**FLOW DIAGRAM FOR AUTOMATED DEPLOYMENT USING JENKINS:**

GIT

JENKINS

REMOTE SERVER

W webhooks pipeline

**Description:**

**----------------**

Here we are using GITHUB as a VCS (version control system) tool, where we are storing our code, Jenkins as CI-CD tool by which we are compiling, building the code and deploying the artefact to a remote server.

**Integration:**

**----------------**

For GitHub and Jenkins integration we are using GitHub webhooks like once any code changes in the GitHub repository it will trigger to Jenkins and Jenkins will start building the code with latest changes, artefact will be pushed to remote server, here we are using a another server as a remote where are storing all our build artifact, that is configured in **Publish Over SSH** section.

**Prerequisites:**

**------------------**

* Jenkins should be working properly with below mentioned plugin installed

1. Publish Over SSH: for pushing the artifact to the remote server, a server name with remote should be configured in publish over section under MANAGE JENKINS 🡪 CONFIGURE SYSTEM.
2. Pipeline: For automate you Jenkins job as a code (pipeline as a code)
3. GitHub Plugin: for integration of Jenkins to GitHub.

**Pipeline Script:**

**--------------------**

In our pipeline we are having four stages as described below:

1. Checkout SCM:

In this stage we are cloning GitHub repo if any changes made by anyone in master branch.

1. Compile:

In this stage we are compiling our code using **javac** command and storing all the .class files in bin directory.

1. Build:

In this stage we build our code and creating a executable jar file using jar cvfm command and storing the jar file in bin folder itself.

1. Publish Artifact:

In this stage we publishing the artifact that is jar file only to a remote server in the target directory provided by user in the parameter section.

Apart from these stages we have other steps mentioned below:

1. We are using java datetime library for current date time.
2. We are using parameter block for string because we have to ask user to provide target directory for storing the artifacts and if user does not provide any input then it will pick the default value.

Pipeline Execution scenarios:

1. When pipeline run automatically by webhook:

This will be the automatic process when any new change comes in repository, pipeline will be triggered. In this case remote directory path will be picked default by the job because this is automatic pipeline execution.

1. When user triggers the pipeline:

This will be the manual trigger and user can provide remote directory path where the artifacts will be stored.

A screenshot of a cell phone

Description automatically generated

**Pipeline Code:**

**------------------**

// Define variable

import java.time.LocalDateTime

final LocalDateTime currentTime = LocalDateTime.now()

pipeline {

agent any

parameters {

string (

name: 'file\_path',

defaultValue: 'artifacts',

description: 'Please left as default or pass a remote directory.')

}

stages {

/\* checkout repo \*/

stage('Checkout SCM') {

steps {

checkout([

$class: 'GitSCM',

branches: [[name: 'master']],

userRemoteConfigs: [[

url: 'https://github.com/AMXjavaproject/firstjavaproject.git',

]]

])

}

}

stage('Compile') {

steps {

sh 'javac -d bin/ src/\*.java'

}

}

stage('Build') {

steps {

sh 'cd bin/ ; jar cvfm Library.jar MANIFEST.MF \*.class'

}

}

stage('Publish Artifact') {

steps([$class: 'BapSshPromotionPublisherPlugin']) {

sshPublisher(

continueOnError: false, failOnError: true,

publishers: [

sshPublisherDesc(

configName: "remote",

verbose: true,

transfers: [

sshTransfer(sourceFiles: "\*\*/Library.jar"),

sshTransfer(execCommand: "mkdir -p $file\_path/$currentTime ; mv bin/Library.jar $file\_path/$currentTime ; rm -rf bin ")

]

)

]

)

}

}

}

}

**Some Snapshots:**

**----------------------**

1. **Webhook creation:**

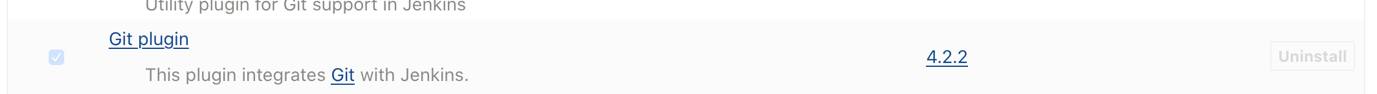
GitHub 🡪 repository 🡪 setting 🡪 webhooks 🡪 create new webhook 🡪 project URL: <Jenkins URL>/GitHub-webhooks/

A screenshot of a cell phone

Description automatically generated

Webhook snapshot

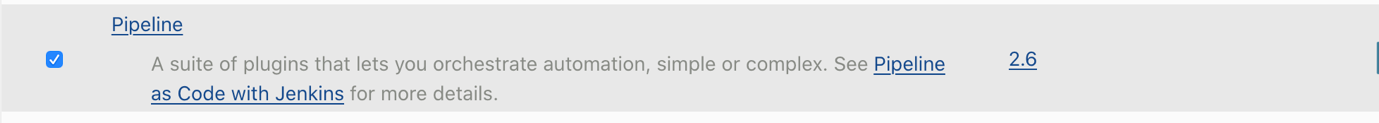
1. **Jenkins plugins:**

****

GitHub Plugin



Publish Over SSH



Pipeline plugin

1. **Jenkins Job:**

**A screenshot of a cell phone

Description automatically generated**

1. **Console Output:**

**A screenshot of a social media post

Description automatically generated**

**Logging Snapshots:**

**-------------------------**

**In our pipeline we are having four stages and below are some snapshots for each individual pipeline stages logs.**

1. **Checkout SCM:**

**A screenshot of a computer

Description automatically generated**

1. **Compile:**

**A screenshot of a computer

Description automatically generated**

1. **Build:**

A screenshot of a cell phone

Description automatically generated

1. **Publish Artifacts**

**A screenshot of a computer

Description automatically generated**

1. **Overall pipeline snapshot:**

**A screenshot of a cell phone

Description automatically generated**

**Reference**:

**-------------**

**GitHub repo(public):** https://github.com/AMXjavaproject/firstjavaproject

**Jenkins URL:** 52.15.238.5:8080/ (user: admin\_amx, pass: admin\_amx, job\_name = first\_java\_project)