Tehnoloogilise võla mõõtmine (SonarQube)

Tarkvaralahenduste tehnoloogilise võla mõõtmiseks kasutame täna ja edaspidi Sonar tarkvara *Quality gate* konfiguratsioonis määratletud mõõdikuid. Kasutada tuleb **SonarQube Quality Gate** mõõdikut, mis on ka vaikimisi versioon (**Sonar way**). Projektid mis on rohelised, läbivad edukalt SMIT arendusvaldkonna poolt kehtestatud alampiire. Kvaliteedimõõdikut mitteläbivad lahendused tuleks üle vaadata ja teha vastavates kohtades parandusi, kas sonari raportite alusel parandada koodi või kirjutada täiendvaid teste. Allpool on lahti kirjutatud kõik erinevad mõõdikud, millest SMIT-i kvaliteedimõõdik koosneb ning millistel kuidas rakendus nendele vastama peab.

Teenuse kasutusjuhend: SonarQube kasutamine

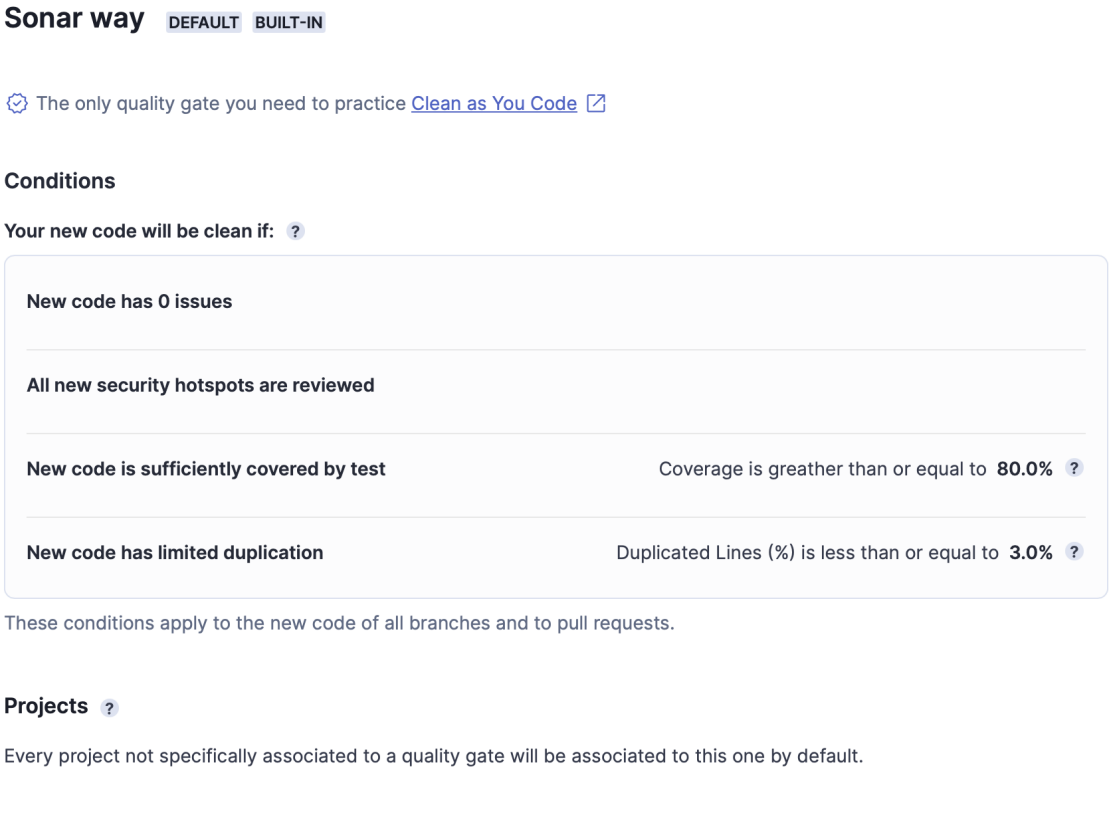
SonarQube roadmap: https://portal.productboard.com/sonarsource/3-sonarqube/tabs/12-planned-for-10-6

Sonaris kasutame projekti KEY nimekujuna "ee.smit.<mskxx>.<toode>.<komponent/rakendus>". Mskxx on konkreetse meeskonna servicedeski MSK projekti key, aga sidekriipsuta ja väikeste tähtedega.

Alates 2024 kasutame uut **Clean as You Code** põhimõtet Sonari analüüsides. Lühidalt paneme fookuse uue ja muudetava koodi kvaliteedi mõõtmisele, mitte sellele mis on juba kunagi kirjutatud. Täpsemalt saab selle kohta lugeda: https://docs.sonarsource.com/sonarqube/10.6/user-guide/clean-as-you-code/. Uus mõõdik analüüsib loodud või muudetud koodi viimase **90 päeva** ajaaknas . Uue projekti lisamisel tuleb esmane skaneerimine teha võimalikult minimaalse koodi pealt, muidu see ei rakendu kui uus kood kvaliteedimõõdiku all ja läheb koheselt "Overall" koodi alla. **Suurim muudatus on 80% automaattestide kattuvuse nõue.**

NB! Sonar ei analüüsi kas teie lahenduse Java/PostgreSQl/Spring Boot/Hibernate/Grails või mõni muu tehniline komponent/raamisitk on aegunud ja nö "võlas". Seda peaksite jälgima ise vastava komponendi lehelt, kus arendus toimub. Tulevikus võtame selle paremaks haldusesk kasutusele eraldi tarkvara.

Kõikide sonaris analüüsitavate tarkvarade ülevaade, kuidas nad vastavad kvaliteedimõõdikule, on leitav siit: https://sonar.smit.sise



Coverage

Mõõdik määrab, mitu protsenti kogu lahenduse lähtekoodist on kaetud automaattestidega. Meil on määratud kriitiliseks alampiiriks **80%.** Selleks et efektiivselt tõsta kaetud koodi mahtu tasub kirjutada rohkem kõrgema taseme teste (integratsioonitestd ja funktsionaaltestid) ja lasta Sonar analüsaatoril koguda tulemusi kaetud ridade osas. Siin aitab kaasa näiteks Jacoco plugina kasutamine, mis antud info Sonarile edasi annab. Kuna pull-requestide üleandmisel on vajali kontrollida Sonari kvaliteedimõõdikut, tuleb testida käivitada iga ehituse käigus.

Duplicated Lines (%)

Mõõdik määrab mitu protsenti lähtekoodi koodist võib olla dubleeritud. Kvaliteedivärava läbimiseks on uus ülempiir 3%-i. Dubleeritud kood tekitab täiendavat ja põhjendamatut halduskoormust ning näitab, et koodis on palju refaktooringut tegemata sisulise funktsionaalsuse kasvades.

Maintainability Rating

Mõõdik määrab mitu protsentsi kogu rakendusele kulunud ajast on tekitanud tehnilist võlga *(Code Smell)* koodi stiili ja standarditele vastavuse osas. Lubatud maht on alla 5%-i (**tase A**). Mõõdik annab välja ka info, mitu päeva kuluks koodi muutmiseks, eeldusel, et tegemist on 8 tunnise tööpäevaga.

Reliability Rating

Mõõdik näitab palju on koodis erinevaid vigu. Kuna mõõdame nüüd ainult uut ja refaktooritud koodi ja mitte olemasolevat, siis ühtegi viga koodis ei tohi olla (**tase A**). Täpsema ülevaate saab dokumentatsioonist, mille leiate, kui klikkate pealkirjas olevale lingile. Kood tuleb hoida veavaba, mida Sonar veaks defineeerib. Näiteks kui mõnel meetodil on ABC skoor liiga kõrge, tuleks see koheselt ringi kirjutada, ennem kui sinna tulevikus uut loogikat hakatakse lisama, sest see võib tekitada kas uusi vigu või on konkeetne meetod järgmisele arendajale loetamatu. Mõõdik annab välja ka info mitu päeva kuluks vigade parandamiseks, eeldusel, et tegemist on 8 tunnise tööpäevaga.

Security Rating

Mõõdik näitab palju on turvanõrkuseid rakenduses, mis on Sonari poolt tuvastatud (see ei tähenda et see kataks kõik turvatestide osa ära, aga teatud hulk owasp, cwe jms on välja toodud). Kuna turvanõrkused on oma olemuselt kõrge riskiga vead, ei ole lubatud ühtegi turvanõrkust (**tase A**). Mõõdik annab välja ka info mitu päeva kuluks turvanõrkuste parandamiseks, eeldusel, et tegemist on 8 tunnise tööpäevaga.

Näidiskonfiguratsioon

sonar.properties Java

```
# Required metadata
sonar.projectKey=ee.smit.mskXX.application
sonar.projectName=App

# SonarQube server URL
sonar.host.url=https://sonar.smit.sise

# Path to the source code
sonar.sources=src/main/java,src/main/resources

# Path to the test code
sonar.tests=src/test/java

# Language and encoding
sonar.language=java
sonar.sourceEncoding=UTF-8

# Java binaries and libraries
sonar.java.binaries=build/classes/java/main
sonar.java.libraries=build/libs

# Test report paths
sonar.junit.reportPaths=build/test-results/test,build/test-results/integrationTest
sonar.coverage.jacoco.xmlReportPaths=build/reports/jacoco/test/jacocoTestReport.xml,build/reports/jacoco
/integrationTest/jacocoIntegrationTestReport.xml

# Java version
sonar.java.source=21
sonar.java.target=21

# Exclusions
sonar.exclusions=**/generated/**, **/third-party/**

# Additional analyzers
sonar.yaml.file.suffixes=.yaml,.yml
sonar.json.file.suffixes=.json
```

sonar.properties Vue

```
# Required metadata
sonar.projectKey=your_project_key
sonar.projectName=App

# Path to the source code
sonar.sources=src

# Language and file inclusions
sonar.language=js
sonar.inclusions=**/*.js, **/*.vue

# Encoding of the source code
sonar.sourceEncoding=UTF-8

# Exclude test files if needed
sonar.exclusions=**/node_modules/**, **/*.spec.js, **/*.test.js

# Additional settings for JavaScript/TypeScript
sonar.javascript.lcov.reportPaths=coverage/lcov.info

# Treat .vue files as JavaScript files
sonar.javascript.file.suffixes=.js,.jsx,.vue

# Include CSS/SCSS analysis
sonar.css.file.suffixes=.css,.scss,.vue

# Include HTML analysis
sonar.html.file.suffixes=.html,.vue
```

Execute SonarQube code analysis.

```
if [[ "${bamboo_planRepository_branch}" == "develop"
|| "${bamboo_planRepository_branch}" == "main" ]];
then
    sonar-scanner -Dsonar.login="${bamboo_sonarTokenSecret}"
-Dsonar.branch.name="${bamboo_planRepository_branch}"
-Dsonar.projectVersion="${bamboo_planRepository_branch}"
  else
    sonar-scanner -Dsonar.login="${bamboo_sonarTokenSecret}"
-Dsonar.pullrequest.key="${bamboo_repository_pr_key}"
-Dsonar.pullrequest.branch=
"${bamboo_repository_pr_sourceBranch}"
-Dsonar.pullrequest.base=
"${bamboo_repository_pr_targetBranch}"
fi
```

Sobiliku image millega Sonarit käivitada leiab siit: https://source.smit.sise/projects/DOCKERHUB

Näiteks: docker.artifacts.smit.sise/dockerhub/eclipse-temurin-ci:21-jdk-alpine-latest