

Software Quality Assurance

Course Project

Section A

Zaid Mustafa

22-10249

Hatib Zubair

241547206

Taimoor Malik

221434839

[Jenkins Installation 3](#_Toc112780697)

[Step 1 3](#_Toc112780698)

[Step 2 3](#_Toc112780699)

[Step 3 3](#_Toc112780700)

[Step 4 3](#_Toc112780701)

[Step 5 3](#_Toc112780702)

[Step 6 3](#_Toc112780703)

[Creating CI/CD pipeline: Right configurations & Pipeline running 3](#_Toc112780704)

[Building Phase 3](#_Toc112780705)

[Step 1 3](#_Toc112780706)

[Step 2 3](#_Toc112780707)

[Step 3 3](#_Toc112780708)

[Step 4 4](#_Toc112780709)

[Step 5 4](#_Toc112780710)

[Step 6 4](#_Toc112780711)

[Step 7: Create a Jenkinsfile 4](#_Toc112780712)

[Testing Phase 5](#_Toc112780713)

[Jenkins Pipeline Script 6](#_Toc112780714)

# Jenkins Installation

## Step 1

* Got to <https://www.jenkins.io/download/> select the version and download.

## Step 2

* Go to download area and run it

## Step 3

* Click “Next” in the Jenkin Setup screen.

## Step 4

* Choose the path where you need to have the Jenkins occurrence introduced (default area is C:\Program Files (x86) \Jenkins), then, click on Next button.

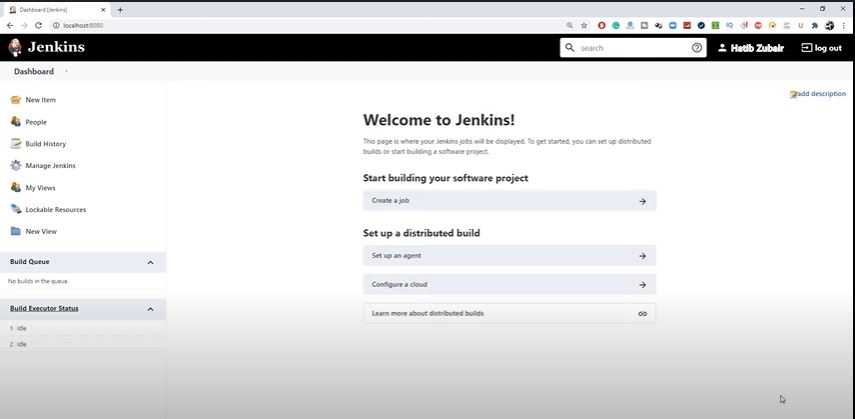
## Step 5

* Start installation by click on the Install button.

## Step 6

* Once installation is finished, click Finish.





# Creating CI/CD pipeline: Right configurations & Pipeline running

# Building Phase

## Step 1

* Opening Jenkins Login to Jenkins and click on “New Item.”

## Step 2

* Select the “Pipeline” option from the menu, provide a name for the pipeline, and click “OK.”

## Step 3

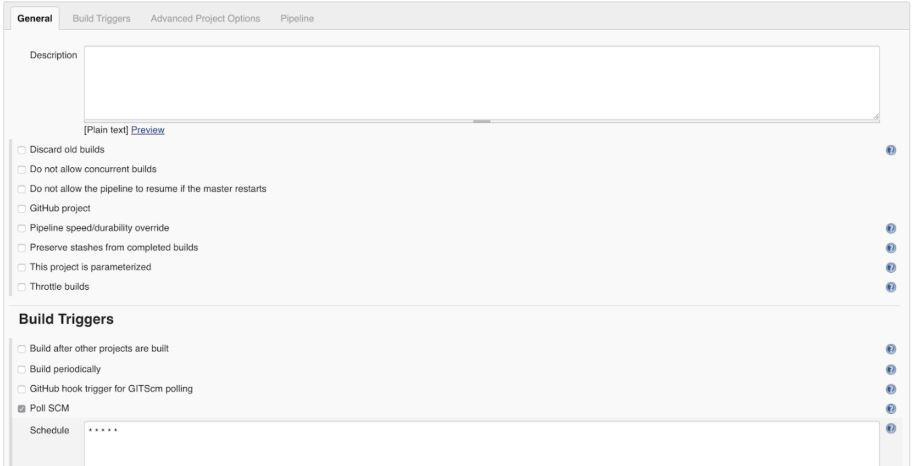
* We can configure the pipeline in the pipeline configuration screen. There, we can set build triggers and other options for the pipeline

## Step 4

* Click on “Build Now” to execute the pipeline.

## Step 5

* Click on the “Configure” option and change the pipeline definition according to the code block. The parameters that we modified are as follows:



* We used the Poll SCM as the form trigger; selecting this option teaches Jenkins to check the Git store on an ad hoc basis (as repeatedly demonstrated by \* \* \* \* \*). The occupation is activated if the repo has changed since the last survey.
* We chose the archive URL and the requirements for the actual pipeline. Ace is the branch.
* In a Jenkinsfile that is stored in a vault with the code, we are adding all the work's code.

## Step 6

* Configure Jenkins Credentials for GitHub
* Given that you will refer to each later, make sure you give them each a meaningful ID and description. Give credentials to the 2 targets by going to the accreditations.

## Step 7: Create a Jenkinsfile

* The information in the Jenkinsfile covers our application's assembling, testing, containerizing, distributing, and communicating aspects.

Jenkinsfile (Declarative Pipeline)

pipeline {

agent any

stages {

stage('Example') {

steps {

echo "Running ${env.BUILD\_ID} on ${env.JENKINS\_URL}"

}

}

}

}

* During the building process, we will make the Go parallel and check to see that the form cycle was completed successfully.
* The test is where some essential UAT testing is done to ensure that the app performs as anticipated.
* The Docker image is constructed and uploaded to the repository during the distribution process. After that point, it is suitable for application in any setting.
* Send in this direction. Ansible is alerted to make the appropriate connections so that Kubernetes may apply the definition records. It is done for the definition records to be used.

# Testing Phase

* As we get closer to this procedure's conclusion, we will look at every aspect of it very carefully. All of our work will be uploaded to GitHub, and we will make sure that it is communicated to the other members of the team through the protocols that have been established.
* Add our necessary documentation: the code for adding files to git is git add.
* Follow our updated procedures: "Beginning Commit" is the abbreviation for "m git commit."
* Simply typing "git push" will send your modifications to the public repository.
* Jenkins allows us to either wait for the work to be launched automatically or actively initiate it by selecting the "Form Now" button. Either way, we have the choice.
* In the case that the mission is completed successfully, the following command will allow us to review the application that we have submitted:
* kubectl get nodes -o wide as the command

"Construct" is one of the available choices for Stage 1, while "Assemble Information" is another. It continues throughout the rest of the Stage Blocks. The word "Step" can be broken down into its most elemental form to describe what should be done. It might be a specific print order such as echo "Hey, People", the execution of a program such as java HelloPeople, the execution of a shell command such as chmod 755 Hey, or anything else, such as long as it is considered to be executable within the Jenkins environment. The Jenkins pipeline is described in a standardized format commonly known as a Jenkinsfile, although the name of each file might be different. A straightforward example of a Jenkins pipeline log.

# Jenkins Pipeline Script

Jenkinsfile (Declarative Pipeline)

pipeline {

agent any

options {

skipStagesAfterUnstable()

}

stages {

stage('Build') {

steps {

sh 'make'

}

}

stage('Test’) {

steps {

sh 'make check'

junit 'reports/\*\*/\*.xml'

}

}

stage('Deploy') {

steps {

sh 'make publish'

}

}

}

}

Graphical user interface, text

Description automatically generated