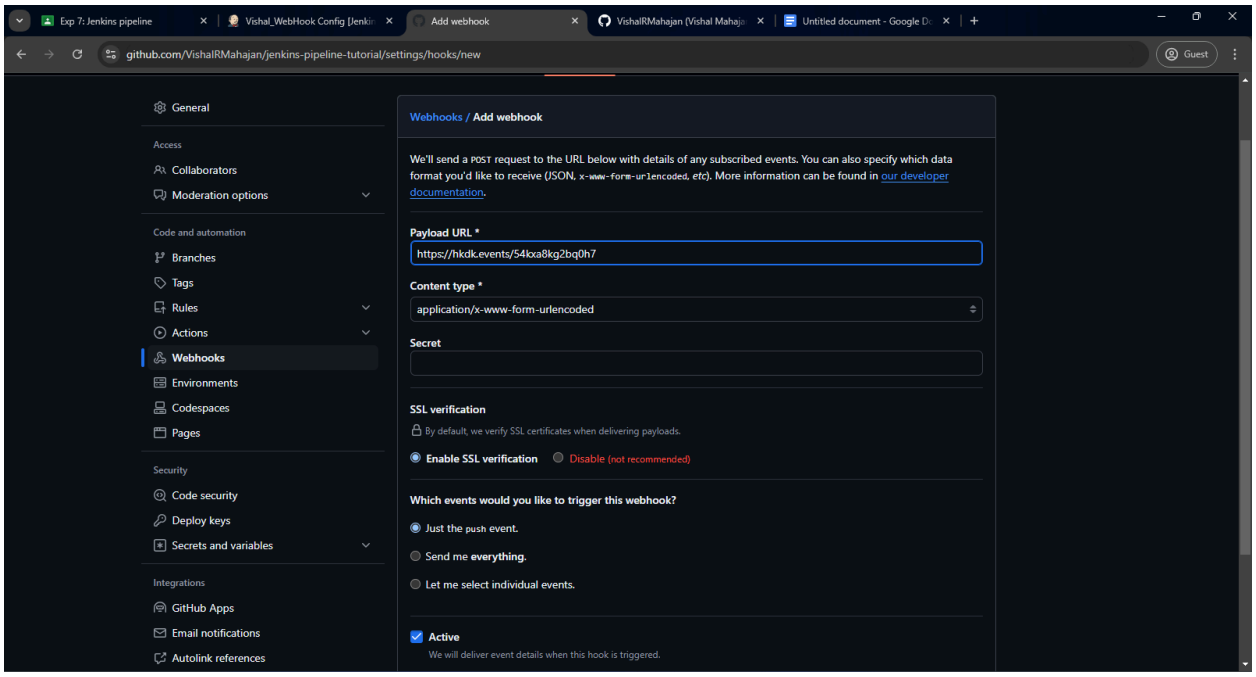
- **Create pipeline project with pipeline script from SCM**

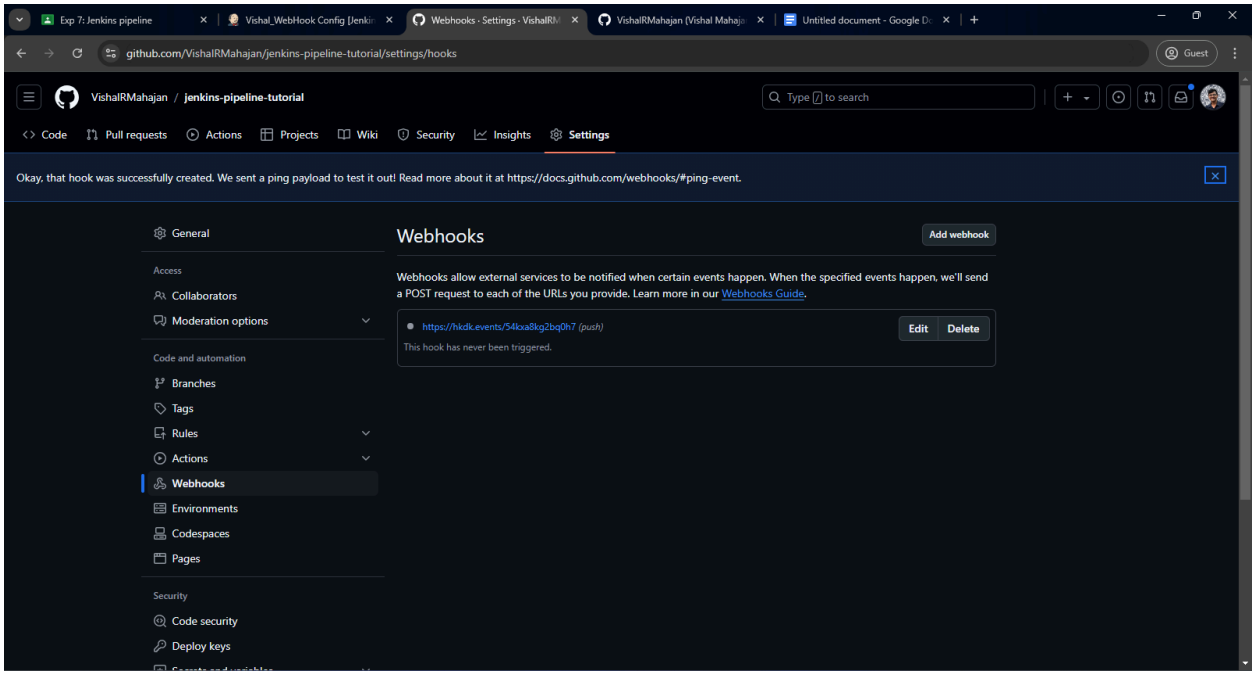



Exp 7: Jenkins pipeline
Vishal_WebHook Config [Jenkin
VishalRMahajan/jenkins-pipeli
Untitled document - Google D
localhost:8080/view/VishalRMahajan/job/Vishal_WebHook/configure#
Guest
Dashboard > VishalRMahajan > Vishal_WebHook > Configuration
Configure
General
Advanced Project Options
Pipeline
Throttle builds
Build Triggers
Build after other projects are built
Build periodically
GitHub hook trigger for GITScm polling
Poll SCM
Schedule
H/5 * * * *
Ignore post-commit hooks
Quiet period
Trigger builds remotely (e.g., from scripts)
Advanced Project Options
Save
Apply

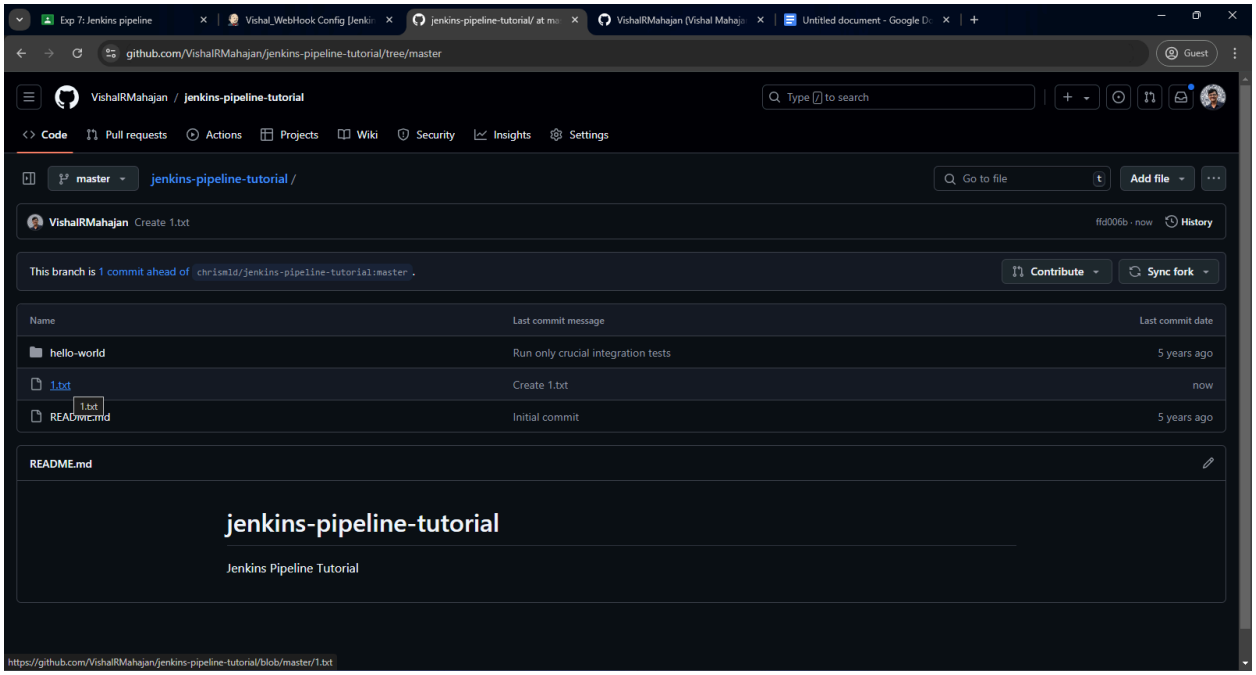

Windows PowerShell
* e2d7072d43b3 selenium-manager: Update to version 4.25.0 main 4 days ago
* f311ce0874ae resharper-clt: Update to version 2024.2.5 main 4 days ago
* b12a36e96b3d pnpm: Update to version 9.11.0 main 4 days ago
* 20fbdad6b618 mob: Update to version 5.2.0 main 4 days ago
* b11e36533659 mdcat: Update to version 2.4.0 main 4 days ago
* 1ce79b802a34 dottrace-clt: Update to version 2024.2.5 main 4 days ago
* b0bdcbd878bb dotmemory-clt: Update to version 2024.2.5 main 4 days ago
* c18852868a3a dotcover-clt: Update to version 2024.2.5 main 4 days ago
* f4a5a28e16be cmctl: Update to version 2.1.1 main 4 days ago
* 083f101f5894 scala: Update to version 3.5.1 main 4 days ago
* c38c441c6dae go-feature-flag: Update to version 1.34.3 main 4 days ago
* 977706ad98eb docker-compose: Update to version 2.29.7 main 4 days ago
Scoop was updated successfully!
WARN 'hookdeck' (0.11.2) is already installed.
Use 'scoop update hookdeck' to install a new version.
PS C:\Users\Student> hookdeck listen 8080
? Select a source Create new source
? What should be your new source label? VishalRMahajan
Dashboard
Inspect and replay events: https://console.hookdeck.com
Sources
vishalrmahajan URL: https://hkdk.events/54kxa8kg2bq0h7
Connections
vishalrmahajan -> vishalrmahajan_to_cli-vishalrmahajan forwarding to /
> Ready! (^C to quit)

- **Add webhooks to the forked repository**



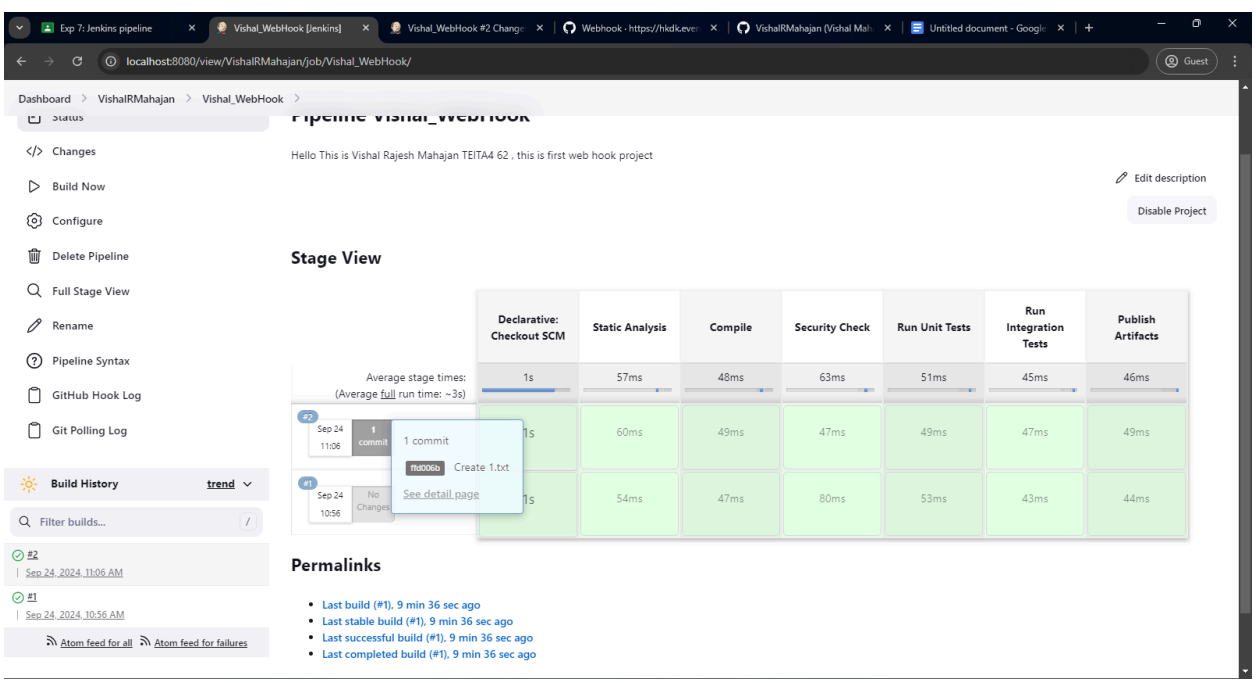


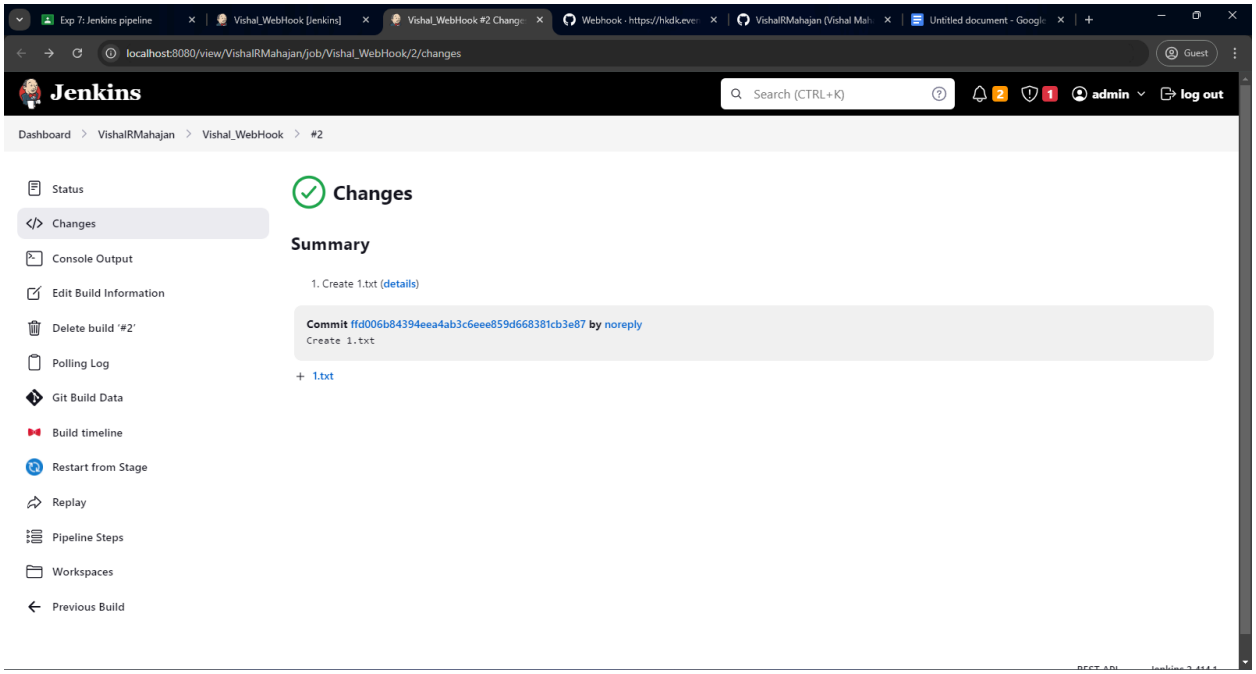
- **Add file to forked repository and observe the automated build**



,

- **Make changes to Jenkins file on forked repository and observe the automated build**




Exp 7: Jenkins pipeline
Vishal_WebHook [Jenkins]
Vishal_WebHook #2 Change
Webhook · https://hkdk.ever
VishalRMahajan (Vishal Mah
Untitled document - Google
localhost:8080/view/VishalRMahajan/job/Vishal_WebHook/2/changes
Guest
Jenkins
Search (CTRL+K)
admin
log out
Dashboard > VishalRMahajan > Vishal_WebHook > #2
Status
Changes
Console Output
Edit Build Information
Delete build '#2'
Polling Log
Git Build Data
Build timeline
Restart from Stage
Replay
Pipeline Steps
Workspaces
Previous Build
Changes
Summary
1. Create 1.txt (details)
Commit ffd006b84394eea4ab3c6eee859d668381cb3e87 by noreply
Create 1.txt
+ 1.txt