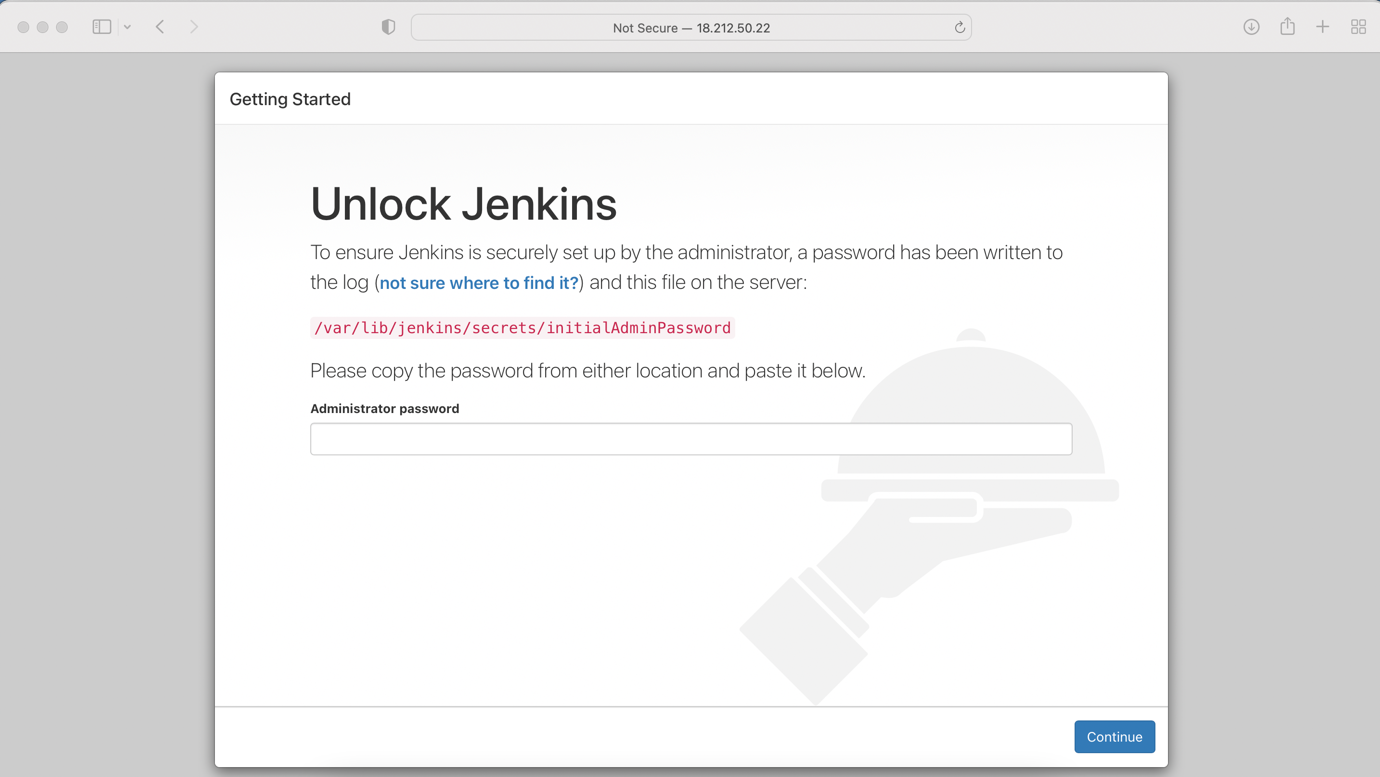
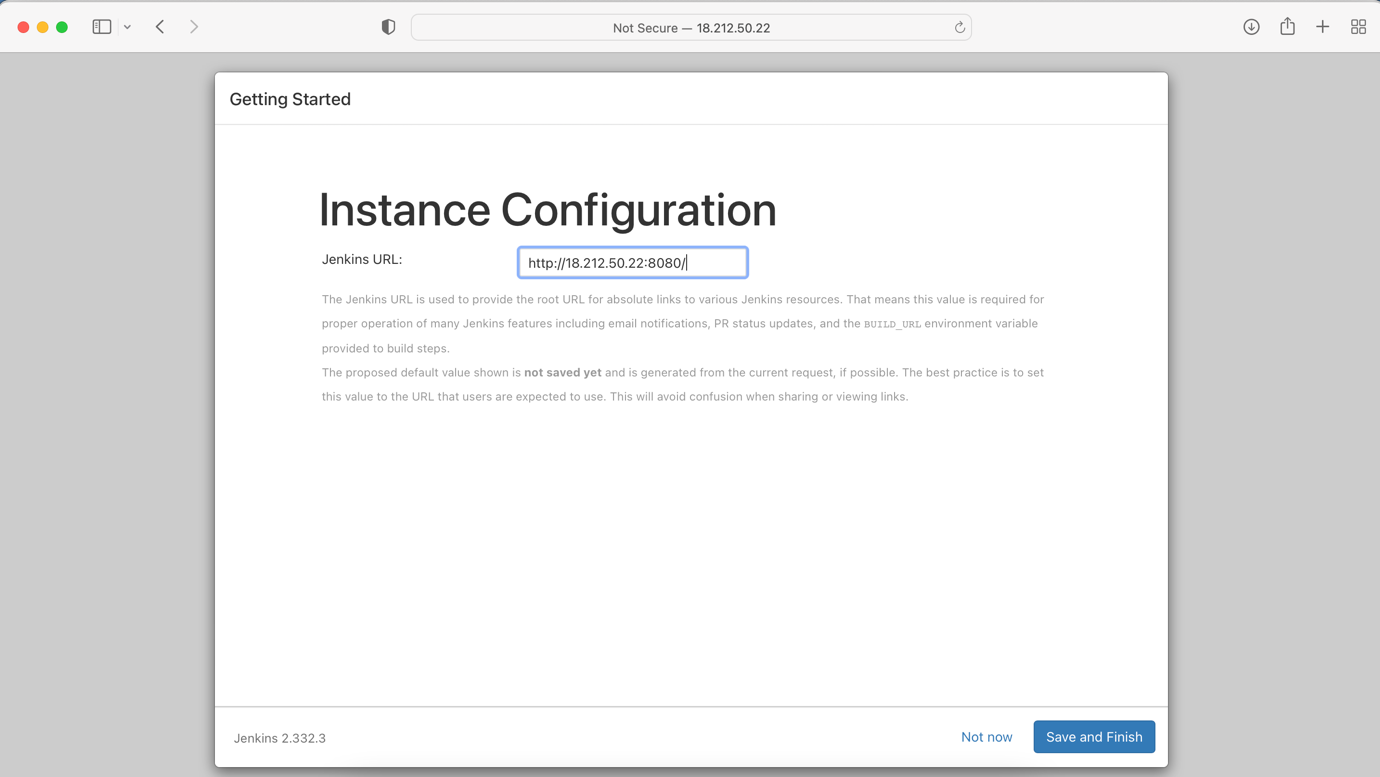
**Configuring the Jenkins :**

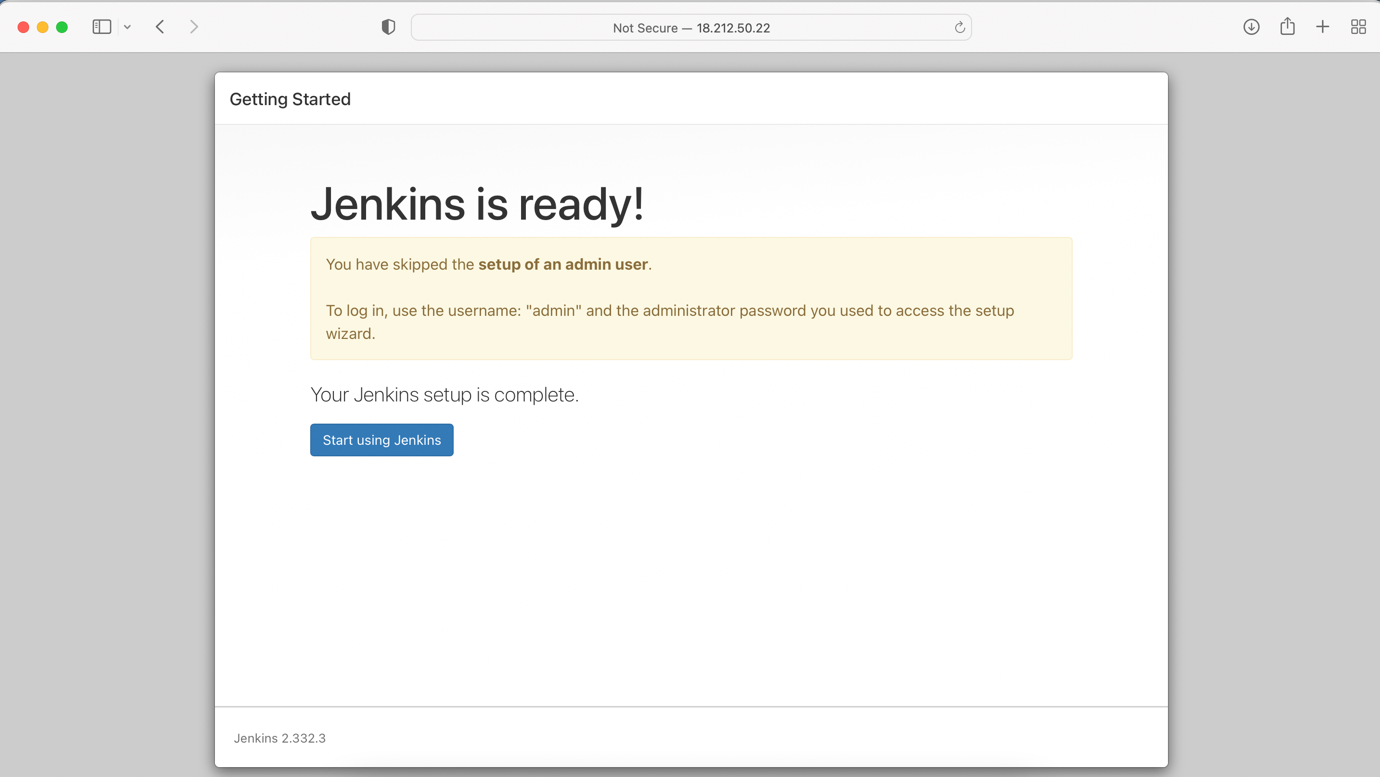


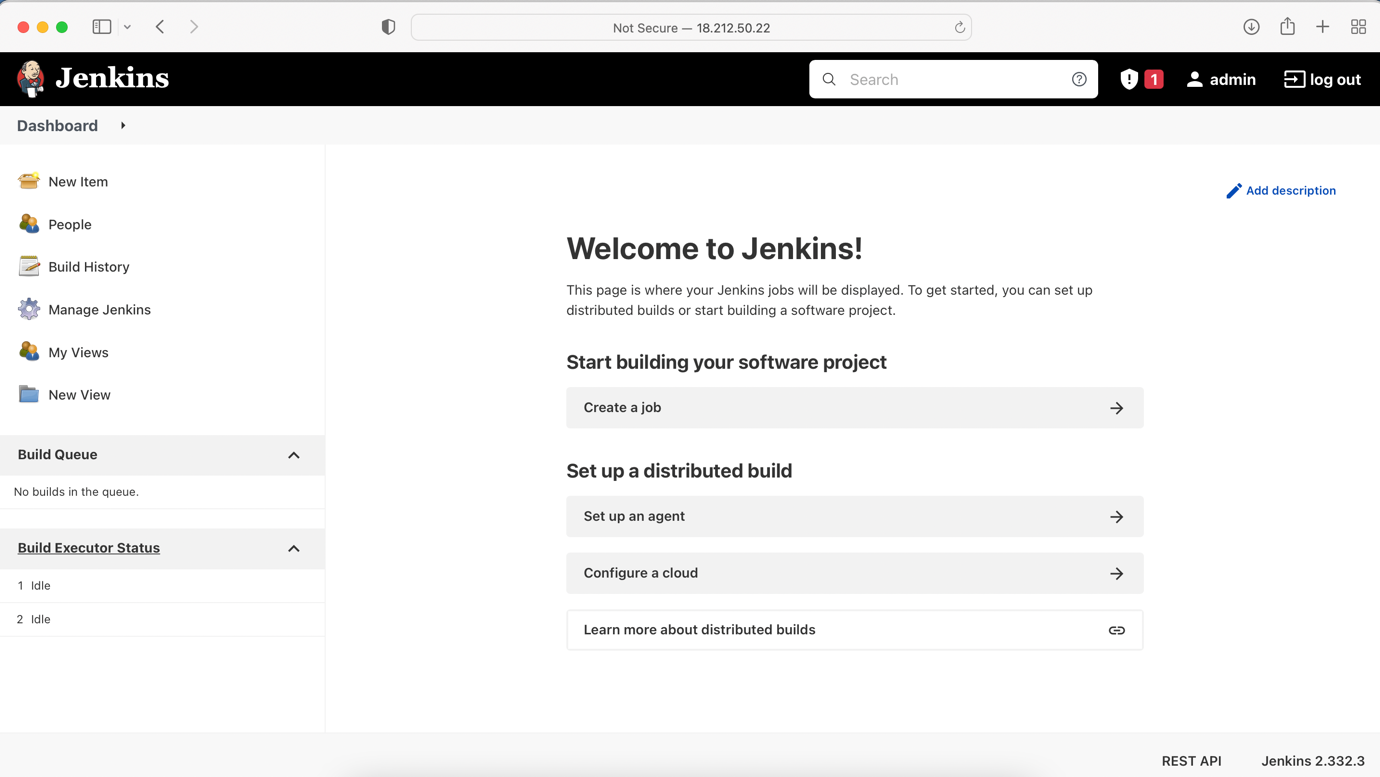
We can get the password will installation of Jenkins completed

sudo cat /var/lib/jenkins/secrets/initialAdminPassword

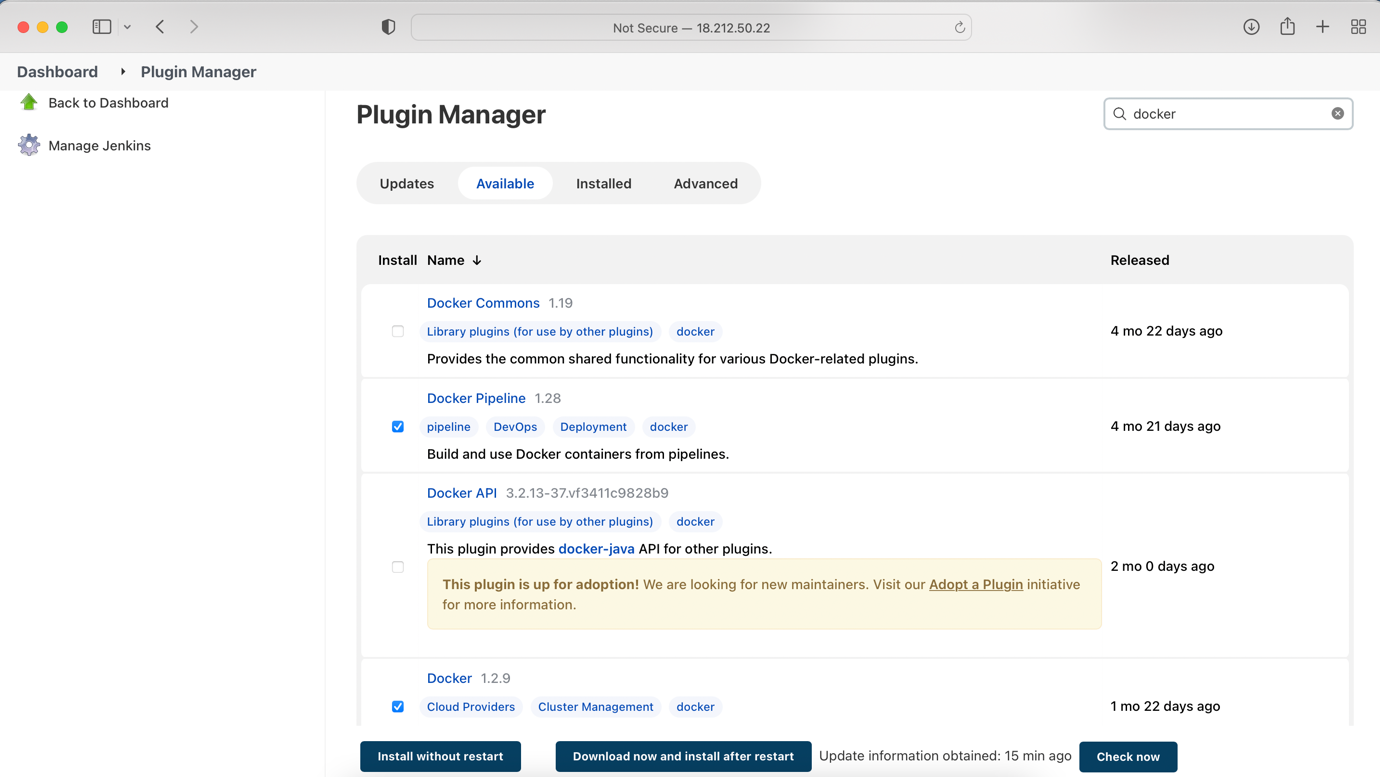
Please follow the below screenshots to configure the Jenkins

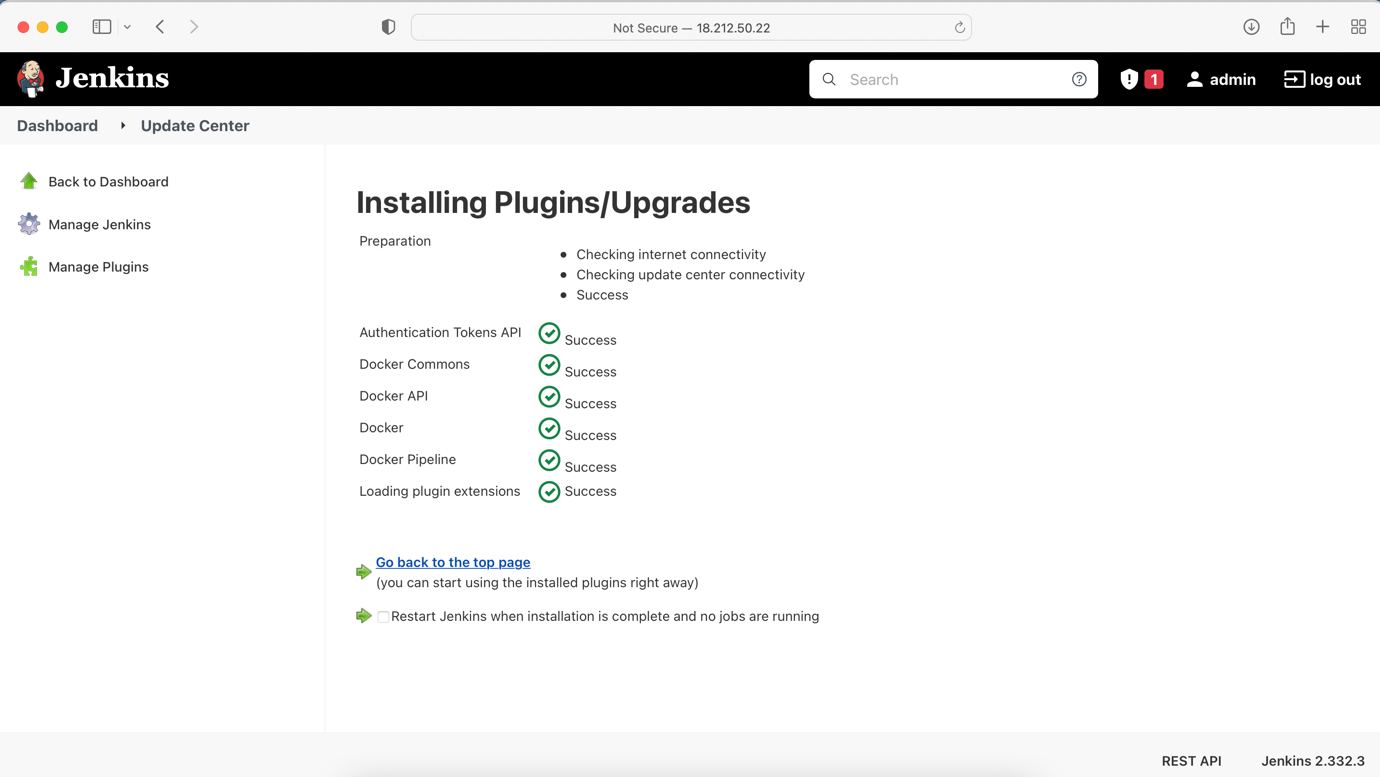




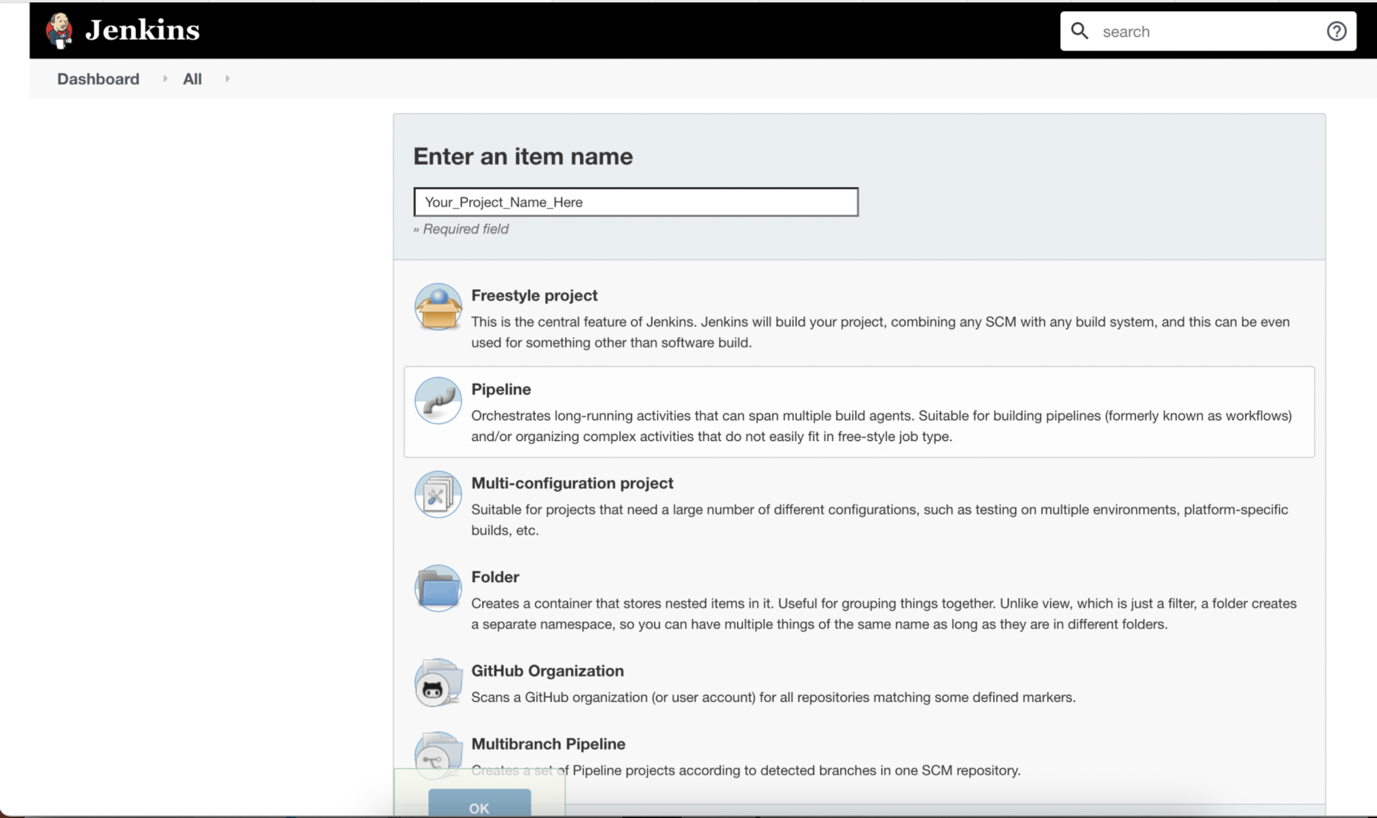


Install the docker plugins from the plugin manger.



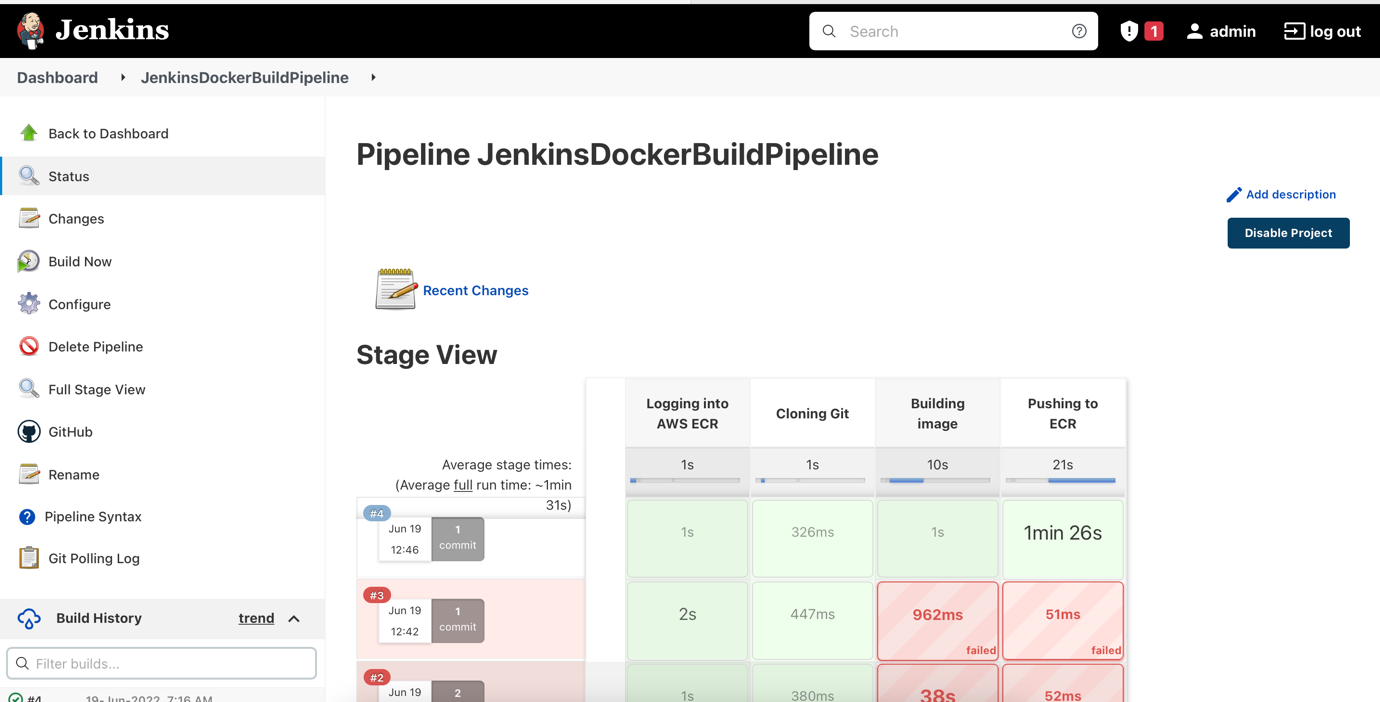


**Step 1 – Create a pipeline in Jenkins, with project name**



**Step 2: Use below pipeline code**

pipeline { agent any environment { AWS\_ACCOUNT\_ID="YOUR\_ACCOUNT\_ID\_HERE" AWS\_DEFAULT\_REGION="CREATED\_AWS\_ECR\_CONTAINER\_REPO\_REGION" IMAGE\_REPO\_NAME="ECR\_REPO\_NAME" IMAGE\_TAG="IMAGE\_TAG" REPOSITORY\_URI = "${AWS\_ACCOUNT\_ID}.dkr.ecr.${AWS\_DEFAULT\_REGION}.amazonaws.com/${IMAGE\_REPO\_NAME}" } stages { stage('Logging into AWS ECR') { steps { script { sh "aws ecr get-login-password --region ${AWS\_DEFAULT\_REGION} | docker login --username AWS --password-stdin ${AWS\_ACCOUNT\_ID}.dkr.ecr.${AWS\_DEFAULT\_REGION}.amazonaws.com" } } } stage('Cloning Git') { steps { checkout([$class: 'GitSCM', branches: [[name: '\*/main']], doGenerateSubmoduleConfigurations: false, extensions: [], submoduleCfg: [], userRemoteConfigs: [[credentialsId: '', url: ' https://github.com/sirishrayaprolu/webapp-orderbird.git']]]) } } // Building Docker images stage('Building image') { steps{ script { dockerImage = docker.build "${IMAGE\_REPO\_NAME}:${IMAGE\_TAG}" } } } // Uploading Docker images into AWS ECR stage('Pushing to ECR') { steps{ script { sh "docker tag ${IMAGE\_REPO\_NAME}:${IMAGE\_TAG} ${REPOSITORY\_URI}:$IMAGE\_TAG" sh "docker push ${AWS\_ACCOUNT\_ID}.dkr.ecr.${AWS\_DEFAULT\_REGION}.amazonaws.com/${IMAGE\_REPO\_NAME}:${IMAGE\_TAG}" } } } } }



ECR ([Amazon Elastic Container Registry](https://us-east-1.console.aws.amazon.com/ecr/get-started?region=us-east-1))

Once the pipeline is completed we can view the image in the ECR

