PROBLEM STATEMENT

1. Create container image that’s has Python3 and Keras or numpy installed using dockerfile

2. When we launch this image, it should automatically starts train the model in the container.

3. Create a job chain of job1, job2, job3, job4 and job5 using build pipeline plugin in Jenkins

4. Job1 : Pull the Github repo automatically when some developers push repo to Github.

5. Job2 : By looking at the code or program file, Jenkins should automatically start the respective machine learning software installed interpreter install image container to deploy code and start training( eg. If code uses CNN, then Jenkins should start the container that has already installed all the softwares required for the cnn processing).

6. Job3 : Train your model and predict accuracy or metrics.

7. Job4 : if metrics accuracy is less than 80% , then tweak the machine learning model architecture.

8. Job5: Retrain the model or notify that the best model is being created

9. Create One extra job job6 for monitor : If container where app is running. fails due to any reason then this job should automatically start the container again from where the last trained model left

SOLUTION

1. ****Run you jenkins ,docker in rehl8.****

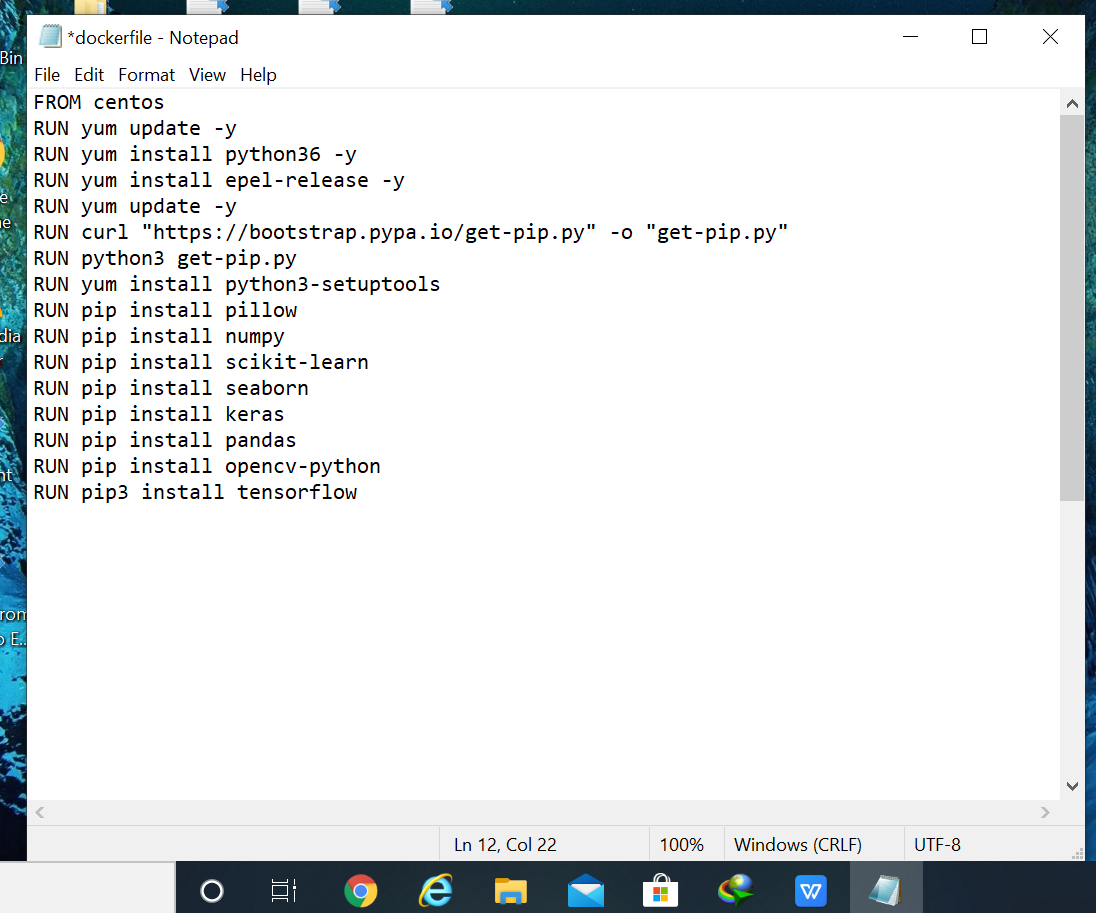
systemctl start jenkins

systemctl start docker

1. ****Next, I created a folder in my system where my dockerfile would be there, also i created a directory where all the files from github would be downloaded!****

mkdir /root/mlops/task3

cd task3/



mkdir /root/task3/

1. ****All my files would be downloaded from github.****
2. ****Build the image by running this****

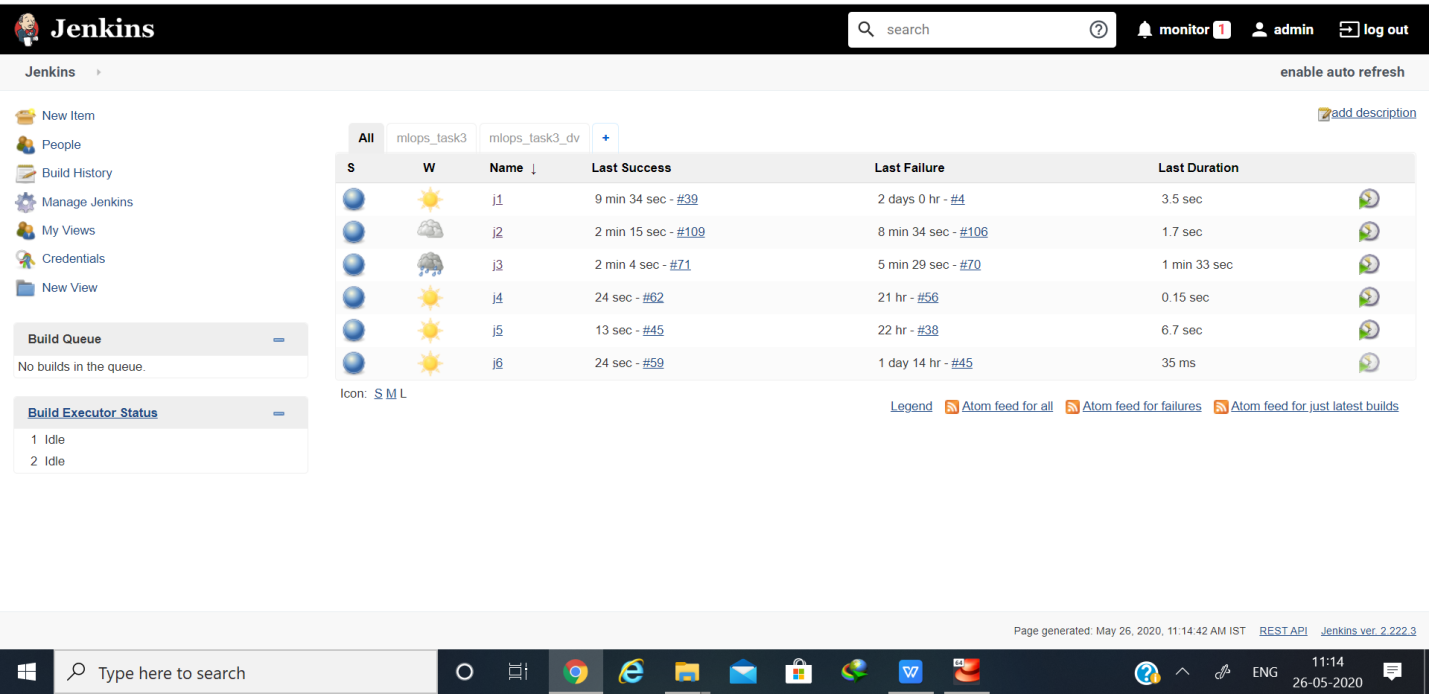
docker build -t mytensor:v1 task3/

1. ****Go to windows and add all the files to be uploaded in a repo .Run git and the following commands****

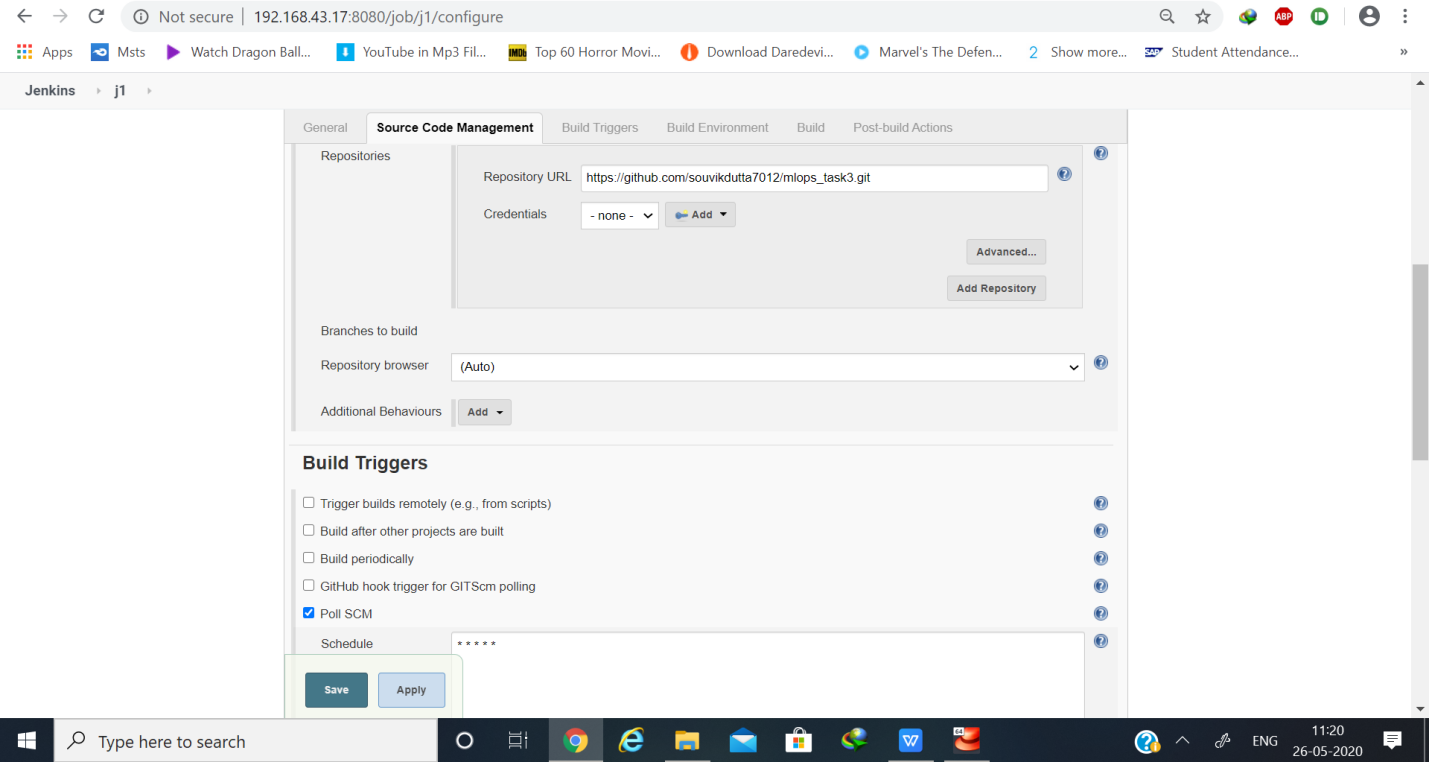
git comit . -m “v1”

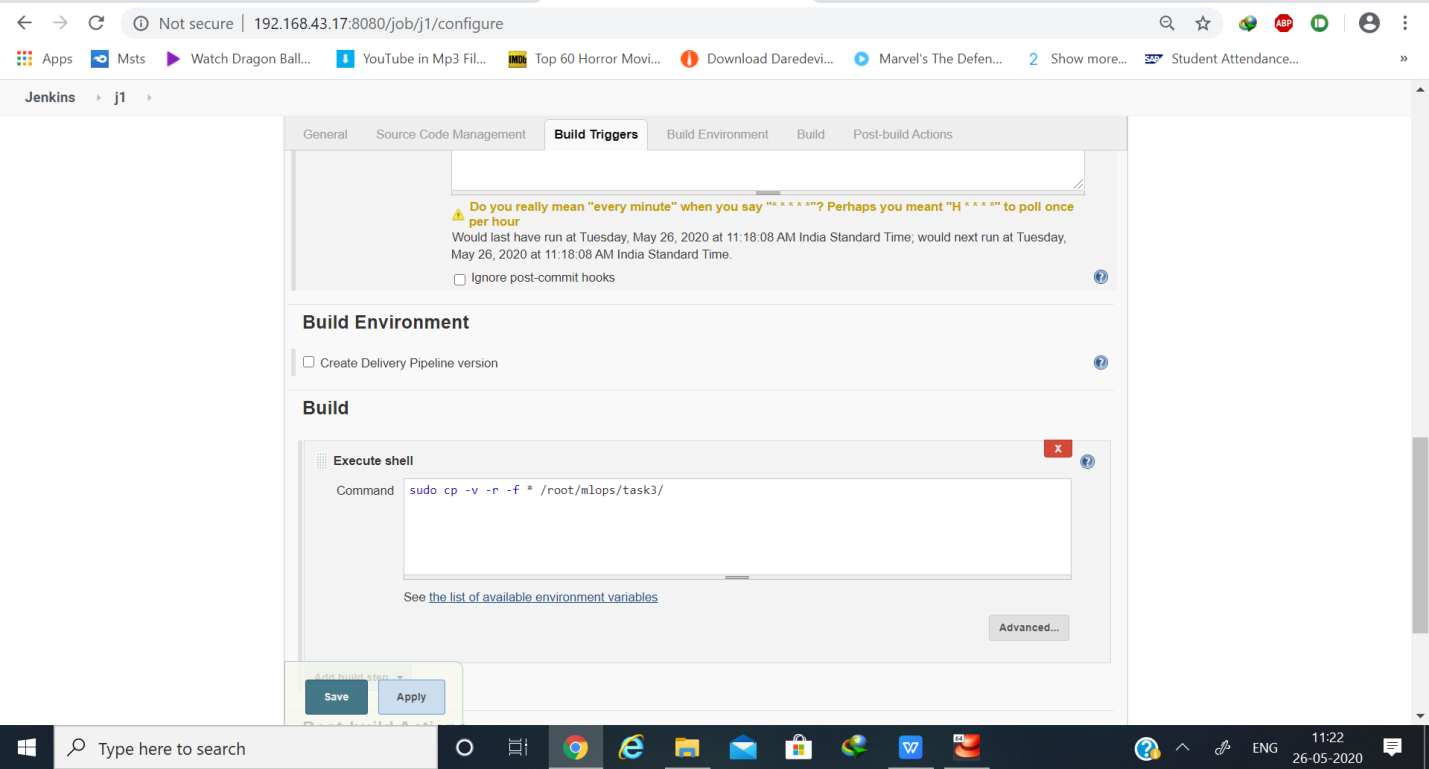
git push

1. ****I have created 6 jobs in jenkins****

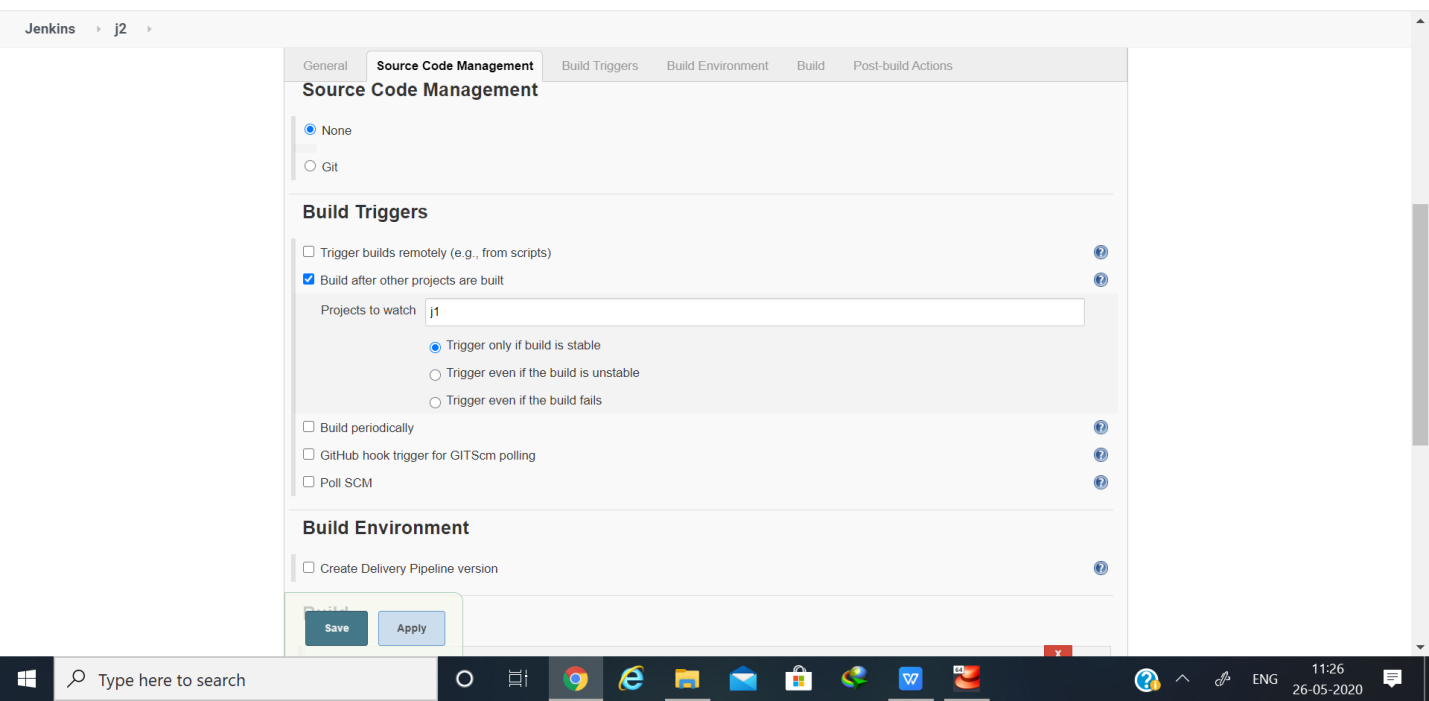


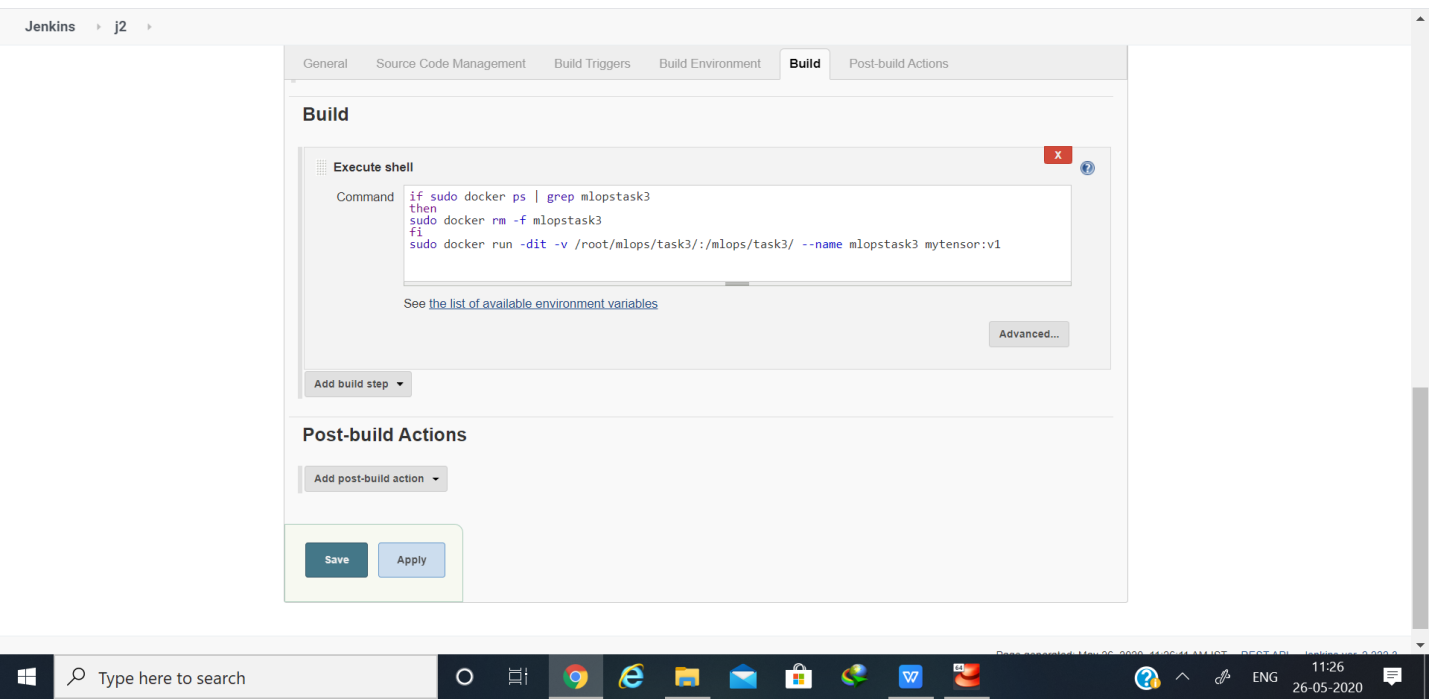
****Job 1: This will automatically pull all the files from github only if it has been changed by developer. It will check after every one minute. It will also copy my code to folder :- /root/mlops/task3/****

********

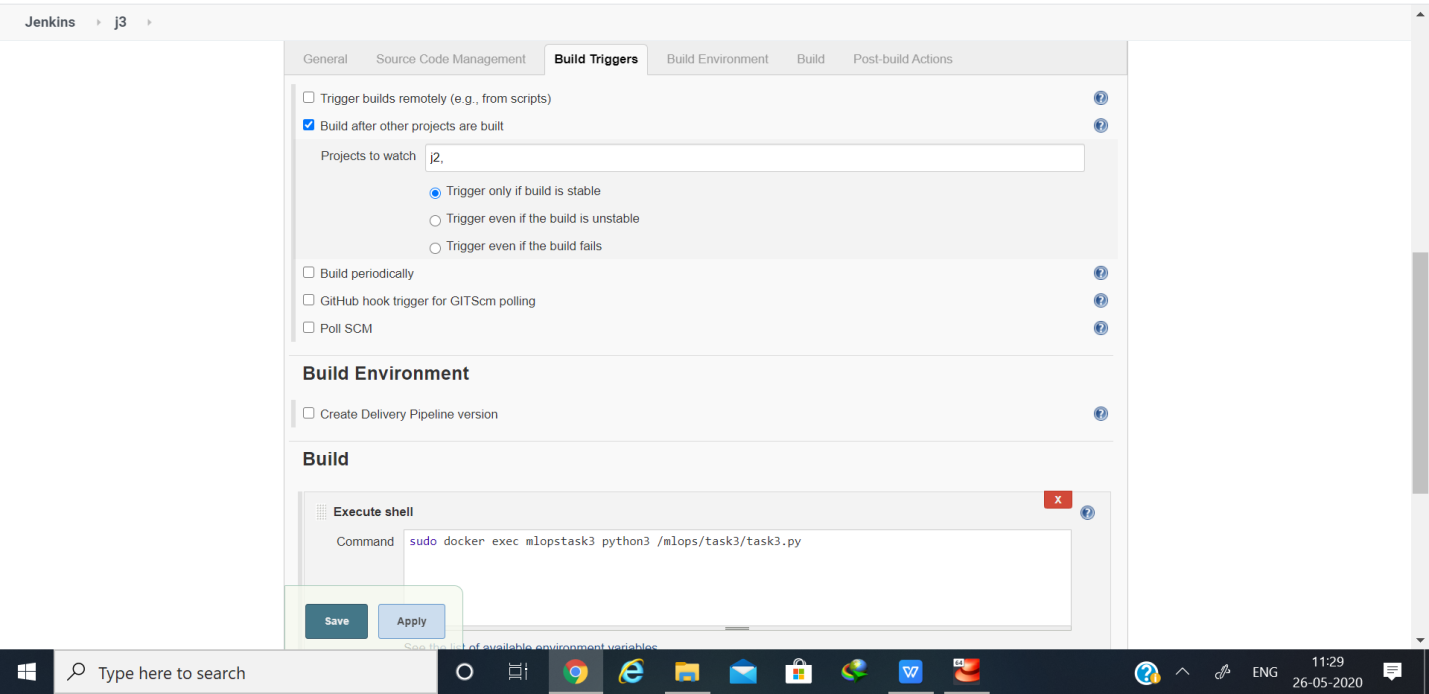
********

****Job 2: This job will launch my deeplearning environment. And it will only get triggered if job1 runs successfully.****

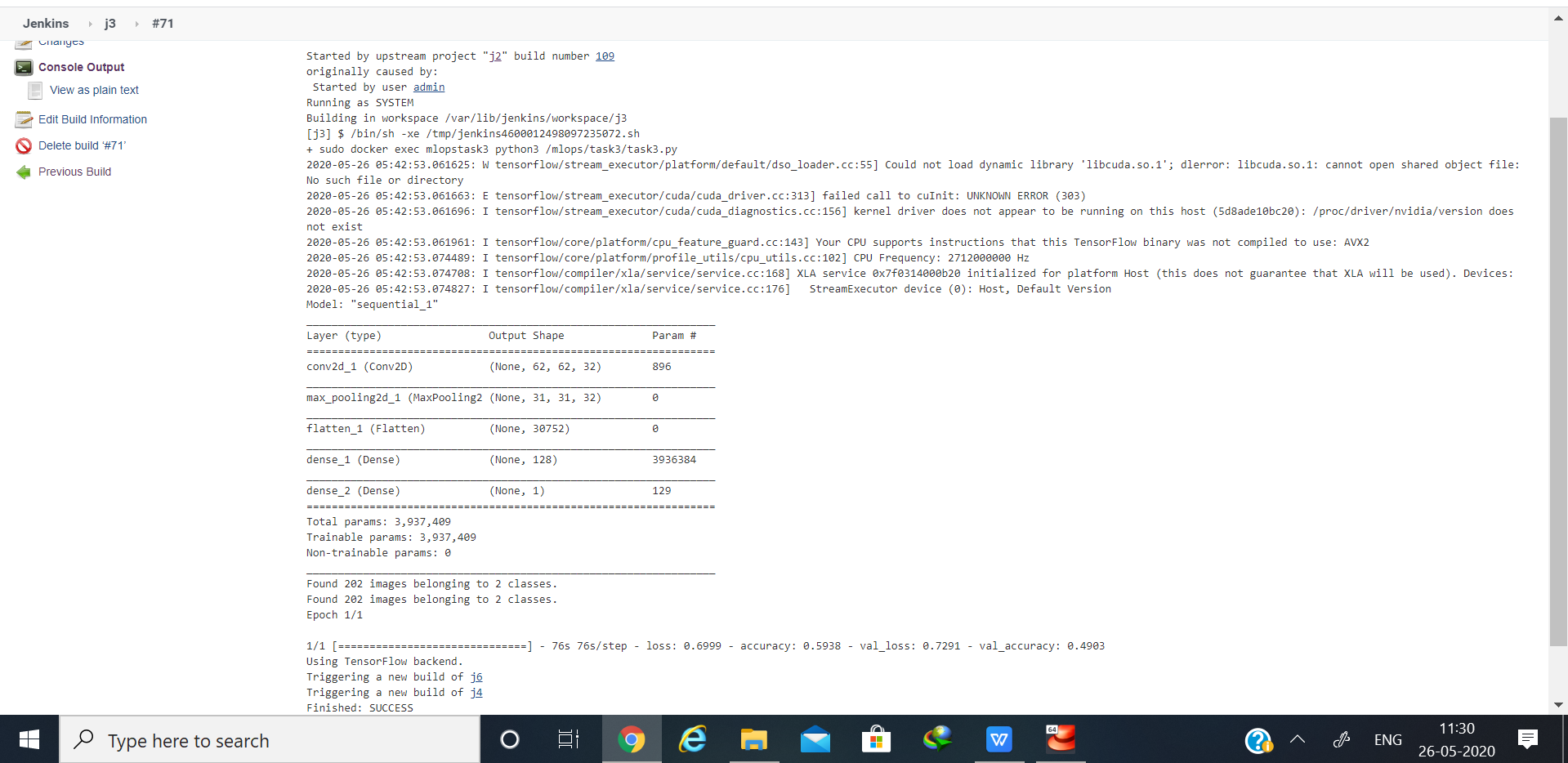
********

********

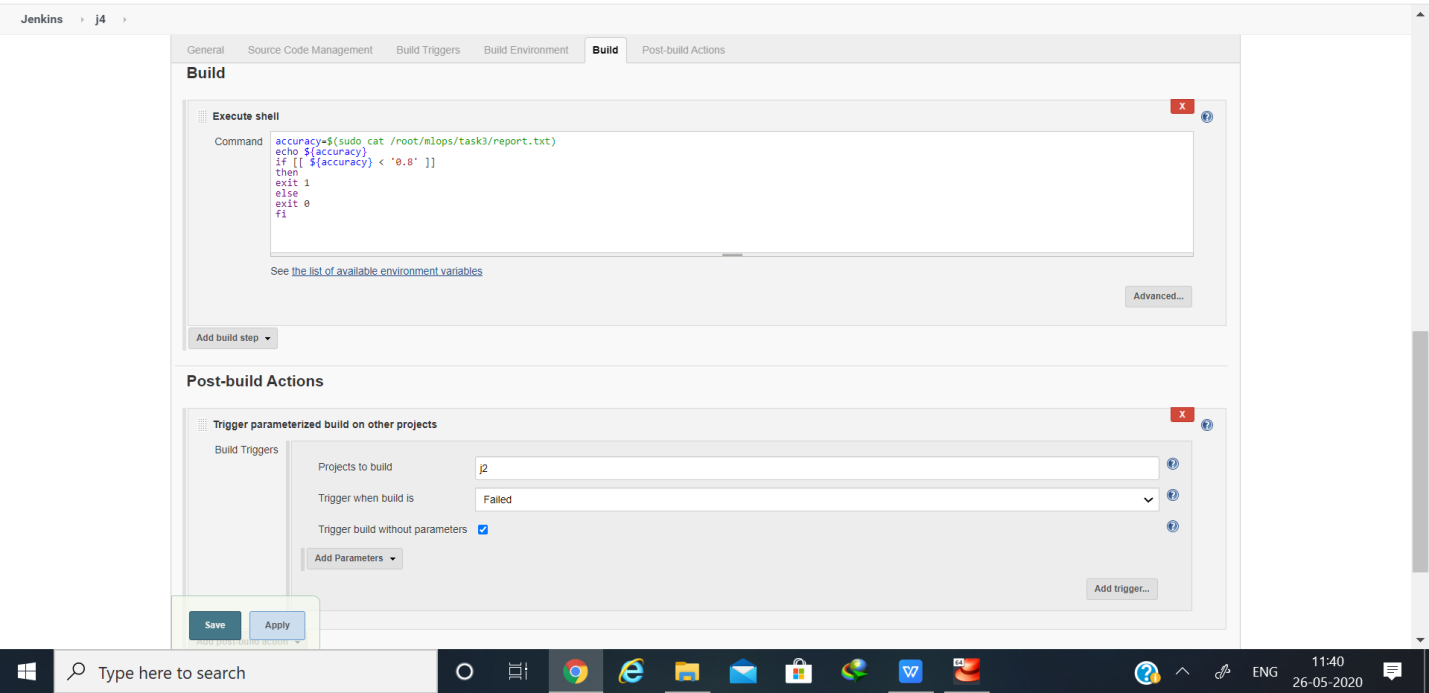
****Job 3: This job will train my model .****

********

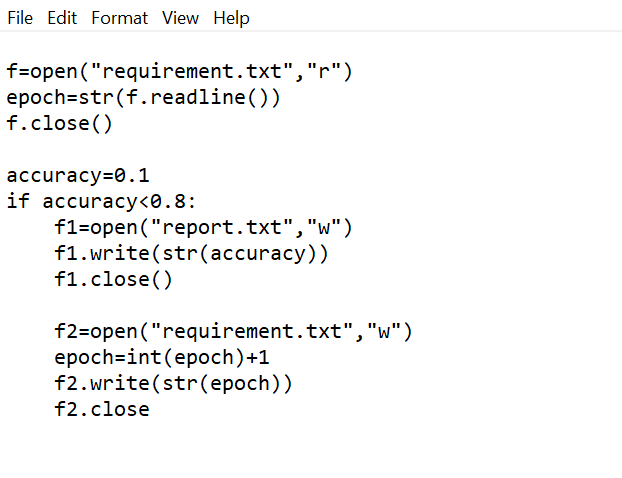
****This my console output of job3****

********

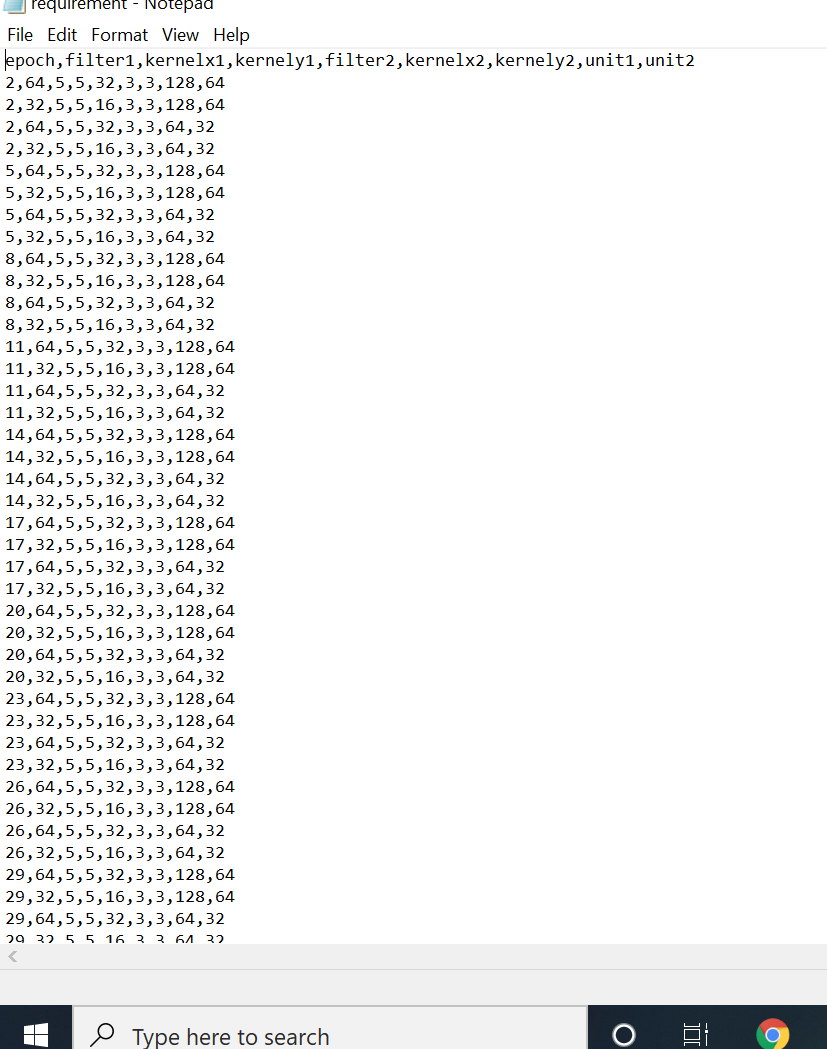
****Job 4: This will check my accuracy and if accuracy < 80 % then it will fail itself which will trigger Job2 with new requirements for the model but if accuracy > = 80% then it will run successfully which triggers job5.****

********

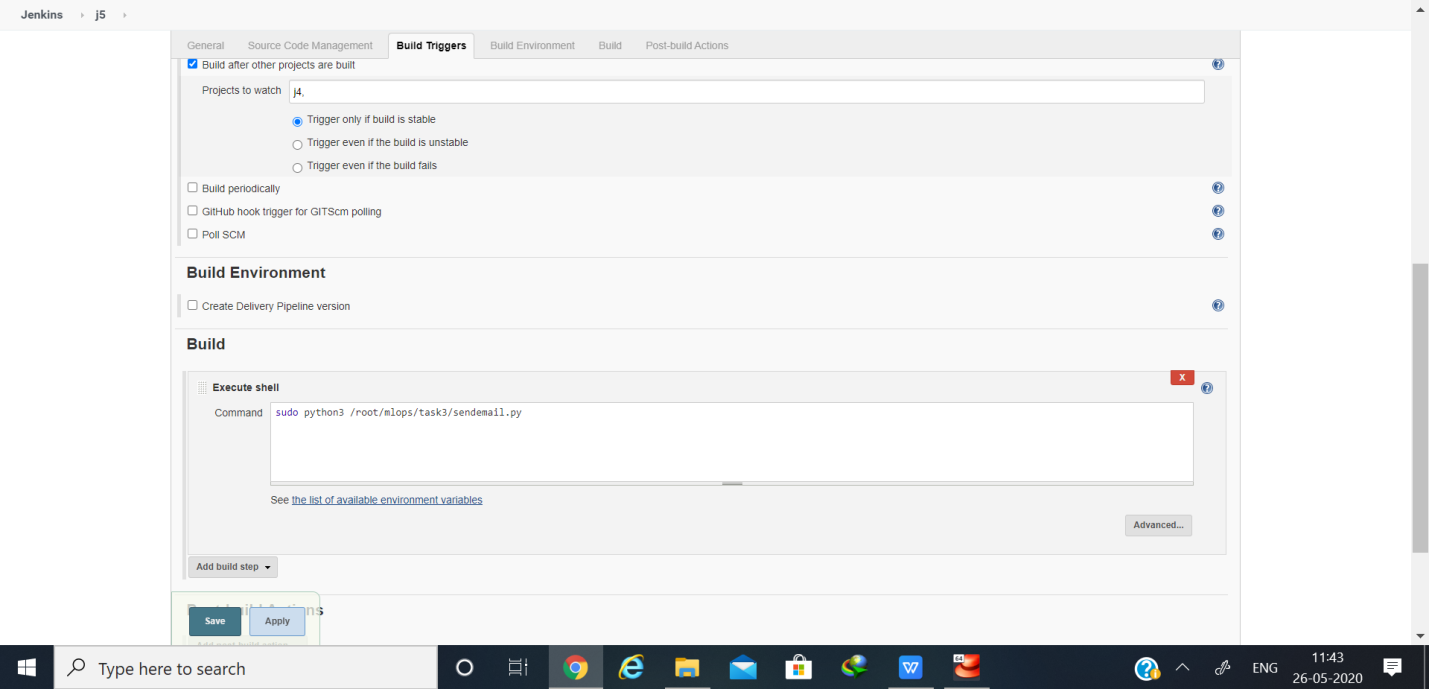
****Demo code to show how I am twicking my model.Basically I am just increasing my number of epochs.****

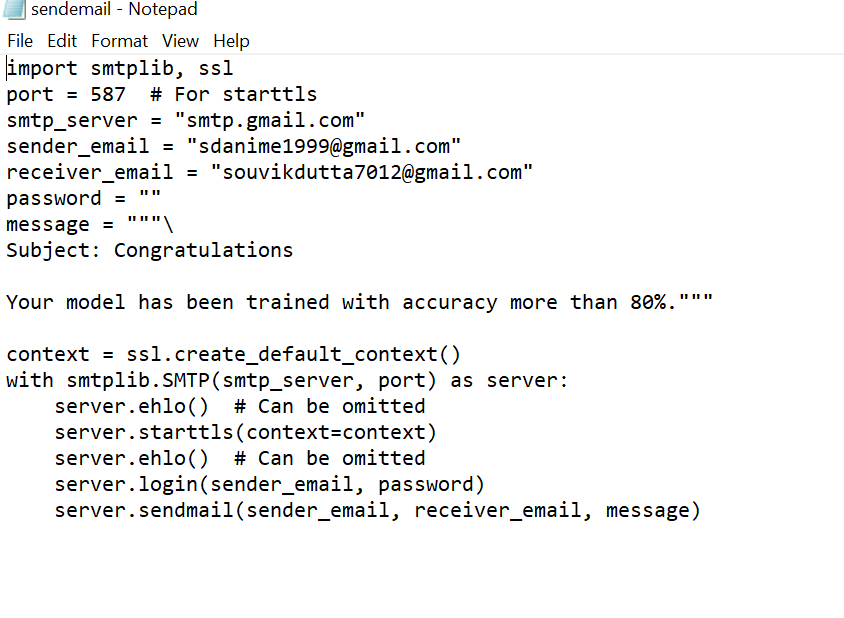
********

****My requirement.csv file where al the combinations of filters ,units and kernel size is peresent****

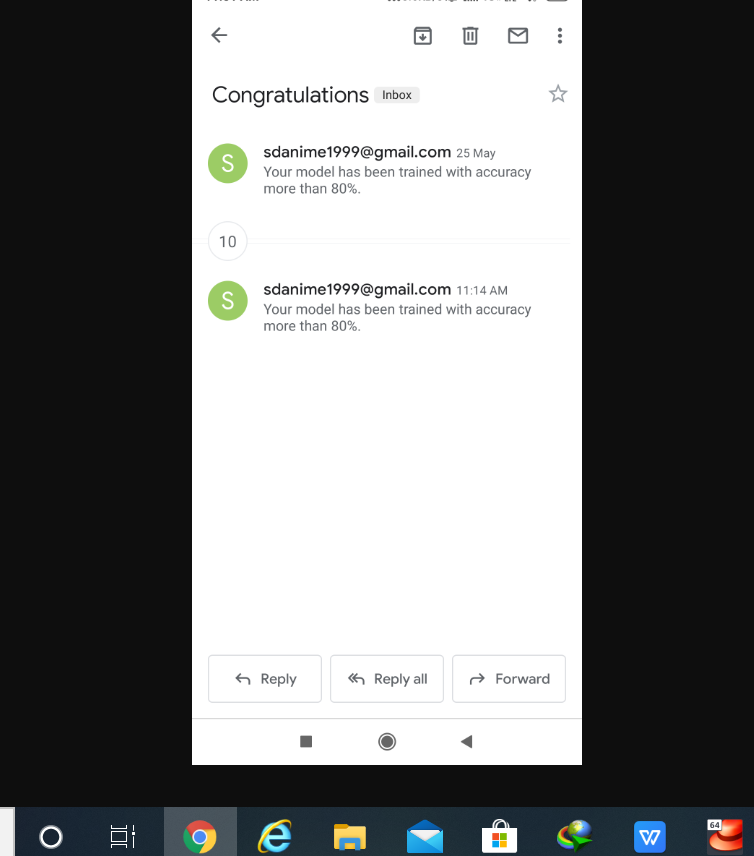
********

****Job 5: This job will send the mail to the developer that the model accuracy is good and accuracy is more than 80%****

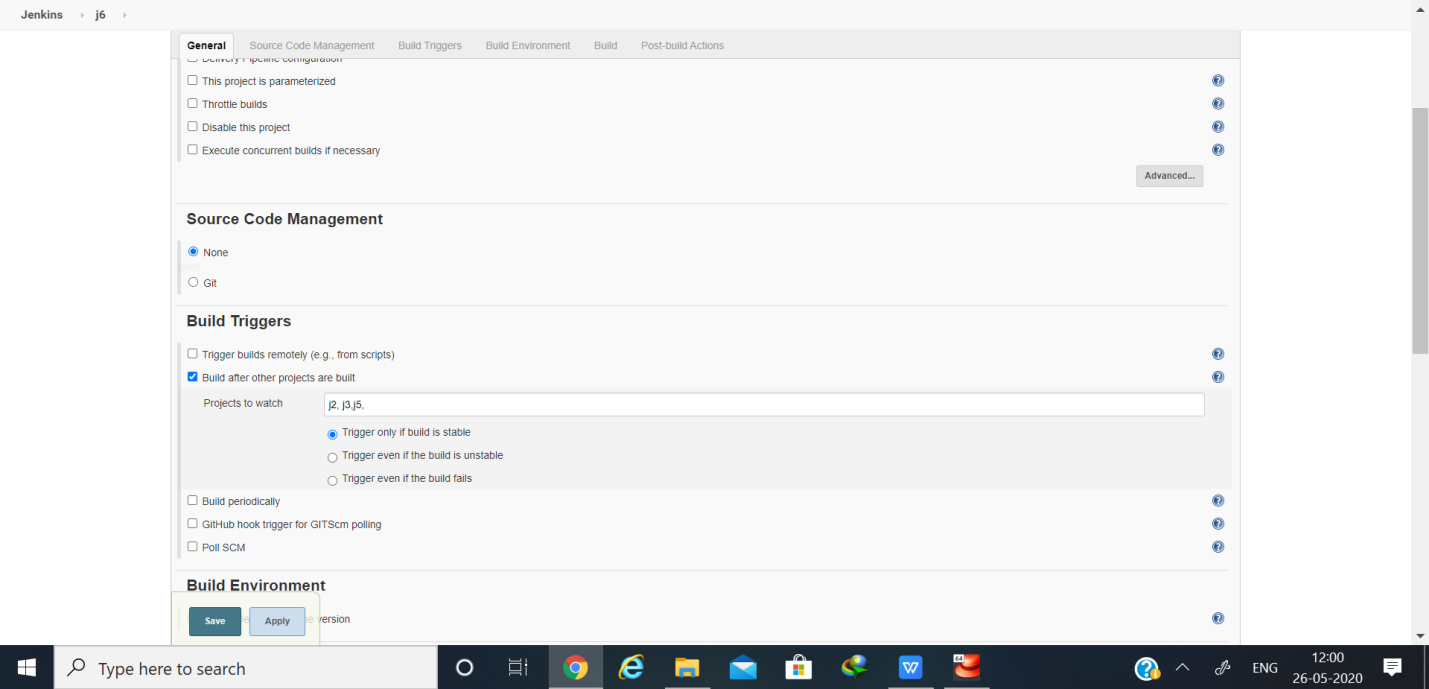
********

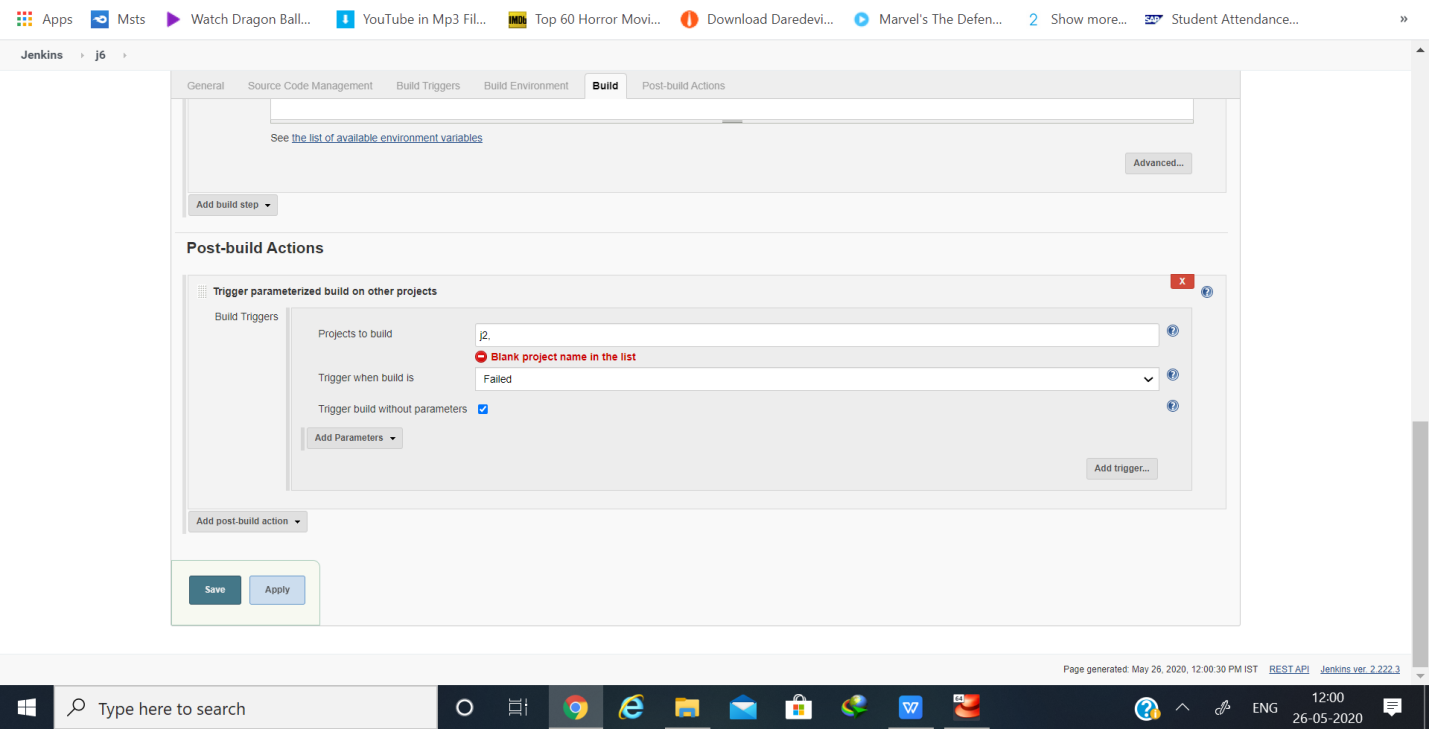
********

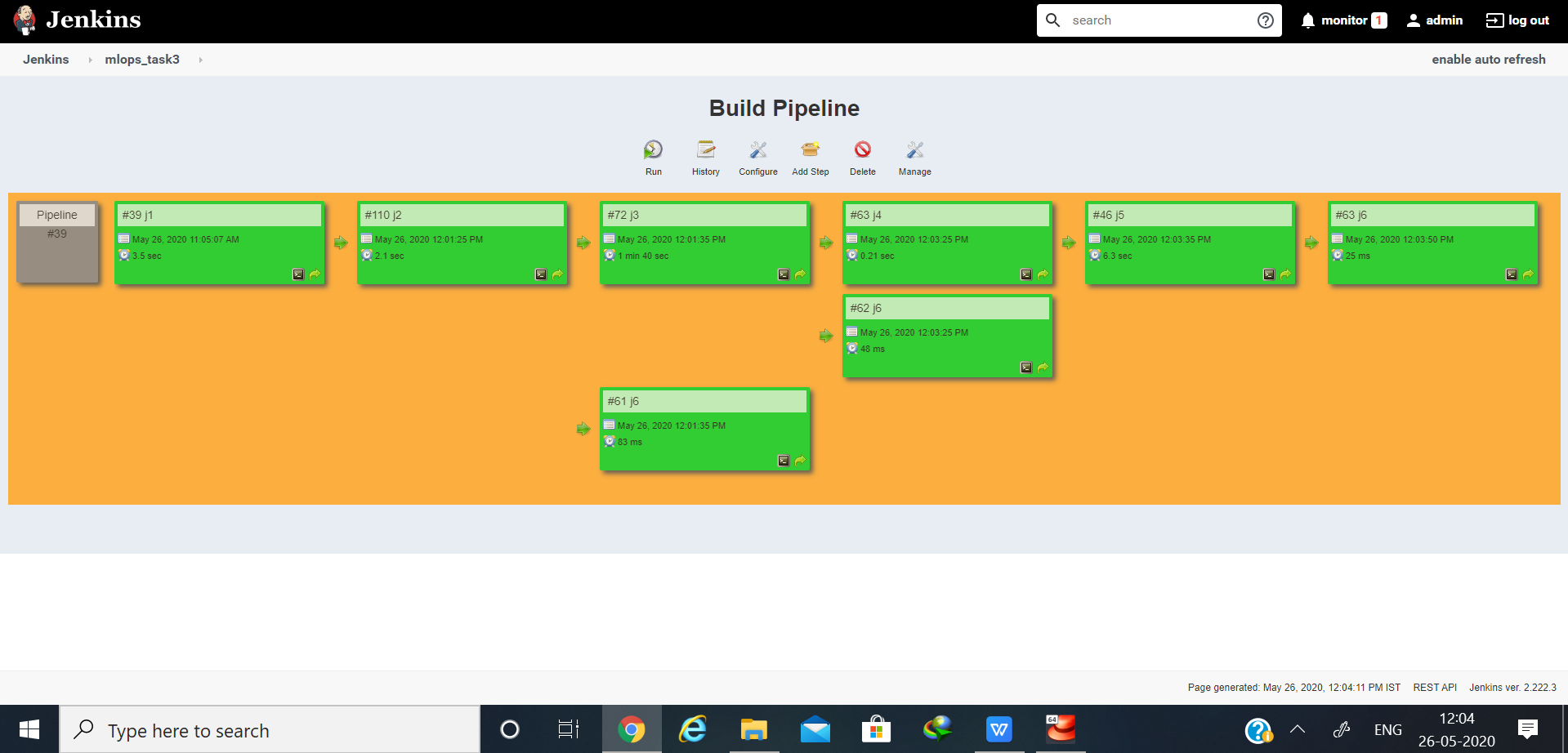
****Email confirmation****

********

****Job 6: This will keep on monitoring the system. If any jobs failed due to any reason, this will trigger the job2 to again build environment****

********

********

********

****Github URl:-**** <https://github.com/souvikdutta7012/mlops_task3>