

# Multibranch Pipeline with a Webhook on Jenkins

Sometimes it may be necessary to create a pipeline on Jenkins for each Git branch. In this case it could be difficult to create independent pipeline for each branch. Besides that what if we create or delete a branch in future? So someone has to take care of the pipelines on Jenkins whenever there is change in branches. that's where Multibranch pipeline comes into picture.

Through Multibranch pipeline we can create a pipeline for each branch in a repository and it also create or remove when there is a change in the branches.

## **Pre-Requisites:**

1. Expected a Jenkins server is already up and running
2. A Git repository with more than one branch

## **Create a Multibranch Pipeline**

1. Login to Jenkins GUI
2. Click on “New Item” → Specify a job name → Select “Multibranch Pipeline option”

Enter an item name

cantire

> Required field

- Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.
- Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
- Folder**  
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
- Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected branches in one SCM repository.
- Organization Folder**  
Creates a set of multibranch project subfolders by scanning for repositories.

If you want to create a new item from other existing, you can use this option:

OK

### 3. Give Display Name and Description

4. Under Branch Sources → Add source → Chose Git → and provide GitHub URL and Credentials (Credentials are optional if it is public repo)

Dashboard > cantire > Configuration

**Configuration**

- General
- Branch Sources
- Build Configuration
- Scan Multibranch Pipeline Triggers
- Orphaned Item Strategy
- Appearance
- Health metrics
- Properties

**General** Enabled

Display Name ?  
cantire Project

Description  
TMS  
[Plain text] Preview

**Branch Sources**

**Git**  
Project Repository ?  
https://github.com/mancj701m/hellowebapp.git

Credentials ?  
- none -  
+ Add

Save Apply

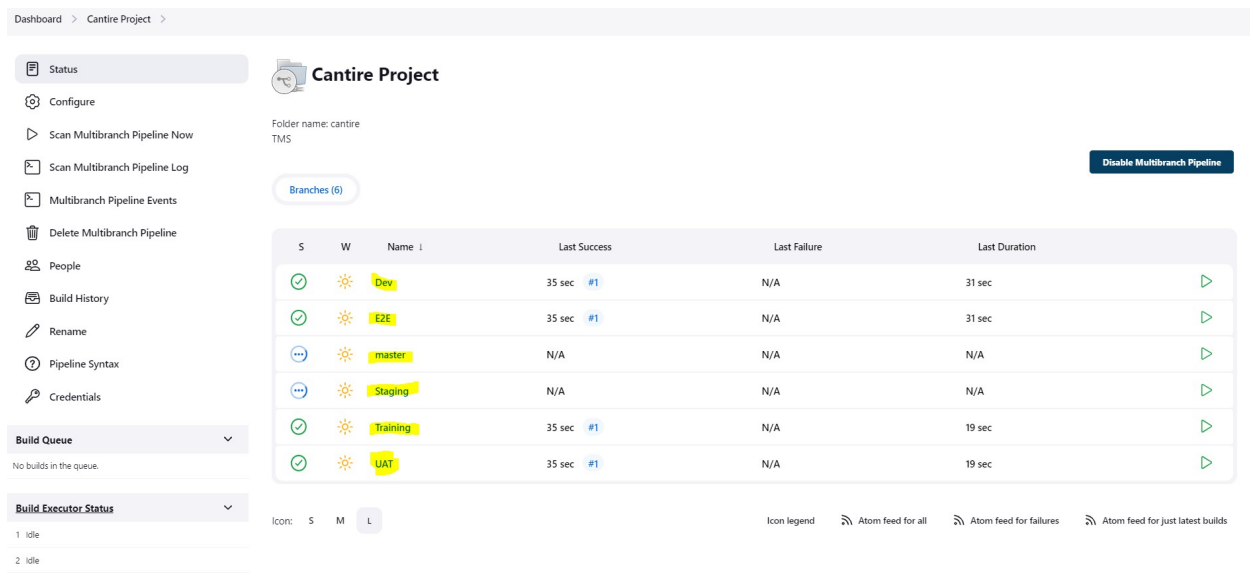
5. Under Build Configuration → chose Jenkinsfile path. Most of the case it will be under Repo root directory.



The screenshot shows the 'Build Configuration' section of the Jenkins job configuration page. It features a 'Mode' dropdown menu set to 'by Jenkinsfile' and a 'Script Path' text input field containing 'Jenkinsfile'. A blue help icon is visible to the right of the input field.

6. Apply and Save the job

Now Jenkins automatically scans the repository and create a job for each branch wherever it finds a Jenkinsfile and initiate first build.



The screenshot displays the Jenkins dashboard for the 'Cantire Project'. The left sidebar contains navigation links for Status, Configure, and various pipeline management options. The main area shows the project's configuration, including the folder name 'cantire' and a 'TMS' label. A 'Disable Multibranch Pipeline' button is present. Below this, a 'Branches (6)' section displays a table of branches.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	Dev	35 sec #1	N/A	31 sec
✓	☀	E2E	35 sec #1	N/A	31 sec
...	☀	master	N/A	N/A	N/A
...	☀	Staging	N/A	N/A	N/A
✓	☀	Training	35 sec #1	N/A	19 sec
✓	☀	UAT	35 sec #1	N/A	19 sec

At the bottom, there is an 'Icon legend' and 'Atom feed' links for all, failures, and latest builds. The 'Build Queue' and 'Build Executor Status' sections on the left show no builds in the queue and two idle executors.

## Using Webhook

If you wish to automate the build process in the multibranch pipeline we can use Webhook. This feature is not enabled until we install “Multibranch Scan Webhook Trigger” plugin. This enables an

option “scan by webhook” under “Scan Multibranch Pipeline Triggers”. Here we should give a token.

#### Scan Multibranch Pipeline Triggers

☐ Periodically if not otherwise run ?

☒ Scan by webhook ?

Trigger token ?

cantiretoken

The token to match with webhook token. Receive any HTTP request, JENKINS\_URL/multibranch-webhook-trigger/invoke?token=[Trigger token] If a token match, then a multibranch scan will be triggered.

(from [Multibranch Scan Webhook Trigger](#))

Now to enable auto build process we should provide Jenkins URL with token in the GitHub. <http://20.239.82.80:8080/multibranch-webhook-trigger/invoke?token=cantiretoken>

for this log into GitHub → settings → Webhooks → Add webhook

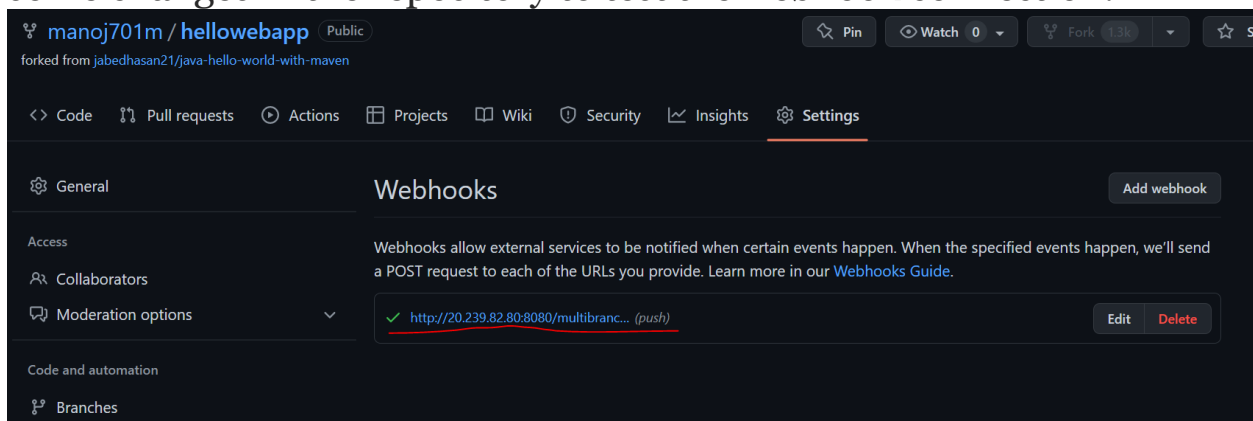
The screenshot shows the GitHub repository settings for 'manoj701m / hellowebapp'. The 'Settings' tab is selected, and the 'Webhooks' section is active. The 'Add webhook' form is displayed with the following configuration:

- Payload URL:** `http://20.239.82.80:8080/multibranch-webhook-trigger/invoke?tok`
- Content type:** `application/json`
- Secret:** (Empty text box)
- Which events would you like to trigger this webhook?**
  - ☒ Just the push event.
  - ☐ Send me everything.
  - ☐ Let me select individual events.
- Active:** ☒ (We will deliver event details when this hook is triggered.)


The 'Add webhook' button is highlighted with a red checkmark.

Provide Payload URL as “**http://20.239.82.80:8080/multibranch-webhook-trigger/invoke?token=cantiretoken**” and Content type as “**application/json**” and click on Add webhook

Once this is done you can see a new webhook and its time to do some changes in the repository to test the webhook connection.



In this, Now push the changes onto remote repo (master). Now you could see a build has been triggered automatically on Jenkins and it scan and create a job for the new branch as well.

 **Cantire Project**

Folder name: cantire  
TMS

[Disable Multibranch Pipeline](#)

Branches (6)

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	Dev	14 min #1	N/A	31 sec
✓	☀	E2E	14 min #1	N/A	31 sec
✓	☀	master	16 sec #2	N/A	7.8 sec
✓	☀	Staging	14 min #1	N/A	43 sec
✓	☀	Training	14 min #1	N/A	19 sec
✓	☀	UAT	14 min #1	N/A	19 sec

Icon: S M L

Icon legend: Atom feed for all Atom feed for failures Atom feed for just latest builds

# Branches :

manoj701m / hellowebappPublicforked from javedhasan21/java-hello-world-with-maven

<> Code

⇄ Pull requests

🔄 Actions

📁 Projects

📖 Wiki

🔒 Security

📊 Insights

⚙ Settings

🔍 Search branches...

OverviewYoursActiveStaleAll branchesNew branch

All branches

master	Updated 26 minutes ago by manoj701m	Default	
uat	Updated 26 minutes ago by manoj701m	0   0	New pull request
Training	Updated 26 minutes ago by manoj701m	0   0	New pull request
Staging	Updated 26 minutes ago by manoj701m	0   0	New pull request
E2E	Updated 26 minutes ago by manoj701m	0   0	New pull request
Dev	Updated 26 minutes ago by manoj701m	0   0	New pull request