Tere,

Sooviksime täpsustada testikatte protsendiga seotud tehnilist olukorda.   
  
08.07.2025 15: 14 esitatud Pakkuja küsimuse palume jätta tähelepanuta ja Hankija ei ole Sonar valesti seadistanud.

Pakkuja arendaja on unustanud kaasa panna testikattuvuse seadistused. Selgitame alljärgnevalt.  
  
Analüüsides projekti, selgus, et testid on korrektselt implementeeritud, kuid SonarQube konfiguratsioonifailid jäid kogemata projekti GITi lisamata. Pakkuja poolt pakkumuse esitamisel oli antud Pakkuja esindajale info, et testide kattuvused on 81% ja 68% ning samuti esialgses täpsustavas küsimuses anti Pakkuja esindajale sama vastus, kuid asjaolude selgitamisel selgus, et arendaja oli unustanud seadistused GIT versiooni viia, mis moonutab tegelikku testide kattuvust.

**Tegelik vs SonarQube testikate:**

* **Backend:** Maven näitab 81% (tegelik), SonarQube näitab 28.7%
  + Täpsustatud andmetel 81.3% Sonar vaates
* **Frontend:** Karma näitab 68% (tegelik), SonarQube näitab 0%
  + Täpsustatud andmetel 69.6% Sonar vaates

**Lomboki mõju backend testikattele:**

Lombok genereerib automaatselt standardmeetodeid (equals, hashCode, toString, konstruktorid), mida SonarQube loeb testikatte arvutamisel, kuid mida praktikas pole mõtet testida. See vähendab kunstlikult testikatte protsenti umbes 20-30% võrra. Näiteks üks DTO klass võib näidata 15% katet, kuigi kõik äriloogika on 100% kaetud.

**Puuduvad/parandamist vajavad failid:**

**Backend (Spring Boot):**

**1. lombok.config** (projekti juurkaustas)

lombok.addLombokGeneratedAnnotation = true

**2. pom.xml** (JaCoCo, Sonar konfiguratsioon)

Uuendatud backend pom.xml manuses. Hankija saab seda diffida, kuid muutused on seotud ainult Sonar, Lombok seadistusega (käsurea käivitamise jaoks).

**3. application.yaml** (SonarQube seaded ja välistused)

sonar:

coverage:

jacoco:

xmlReportPaths: target/site/jacoco/jacoco.xml

exclusions:

- "\*\*/\*equals\*"

- "\*\*/\*hashCode\*"

- "\*\*/\*toString\*"

- "\*\*/\*canEqual\*"

Enne uue raporti tegemist teha kindlasti puhas build, kuna Lombok genereerib koodi kompileerimise ajal:

mvn clean või mvn clean verify

**Pilt, millel on kujutatud tekst, kuvatõmmis, Font, number

Tehisintellekti genereeritud sisu ei pruugi olla õige.**

**Frontend (Angular):**

**1. karma.conf.js** (LCOV reporter lisamine)

coverageReporter: {

dir: require('path').join(\_\_dirname, './coverage/laptop-request-frontend'),

subdir: '.',

reporters: [

{ type: 'html' },

{ type: 'text-summary' },

{ type: 'lcov' } *// See rida puudus!*

]

}

**2. sonar-project.properties** (projekti juurkaustas)

sonar.projectKey=laptop-request-frontend

sonar.sources=src

sonar.exclusions=\*\*/\*.spec.ts,\*\*/test.ts

sonar.tests=src

sonar.test.inclusions=\*\*/\*.spec.ts

sonar.javascript.lcov.reportPaths=coverage/laptop-request-frontend/lcov.info

sonar.typescript.lcov.reportPaths=coverage/laptop-request-frontend/lcov.info

Pilt, millel on kujutatud tekst, kuvatõmmis, Font, tarkvara

Tehisintellekti genereeritud sisu ei pruugi olla õige.

**Kokkuvõte:**

* Kõik funktsionaalsed testid on implementeeritud ja töötavad ja selgitus **ei lisa teste** pakkumusele
* Backend'i madal protsent tuleneb Lombok genereeritud koodist ja puuduvatest välistustest (näiteks Spring Boot rakenduse main klass)
* Frontend'i 0% tuleneb puuduvast LCOV raportist
* Tegemist on ~<5 konfiguratsioonifaili lisamise/muutmisega
* Koodi kvaliteet ja testide olemasolu vastavad nõuetele
* Kui testide käivitaja avab JaCoCo raporti, siis seal on samuti näha, et backend äriloogika on sisuliselt kaetud testidega

Sisuliselt on hanke testimisnõuded täidetud, puudub vaid tehniline konfiguratsioon SonarQube korrektse raporteerimise jaoks.

Testide kogukattuvus on Pakkuja hinnangul 75.45%.