**Problem Statement**

The objective of the assignment is the engineer to get the real time experience on Jenkins. The engineer need to learn and perform the below topics:

**1. Understanding Jenkins and its Overview – Going through e-learning contents.**

Jenkins is a cont. integration tool which increases the productivity of the developer in releasing application. With usage of integration, we can cache the build failure and can generate automatic build report notification to developers so that they can work on that failure and commit again after any changes . We can maintain easy track of any bugs at development stage than in production stage.

Pre-requisite for working with Jenkins is to install s/w which we require to work with i.e.,git, maven, tomcat, SonarQube etc. We just need to integrate these S/w with Jenkins via manage plugins and we are good to use. Pre-Build setup for Jenkins include: New Job🡪select free-style, maven project, pipeline etc.,🡪select optional SCM i.e., git, mercurial or subversion where we commits our source code repository🡪select optional triggers to control Jenkins build🡪some sort of build script to perform build🡪optional step to notify people with build result.

Post build contains additional steps to be followed once job or our pipeline is executed for example-any other build project to be followed after one job is executed or sending email notification to developers or archive the artifacts, create issues with jira.

**2. Jenkins Installation**

Pre-Requisite🡪We need to have jdk,JRE install(openJDK-8-jre) and need to set in environment variable.

By default Jenkins listen on port 8080.For installing into window we can refer Jenkins.io page from where we can download the jenkin war file and start jenkin which java –jar Jenkins.war command in command prompt ,embedded web server winstone will extract the war file and jenkin is up and running then followed by unlocking it with secret key which will be stored in /secret-key path and customize the Jenkins depending upon the requirement and finish button to complete installation.

On debian based distribution, such as ubuntu, we can install jenkin using apt-get.

**wget -q -O - https://pkg.jenkins.io/debian/jenkins-ci.org.key | sudo apt-key add –**

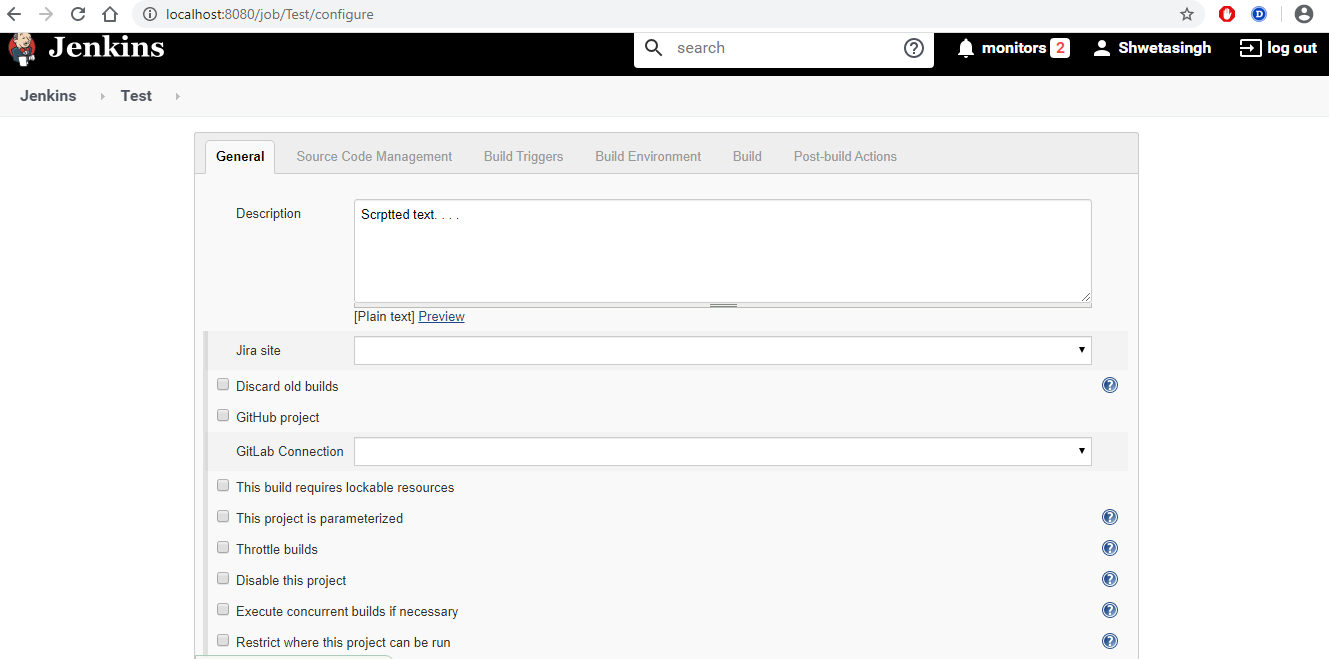
**sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'**

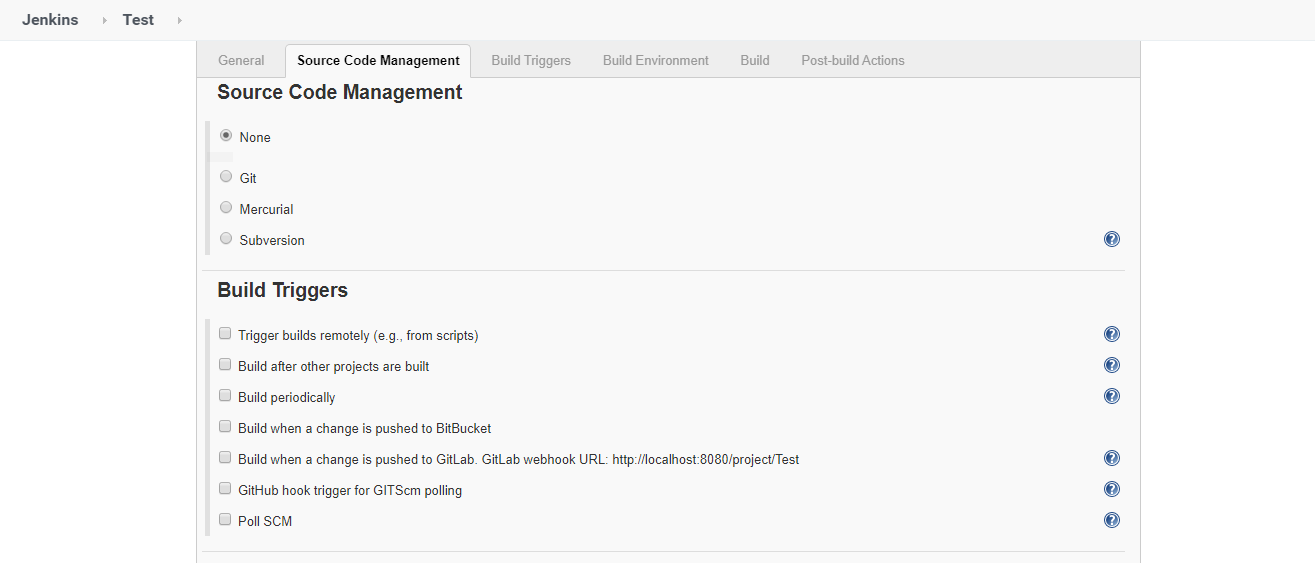
**sudo apt-get update**

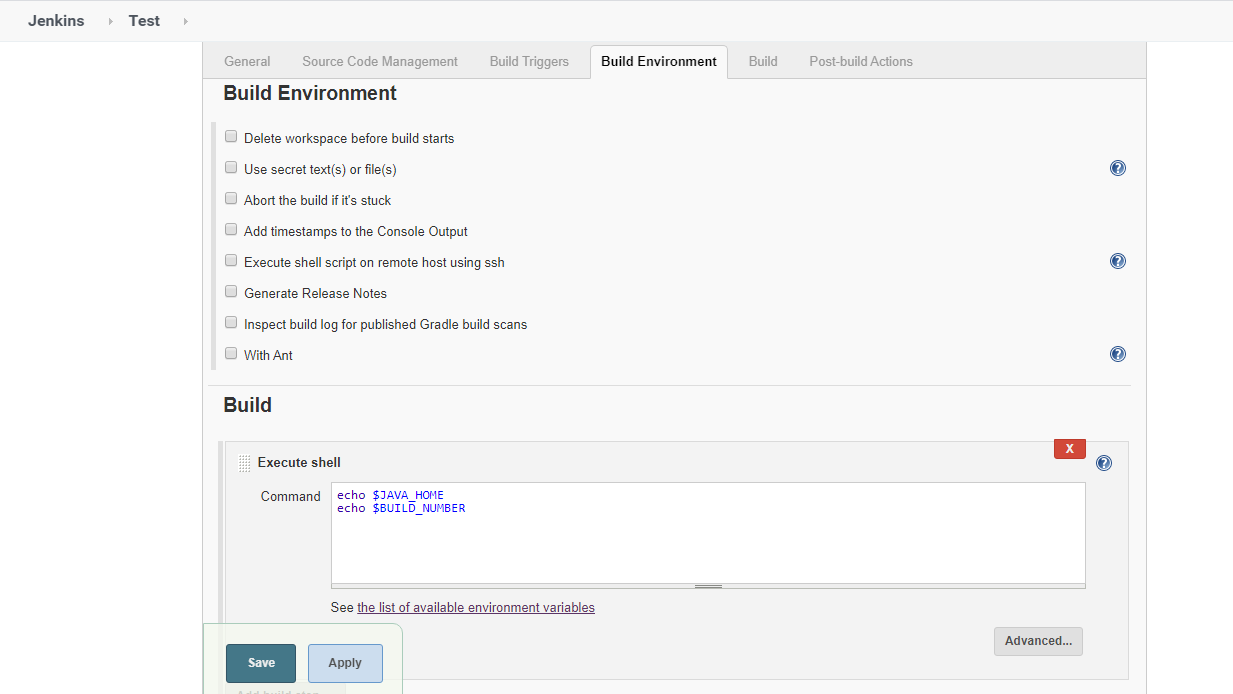
**sudo apt-get install Jenkins**

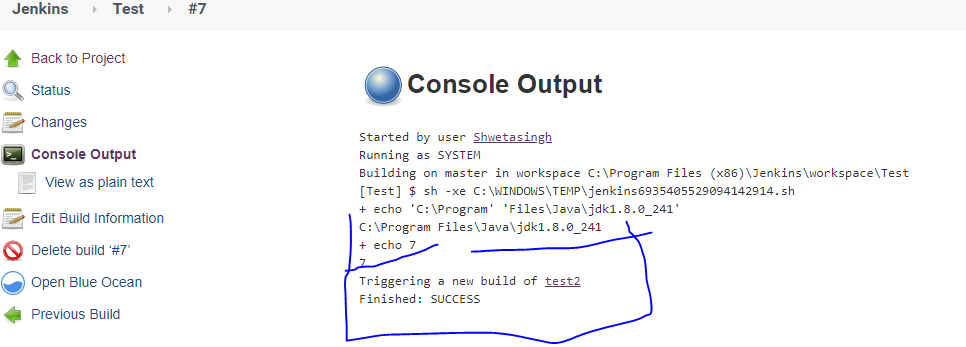
**3. Job creation in Jenkins**

Below screenshot of job creation in Jenkins:-

I have used free style project 🡪

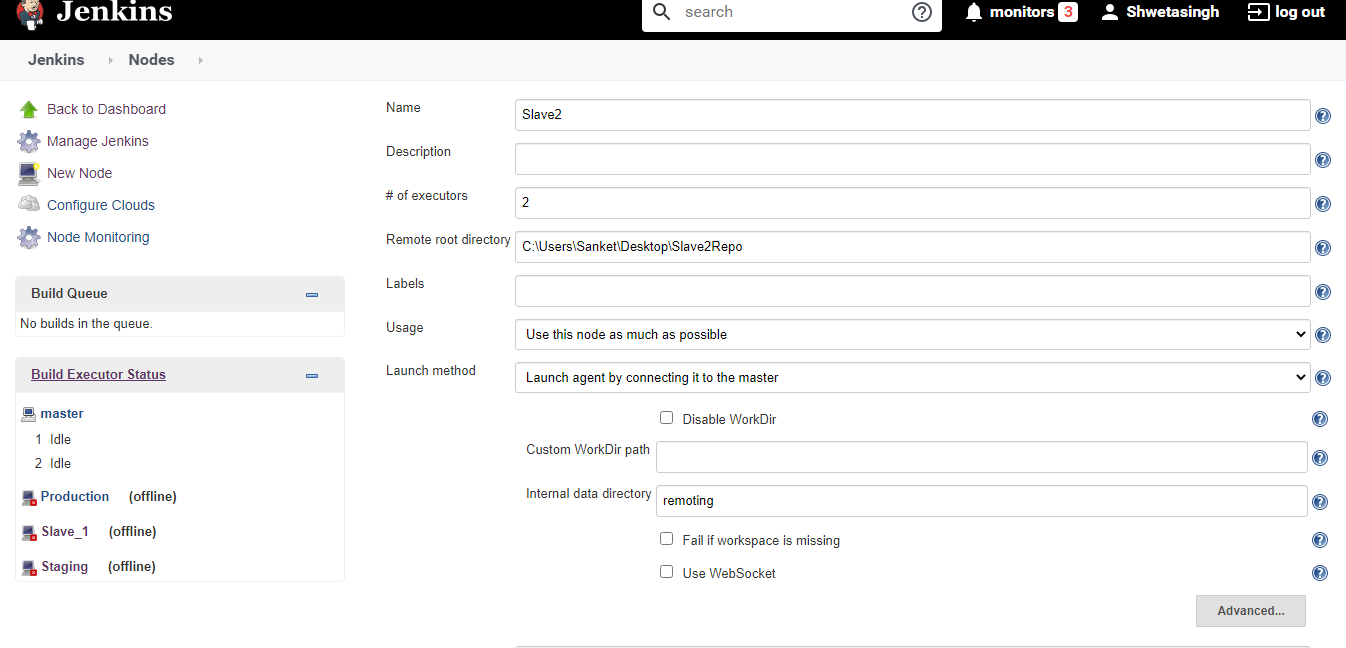
****

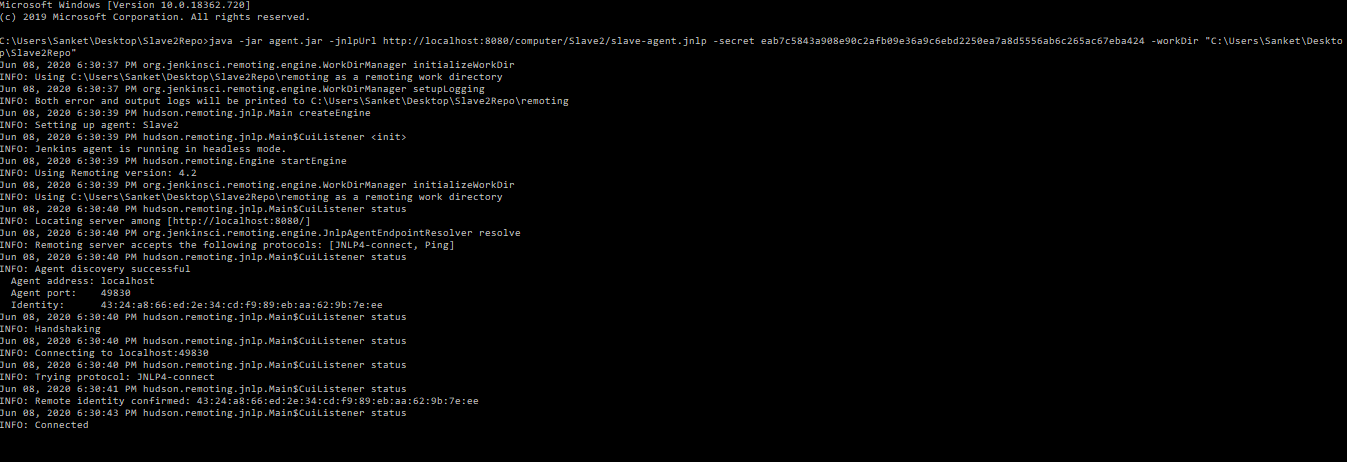


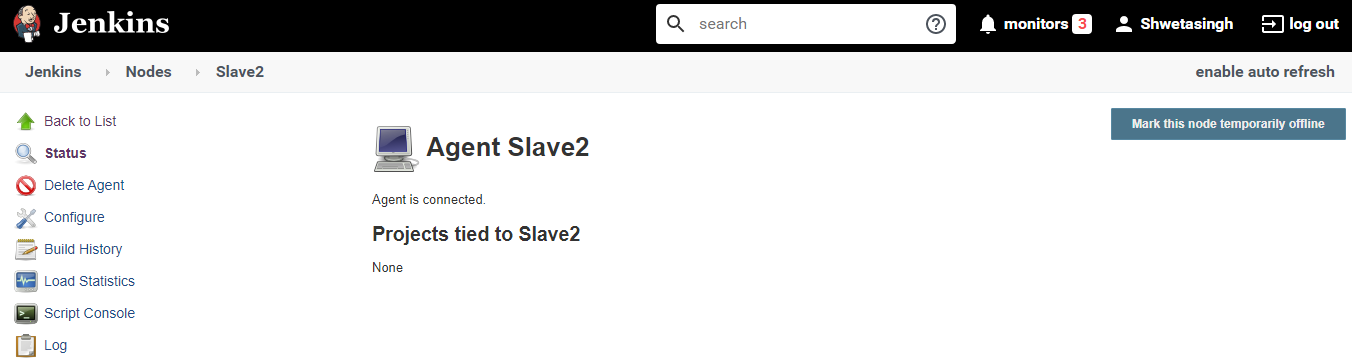


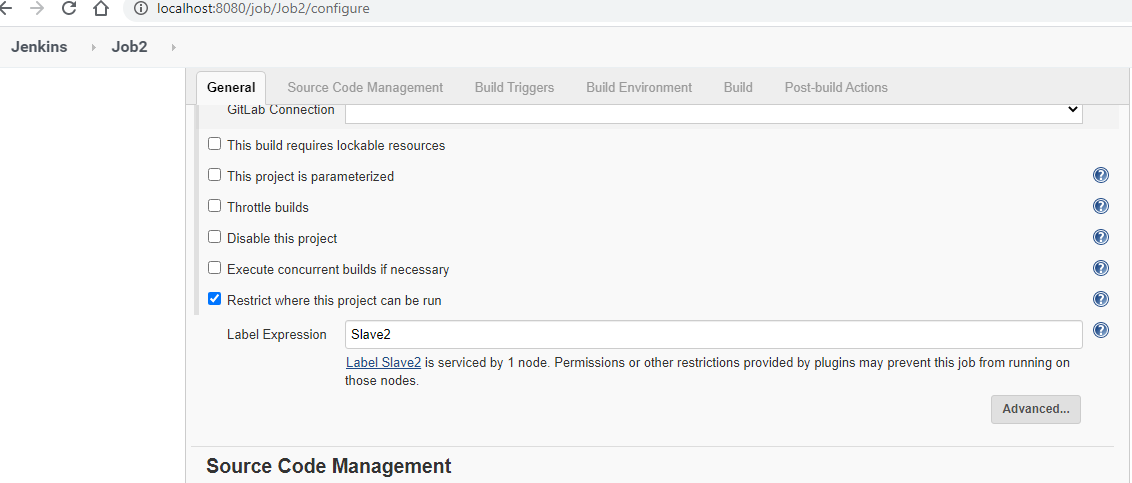
**4. Creating Slaves🡪**Jenkins master handles all the tasks for the build system. Suppose we have to run many projects, so we have to run multiple jobs on each and every project. So Jenkins provides us to manage nodes and run some jobs into it. Jenkins master connect its slave via Jenkin n/w launch protocol.

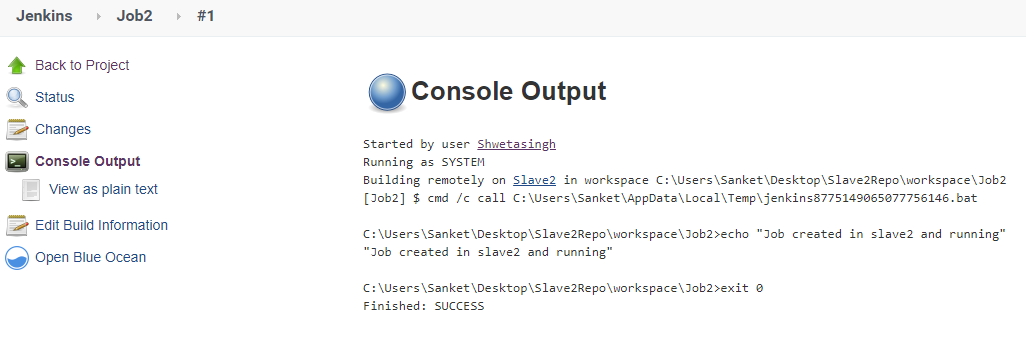
The Jenkins master acts to schedule the jobs and assign slaves and send builds to slaves to execute the jobs.

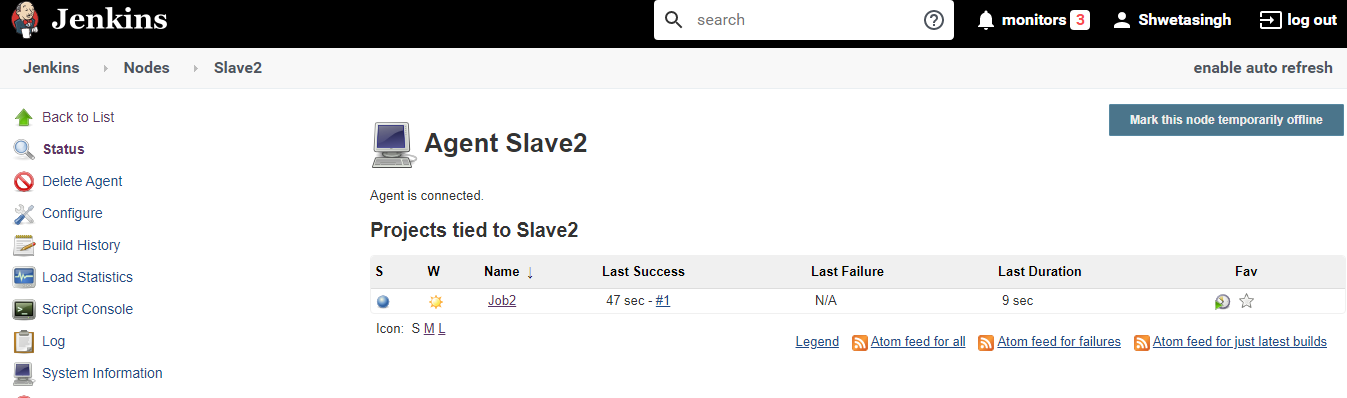
****

****

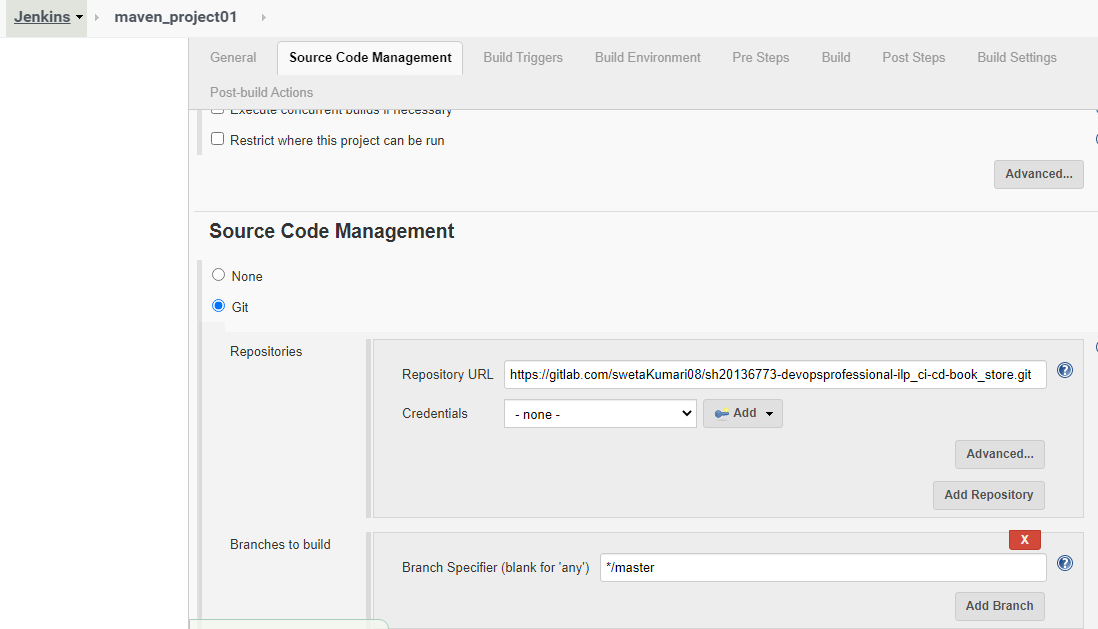
****

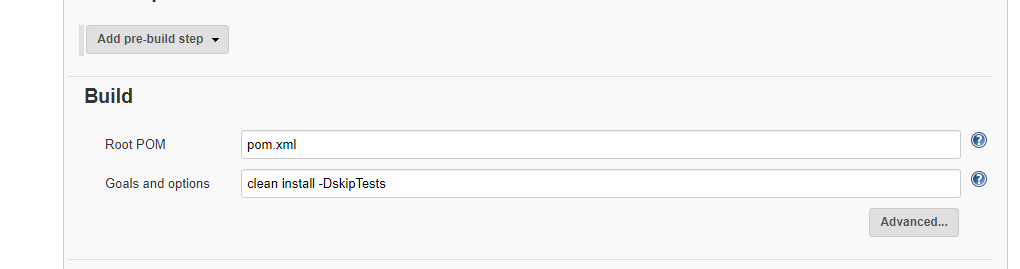
****

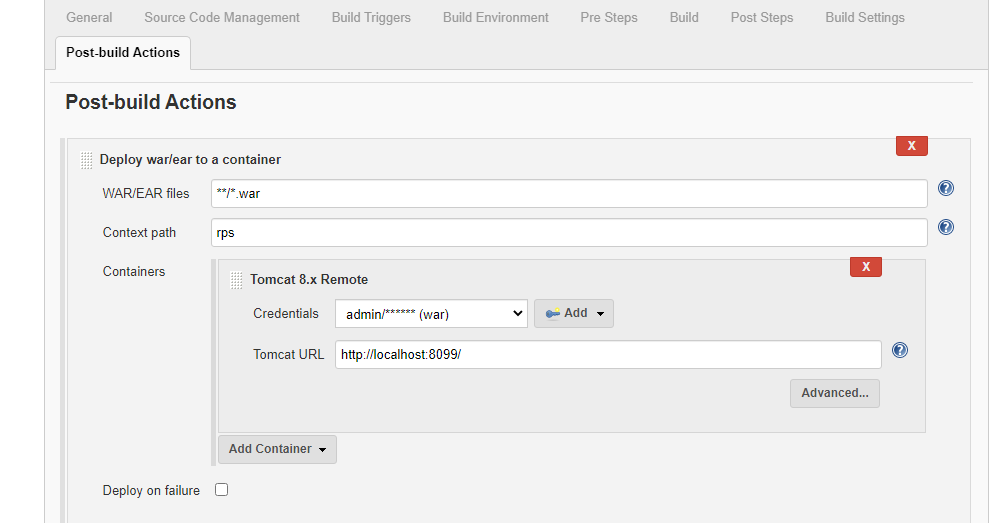
****

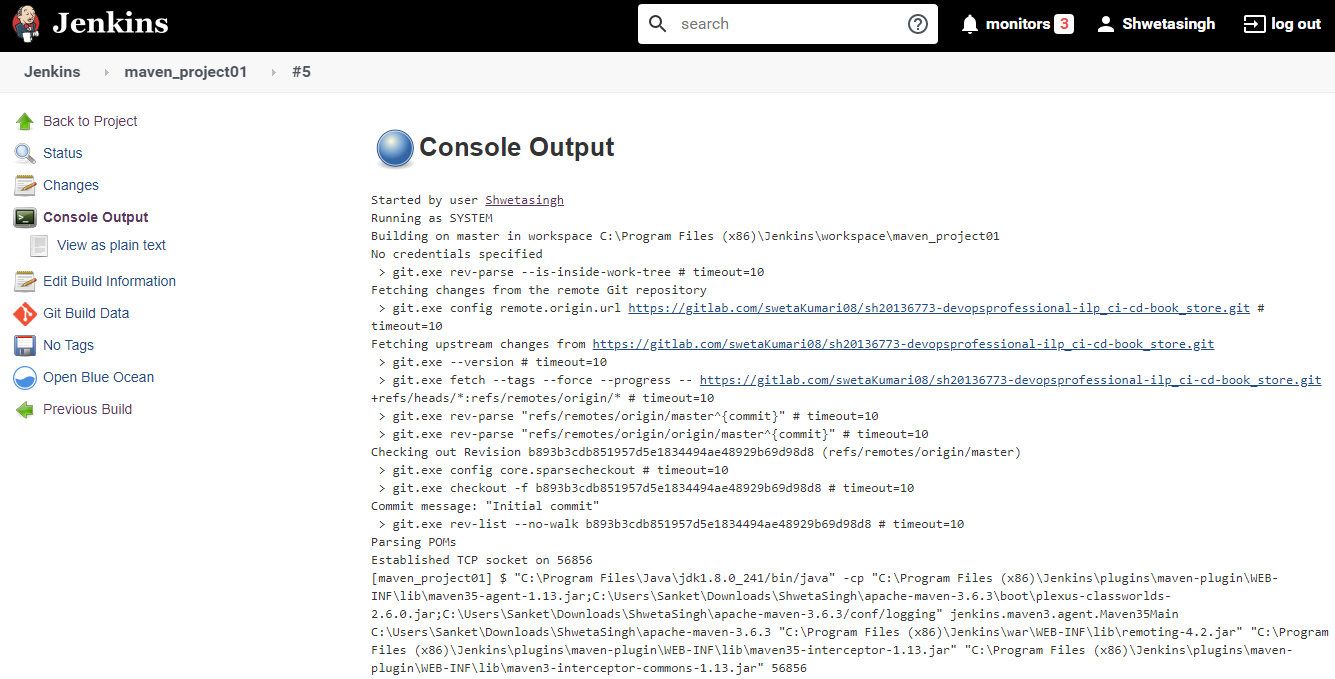
****

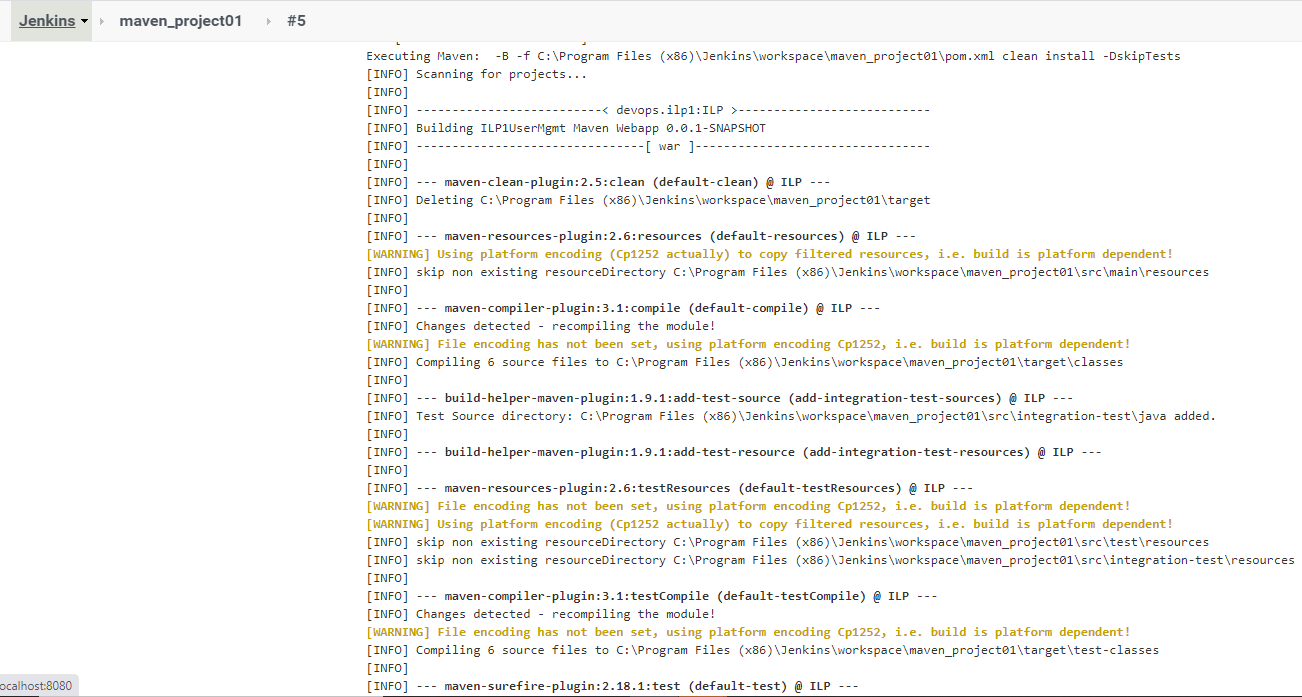
**5. Installing Plugins🡪Jenkins support 1000+ plugins for managing and handling a automated flow of CI/CD. For example- Git plugin to pull all source code commited by developer in GitHub. Maven plugin for builds. Deploy to container plugin for deploying an application into servers such as Tomcat etc.,**

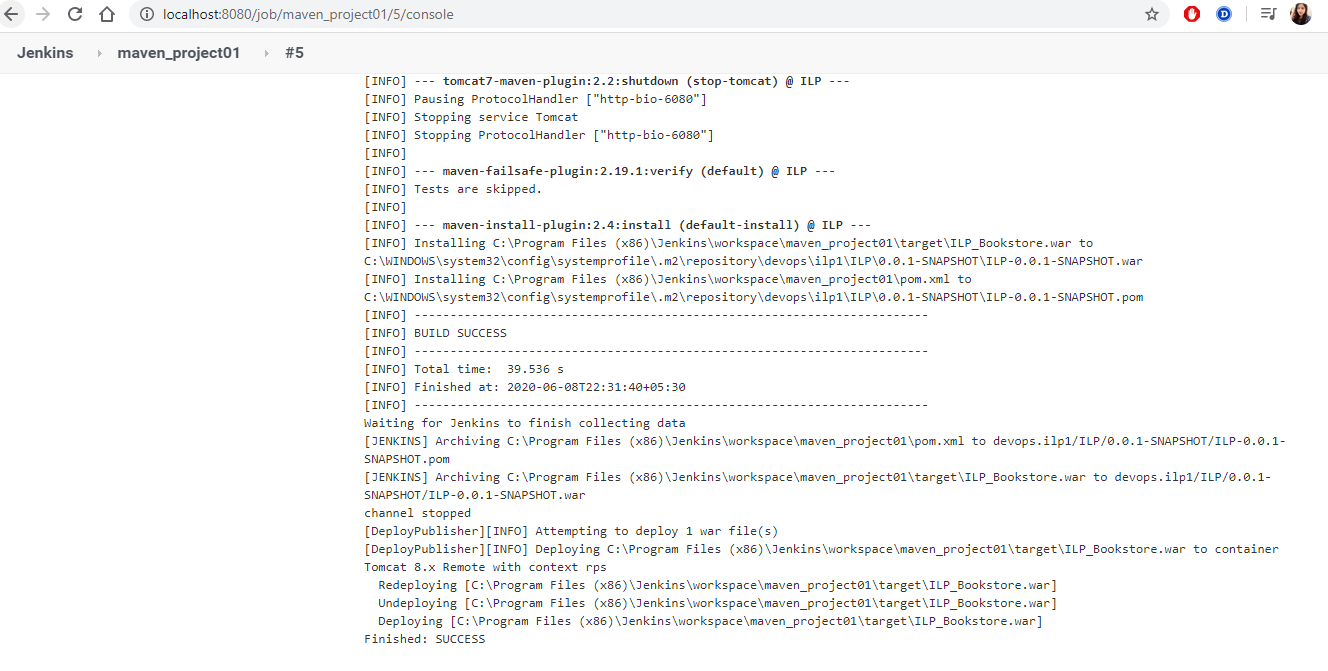
****

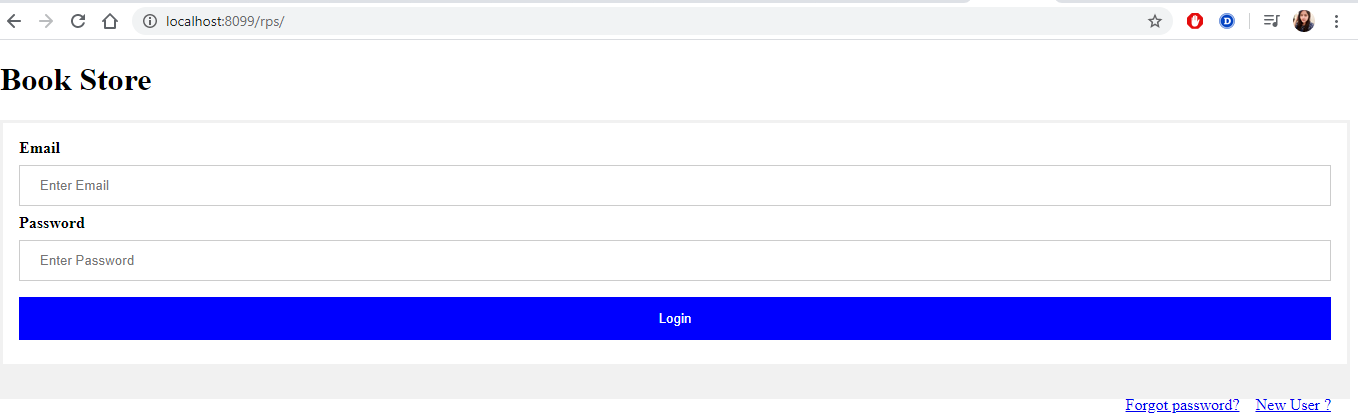
****

****

****

****

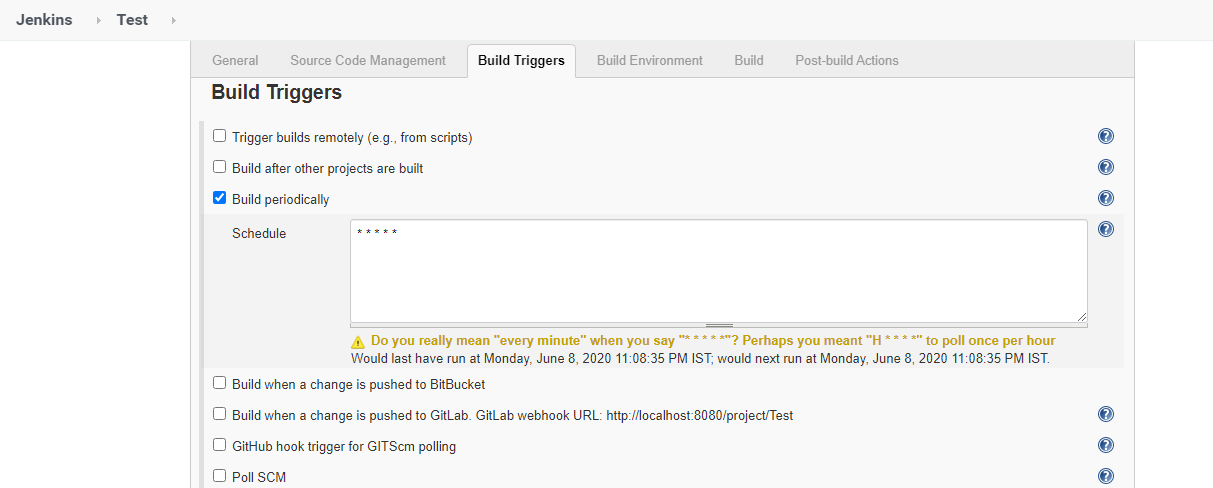
****

****

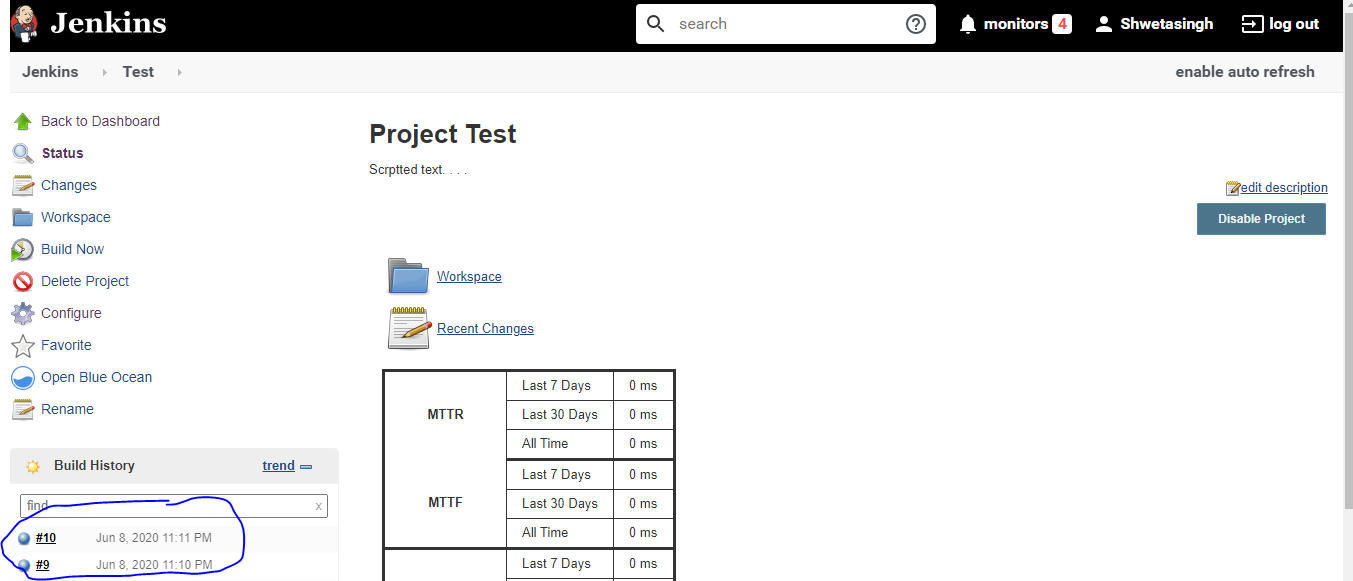
**6. Scheduling cron jobs In Jenkins🡪** it means tasks you need to schedule some repetitive jobs to trigger some pre-specified commands automatically; you can schedule any task or job to run after fixed hours or periodically throughout the day, number of times in a month, on specific days or dates etc.

In jenkin, we have build periodically and SCM pool to triggers our job as scheduled using CRON.

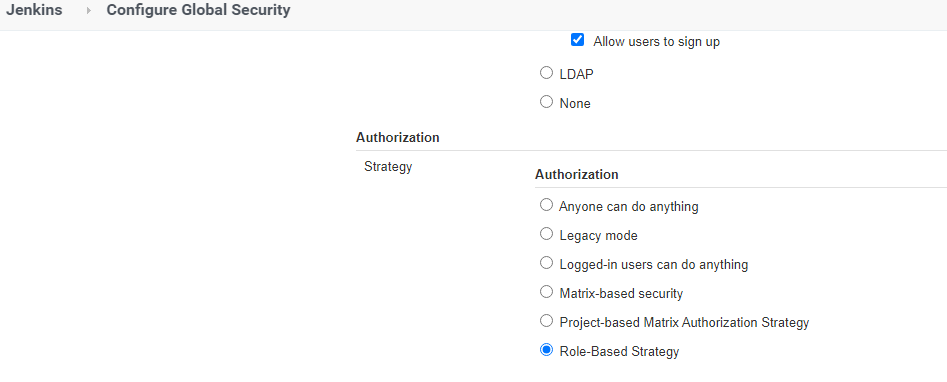
//Staged as {Minutes}{Hour}{DayOfMonth}{Month}{DayofWeek}

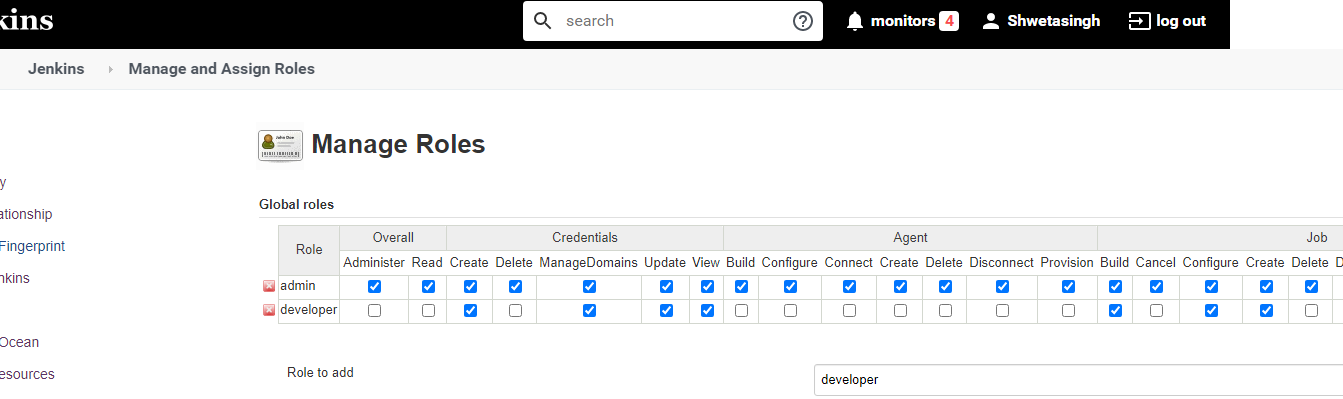
****

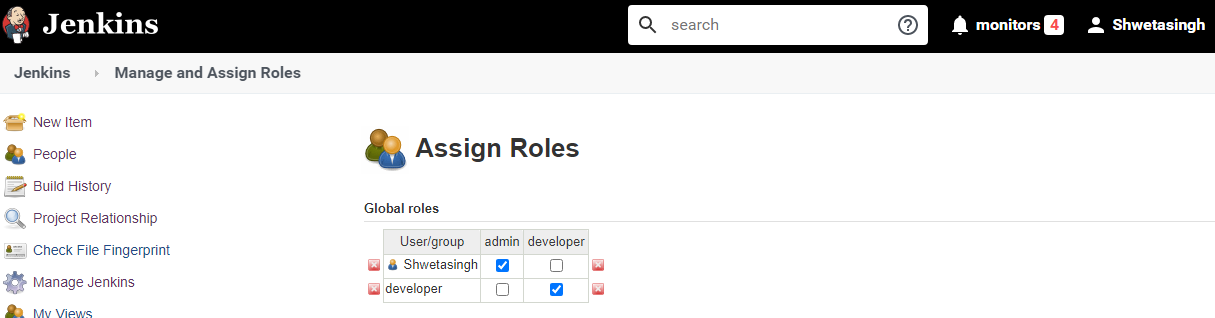
**Build is triggered every one minutes.**

****

**7. Authorization Techniques in Jenkins🡪**Defines roles based authorization strategy to manges the user authorization to Jenkins or for particular jobs by creating global role, project roles and agent roles. Assign these roles to user and groups.







All the above activities need to be performed using GIT and BroadLeaf application as the project to be continuous integrated, tested and deployed.