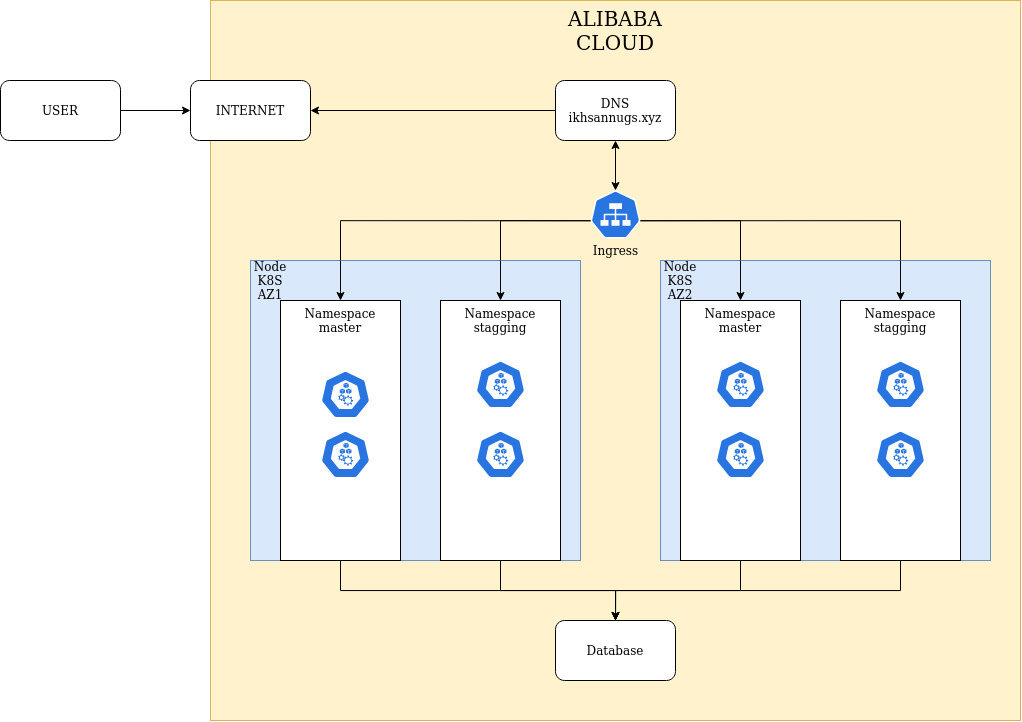
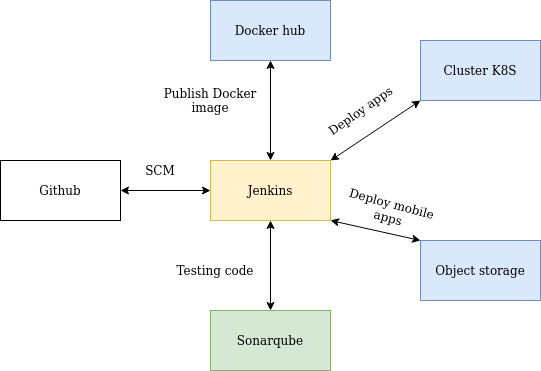
**KDS DevOps Practical Test**

  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
The architecture above uses alibaba cloud's kubernetes service. which there is a cluster type kubernetes master managed. so we don't have to worry about the availability of the master node.

Then in the architecture above I made two environtment. i.e. environtment master as production and environtment stagging as pre production. in that cluster I use ingress as an incoming traffic manager for users who want to access the applications inside our cluster.

in this cluster I do not use persistent volumes because the contents of this cluster no one needs a persistent volume to store the data because in the application indeed the data does not grow. for my database I do not put in the cluster because it will risk losing data according to some people. so that for the database I run in a regular instance.

  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
This is the environtment used for flow cicd:

1. **Git** for source code management

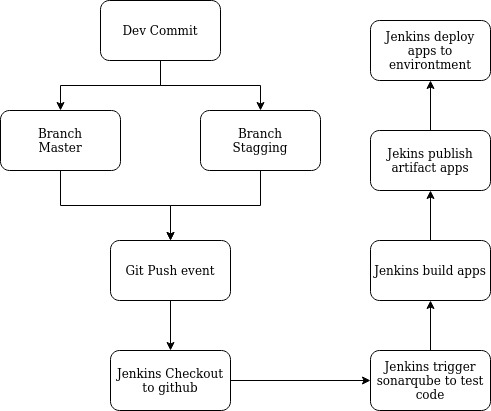
2. **Jenkins** for ci/cd tools

3. **Docker** hub to store artifact docker image

4. **Sonarqube** for testing code

5. **Object storage** for storing mobile application artifacts

6. **Kubernetes** cluster for deployment purposes

  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
This is the flow for ci/cd:

1. Dev team will commit the code it creates to the branch whether it is master or stagging

2. Github will push the event to the previously installed webhook jenkins.

3. Jenkins will checkout the git repository that we registered earlier in the job jenkins.

4. Jenkins will then scan the code using the sonar scanner plugin and the plugin will push the task to the environtment sonarqube server.

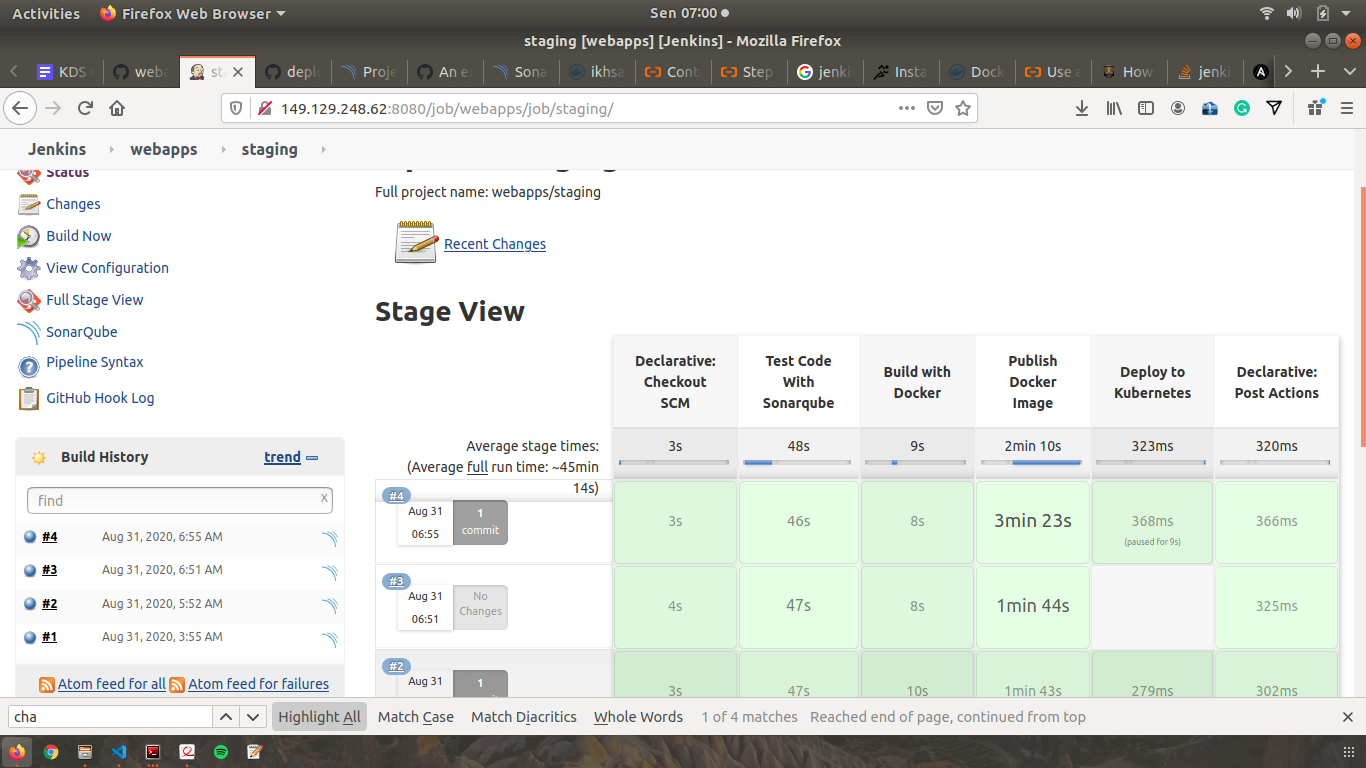
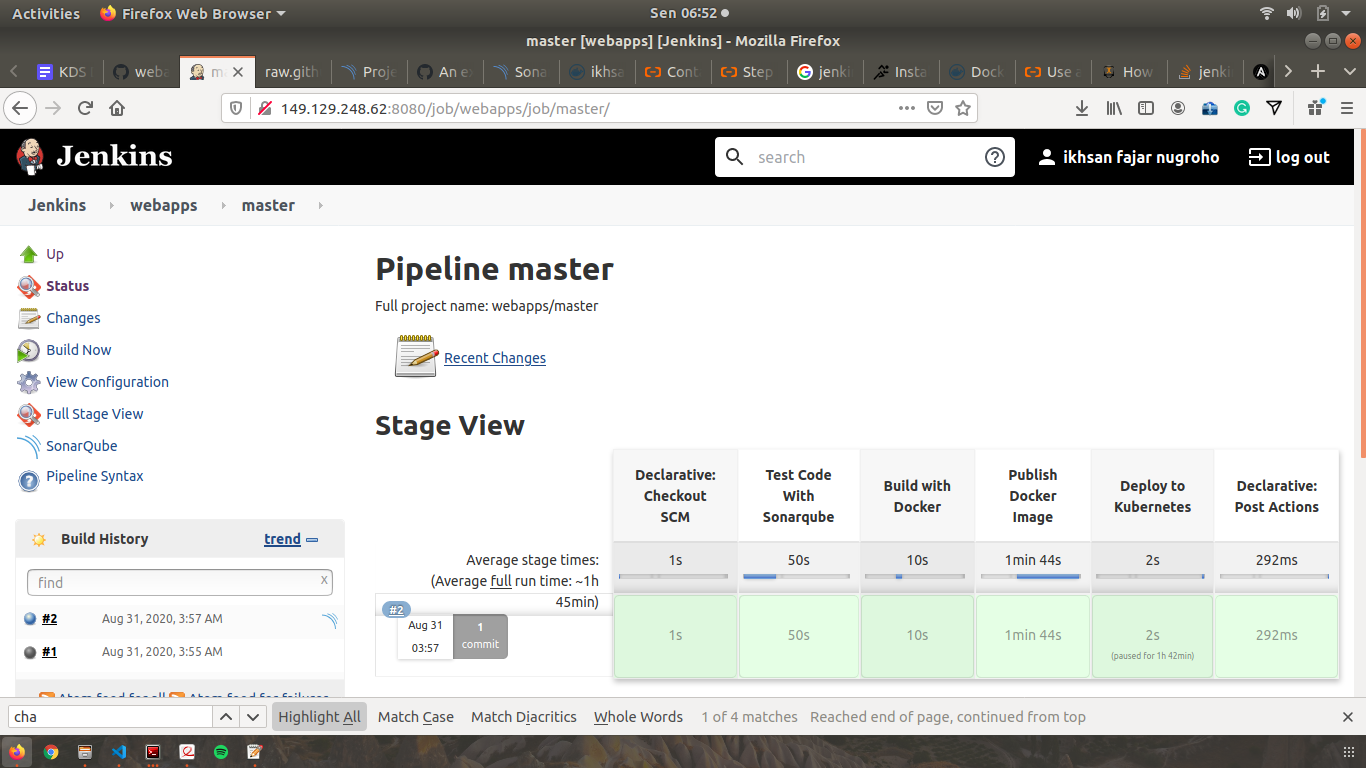
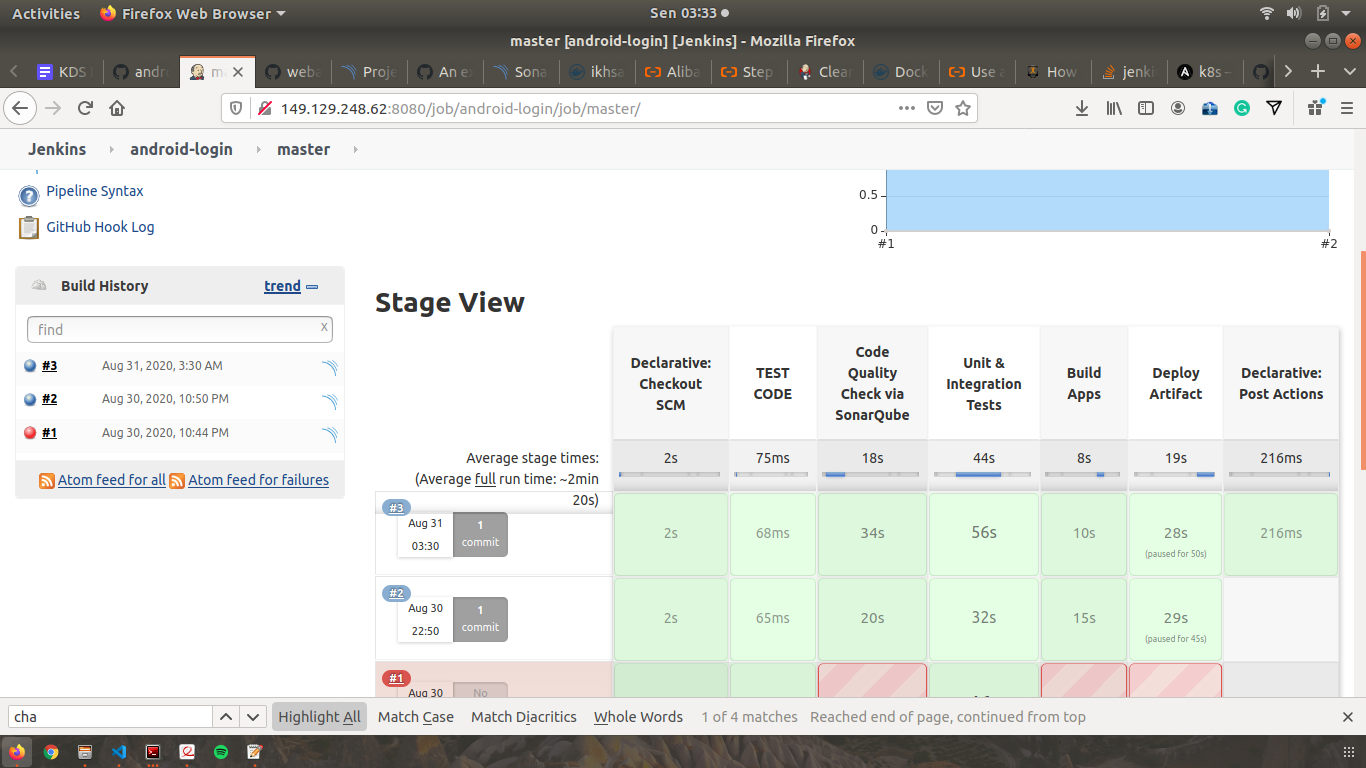
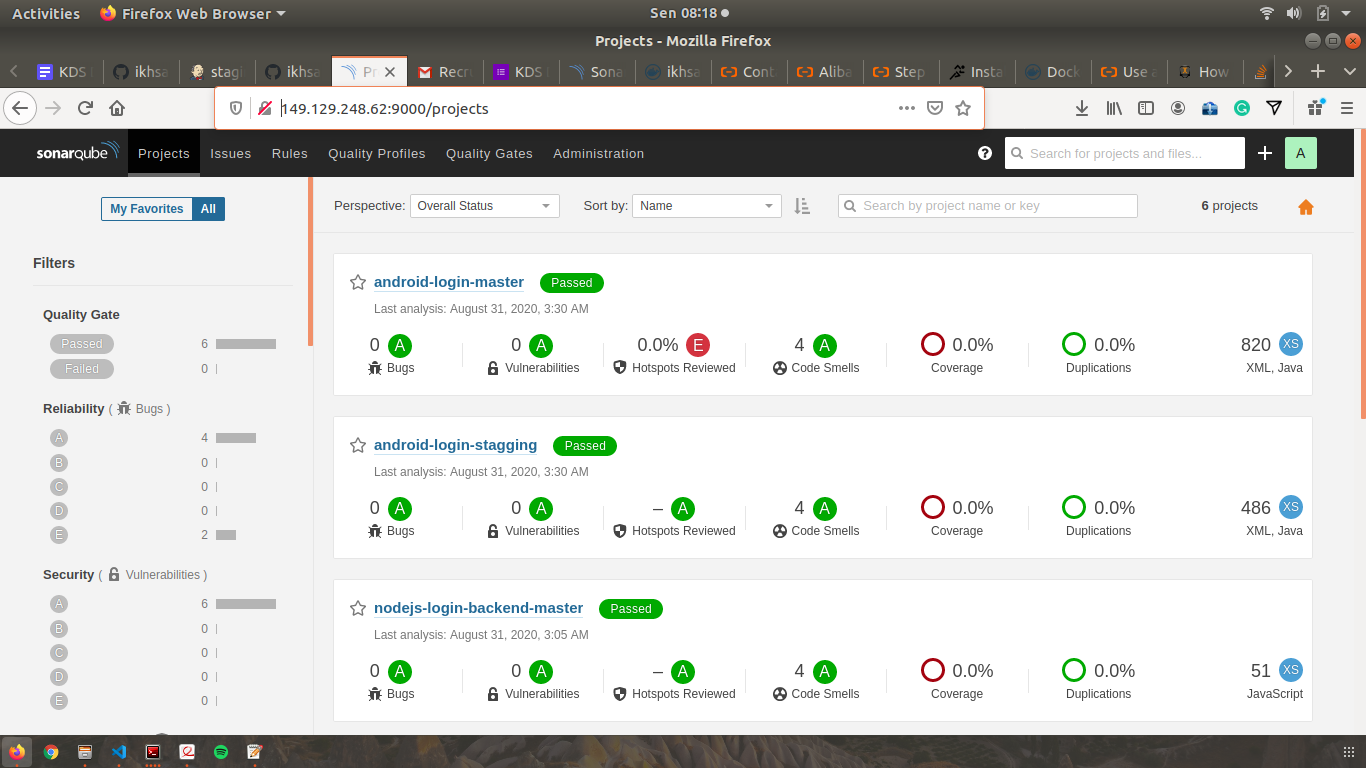
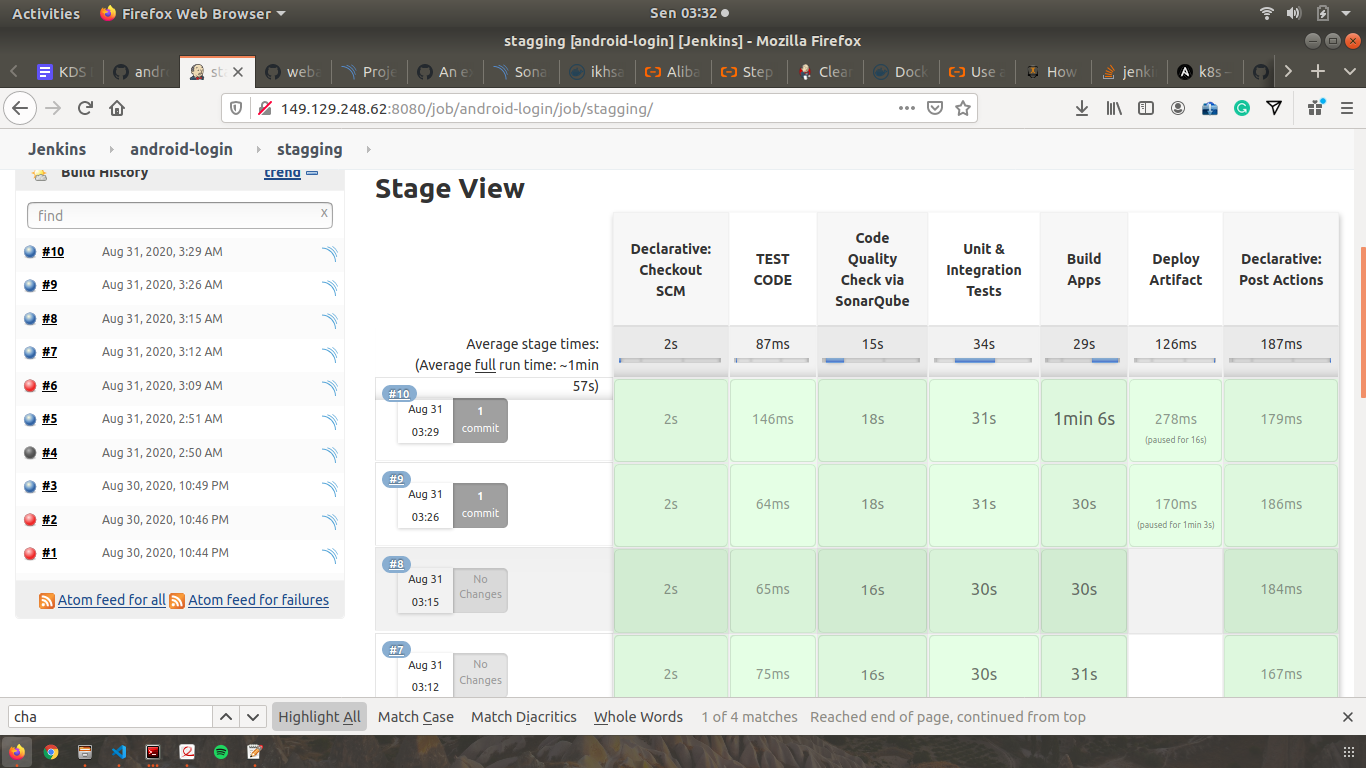
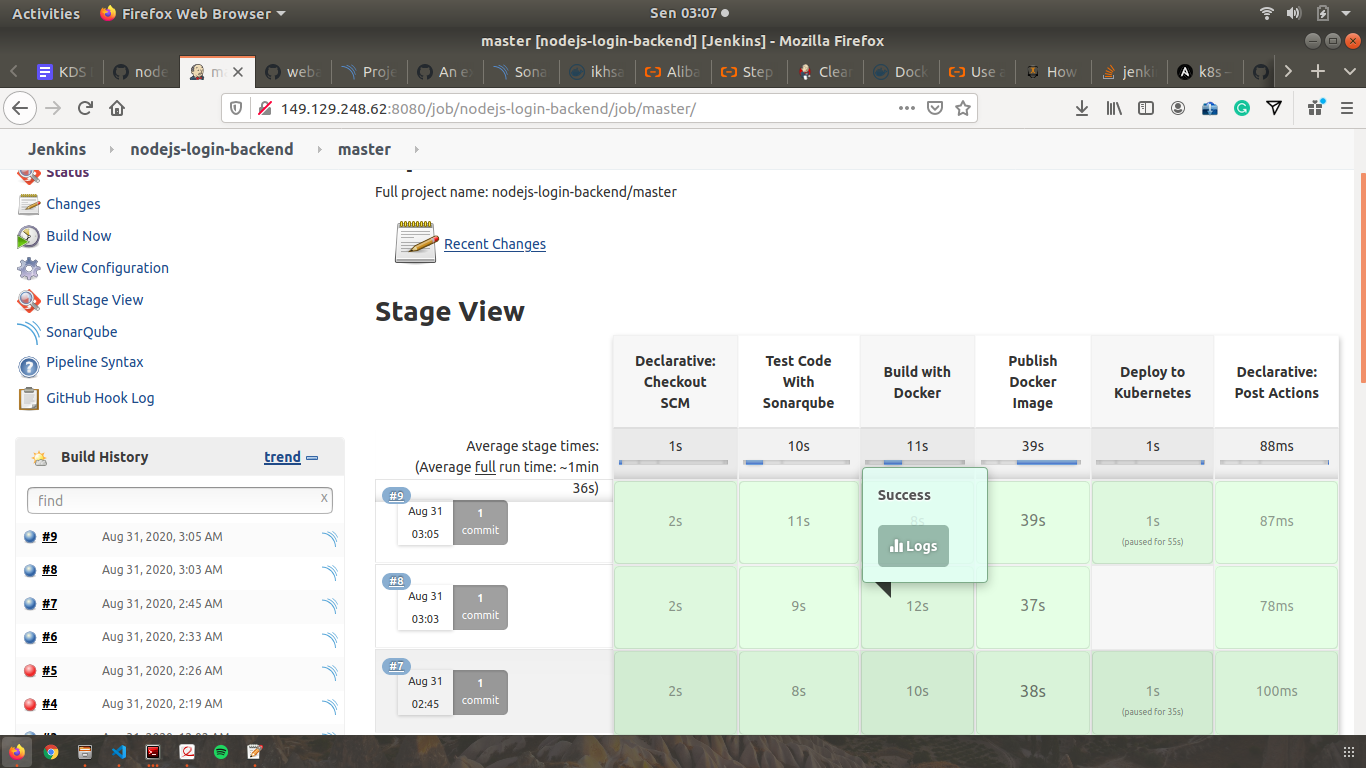
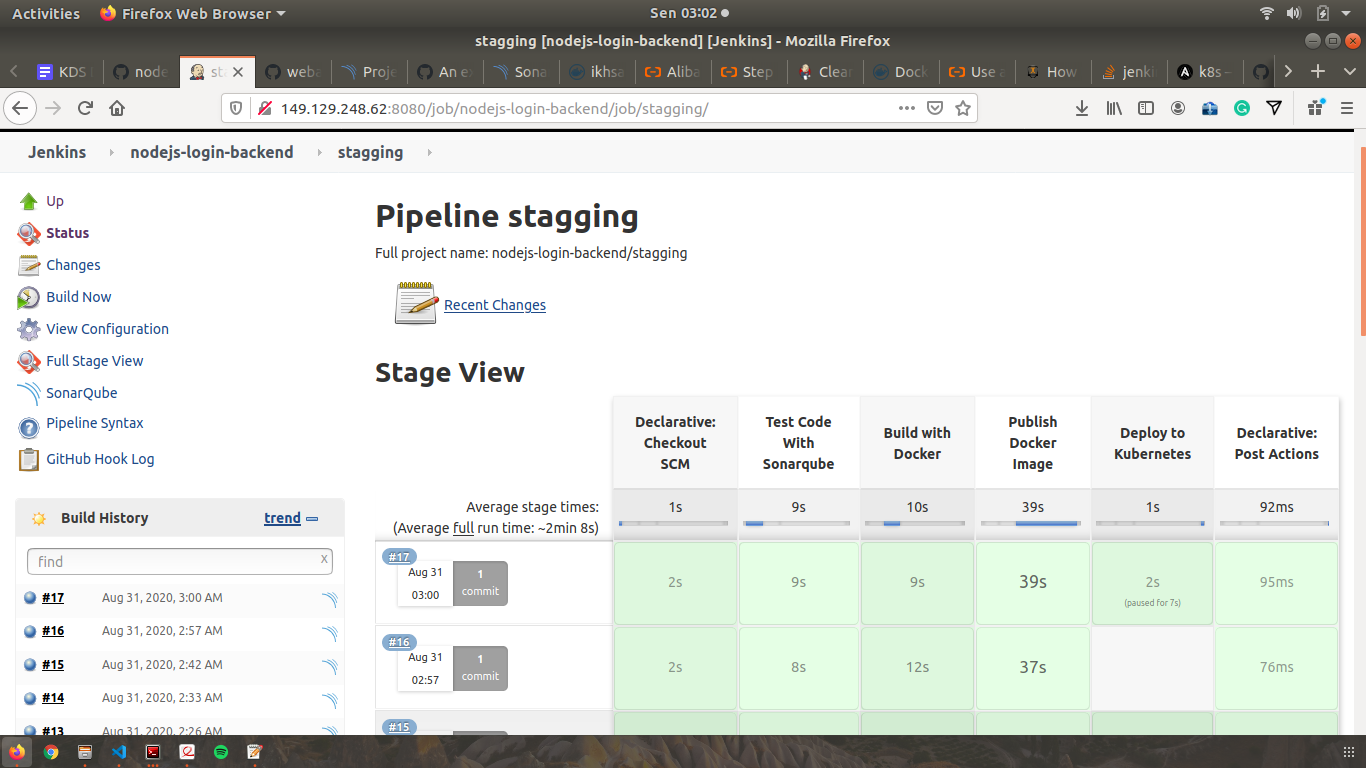
5. If the code passes the scan, jenkins will proceed to the build stage, otherwise jenkins will stop the flow.

6. Next to the build stage. jenkins will build based on what we defined in the previous jenkinsfile.

7. Jenkins will publish artifact build results to other environtment. for example, if the application will use docker, jenkins will publish artifacts to the docker repository.

8. Last jenkins will perform deployment. but before the deployment is done. jenkins will check the commit message. if the message commit is the same as "deployment" then jenkins will deploy. but before that you have to click the button to deploy manually on the jenkins dashboard. that's to prevent deployments that we don't want.

9. Jenkins will clean up the files.

**CICD Documentation  
  
**