**Edureka Final Project Presentation from Lanre Bolaji**

**Reference GitHub. https://github.com/bolajil/Edureka\_my\_final\_project**

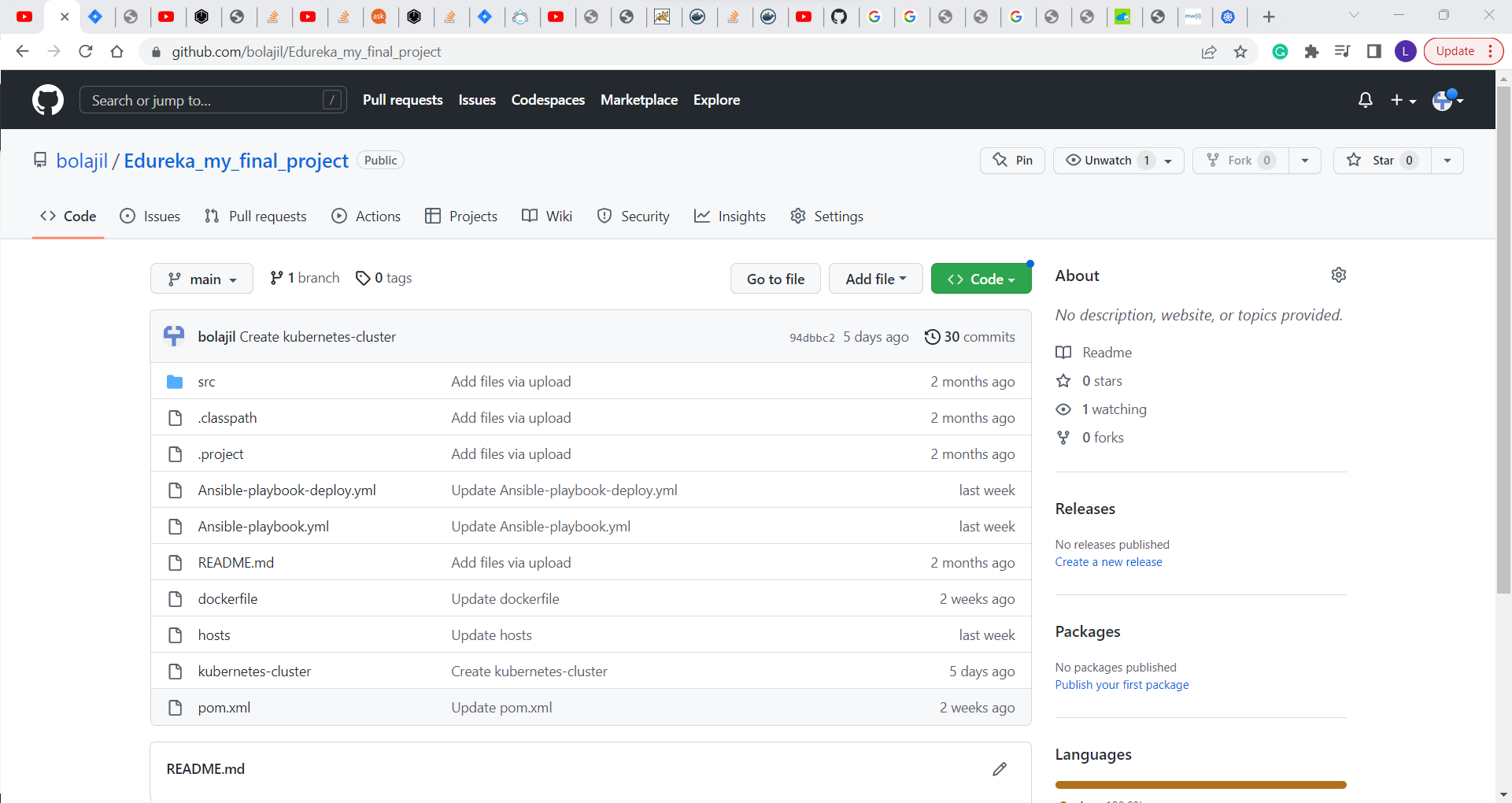
**To resolve Issues faced by Business owners:**

ABC Technologies is a leading online retail store, and it has recently acquired a large retail offline business store. The business store has many stores across the globe but is following the conventional pattern of development and deployment. As a result, it has landed at a significant loss and is facing the following challenges. • Low available • Low scalable • Low performance, hard to build and maintain • Developing and deploying is time-consuming.

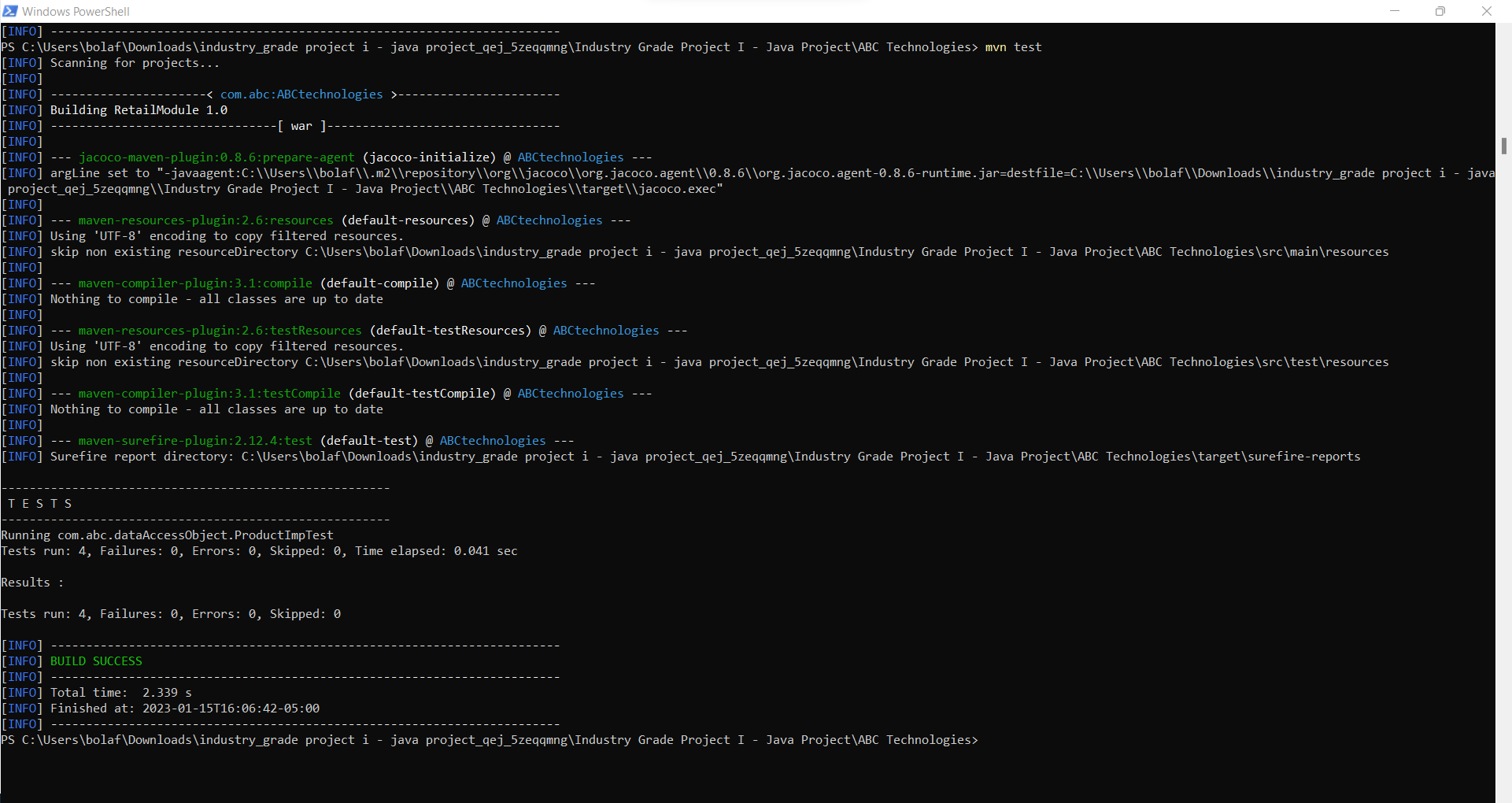
Below are some of the high-level goals of this project: • Implement CI/CD such that ABC Company can be— ▪ highly available ▪ highly scalable ▪ highly performant ▪ quickly built and maintained ▪ developed and deployed quickly.

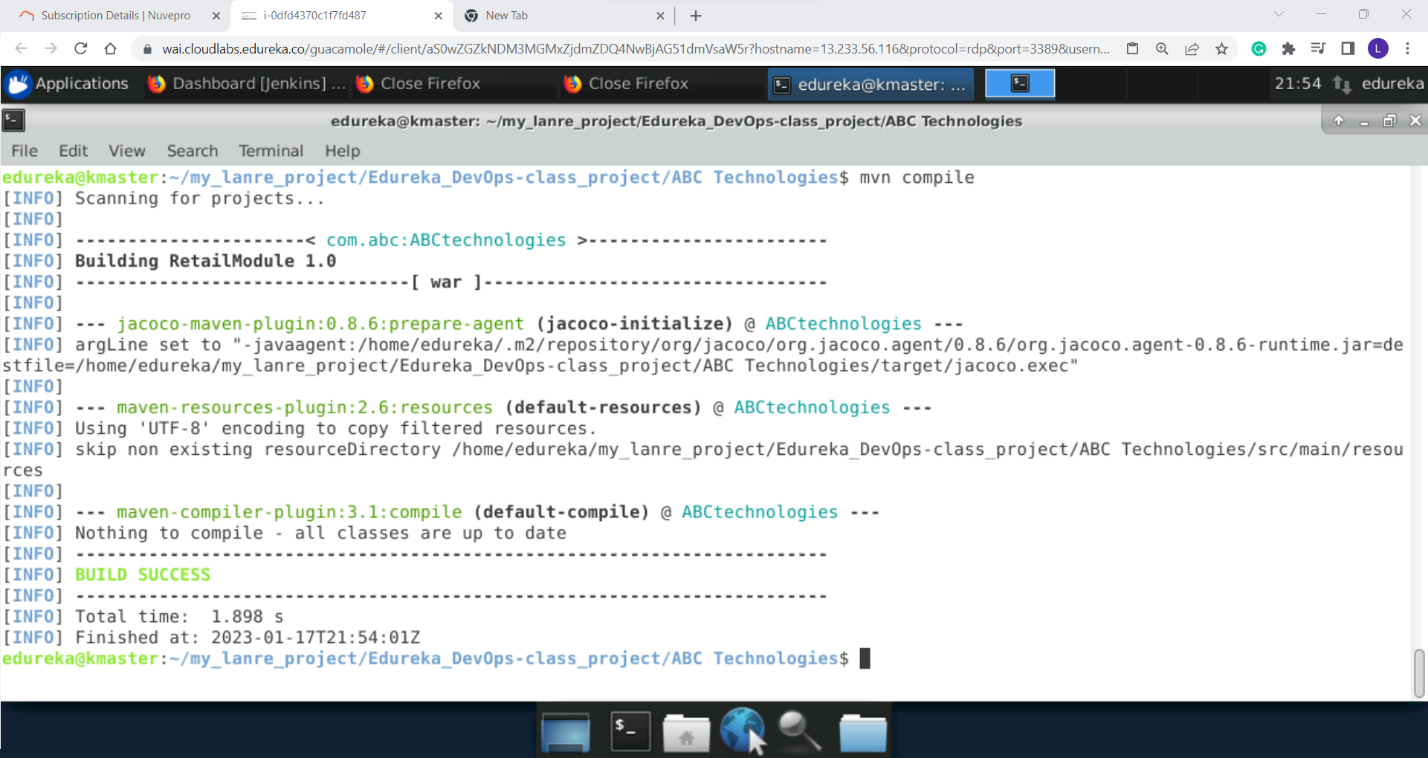
**Solutions:**

Task 1: Clone the project from the GitHub link shared in resources to your local machine. Build the code using Maven commands.

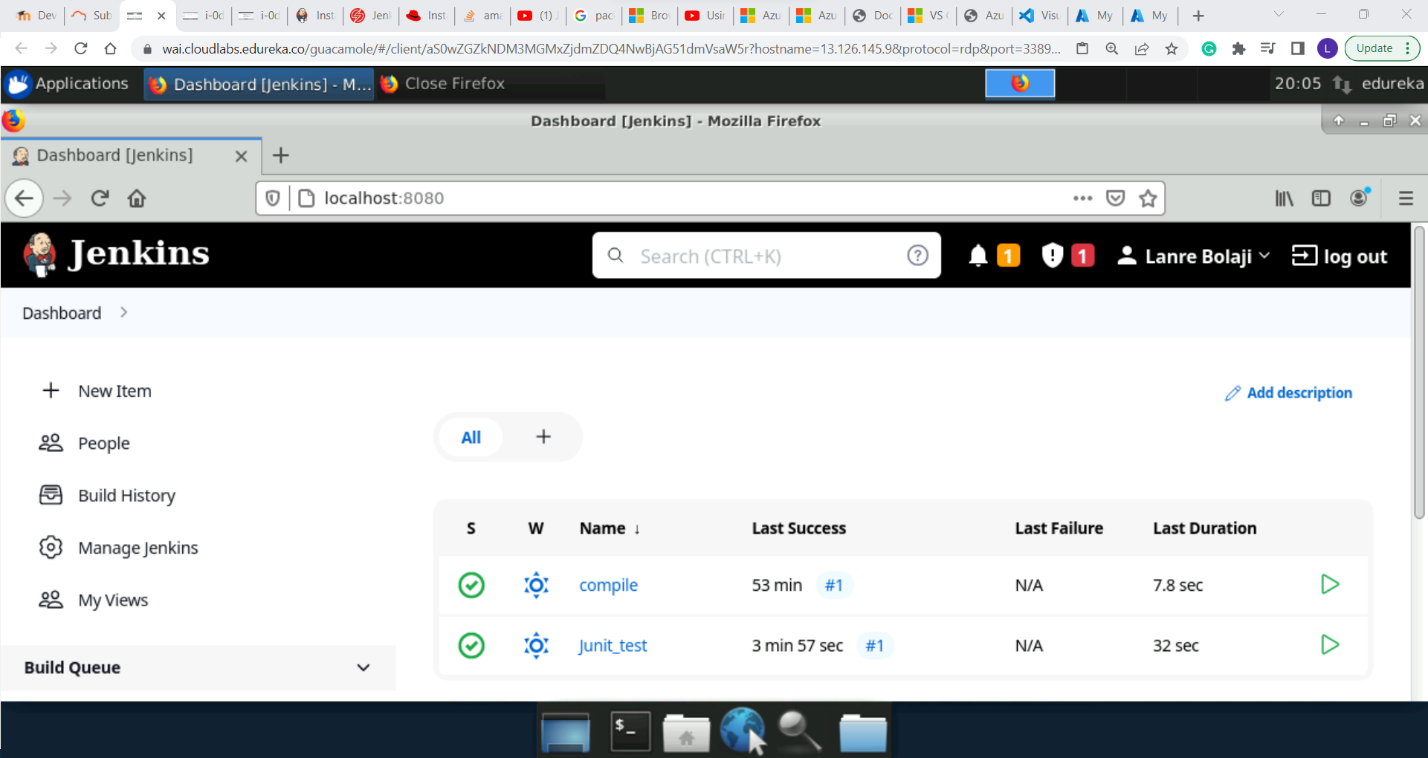


Maven test on repository output:

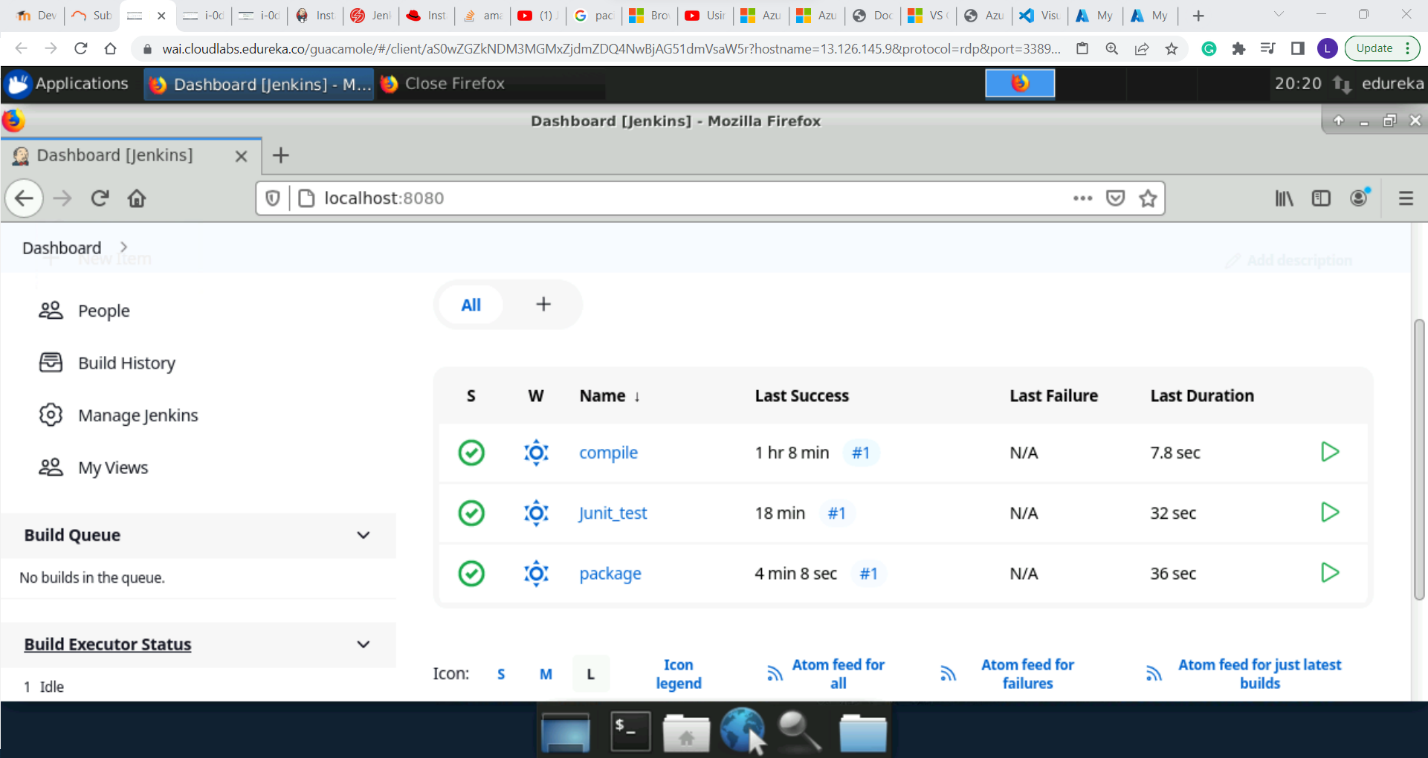




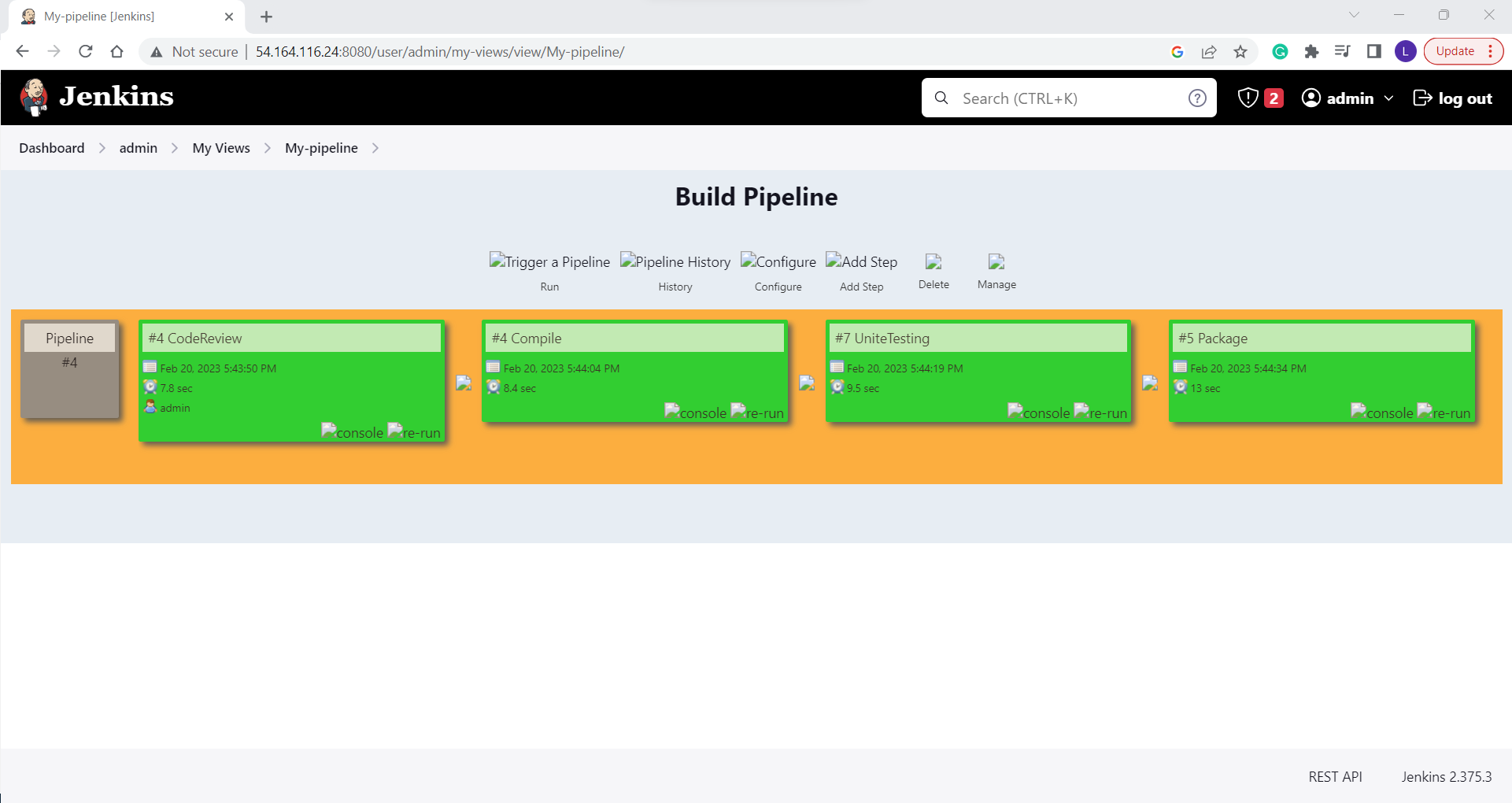
Task 2: Set up the Git repository and push the source code. Then, log in to Jenkins. 1. Create a build pipeline containing a job for each • One for compiling source code • Second for testing source code •

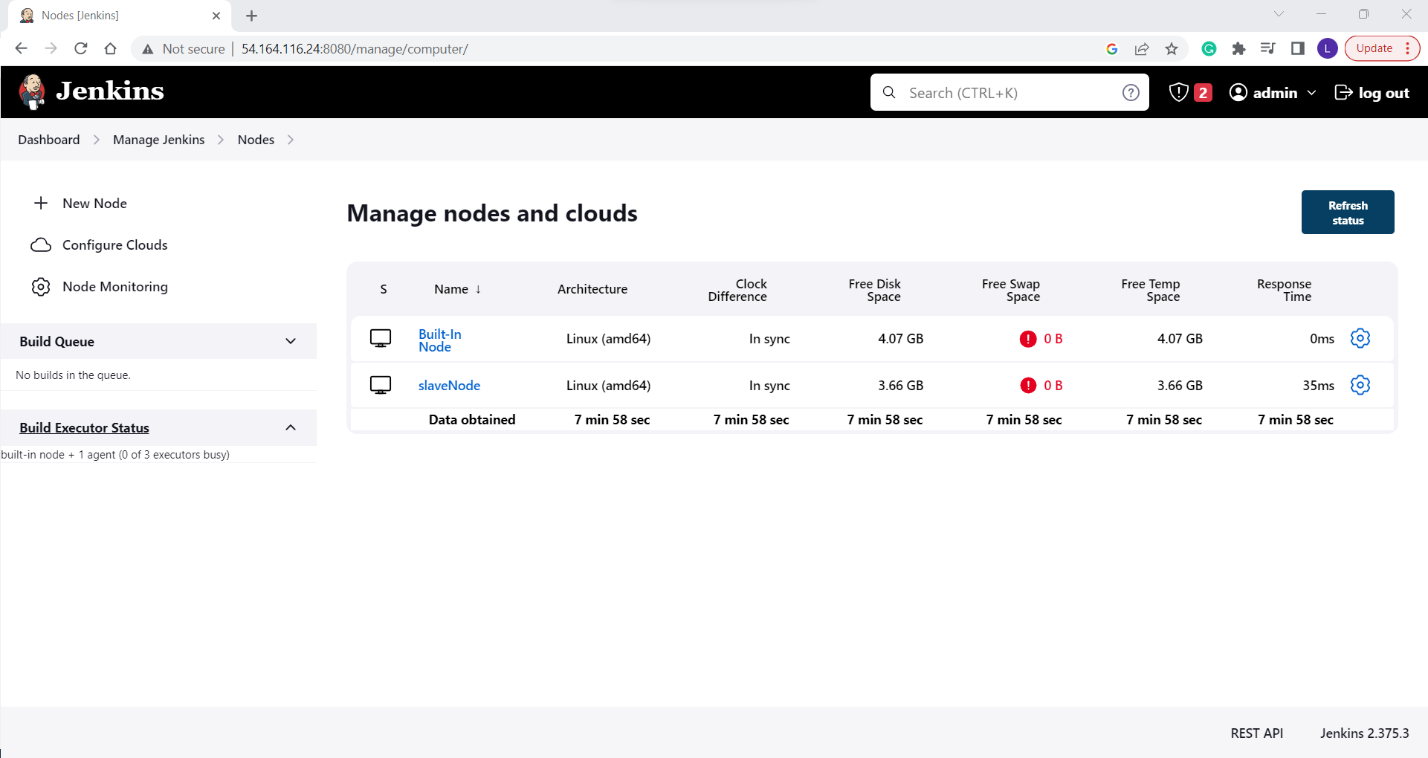


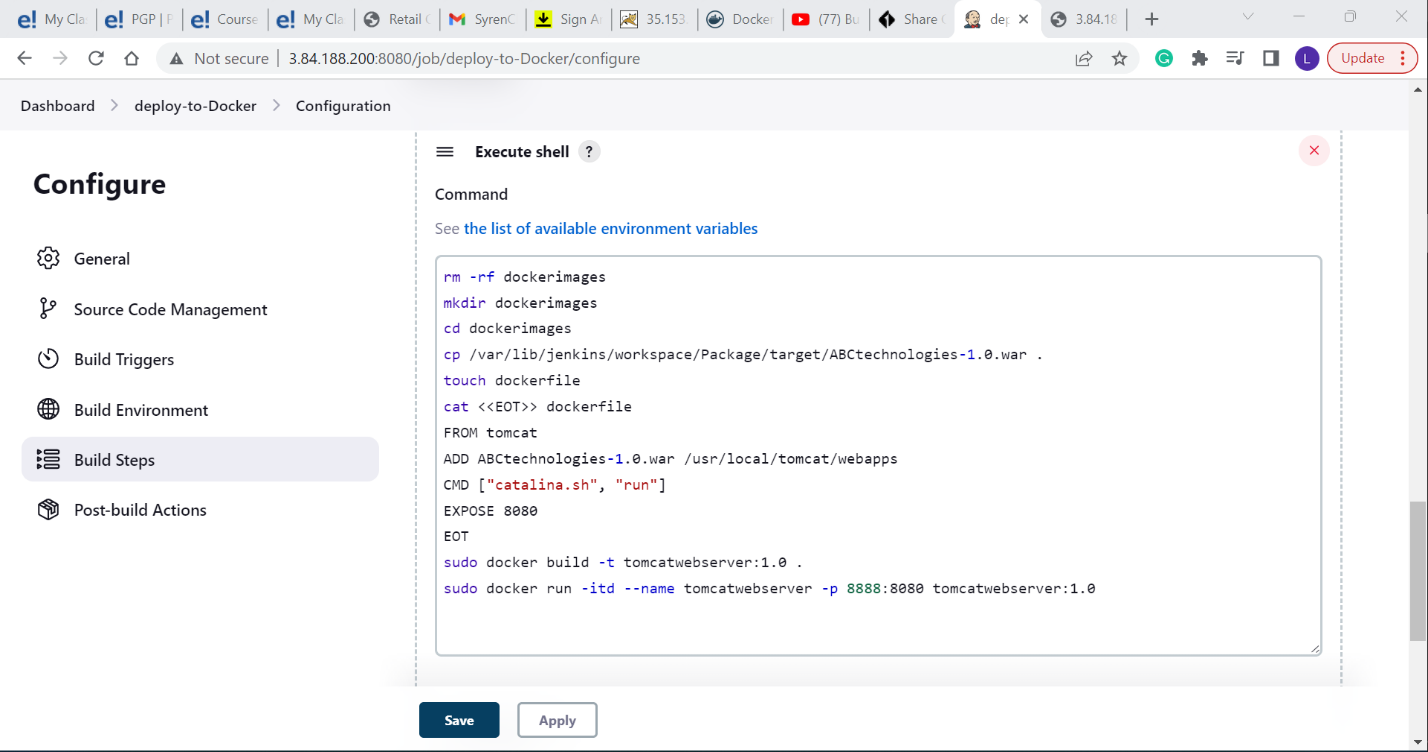
Third for packing the code 2.

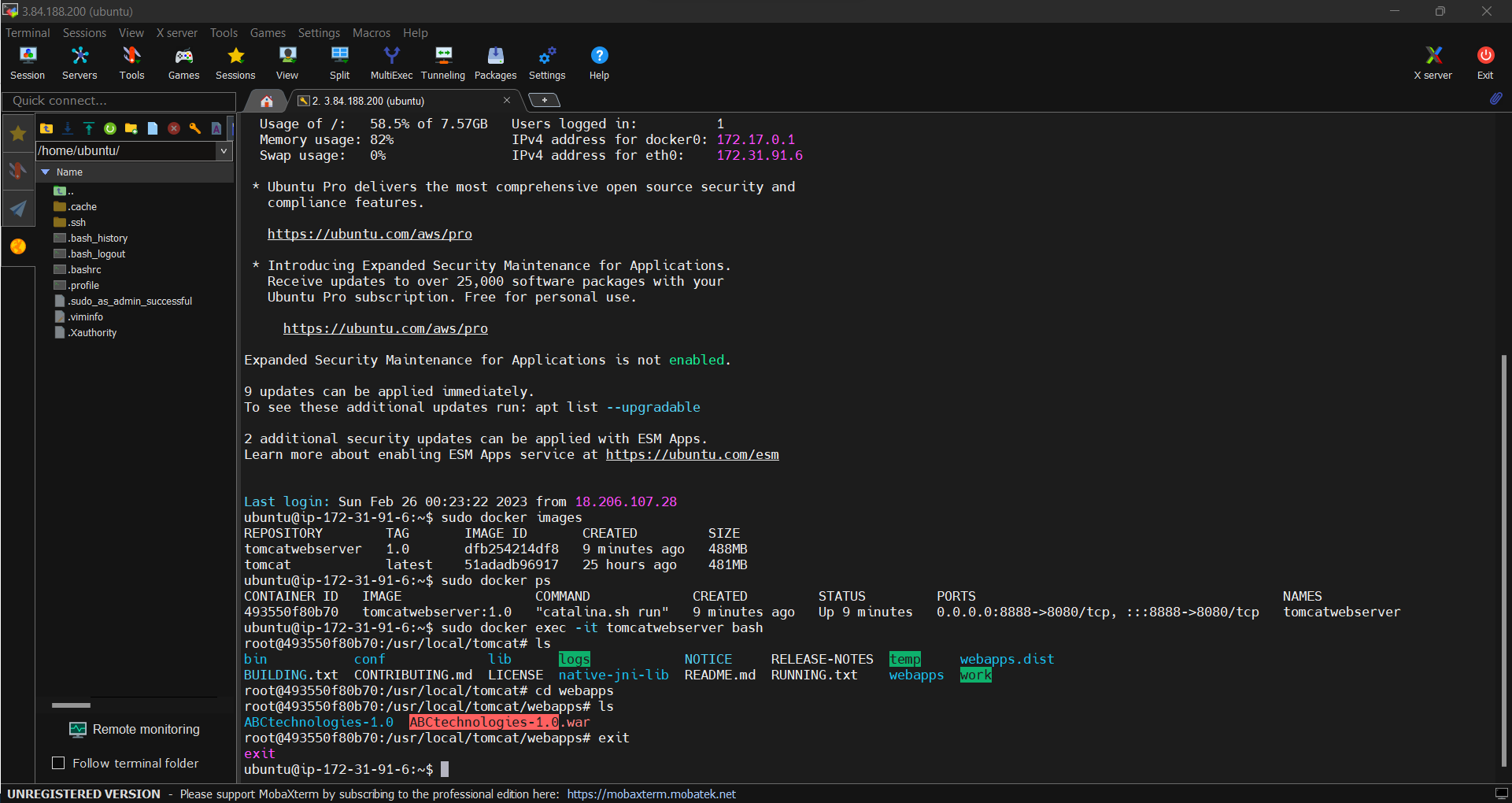


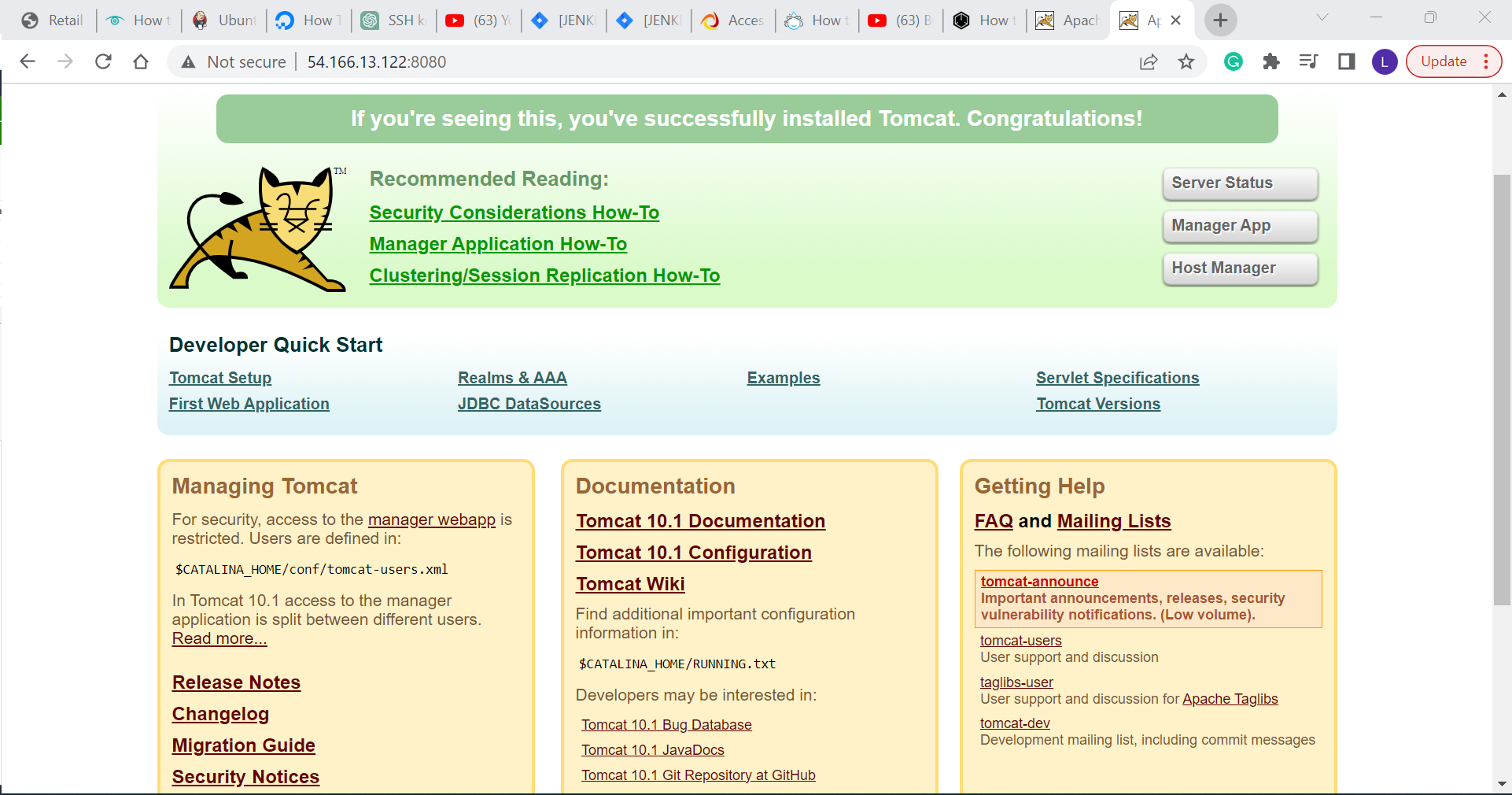
Execute the CI/CD pipeline to execute the jobs created in step 1 3. Set up a master-slave node to distribute the tasks in the pipeline.

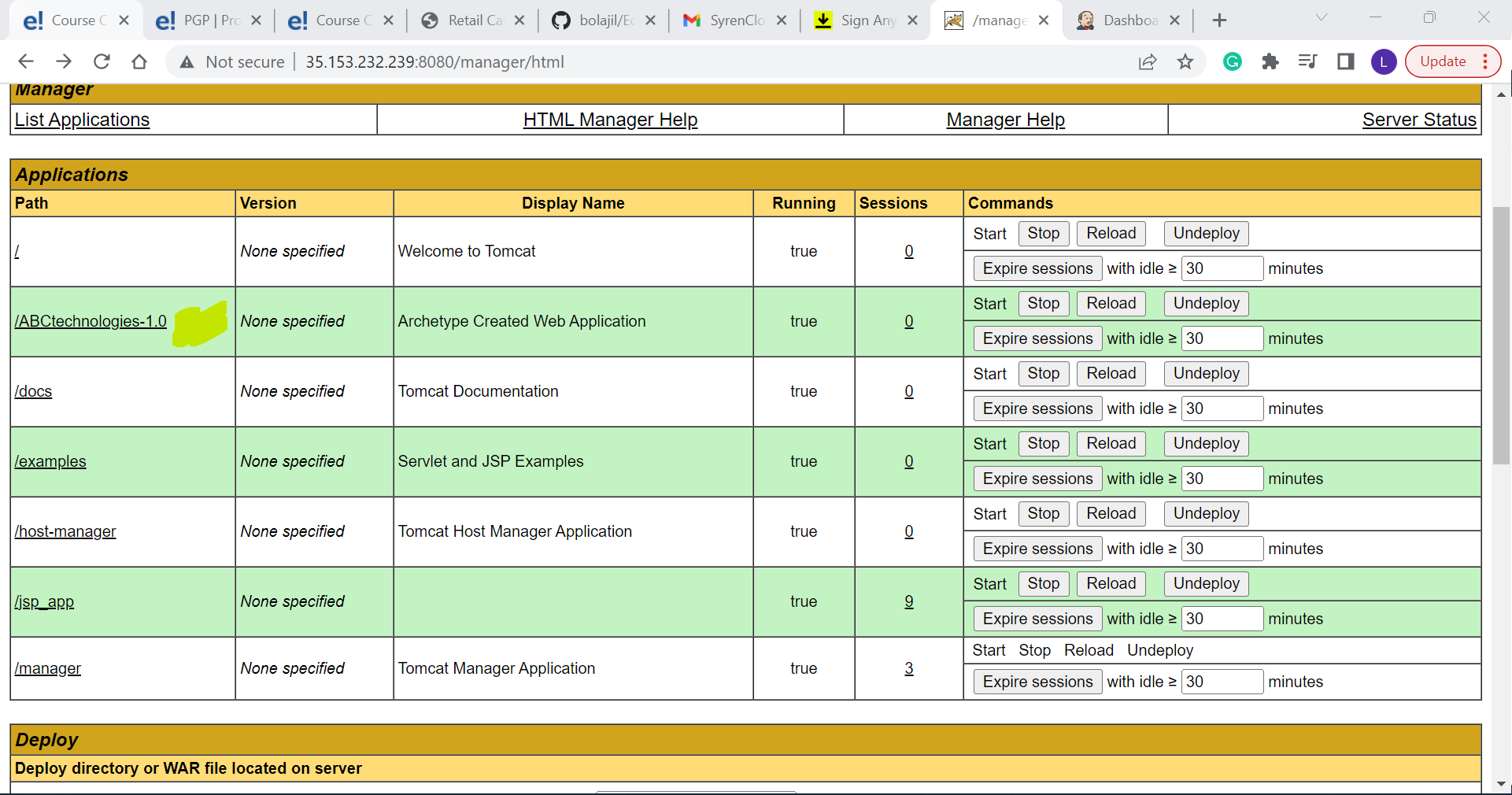


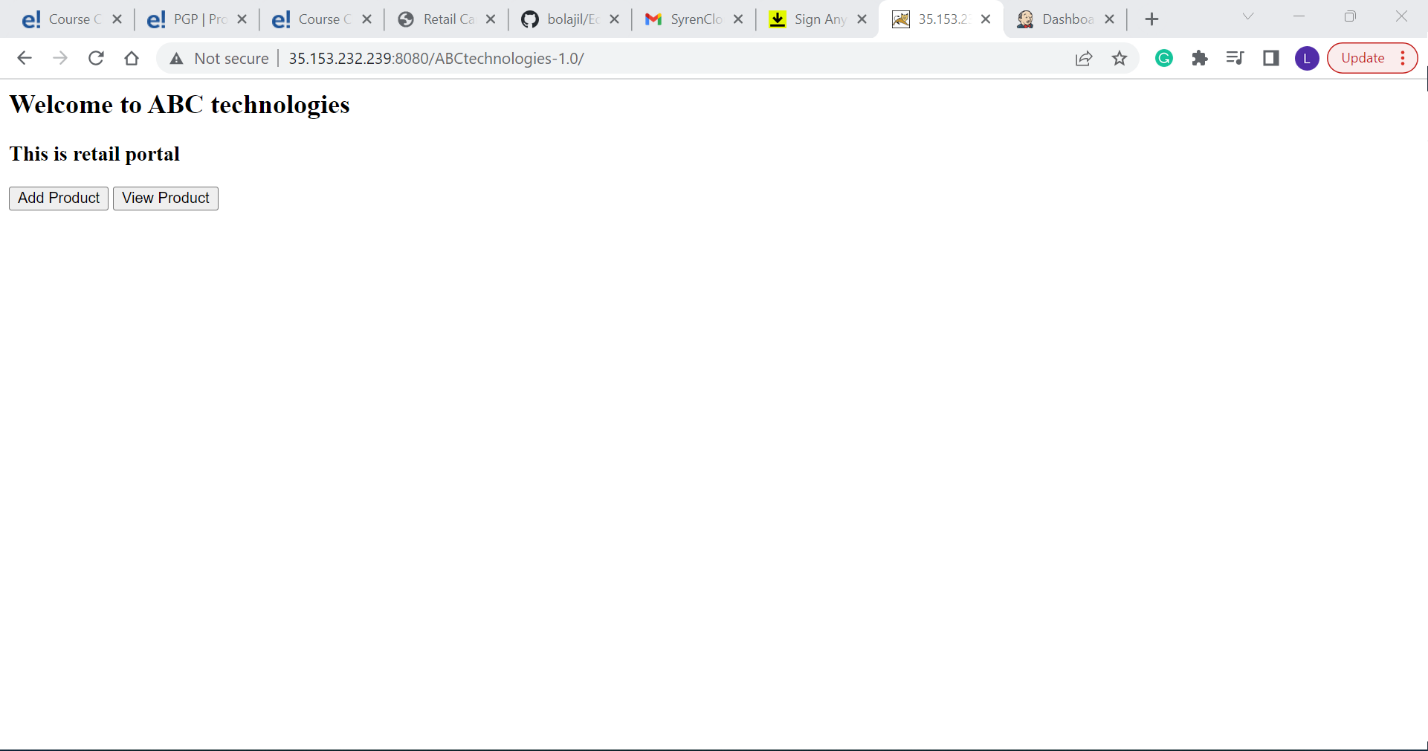
 Task 3: Write a Docket file. Create an Image and container on the Docker host. Integrate docker host with Jenkins.



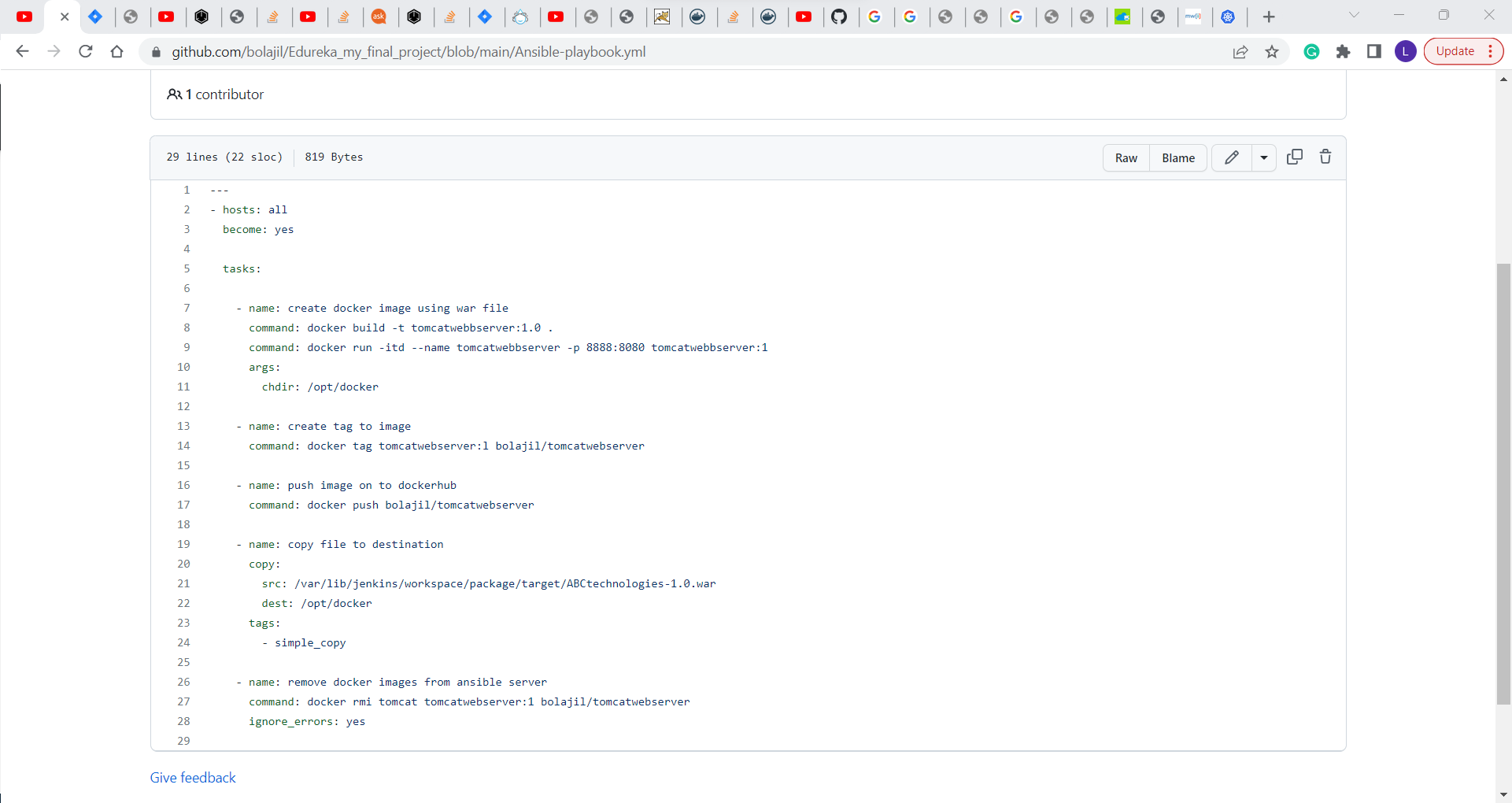
Create CI/CD job on Jenkins to build and deploy on a container. 1. Enhance the packaging job created in step 1 of task 2 to create a docker image. 2. In the Docker image, add code to move the war file to the Tomcat server and build the image.

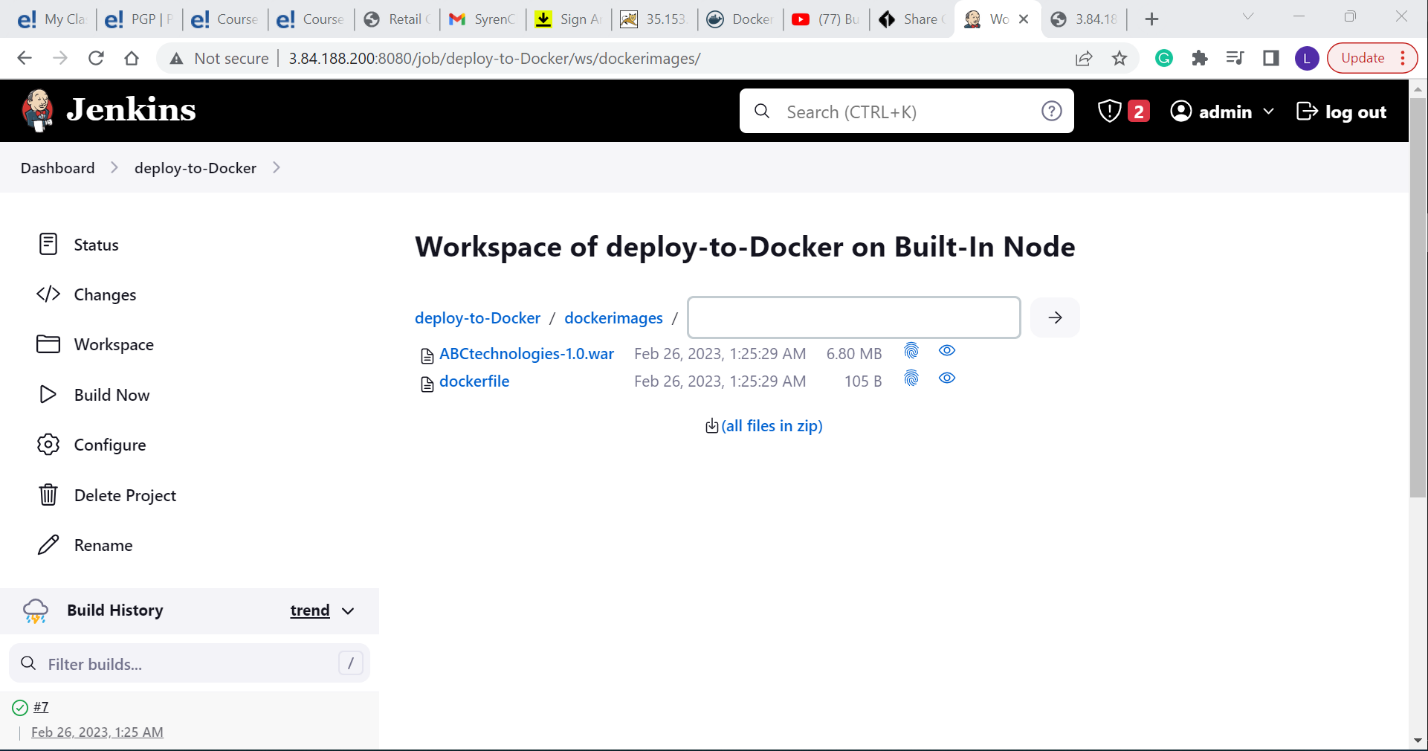


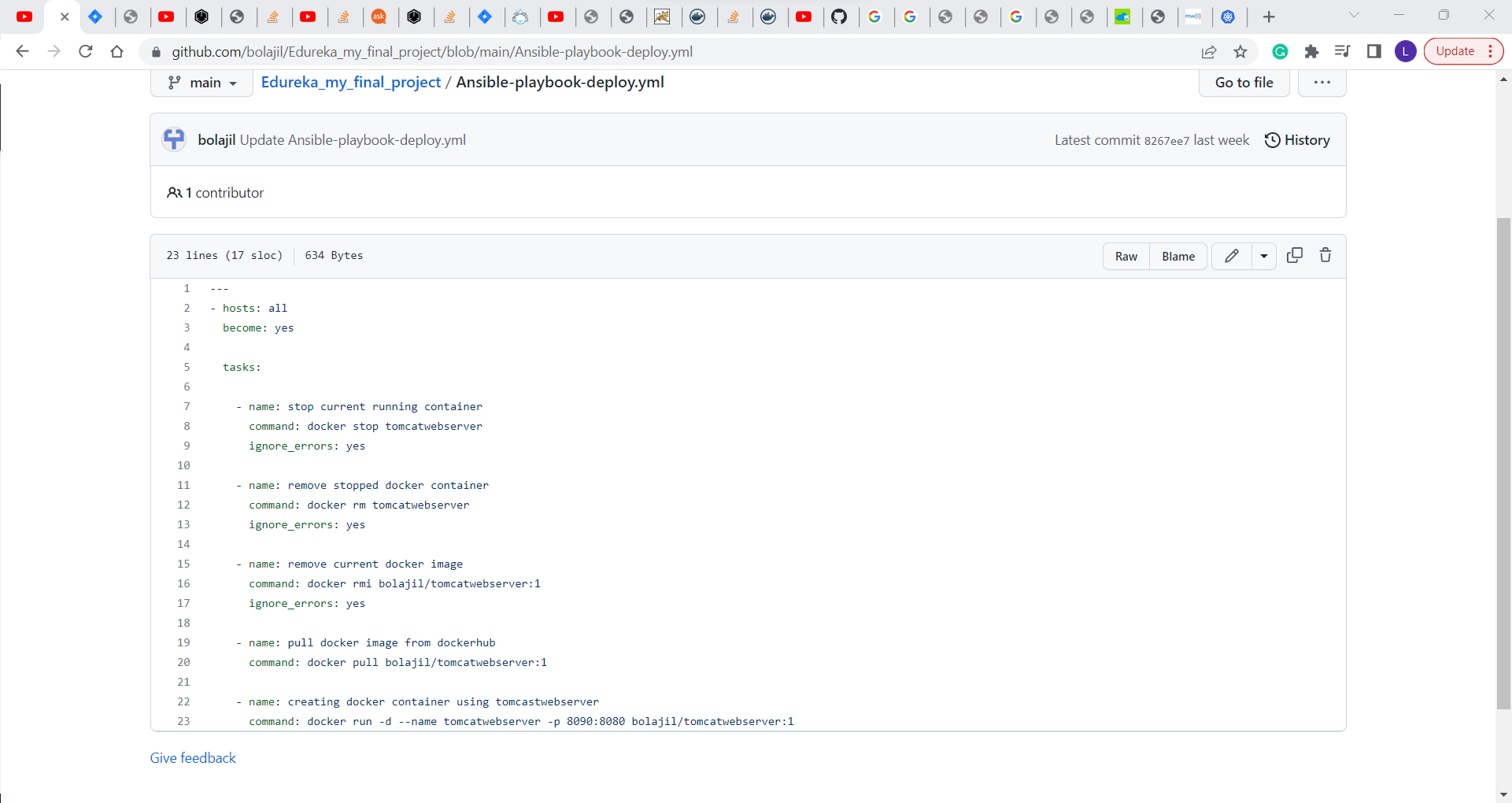




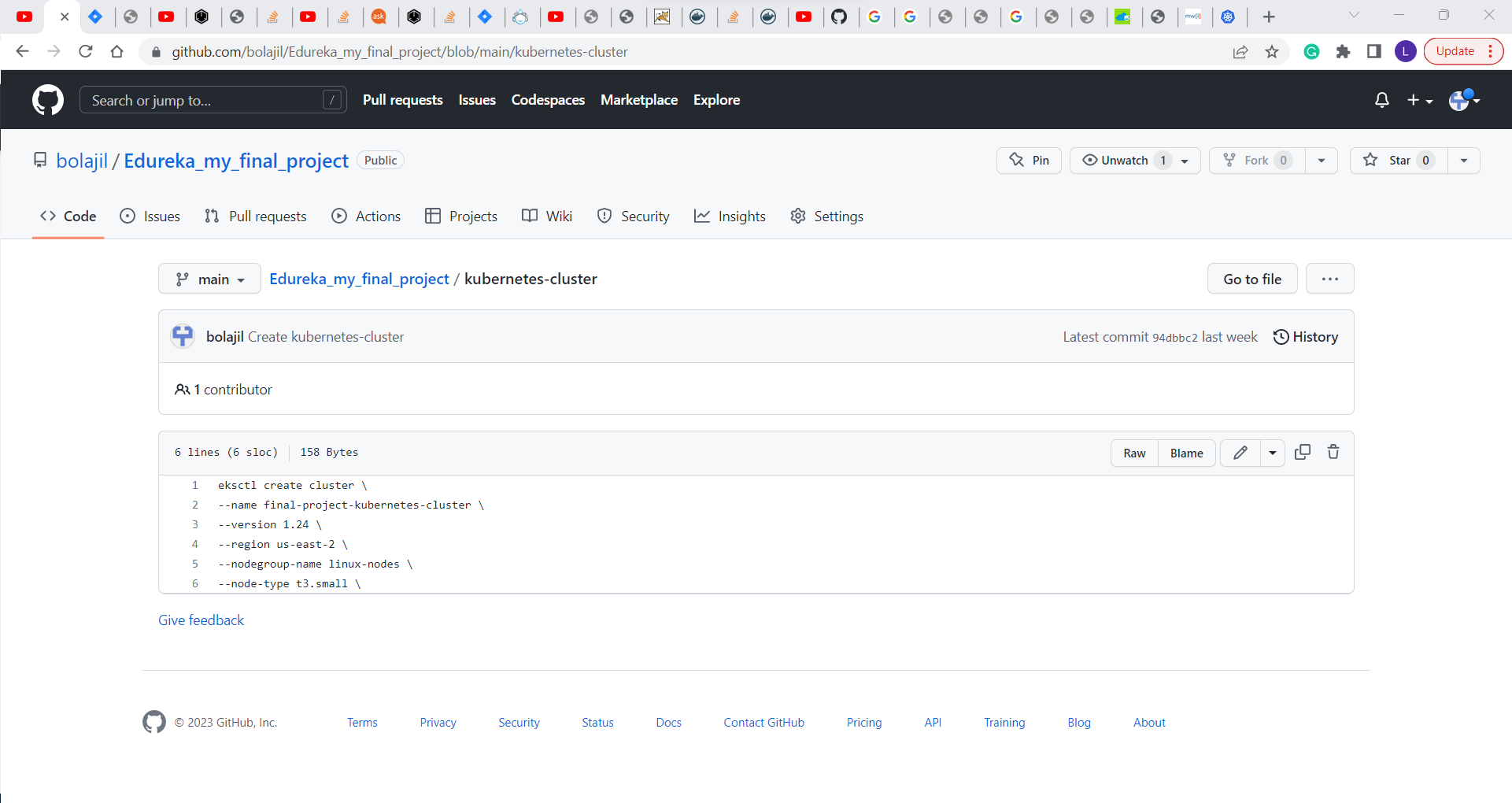
Task 4: Integrate the Docker host with Ansible. Write an Ansible playbook to create an image and create a continuer. Integrate Ansible with Jenkins. Deploy Ansible-playbook. CI/CD job to build code on ansible and deploy it on the docker container

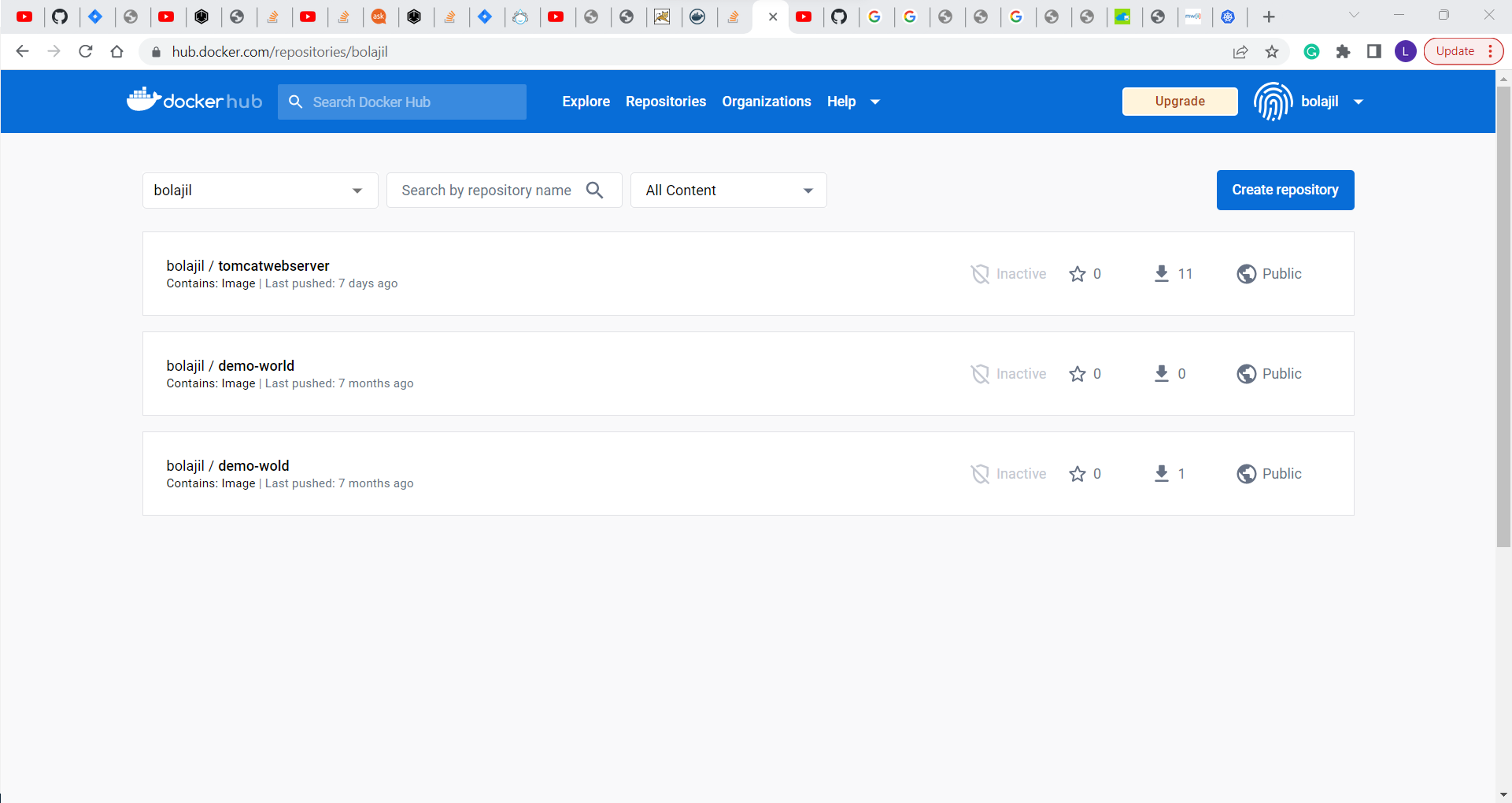


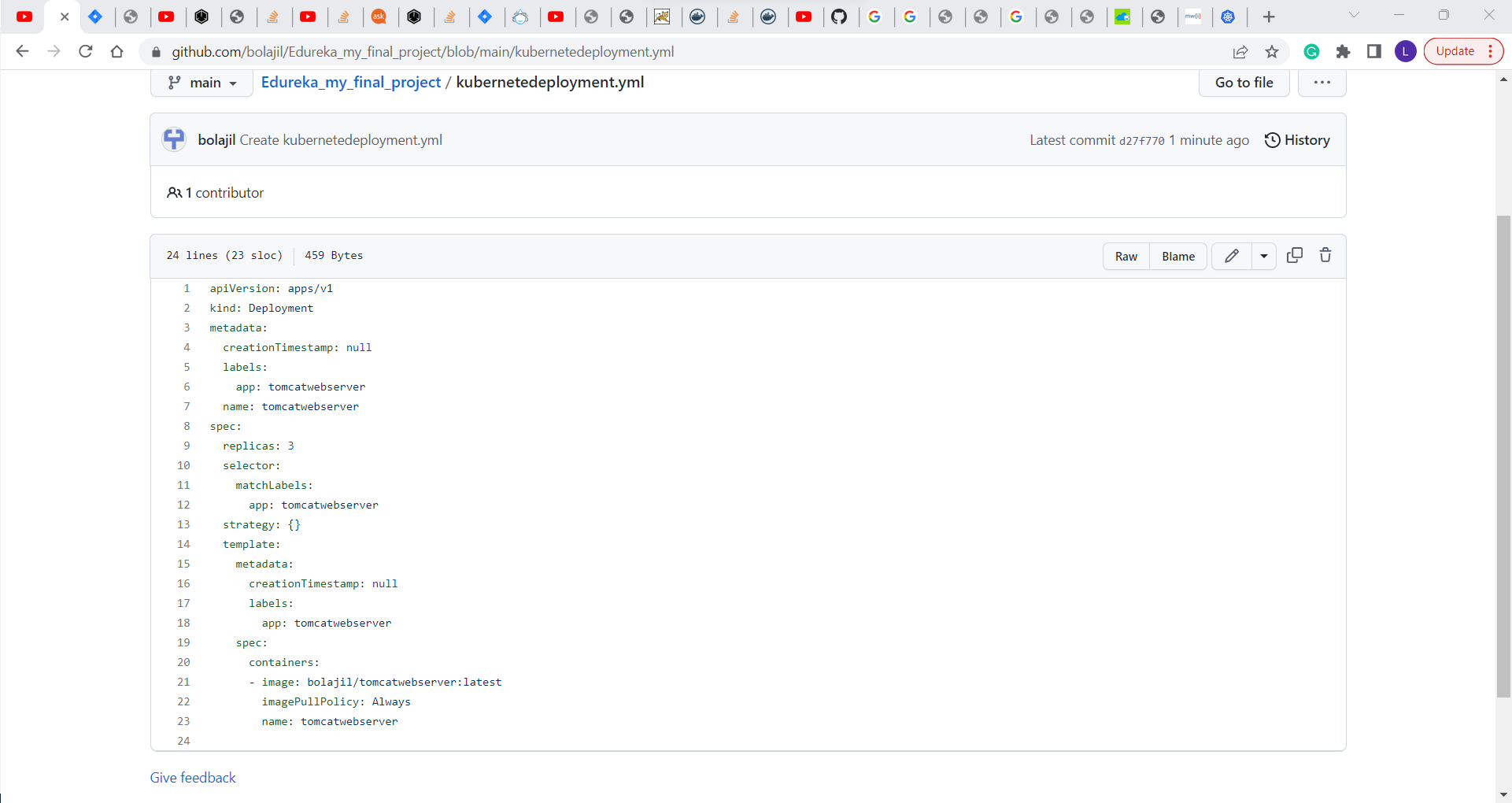


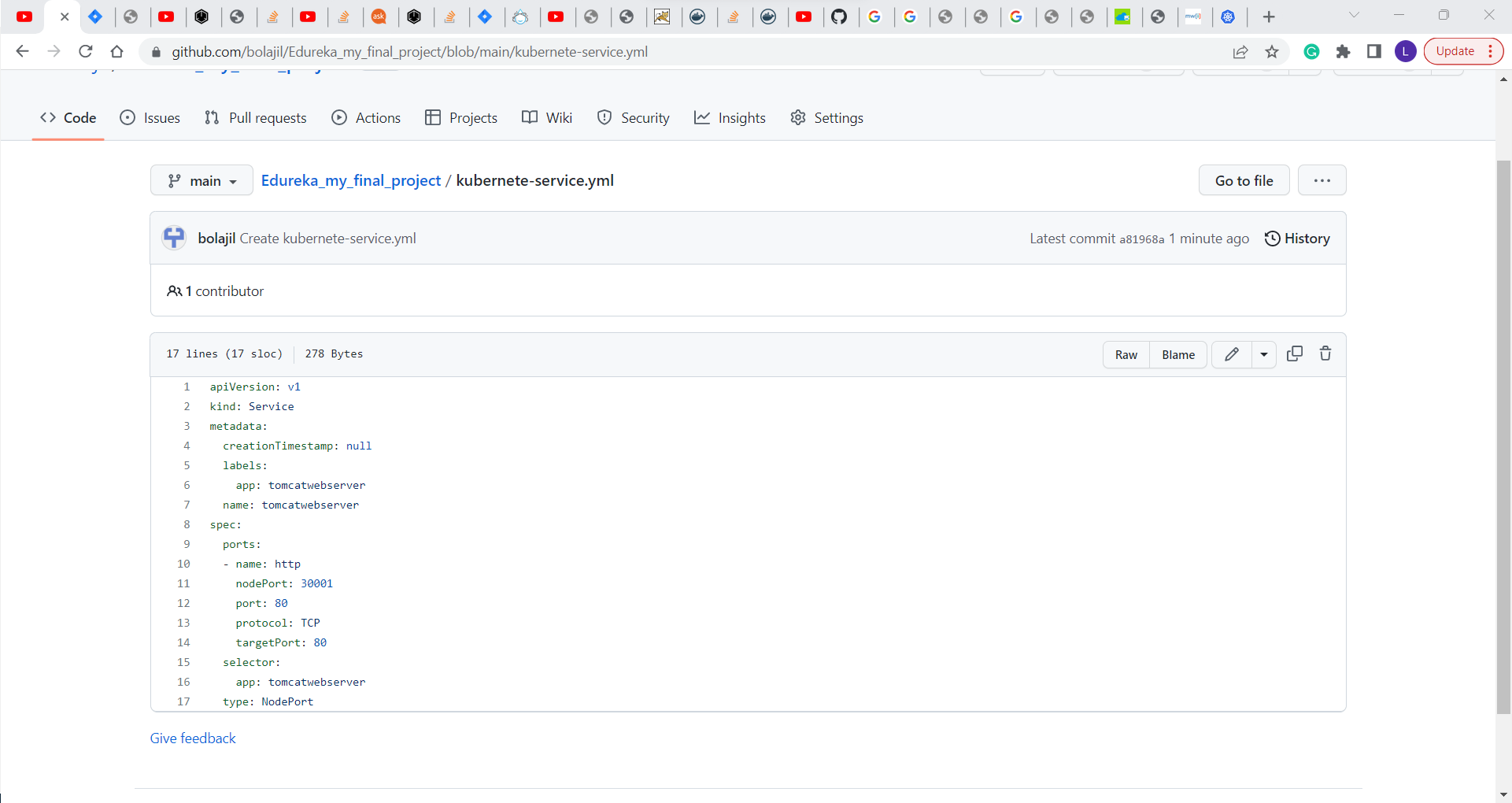


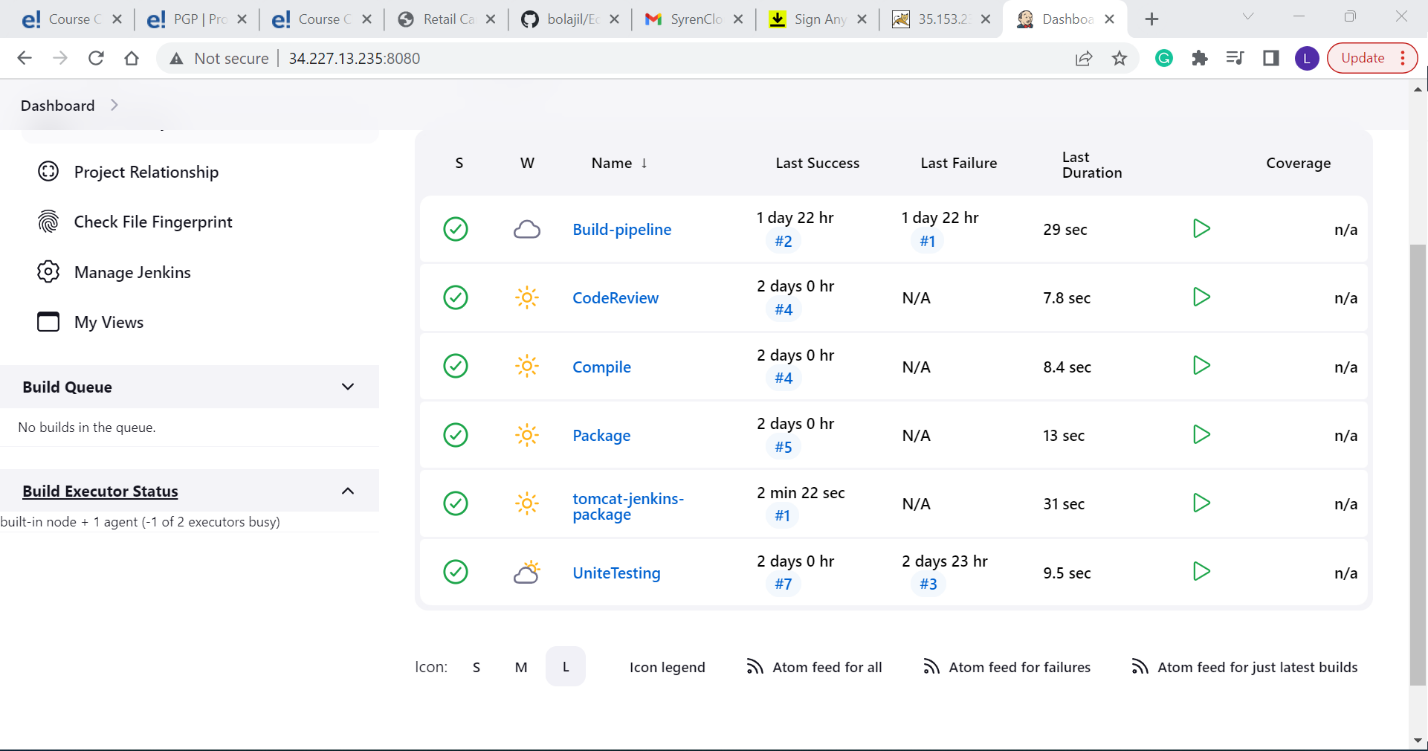
1. Deploy Artifacts on Kubernetes 2. Write pod, service, and deployment manifest files 3. Integrate Kubernetes with Ansible 4. Ansible playbook to create deployment and services.

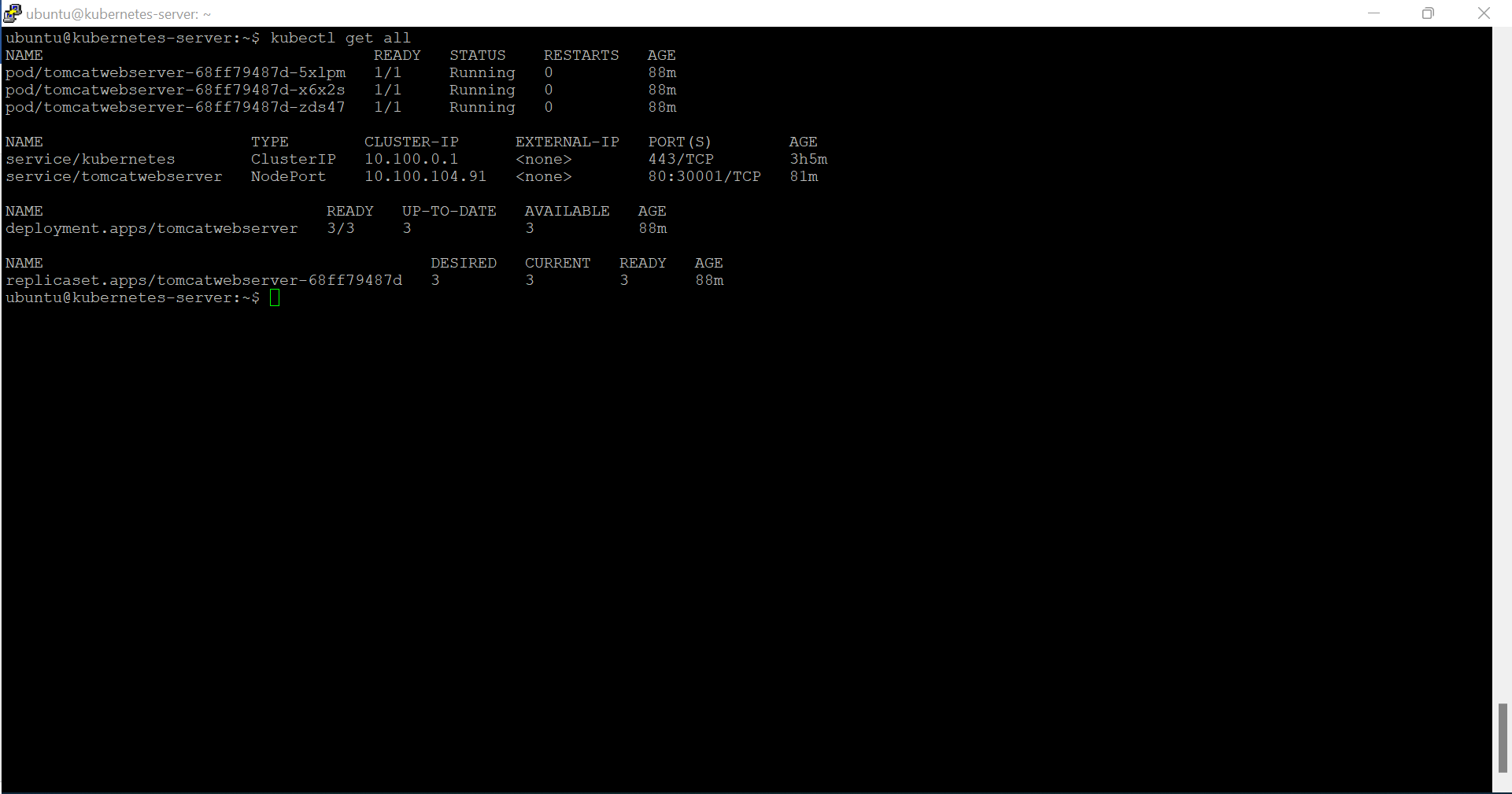
****

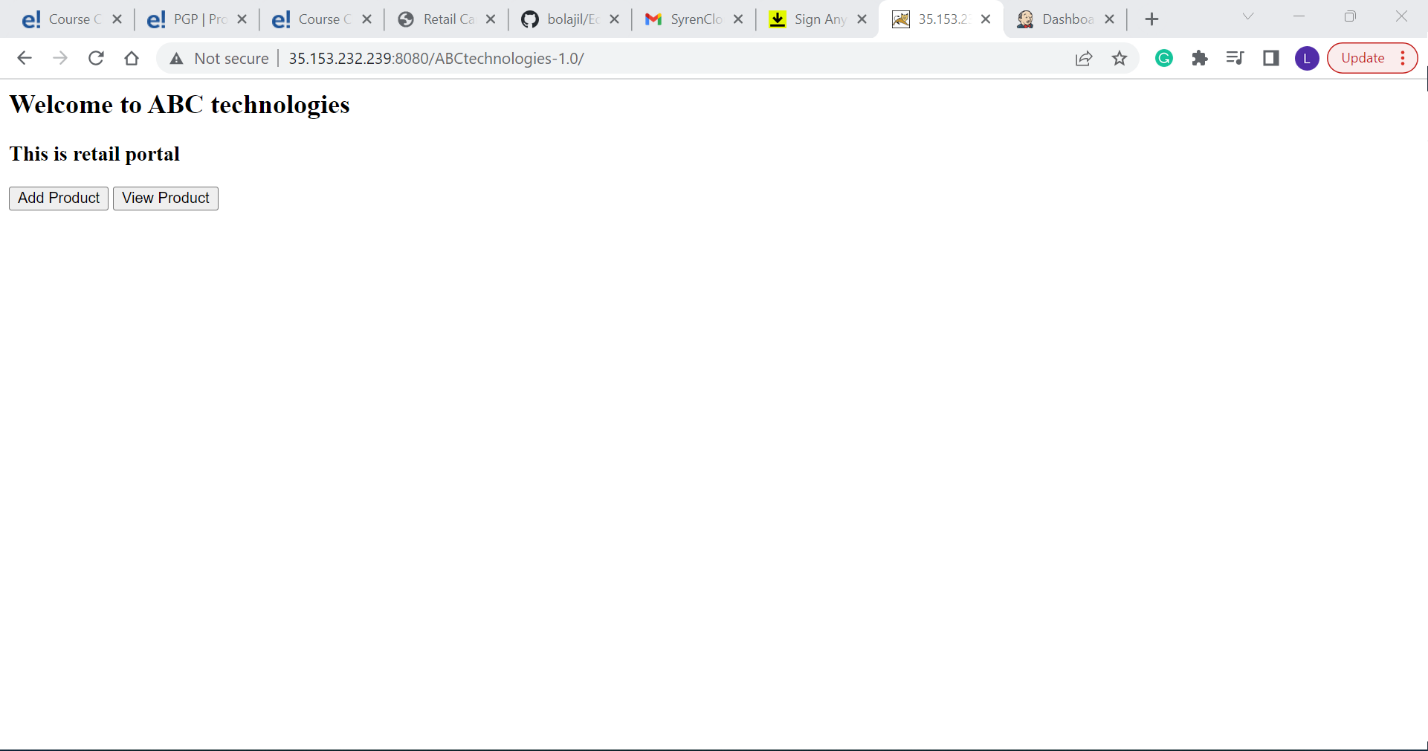
****

****

****

****

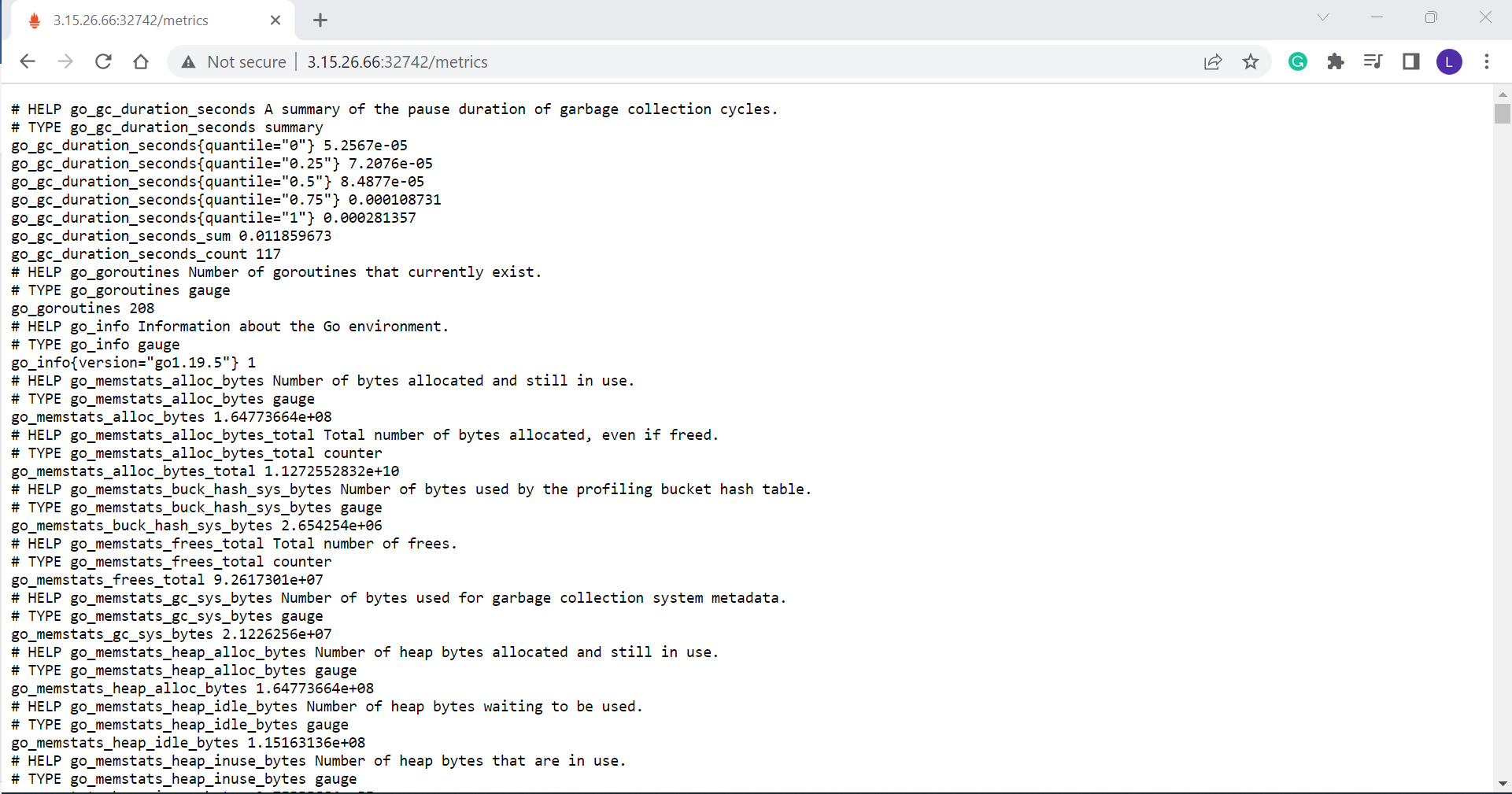
****

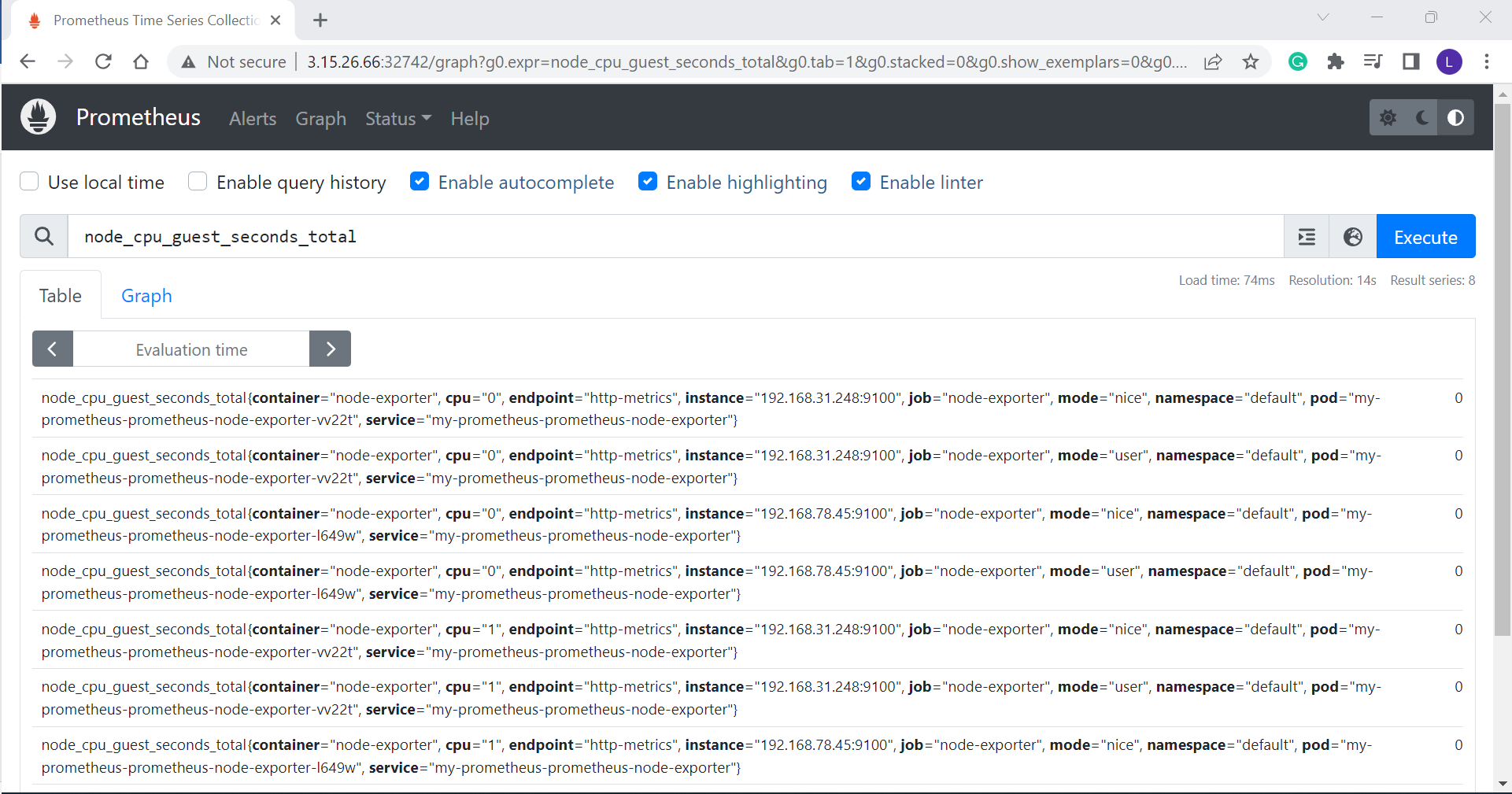
****

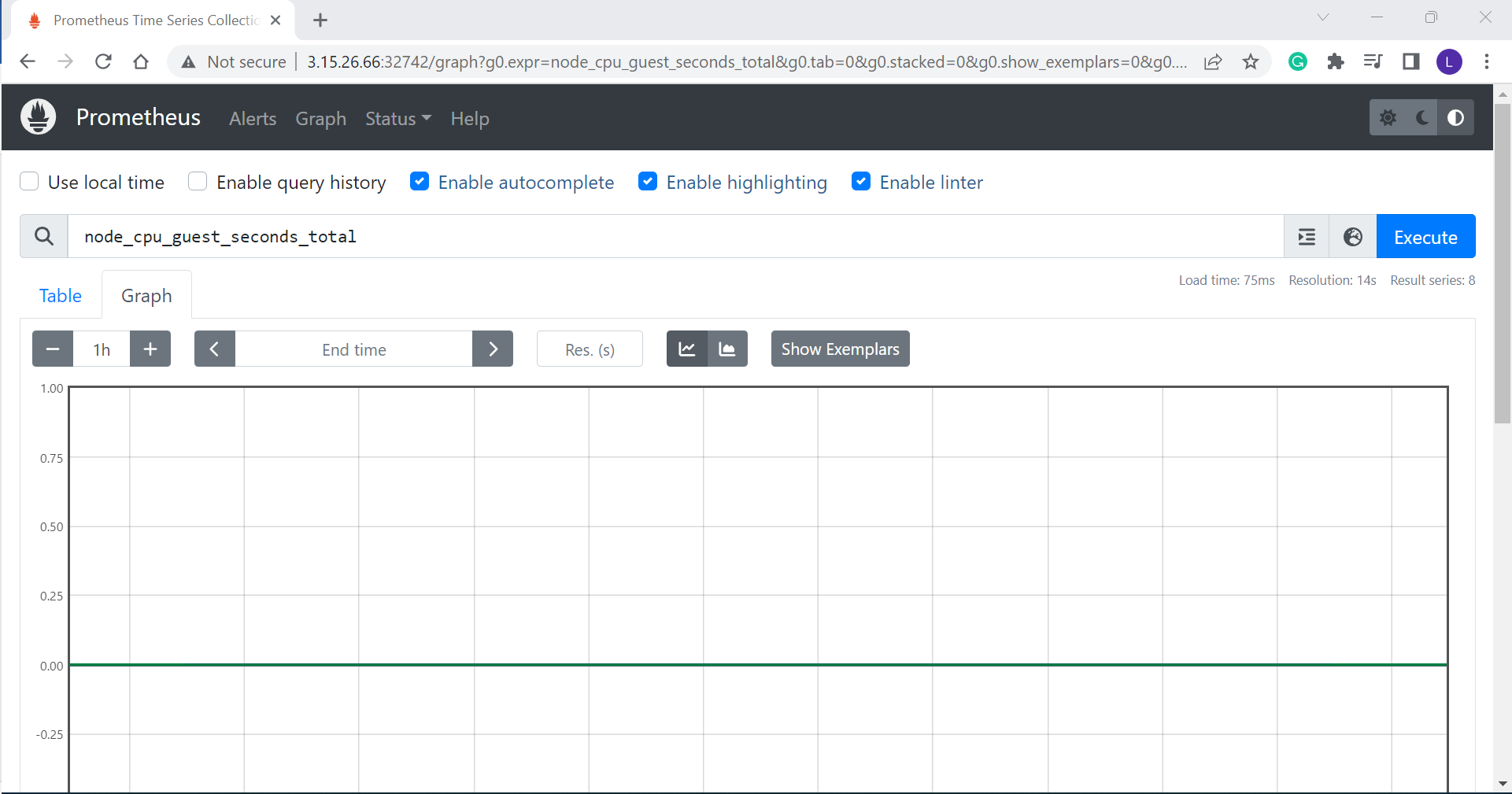
Using Prometheus, monitor the resources like CPU utilization: Total Usage, Usage per core, usage breakdown, memory, and network on the instance by providing the endpoints on the localhost. Install the node exporter and add the URL to the target in Prometheus. Using this data, log in to Grafana and create a dashboard to show the metrics.

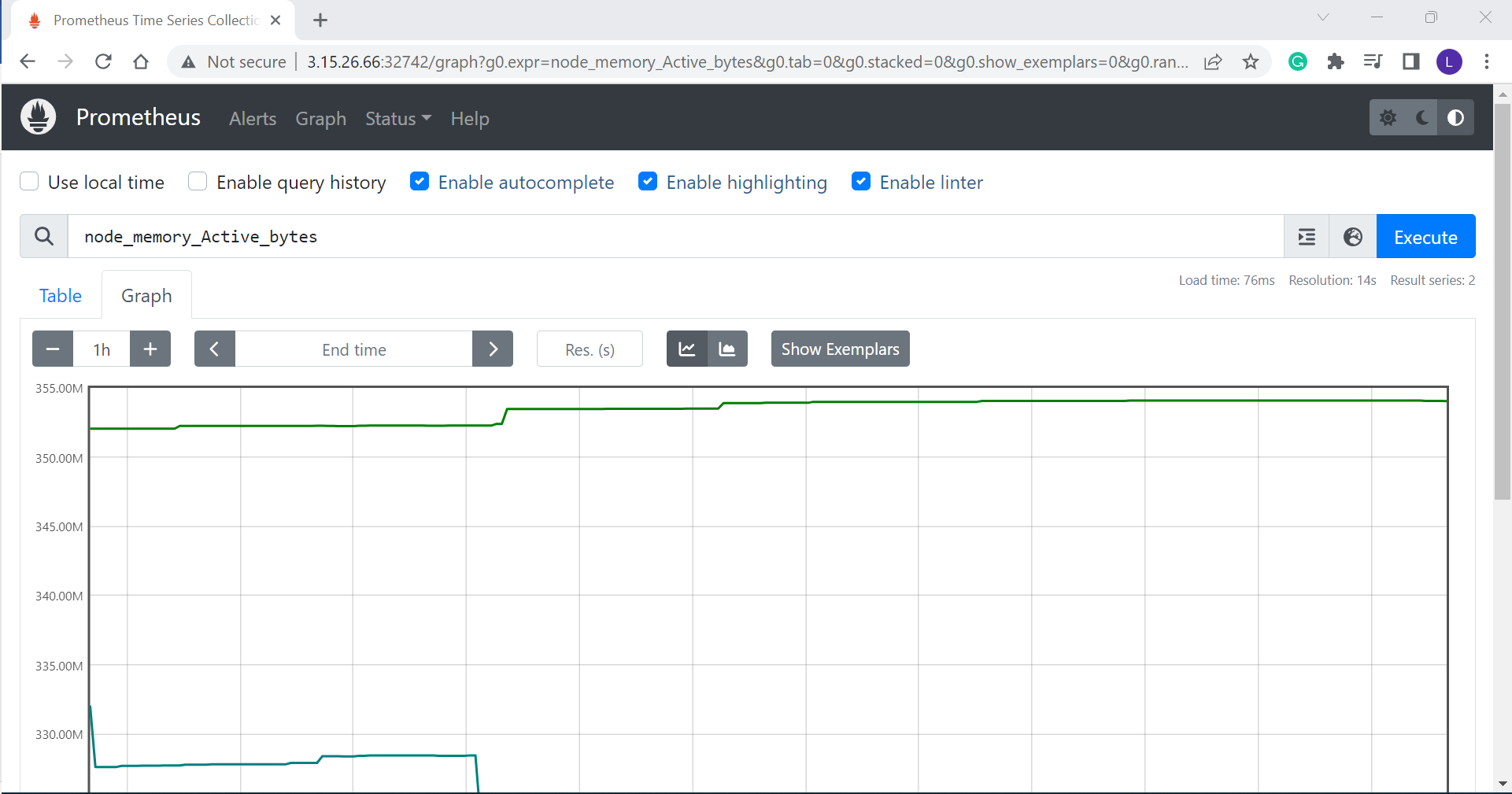
****

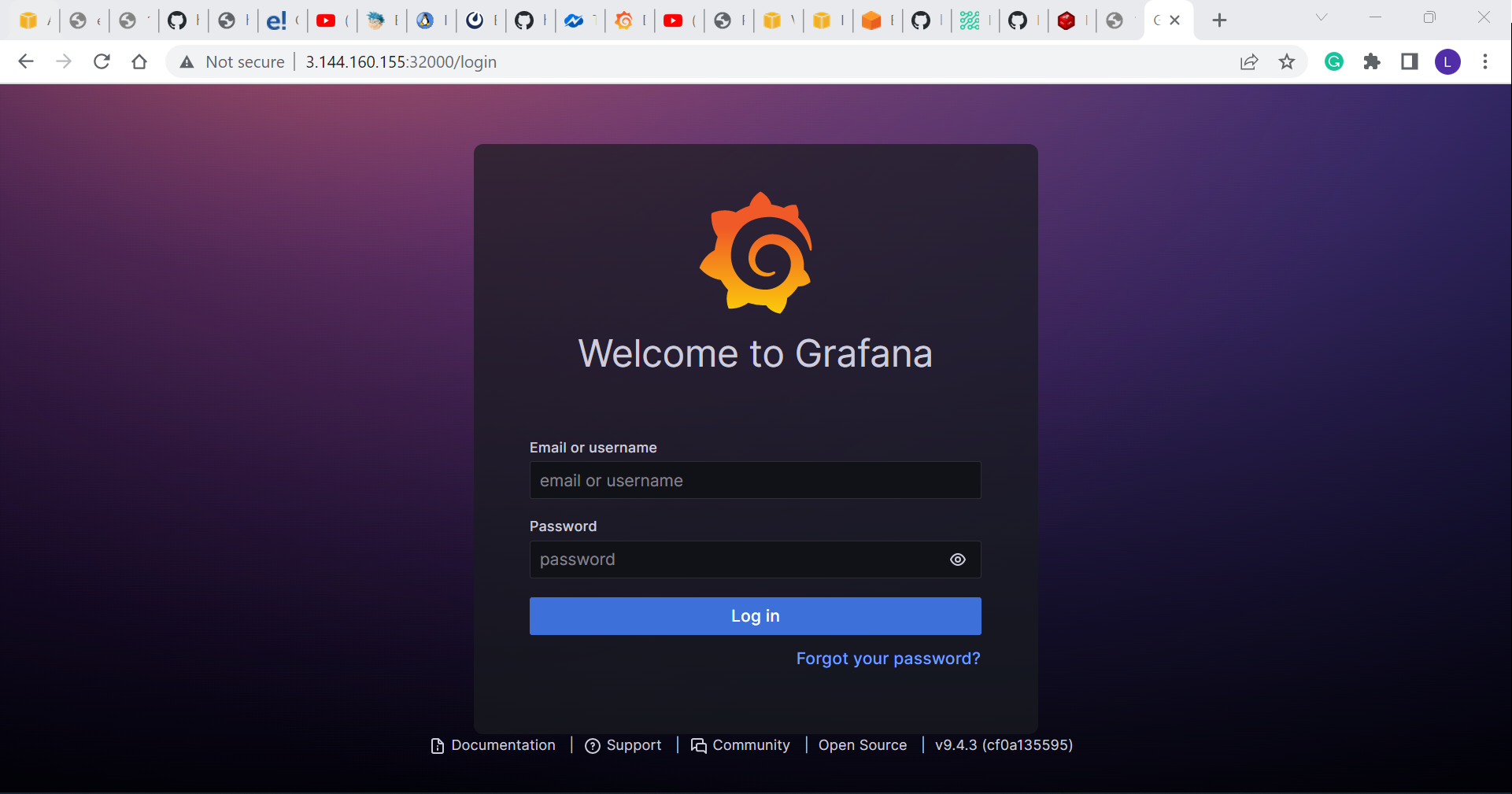
**Metrics output:**

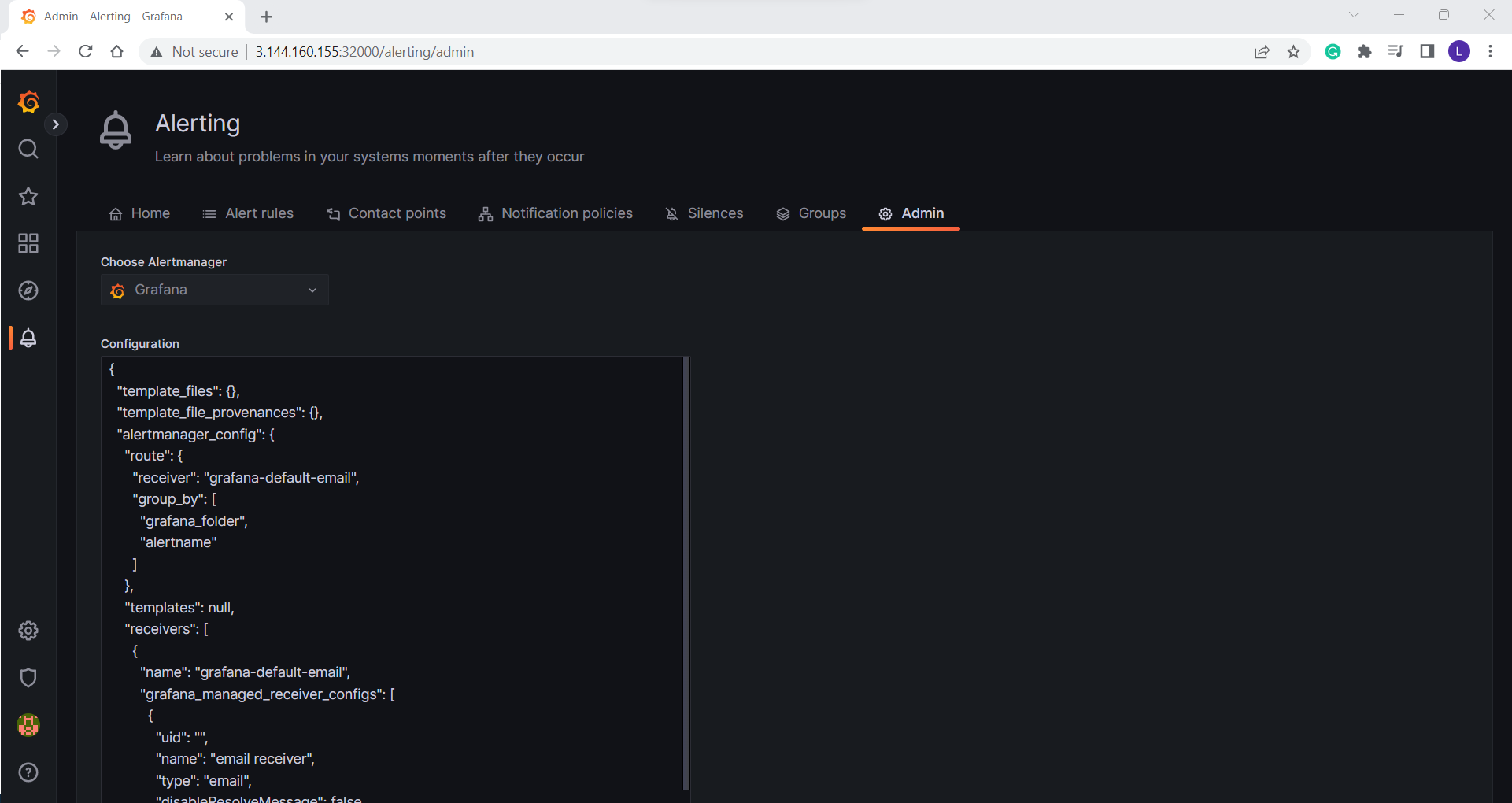
****

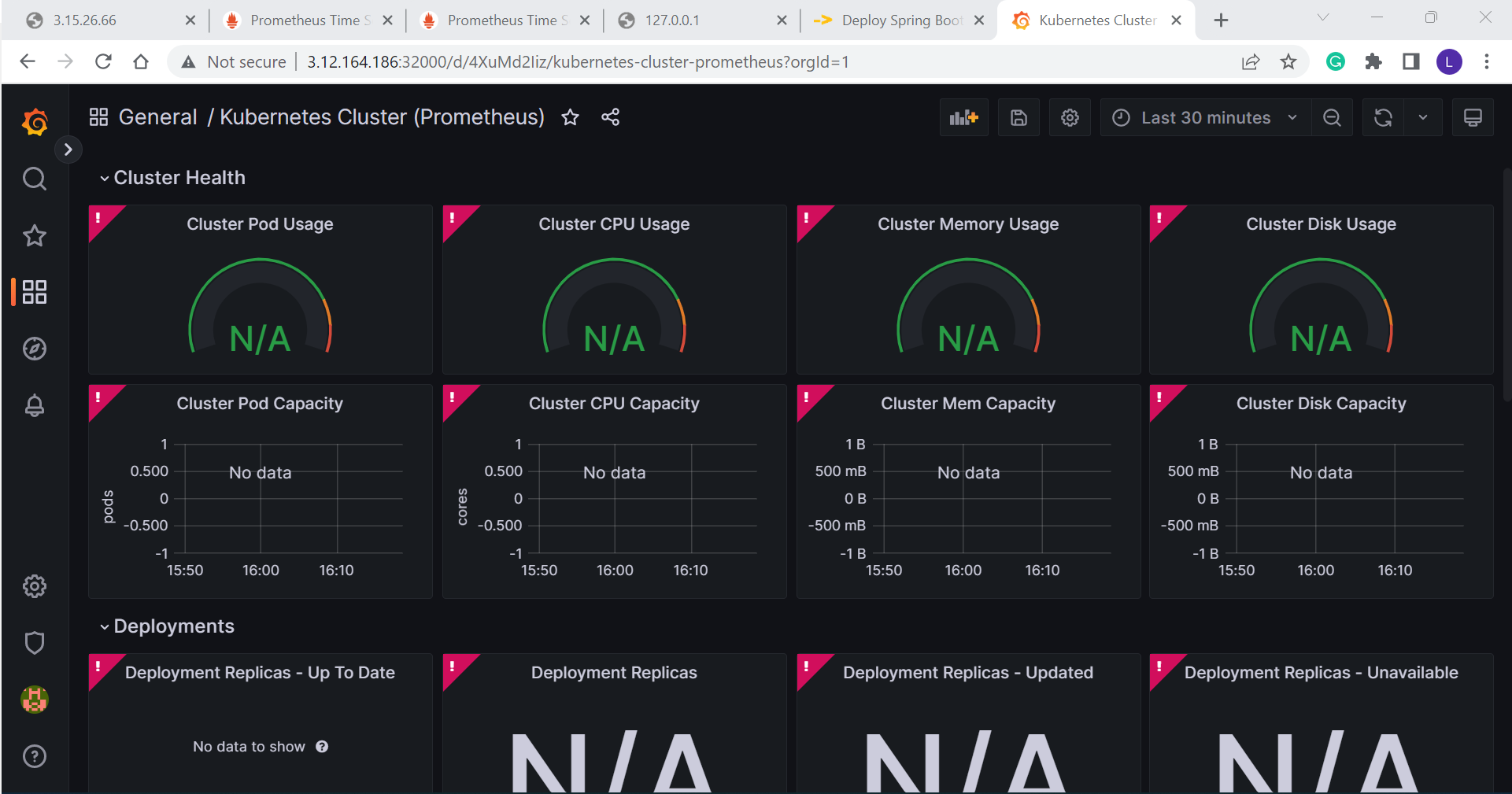
****

****

****

****

****

****

**Additional alert rules can be created to view more resources as needed. This concludes the**

After the solution is built, the business will have highly available service, high scalability, highly performant, easily built and maintained, developed and deployed quickly, Lower production bugs, Frequent releases, better customer experiences, Lesser time to market.