**Name: Syeda Fatima Kazmi**

**Roll No: i21-1212**

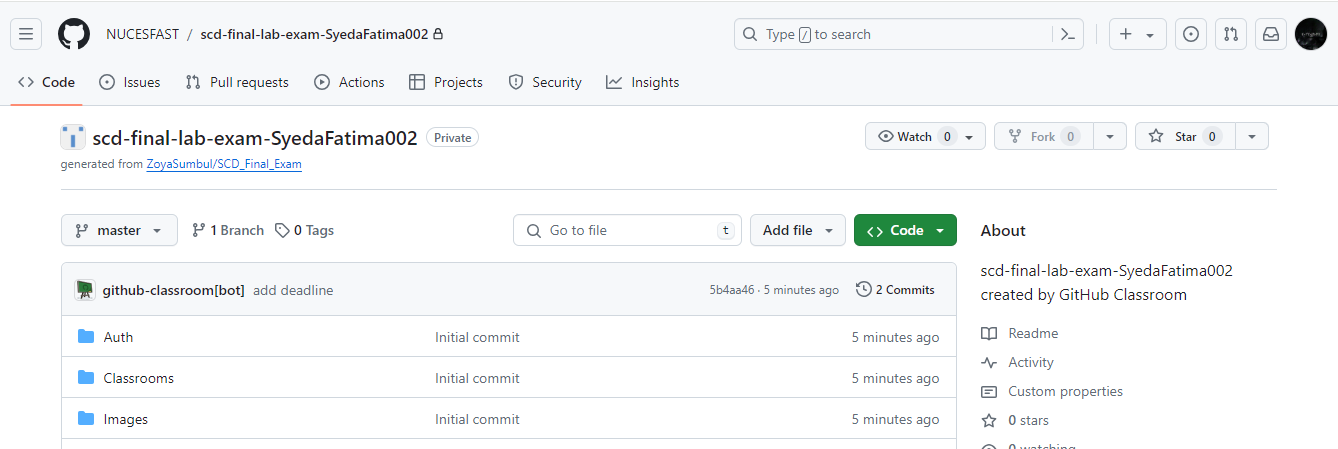
**SE - Q**

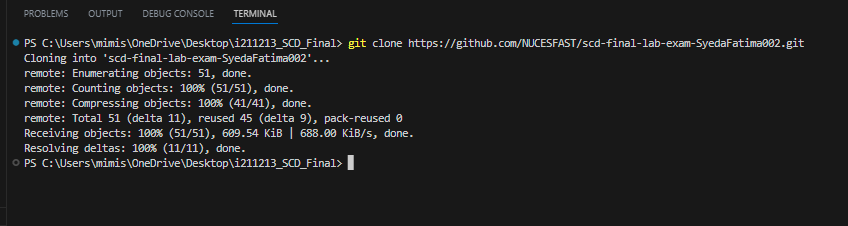
**SCD - Final Exam Report**

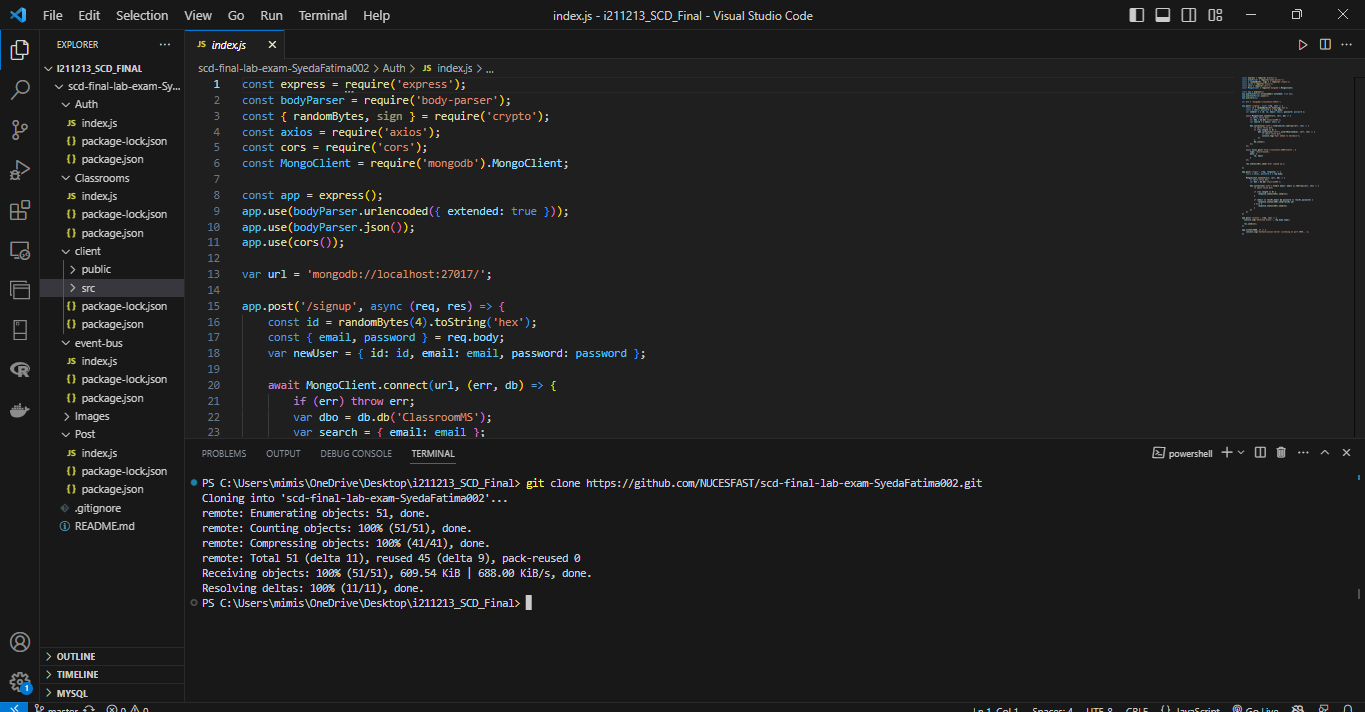
**Github link:** [**https://github.com/NUCESFAST/scd-final-lab-exam-SyedaFatima002.git**](https://github.com/NUCESFAST/scd-final-lab-exam-SyedaFatima002.git)

**—Jenkins Pipeline Creation—**

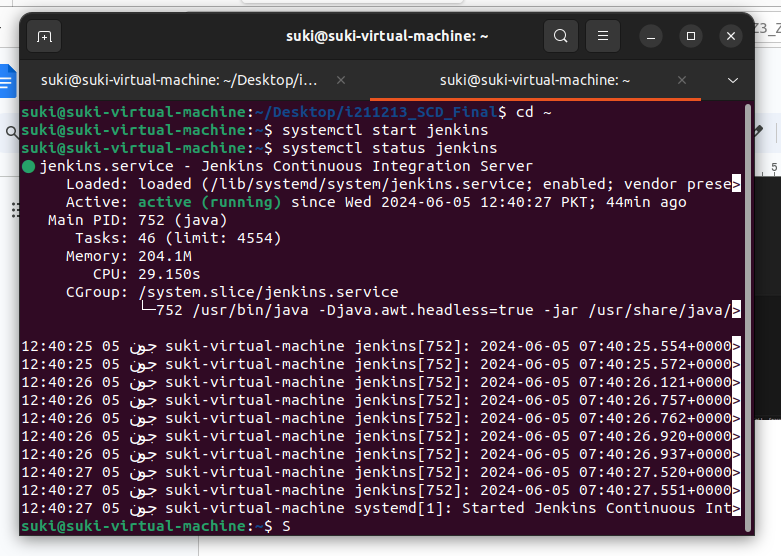
**1- Cloning github repository:**

****

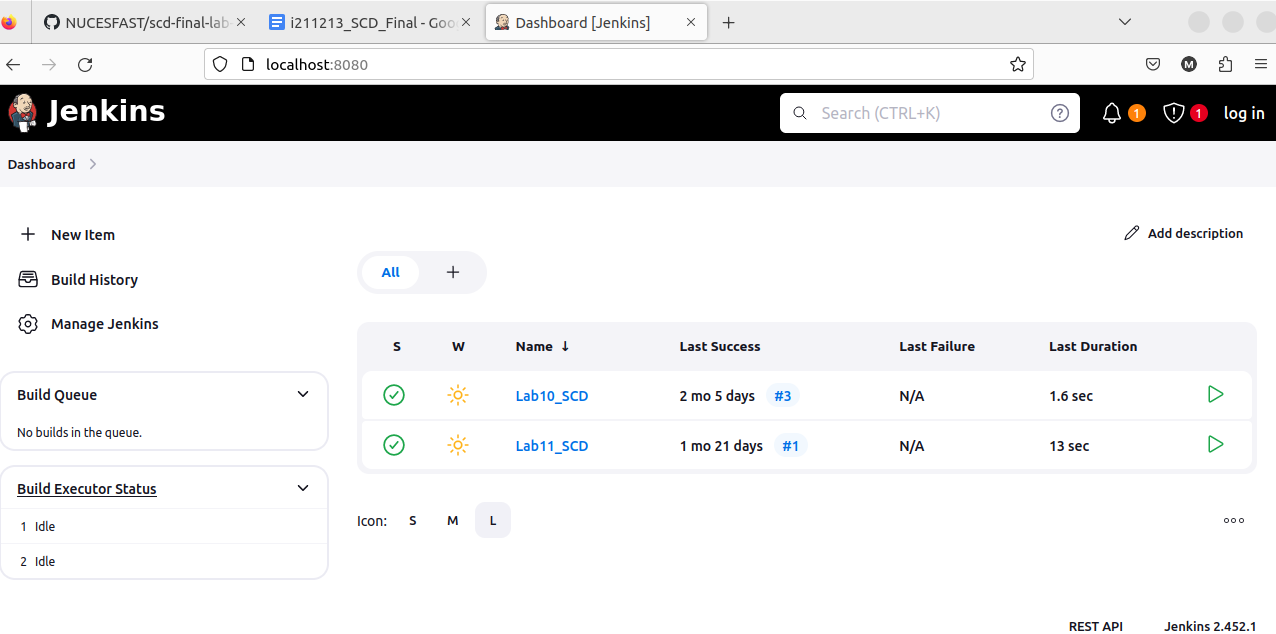
****

****

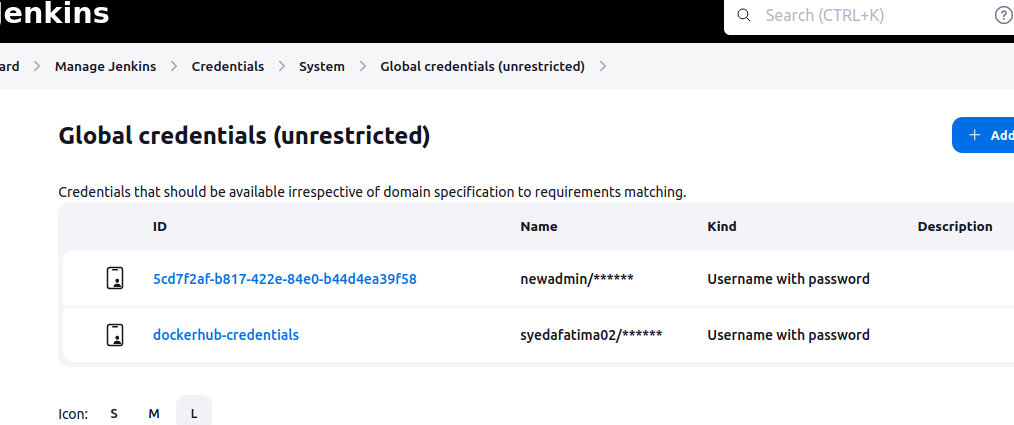
**2- Running jenkins service**

****

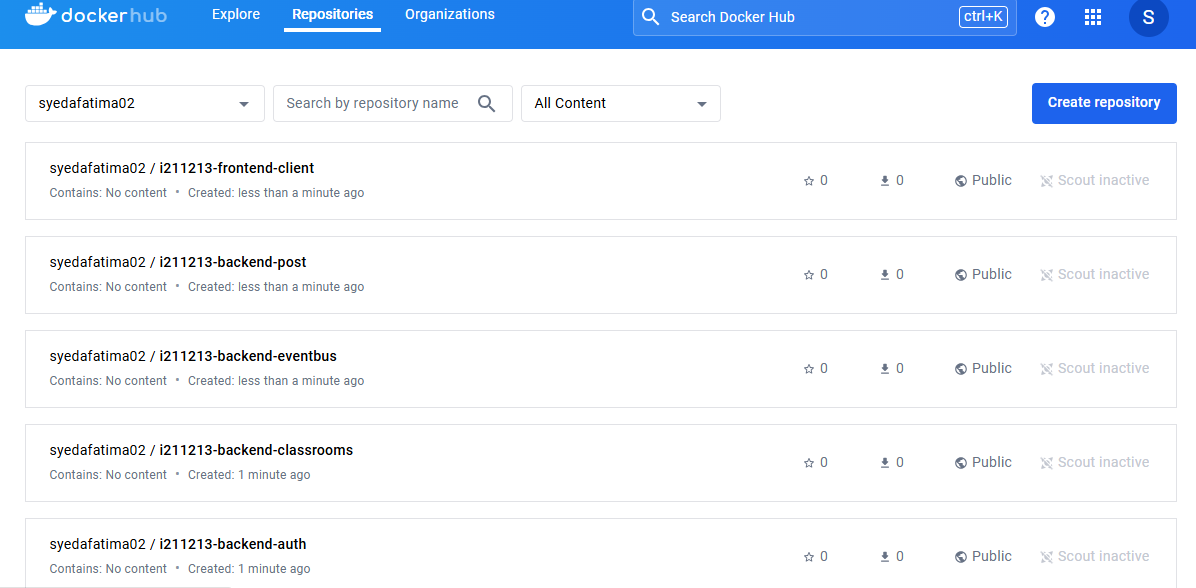
**3- Opening jenkins using localhost and creating credentials for the jenkins file**

****

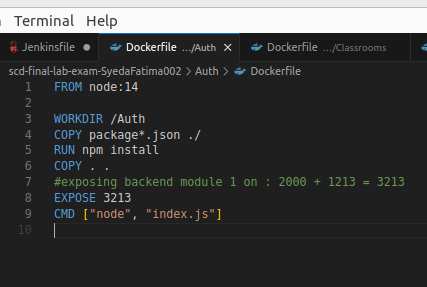
Created credentials

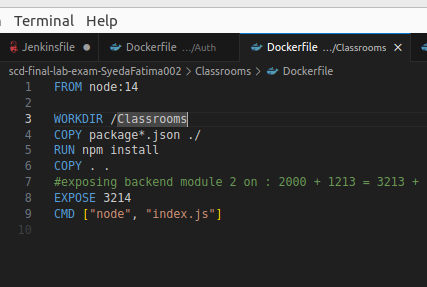
****

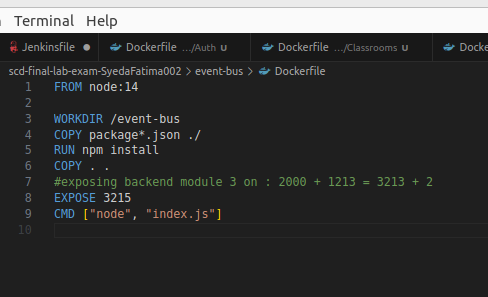
Creating docker repositories for containerization

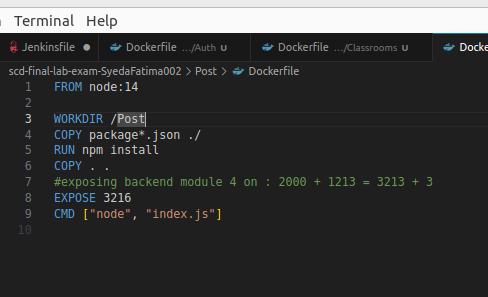


**3- Creating docker files for the backend modules (auth / classrooms / event-bus / post)**

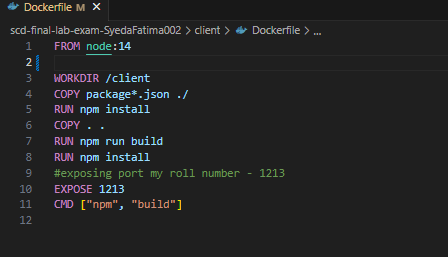




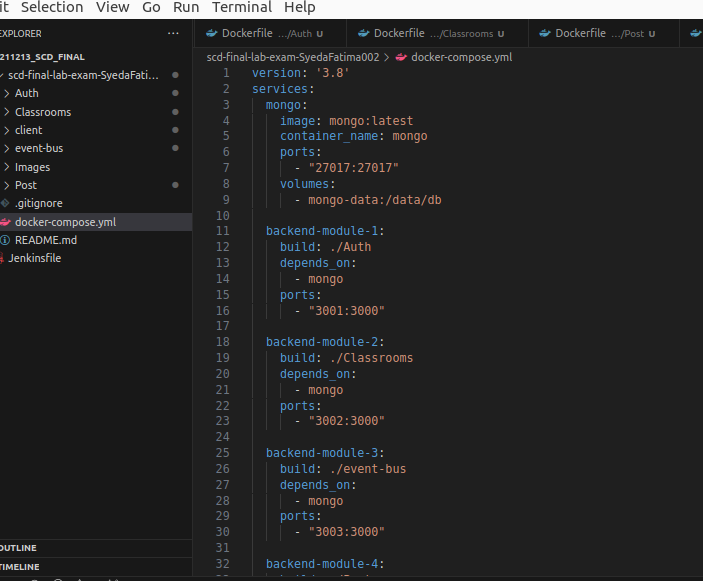


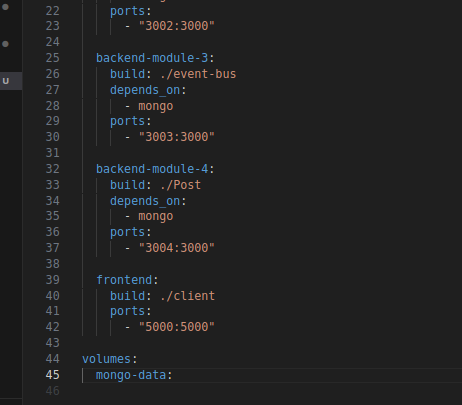


**4- Creating docker file for the frontend module (client)**

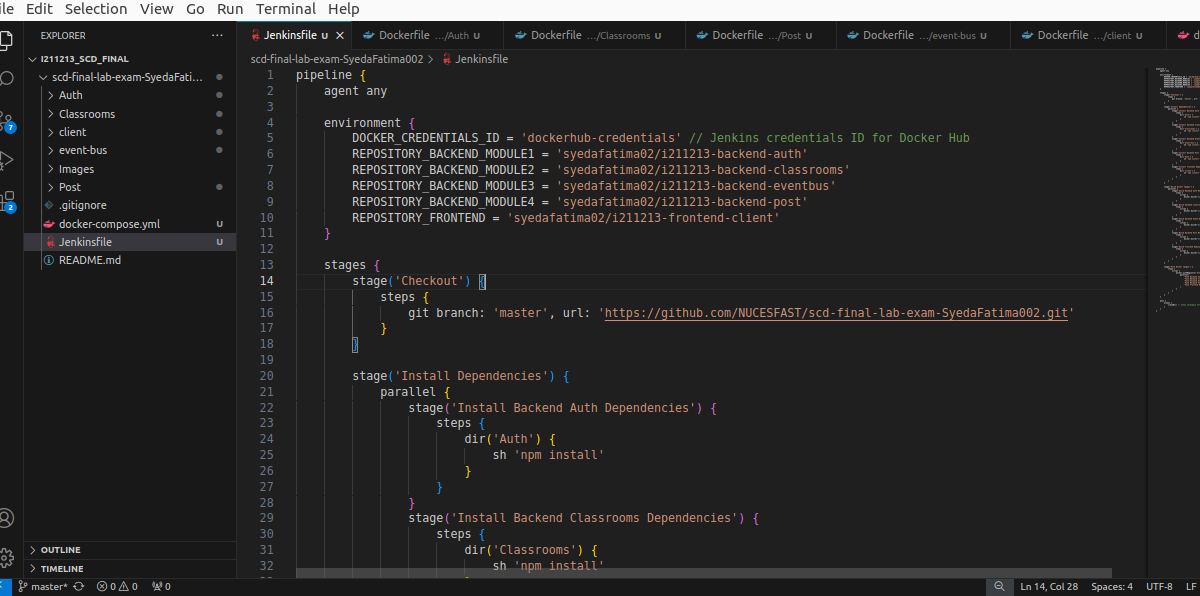


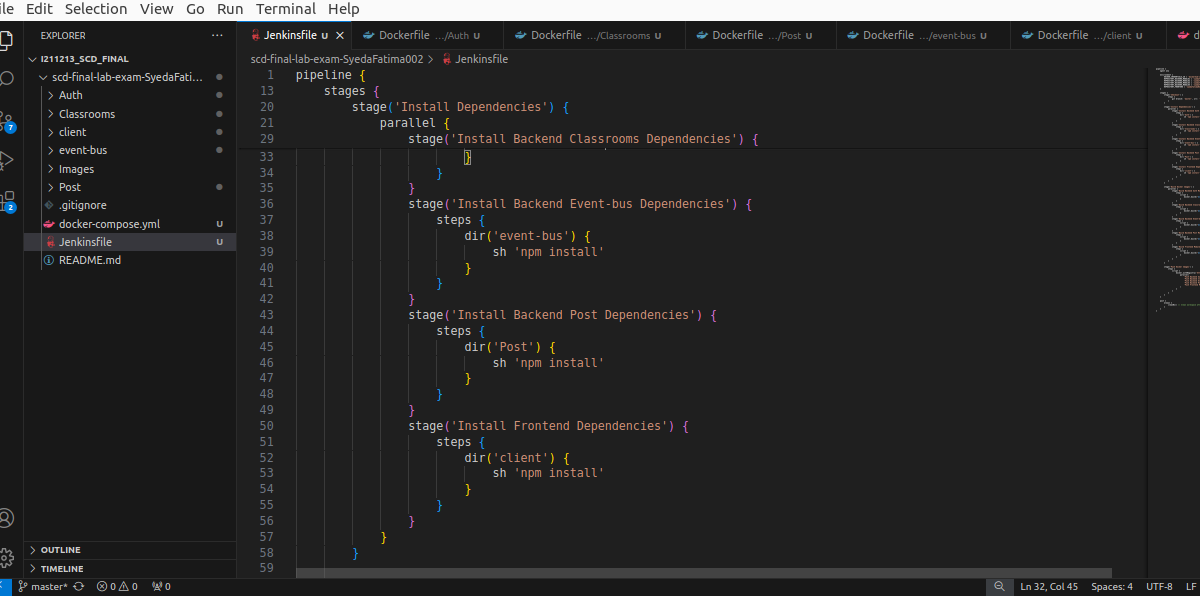
**5- Creating docker compose file for these docker files**

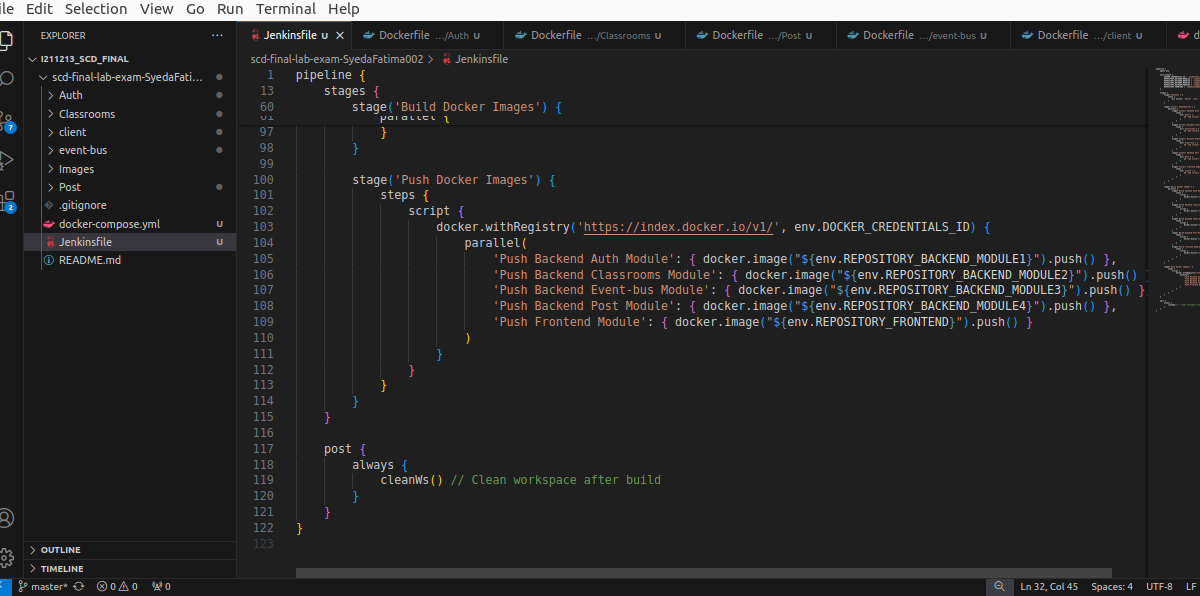
****

****

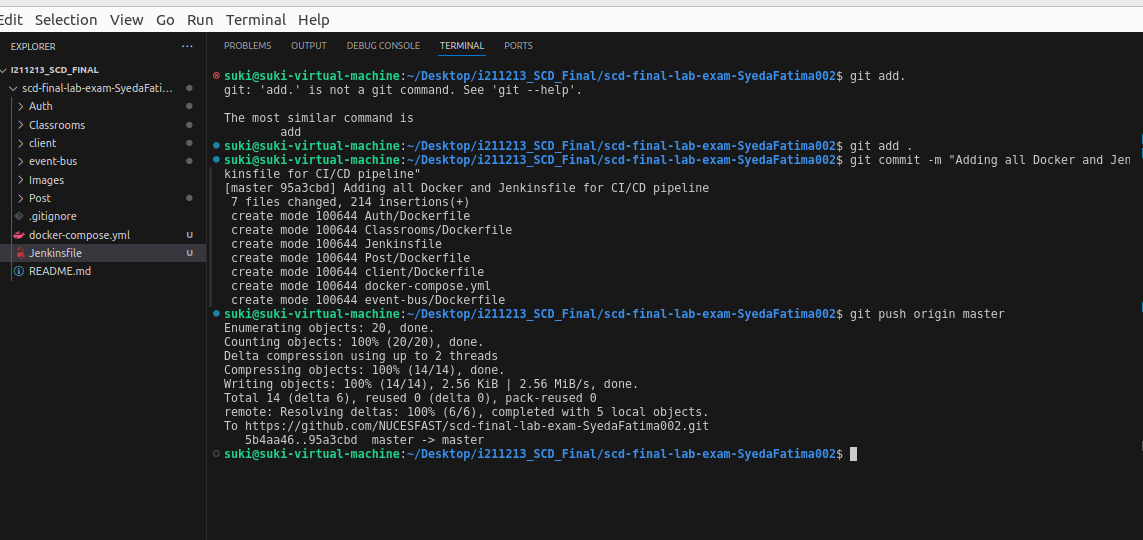
**6- Created Jenkins file**

****

****

****

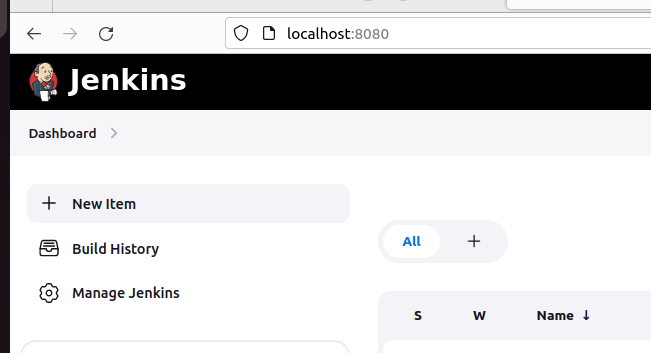
**7- Pushing changes to github**

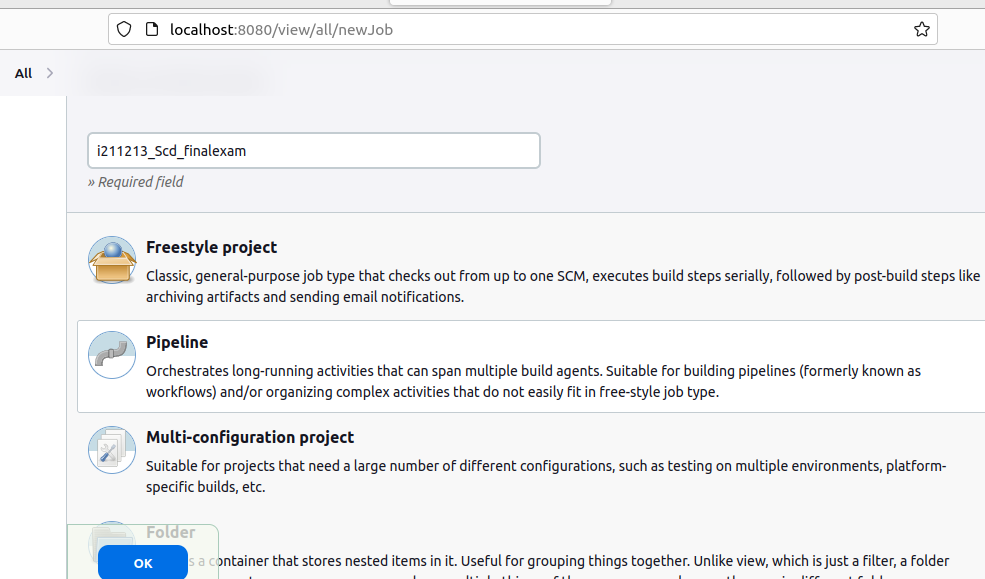
****

**8- Creating & Running Jenkins Pipeline**

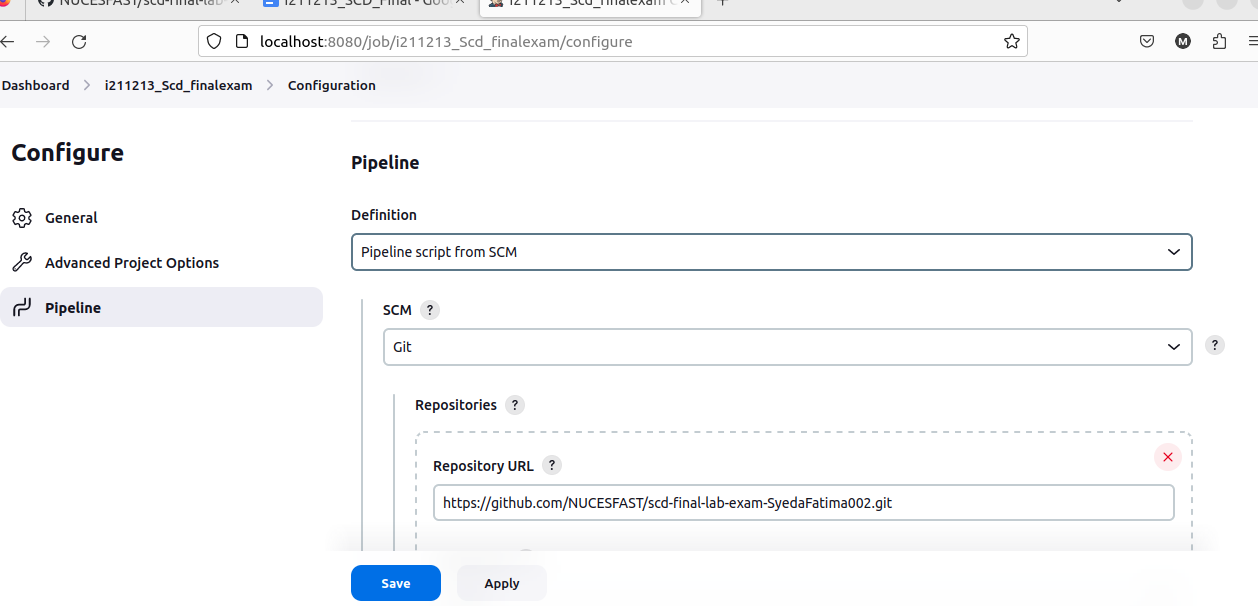
Jenkins was already started before (shared ss above)

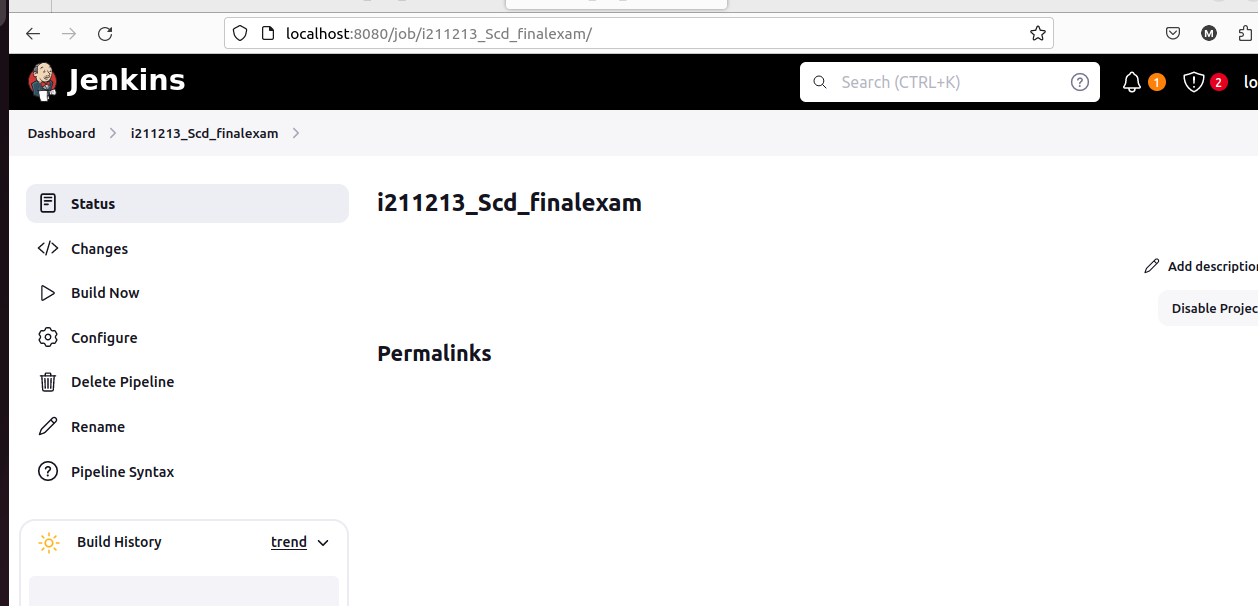
Clicking on new item to create a new pipeline



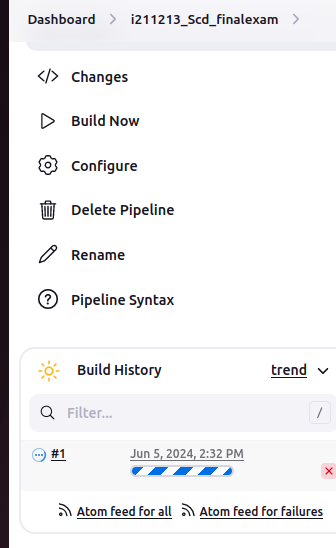


Configuring new pipeline and saving

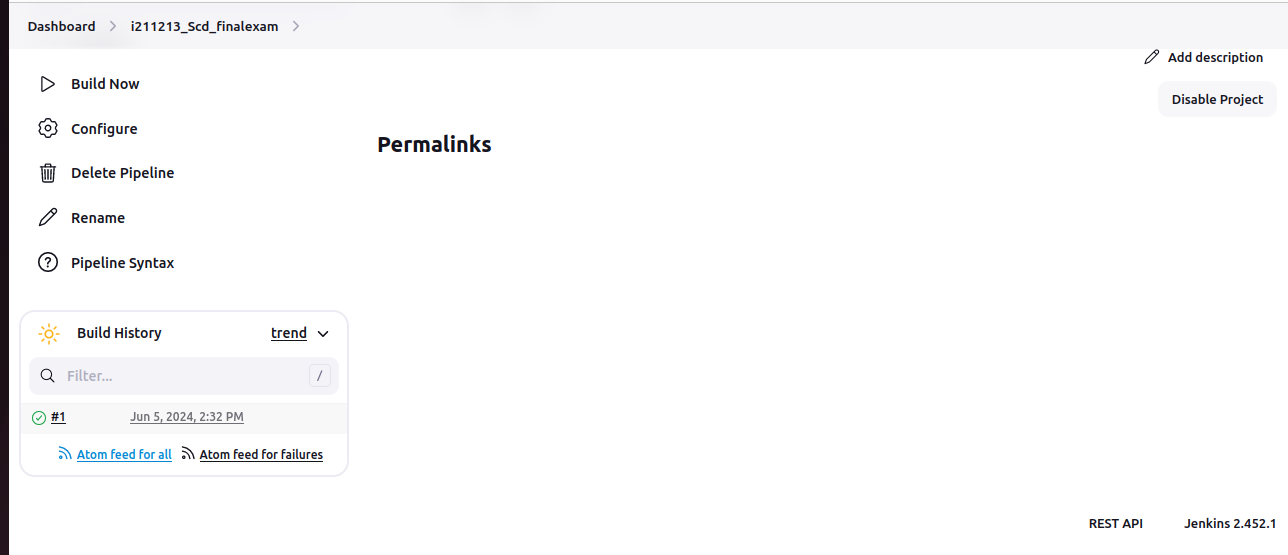




Running pipeline by clicking on “Build Now”



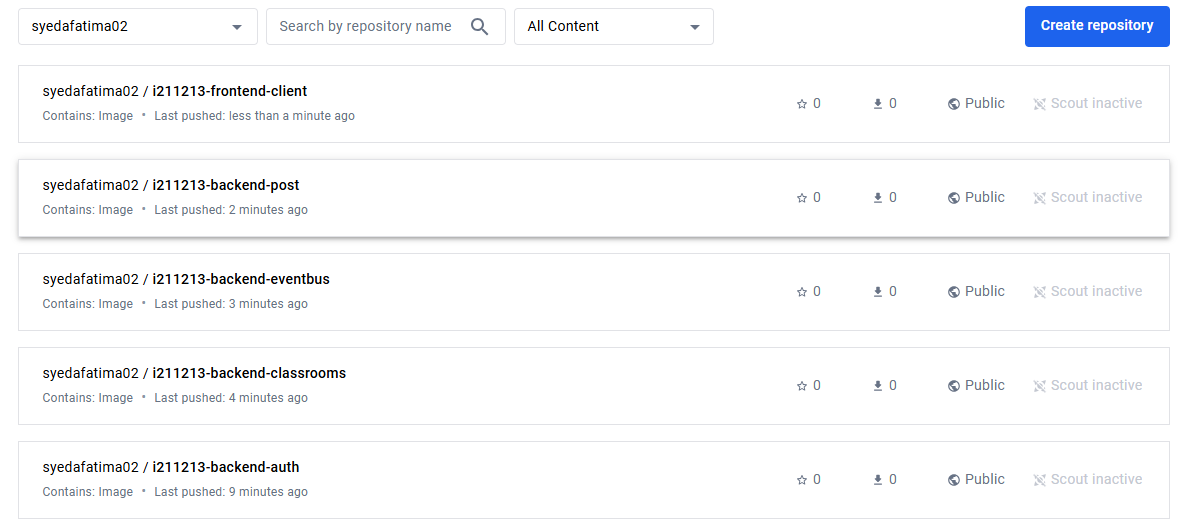
Below it shows the build was successful.



**(Note: I don’t know why it does not show the individual stage details like it should to me, it has been like this since the lab tasks too, i tried fixing it but it did not work out.)**

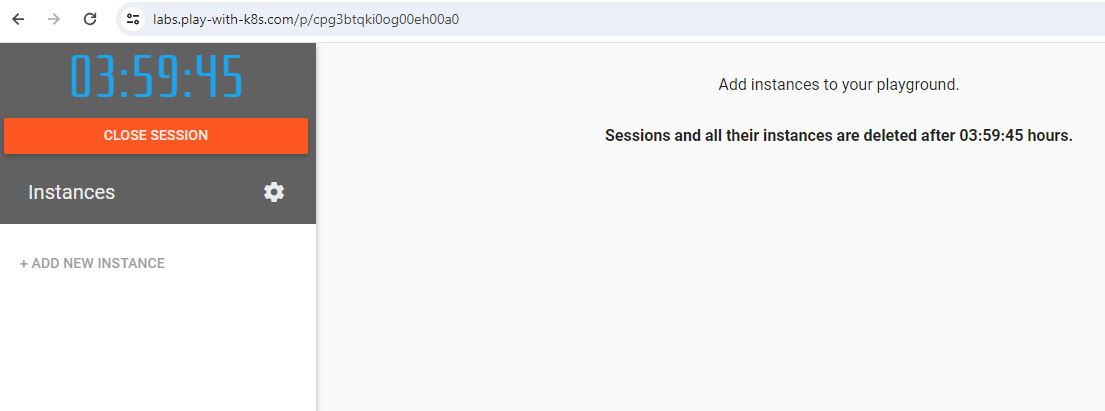
**9- Confirming Build**

I confirmed if the pipeline worked by opening the docker repository and seeing if the images got pushed and created using “docker images command”.



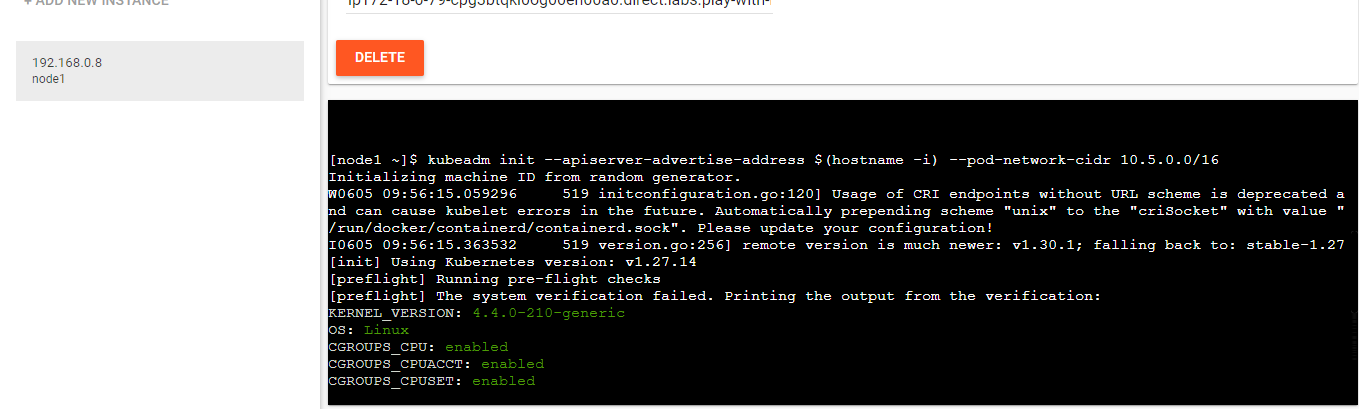
**—Kubernetes Deployment—**

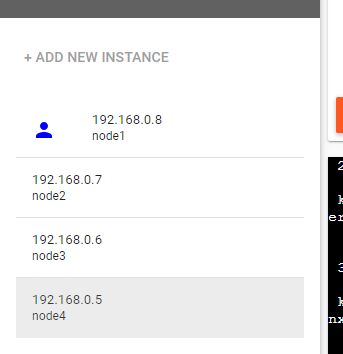
**1- Start Kubernetes (play with k8s)**

****

**2- Created 4 instances, initializing the first node as the master node**

Node1: initialized as master node

****

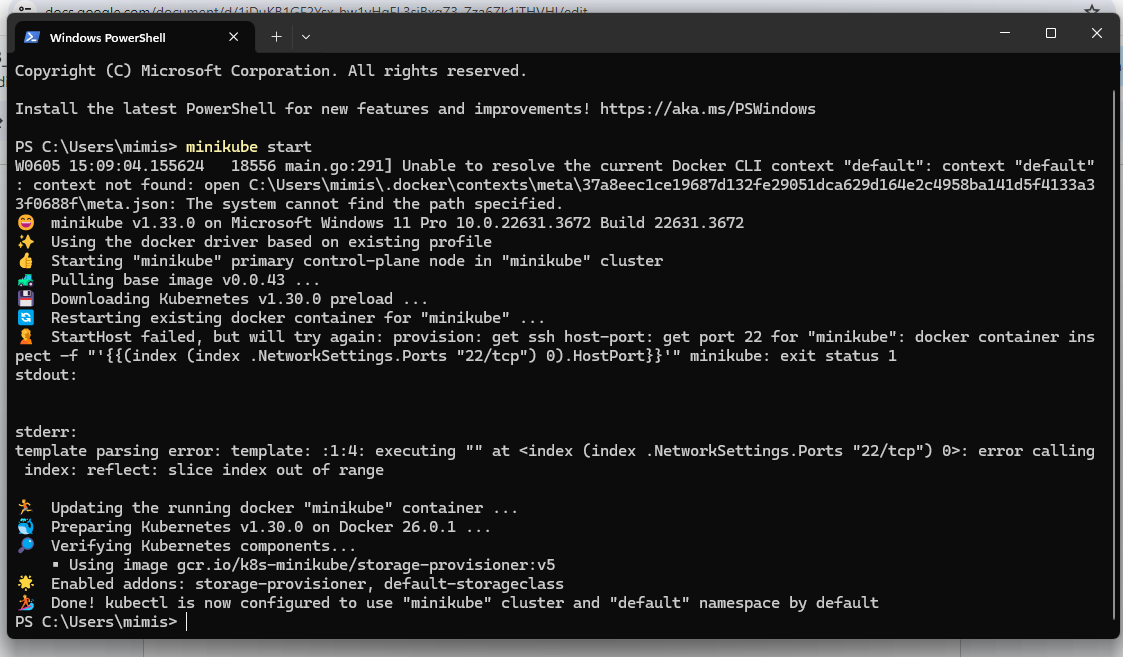
****

Making node 2,3 and 4 worker nodes. Joining them with the cluster node:

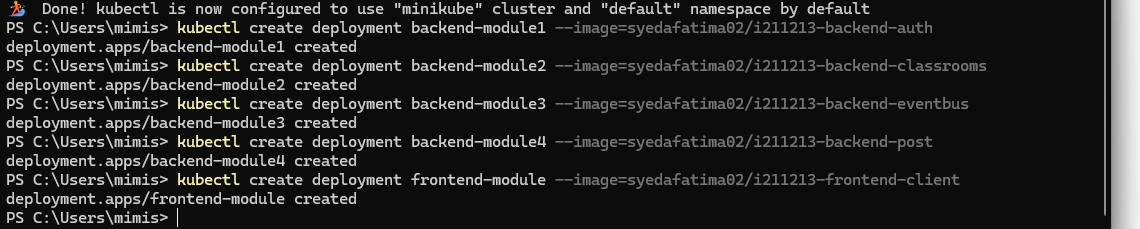
(Because the internet is so slow, my playwithk8s was not working, so i used MiniKube)

**—MINIKUBE—**

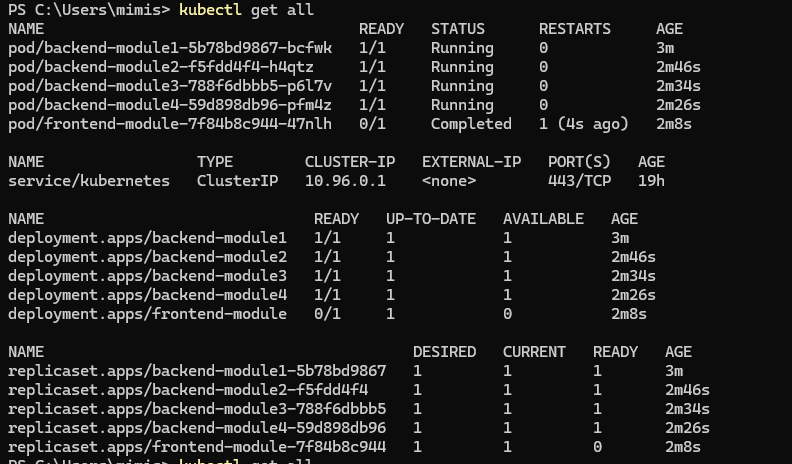
**1- Start cluster**

****

**2- Creating 5 deployments separately for all the 5 modules (backend and frontend)**

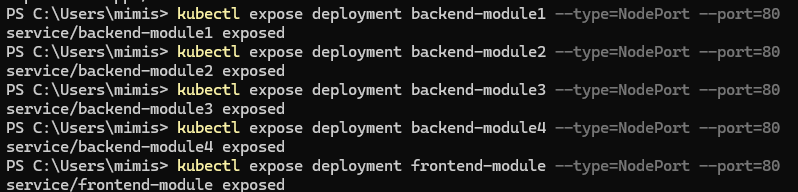
****

**3- Checking if the deployments are created correctly**

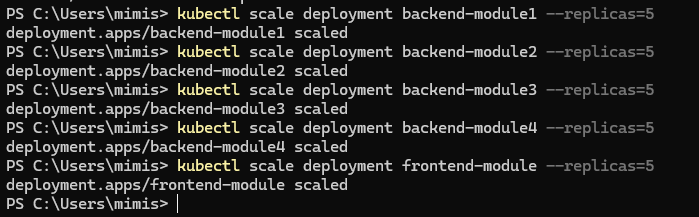
****

And here we can see there is one kubernetes service, 5 deployments we created, the workload is distributed between 5 pods created for the deployments, and 5 replicas for each deployment.

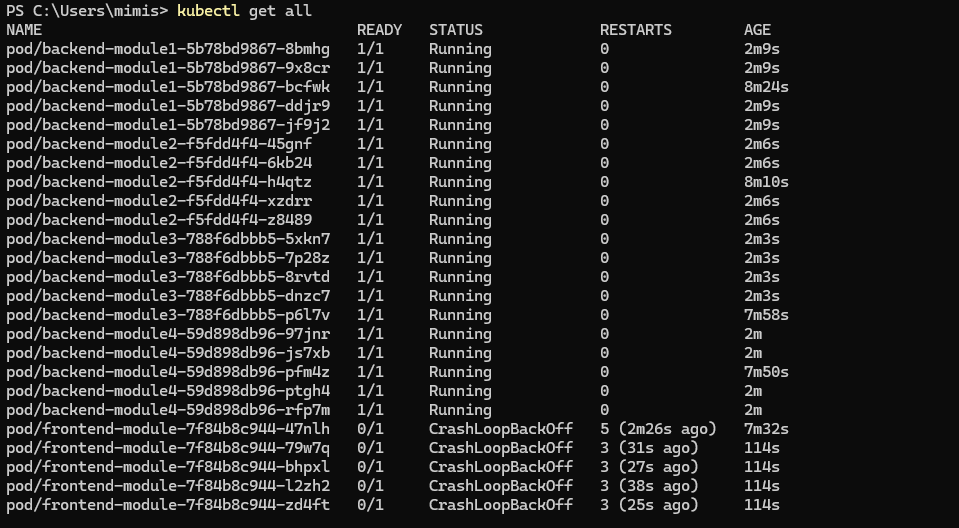
**4- Exposing the deployments to that they are accessible by the other modules**

****

**5- Now scaling the deployments horizontally by creating 5 replica sets for each**

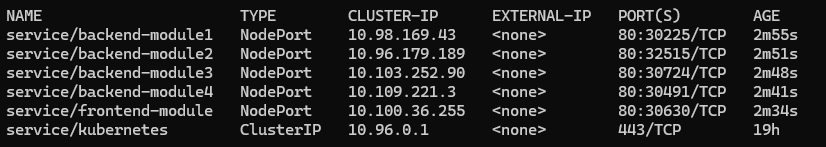
****

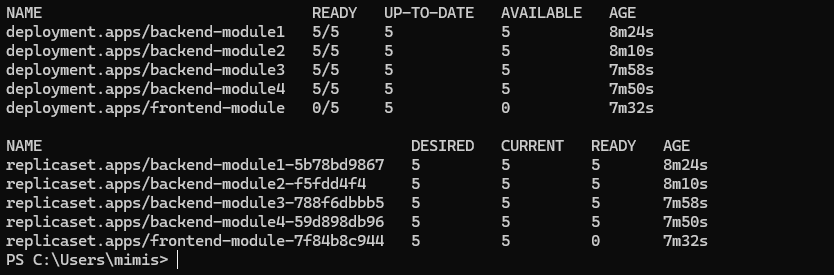
**6- Checking deployments and services etc**

****

Here we can see that the frontend deployment crashed.. (i do not know why) but the rest of the images are running successfully.

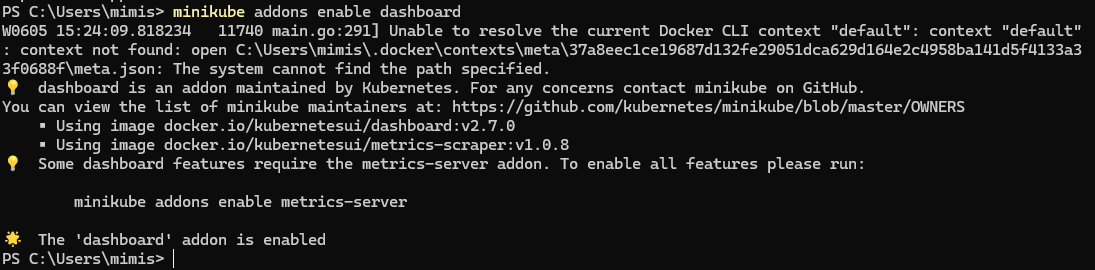
And below you can see the services, deployments and the replica sets successfully deployed.



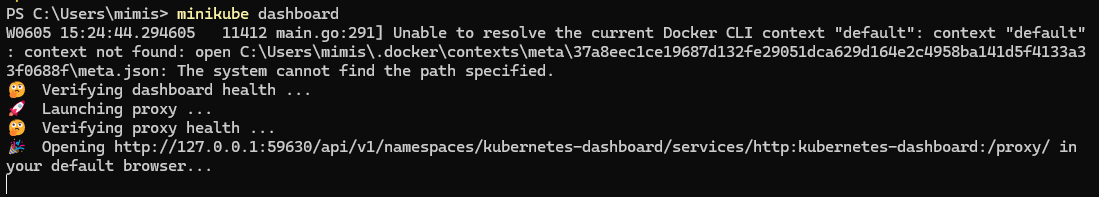


**7- Opening kubernetes dashboard**

1- Enabling the dashboard on minikube

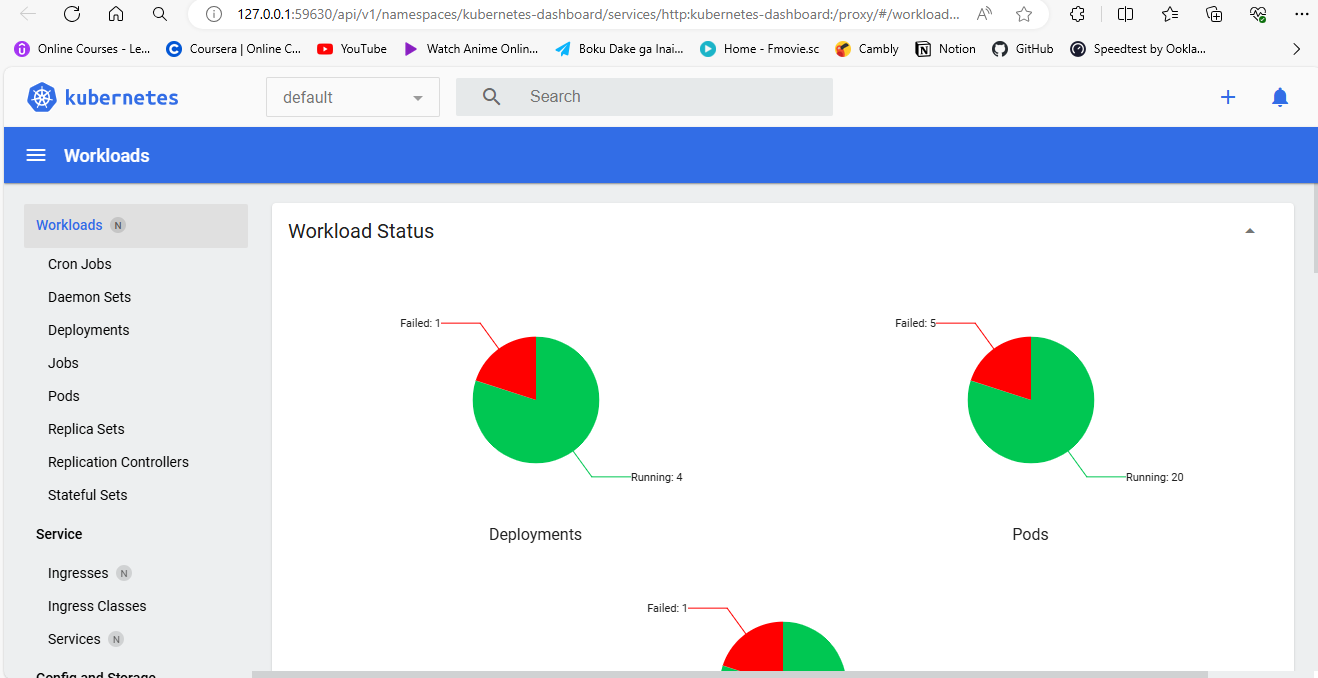


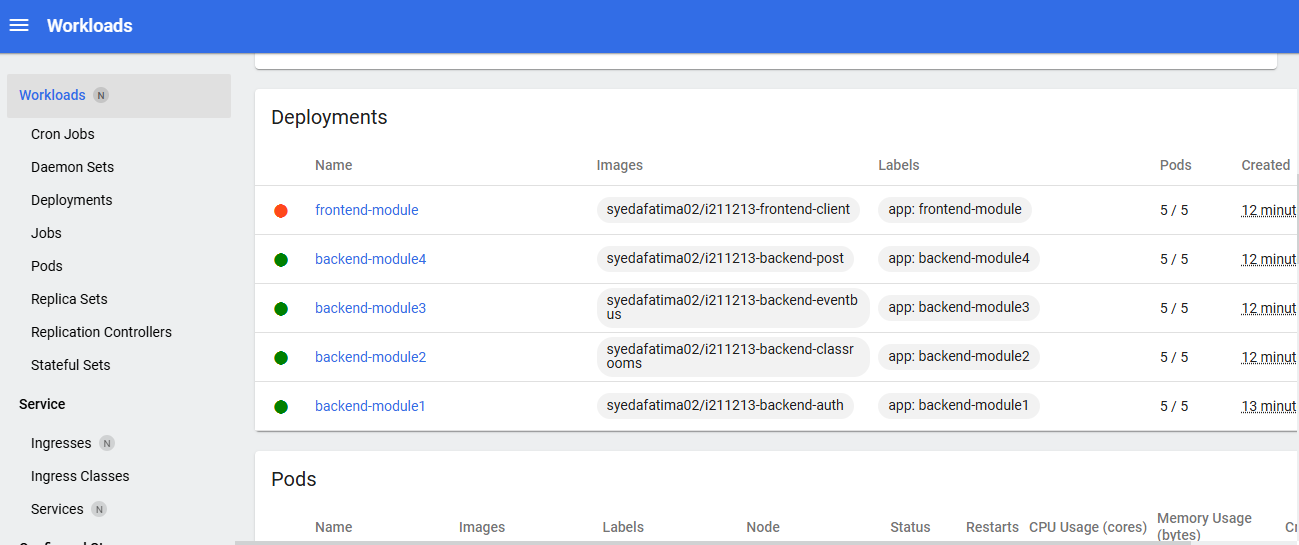
2- Opening dashboard



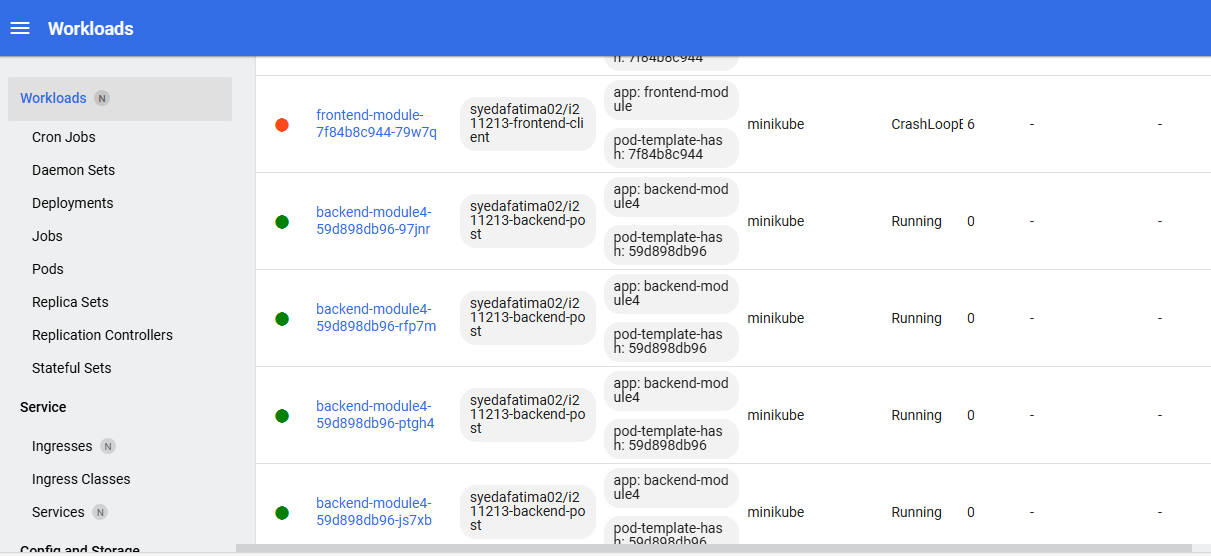
The dashboard opened on the browser (images below), it can be used to monitor the pods and deployments easily.

(next page)

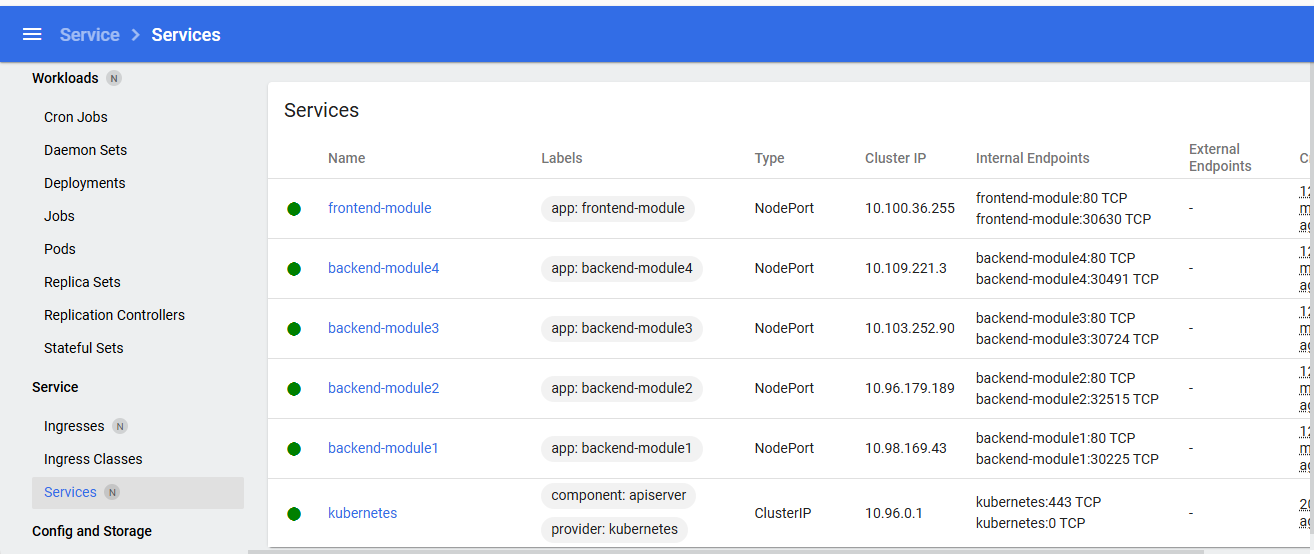




Pods: (the running pods are shown in green and failed in orange)

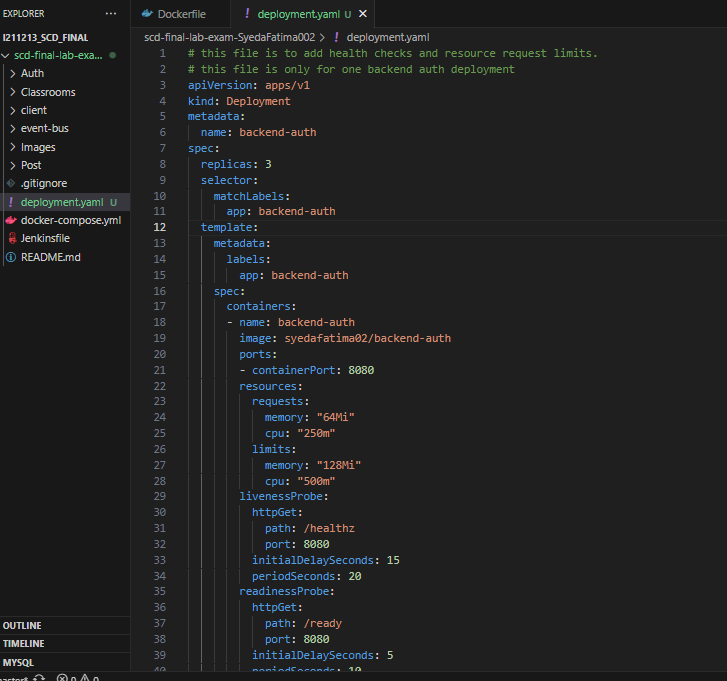


Services:

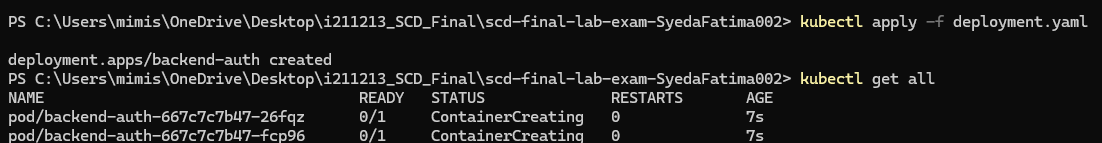


**—Additional Functionalities—**

**1- Creating yaml file for one of the deployments:**

****

**2- Applying the following file with the health checks and resource limits**

****

Here we can see the yaml being used, and backend-auth deployment being created with the additional functionalities.

