**Name:** Atif Ansari **Roll no:** 04 **Class:** D15B

**Experiment No 8**

**AIM: Integrating Jenkins with SonarQube.**

**Theory:**

SonarQube is a powerful tool for static code analysis, enabling developers to identify and fix code quality issues, security vulnerabilities, and technical debt. Integrating it with Jenkins allows for automated code analysis during the continuous integration process. This integration enhances code quality by providing immediate feedback to developers, ensuring that code adheres to predefined quality standards.

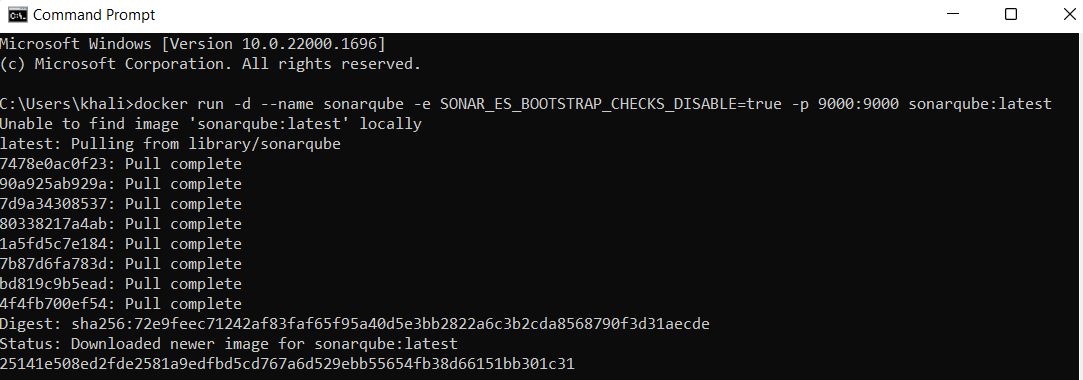
**Steps:**

Open up Jenkins Dashboard on localhost, port 8080

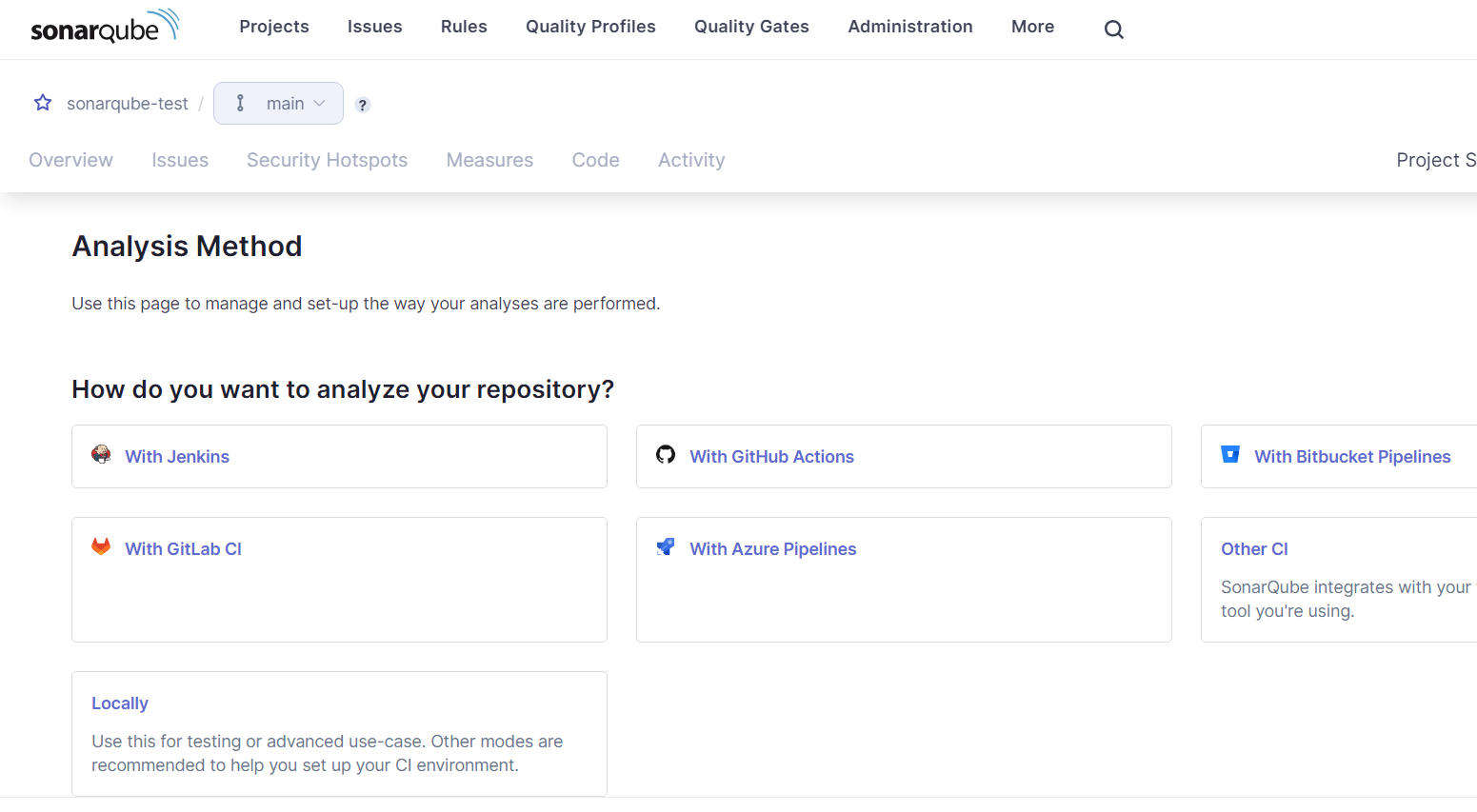
Run SonarQube in a Docker container using this command -

$ docker run -d --name sonarqube -e SONAR\_ES\_BOOTSTRAP\_CHECKS\_DISABLE=true -p 9000:9000

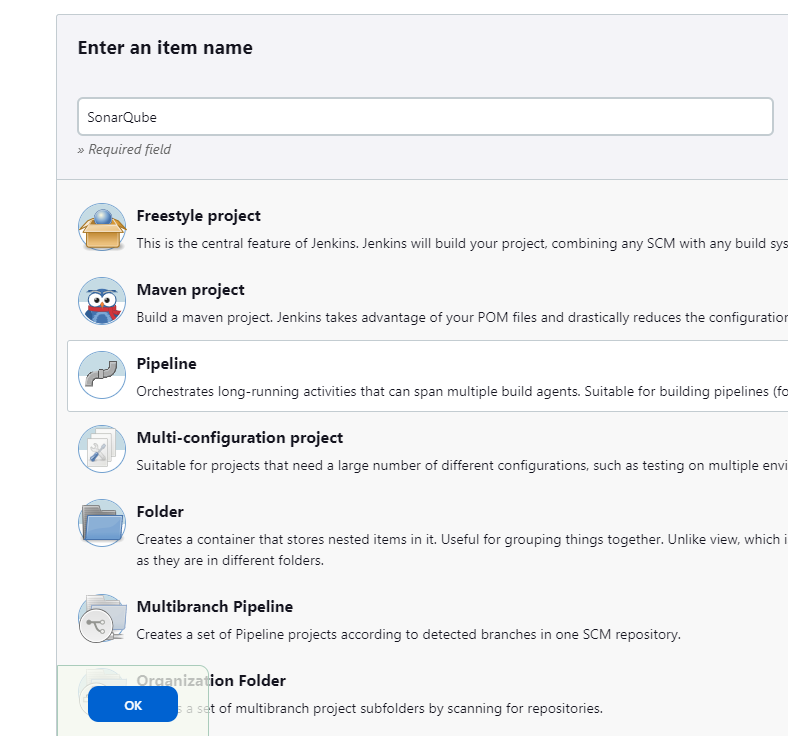
sonarqube:latest



**Create project manually**



In Jenkins create a pipeline here named “SonarQube”



Enter the following in pipeline script:

node {

stage('Cloning the GitHub Repo') {

git 'https://github.com/PrajaktaUpadhye6/MSBuild\_firstproject.git'

}

stage('SonarQube analysis') {

withSonarQubeEnv('sonarqube') {

bat "D:/sonar-scanner-cli-5.0.1.3006-windows/sonar-scanner-5.0.1.3006-windows/bin/sonar-scanner.bat \

-D sonar.login=admin \

-D sonar.password=abc \

-D sonar.projectKey=AdDevops \

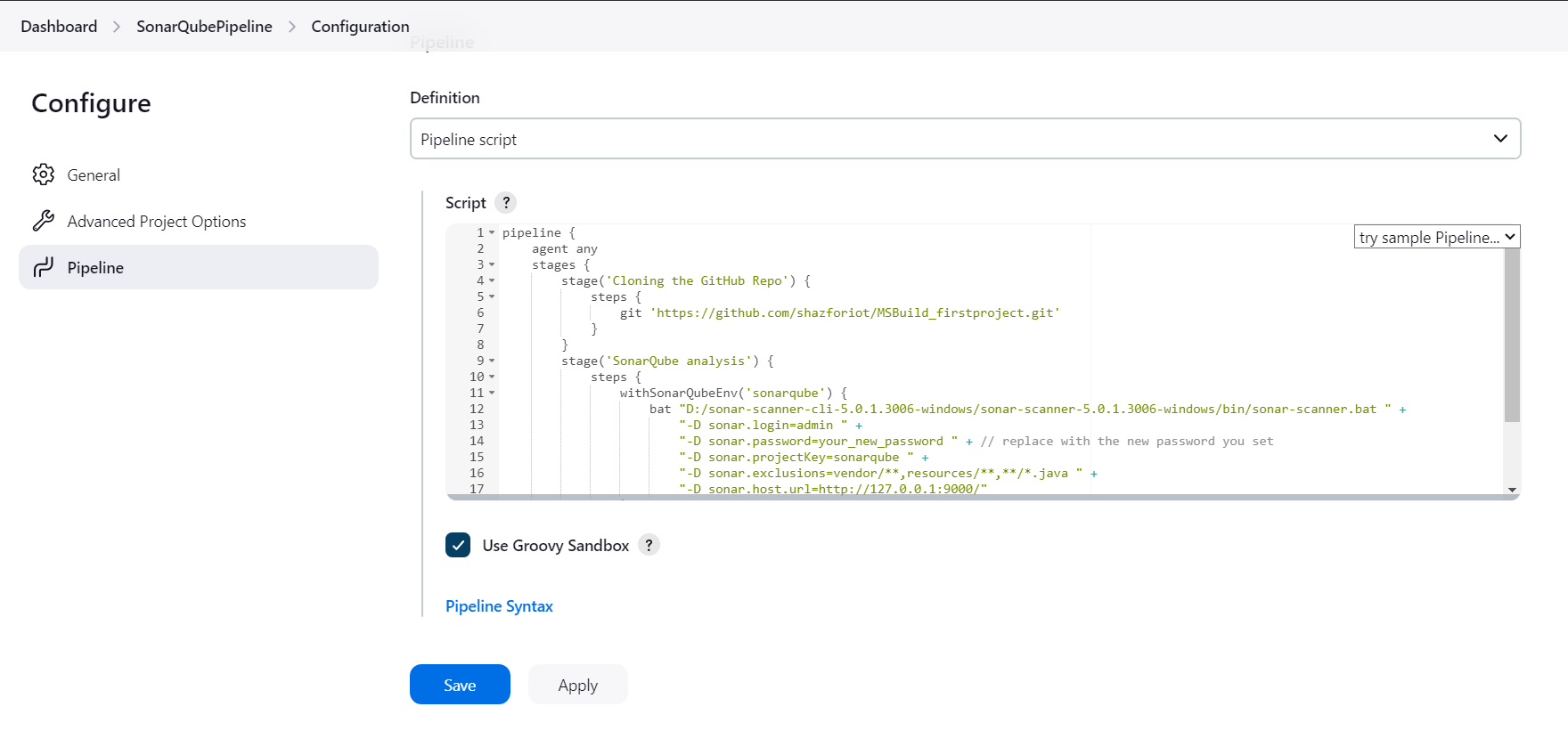
-D sonar.exclusions=vendor/\*\*,resources/\*\*,\*\*/\*.java \

-D sonar.host.url=http://127.0.0.1:9000/"

}

}

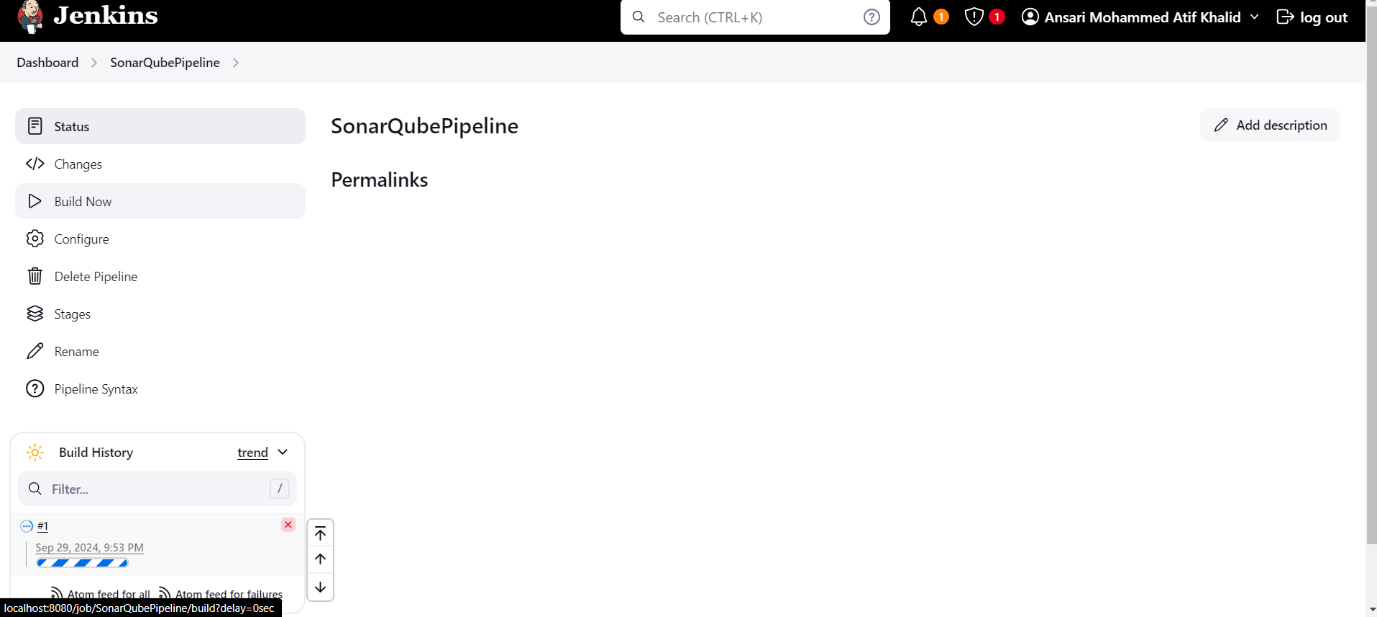
}



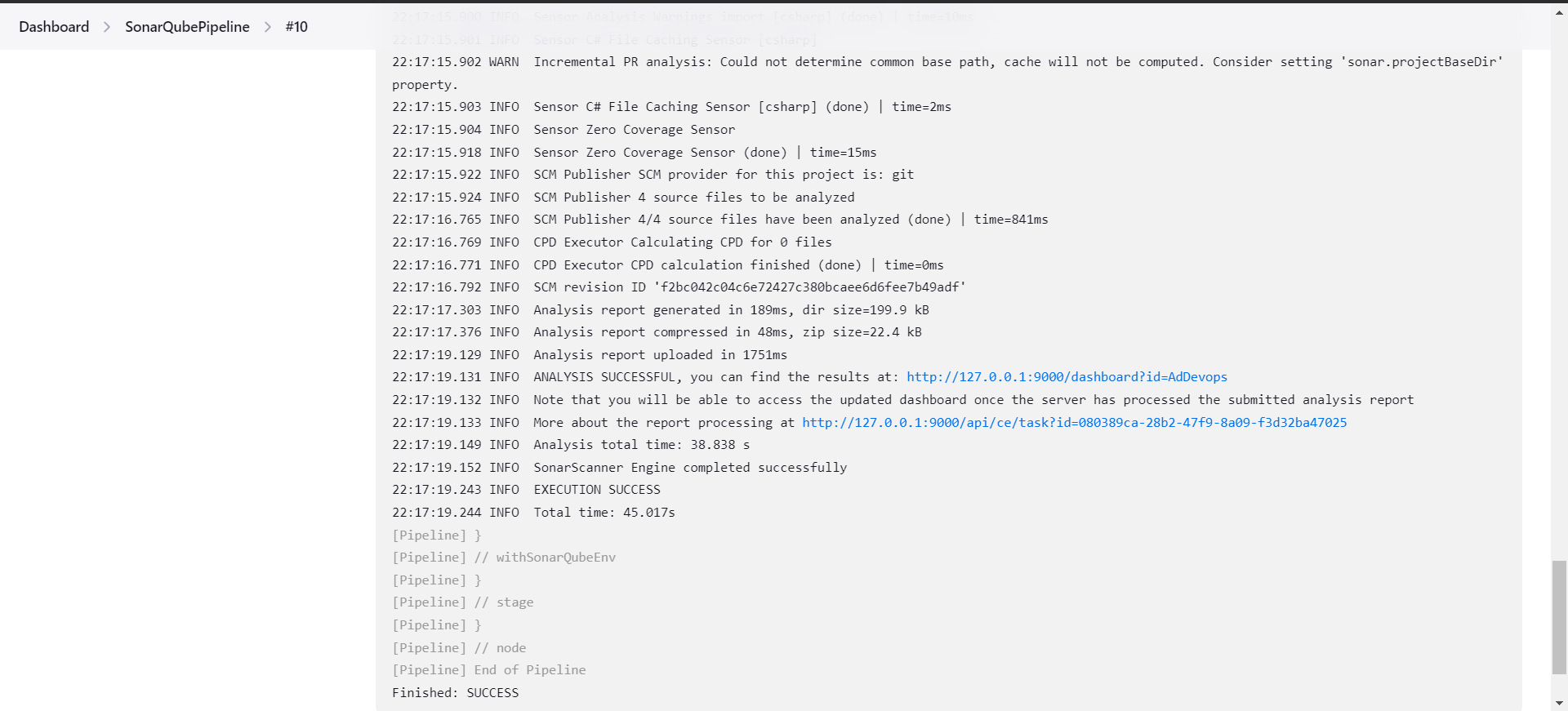
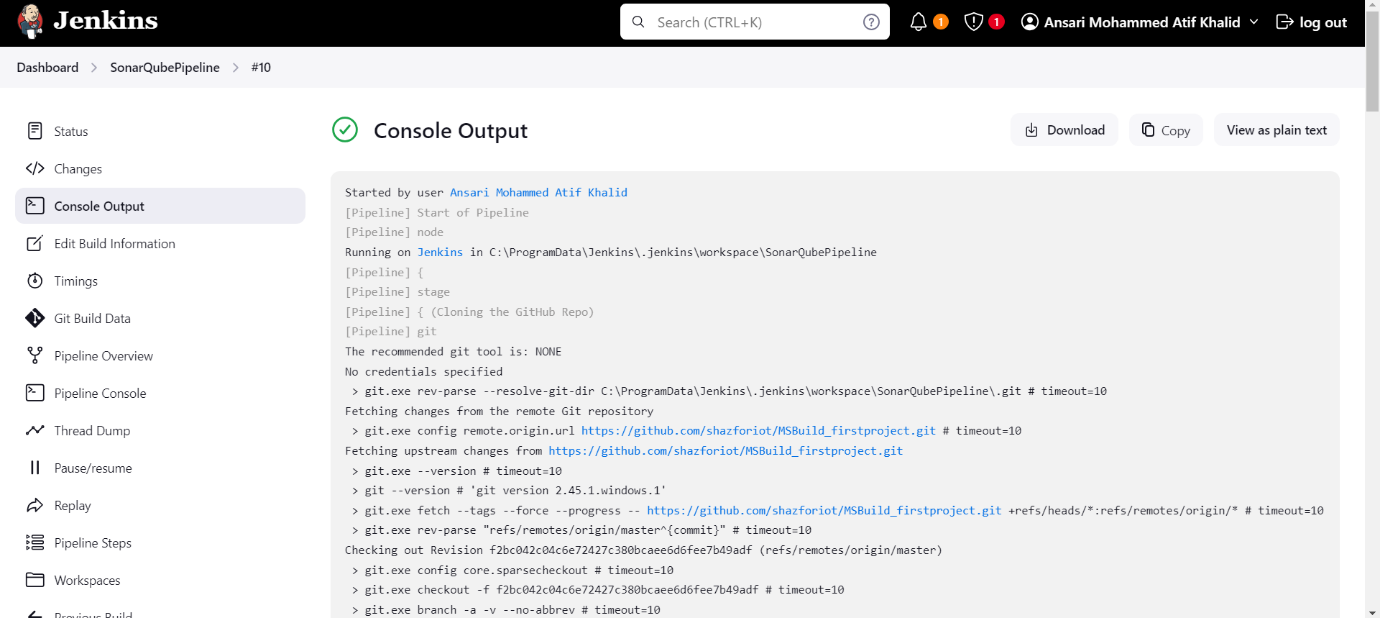
It is a java sample project which has a lot of repetitions and issues that will be detected by

SonarQube.

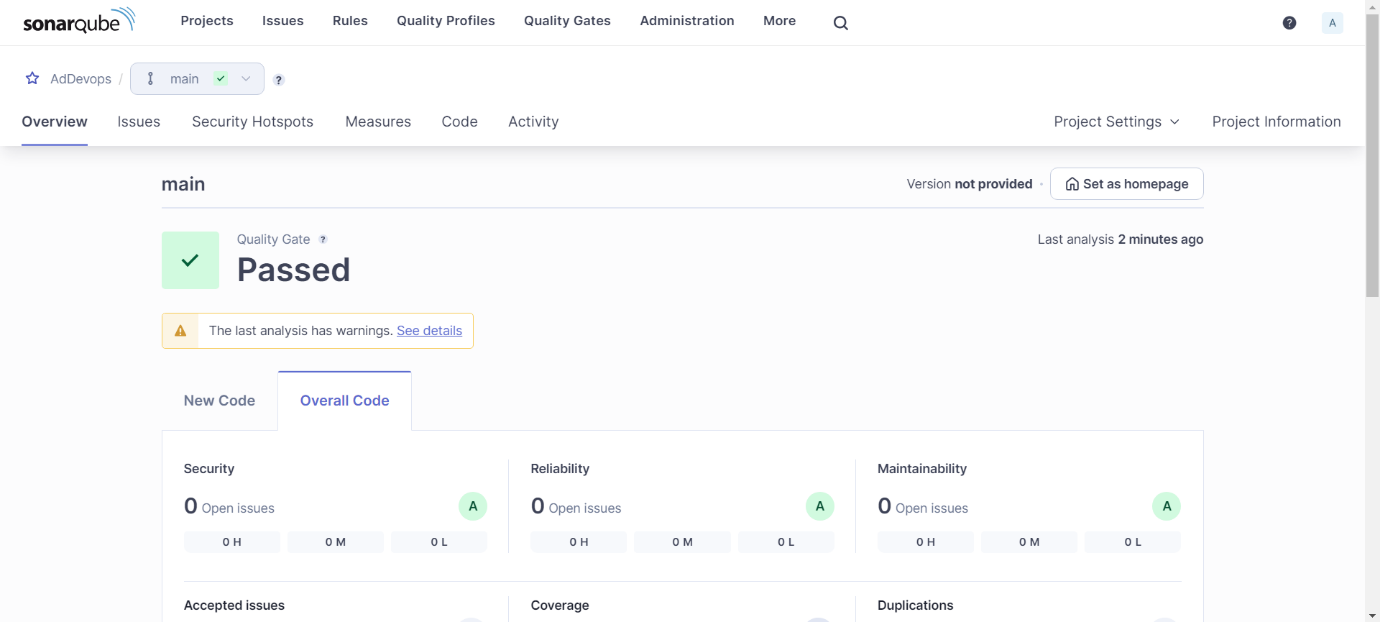
Build and run:



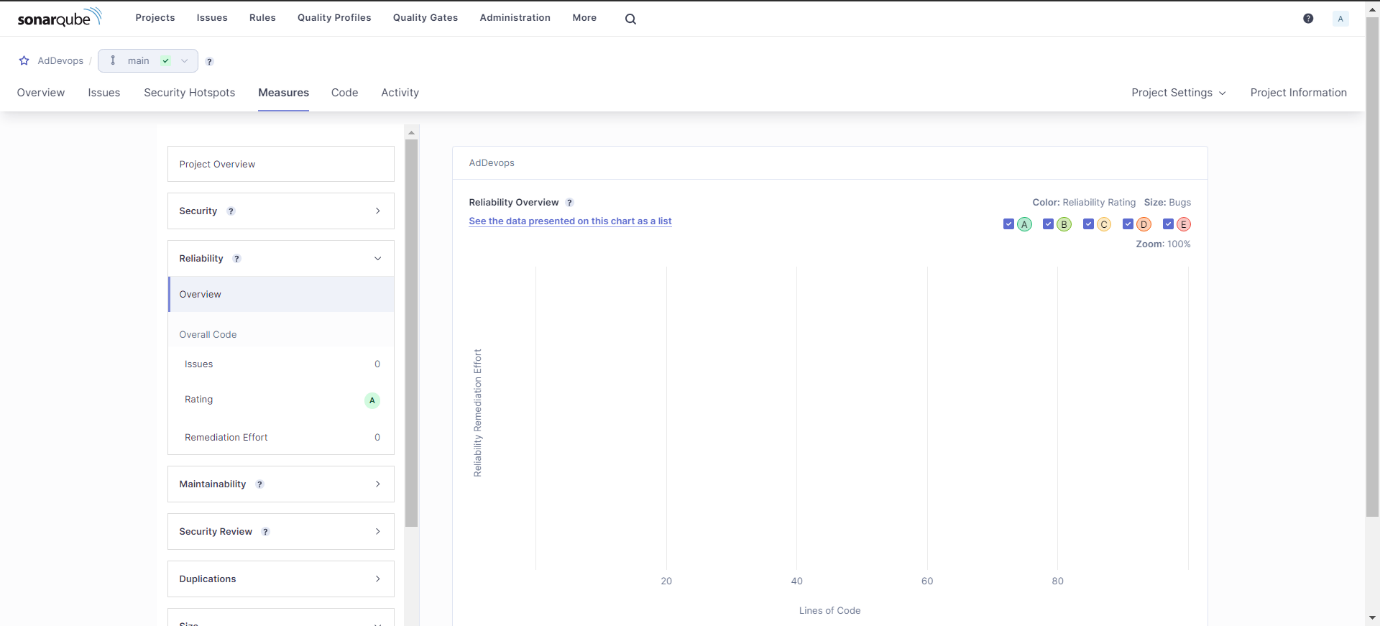
Console output:



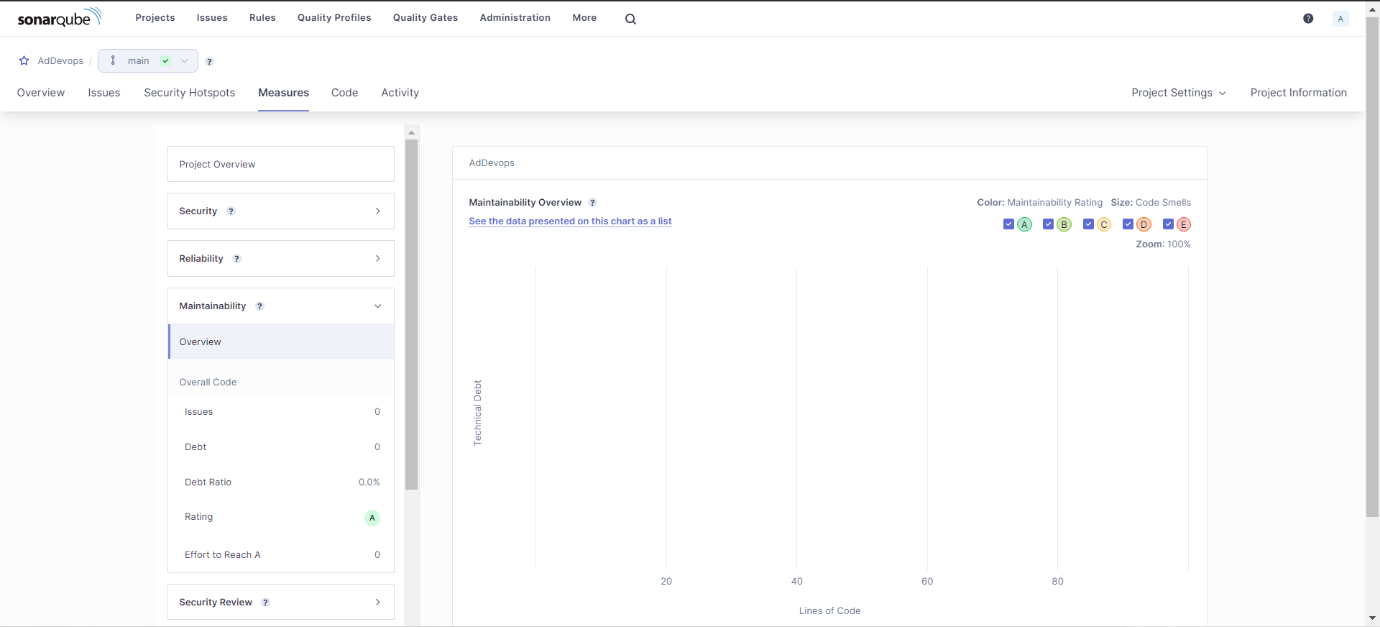
sonarqube:



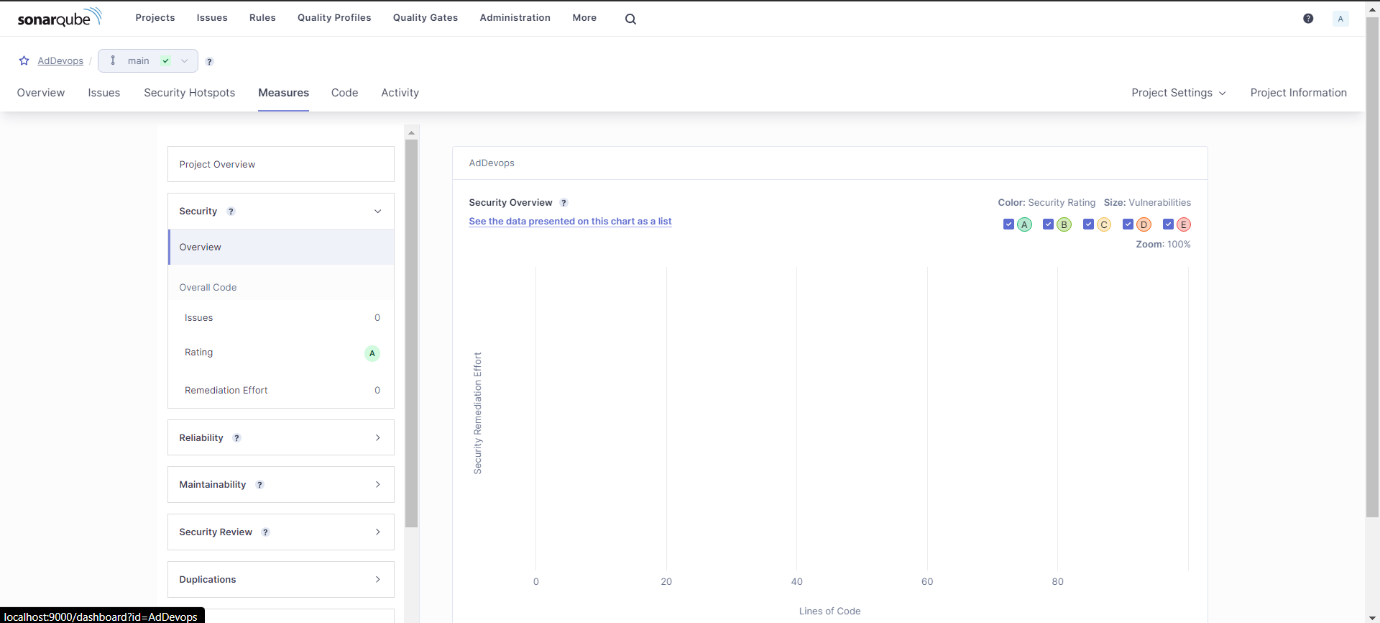
**Reliability:**



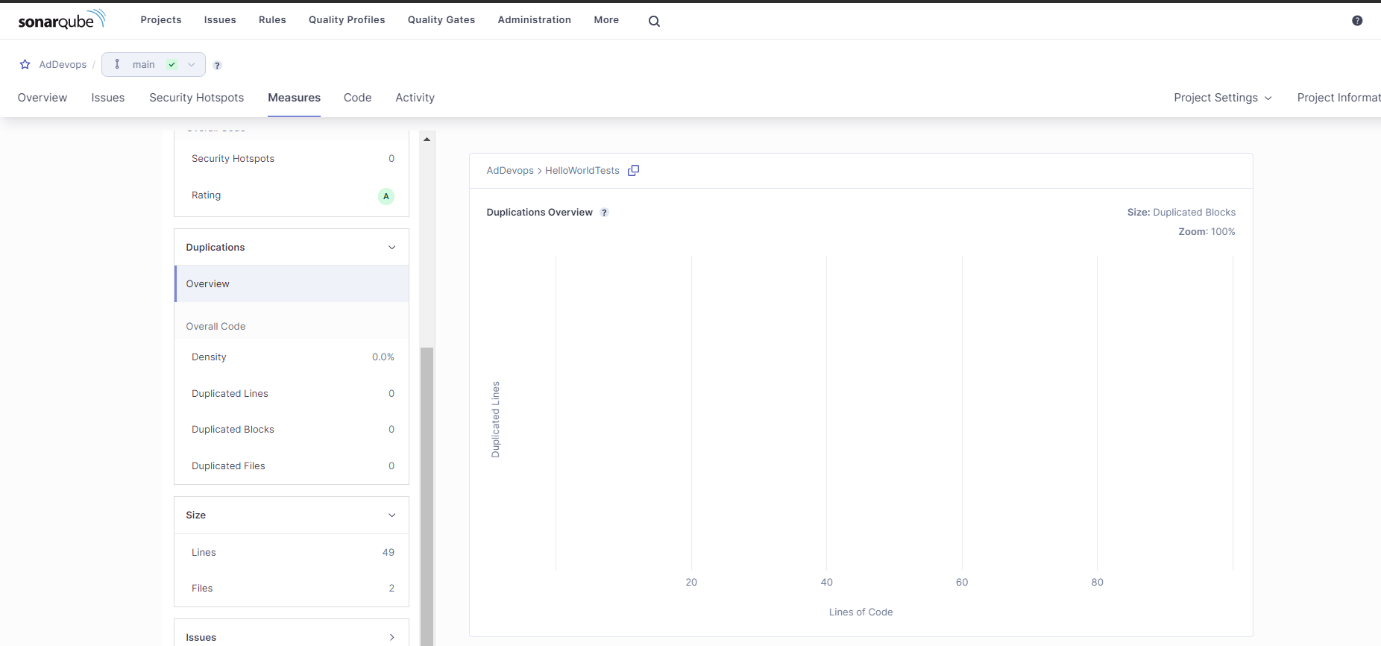
**Maintanaibility:**



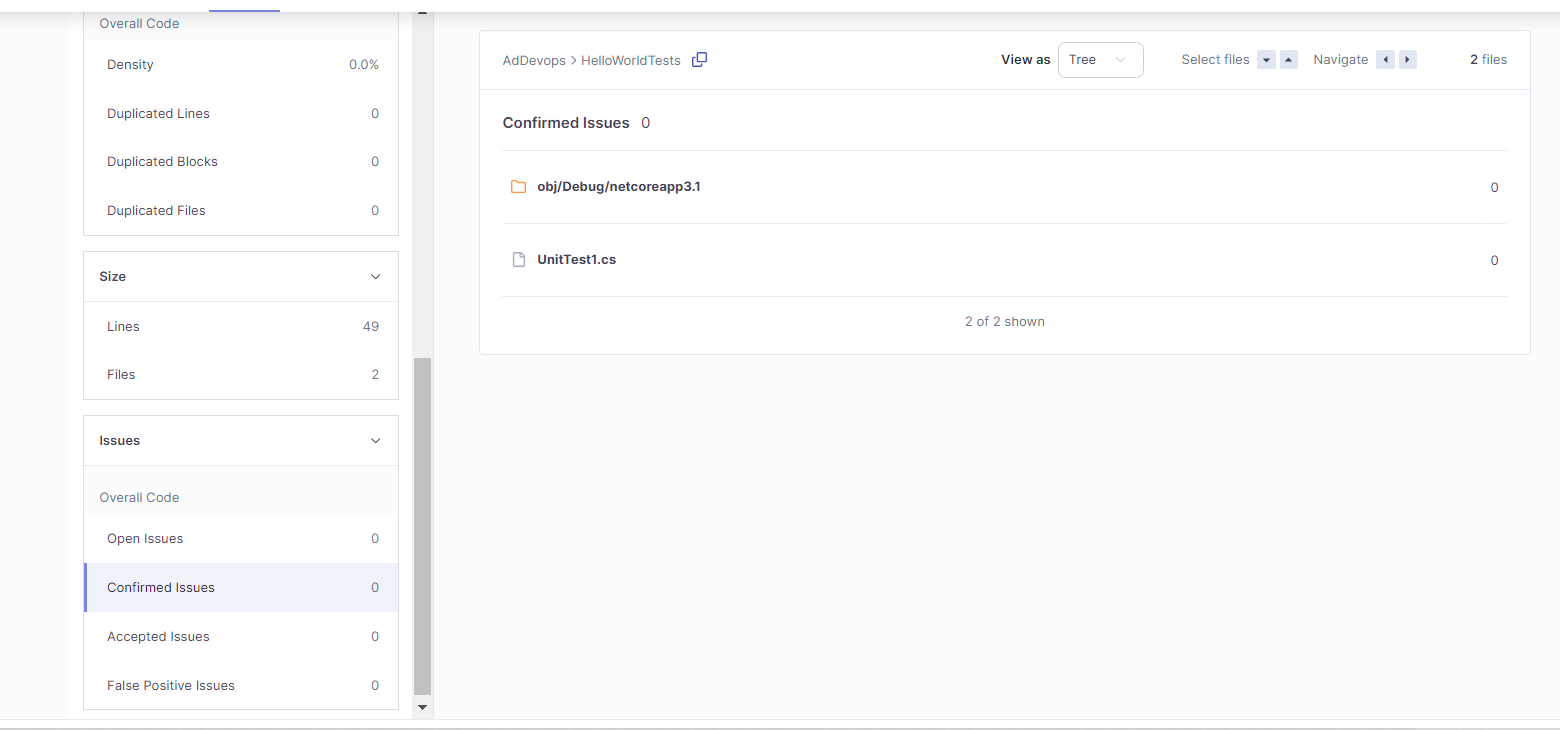
**Security:**



**Duplications:**



**Issues:**



**Conclusion:** Thus, we have successfully integrated Jenkins with SonarQube.