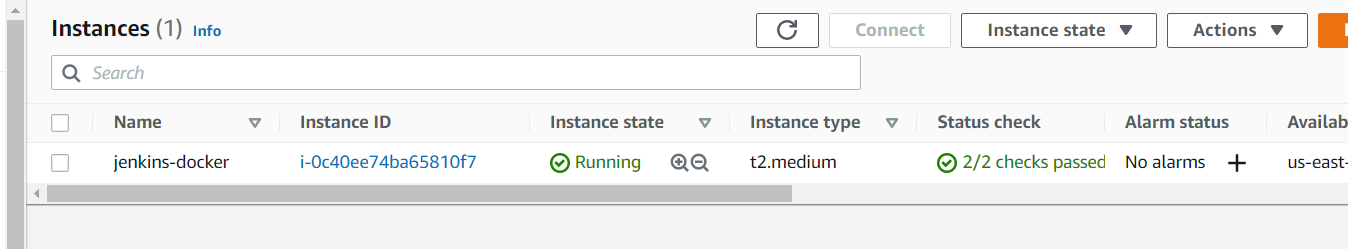
CI-PROJECT

**AIM**: When ever a developer writes a **Dockerfile** and commits it to **Github** ,**Jenkins** should automatically fetch the code Build the docker image from docker file using **Docker** and Test using Tool called **trivy** and upload it to **Amazon ECR** Repository

Required Tools:

* Jenkins for Automation
* Docker to Build Image
* Github as SCM
* ECR to store Images



Install Jenkins with java dependency

Then configure Jenkins and install below **plugins**

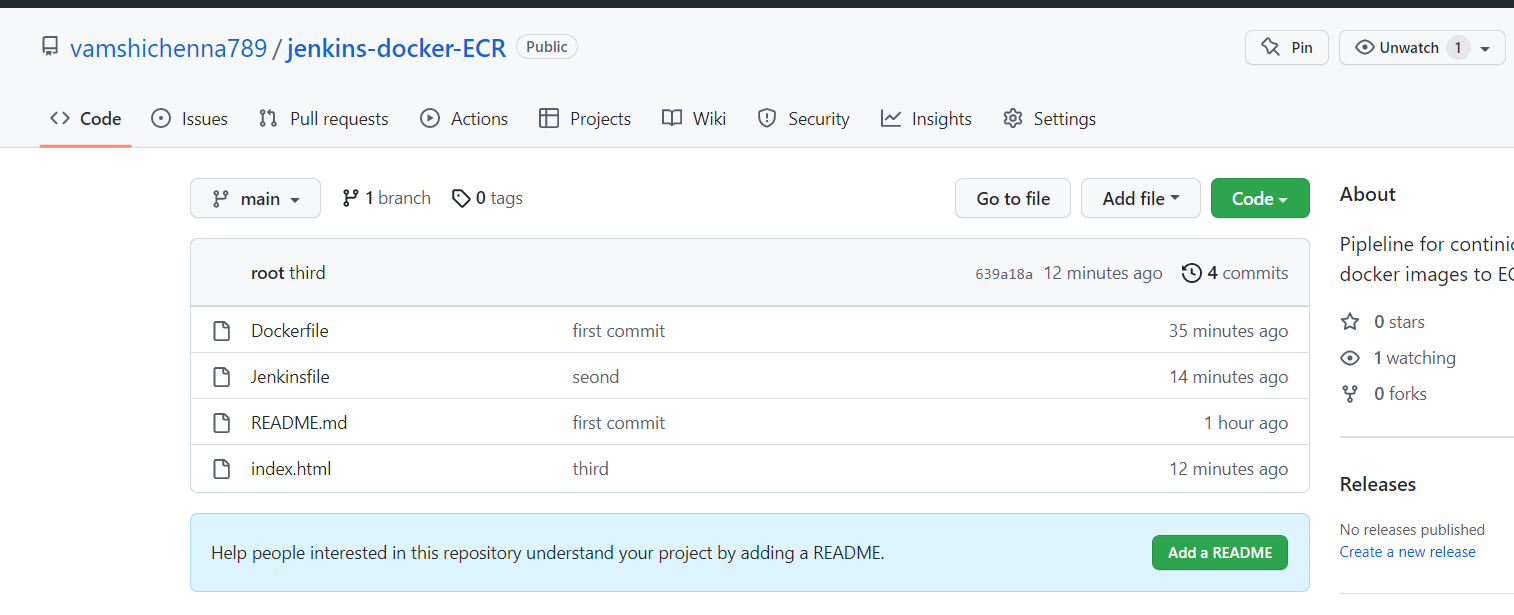
1.Cloud Bees AWS Credentials: is needed to communicate to aws with credentials

2. Docker pipeline: needed to communicate to docker

3. Amazon ECR: generates docker authentication tocken from amazon credentials to access amazon ecr

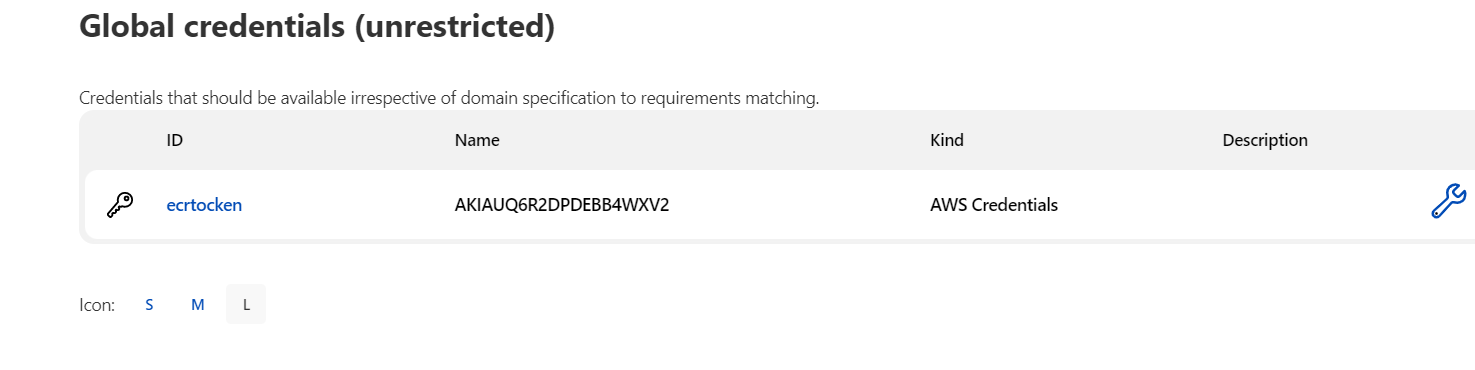
Install docker on Jenkins server

Create a repository on github to push our code to it



Go to Jenkins

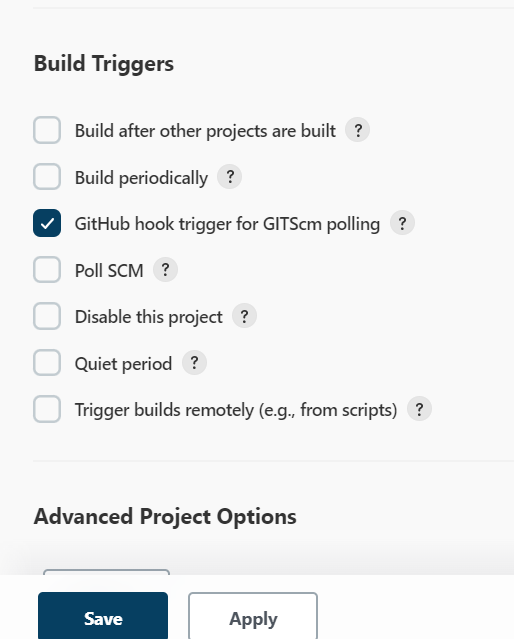
Give Iam user credentials to access aws under manage credentials option



Go to Jenkins 🡪 New Items selct Pipeline

Give name

Enable Build trigger for github hook trigger scm



Give repo url and give branch main

Now go to github and add webhook for same

Wirte the Jenkins pipeline accordingly as shown below

pipeline {

agent any

options {

skipStagesAfterUnstable()

}

stages {

stage('Clone repository') {

steps {

script{

checkout scm

}

}

}

stage('Build') {

steps {

script{

app = docker.build("docker-demo:${env.BUILD\_ID}")

}

}

}

stage('Test'){

steps {

sh 'trivy image docker-demo:latest || exit 0'

}

}

stage('Deploy') {

steps {

script{

docker.withRegistry('https://017187748261.dkr.ecr.us-east-1.amazonaws.com', 'ecr:us-east-1:ecrtocken') {

app.push()

}

}

}

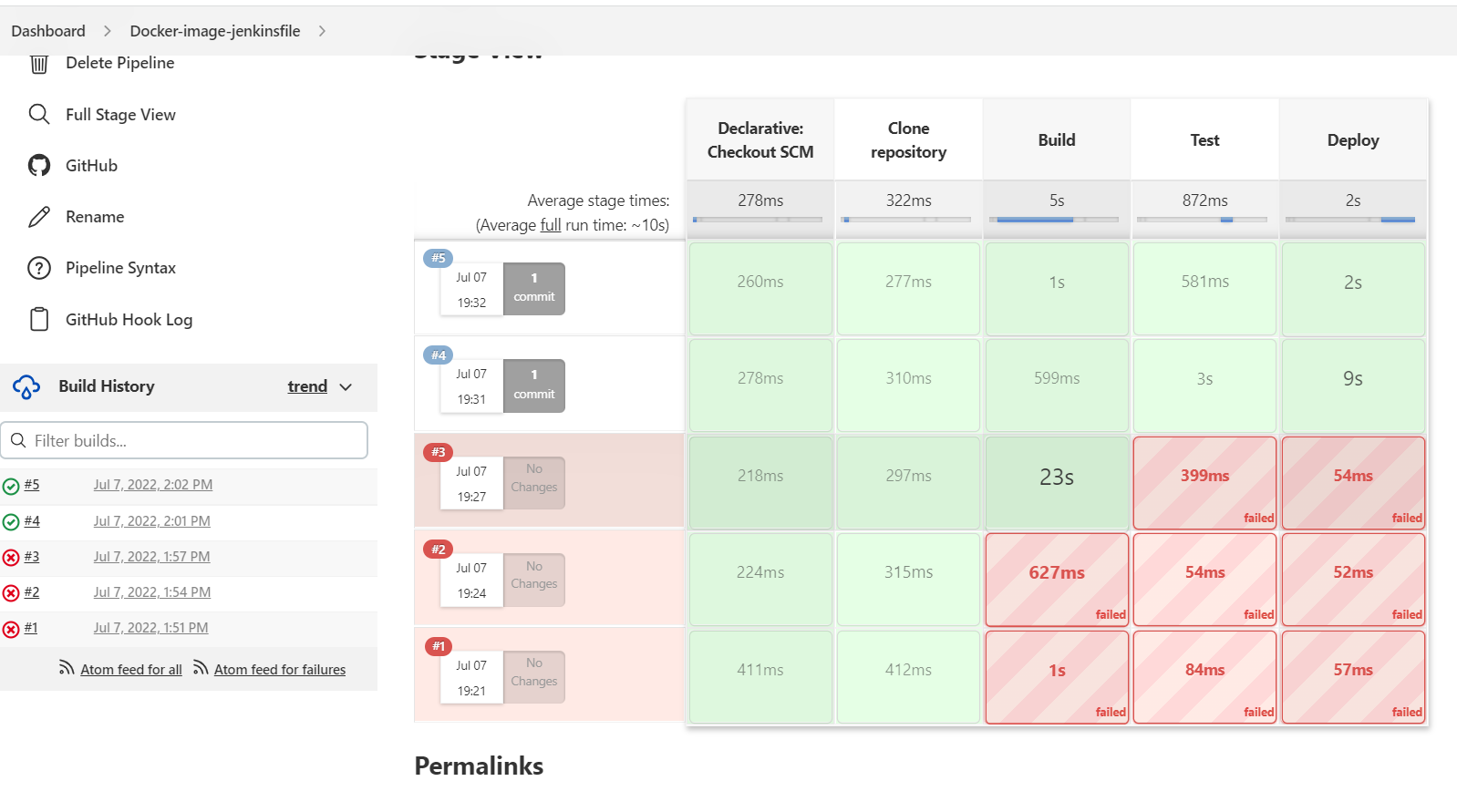
}

}

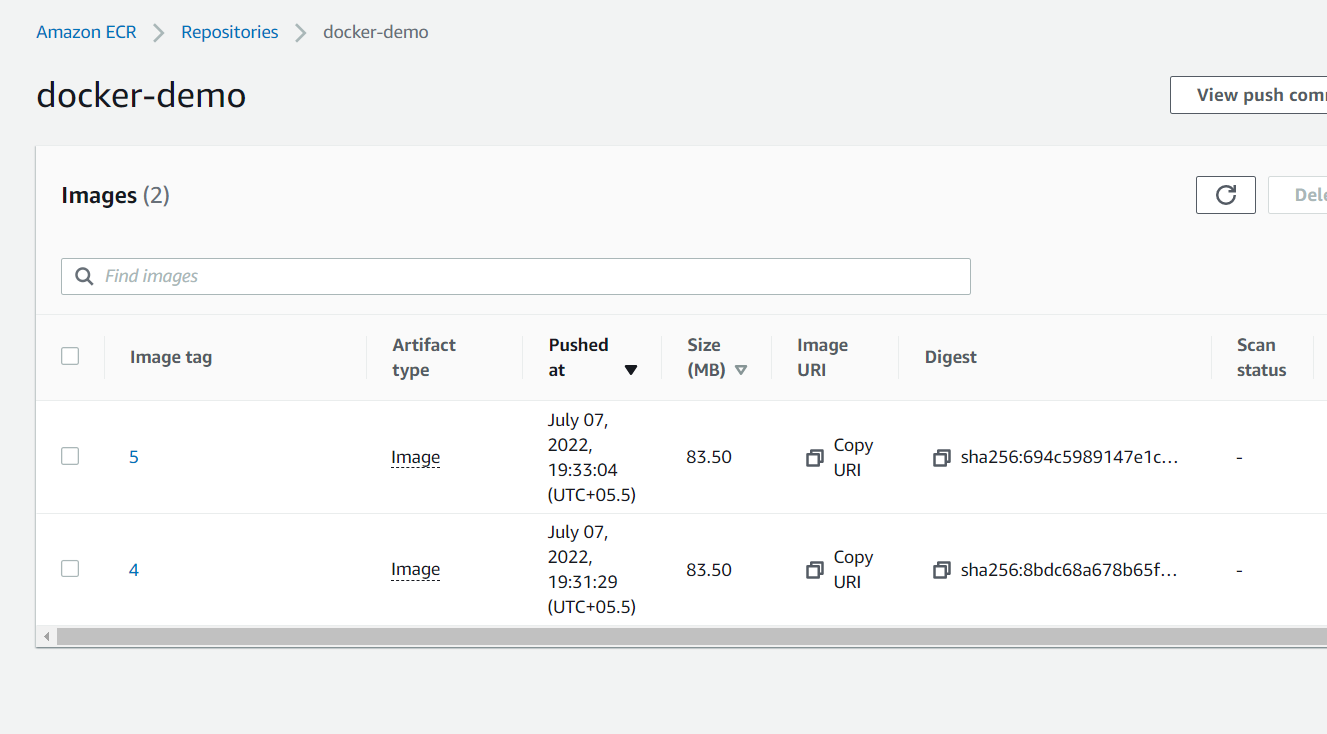
}

Write your docker file and push them to git repo

Click on initial build and after that for every push it will triger the pipeline



Images are uploaded tp ecr as below



**Conclusion**: With out any manual interruption we have automated the process of building and uploading docker images to the ECR repository