Softwarequalitätssicherung

Dennis Jongebloed, Julian Hinxlage, Johannes Theiner

Projektvorstellung

Fachbereich Technik Abteilung Elektrotechnik und Informatik

23. April 2020





Softwarequalitätssicherung Projektyorstellung

Dennis Jongebloed, Julian Hinxlage, Johannes Theiner

23. April 2020

Qualitätssicherung

Übersicht

Vorstellung

Warum gerade PMD ?

∟Übersicht

1/15





Qualitätssicherung

PMD

University of Applied Sciences

HOCHSCHULE

EMDEN-LEER



Tool zur statischen Analyse von Java, JavaScript, Salesforce.com, Apex, Visualforce, PLSQL, Apache Velocity, XML und XSL Anwendungen.



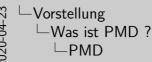
PMD

University of Applied Sciences

HOCHSCHULE

FMDFN-I FFR

Enthält zusätlich einen CPD(copy-paste-detector) mit Unterstützung von Java, C, C++, C#, Groovy, PHP, Ruby, Fortran, JavaScript, PLSQL, Apache Velocity, Scala, Objective C. Matlab, Python, Go, Switch, Salesforce.com, Apex und Visualforce.



Enthält zusätlich einen CPD(copy-paste-detector) mit Fortran JavaScrint PLSQL Anache Velocity Scala Objective C Matlab. Python. Go. Switch. Salesforce.com. Apex und Visualforce

Codierungskonvention und die Kommentare sind oft hilfreich

Warum PMD?

Die Dokumentation von PMD ist sehr übersichtlich gestaltet, die Beschreibungen sind umfangreich. Der Code besitzt eine gute Codierungskonvention und die Kommentare sind oft hilfreich.

-Warum gerade PMD ? 2020-04-└─Warum PMD ? └─Warum PMD ?

ichtlinien
onto Stripe
onto Stripe
onto Stripe
onto Stripe
onto Tests
onto Stripe
onto Stripe
onto Tests
onto Stripe
onto Stri

Richtlinien

Um einen erfolgreichen Pullrequest zu erstellen muss:

- b dieser gegen den master Branch erstellt werden
- ▶ ohne Fehler gebaut werden können

Richtlinien

Um einen erfolgreichen Pullrequest zu erstellen muss:

dieser gegen den master Branch erstellt werden
 ohne Fehler gebaut werden können

linien nuous Integratio Style Tests & CPD cy

Travis CI



PMD nutzt das Tool Travis CI für die Continuous Integration.
Dabei wird der Code automatisch compiliert, wenn ein neuer Commit ins Repository gepusht wird.

E740 Continuous Integration

☐ Travis CI

Travis CI



PMD nutzt das Tool Travis CI für die Continuous Integration. Dabei wird der Code automatisch compiliert, wenn ein neuer Commit ins Repository gepusht wird.

Der Compiler Vorgang wird auf einem Server ausgeführt. In einer Übersicht wird angezeigt, ob ein Build erfolgreich war. Der CI Server führt auch andere nachfolgende Qualitätssicherungs-Tools aus.

Richtlinien Continuous Integrat Code Style Unit Tests PMD & CPD Codacy SonarCloud

Code Style



Um einen einheitlichen Stil beizubehalten wird Checkstyle verwendet.

Bewertet werden: Klammersetzung, Importreihenfolge, Leere Zeilen, etc.

Code Style

Code Style

Code Style

Code Style

checkstyle

Um einen einheitlichen Stil beizubehalten wird Checkstyl verwendet.

Bewertet werden: Klammersetzung, Importreihenfolge, Leere Zeilen etc.

Code Style Beispiel

```
/**
             * Adapter for the JavaParser, using the specified grammar version.
             * Cauthor Pieter Van Raemdonck - Application Engineers NV/SA - www.ae.be
             * Qauthor Andreas Dangel
             * Odeprecated This is internal API, use {Olink LanguageVersionHandler#getParser(ParserOptions)}.
             */
             @InternalApi
             @Deprecated
             public class JavaLanguageParser extends AbstractJavaParser {
                 private final int idkVersion:
                 private final boolean preview;
                 public JavaLanguageParser(int jdkVersion, ParserOptions parserOptions) {
                     this(jdkVersion, false, parserOptions);
                 public JavaLanguageParser(int jdkVersion, boolean preview, ParserOptions parserOptions) {
                     super(parserOptions);
                     this.idkVersion = idkVersion:
                     this.preview = preview;
Dennis Jongebloed, Julian Hinxlage, Johannes Theiner
                                                                                     23. April 2020
                                                                                                                                        8/15
```

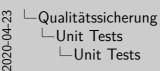
-Qualitätssicherung └─Code Style 2020-04 └Code Style Beispiel

Code Style Beispiel

Supramusi philis sines derdangagabener artesis Stateachlesduraer princia final ins jühlesin; princia final hullan merine

Unit Tests

Um den Code zu testen werden Unit Tests angelegt. Units Tests testen die Funktionalität von Teilen des Sourcecodes. Dabei werden in einer Testfunktion die zu testende Funktionen aufgerufen und die Ergebnisse überprüft. Auch ganze Klassen und zusammenhängende Funktionsaufrufe können getestet werden.



Unit Tests

Um den Code zu testen werden Unit Tests angelegt. Units Tests testen die Funktionalität von Teilen des Sourcecodes. Dabei werde in einer Testkinktion die zu testende Funktionen aufgerufen und die Ergebnisse überprüft. Auch ganze Klassen und zusammenhängende Funktionsaufrufe können getestet werden.

Unit Test Beispiel

Dieser Unit Test überprüft beispielsweise ob der C++ Parser mit Unicode Zeichen zurechtkommt. Dazu wird C++ Code in Form eines Strings angelegt. Der Code enthält auch Unicode Zeichen. Dann wird überprüft ob der Parser den Code richtig geparst hat.

© Unit Test Beispiel

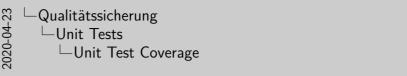
Unit Test Beispiel

Dieser Unit Test überprüft beispielsweise ob der C++ Parser mi Unicode Zeichen zurechtkommt. Dazu wird C++ Code in Form eines Strings angelegt. Der Code enthält auch Unicode Zeichen. Dann wird überprüft ob der Parser den Code richtig geparst hat

Unit Test Coverage

▶ — 0.97 pmd-go/
 ▶ — 63.64 pmd-groovy/
 ▶ — 81.71 pmd-java/
 ▶ — 27.72 pmd-javascript/
 ▶ — 59.94 pmd-jsp/
 ▶ — 81.82 pmd-kotlin/

Die Code Coverage gibt an wie viele Zeilen des Codes durch die Unit Tests abgedeckt werden. Die Code Coverage wird automatisch mit den Tool Coveralls ermittelt.



Unit Test Coverage

a Francisco

Aktuell ist 45% des Codes über Unit Test abgedeckt. Dabei haben die unterschiedlich Sprachen unterschiedlich viel Code Coverage. Java hat z. B. 81.71% coverage, JavaScript hat 27.72% und Swift nur 0.27%.

htlinien
ntinuous Integration
le Style
t Tests
D & CPD
lacy
harCloud

CPD & CPD & CPD & CPD



PMD nutzt die eigenen Features um statische Analysen des Codes durchzuführen.

PMD & CPD



PMD nutzt die eigenen Features um statische Analysen des Codes durchzuführen.

Richtlinien Continuous Integration Code Style Unit Tests PMD & CPD Codacy SonarCloud

Codacy



Codacy ist eine SaaS Anwendung welche die Codequalität und Sicherheit anhand statischer Analysen bewertet. © Codacy
Codacy
Codacy
Codacy

Codacy



Codacy ist eine SaaS Anwendung welche die Codequalität und Sicherheit anhand statischer Analysen bewertet.

htlinien ntinuous Integrat de Style it Tests ID & CPD dacy narCloud

? Unit Ig PMI Cod Son:

SonarCloud



SonarCloud ist eine Cloud basierte Lösung für Softwarequalitätsanalysen, Verlässlichkeit und Sicherheit, Schwachstellen, schlecht strukturierter Code und Fehler werden automatisch gefunden. Es wird für mehr als 20 Sprachen angeboten. © CQualitätssicherung
USonarCloud
USonarCloud

SonarCloud

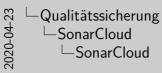


SonarCloud ist eine Cloud basierte Lösung für Softwarenvalitäte analyser /erlässlichkeit und Sicherheit, schwachstellen, schlecht trukturierter Code und Fehler verden automatisch gefunden. is wird für mehr als 20 Sprache ingeboten. ichtlinien ontinuous Integration ode Style nit Tests MD & CPD odacy onarCloud

SonarCloud



Bei der SonarCloud Analyse von PMD sieht man, dass 77 Sicherheitslücken bei der letzten Analyse aufgetreten sind.





Allerdings findet sich SonarCloud nur in Konfigurationsdateien nicht in der Dokumentation, daher wissen wir nicht ob SonarCloud aktiv verwendet wird. Das Bild wurde am 19.03.2020 17:00 Uhr aufgenommen.