Jenkins-Maven-SonarQube-Selenium Test-Docker

1. Installing sonarqube from below url-Long term support(Community Edition)-SonarQube 7.9.x LTS (July 2019)

sonarqube.org/downloads/ and extract the ZIP FILE

2.Install JDK version 11 (download from https://www.oracle.com/technetwork/java/javase/downloads/jdk11-downloads-5066655.html). If the download link not working, google "JDK 11 download".

and change the value for the line below in \conf\wrapper.conf:

wrapper.java.command=C:\Program Files\Java\jdk-11.0.6\bin\java

3. Go to C:\Users\chanrm\Downloads\sonarqube-7.9.3\bin\windows-x86-64 and click on StartSonar.bat

4.Access sonarqube in http://localhost:9000/ (Default port)

Login using default Credentials username:admin pwd:admin

5.We need to create the security token. So in the SonarQube> In top right Under Administration

>My Account>Security>Generation Tokens

Enter token name and generate> Copy the token

65540e498efc981337757b0c8e4787a6549cb97e

6. Go to jenkins file in C:\Program Files (x86)\Jenkins and Launch Jenkins using

java -jar jenkins.war --httpPort=8081

7. Manage Jenkins> Manage plugins> Install SonarQube Scanner for Jenkins plugin

8. Configure system>Under SonarQube servers> Give

Name:Sonarqube

Server URL:http://localhost:9000

Server authentication token: Add the token

Apply

9.Global Tool Configuration>SonarQube Scanner

Name:SonarQube Scanner

Install From Maven Central: SonarQube Scanner 2.8

10.For Selinium Test Case, Install the chrome driver from https://chromedriver.storage.googleapis.com/index.html?path=2.38/ and extract exe file

and set the path of driver inJenkins-Selenium-Docker/src/test/java/testcases/mytheresa/WebdriverSettings.java /

System.setProperty("webdriver.chrome.driver", "C:/Users/chanrm/Downloads/chromedriver\_win32/chromedriver.exe");

So while running the test case, it will open up a chrome browser and run the required tests when given the maven clean package command

=====================================================================================================

[INFO] T E S T S

[INFO] -------------------------------------------------------

[INFO] Running TestSuite

Starting ChromeDriver 2.38.552522 (437e6fbedfa8762dec75e2c5b3ddb86763dc9dcb) on port 5155

Only local connections are allowed.

Jun 08, 2020 10:21:30 PM org.openqa.selenium.remote.ProtocolHandshake createSession

INFO: Detected dialect: OSS

[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 41.384 s - in TestSuite

[INFO]

[INFO] Results:

[INFO]

[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0

[INFO]

[INFO]

[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ E2EUserJourney ---

[INFO] Building jar: C:\Users\chanrm\.jenkins\workspace\Jenkins-Selenium-Maven-Sonarqube\target\E2EUserJourney-0.0.1-SNAPSHOT.jar

[INFO] ------------------------------------------------------------------------

[INFO] BUILD SUCCESS

[INFO] ------------------------------------------------------------------------

[INFO] Total time: 56.590 s

[INFO] Finished at: 2020-06-08T22:22:07+05:30

[INFO] ------------------------------------------------------------------------

============================================================================

11.Pipeline Job and Build

node {

stage('SCM') {

git 'https://github.com/chandana-git/Jenkins-Selenium-Docker.git'

}

stage('maven'){

bat label: '', script: 'mvn clean package'

}

stage('SonarQube analysis') {

withSonarQubeEnv()

bat label: '', script: 'mvn org.sonarsource.scanner.maven:sonar-maven-plugin:3.6.0.1398:sonar'

}

}

stage('Publish TestNg Results'){

step([$class: 'Publisher'])

}

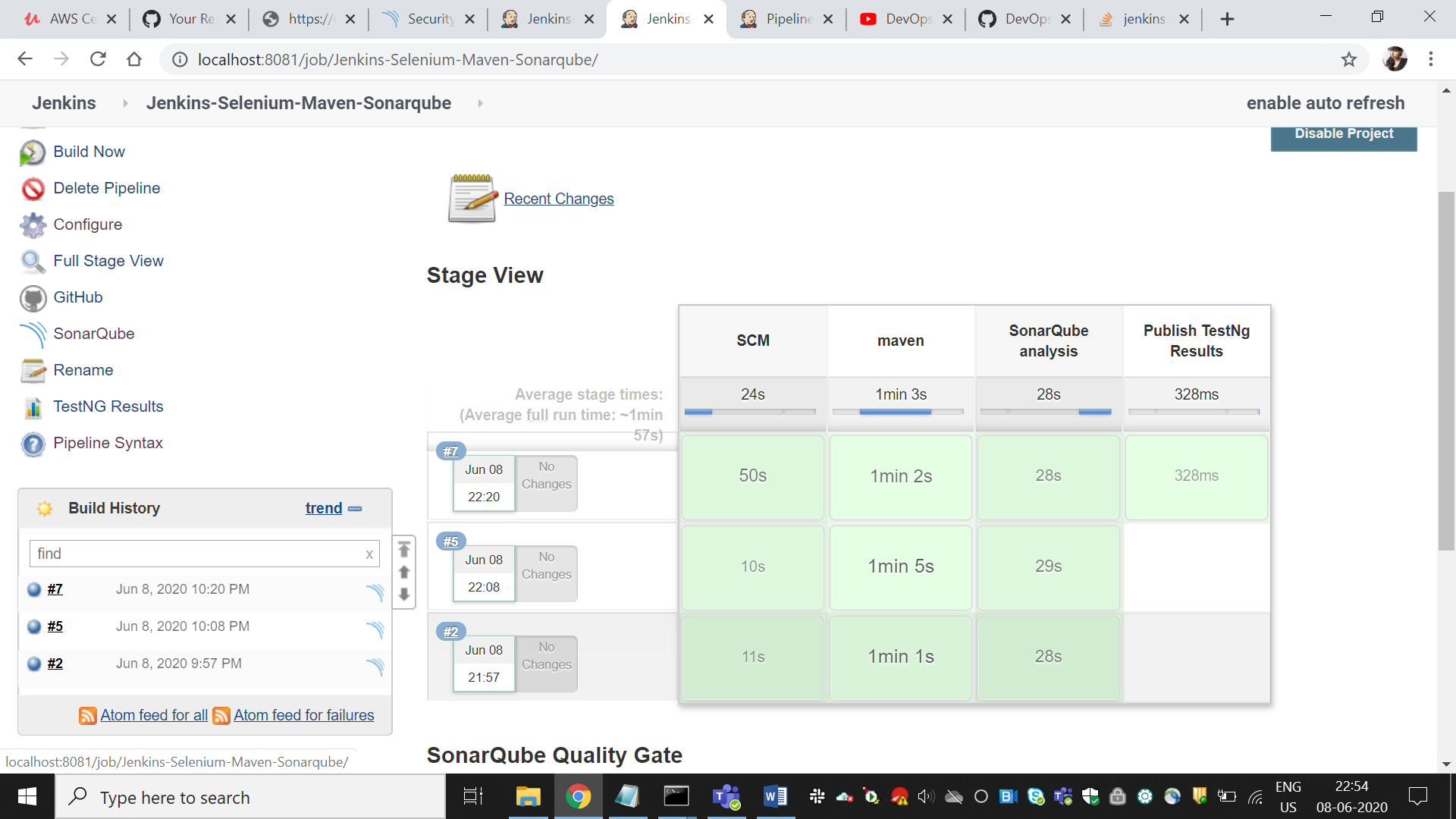
/\*stage('archive Artifacts')

{

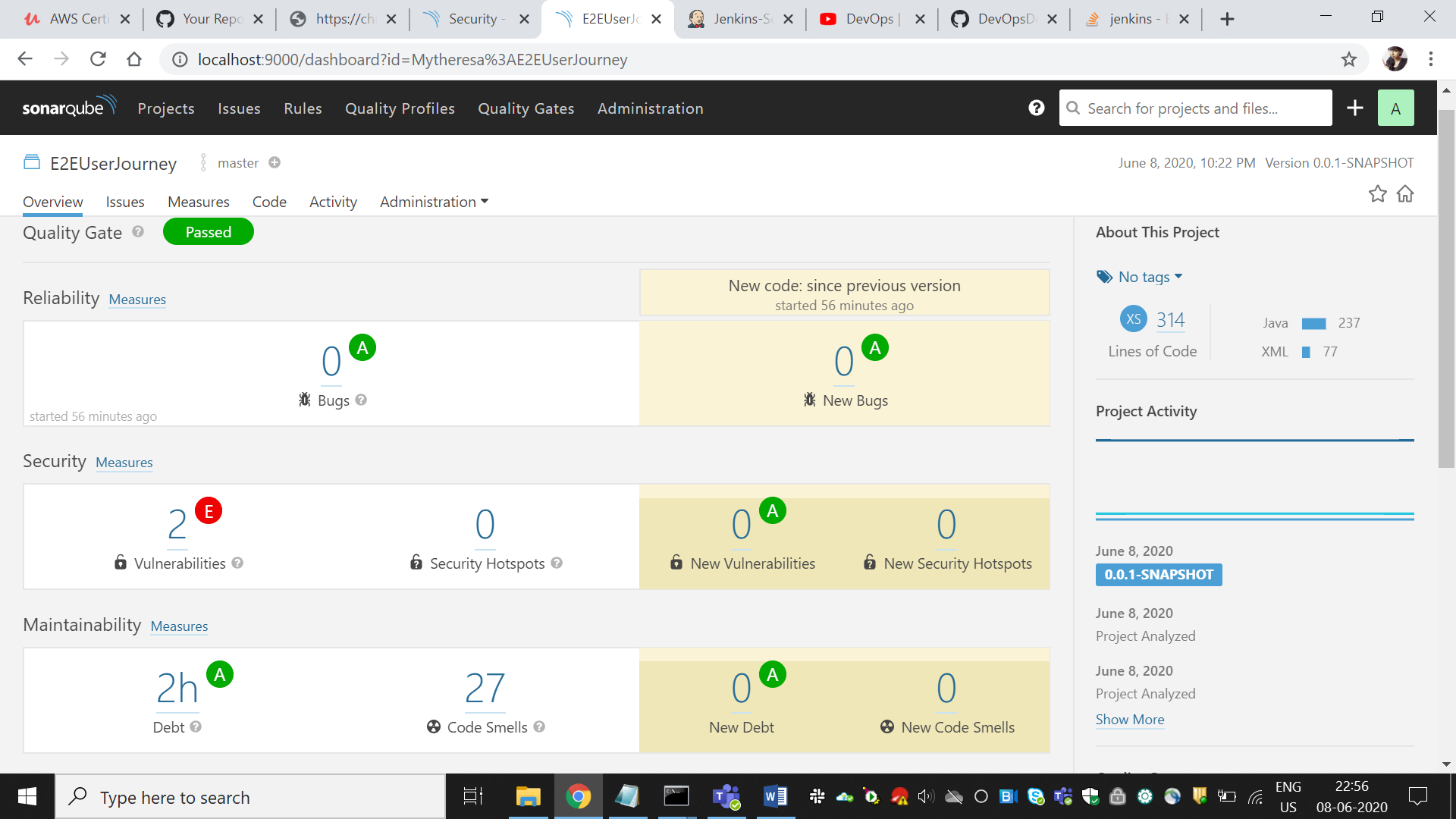
archiveArtifacts '/target/\*.jar'

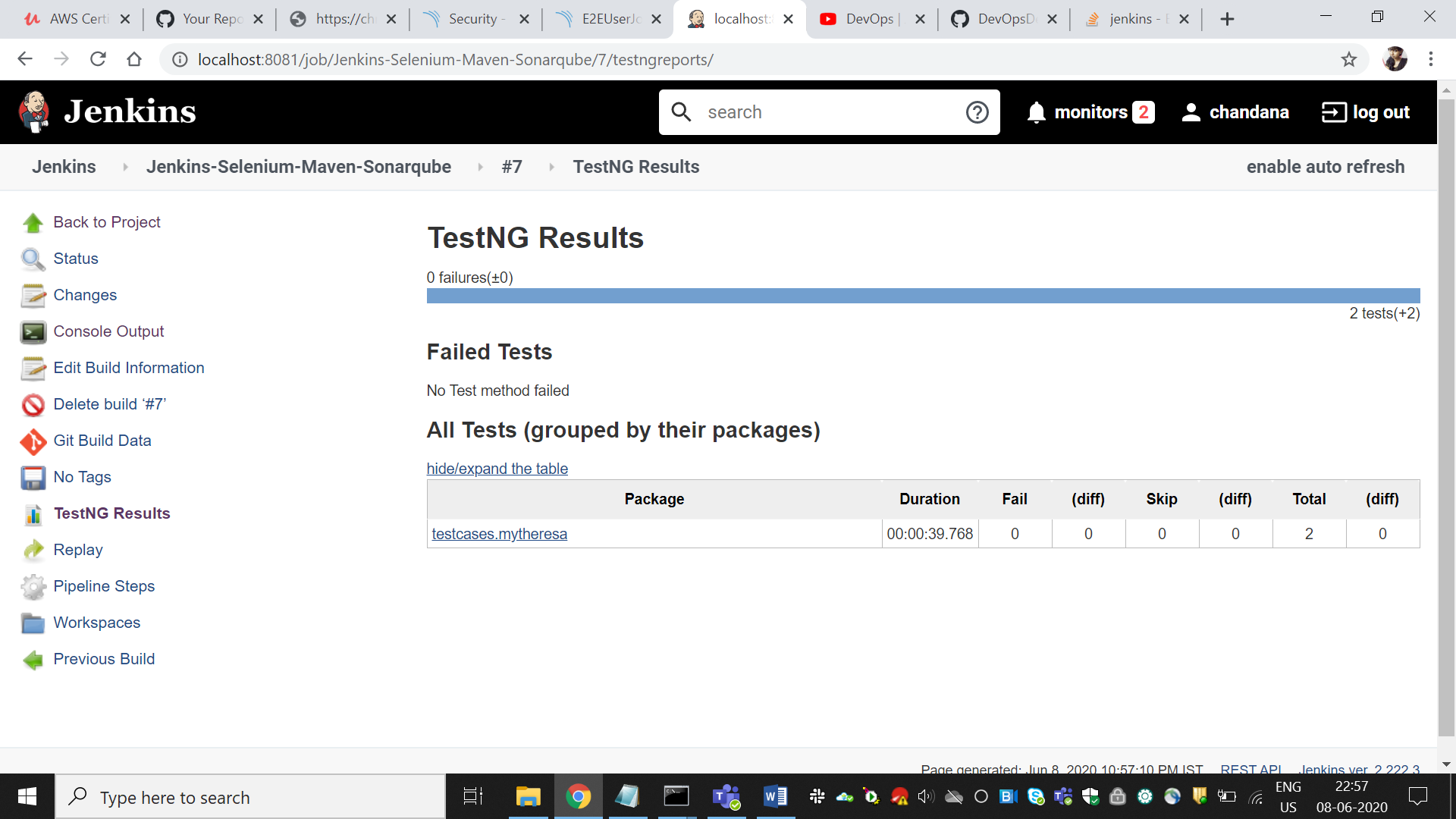
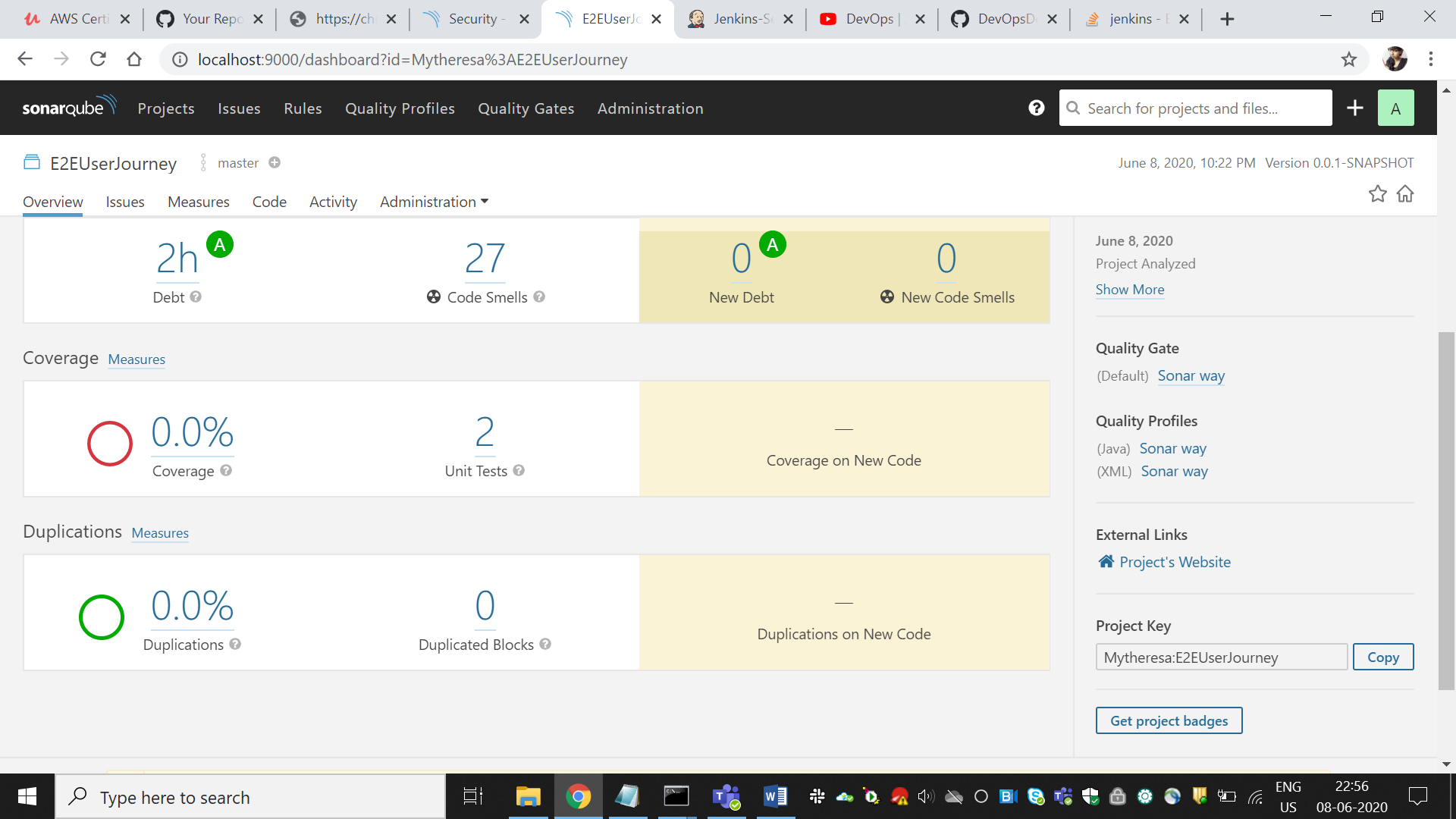
}\*/

}



Click on SonarQube





# PIPELINE

node {

stage('SCM') {

git 'https://github.com/chandana-git/Jenkins-Selenium-Docker.git'

}

stage('maven'){

bat label: '', script: 'mvn clean package'

}

stage('SonarQube analysis') {

withSonarQubeEnv()

bat label: '', script: 'mvn org.sonarsource.scanner.maven:sonar-maven-plugin:3.6.0.1398:sonar'

}

}

stage('Publish TestNg Results'){

step([$class: 'Publisher'])

}

stage('archiving the artifact artifacts')

{

archiveArtifacts 'target/\*.war'

}

/\*

stage('Build Docker Image'){

//sh 'sudo chmod 666 /var/run/docker.sock'

//bat label: '', script: "docker build . -t chandanarm/jenkins-app"

sh 'docker build . -t chandanarm/jenkins-app'

}

stage('Push Docker Image'){

withCredentials([string(credentialsId: 'docker-pwd', variable: 'dockerHubPwd')]) {

//bat label: '', script: "docker login -u chandanarm -p ${dockerHubPwd}"

// bat label: '', script: "docker push chandanarm/jenkins-app:latest"

sh "docker login -u chandanarm -p ${dockerHubPwd}"

}

sh 'docker push chandanarm/jenkins-app:latest'

}

\*/

}