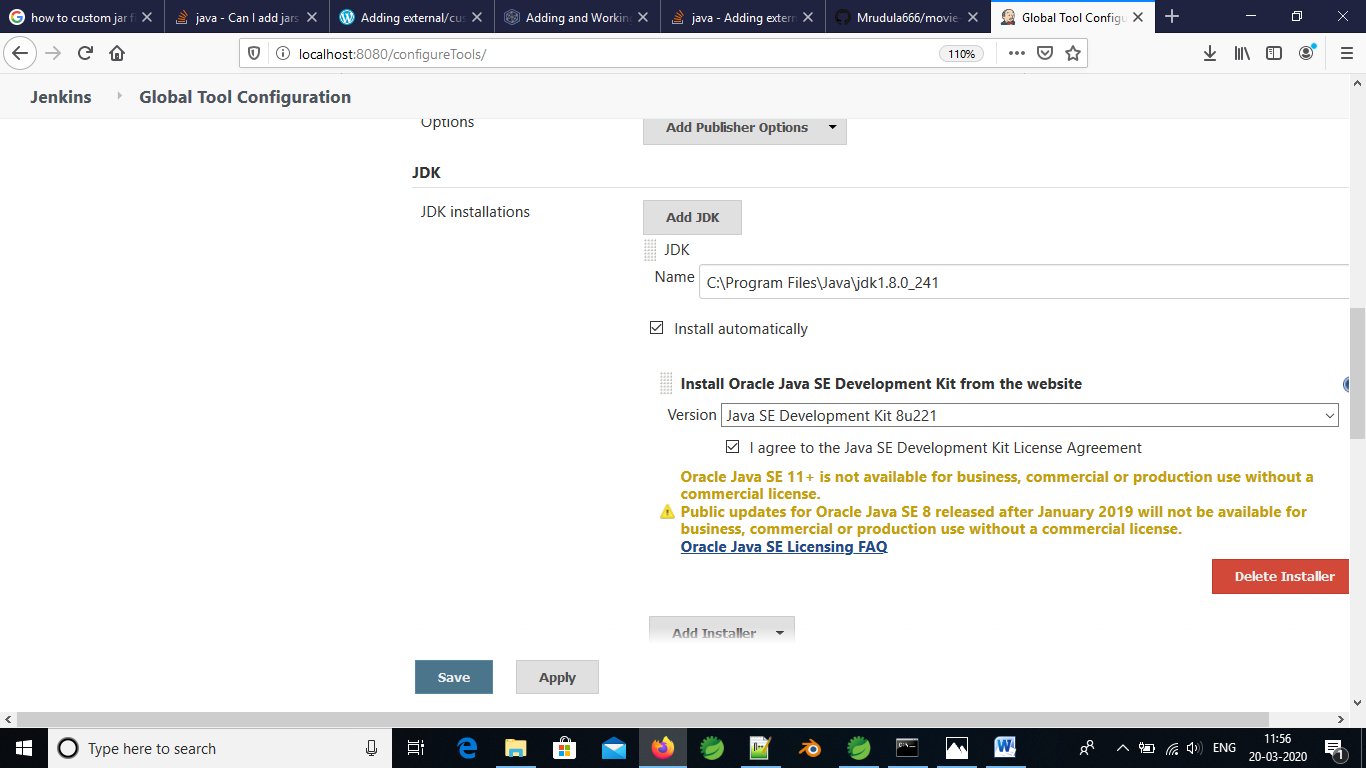
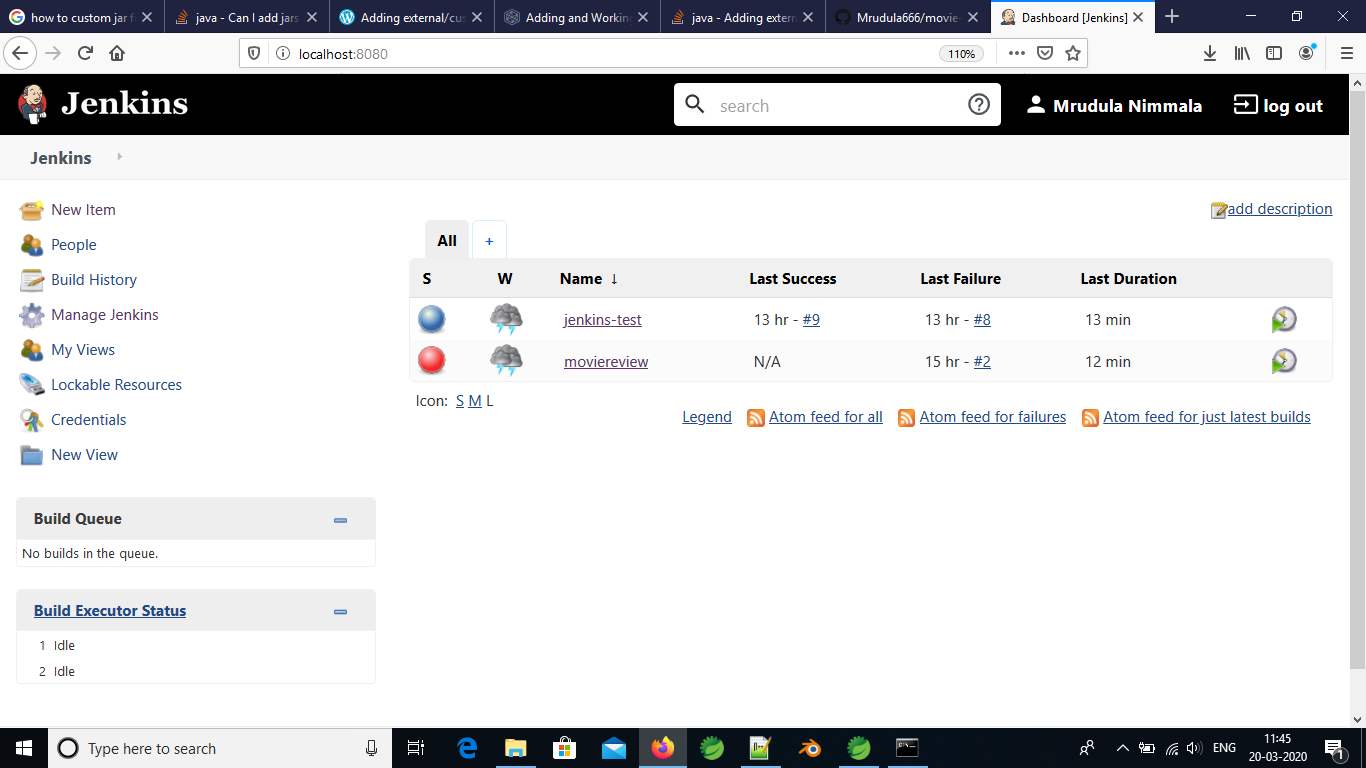
1. I have implemented the movie-catalog-service and uploaded in git.

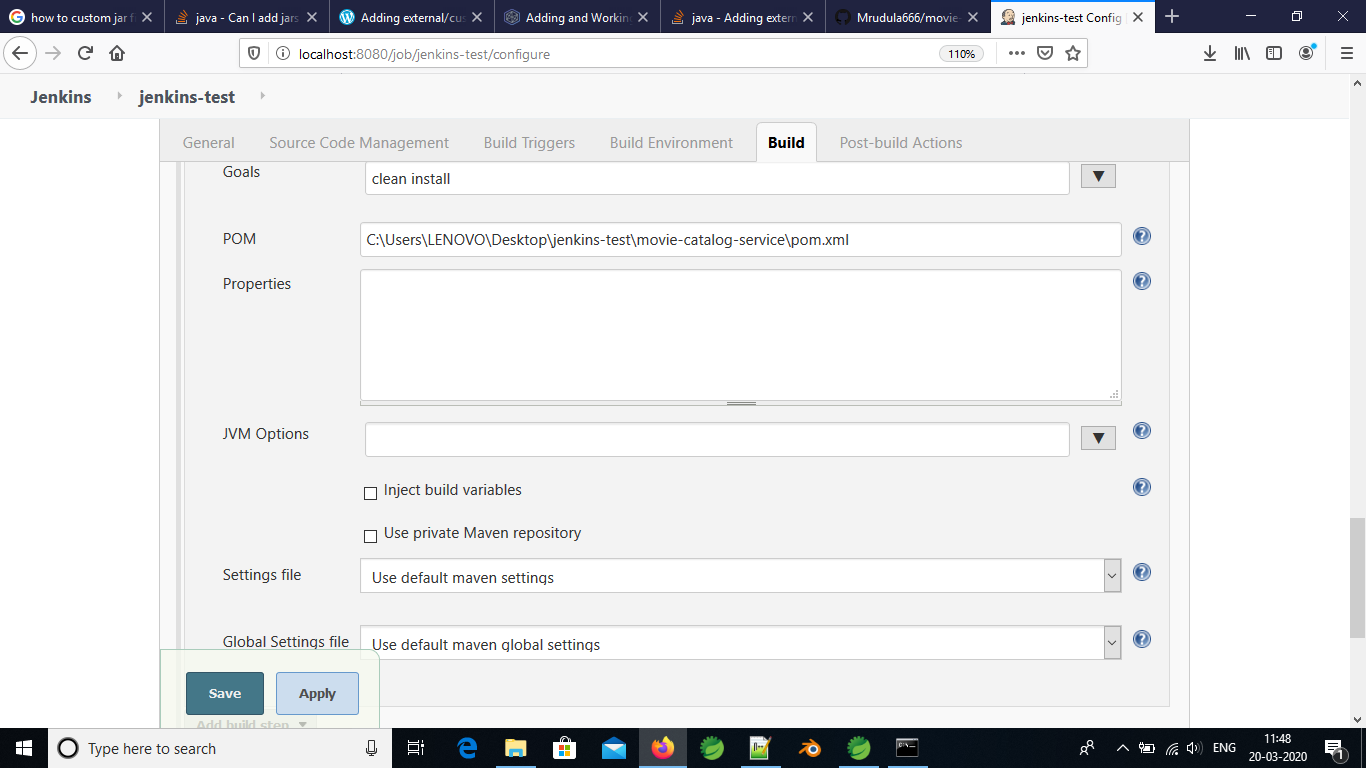
url for git:

<https://github.com/Mrudula666/movie-catalog-service.git>

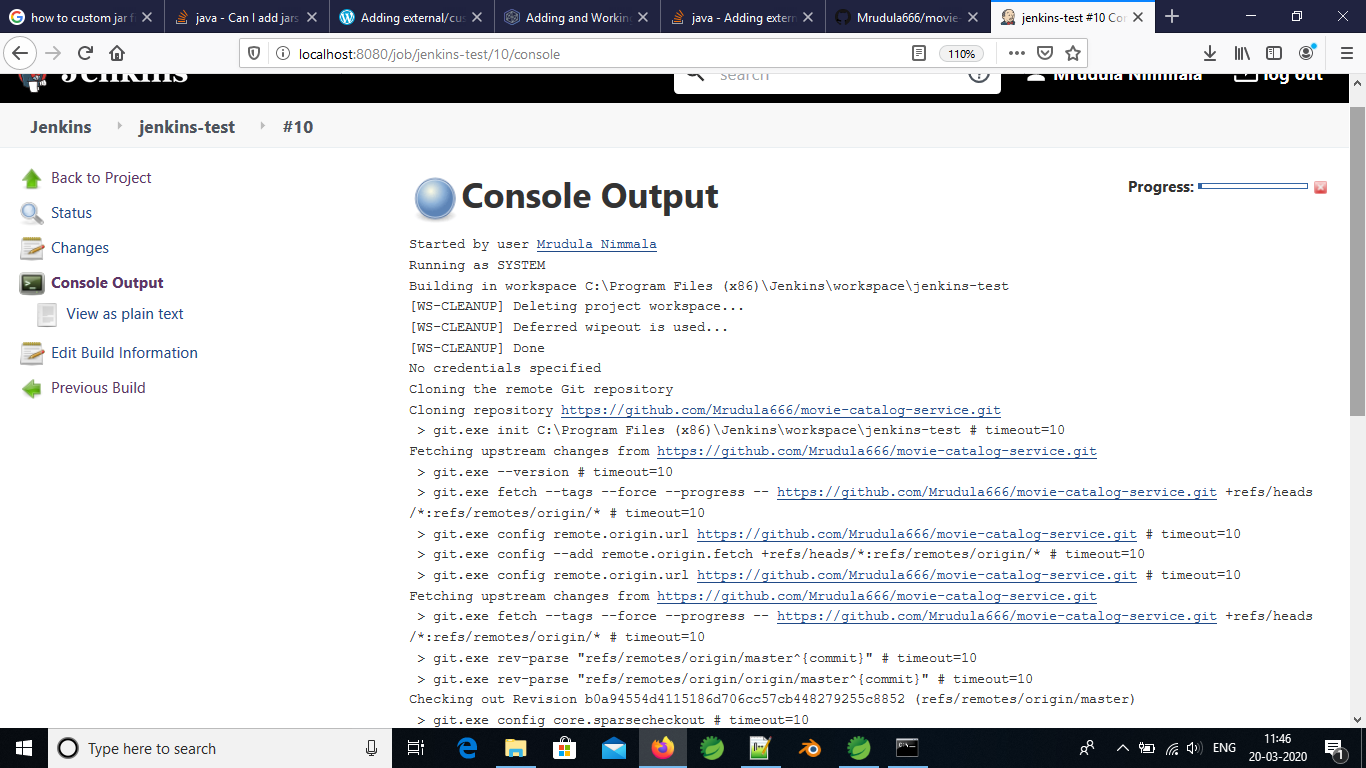
1. Configured global settings in Jenkins for jdk installations.
   1. 
2. Uploaded the maven project in Jenkins.



1. Configured the git url and the maven settings in Jenkins from manage Jenkins option



1. Run the maven command with following console output

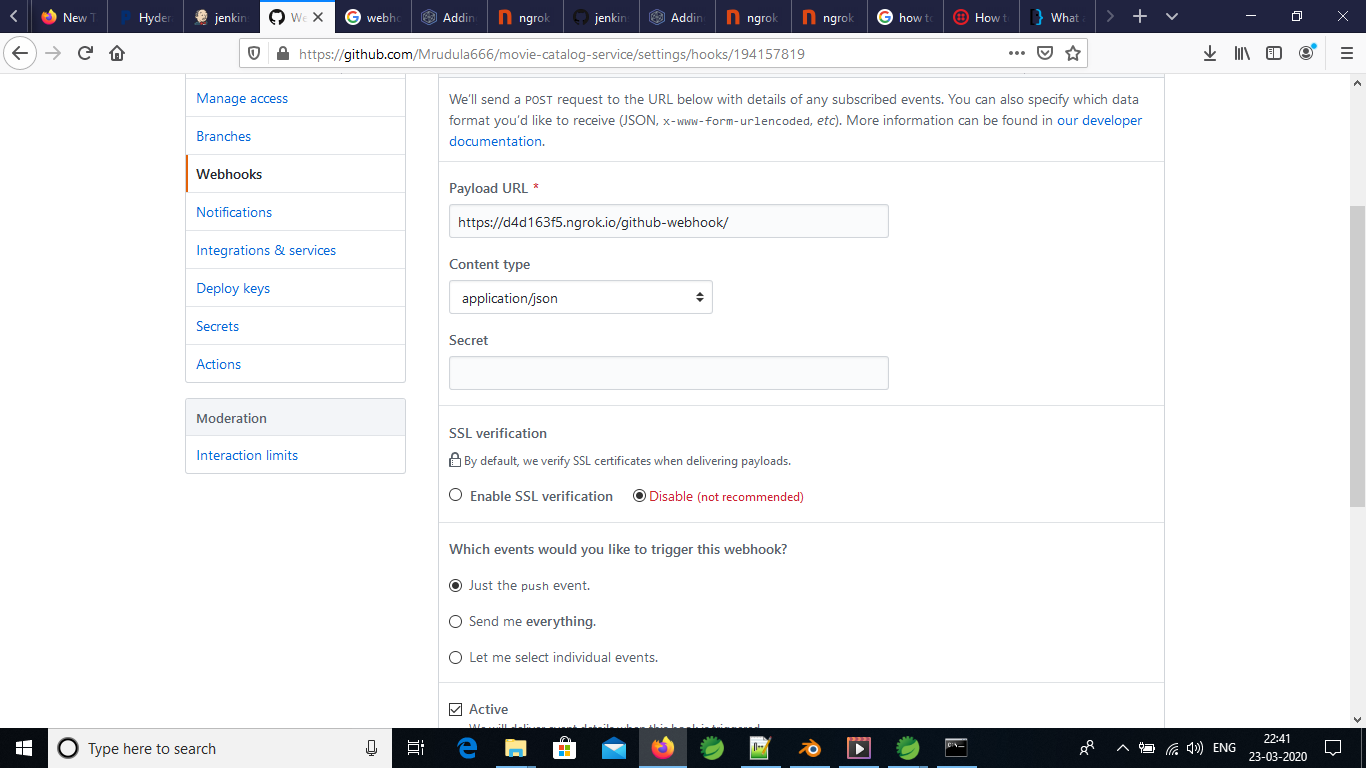




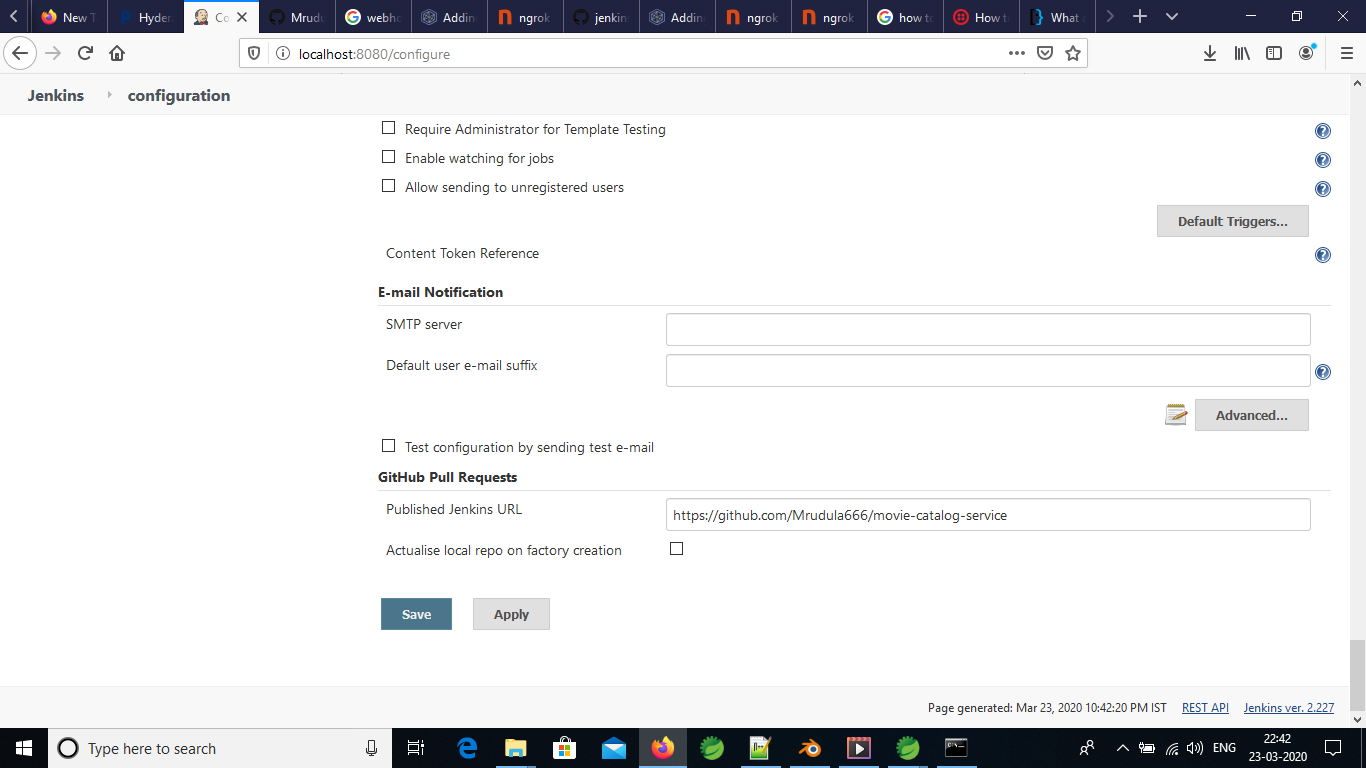
Date: 23-march-2020

**To configure the automatic build trigger for each commit.**

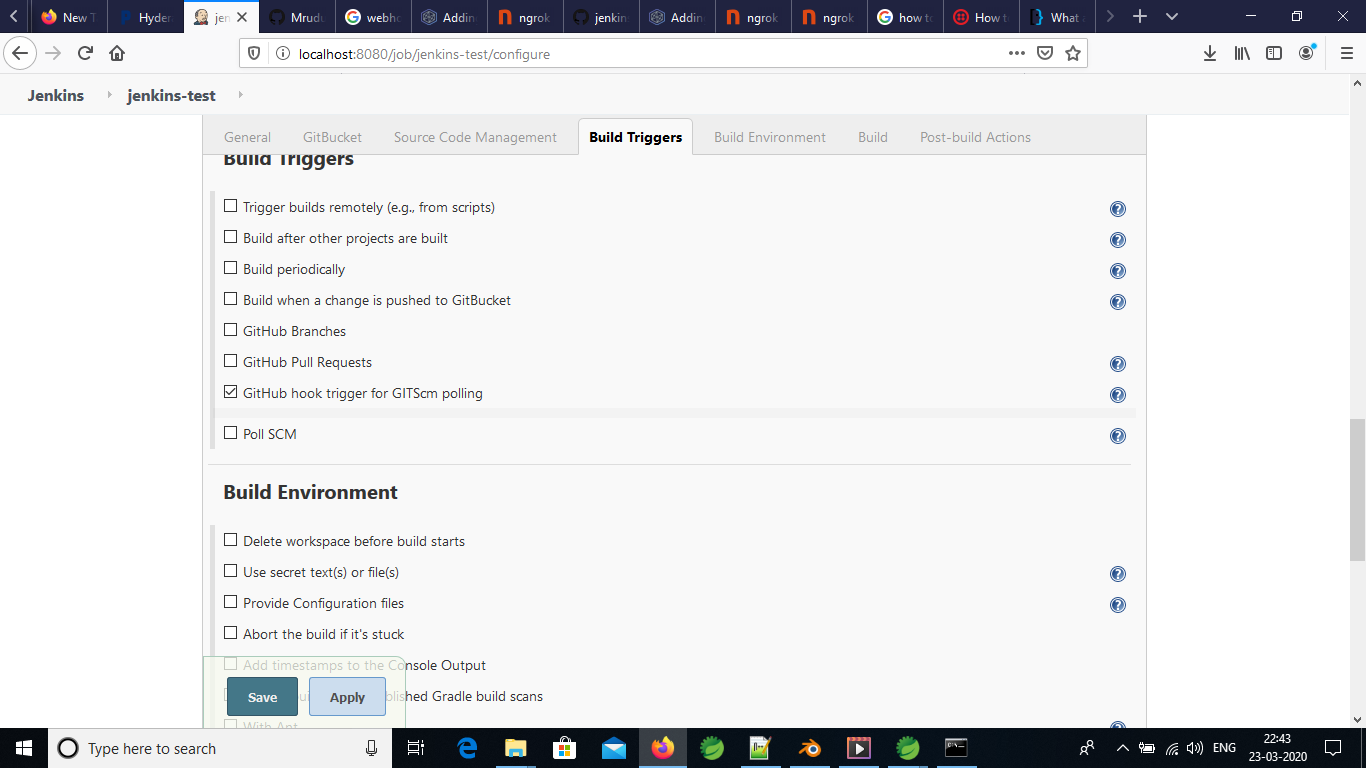
1. To configure the webhook we need to convert the localhost to public network.
2. To do this, we have to install ngrok in the local system and set the path in environment variables.
3. Added the webhook in github as show below.



1. Configure the git pull requests in Jenkins with git repository



1. Modify the manage configure section in job to mark “GitHub hook trigger for GITScm polling” as below.



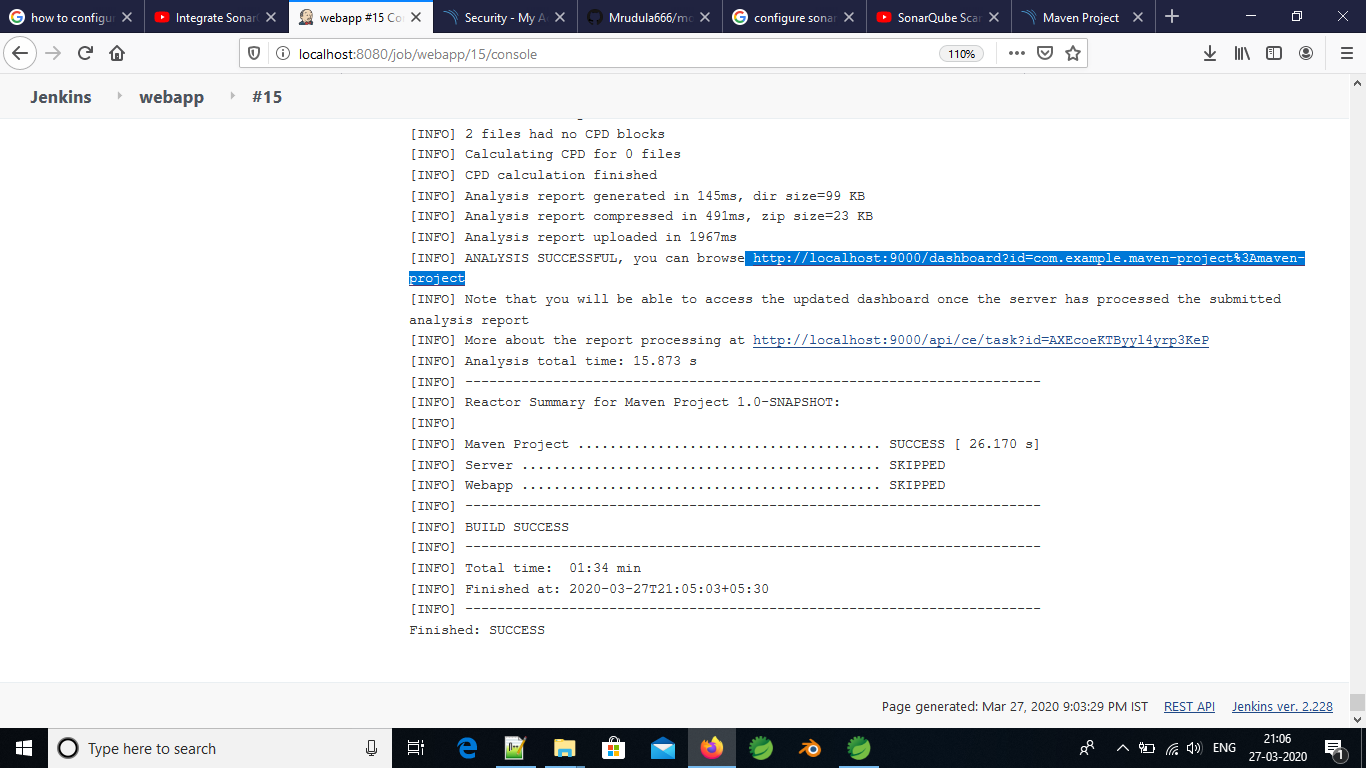
1. Refresh the Jenkins page.

Date: 27-march-2020

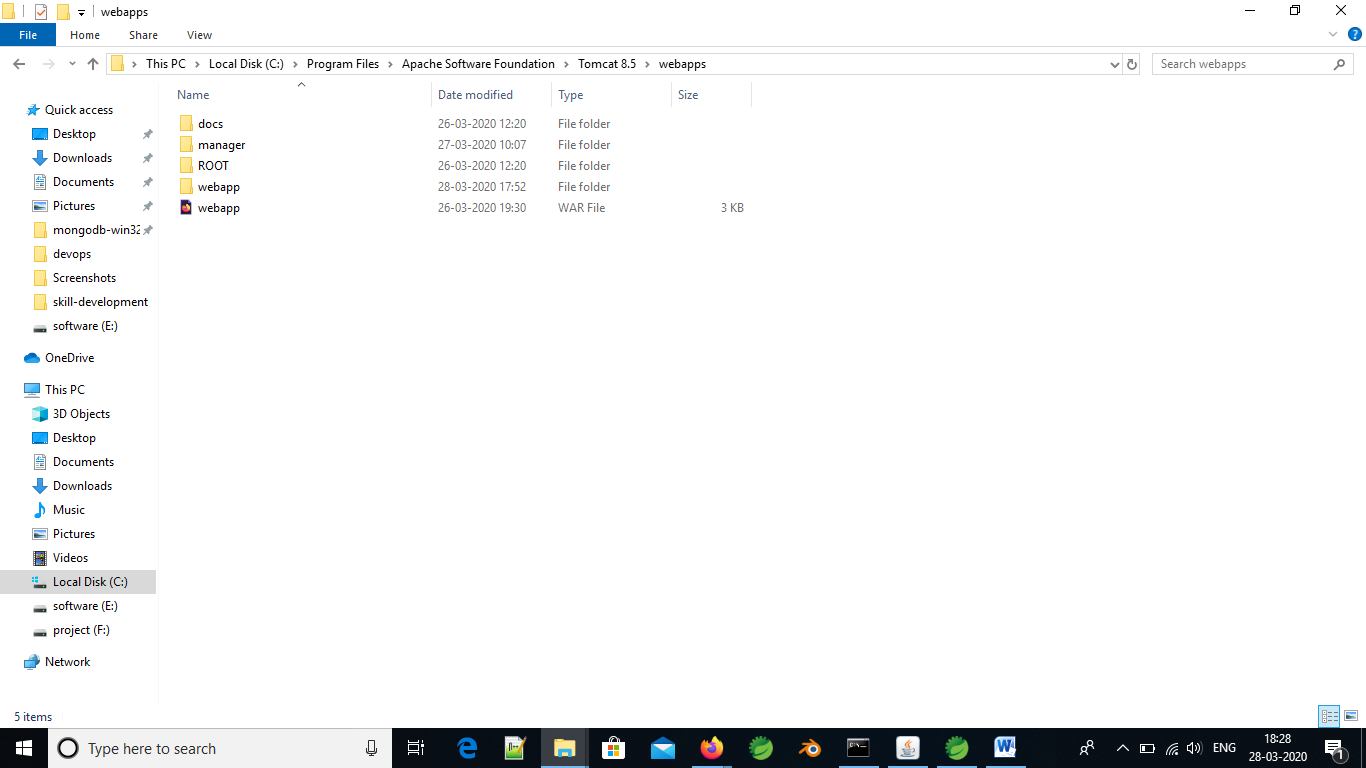
Task: Run the Jenkins created war in tomcat server locally

1. At Jenkins, set the goal in job as clean package where the war file is created at

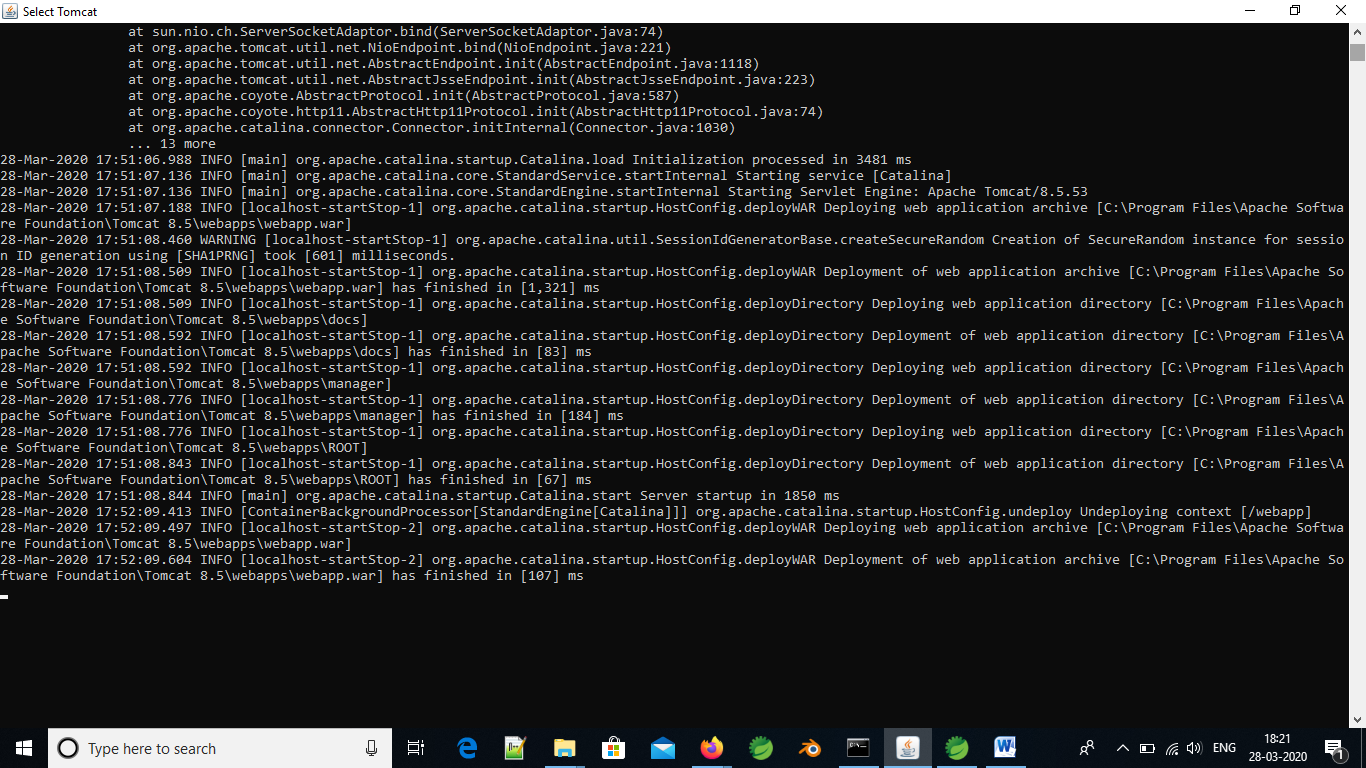
C:\Program Files (x86)\Jenkins\workspace\webapp\webapp\target\webapp.war



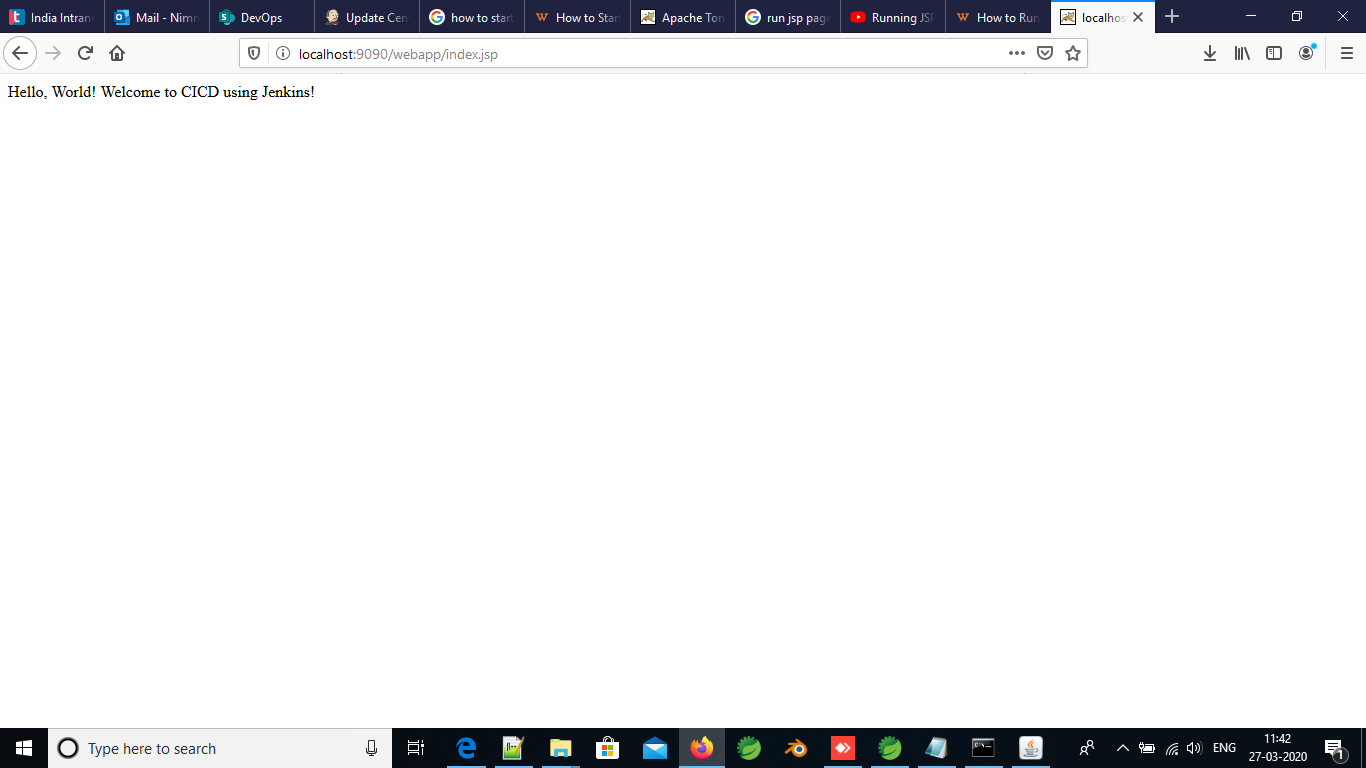
1. Copied the webapp.war file at C:\Program Files\Apache Software Foundation\Tomcat 8.5\webapps location.



1. Started the tomcat server

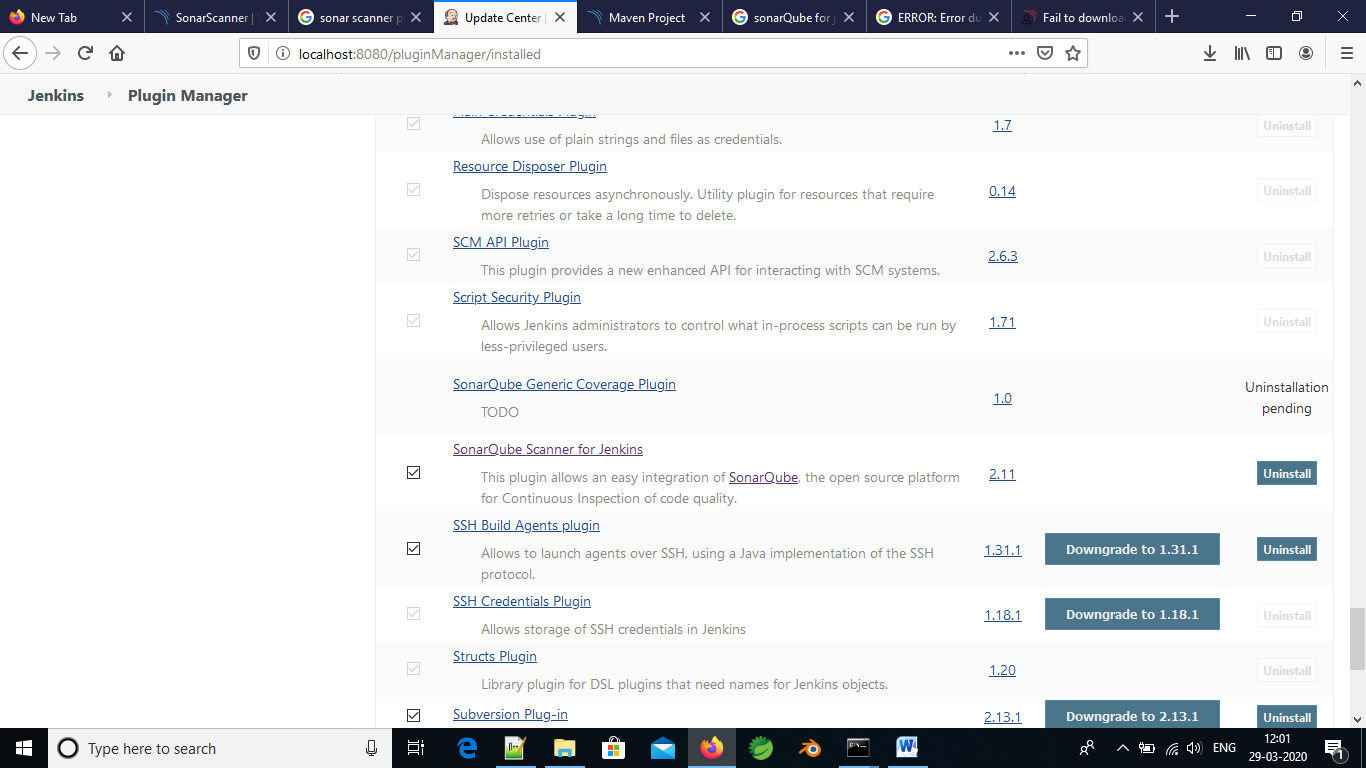


1. Ran the server at 9090 with the url <http://localhost:9090/webapp>

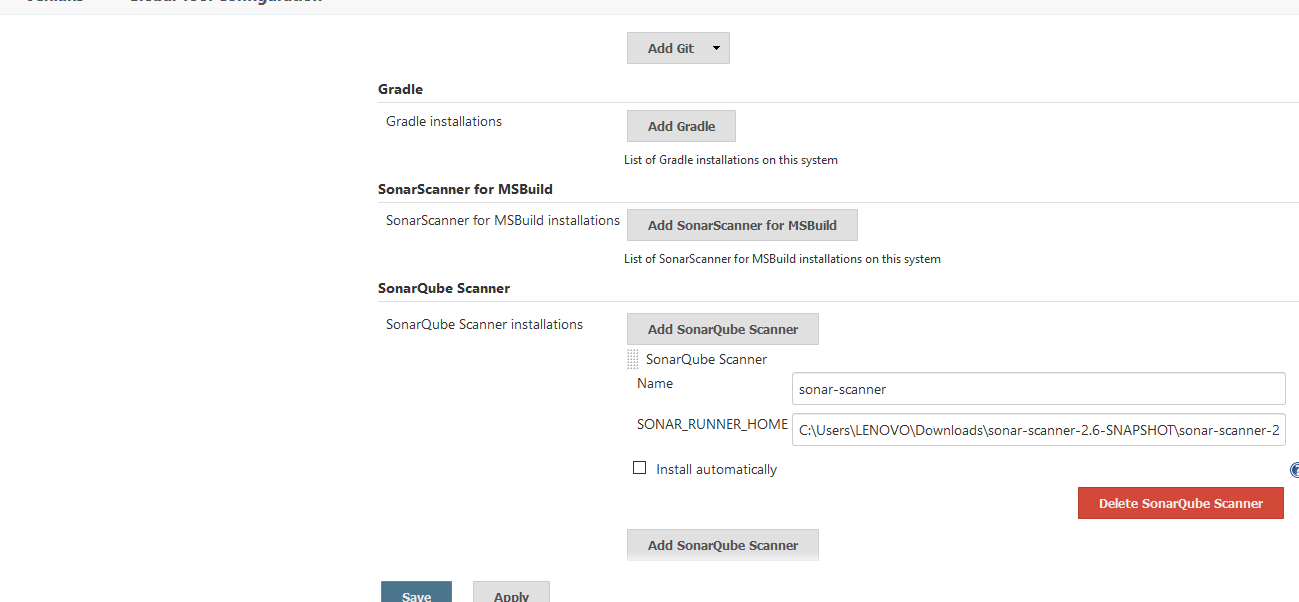


Task: Run the sonarQube in Jenkins

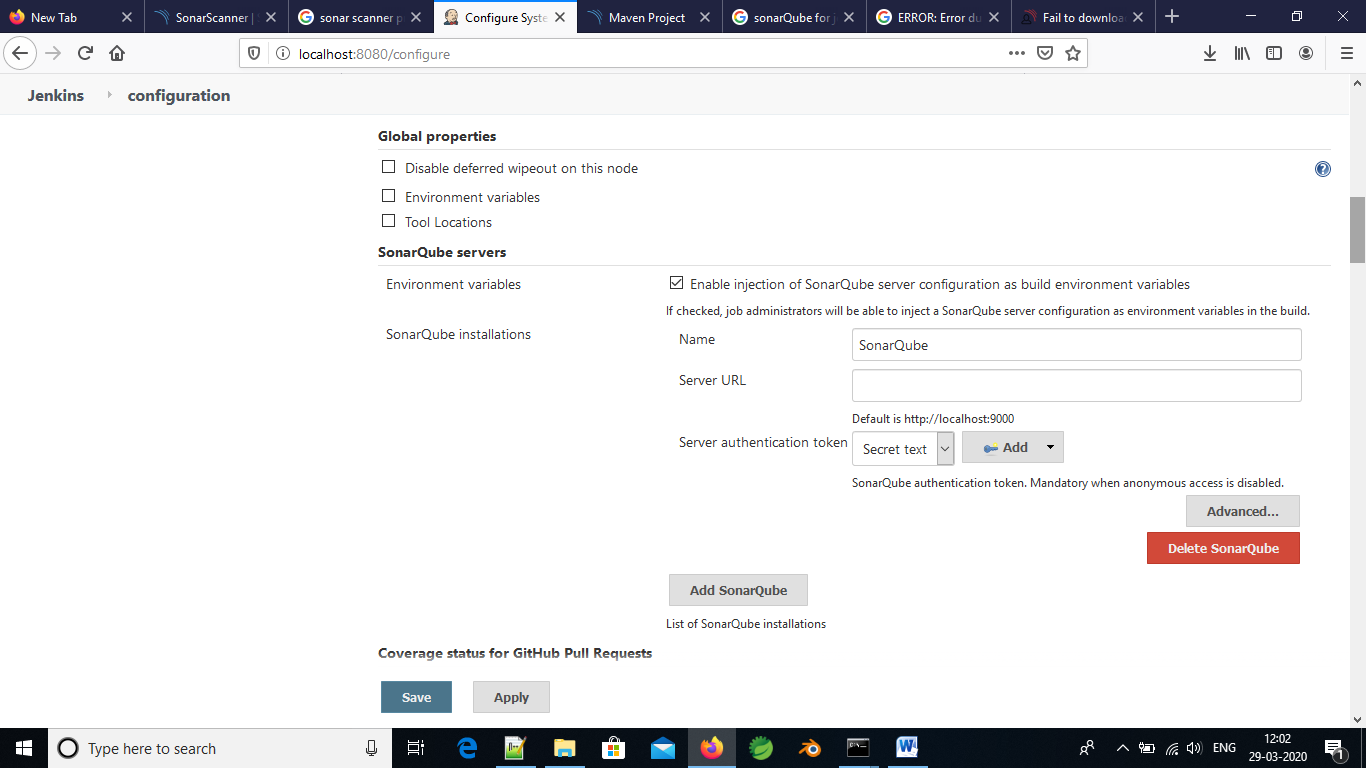
1. Install the sonarQube Scanner in Jenkins at manage Jenkins



1. Configure the Jenkins in global tool configuration



1. Configure the sonarQube server installation in configure option



1. Build the job.
2. See the results in <http://localhost:9000>

