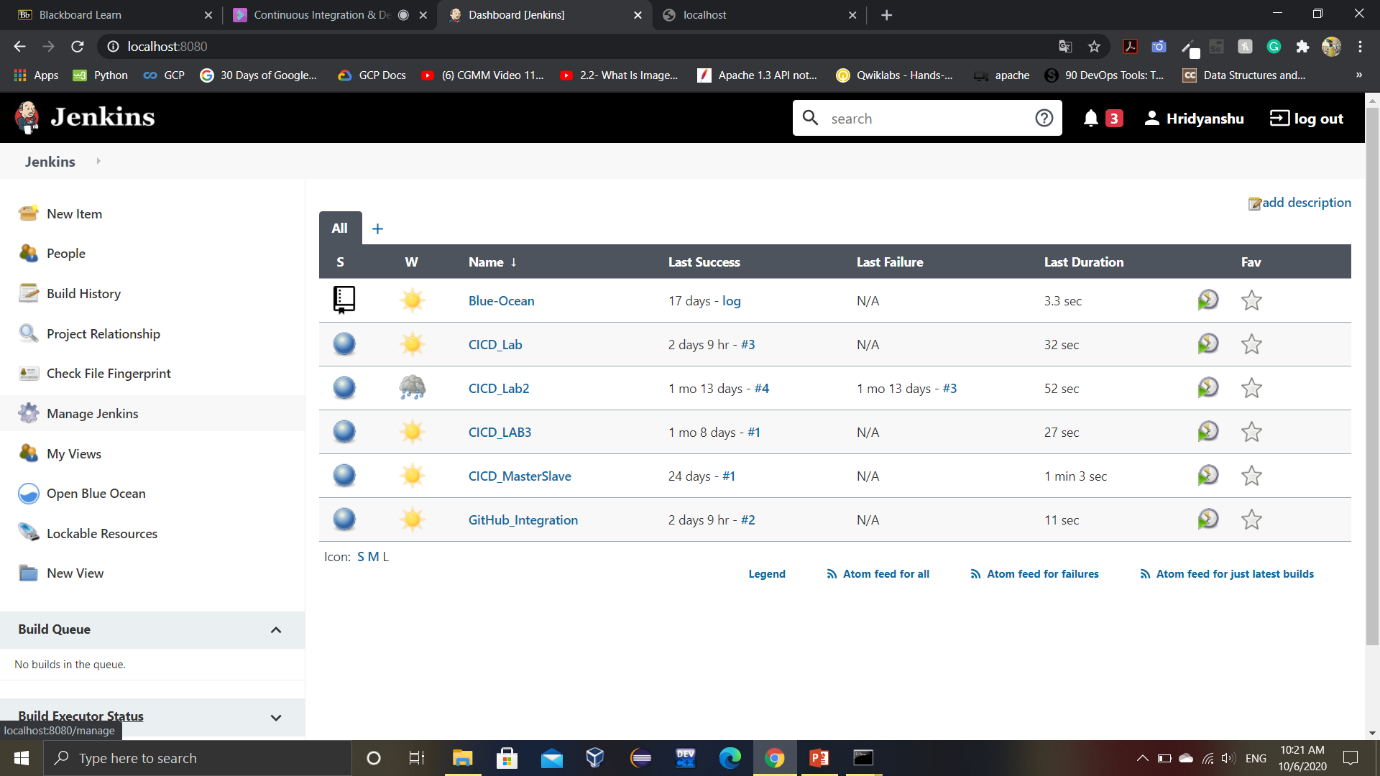
**Nexus Integration with Jenkins.**

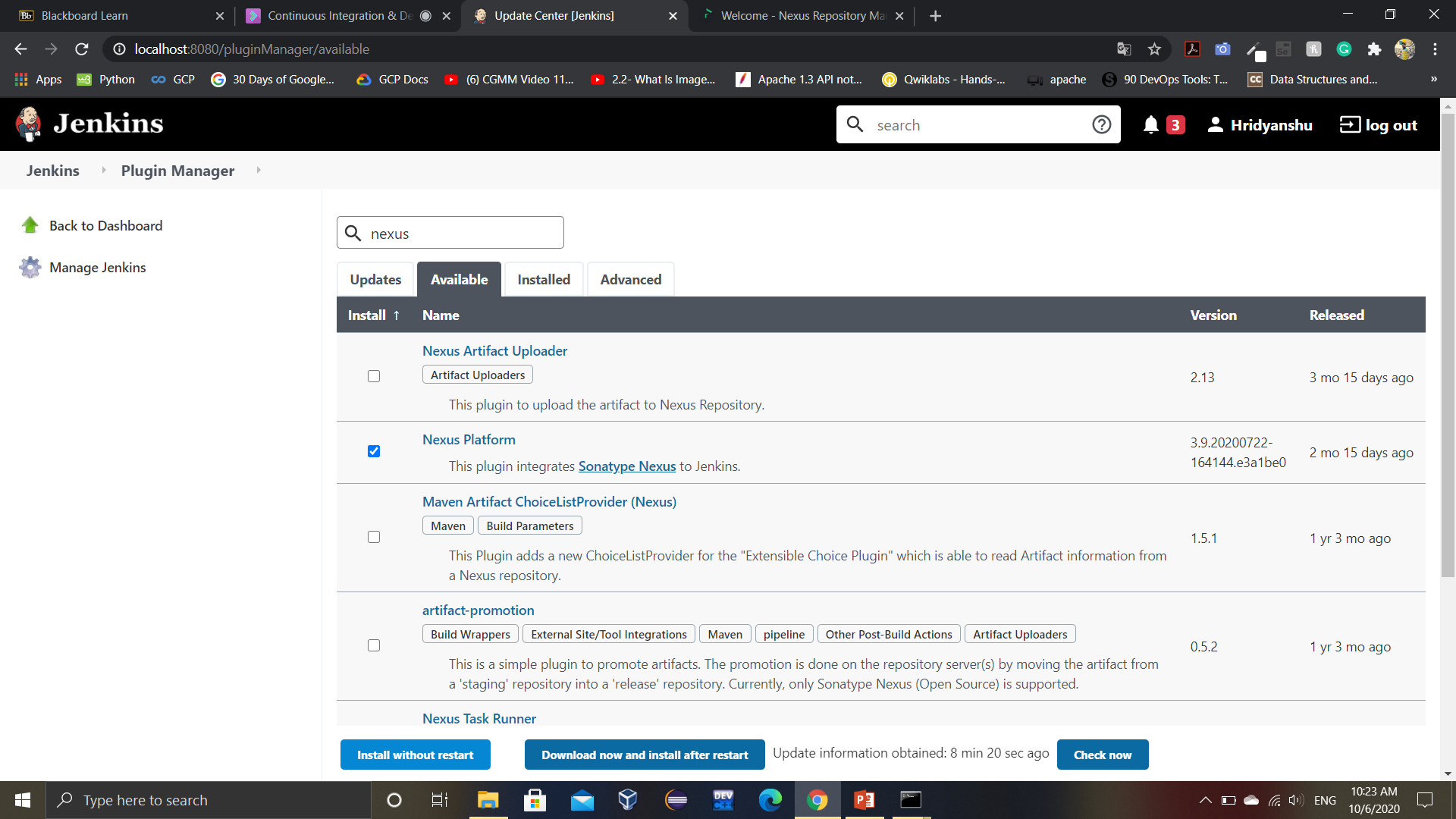
In this experiment we will go through a series of steps that will help us to deploy a maven project on the Nexus Server by integrating Jenkins and Nexus,

The steps that need to be followed are:

1. Go to Manage Jenkins option that is displayed on the Jenkins Dashboard.



Now go to the Manage Plugins option and install the Nexus Plugin.



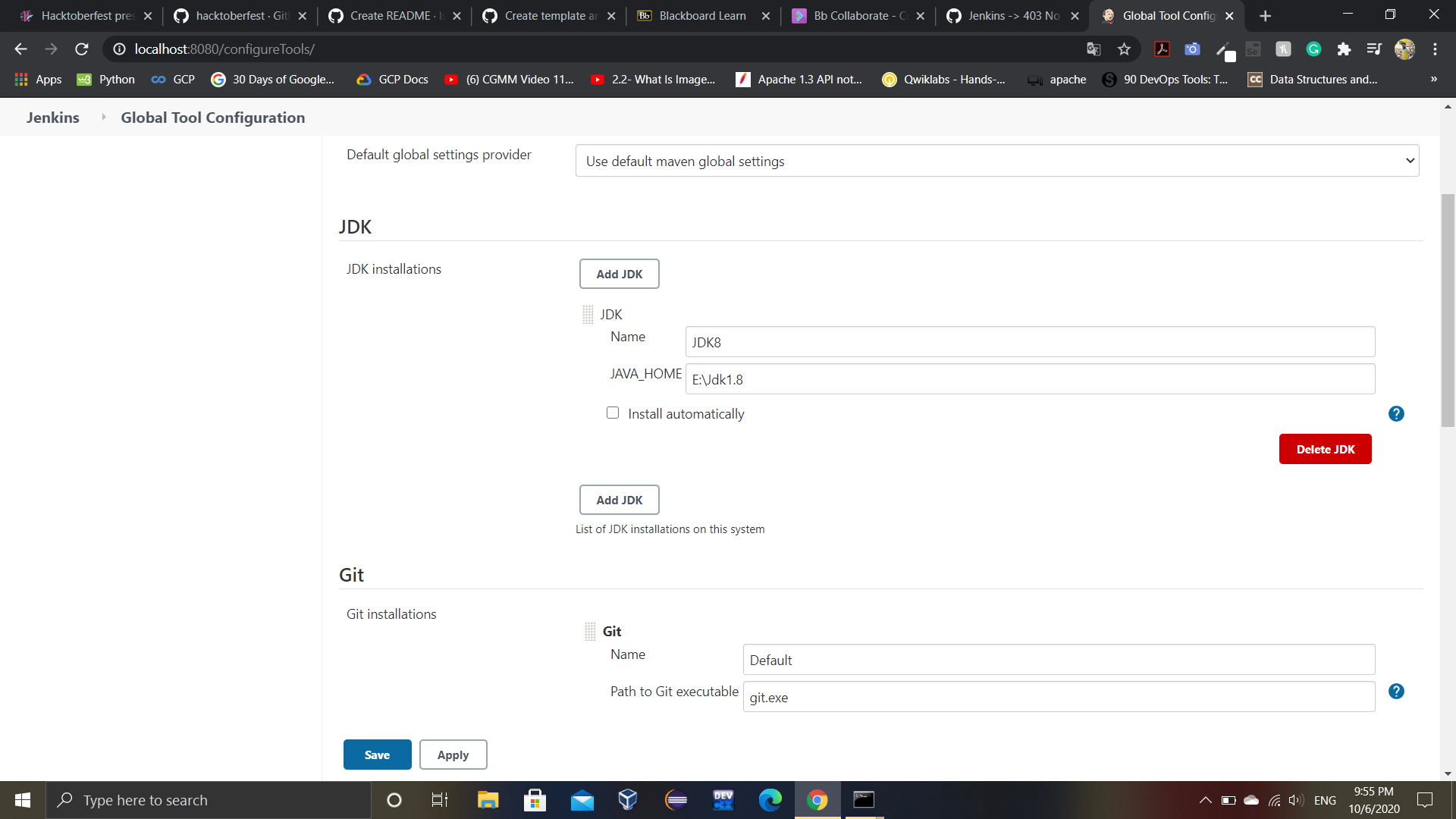
2. After successful installation of the plugin, we need to provide the path for JDK 8 in our Jenkins. This step is necessary because the version of the Nexus Server being used in this experiment (3.18) does not support JDKs later than JDK 8.

a. Download and install JDK 8 in your system

b. Go to the Global Tool Configuration option in Manage Jenkins console.



c. Go to the JDK installations and add a JDK by providing it a name and the path at which JDK 8 is installed.



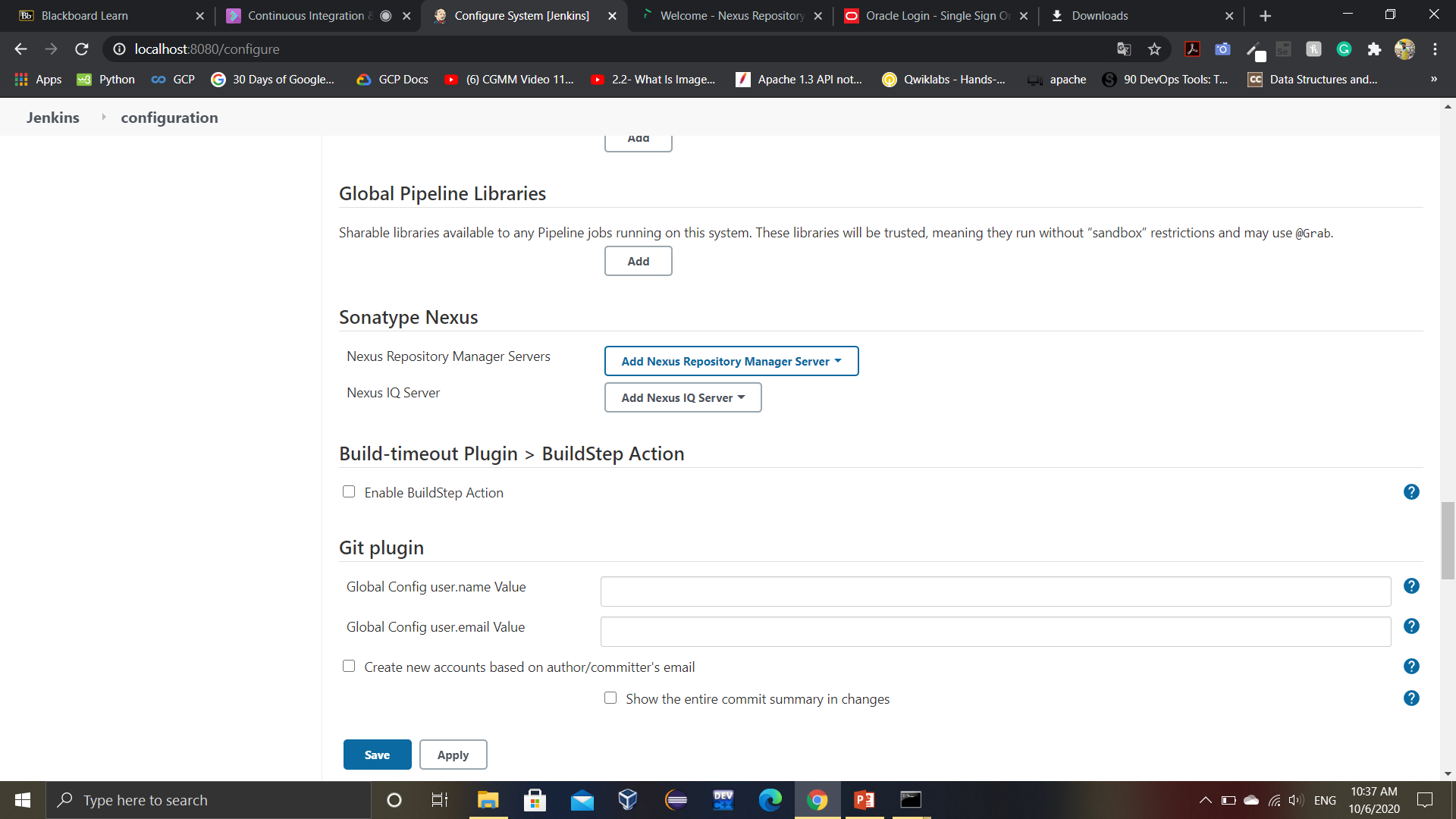
Make sure that install automatically checkbox is unchecked.

3. Now we need to add Nexus Server in our Jenkins.

a. Go to the Configure System option in Manage Jenkins Console.

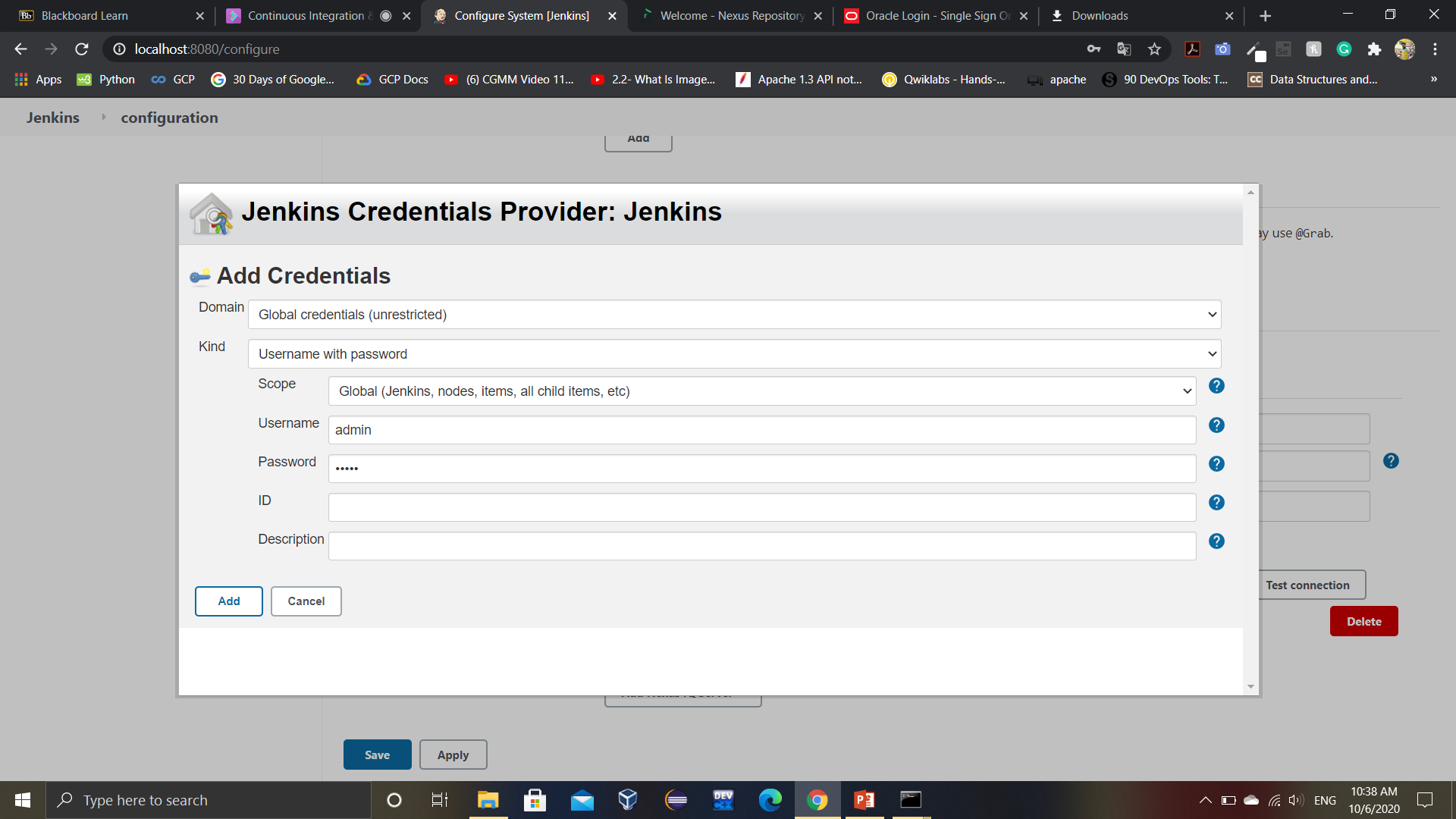


b. Click on Add Nexus Repository Manager Server option and select Server3.x

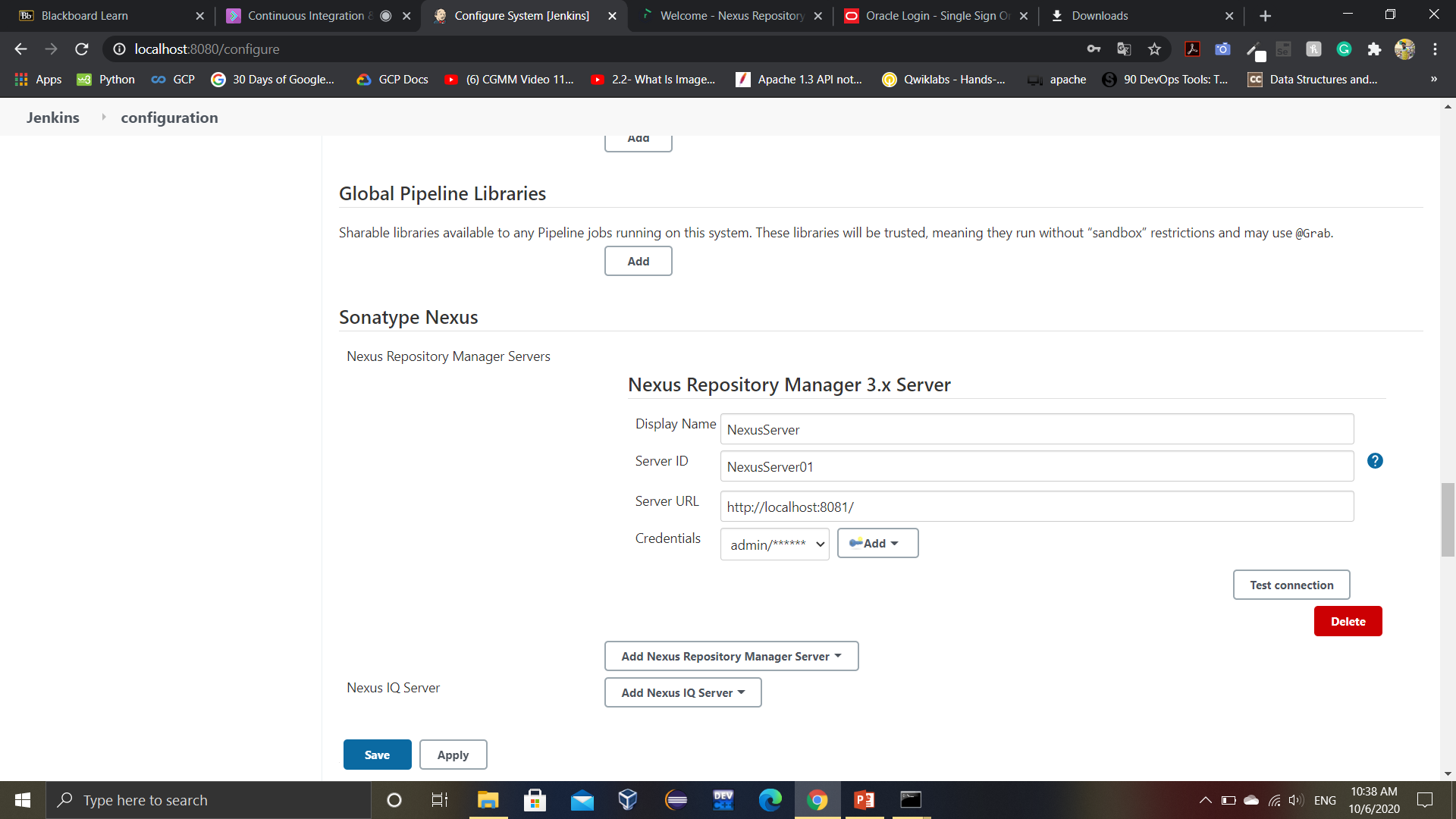


c. Provide the name of the server and its URL (which is <https://localhost:8081/> by default). Next we will need to add our credentials of Nexus Server in Jenkins

Click on add option displayed in front of credentials. You will get a similar window. Provide your Nexus Server credentials here.



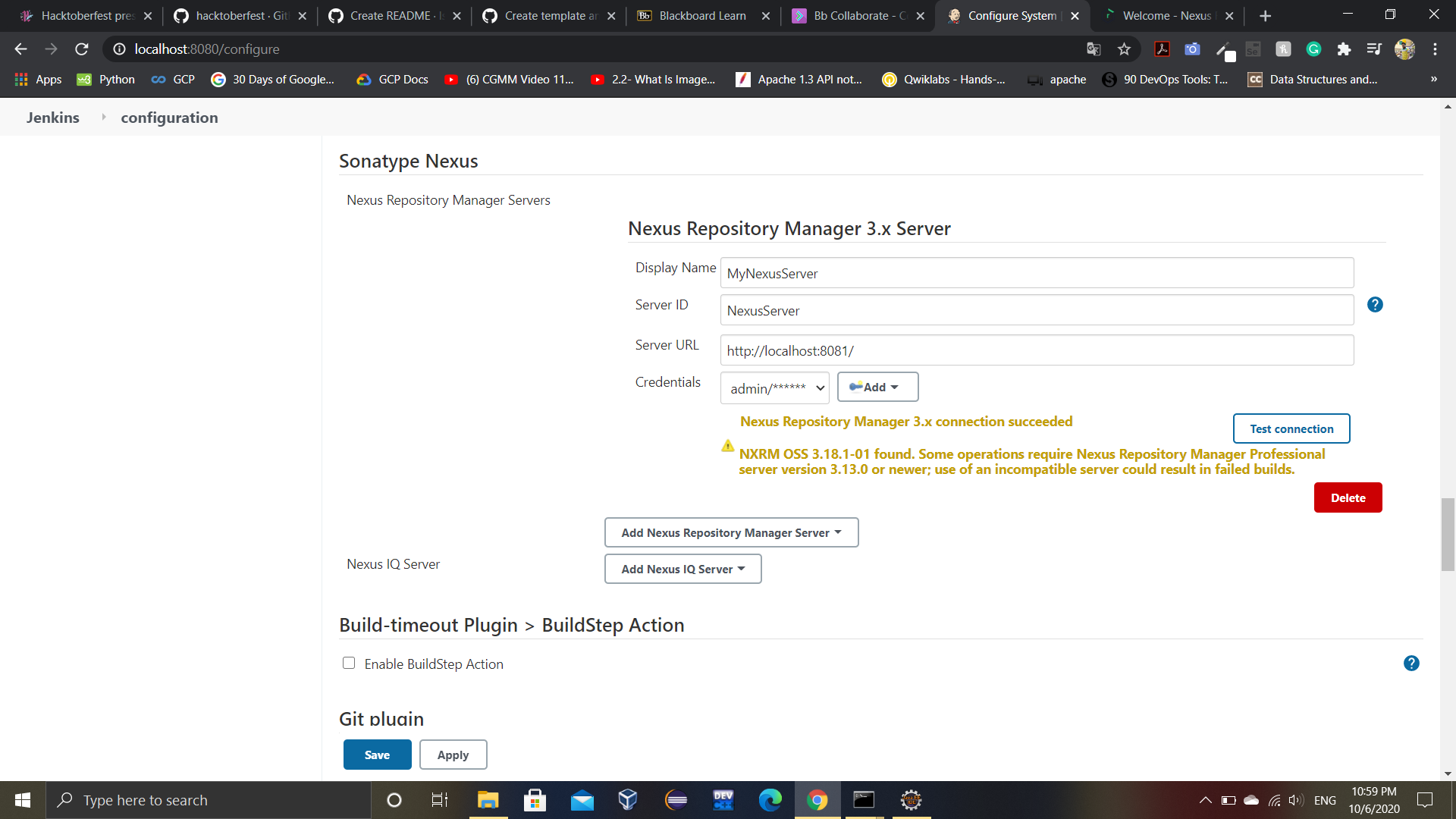
The final configuration will look like:



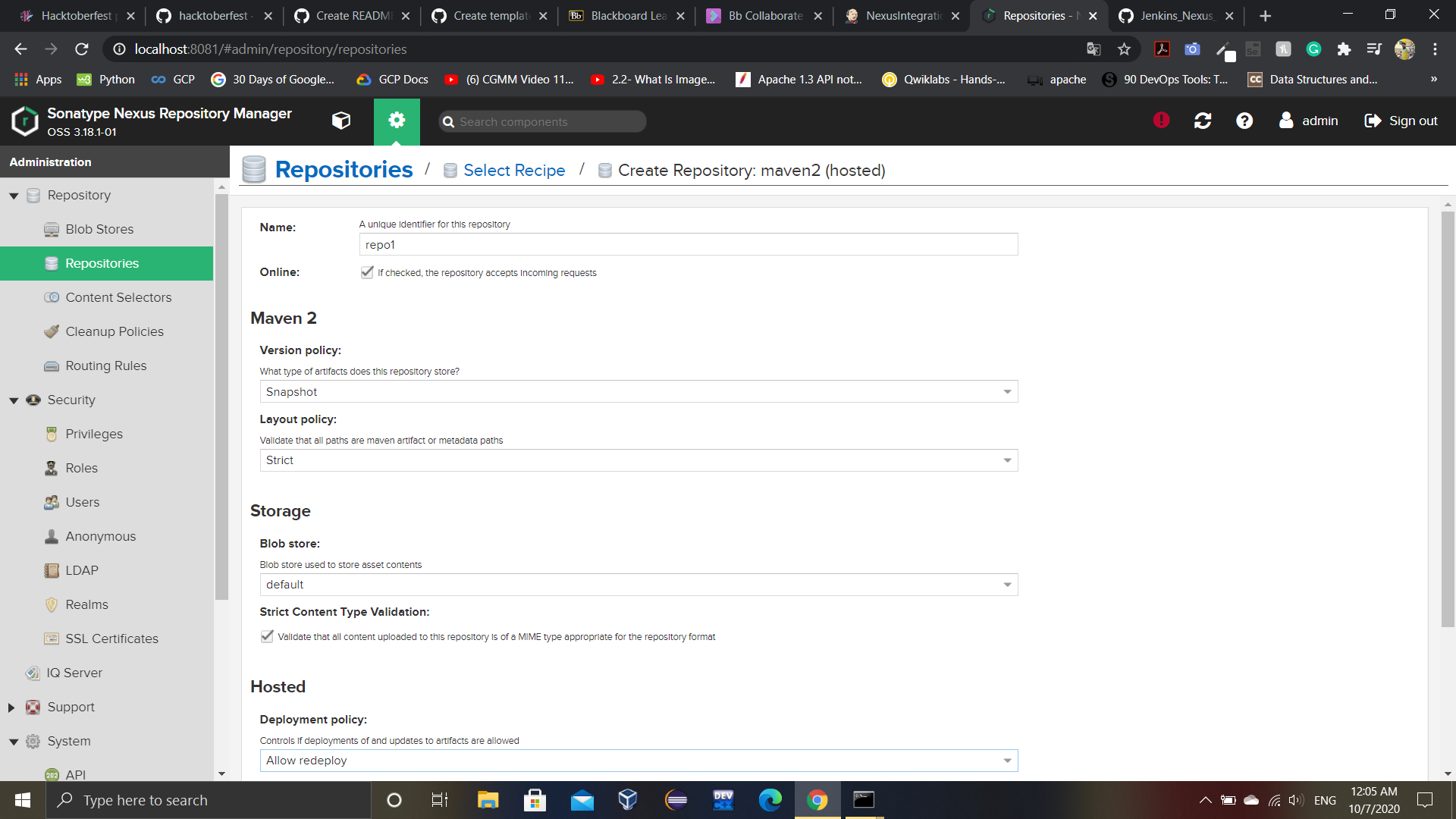
4. Click on test connection button to verify that the Jenkins is successfully connected to Nexus.

Note: It is recommended that before you click on test connection, your Nexus server must be in running state.

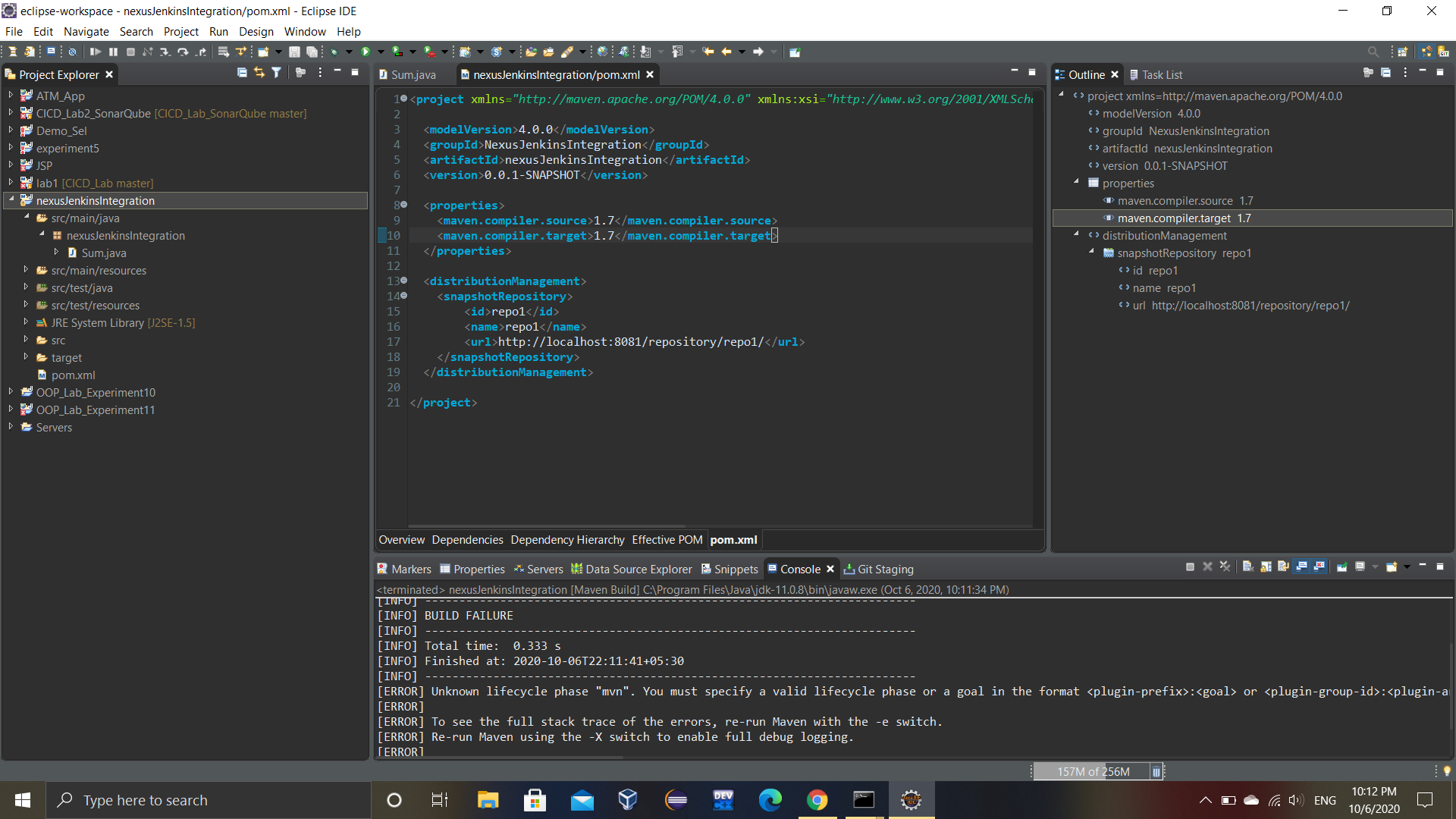
You will get the following output on successful connection:



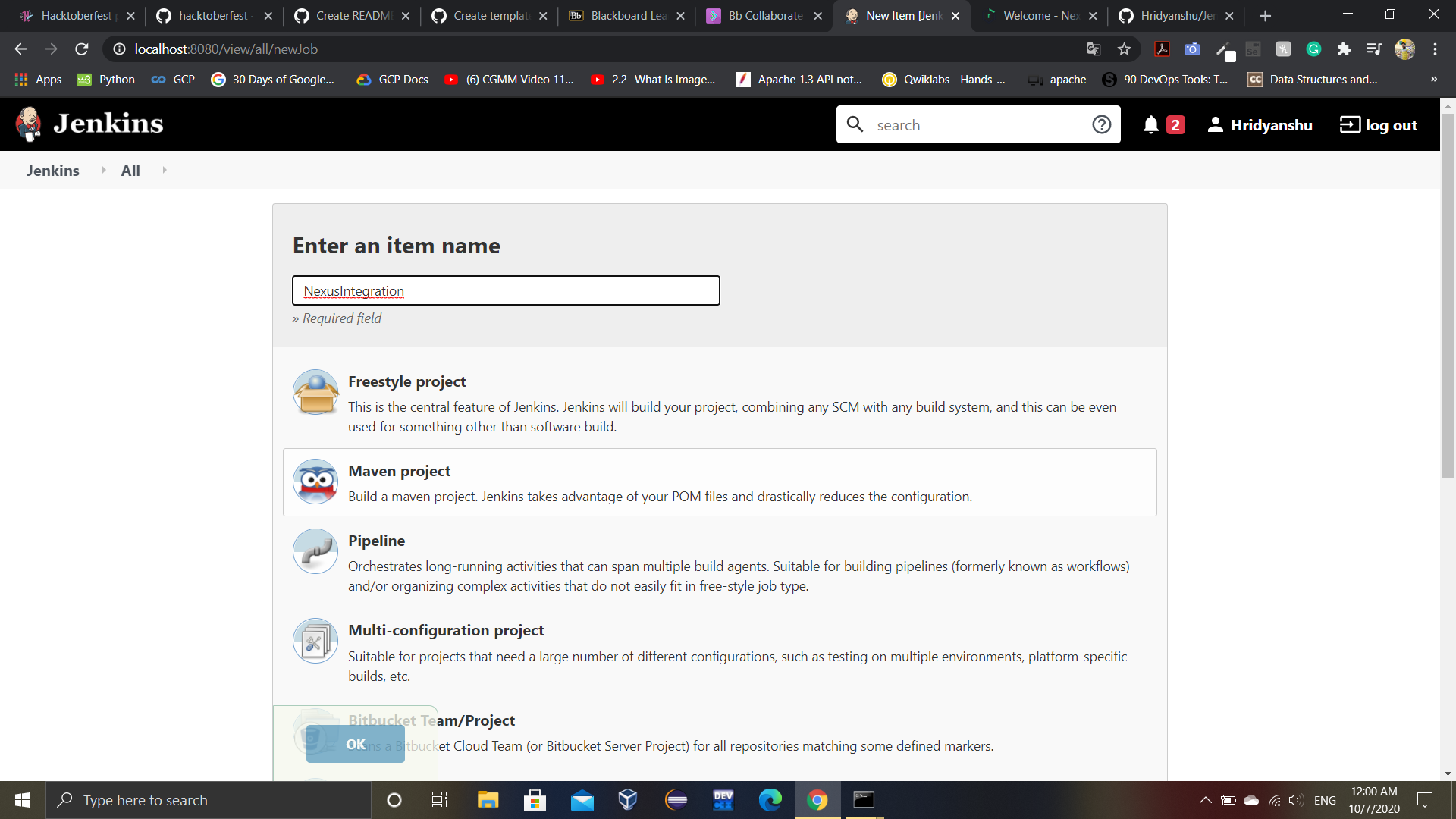
5. Create a new repo on Nexus



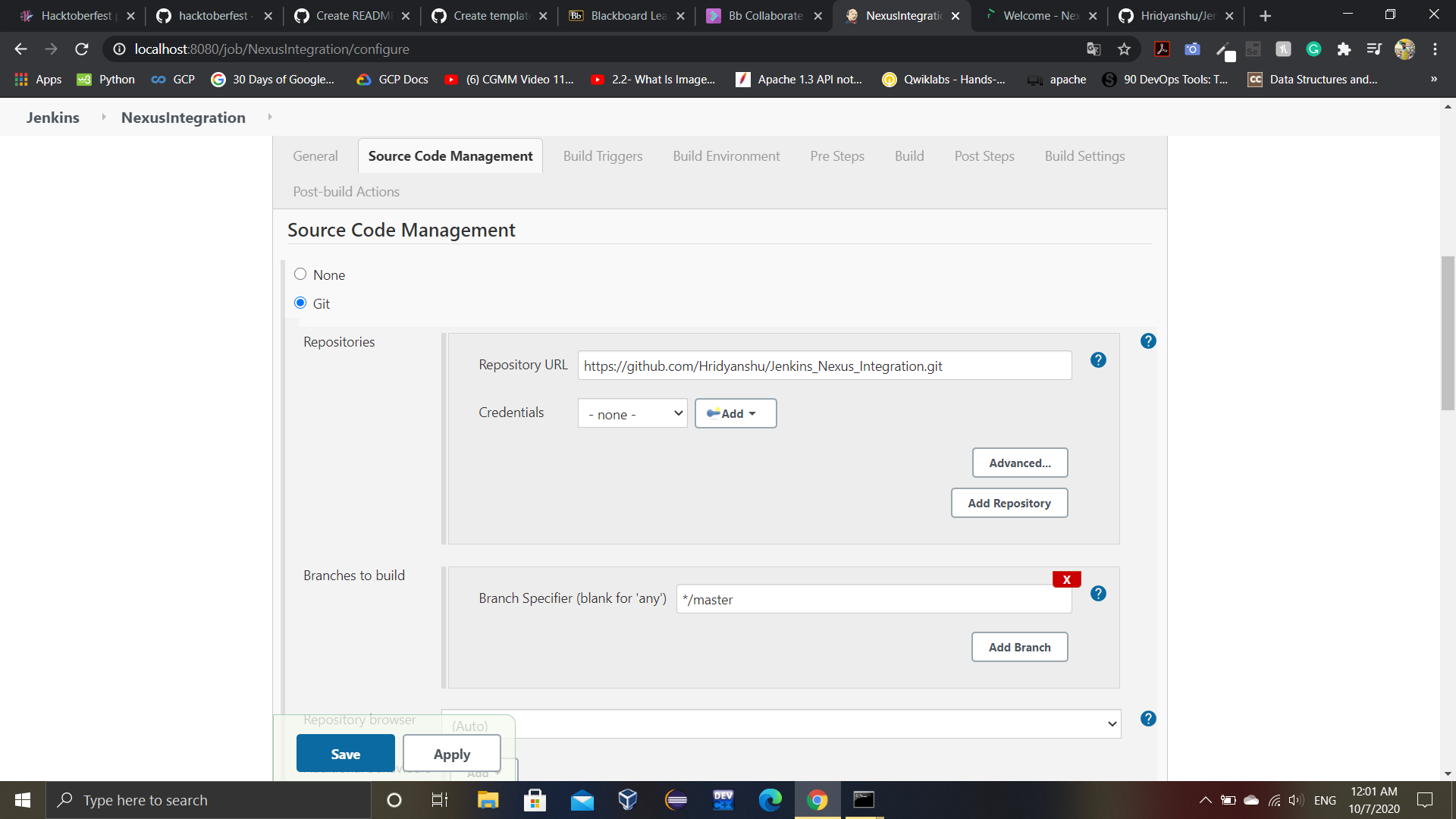
6. Add the following code to the pom.xml of the maven project that is to be deployed to the Nexus server.

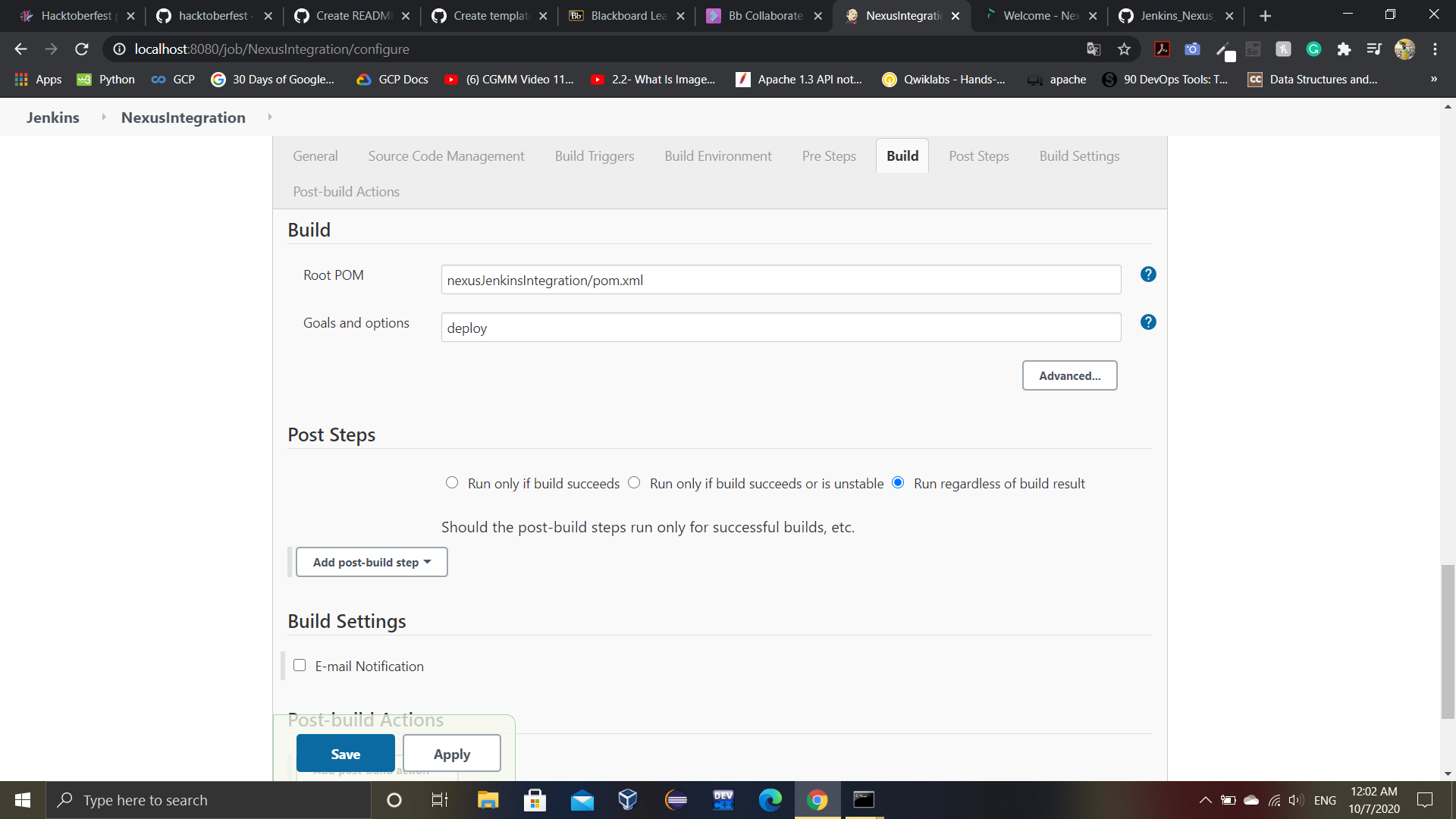


7. Create a new maven project on Jenkins and link it to the GitHub repository of the newly updated maven project. (Note: Make sure that the github repo has updated pom.xml).



The configuration of the project will be as follows:





8. Now build the project. The final output will look like:

