

Detekt Code Review Tool

Alexandru Stoica

March 2019

1 Introduction

Detekt is a static source code analyser for Kotlin programming language. It finds common programming flaws like unused variables, empty catch blocks, unnecessary object creation, and so forth.

2 Results

Metrics

Data	Value
number of .kt files	43
number of packages	9
number of classes	60
number of functions	103
number of properties	63

2.1 Complexity

Definition 1 (ToManyFunctions) *Too many functions inside a file, class, object or interface always indicate a violation of the single responsibility principle. Maybe the file, class, object or interface wants to manage too many things at once. Extract functionality which clearly belongs together.*

Files:

/report/ReportService.kt:18:1
/report/ReportController.kt:21:1

2.2 Naming

Definition 2 (VariableNaming) *Variable names should follow the naming convention set in the projects configuration.*

Files:

/security/SecurityConstants.kt:8:5
/report/Location.kt:38:9

2.3 Performance

Definition 3 (SpreadOperator) *Using a spread operator causes a full copy of the array to be created before calling a method has a very high performance penalty.*

Files:

/taskly/system/Taskly.kt:14:26

2.4 Style

Definition 4 (NewLineAtEndOfFile) *Checks whether files end with a line separator.*

Files:

/configuration/AspectConfiguration.kt:18:1
/configuration/SwaggerConfiguration.kt:27:1

```
/section/CriticalSectionController.kt
... other 28 files
```

Definition 5 (WildcardImport) *Wildcard imports should be replaced with imports using fully qualified class names. Wildcard imports can lead to naming conflicts. A library update can introduce naming clashes with your classes which results in compilation errors.*

Files:

```
/section/CriticalSectionController.kt:5:1
/section/CriticalSection.kt:10:1
/notification/ExpoNotificationService.kt
... other 9 files
```

Definition 6 (MagicNumber) *Report magic numbers. Magic number is a numeric literal that is not defined as a constant and hence it's unclear what the purpose of this number is. It's better to declare such numbers as constants and give them a proper name. By default, -1, 0, 1, and 2 are not considered to be magic numbers.*

Files:

```
/security/SecurityConstants.kt:10:33
/report/ValidatedReport.kt:13:88
/report/ValidatedReport.kt:31:78
... other 7 files
```

2.5 Complexity Report

3 Advantages

Advantage 1 *Detekt is easy to install and use on any platform and has plugin integration for popular IDEs such as IntelliJ and Eclipse.*

Advantage 2 *The static analyser is under an open-source licence and continues to improve each week; having over 90+ contributors and 2360+ contributions within 3 years.*

Data	Value
lines of code	1468
source lines of code	1177
logical lines of code	820
comment lines of code	20
McCabe complexity	153
number of total code smells	62
mcc per 1000 lloc	186
comment source ratio	1%
code smells per 1000 lloc	75

Advantage 3 *Detekt provides a McCabe complexity report over the logical lines of code within our project, and it is configurable with custom set of rules.*

4 Disadvantages

Disadvantage 1 *Detekt is unable to rank bugs based on their impact on the code base. Some problems are more relevant than others such as `NewLineAtEndOfFile`. Although, every line in a given file should end with character `/n`, this rule is enforced due to legacy `POSIX` tools such as `cat`, and does not count as a "code smell". In total, the new line warnings count to 30% of the total code smells, which may be misleading.*

Disadvantage 2 *The static analysis does not cover aspects such as the detection of objects composed of constant properties, irregular dependencies between packages or classes, detection of hardcoded dependencies between objects and others.*

This makes the overall tool mostly useless because it is unable to detect the subtitle but deadly problems in our code base.