Static Code Analysis Findings

Before refactoring

SonarQube analysis was done before and after refactoring the codebase. Screenshots were taken of the overview, issues and measures.

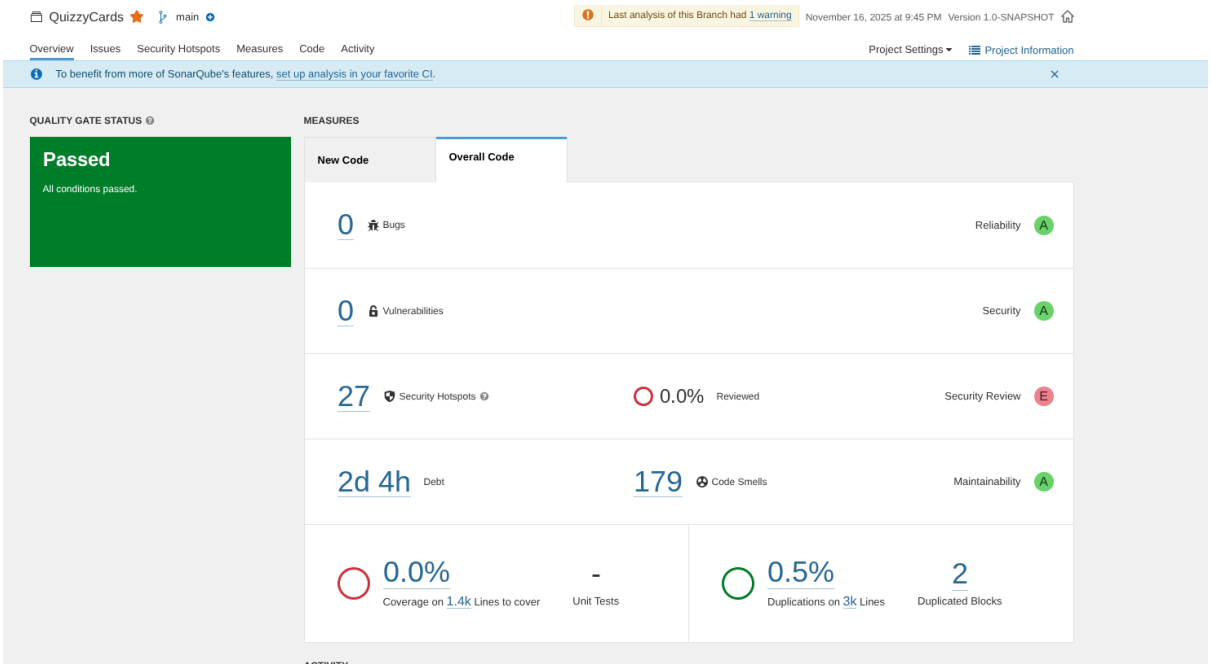
Measures before refactoring:

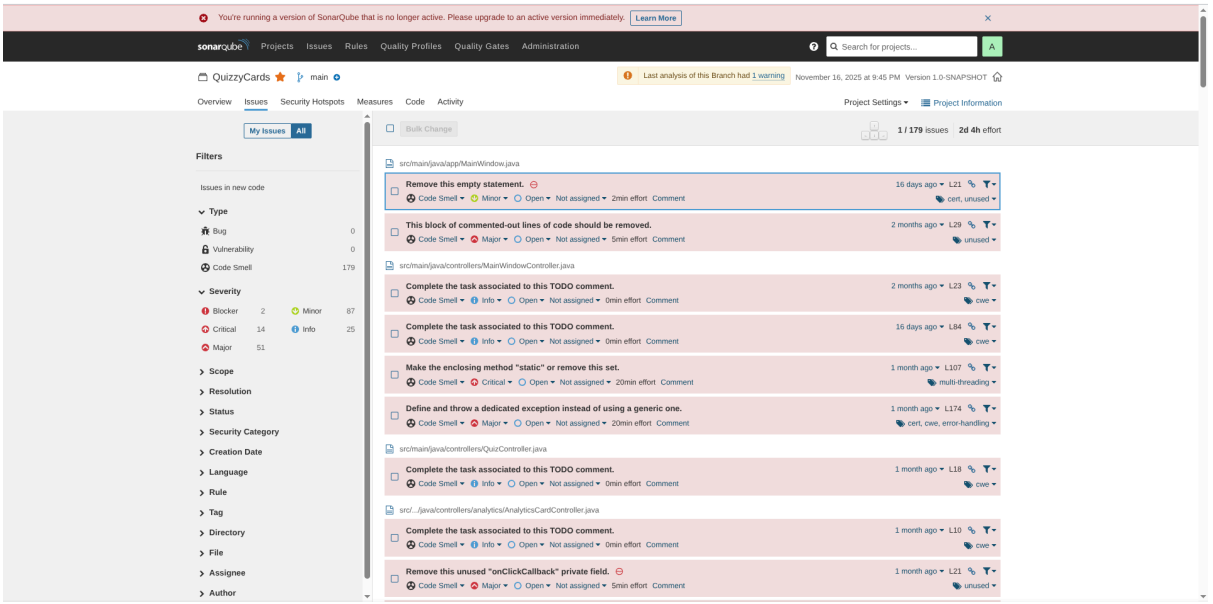
- 179 issues
- 410 cyclomatic complexity
- 11 lines of code per method
- 19 duplicated lines

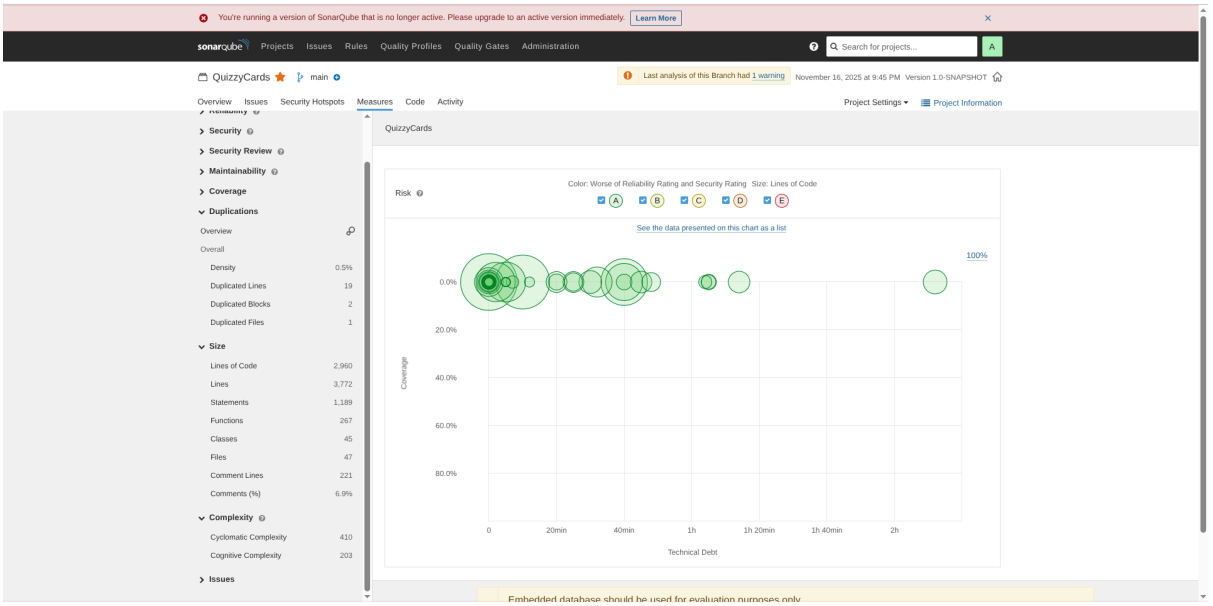
11 lines of code per method is a very good number and cyclomatic complexity is good too. If total cyclomatic complexity is divided by total number of functions (269) to get the average cyclomatic complexity the result is 1.53, which is excellent.

So the biggest improvements that can be made is reducing the number of issues as much as possible without restructuring the codebase too much.

Screenshots before refactoring:







First refactoring

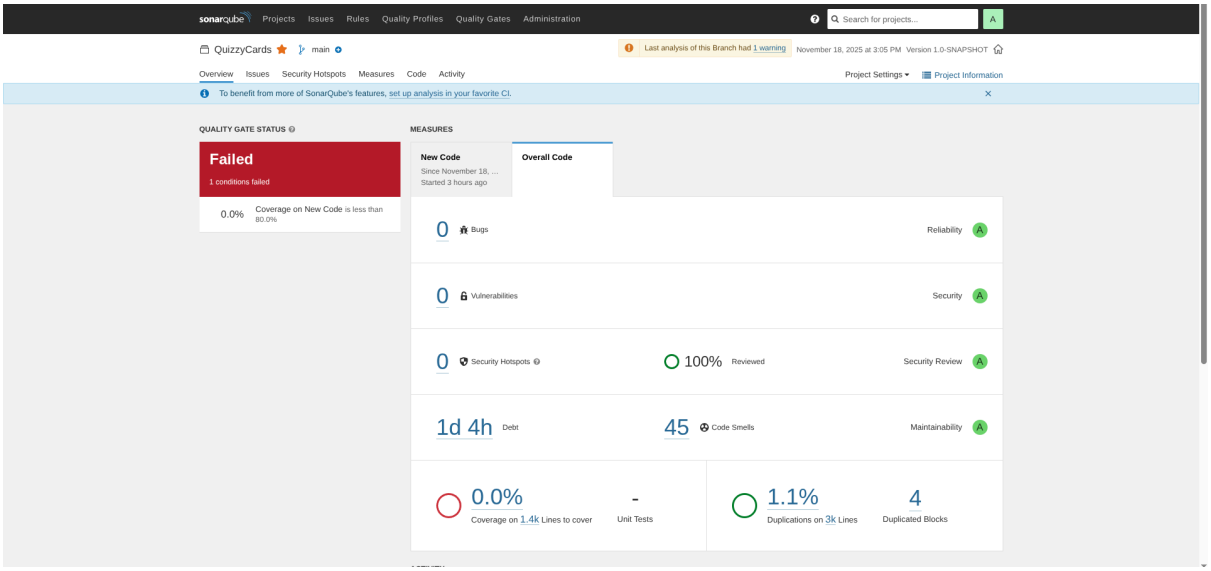
Measures after refactoring:

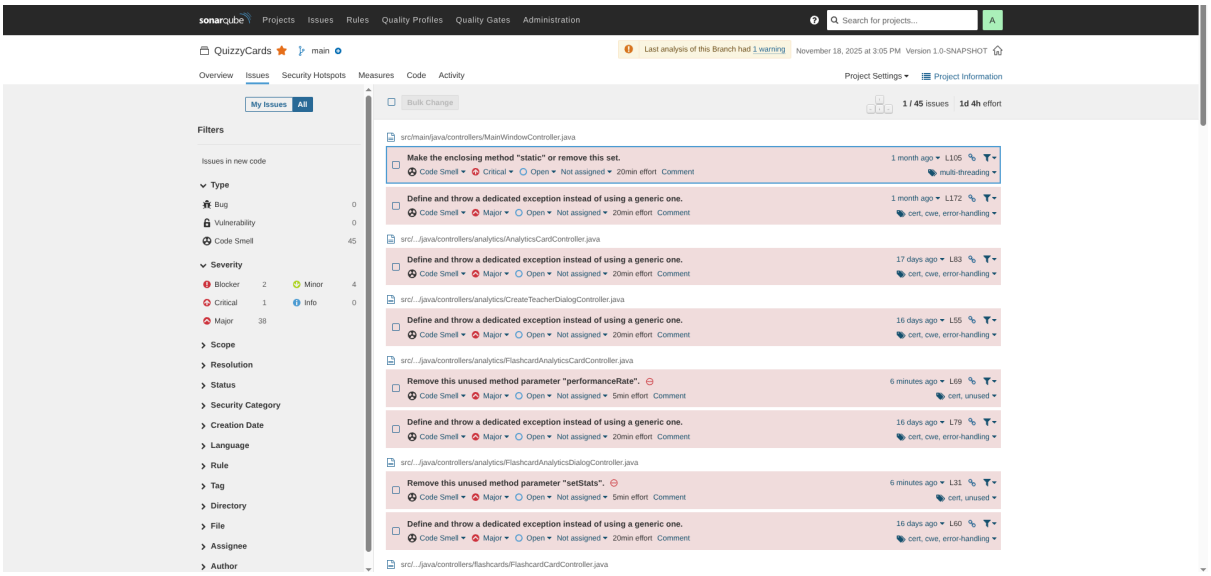
- 45 issues
- 411 cyclomatic complexity
- 11 lines of code per method
- 42 duplicated lines

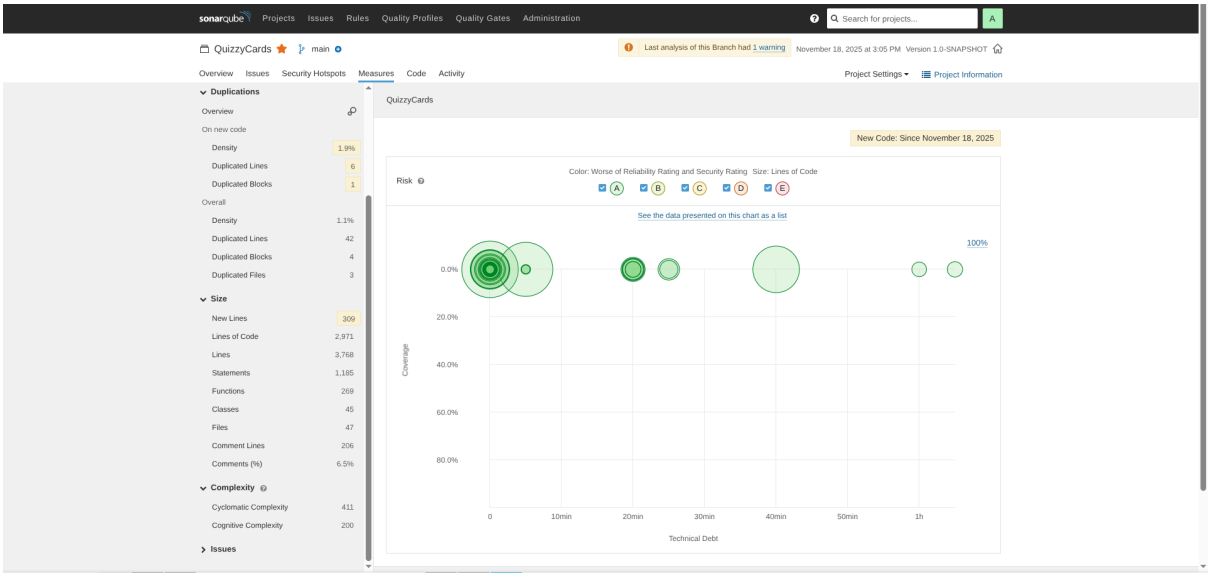
By refactoring the codebase, we removed 134 issues. Some issues remain, but they are not critical issues. Still, we will reduce these to zero.

Duplicated lines increased after refactoring. That needs fixing.

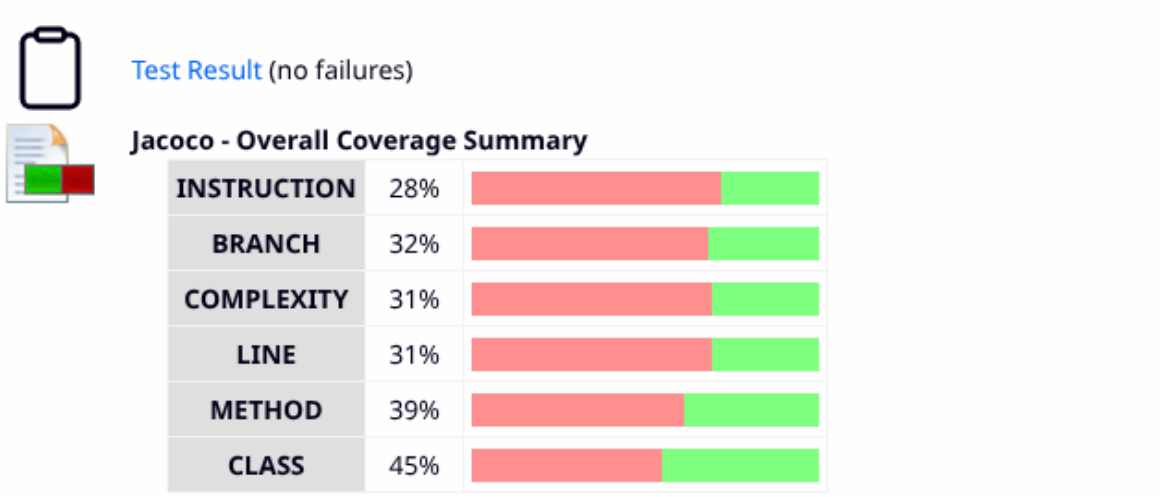
After refactoring (failed due to sonarqube incorrectly thinking coverage is 0, JaCoCo works):







Code coverage that is not shown on sonarqube:

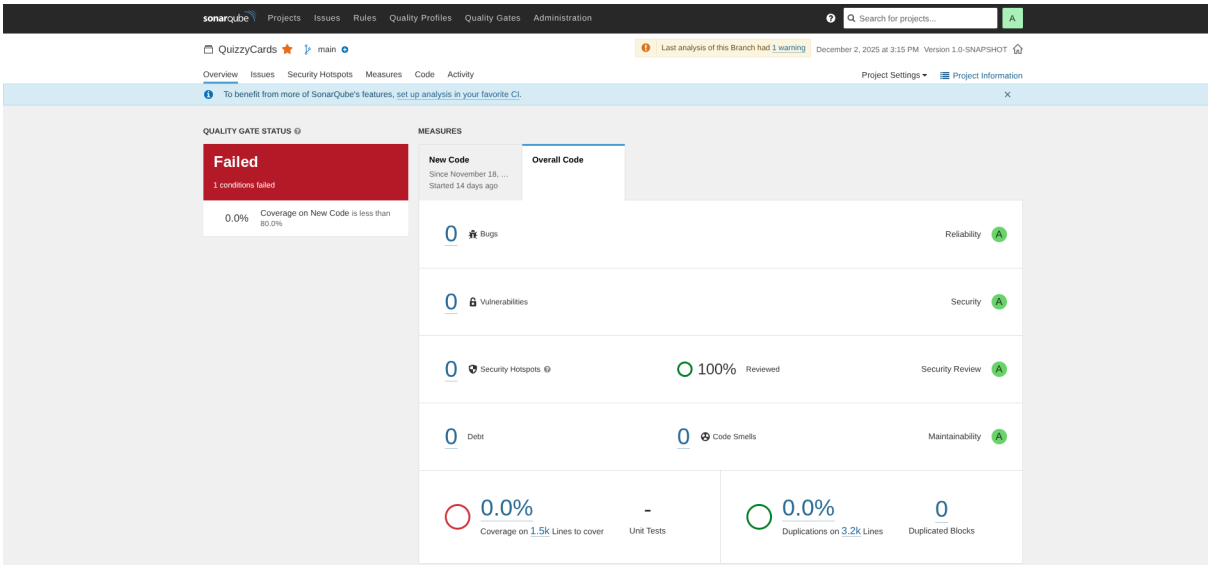


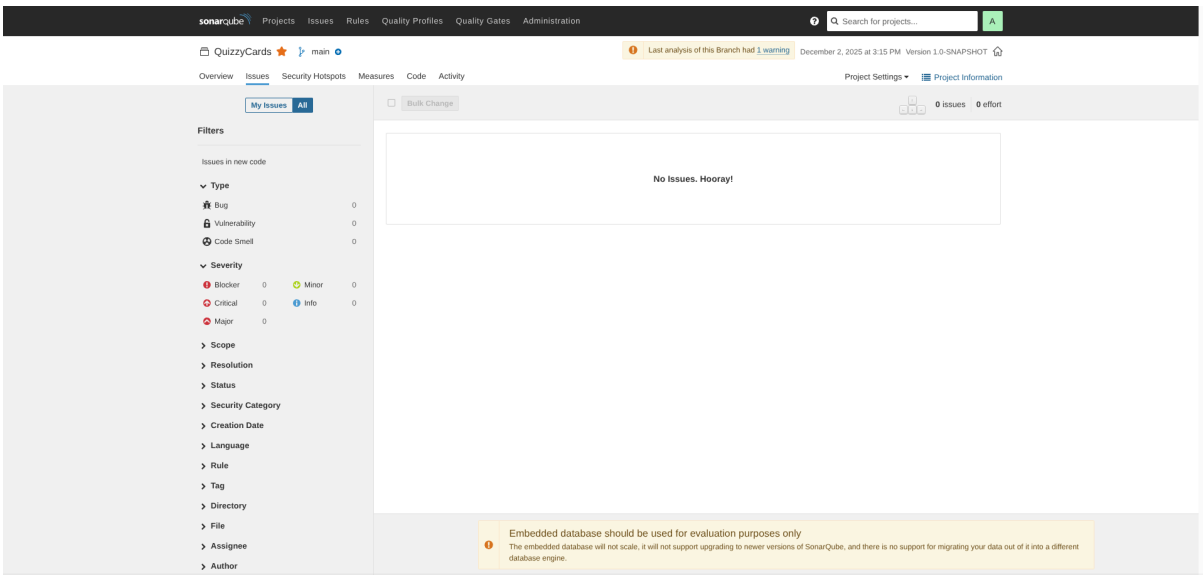
Could be a lot better, but the whole backend has over 90% coverage, frontend has no coverage. Due to this, the results are skewed.

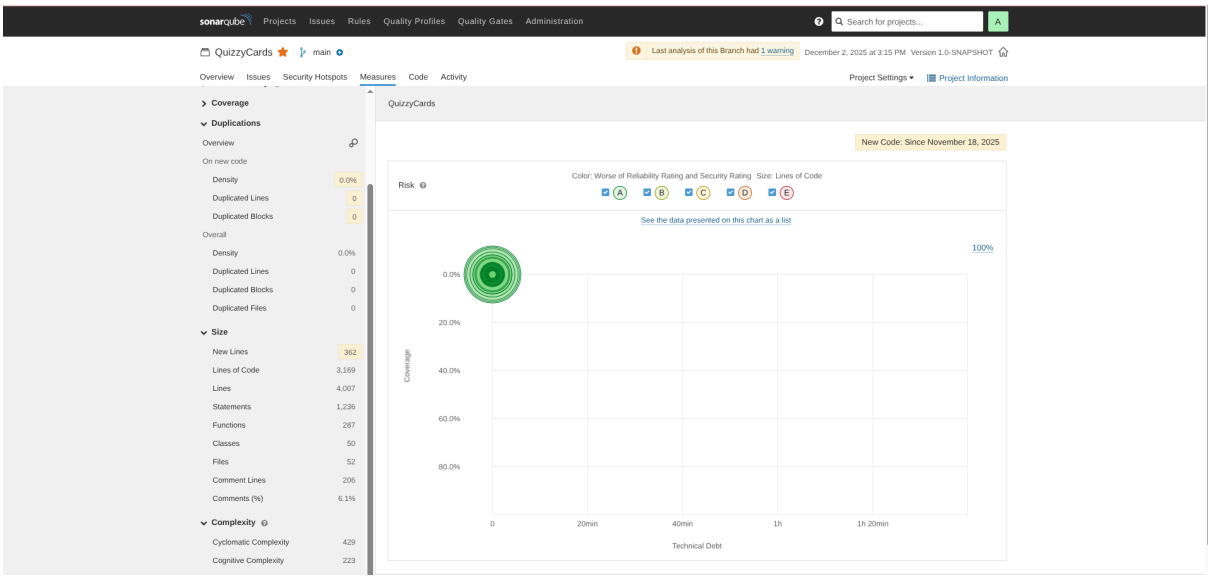
Final refactoring

Measures after final refactoring:

- 0 issues
- 429 cyclomatic complexity
- 11 lines of code per method
- 0 duplicated lines







All issues in the code were fixed, security testing was done and reviewed. Code has no vulnerabilities. All duplications were fixed. SpotBugs tool was also run on the application.

SpotBugs Report

Project Information

Project: QuizzyCards

SpotBugs version: 4.8.3

Code analyzed:

- /home/otto/dev/IdeaProjects/OTP1/target/classes

Metrics

2086 lines of code analyzed, in 50 classes, in 15 packages.

Metric	Total	Density*
High Priority Warnings		0.00
Medium Priority Warnings	26	12.46
Low Priority Warnings	28	13.42
Total Warnings	**54**	**25.89**

(Defects per Thousand lines of non-commenting source statements)*

Contents

SpotBugs report displays 54 warnings, none of which are high priority. The warnings SpotBugs showcased are not critical and a lot of the warnings were false positives.