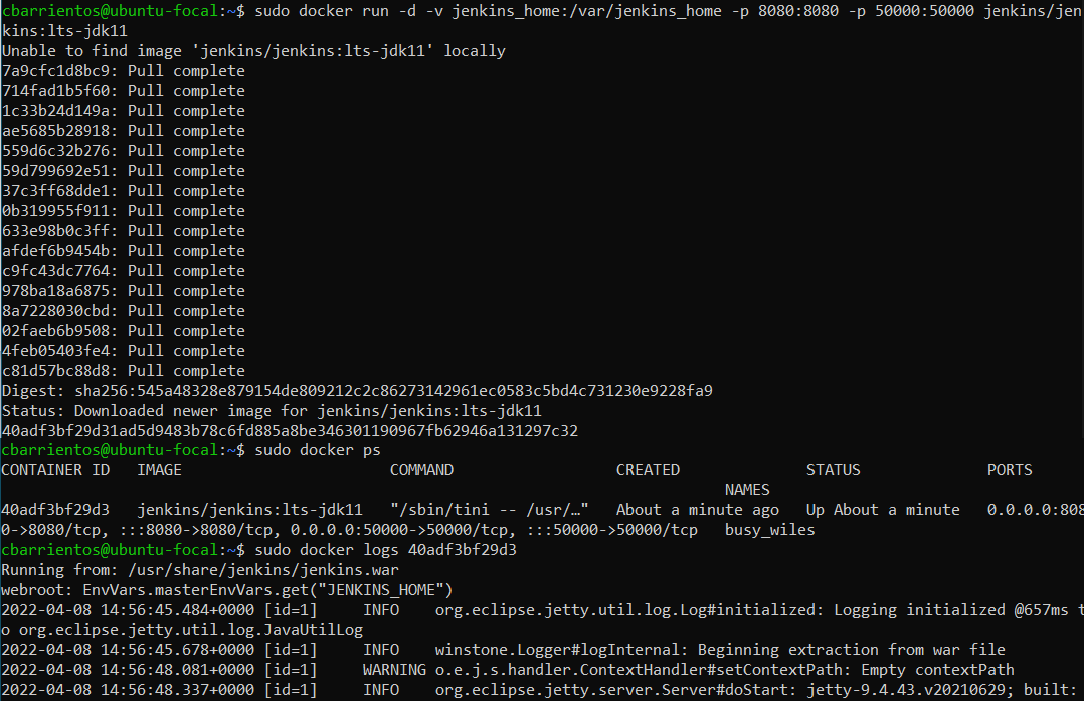
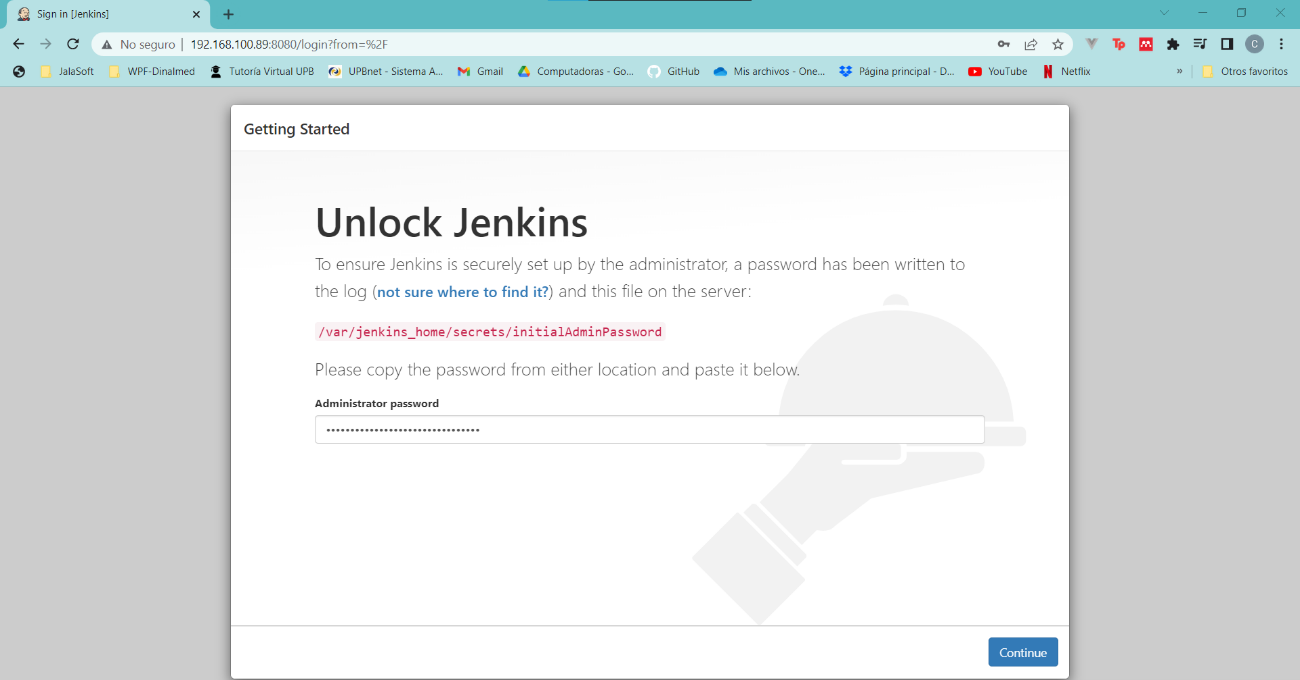
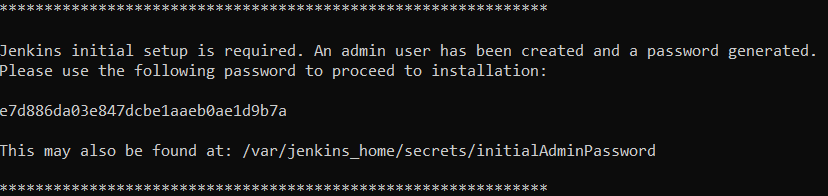
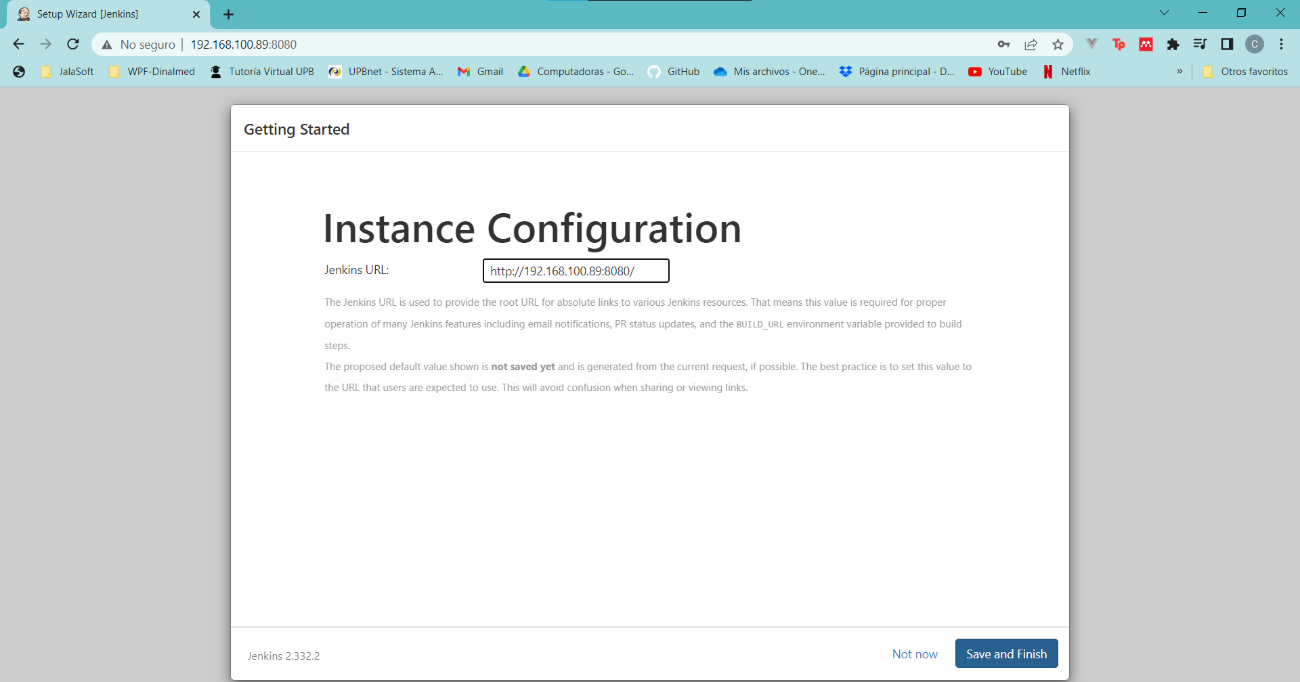
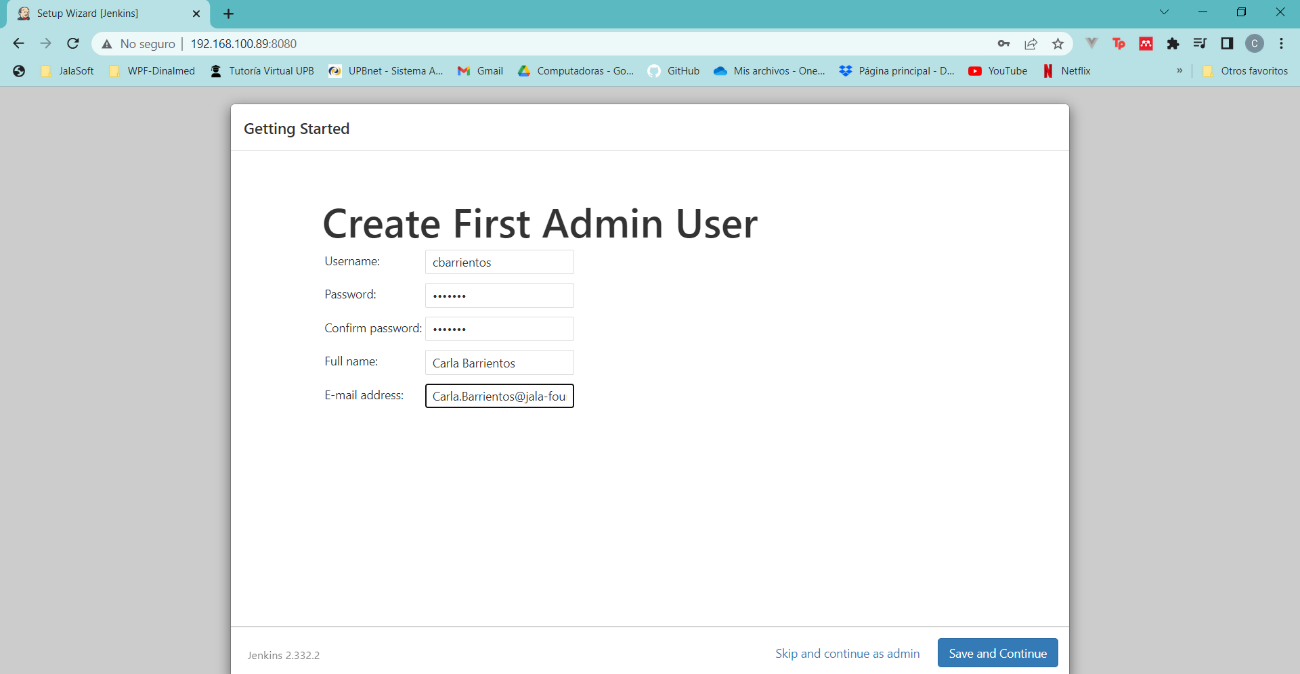
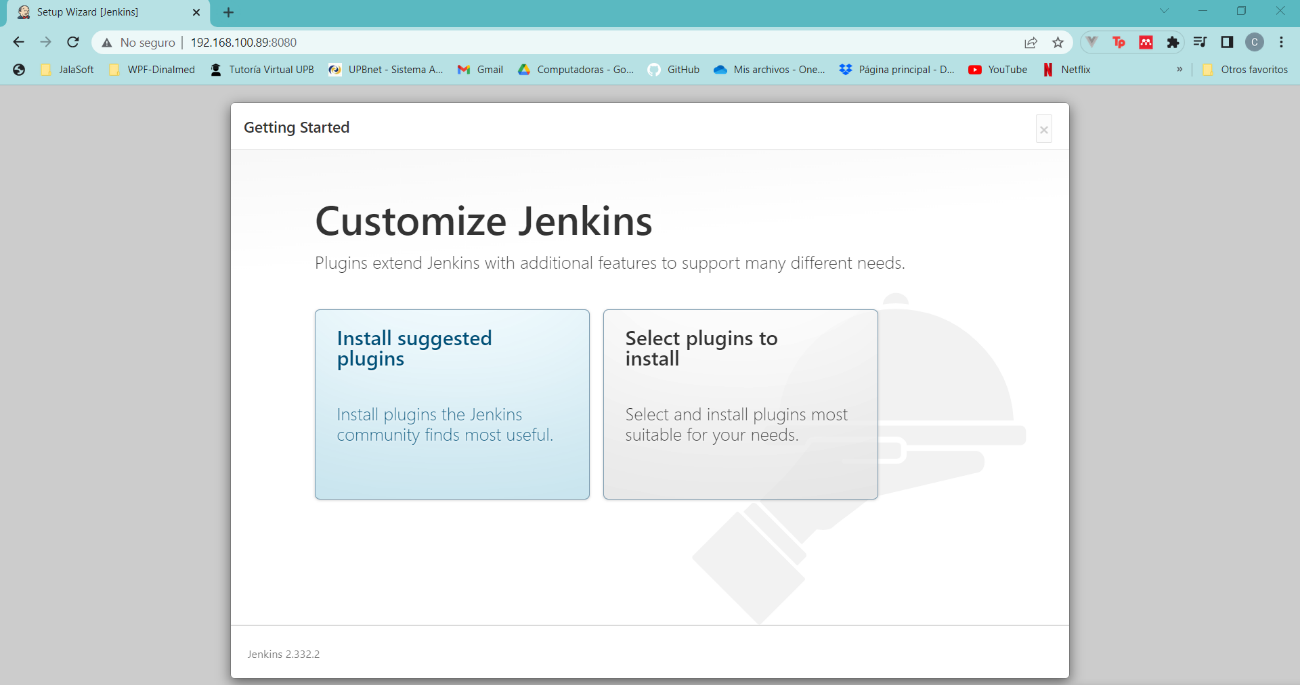
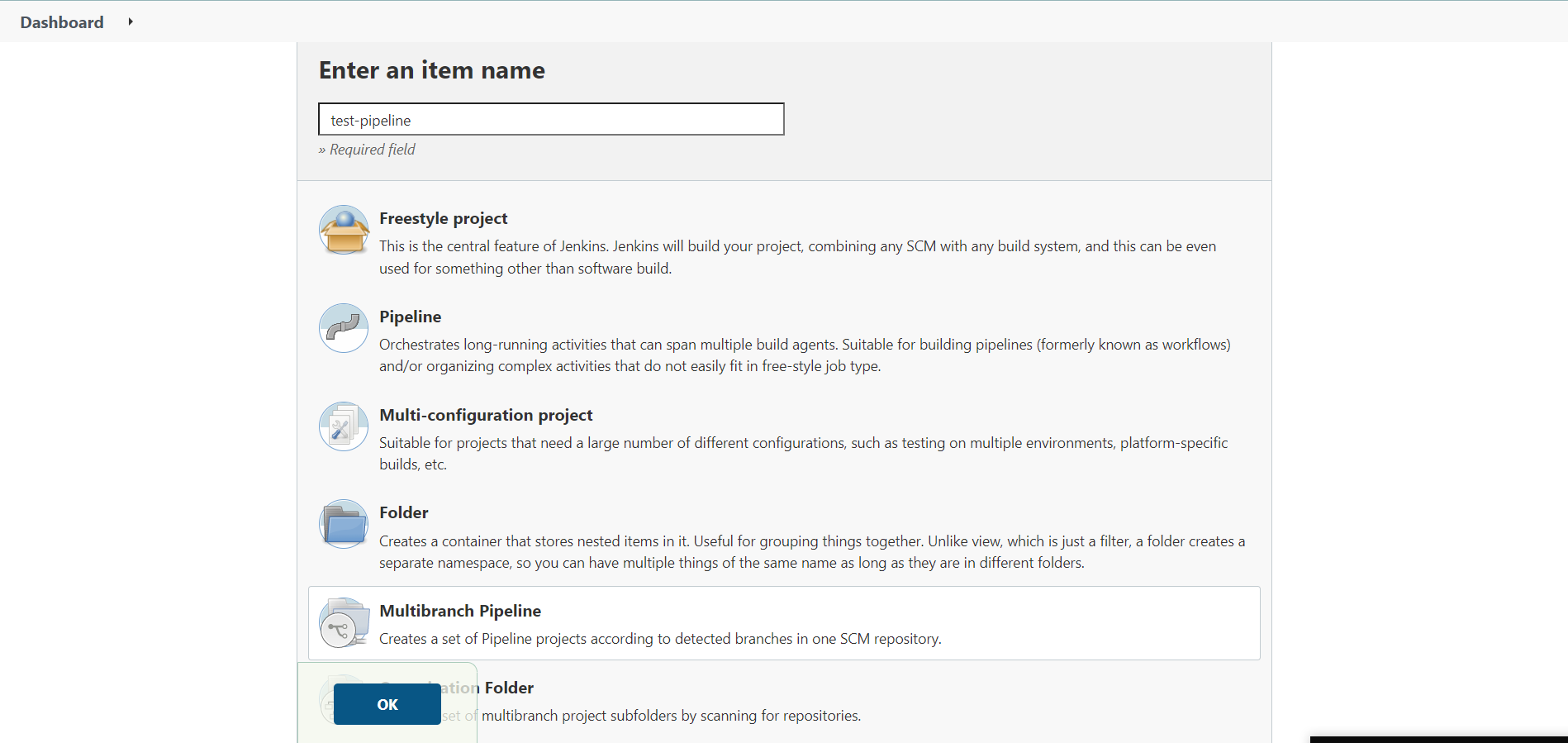
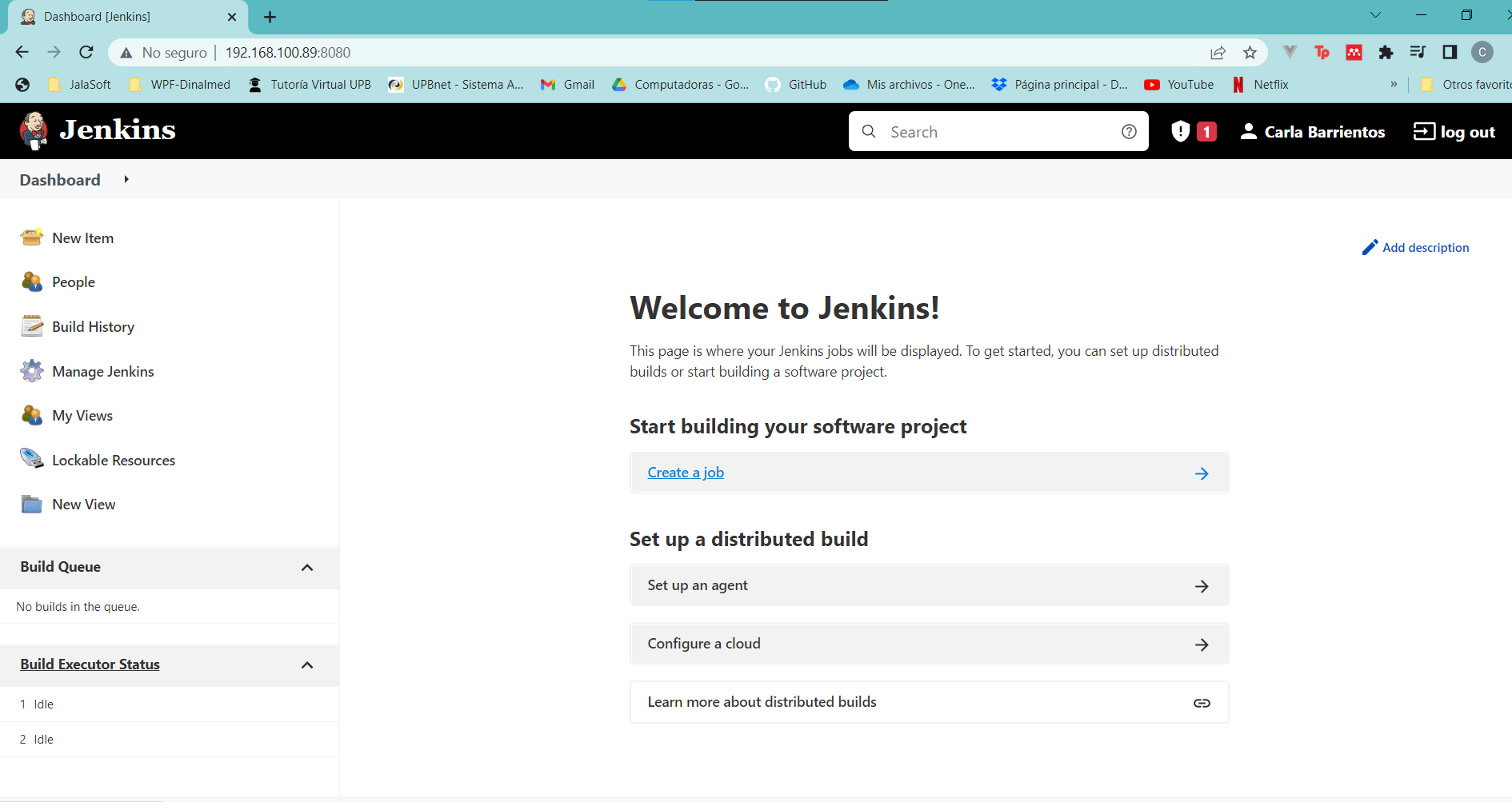
**CI Configuration on Jenkins**

First, we need to start Jenkins in a Docker container, to do so we run the commands shown in the next image:

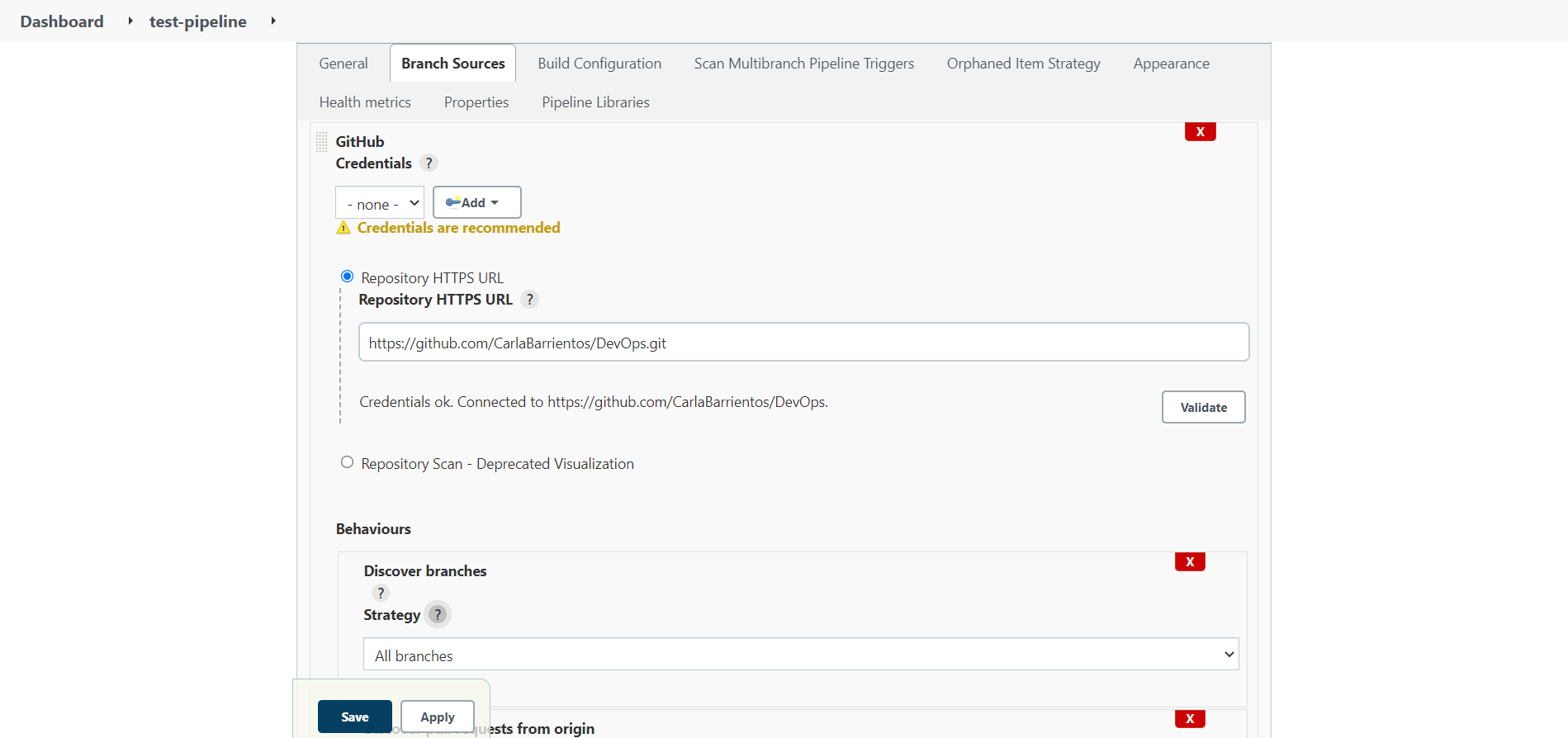


After running this container, we can access the logs to get the administrator password to start configuring Jenkins on the web browser: 

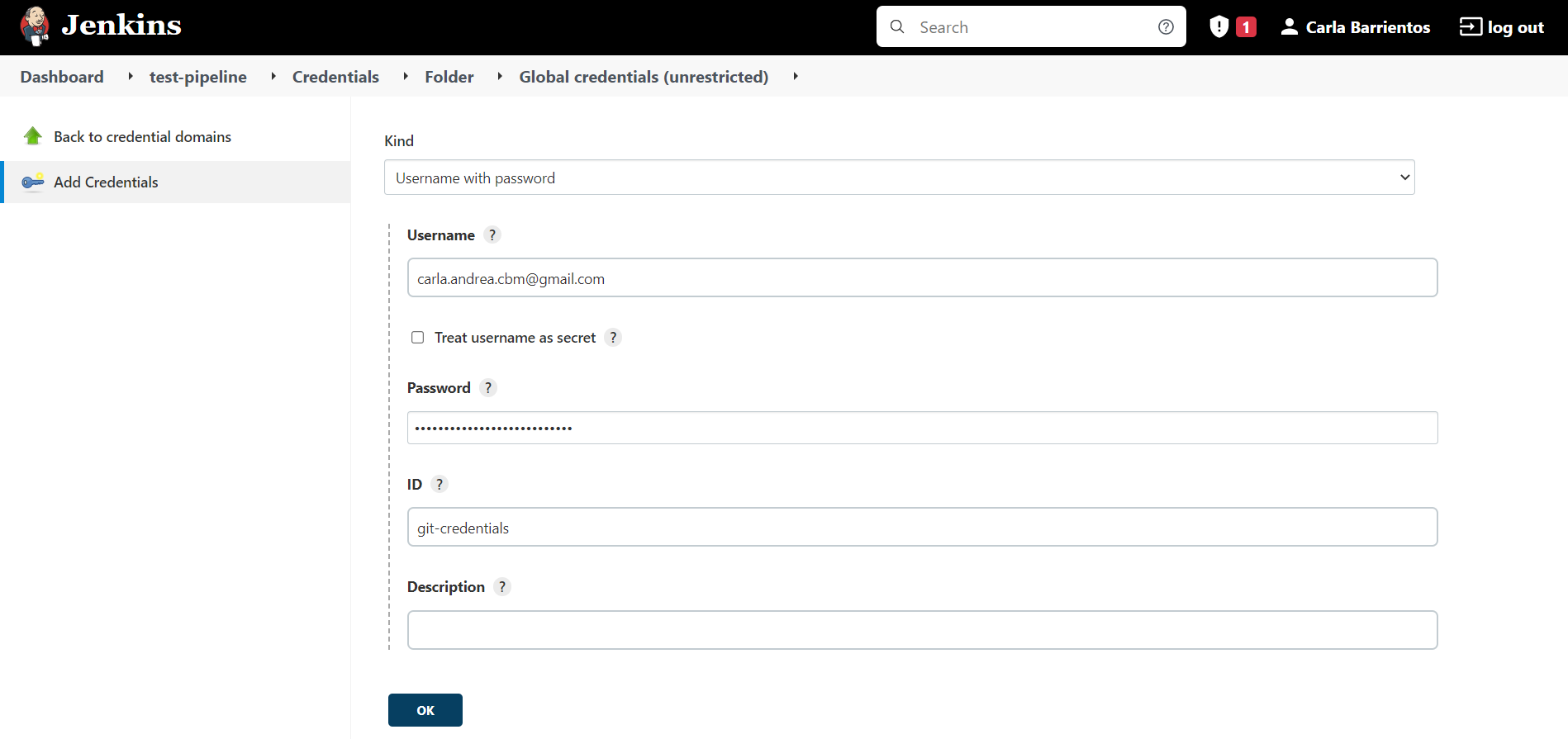
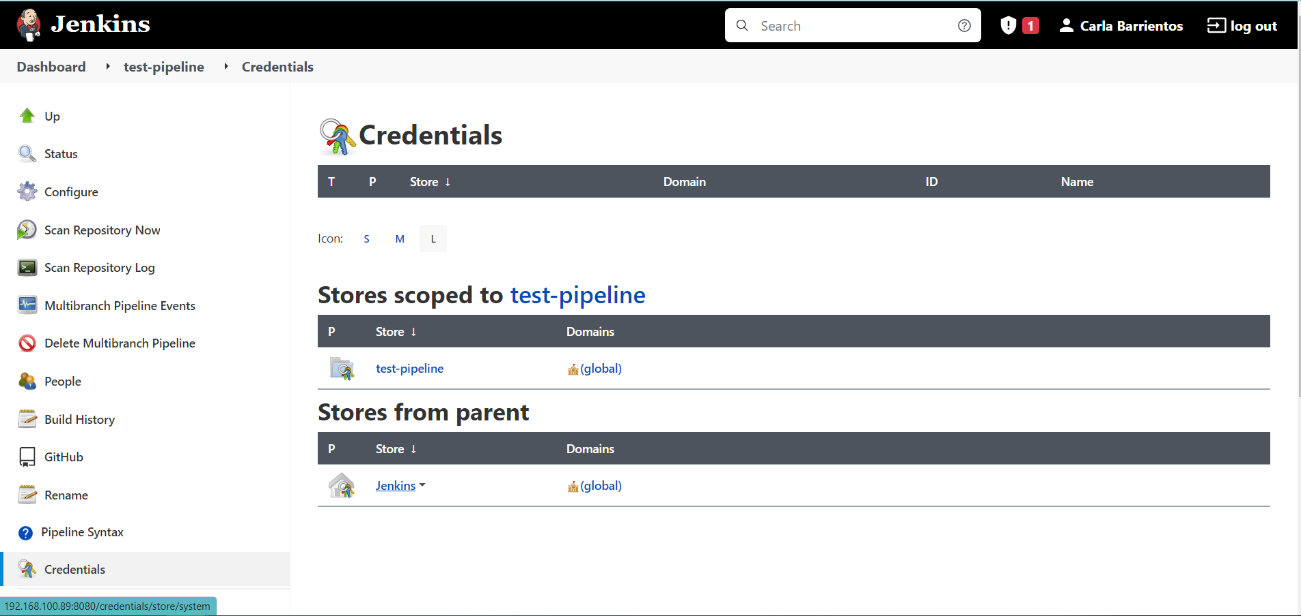
Next we follow the configuration menu, we select to install suggested plugins, create the first admin userand check that the URL of the Jenkins instance is correct:

Then we select the option to create a new job, in this case we named it test-pipeline and selected the *Multibranch Pipeline* option:

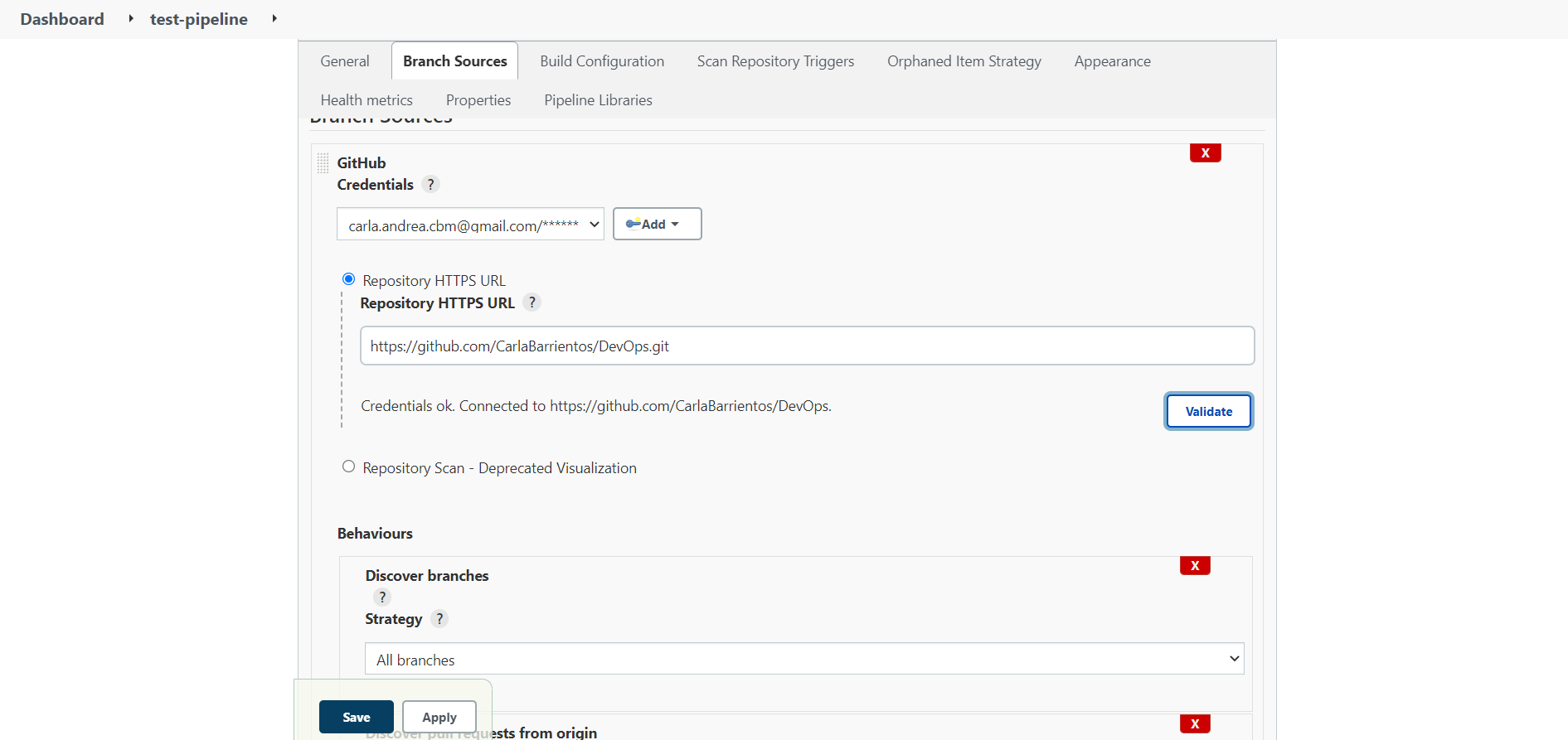
We copy the link of the GitHub repo that we want to track, and for experimental propouse we change the strategy of the behaviours to *All branches*:



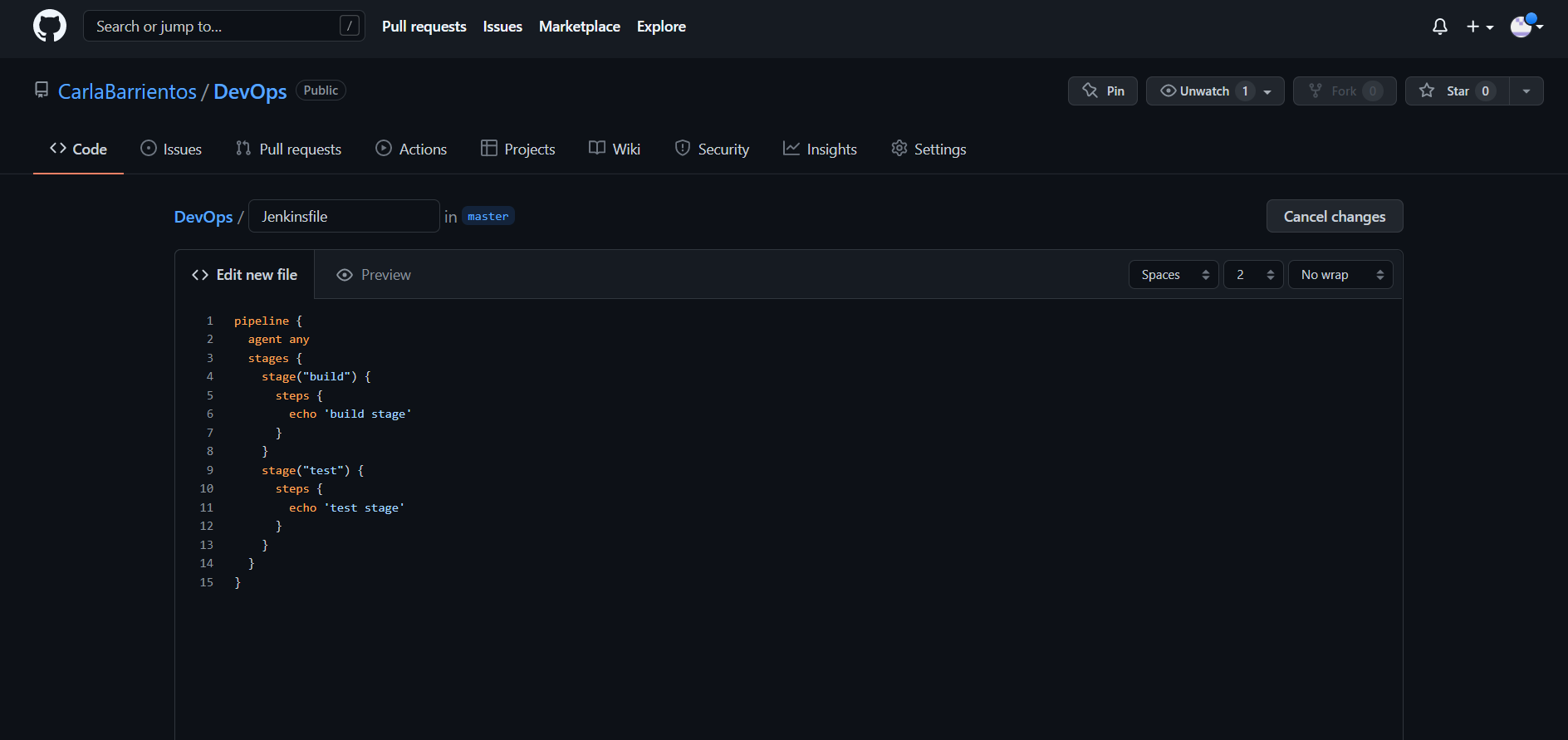
Then we need to create a credential for the pipeline we created, we select test-pipeline and fill the form to create a new credential:

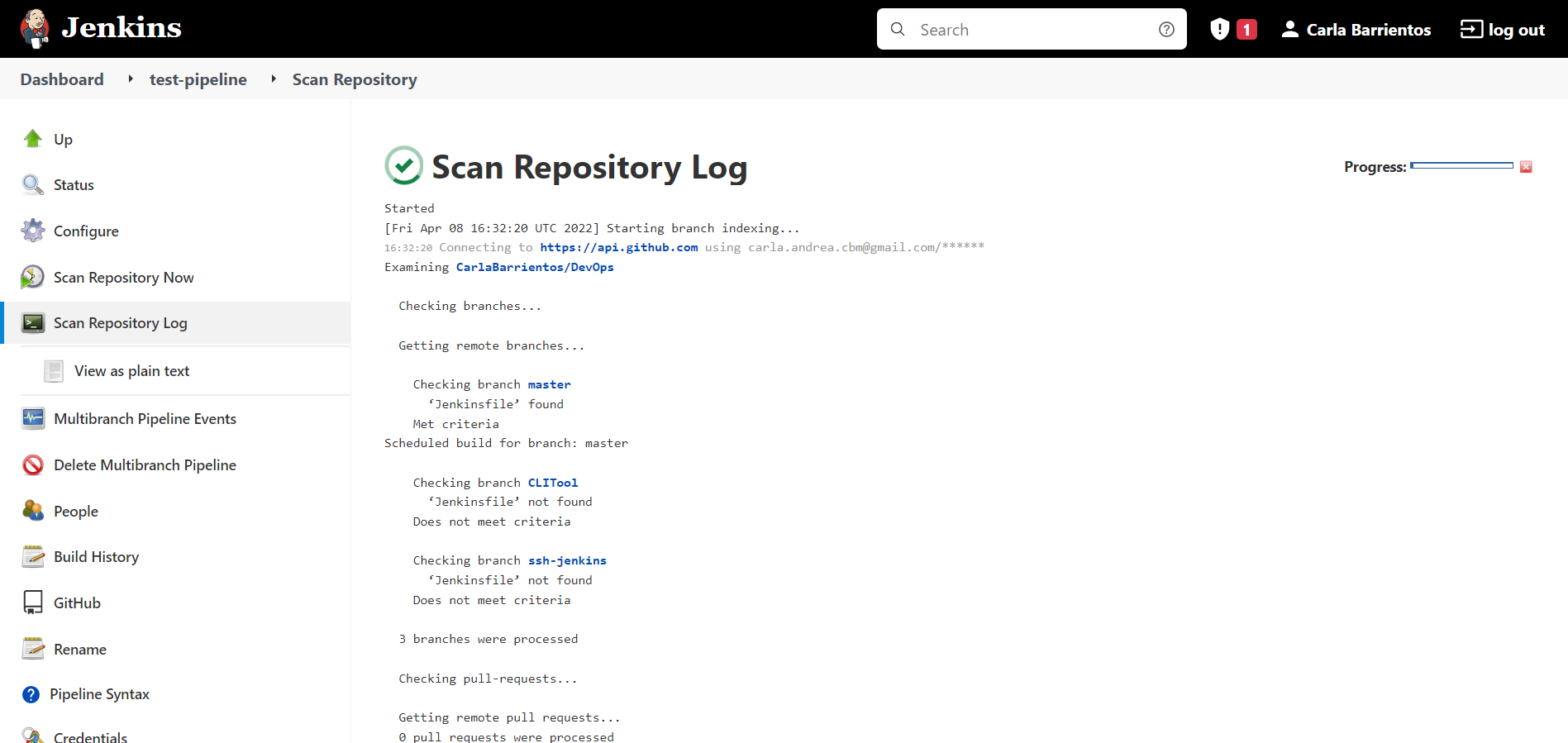


After that, we add the credential created to the GitHub credentials in our pipeline:



Finally, we need to add a Jenkinsfile in our repo, in this case we added it to master branch with the next content:



To check that everything is working we can scan the repository and check the logs, as we can see the Jenkinsfile has been found: 

We can also have more information about the different stages that our Jenkinsfile has, here we can see which one passed or failed and have more information about the errors:

