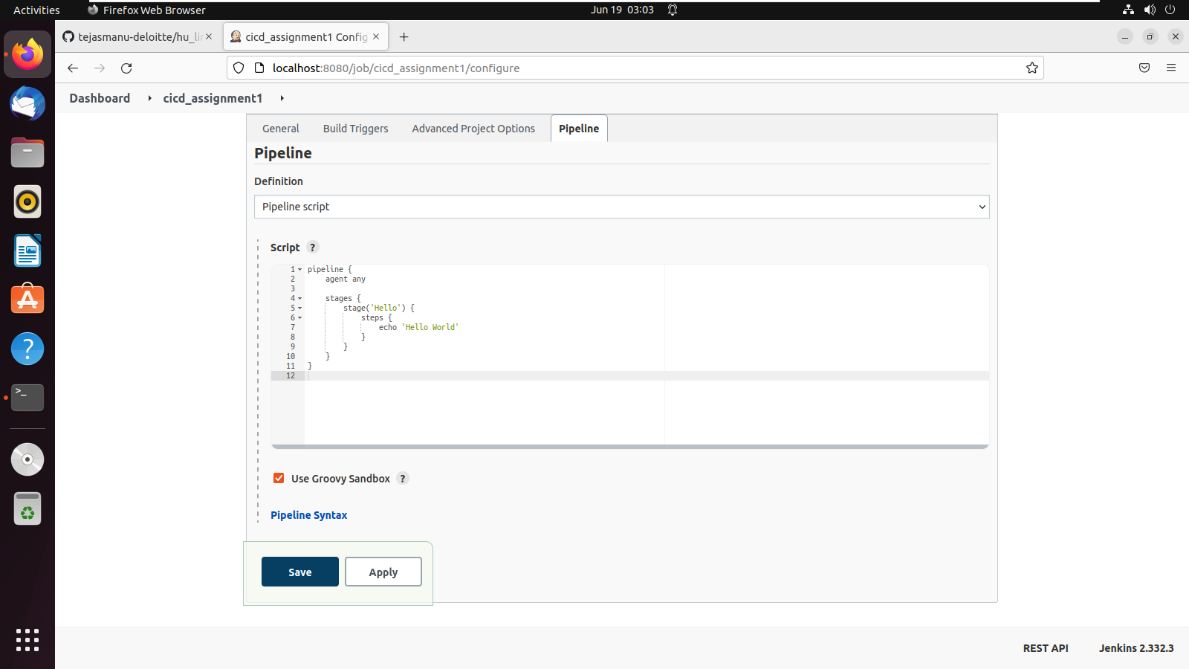
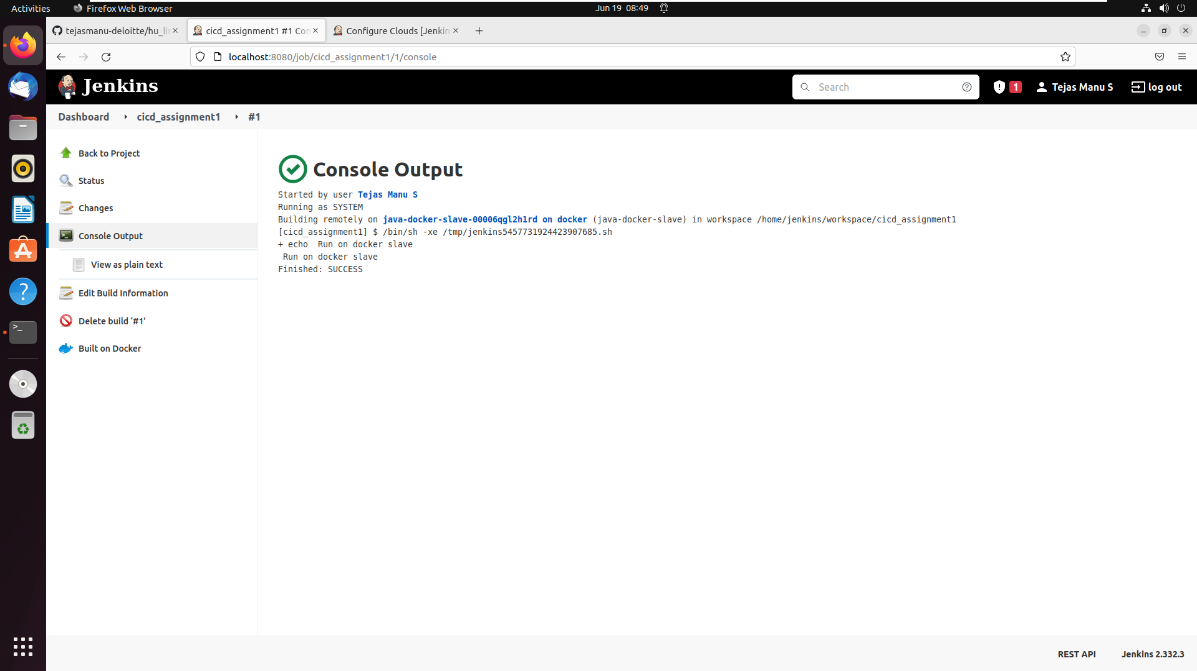
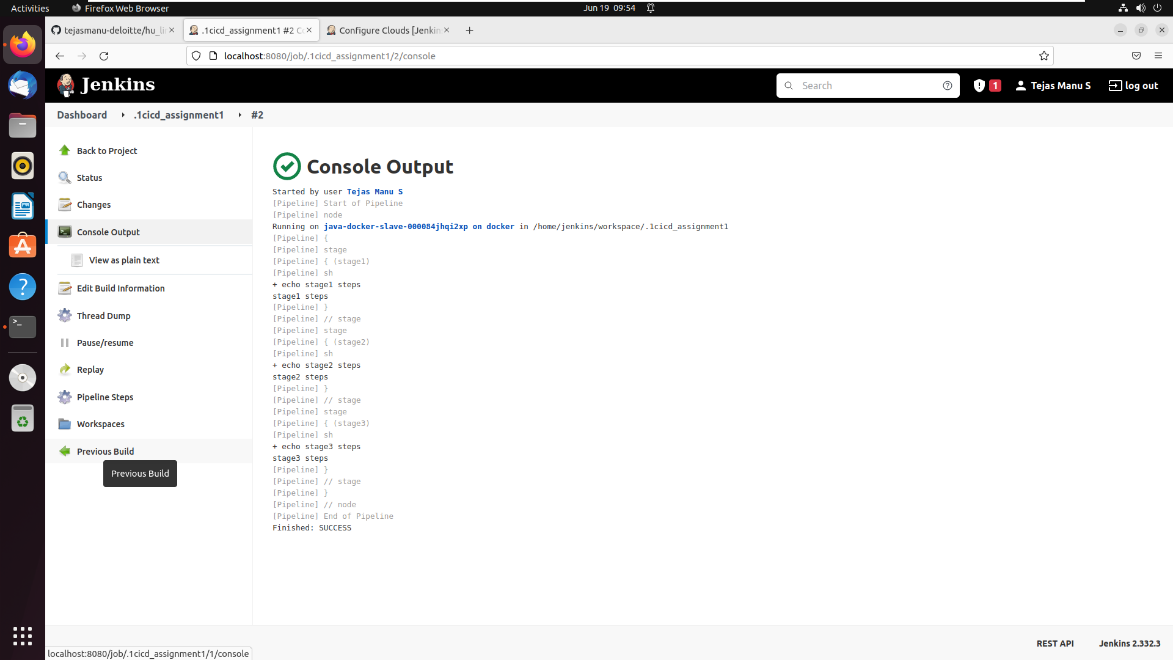
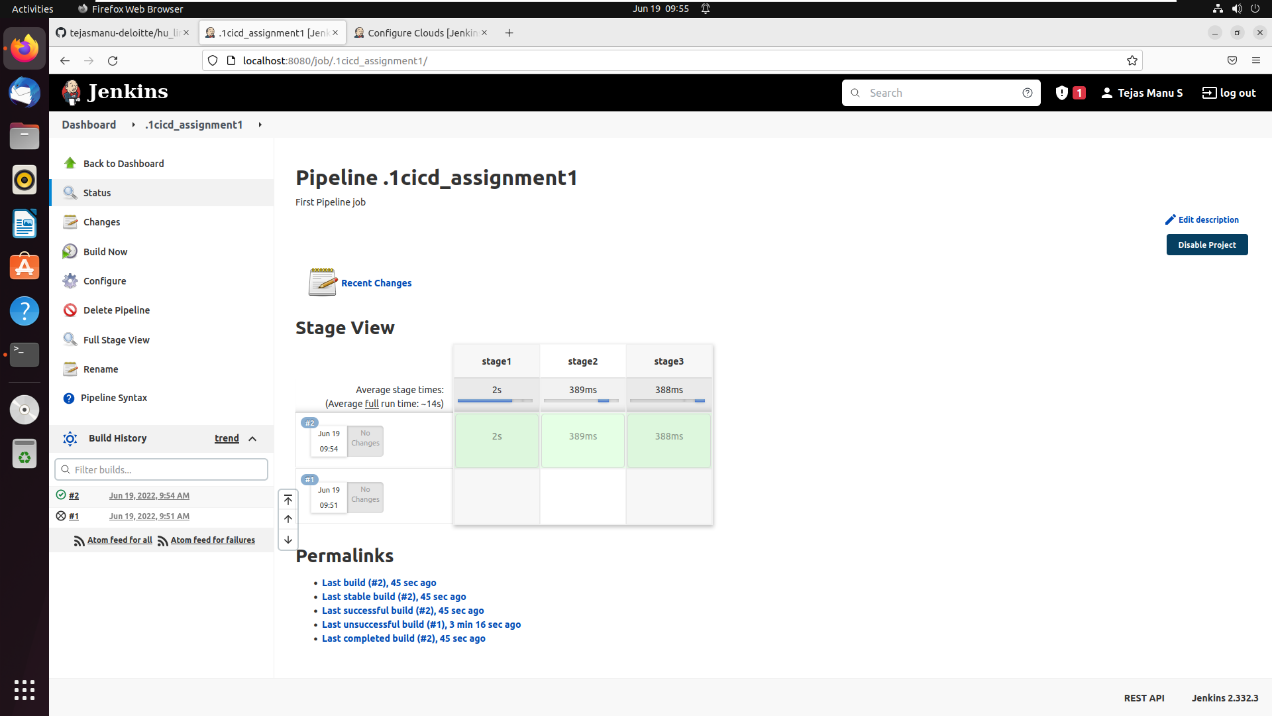
Jenkins Documentation

1. Create a master/slave configuration with docker containers.  
   - Install Docker on the workspace  
   - Install Jenkins on the workspace  
   - Use the docker plugin to setup slaves  
   - Create a docker image necessary for the docker slave and setup the docker slave using the plugin.  
   - Create a simple pipeline that executes on the slave.

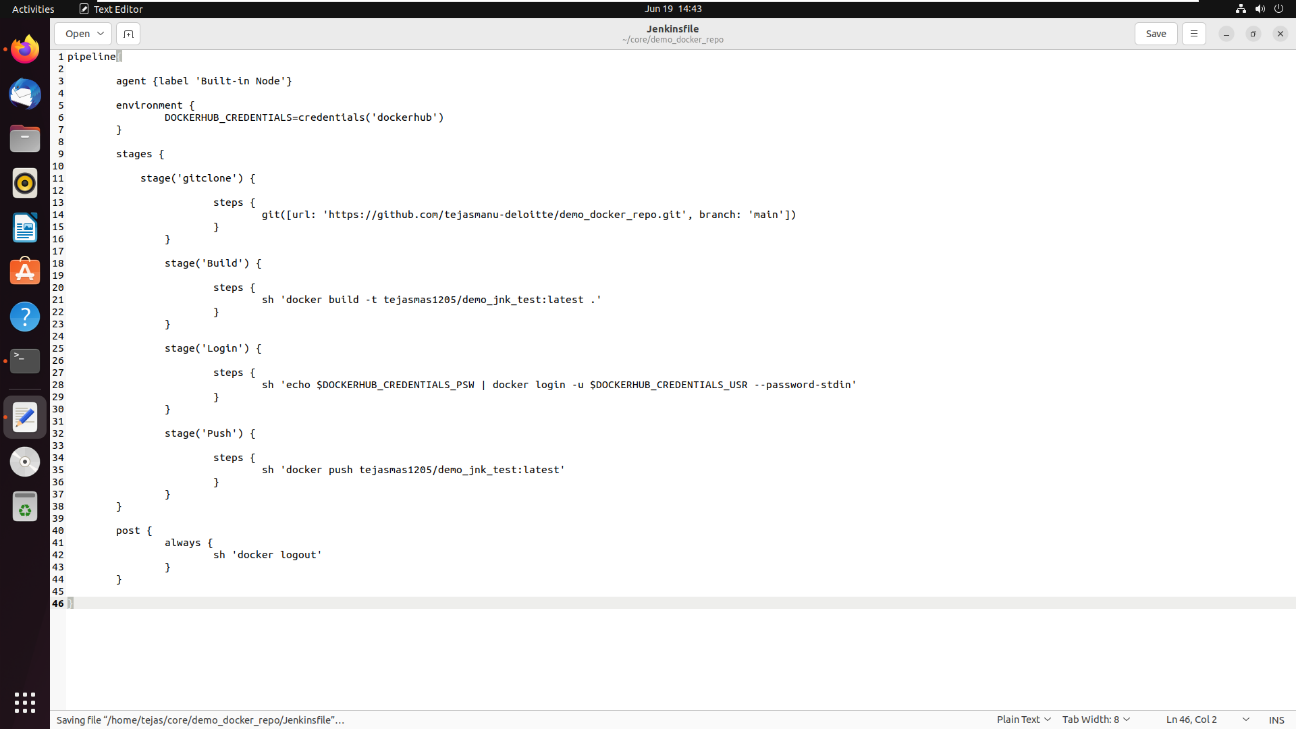


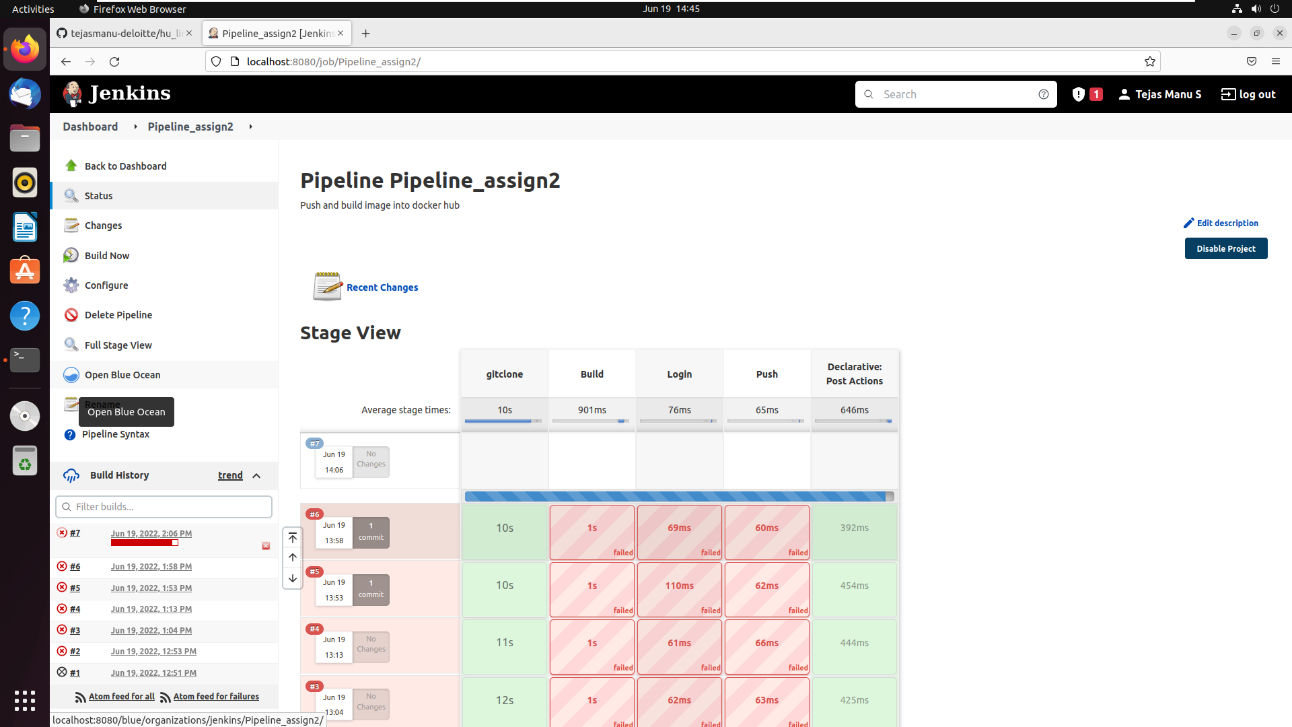


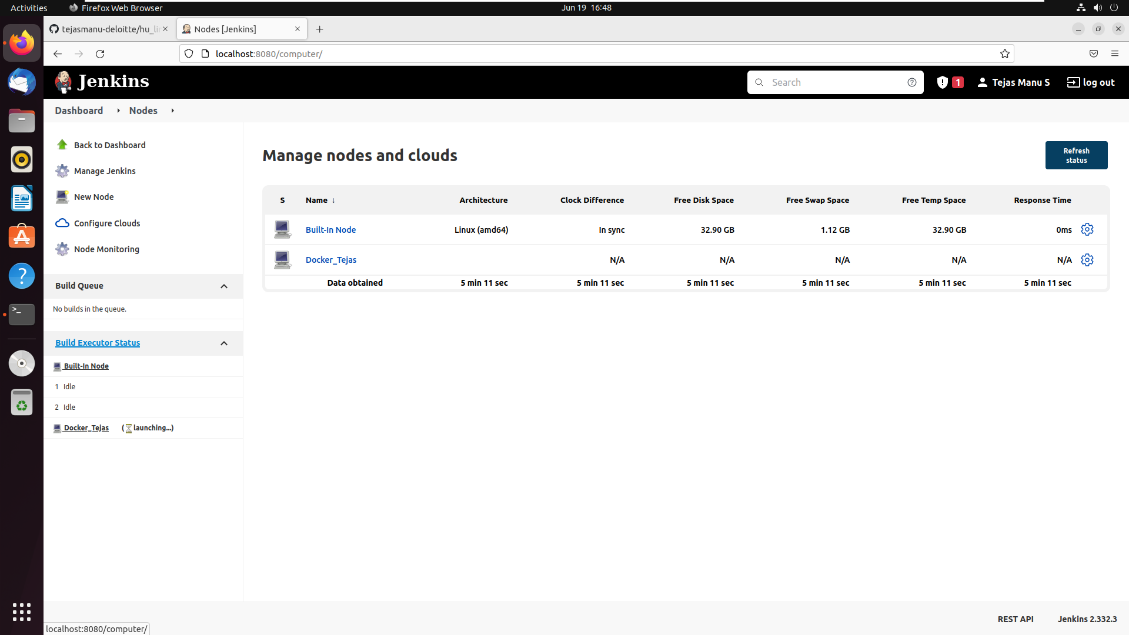


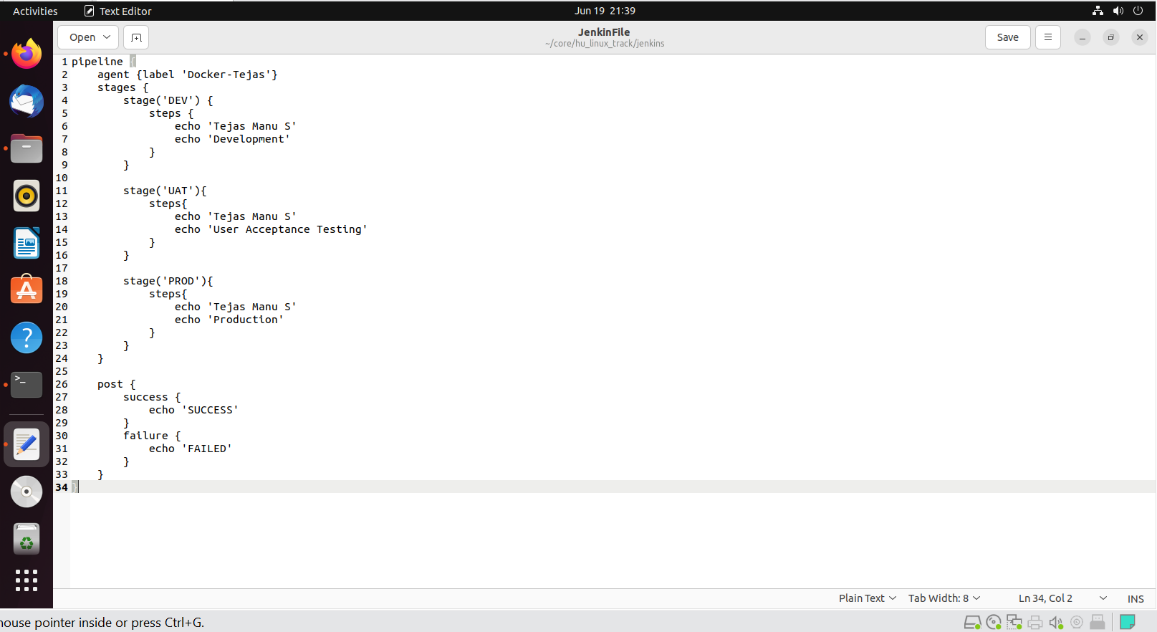


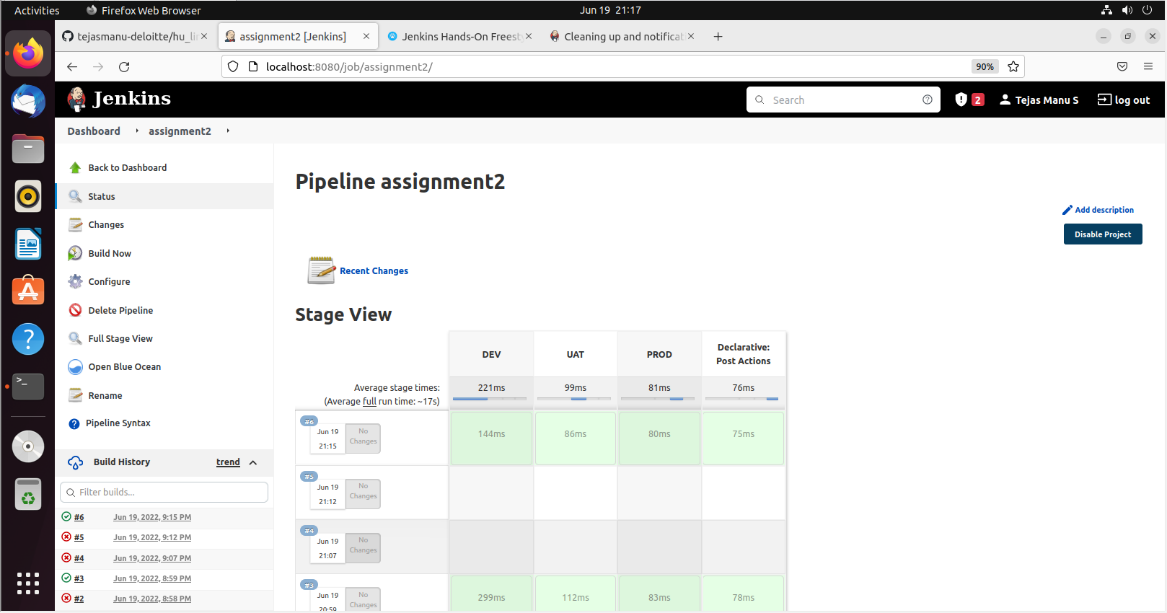
1. Create a **declarative** pipeline to build docker images and push to Dockerhub.  
   - Create a pipeline with the following stages - 1. Checkout 2. Docker build 3. Approval 3. Docker push

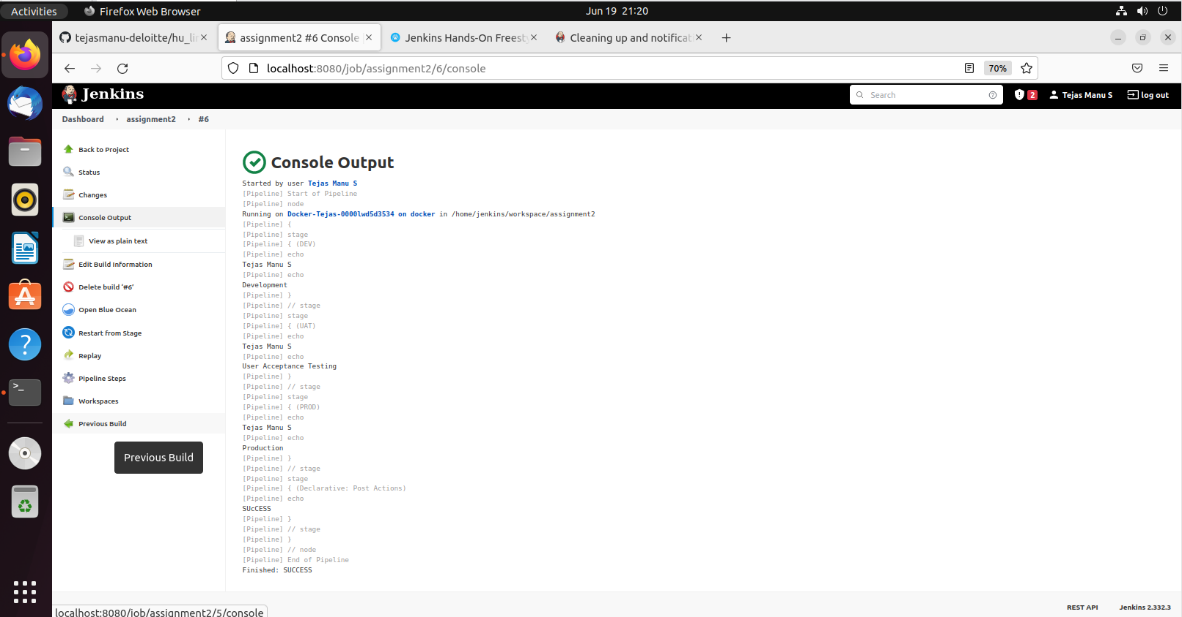




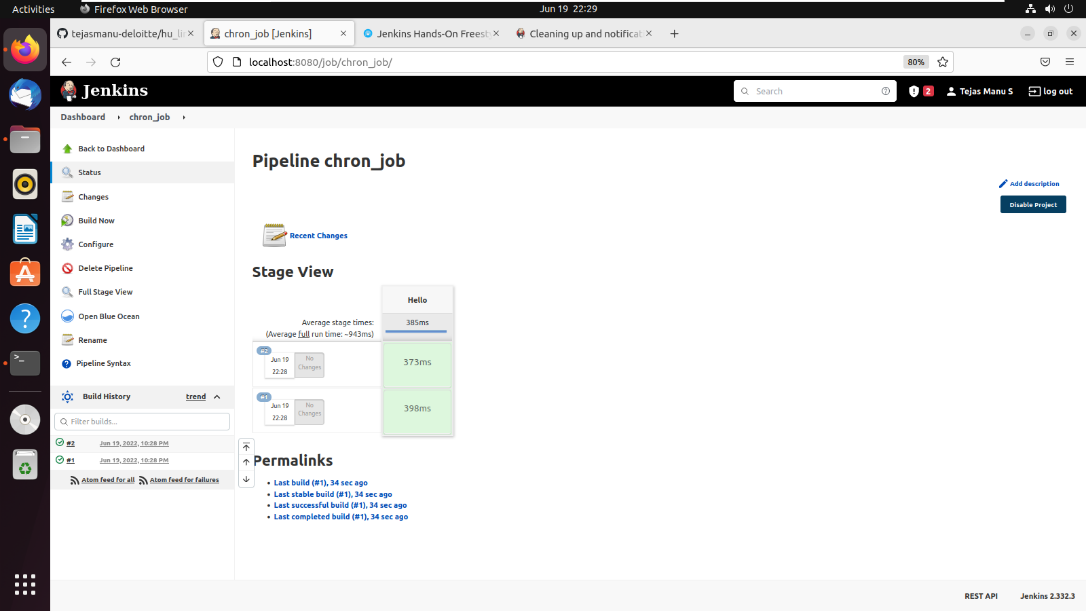
1. Use an agent name it {Docker\_YOUR\_NAME} as a label in multi stage pipeline for DEV,UAT,PROD (echo YOUR\_NAME in each stage) with post-build for messages on success and also for failures.  
   - Create an agent with docker in Jenkins  
   - Create a multistage pipeline and post-build steps.

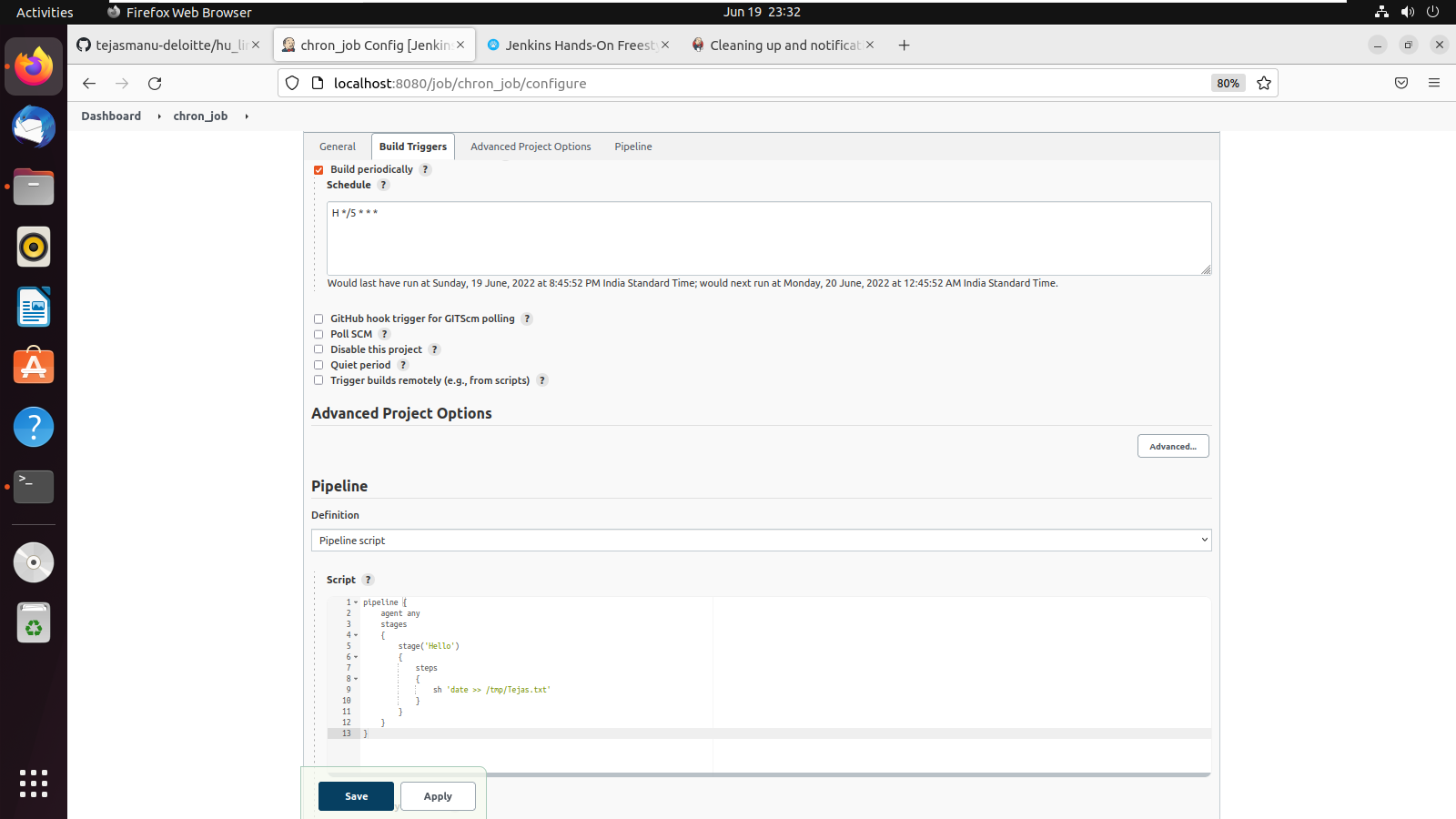






1. Schedule jobs ( cron builds) to execute the pipeline for every 5 hr and echo date into a /tmp/{YOUR\_NAME}.txt .





1. Create a parameterized multi-stage pipeline to take input in different Stages i.e., DEV,UAT,PROD, include the approval block in each stage of the pipeline to move to the next pipeline stage. Inside each stage echo {the environment}.  
   - It should have the following stages - 1. Checkout  2. ApprovalForDev 3. Dev stage 4. ApprovalForQA 5. QA Stage 6. ApprovalForUAT 7. UAT stage 8. ApprovalForProd 9. Prod Stage  
   - Create users for developer and managers.  
   - Developers should be able to trigger the job and only the managers should be able to approve the stage.
2. Create a pipeline that clones the given repository, runs a sonarqube test on it, Build the docker container, scan the docker image with Anchore Engine and then publish the image to the dockerhub private repository If everything Passes successfully. As a post step publishes a text file with the build number in it as an artifact for the job.  
   - Setup sonarqube has a docker container locally.  
   - Setup Anchore Engine has a docker container locally.  
   - Sonarqube Quality gate must be configured properly.  
   - Anchore Engine should get failed if - 22 port is open, USER root is specified, ADD directive is present in file.  
   - Setup the sonar and anchore plugins on Jenkins.  
   - Create a text file with the build number in it as an artifact for the job