CAPSTONE PROJECT  
Cloud Devops Final Project – Docker and Kubernates Deployment

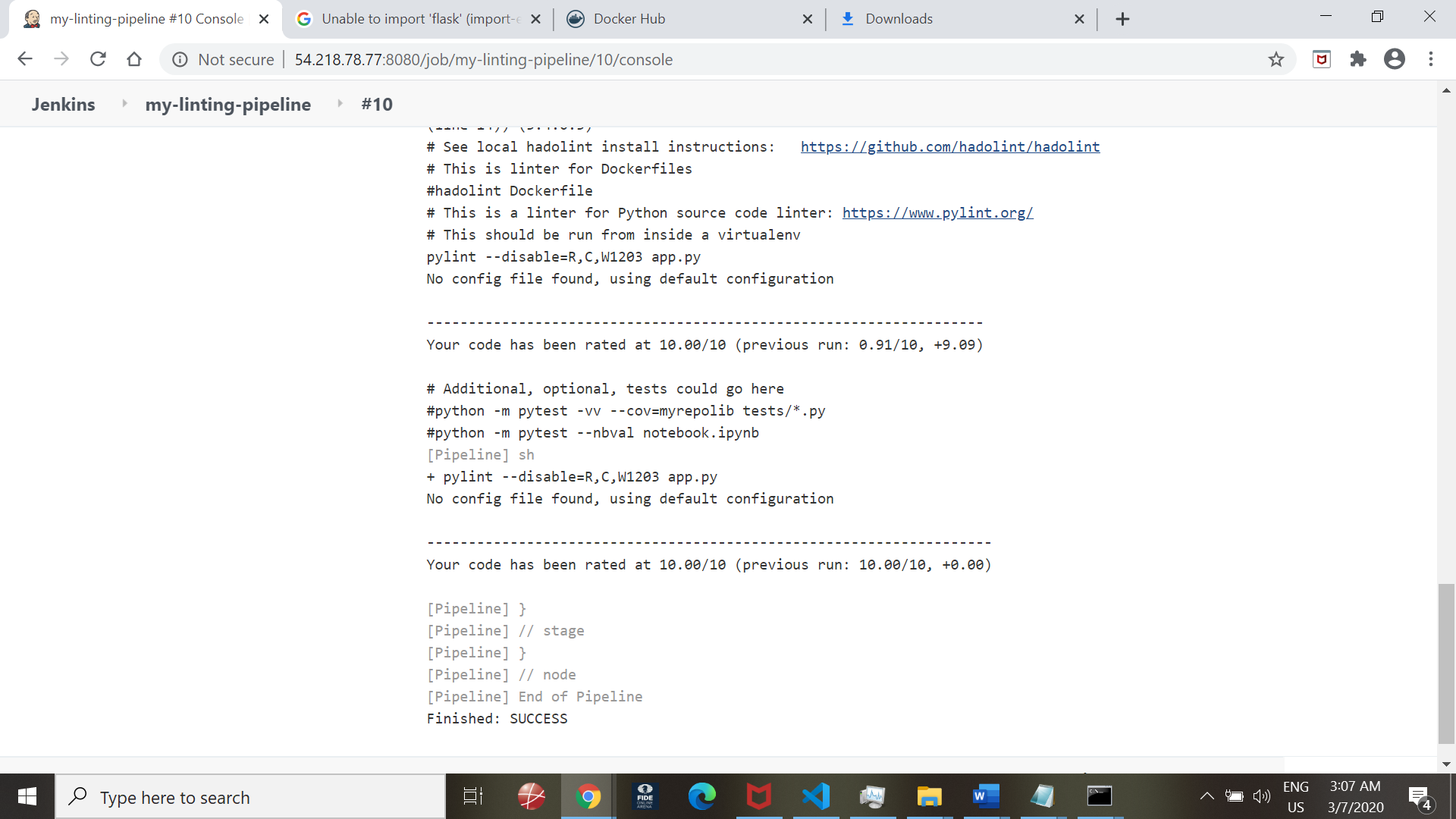
The document has three sections:

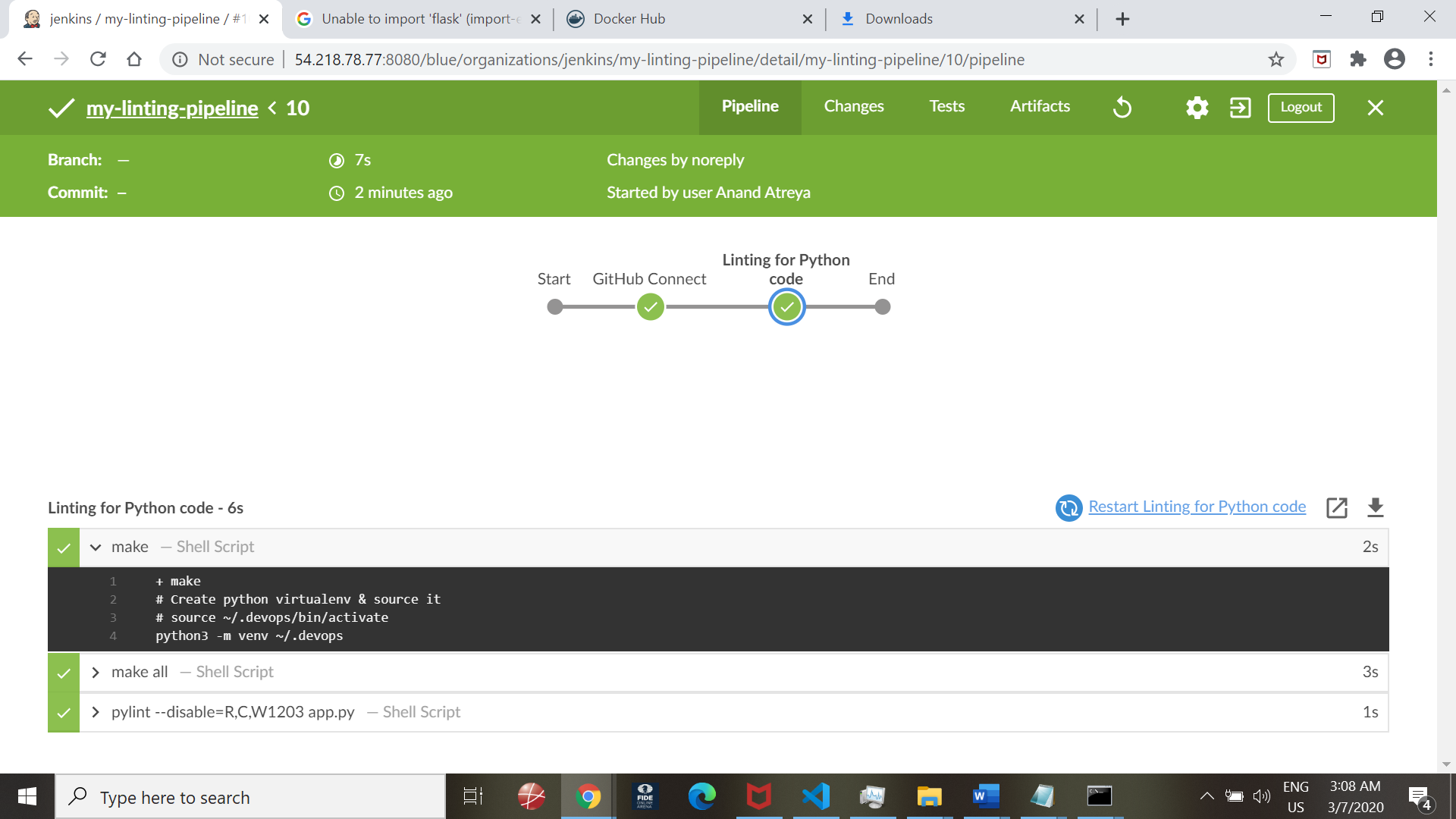
Section 1 describes the Linting approach taken to validate the code including capturing success and failure scenarios.

Section 2 details the Docker Build, Docker push to Docker Hub & Docker run on remote host implemented via the Jenkins pipeline feature including testing of the build.

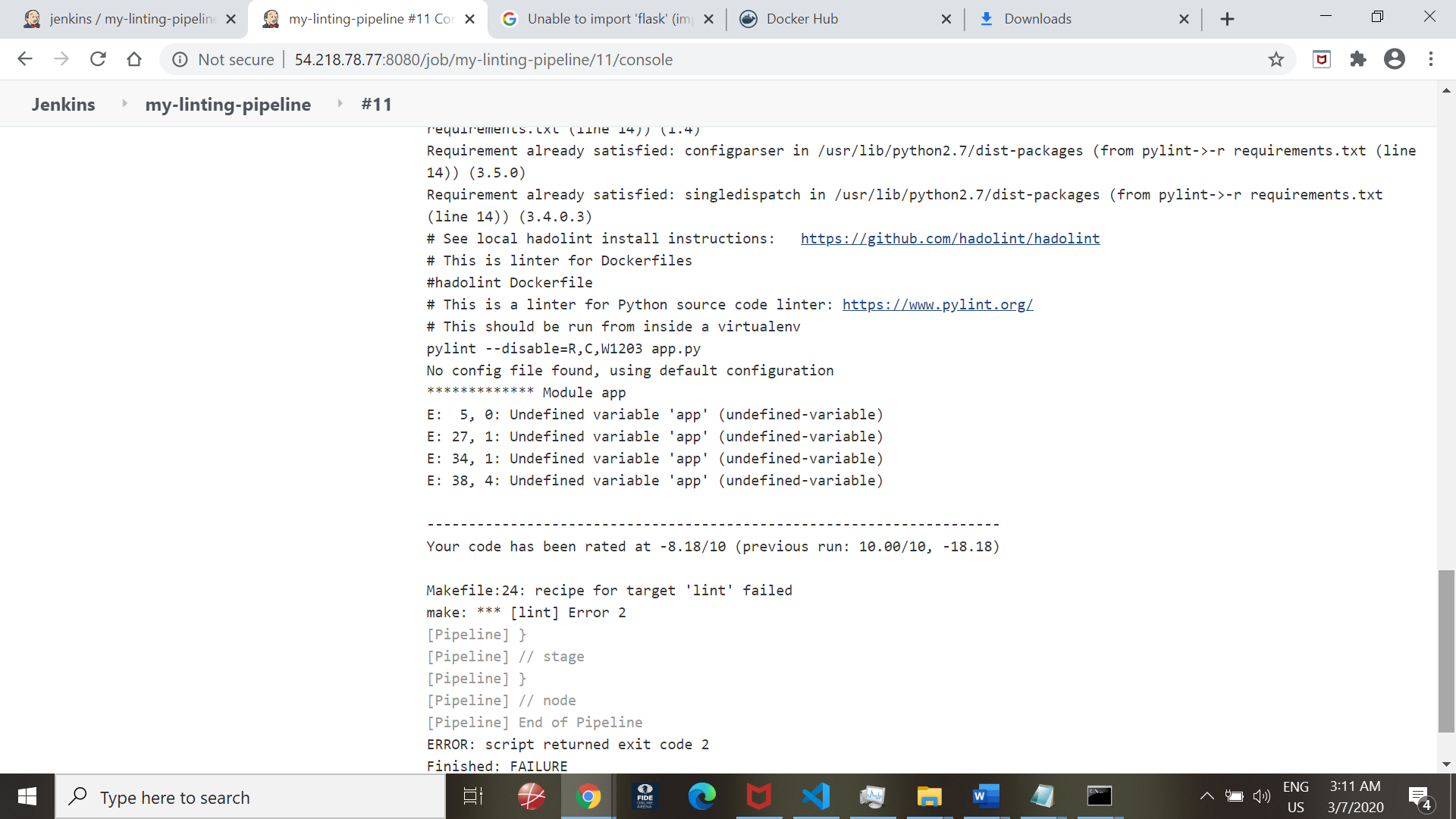
Section 3 explains the Docker Container Deployment in the Kubernetes cluster

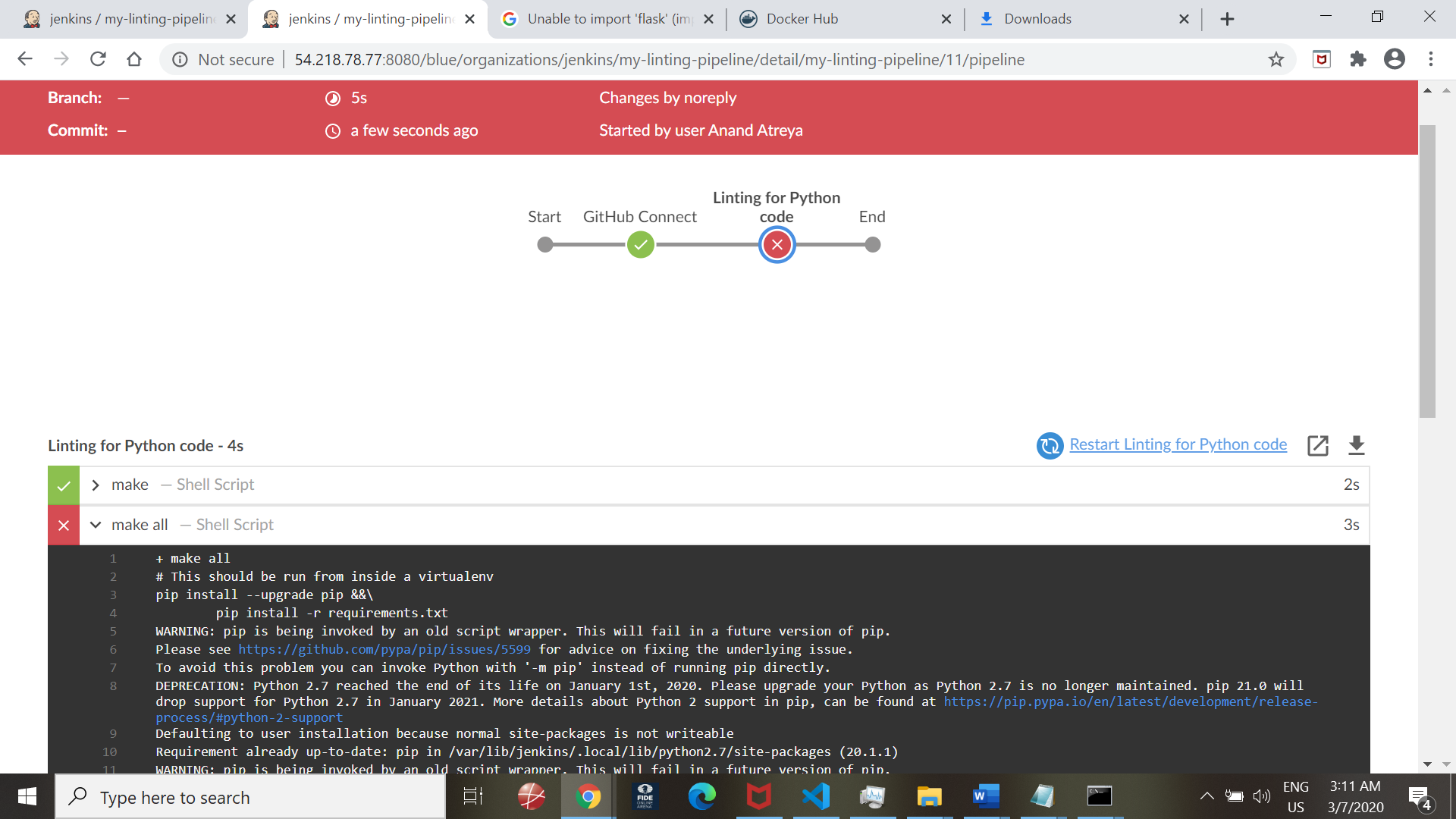
1. **Linting Test:**



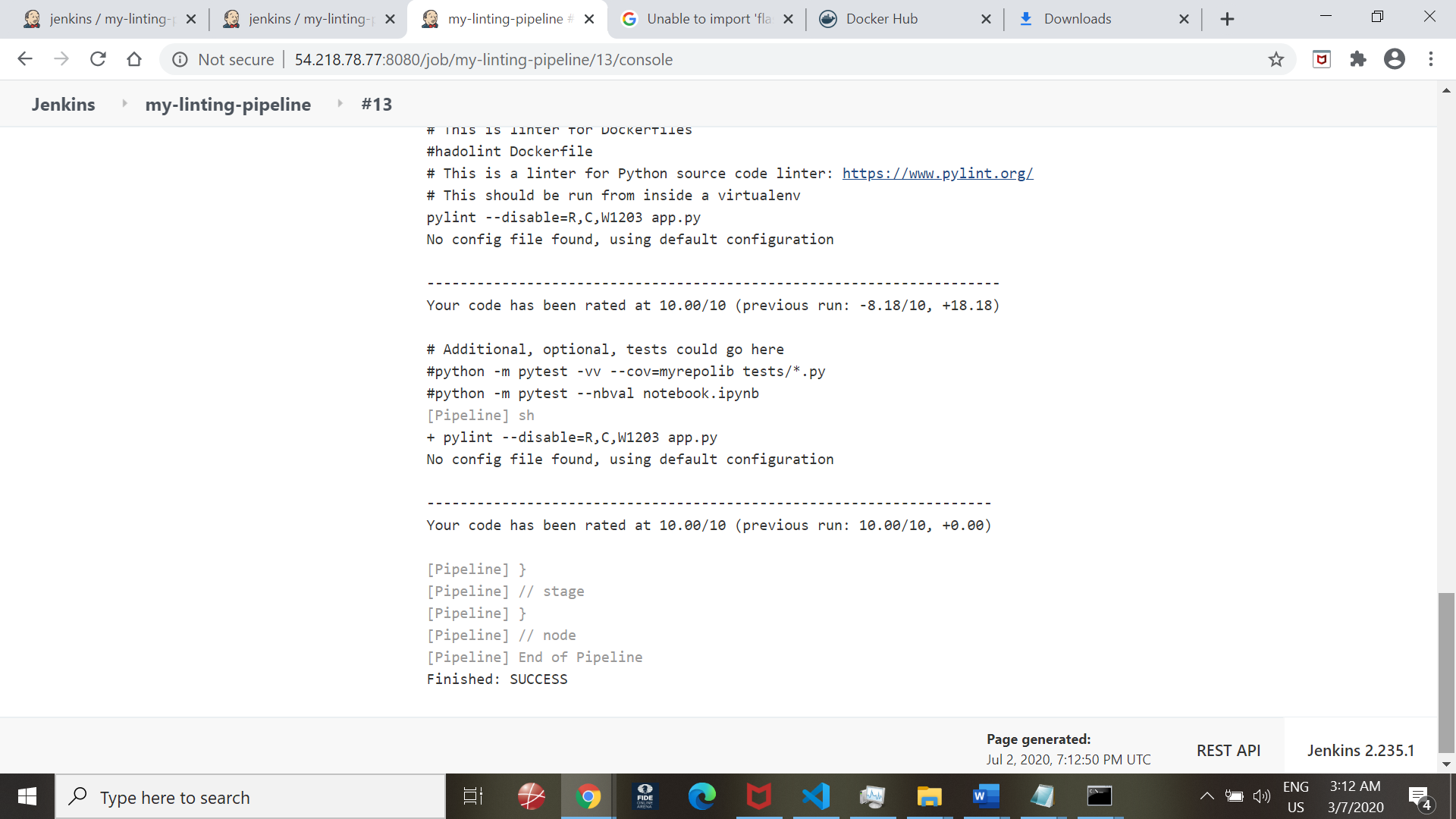


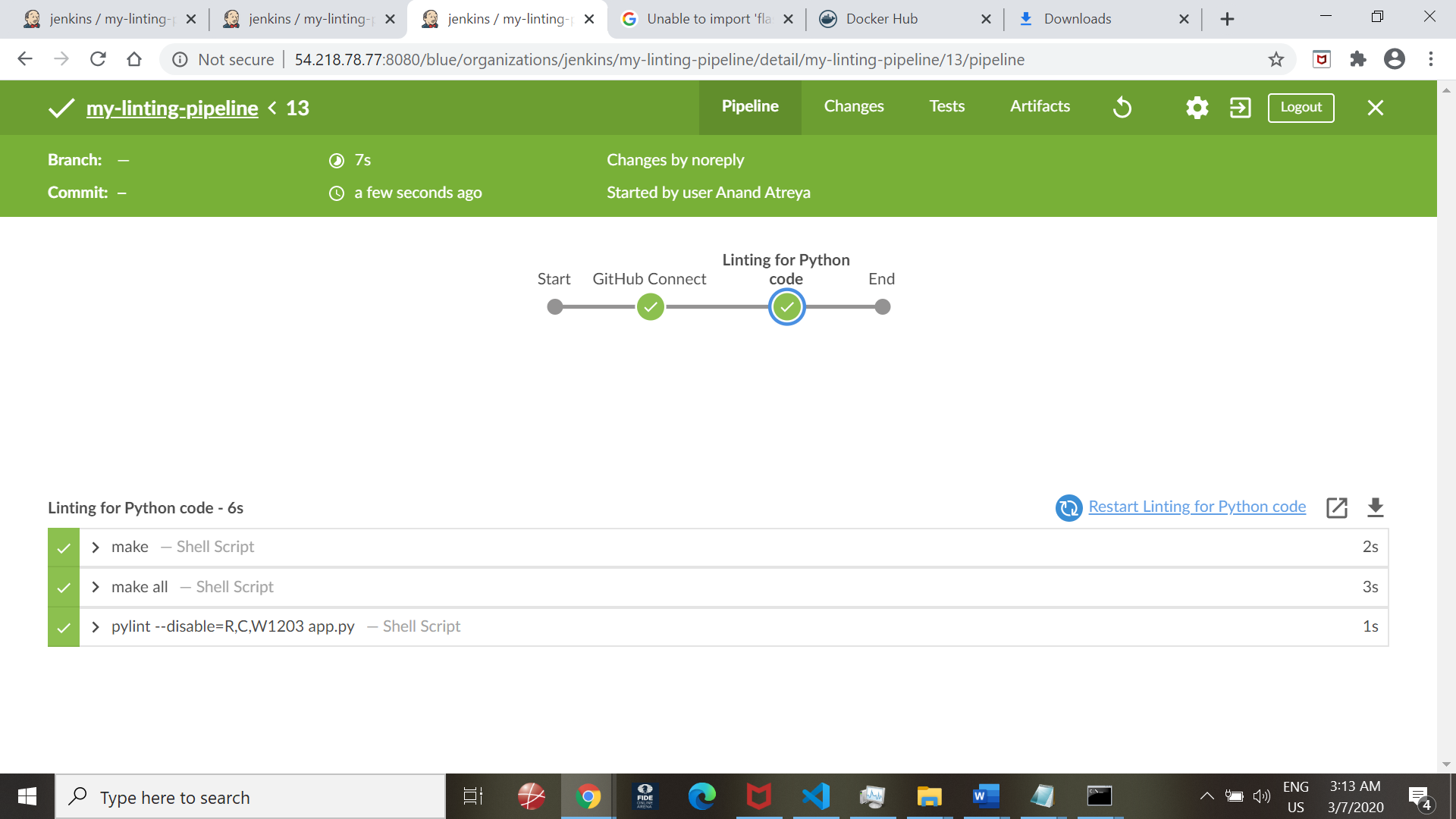
Linting failure case:





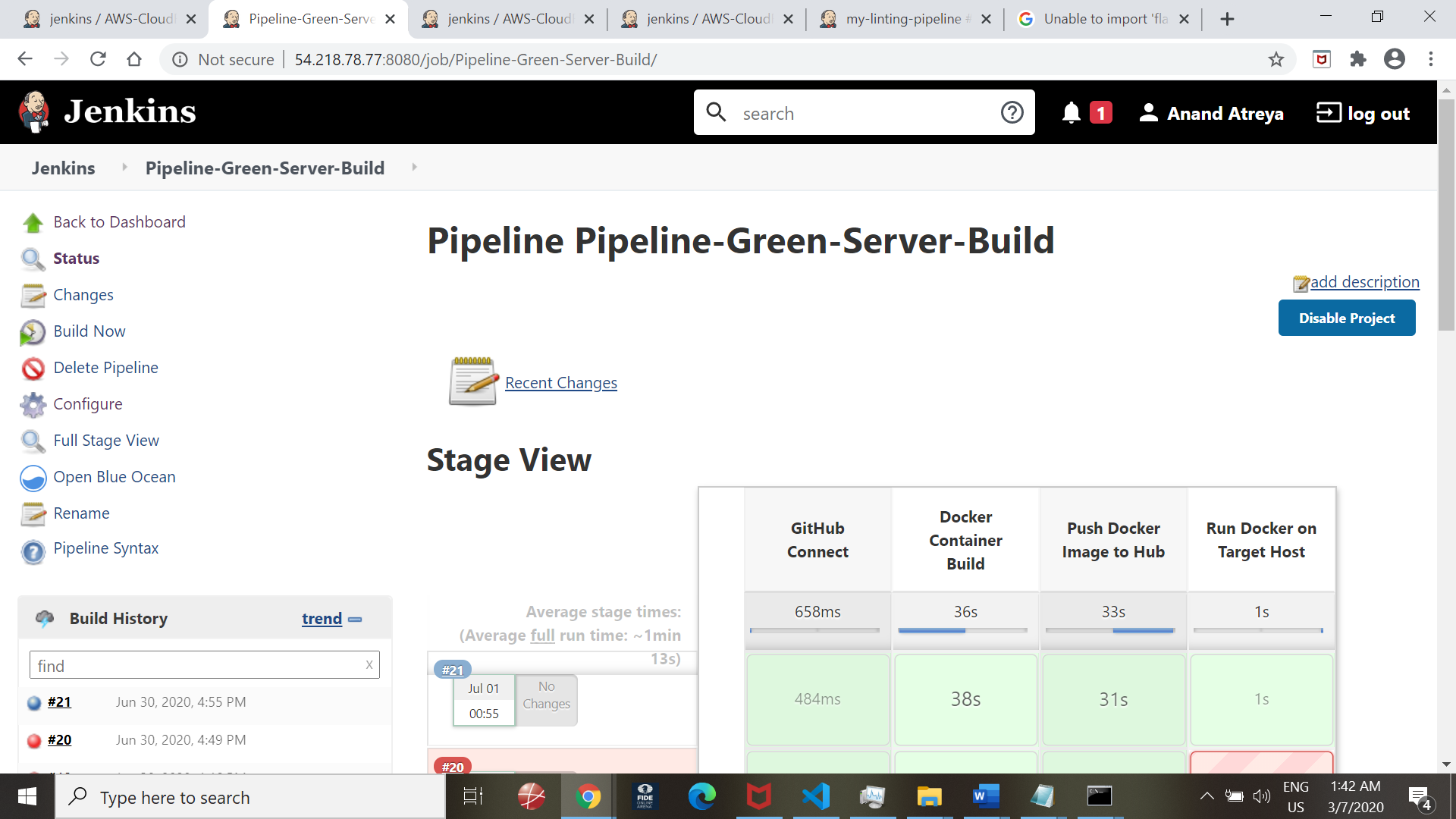
Linting success after code error was resolved



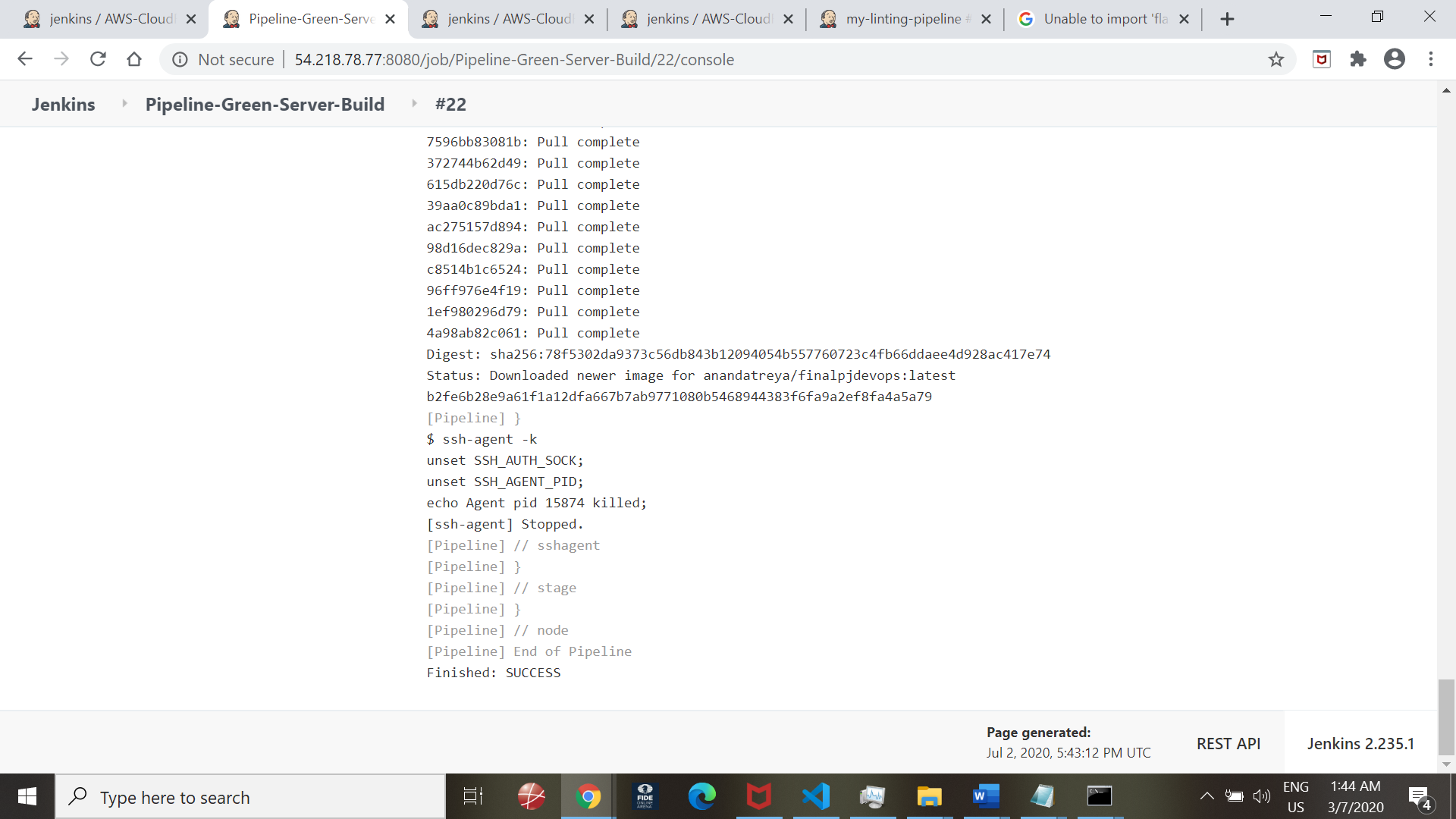


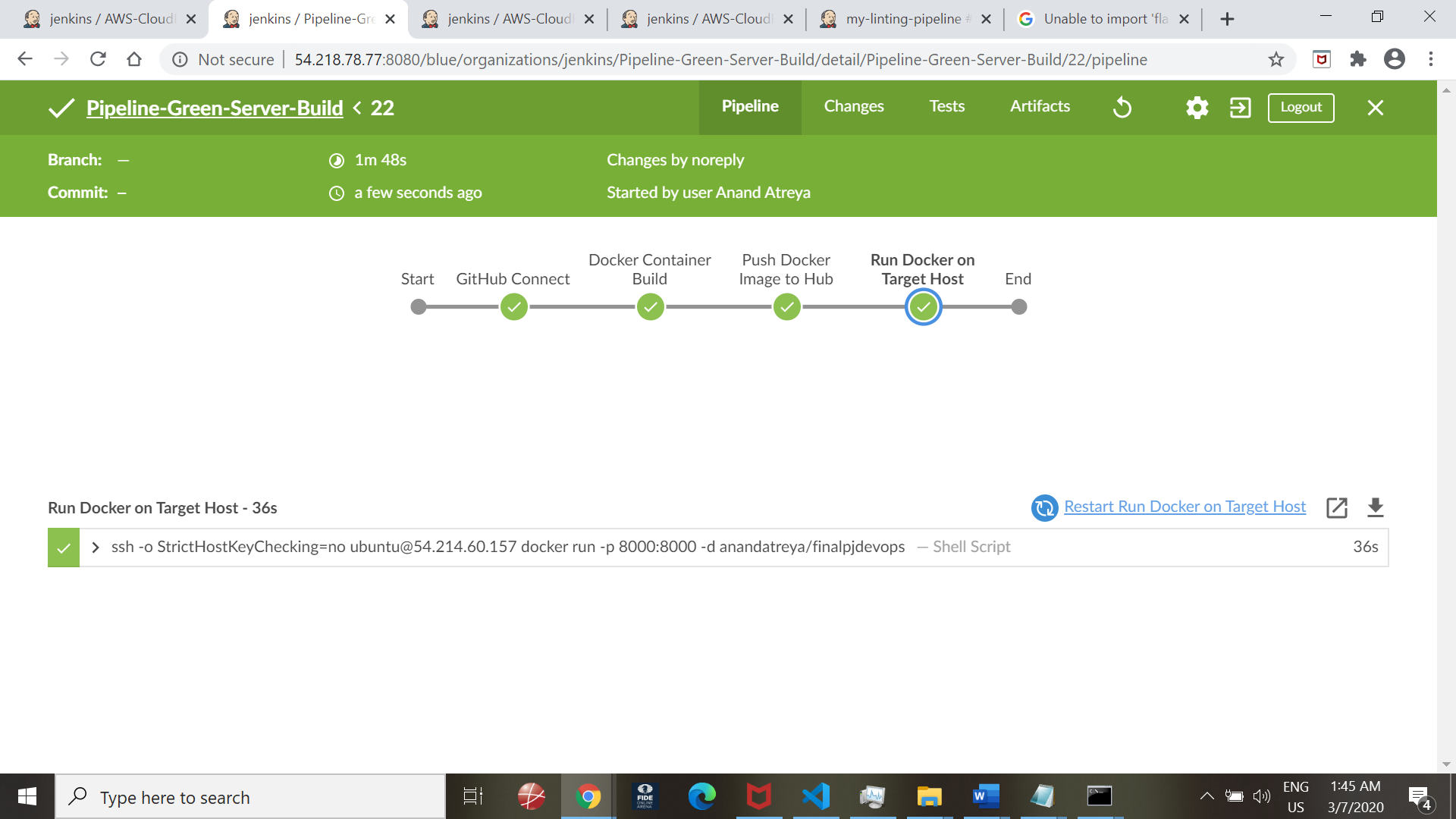
1. **Docker Container Build**

**Green Server Build:**



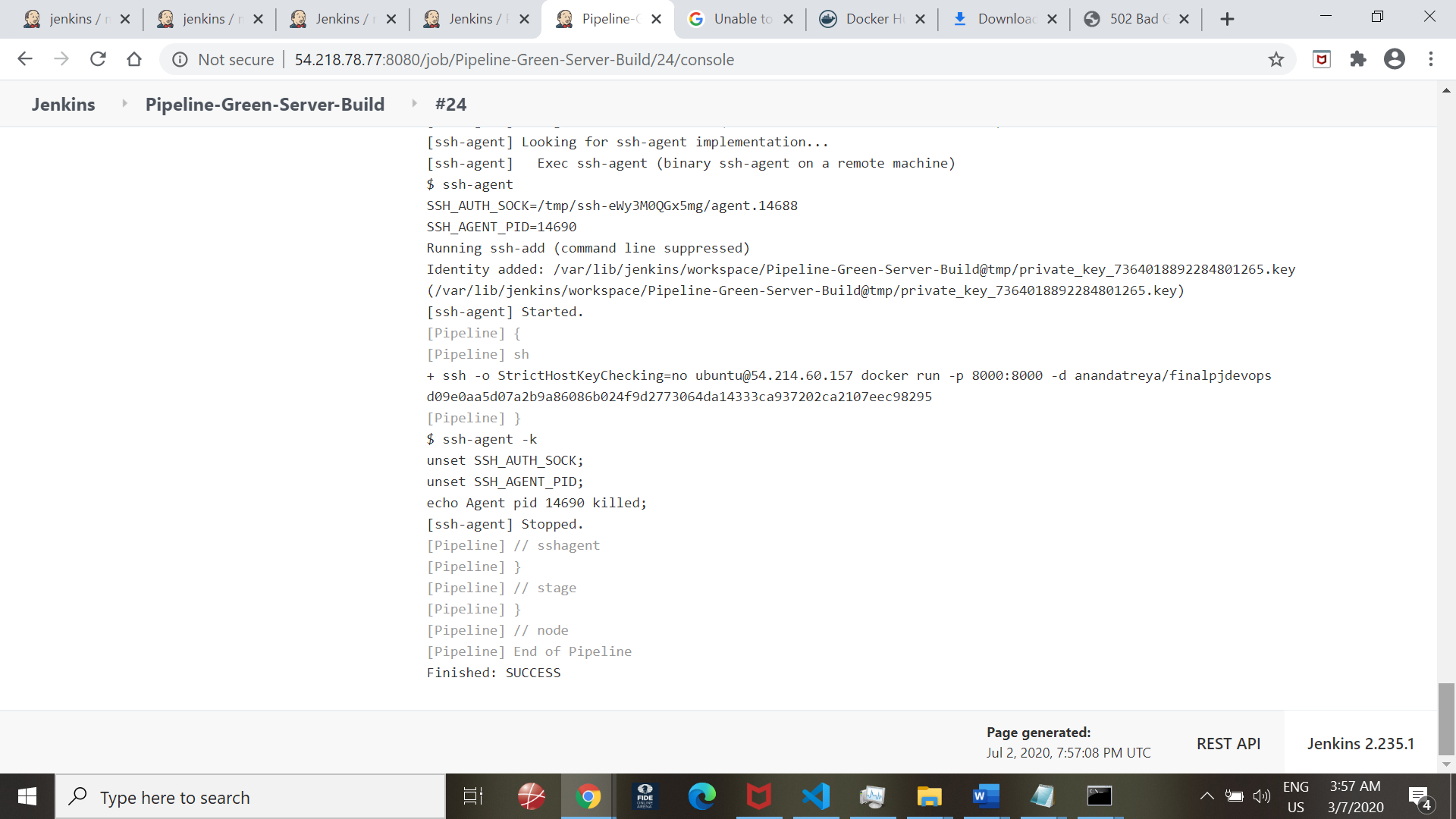
All steps successful:

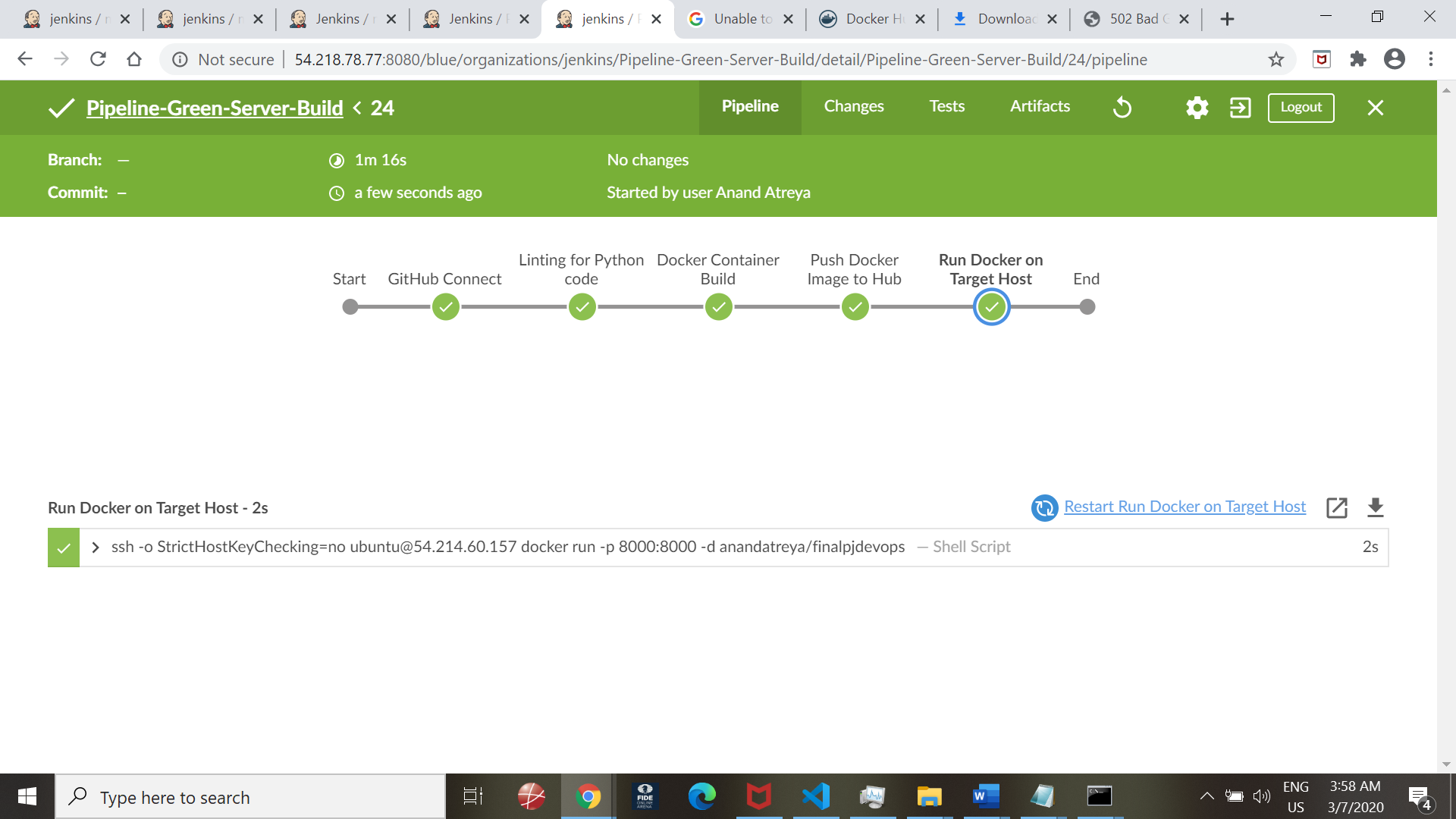




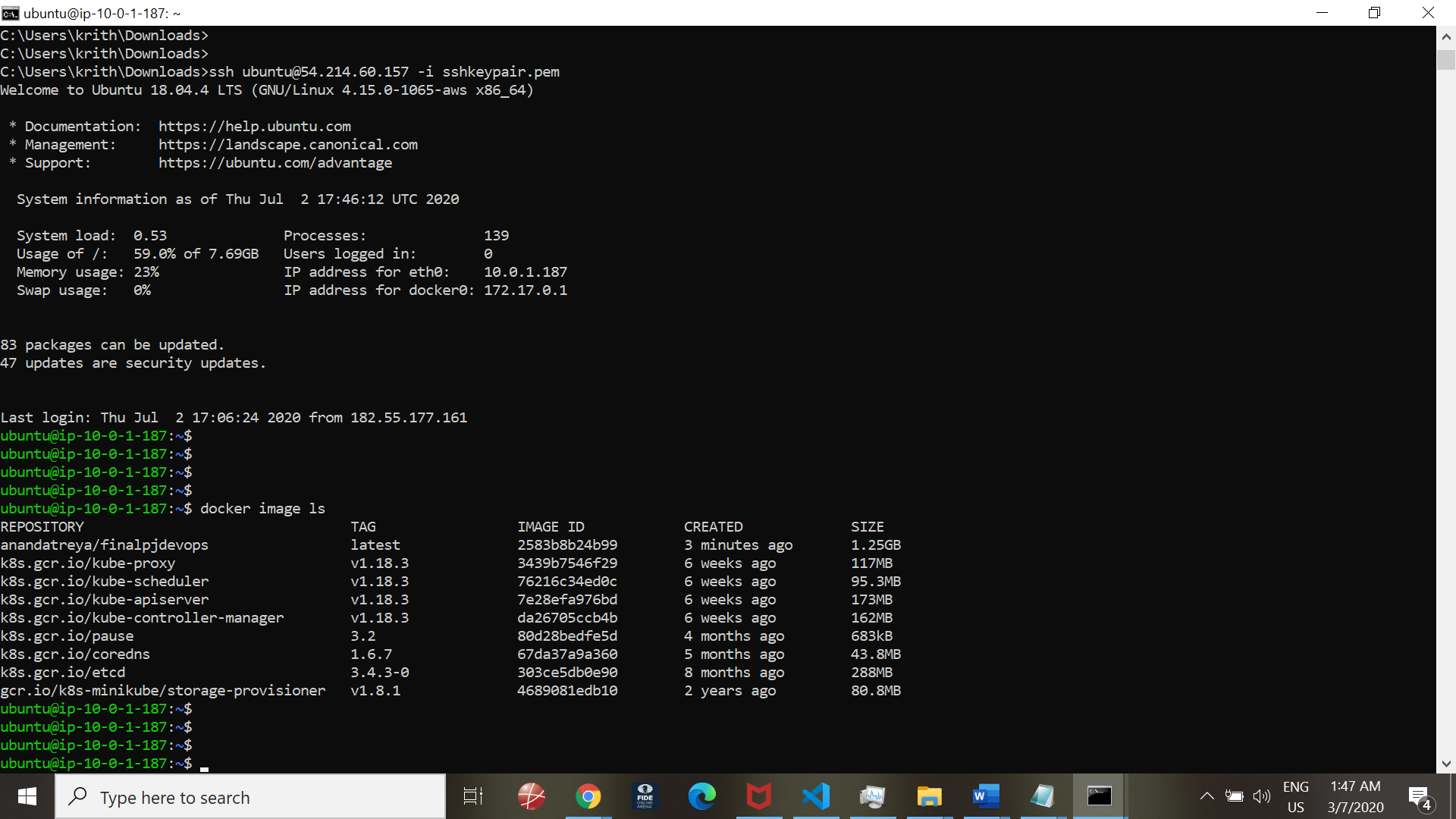
Additional step of Linting included to the overall steps –

Build of Docker container (Including Linting into the overall steps) – Green Server Deployment

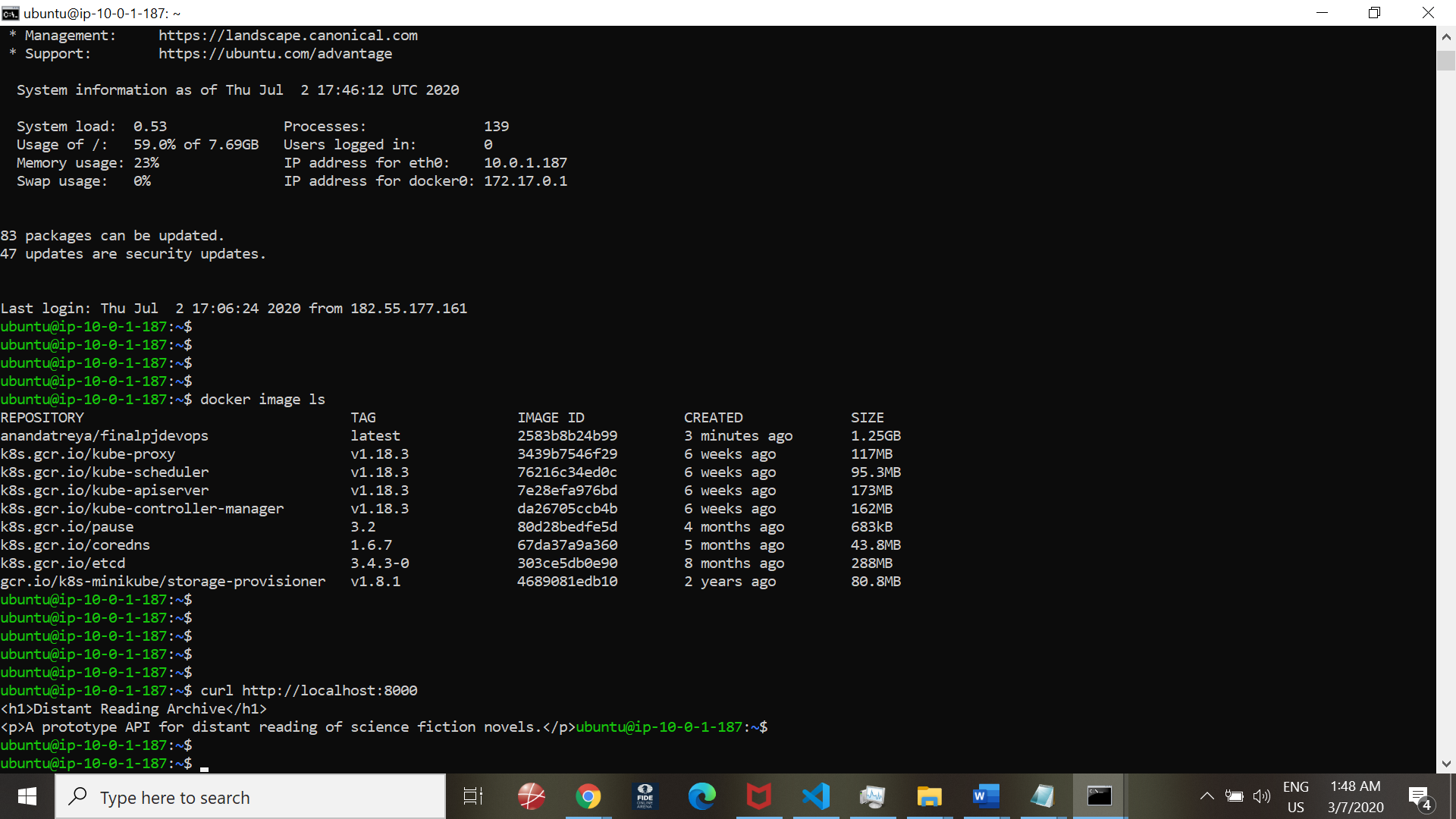




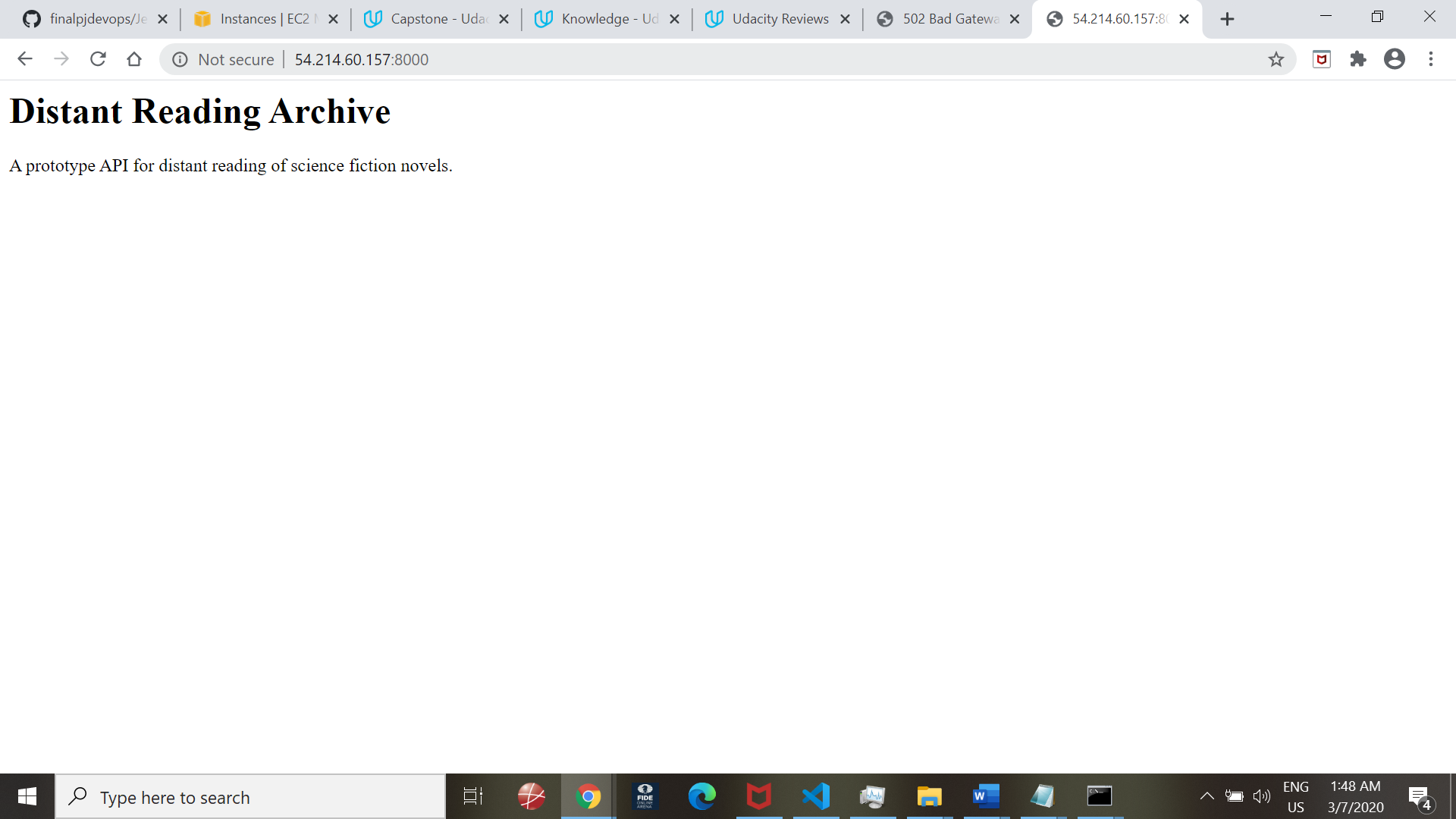
1. **Test / Verify Docker Image on target server:**



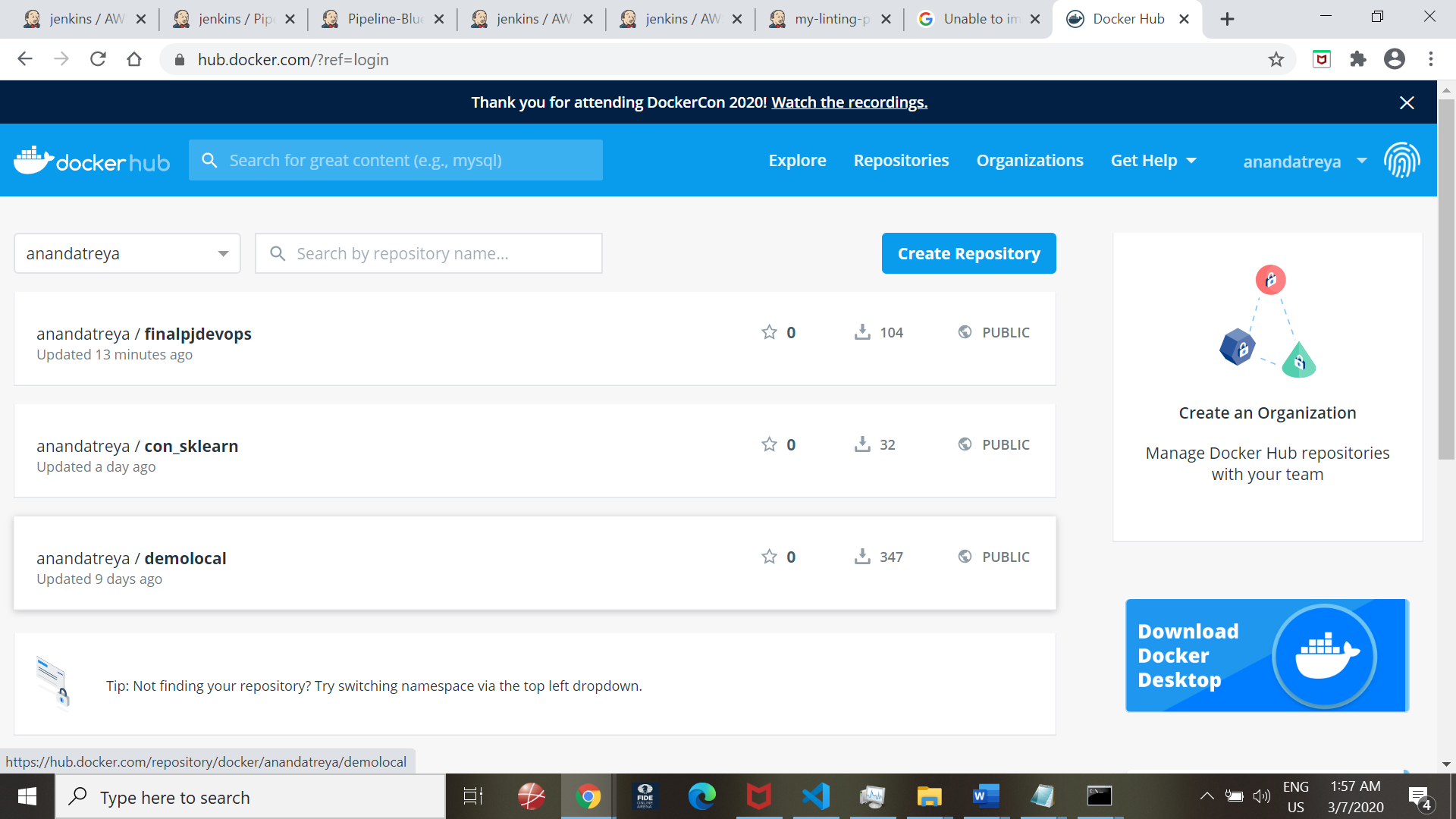
**Testing:**



Testing from Browser:

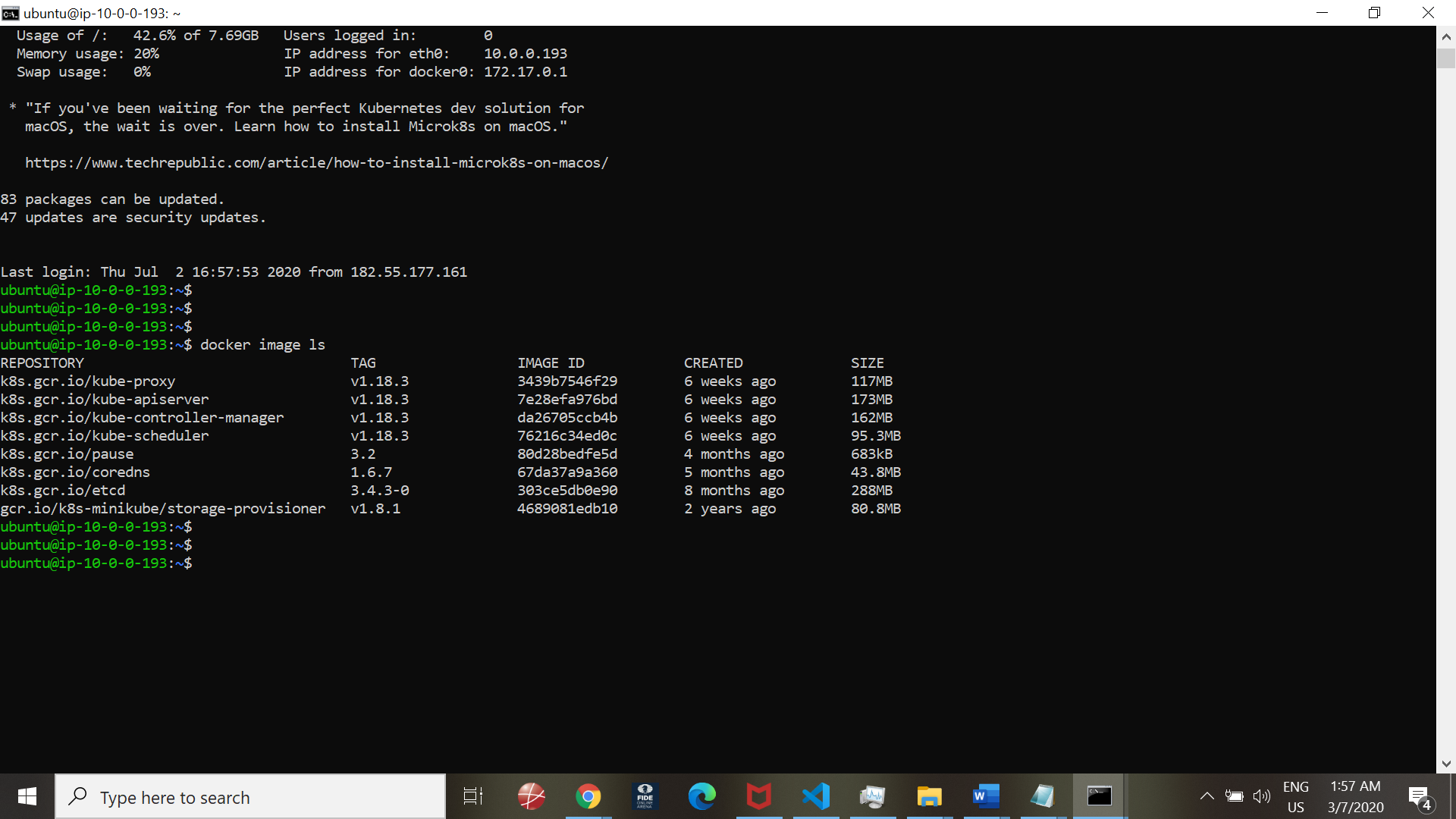


Dockerhub verification for image sync timestamp

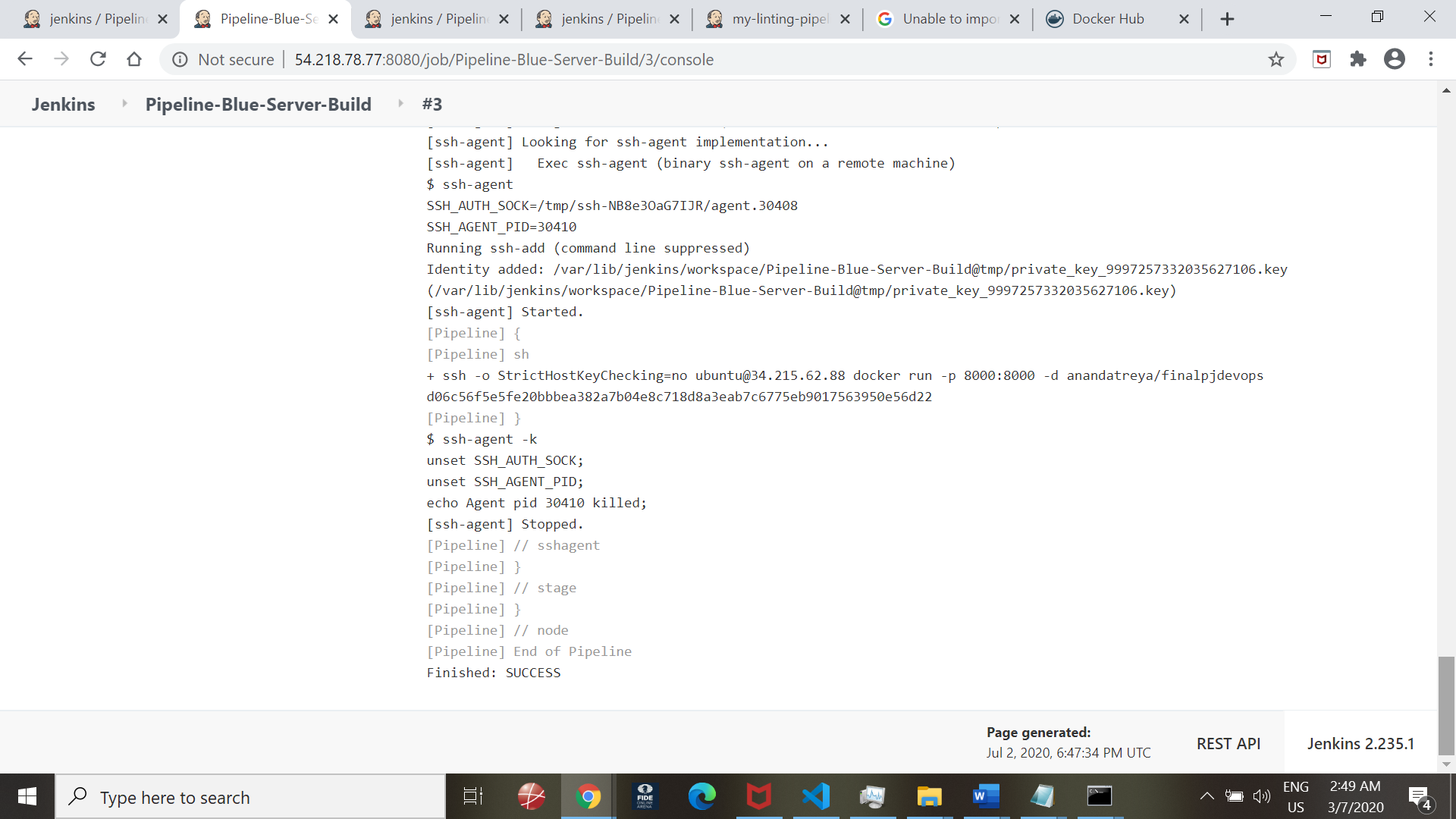


**NEXT: We build the blue server**

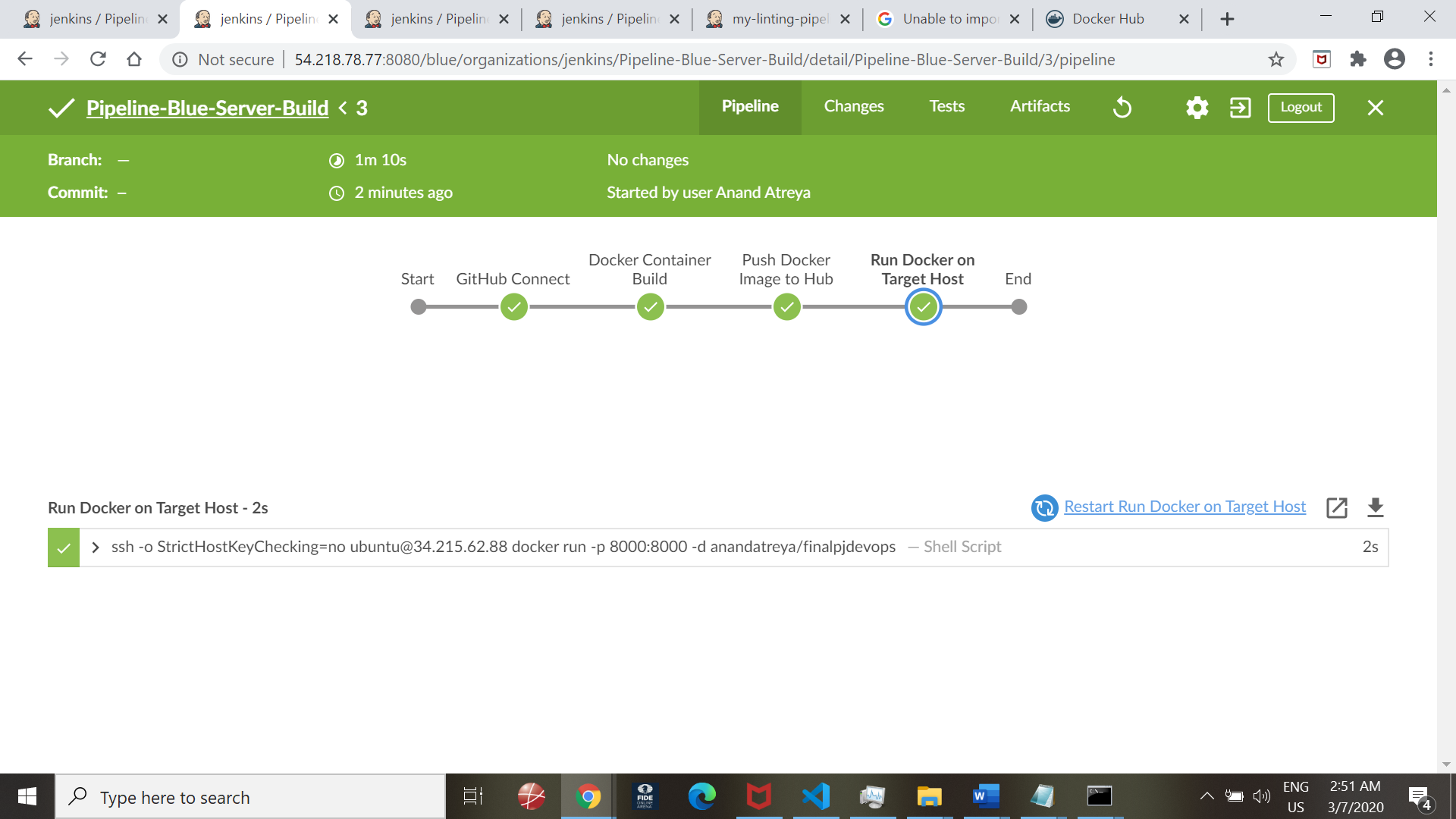
Prechecks:



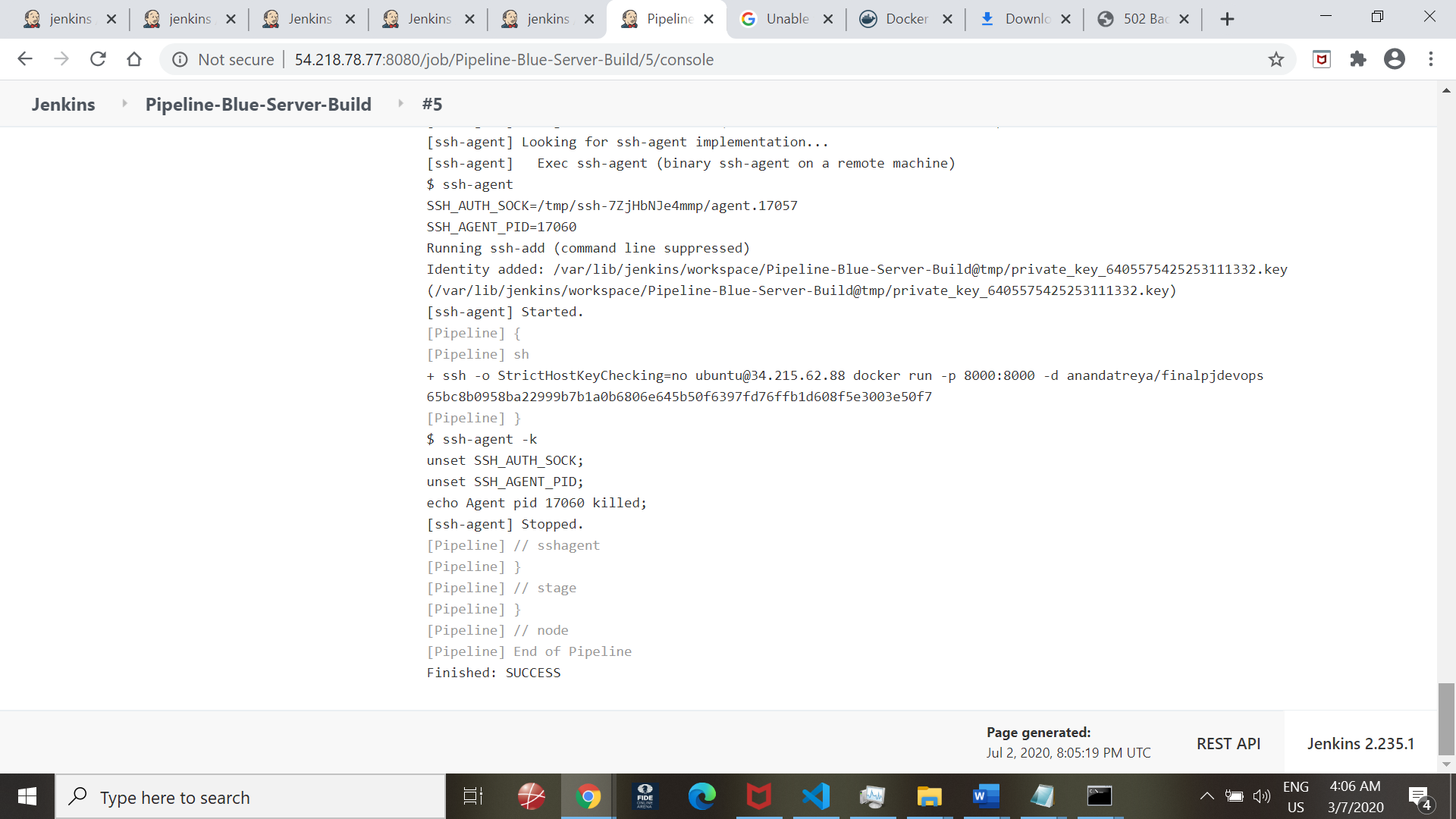
Execution of the Blue Pipeline method:

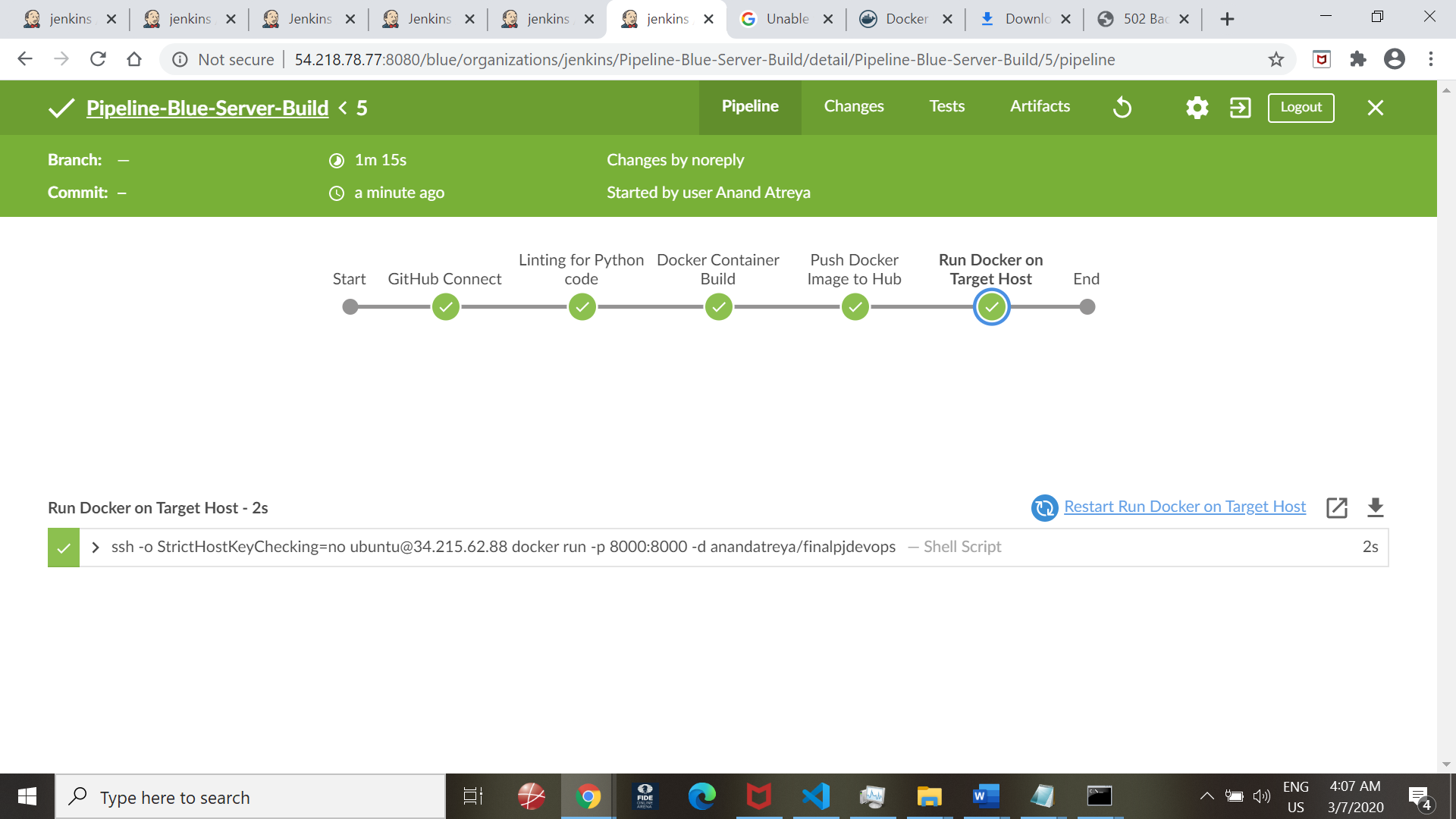


BlueOcean workflow view:

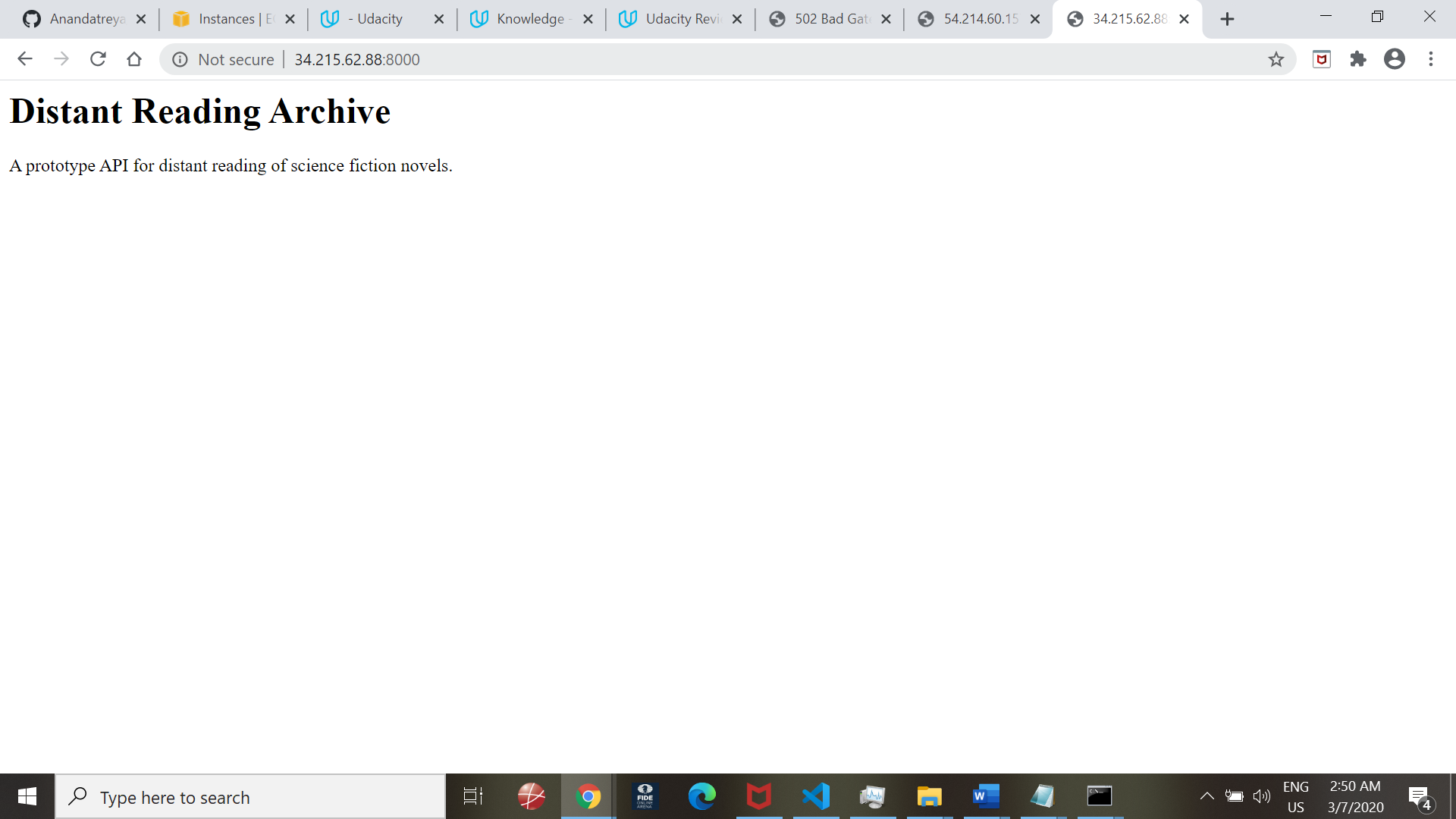


Build of docker container for Blue Deployment server (including Linting step) –





Verification from Browser to query Server IP:8000



**4. Docker container deployment on Kubernetes cluster (Green Node):**

-10-0-0-193\_kube-system\_266247c448a2914846d4f926d428568d\_0

root@ip-10-0-0-193:~# docker stop k8s\_podfinalpjdevops\_podf

Error response from daemon: No such container: k8s\_podfinalpjdevops\_podf

**root@ip-10-0-0-193:~# kubectl port-forward podfinalpjdevops 8000:8000**

**Forwarding from 127.0.0.1:8000 -> 8000**

**Forwarding from [::1]:8000 -> 8000**

**Handling connection for 8000**

**Handling connection for 8000**

**Handling connection for 8000**

**Handling connection for 8000**

Curl commands executed as below to receive response from Kubernetes clustr

ubuntu@ip-10-0-0-193:~$ curl http://localhost:8000

<h1>Distant Reading Archive</h1>

ubuntu@ip-10-0-0-193:~$ curl http://localhost:8000

<h1>Distant Reading Archive</h1>

<p>A prototype API for distant reading of science fiction novels.</p>ubuntu@ip-10-0-0-193:~$ curl http://localhost:8000

<h1>Distant Reading Archive</h1>

<p>A prototype API for distant reading of science fiction novels.</p>ubuntu@i

root@ip-10-0-0-193:~# kubectl get nodes

NAME STATUS ROLES AGE VERSION

ip-10-0-0-193 Ready master 163m v1.18.3

root@ip-10-0-0-193:~#

root@ip-10-0-0-193:~#

root@ip-10-0-0-193:~# kubectl get pods

NAME READY STATUS RESTARTS AGE

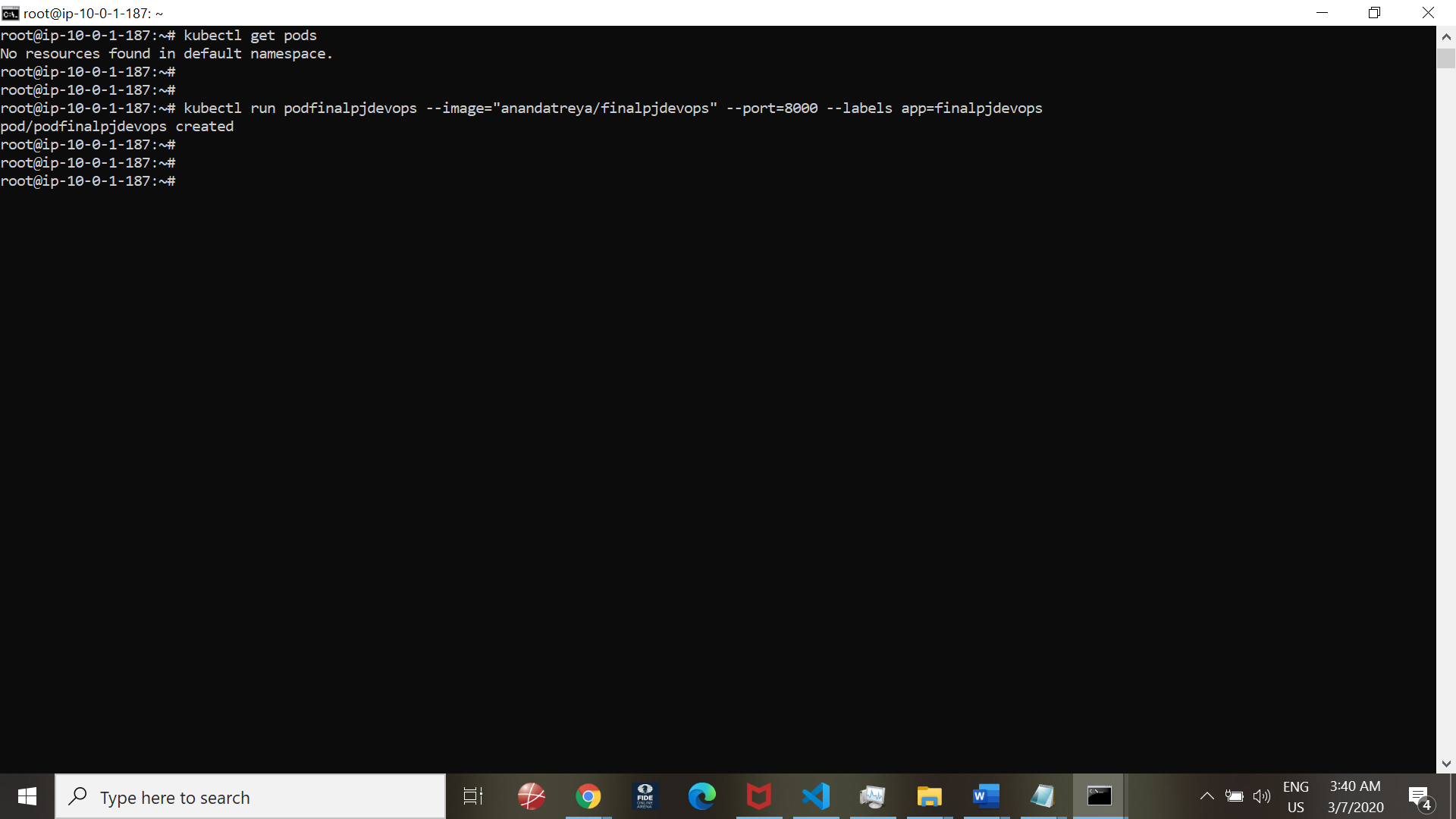
podfinalpjdevops 1/1 Running 0 11m

root@ip-10-0-0-193:~#

root@ip-10-0-0-193:~#

root@ip-10-0-0-193:~#

**Docker container deployment on Kubernetes cluster (Green Node):**



**Testing the deployment:**

root@ip-10-0-1-187:~# kubectl run podfinalpjdevops --image="anandatreya/finalpjdevops" --port=8000 --labels app=finalpjdevops

pod/podfinalpjdevops created

root@ip-10-0-1-187:~#

root@ip-10-0-1-187:~#

root@ip-10-0-1-187:~# kubectl get pods

NAME READY STATUS RESTARTS AGE

podfinalpjdevops 1/1 Running 0 60s

root@ip-10-0-1-187:~#

root@ip-10-0-1-187:~#

Login to the container:

root@ip-10-0-1-187:~# kubectl exec -it podfinalpjdevops -- sh

#

#

# ls -ltra

total 40

-rw-r--r-- 1 root root 503 Jul 2 17:43 run\_kubernetes.sh

-rw-r--r-- 1 root root 1265 Jul 2 17:43 app.py

-rw-r--r-- 1 root root 107 Jul 2 17:43 README.md

-rw-r--r-- 1 root root 1934 Jul 2 17:43 Jenkinsfile

-rw-r--r-- 1 root root 512 Jul 2 17:43 Dockerfile

-rw-r--r-- 1 root root 197 Jul 2 19:16 requirements.txt

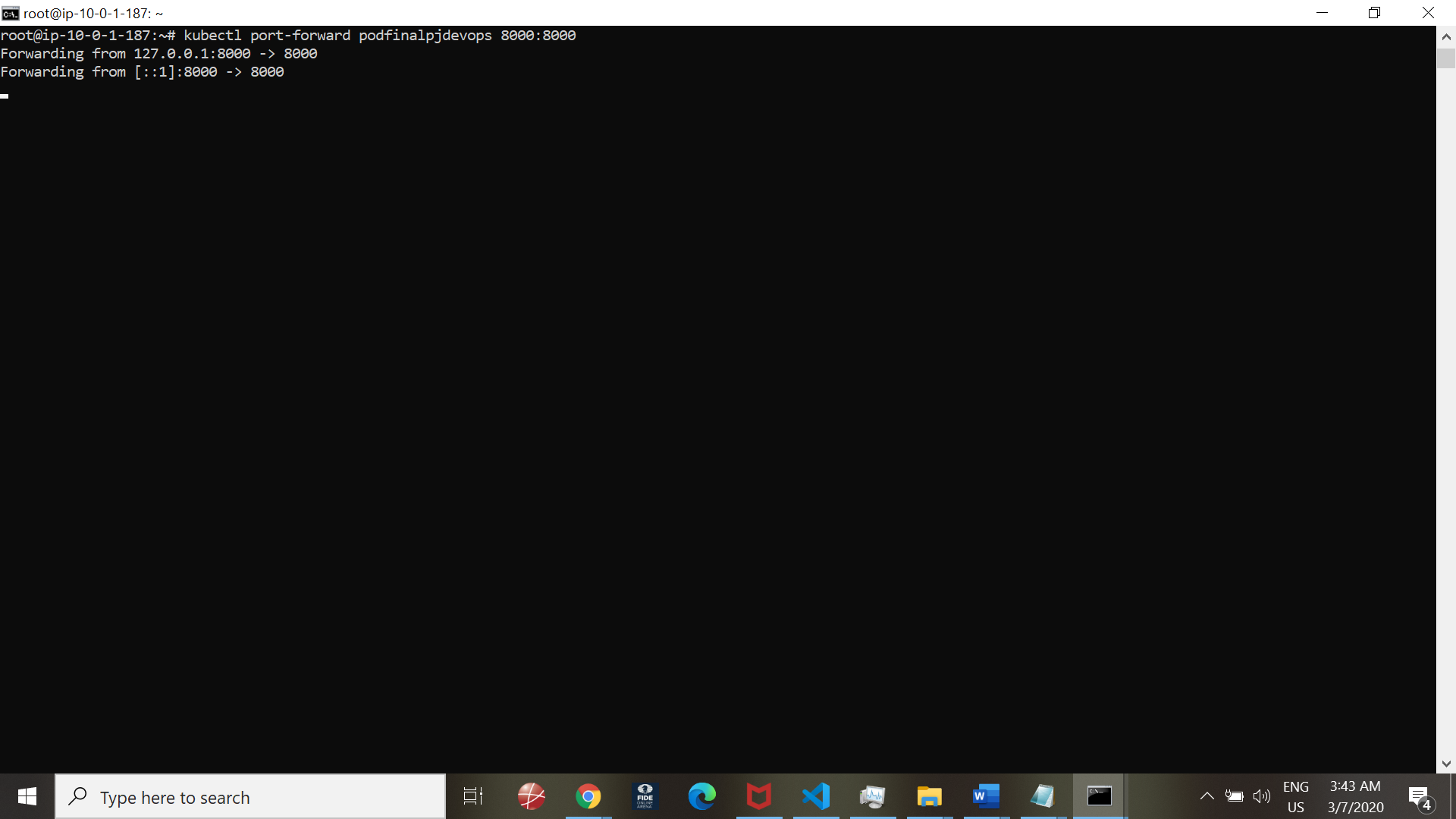
-rw-r--r-- 1 root root 969 Jul 2 19:16 Makefile

drwxr-xr-x 8 root root 4096 Jul 2 19:16 .git

drwxr-xr-x 1 root root 4096 Jul 2 19:16 .

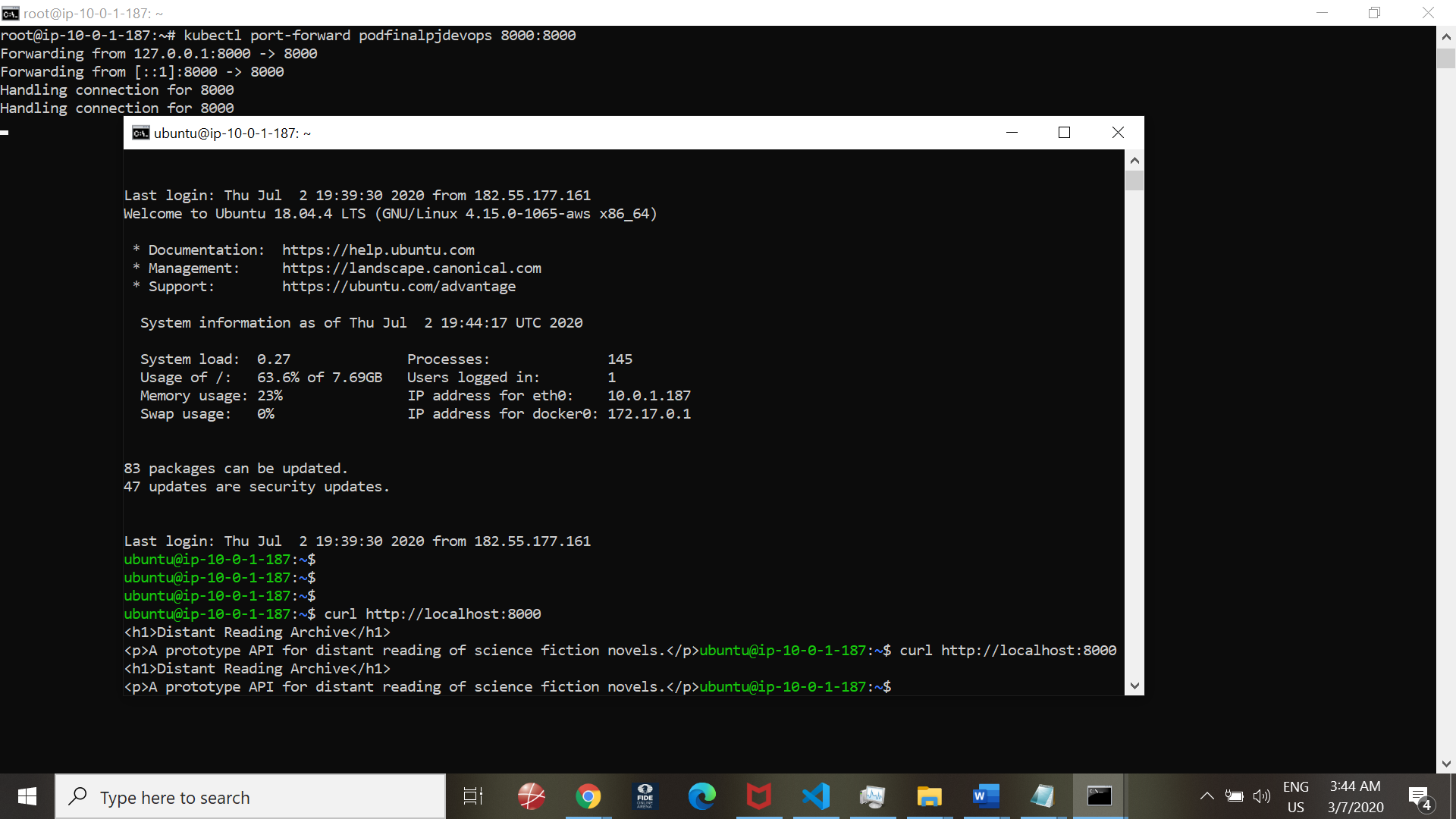
drwxr-xr-x 1 root root 4096 Jul 2

Forward port

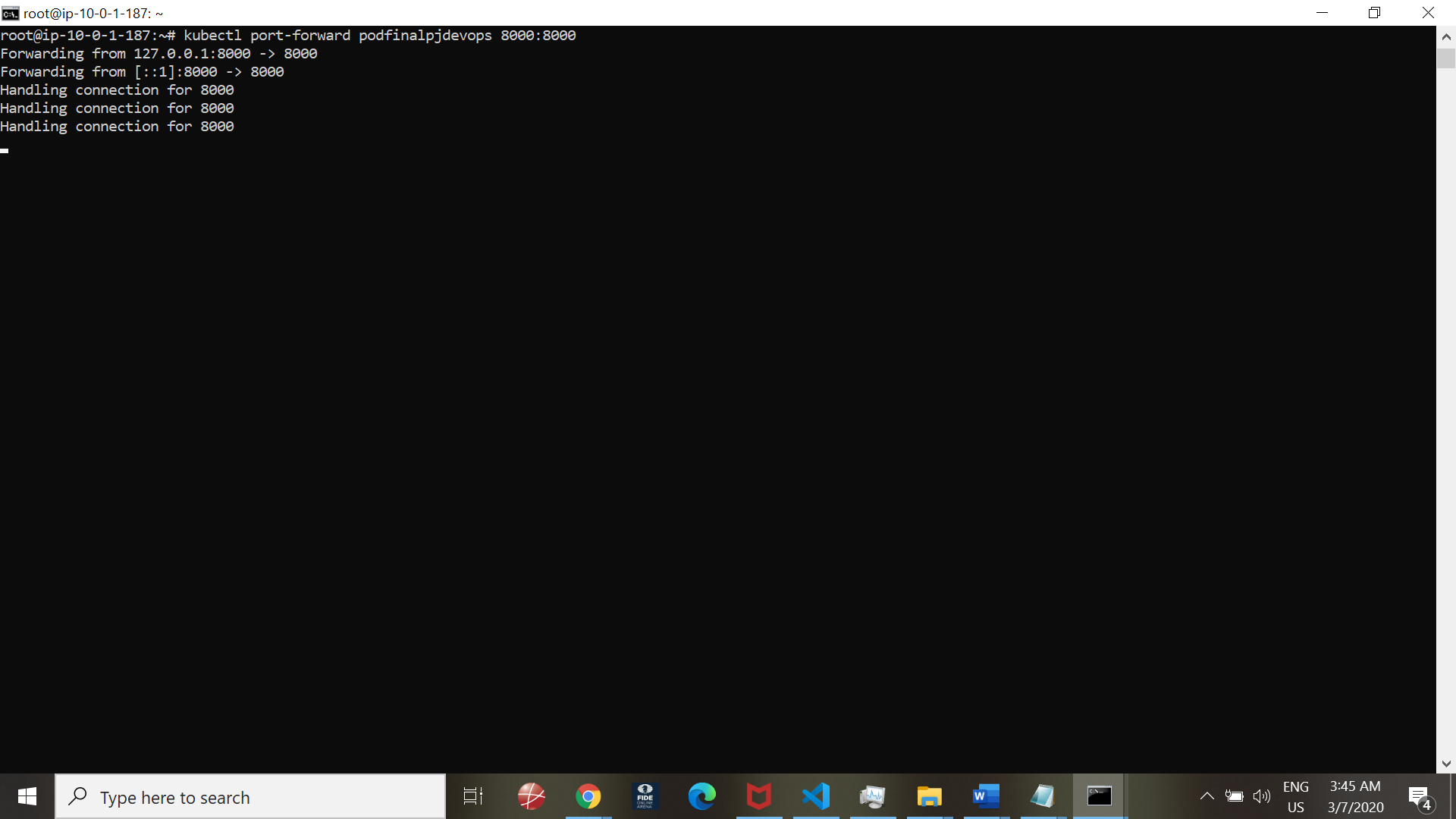


Test:

Use of Curl command to the localhost:8000



Response from server:



END--