

Como manter a qualidade de código sob pressão?

Henrique Schmidt

@henriquels25

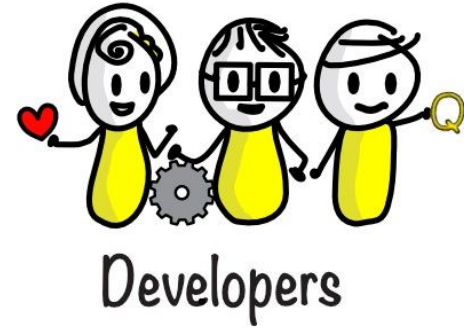
henriquels25.github.io

Agenda

- Papéis em um time de produto
- Qualidade interna vs externa
- Débito Técnico
- Análise Estática de Código
- Sonarqube
- Conclusão

**Papéis em um time de
produto**

Scrum Team



Product Owner

Tem o objetivo de maximizar o valor gerado pelo time

Product Owner

Entende do negócio, cria e prioriza histórias de usuário

Time de desenvolvimento

Tem o objetivo de entregar software em produção que gere valor para o negócio

De onde vem a pressão para entregas rápidas?

Qual a hierarquia entre o PO e o time de desenvolvimento?

Qual o papel é o principal responsável por garantir a qualidade do código?

Qualidade externa x interna

Qualidade externa



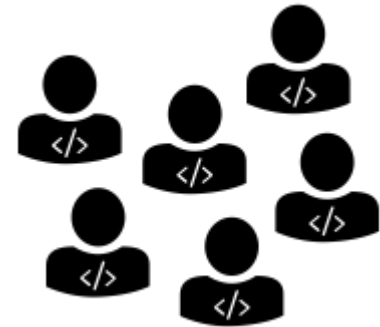
Afeta diretamente os usuários



Qualidade interna



Afeta diretamente o time de desenvolvimento



Qualidade externa

- O software funciona corretamente conforme as necessidades dos usuários;
- É robusto e reage bem a situações adversas;
- Fácil utilização pelos clientes;
- A performance é adequada para a necessidade;
- Entre outras características...

Qualidade externa

“A qualidade externa determina o cumprimento dos requisitos dos usuários.”

Mike Long

[@meekrosoft](#)

[Internal vs External Software Quality](#)



Qualidade interna

- Código sem duplicação;
- Código coeso: Cada classe/módulo faz uma coisa e faz isso bem;
- Baixo acoplamento: Mínimo número de dependências entre objetos;
- Simplicidade e legibilidade do código;
- Entre outras características...

Qualidade interna

“A qualidade interna determina a facilidade de um time na evolução de um projeto.”

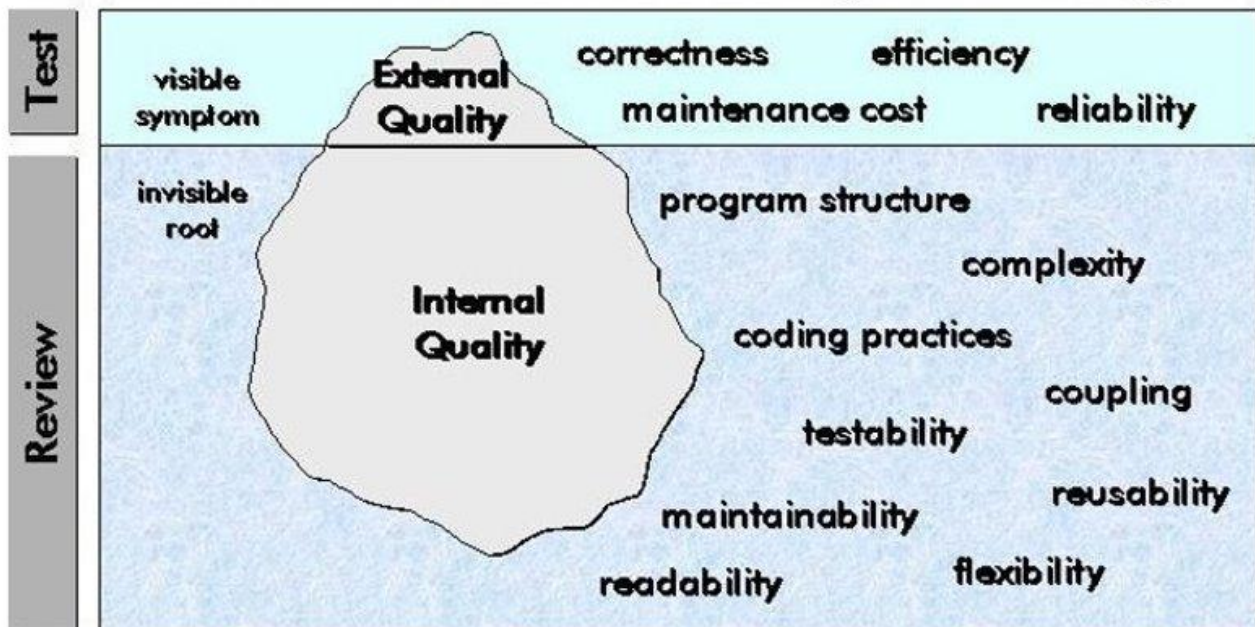
Mike Long

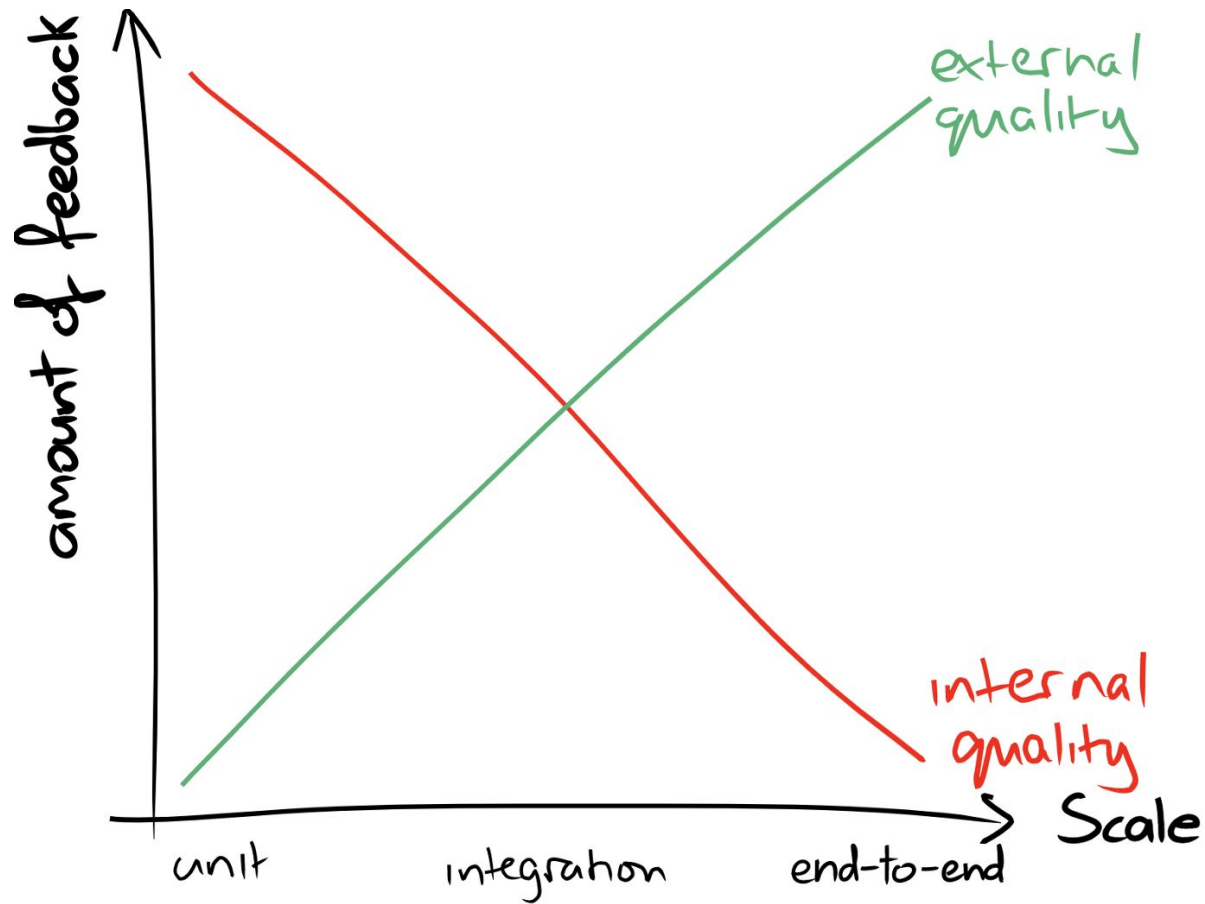
[@meekrosoft](#)

[Internal vs External Software Quality](#)



The Software Quality Iceberg





Steve Freeman e Nat Pryce
[Growing Object-Oriented Software,
Guided by Tests](#)

Débito Técnico

Débito Técnico

Metáfora criada em 1992
por Warn Cunningham



Autor da primeira Wiki e co-autor
do Manifesto Ágil

Débito Técnico

Problemas na qualidade interna funcionam
como juros em uma dívida

Débito Técnico

“Assim como dívidas financeira, ter um pouco de débito técnico (juros) não é um problema. O problema existe quando não há o pagamento deles”

Débito financeiro

Obter o
produto agora

Pagamento
de juros

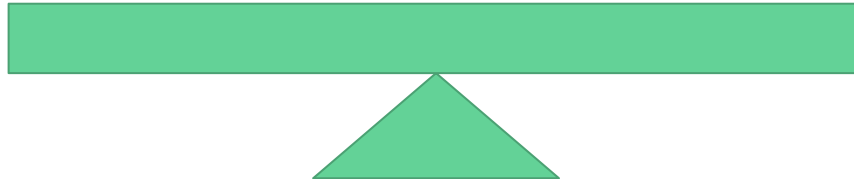
Impossível de
pagar a dívida

Débito técnico

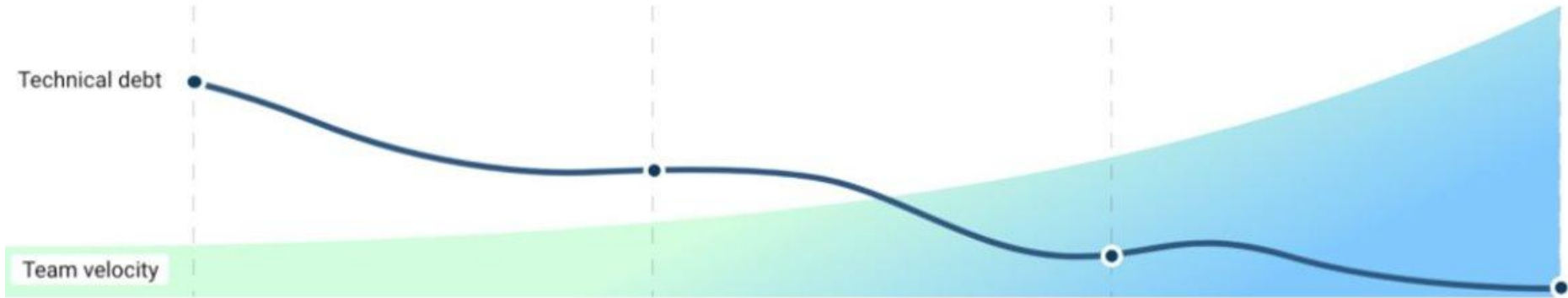
Entregar mais
rápido

Refatoração

Impossível de
manter o
projeto







<https://www.sonarqube.org/features/clean-as-you-code/>

Análise Estática de Código

Análise Estática de Código

O código fonte é analisado, sem ser executado, com uma lista de regras pré-determinadas e métricas são extraídas.



Benefícios

- É possível encontrar bugs automaticamente;
- Auxílio para code review;
- Identificação de problemas de segurança automaticamente;
- O time pode possuir uma métrica da **qualidade interna** do seu código.

Ferramentas



Sonarqube

Sonarqube

Plataforma open-source para inspeção de código-fonte utilizando análise estática de código.



Bugs

Issue que representa algo errado no código-fonte. Se ainda não causou um problema em produção, provavelmente irá causar. Deve ser ajustado.



Code Smells

Códigos que provavelmente fazem o que deve fazer, mas são difíceis de manter. Eles serão tão confusos para futuros devs que podem causar bugs.

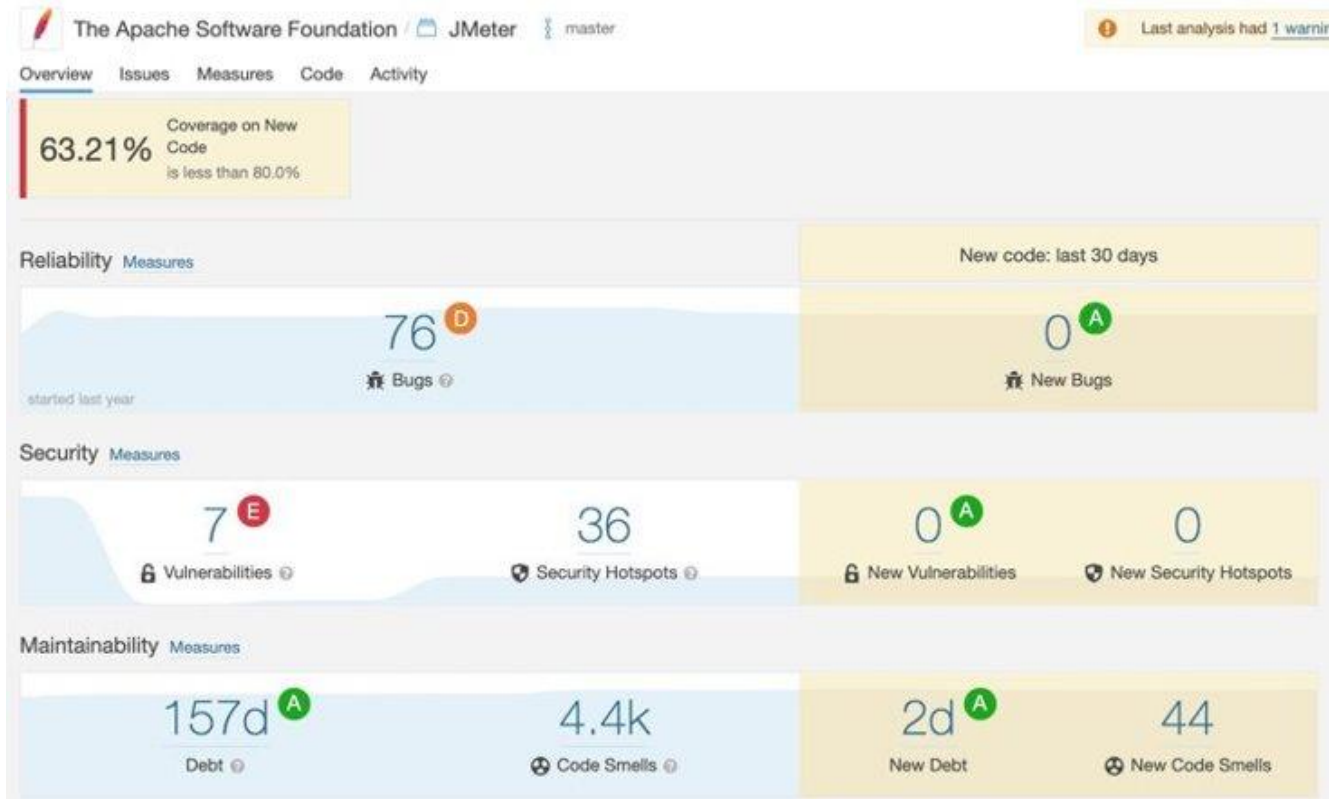


Vulnerabilidades

Issues relacionadas a segurança, exemplos: SQL Injection, erros não tratados corretamente.



Tela principal do projeto



Visualização de issues

```
private transient volatile ReentrantLock currentLock;
```

Use a thread-safe type; adding "volatile" is not enough to make this field thread-safe. [Why is this an issue?](#)

6 years ago ▾ L71 🔗

🐛 Bug 🟢 Minor 🔵 Open Not assigned 20min effort

🔑 cert, multi-threading

```
TransformerFactory factory = TransformerFactory.newInstance(
```

Disable access to external entities in XML parsing. [Why is this an issue?](#)

2 years ago ▾ L70 🔗

🔒 Vulnerability 🚫 Blocker 🔵 Open Not assigned 15min effort

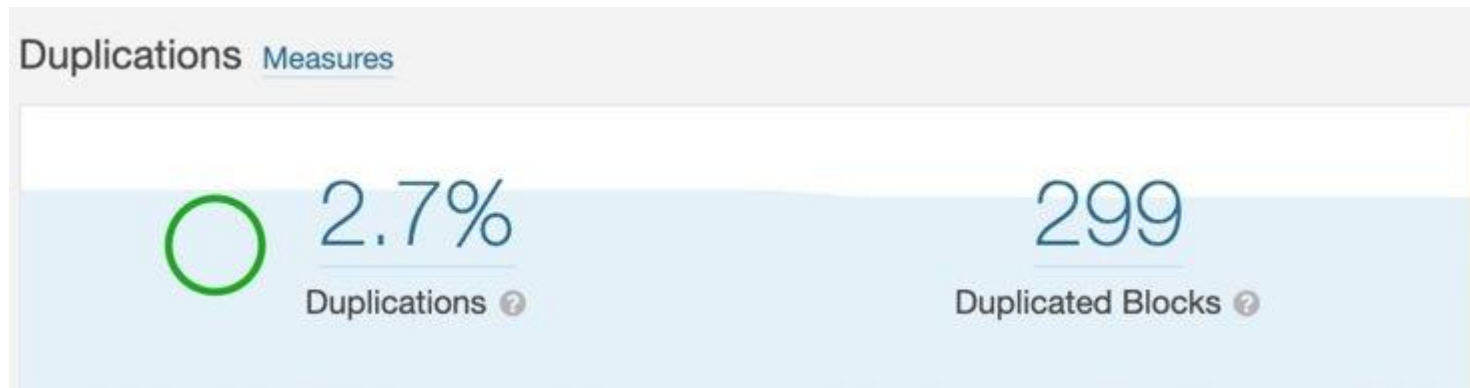
🔑 cwe, owasp-a4

Duplicação de código

O Sonar mostra quais linhas do código estão duplicadas e ajuda o time a evitar este tipo de situação.



Duplicação de código



Cobertura de código

O Sonar auxilia o time a monitorar a cobertura de testes de um projeto.



Atingir
uma
métrica



Encontrar
código
não testado

Cobertura de código

“Cobertura de código é uma ferramenta poderosa para encontrar códigos não testados, mas de pouca importância para indicar quão bom os testes são.”

Martin Fowler
[@martinfowler](https://twitter.com/martinfowler)



Cobertura de código

“Eu espero uma alta cobertura de código. Algumas vezes gerentes obrigam uma. Há uma diferença”

Brian Marick
[@marick](#)



Cobertura de código

Coverage [Measures](#)



55.8%

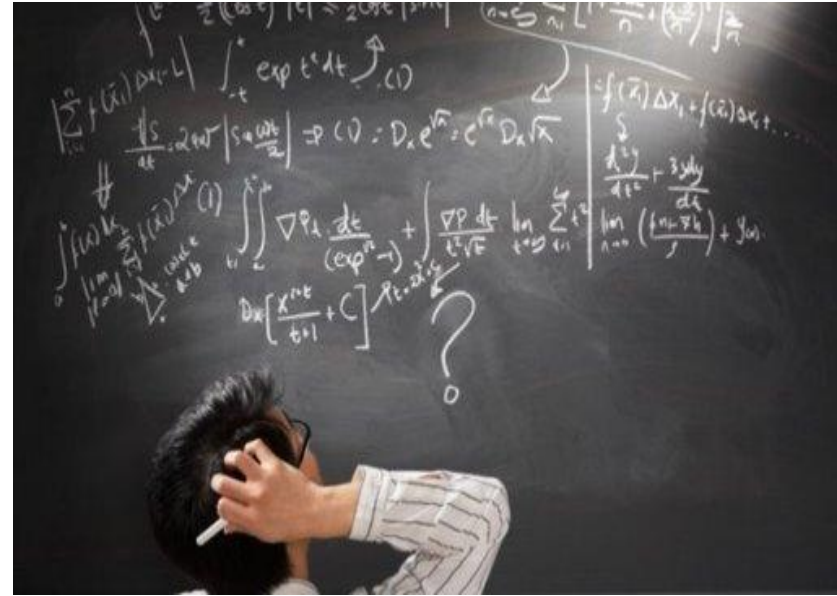
Coverage ?

3k

Unit Tests ?

Complexidade de código

O Sonar auxiliar a encontrar as classes e métodos mais complexos da aplicação.



Complexidade Ciclomática

Métrica criada em 1976 por Thomas J. McCabe.

Indica o quão difícil é **testar** o componente.

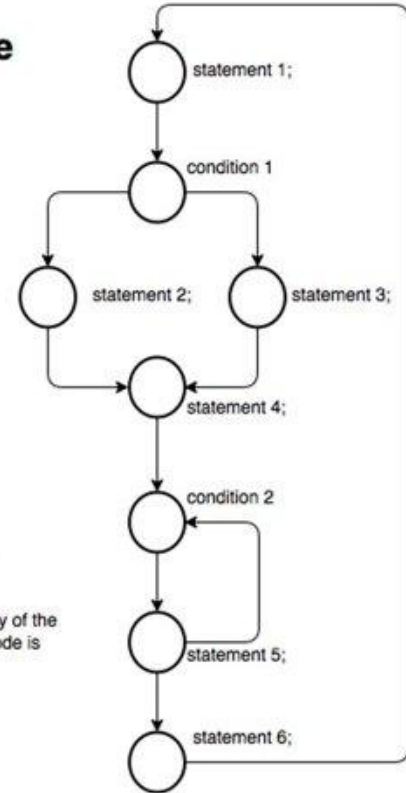
Sample code

```
statement 1;  
If ( condition 1 ) {  
    statement 2;  
} else {  
    statement 3;  
}  
statement 4;  
for( condition 2 ) {  
    statement 5;  
}  
statement 6;
```

Complexity

The cyclomatic complexity of the graph representing the code is

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 9 - 8 + 2 \\ &= 3 \end{aligned}$$



Complexidade Cognitiva

Métrica criada em 2016 por G. Ann Campbell, da empresa SonarSource. Indica o quão difícil é **entender** o componente.

```
int sumOfPrimes(int max) {  
    int total = 0;  
    OUT: for (int i = 1; i <= max; ++i) { // +1  
        for (int j = 2; j < i; ++j) {      // +2  
            if (i % j == 0) {              // +3  
                continue OUT;             // +1  
            }  
        }  
        total += i;  
    }  
    return total;  
}  
// Cognitive Complexity 7
```

```
String getWords(int number) {  
    switch (number) { // +1  
        case 1:  
            return "one";  
        case 2:  
            return "a couple";  
        case 3:  
            return "a few";  
        default:  
            return "lots";  
    }  
} // Cognitive Complexity 1
```

JMeter

View as

List



to select files



to navigate

1,143 files

Cyclomatic Complexity 20,494



New code: last 30 days

 src/protocol/http/src/main/java/org/apache/jmeter/protocol/http/sampler/HTTPSamplerBase.java	322
 src/protocol/http/src/main/java/org/apache/jmeter/protocol/http/sampler/HTTPHC4Impl.java	263
 src/protocol/http/src/main/java/org/apache/jmeter/protocol/http/proxy/ProxyControl.java	259
 src/core/src/main/java/org/apache/jmeter/samplers/SampleResult.java	187
 src/core/src/main/java/org/apache/jmeter/threads/JMeterThread.java	179
 src/core/src/main/java/org/apache/jmeter/JMeter.java	175
 src/core/src/main/java/org/apache/jmeter/samplers/SampleSaveConfiguration.java	175
 src/core/src/main/java/org/apache/jmeter/save/CSVSaveService.java	145
 src/protocol/idbc/src/main/java/org/apache/jmeter/protocol/idbc/AbstractIDBCTestElement.java	141

is excessively complex, with a cyclomatic complexity of 112

4 years ago ▾ L49 🔗 ⌵ ▾

⊕ Code Smell ⓘ Info ○ Open Not assigned

🔗 style

```
private AssertionResult evaluateResponse(SampleResult response) {
```

Refactor this method to reduce its Cognitive Complexity from 38 to the 15 allowed. [Why is this an issue?](#) 3 years ago ▾ L291 🔗

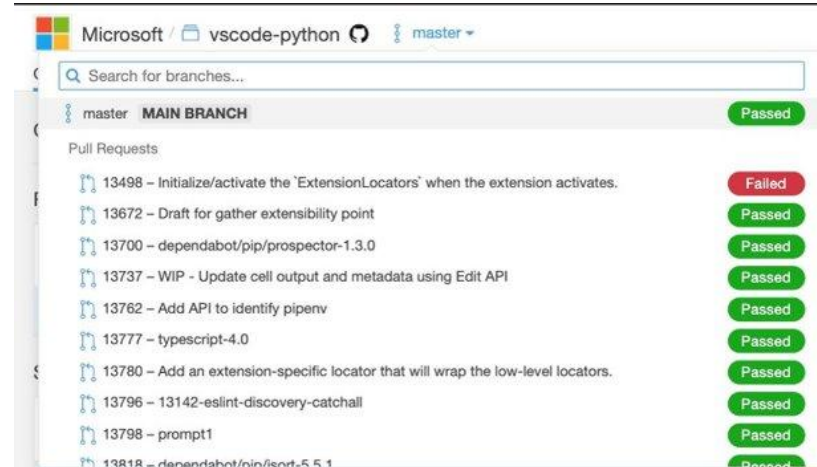
⊕ Code Smell 🔴 Critical ○ Open Not assigned 28min effort

🔗 brain-overload

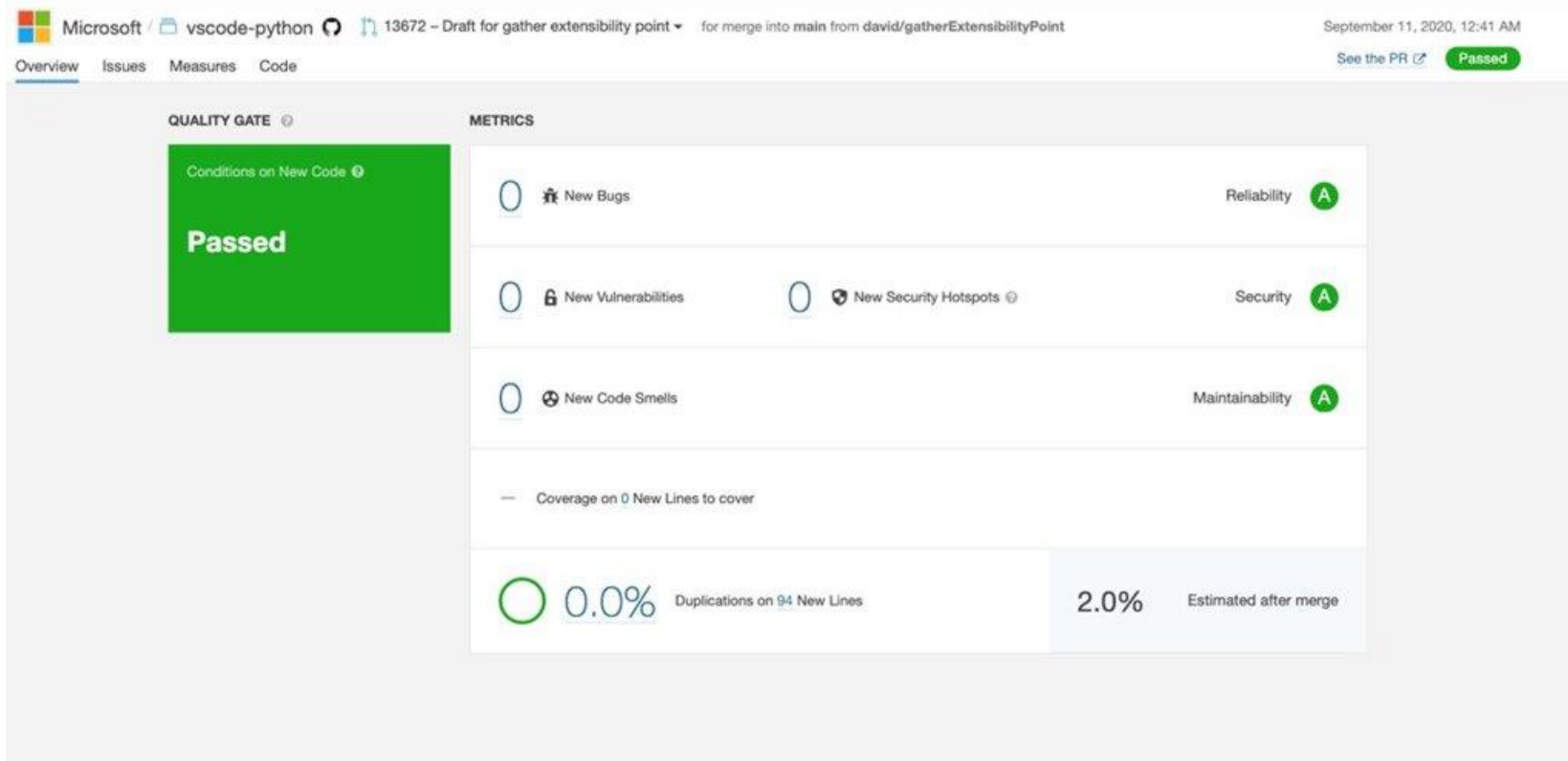
Análise por branch

É possível analisar o código-fonte separadamente por branches.

Isso facilita muito o processo de code review.



Análise por branch



Análise no Pull Request



sonarcloud bot commented yesterday



Kudos, SonarCloud Quality Gate passed!

Passed



A 0 Bugs



A 0 Vulnerabilities



A 0 Security Hotspots



A 0 Code Smells



100.0% Coverage



0.0% Duplication

Badges

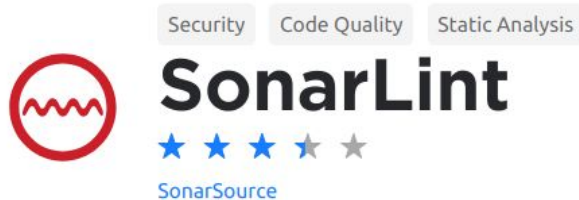
É possível adicionar badges do Sonar no projeto para se obter uma visão rápida da qualidade interna de um projeto.

README.md

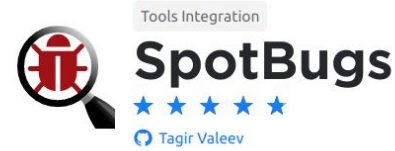


IDEs

Plugins para IDEs são muito úteis para visualização de problemas no código-fonte.



<https://plugins.jetbrains.com/plugin/7973-sonarlint>



<https://plugins.jetbrains.com/plugin/14014-spotbugs>

Conclusão

Como manter a qualidade do código sob pressão?

- A qualidade interna do projeto deve ser monitorada;
- A cada entrega, a qualidade interna do projeto deve melhorar;

Conclusão

Como manter a qualidade do código sob pressão?

- O PO e o time de desenvolvimento devem negociar prazos saudáveis que dêem ao time a possibilidade de cuidar da qualidade interna;

Conclusão

Como manter a qualidade do código sob pressão?

- Em casos urgentes (demandas legais inesperadas, por exemplo), a qualidade interna pode piorar, mas o time deve pagar o débito técnico assim que possível.

Dúvidas ou comentários?