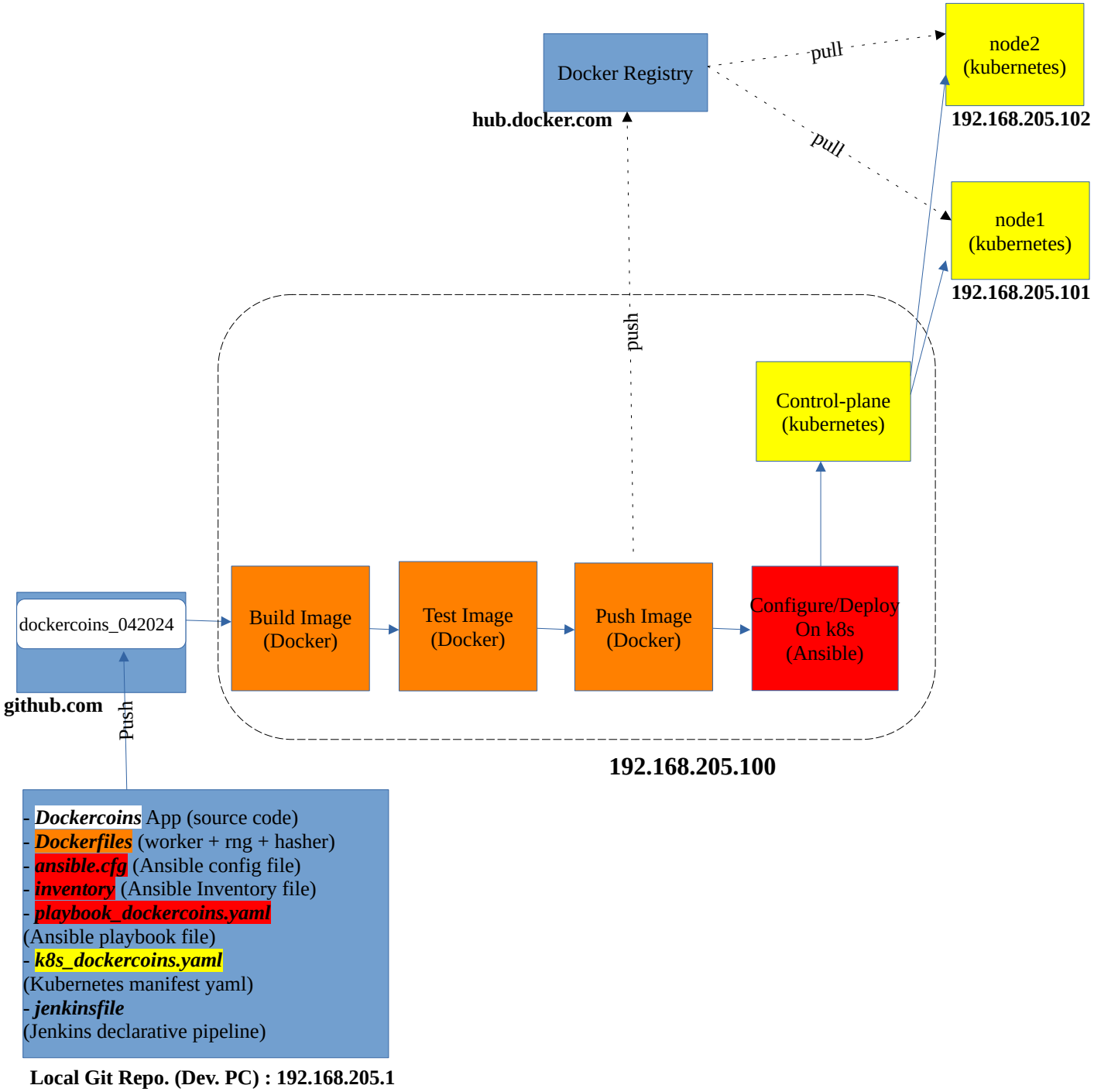
Lab 9 – Jenkins

Architecture :



Environment :

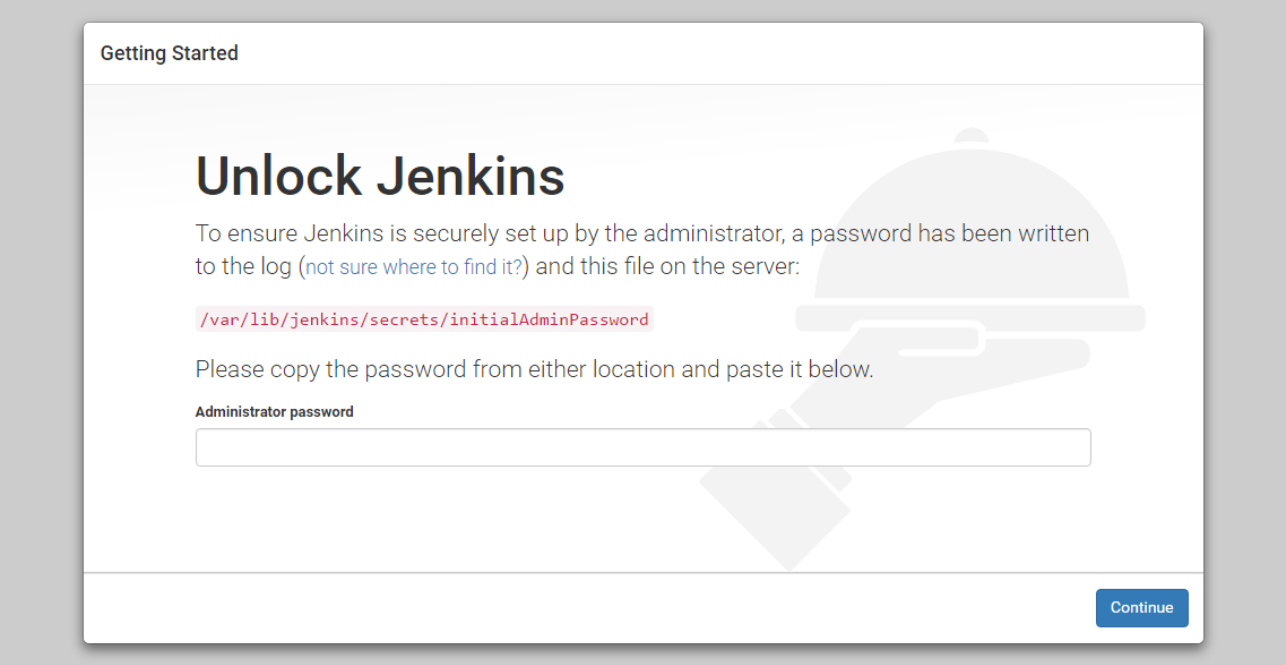
- Local (Dev. PC) : 192.168.205.1
- Git Server : github.com / gitlab.com
- Docker Registry : hub.docker.com
- Jenkins : 192.168.205.100
- Ansible : 192.168.205.100
- k8s-control-plane (master) : 192.168.205.100

Installing Jenkins :

1. Install OpenJDK17 on Jenkins node with this command :
   sudo apt install openjdk-17-jdk

2. Install Jenkins using the following link :
   https://www.jenkins.io/doc/book/installing/linux/#debianubuntu

3. Visit Jenkins on its default port, `8080`, using the server IP address :
   http://192.168.205.100:8080
   You should receive the **Unlock Jenkins** screen, which displays the location of the initial password.



4. Copy the password from the **`/var/lib/jenkins/secrets/initialAdminPassword`** file and paste it into the **Administrator password** field, then click **Continue**.

5. The next screen presents the option of installing suggested plugins or selecting specific plugins.

   click the **Install suggested plugins** option, which will immediately begin the installation process.

6. When the installation is complete, you'll be prompted to set up the first administrative user. It's possible to skip this step and continue as `admin` using the initial password from above, but we'll take a moment to create the user.

7. Next, you'll receive an **Instance Configuration** page that will ask you to confirm the preferred URL for your Jenkins instance. Confirm your server's IP address 192.168.205.100

8. After confirming, click **Save and Finish**. You'll receive a confirmation page confirming that **"Jenkins is Ready!"**. At this point, you have completed a successful installation of Jenkins.

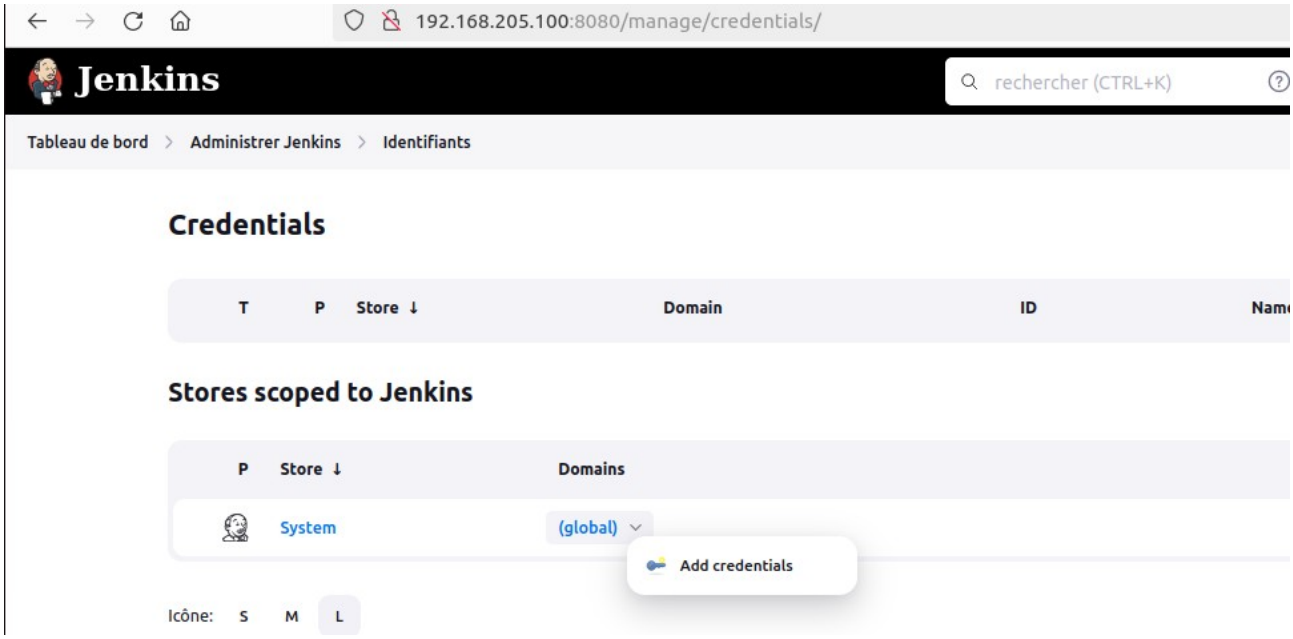
Installing and configuring Docker on Jenkins:

9. Install ***docker-ce***, ***gnupg2*** and ***pass*** packages on Jenkins node using the following command :
   sudo apt install -y docker-ce
   sudo apt install -y gnupg2 pass

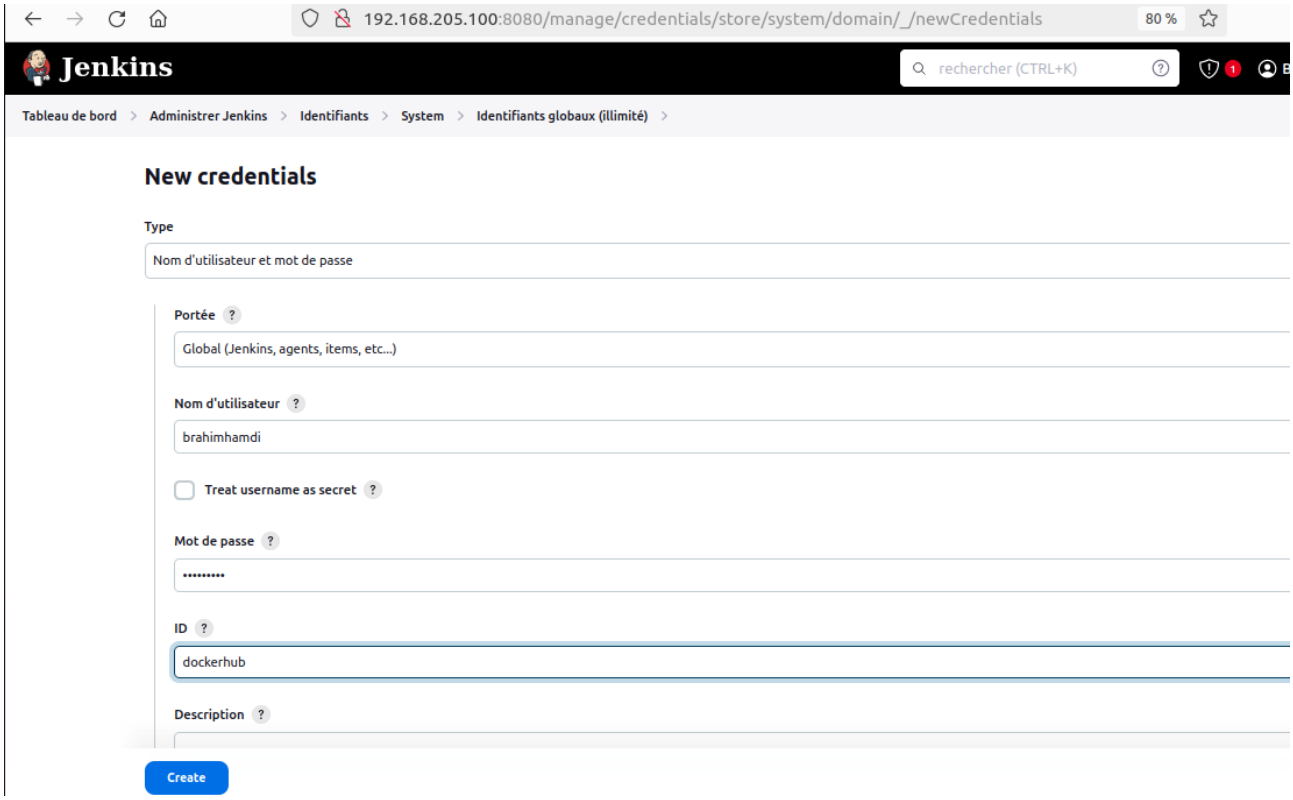
10. Add jenkins user to docker group then restart jenkins service.
    sudo usermod -aG docker jenkins
    sudo systemctl restart jenkins

11. On Jenkins Dashboard, click on ***Manage Jenkins*** > ***Plugins*** > ***Available plugins***, then search ***docker*** and ***docker pipeline*** plugins and install it.

12. Click on ***Manage Jenkins > Credentials > Add Creadentials*** to create DockerHub credentials



13. Add your DockerHub username and password, and put dockerhub as ***ID***, then click on ***create***



14. Execute following commands to run kubectl as jenkins user

    sudo mkdir /var/lib/jenkins/.kube

    sudo cp -i /etc/kubernetes/admin.conf /var/lib/jenkins/.kube/config

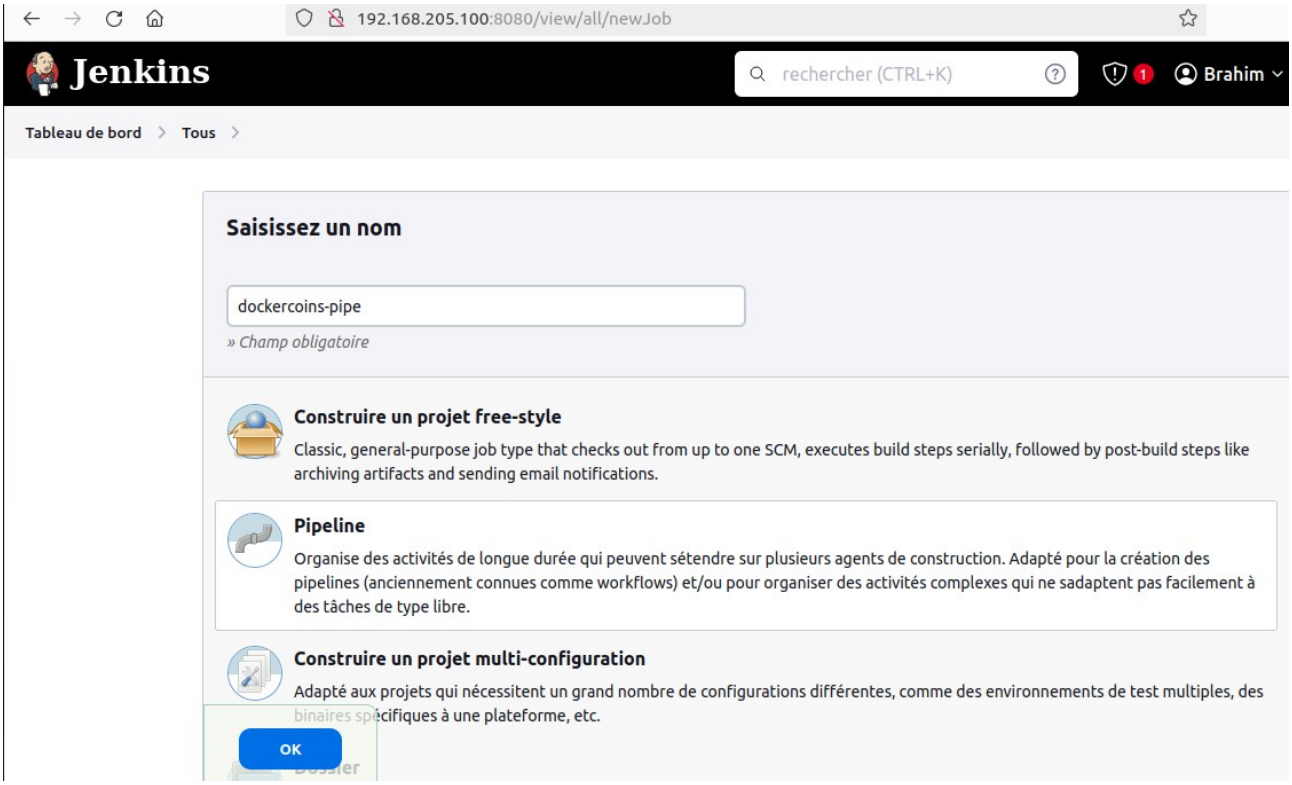
    sudo chown jenkins.jenkins /var/lib/jenkins/.kube/config

Create and build Jenkins pipeline:

15. From the Jenkins home page (i.e. the Dashboard of the Jenkins classic UI), click **New Item** at the top left.

16. In the **Enter an item name** field, specify the name for your new Pipeline project (example dockercoins-pipe)

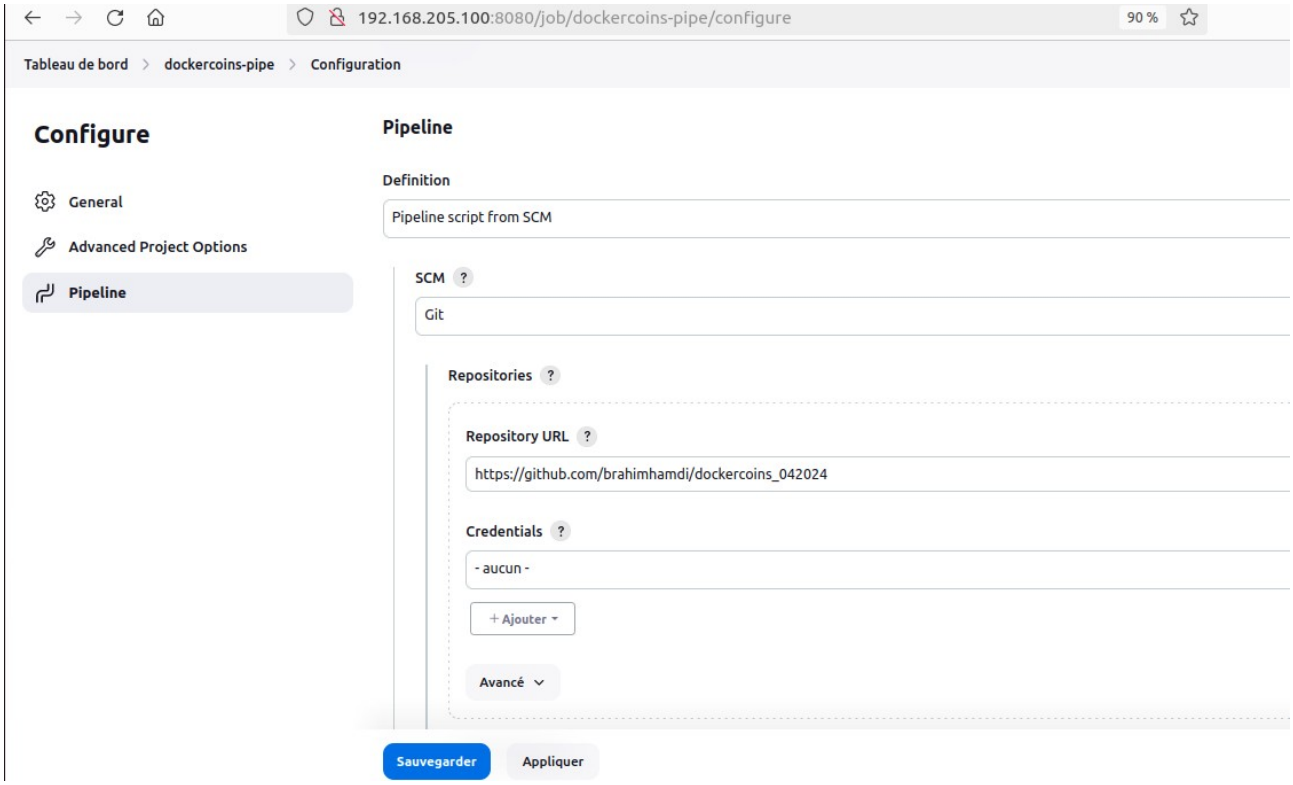
17. Scroll down and click **Pipeline**, then click **OK** at the end of the page to open the Pipeline configuration page (whose **General** tab is selected).



18. Click the **Pipeline** tab at the top of the page to scroll down to the **Pipeline** section.

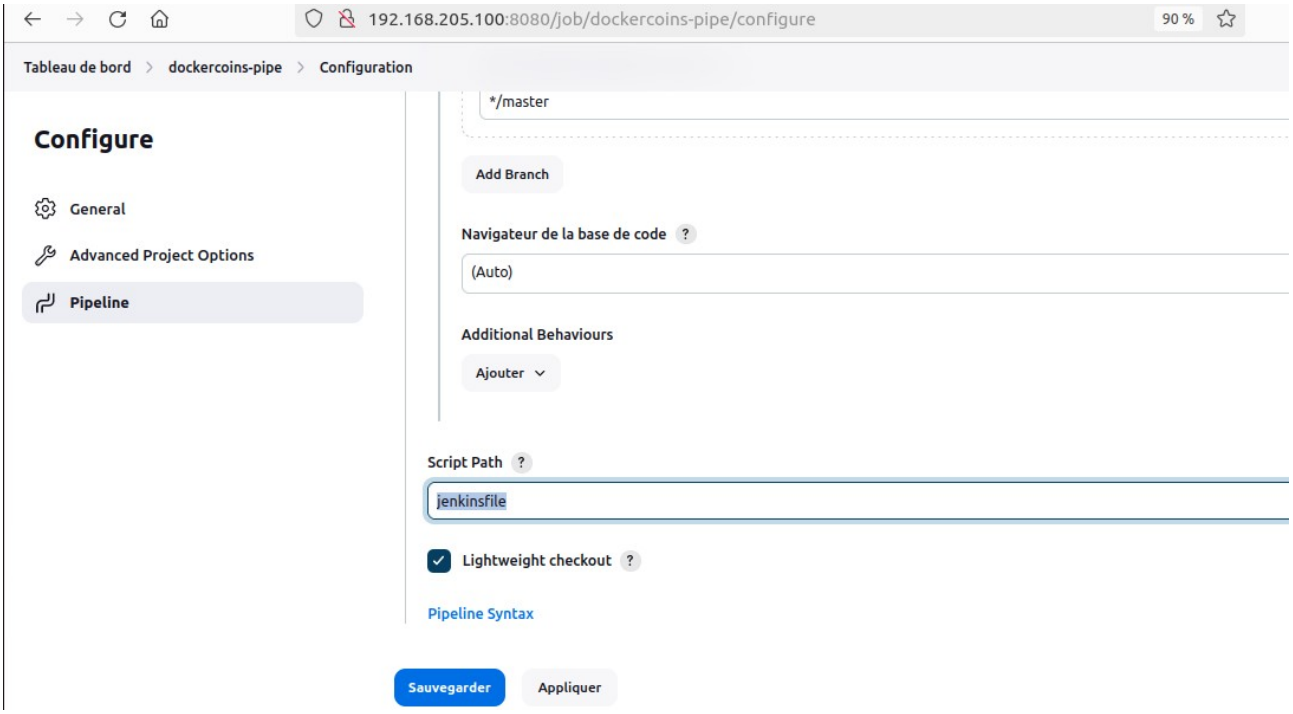
19. From the **Definition** field, choose the **Pipeline script from SCM** option.

20. From the **SCM** field, choose Git and add the repository URL containing your `Jenkinsfile`.



21. In the **Script Path** field, specify the location (and name) of your `Jenkinsfile`. This location is the one that Jenkins checks out/clones the repository containing your `Jenkinsfile`, which should match that of the repository's file structure. The default value of this field assumes that your `Jenkinsfile` is named "Jenkinsfile" and is located at the root of the repository.

...



22. Click on ***save***, then ***launch build***.

23. After successful build of pipeline check if application is running on kubernetes cluster.