PÓS-GRADUAÇÃO ALFA

FACULDADES ALFA ALVES ARIA

A MELHOR ESCOLA DE NEGÓCIOS DO CENTRO-OESTE

Diego Guedes

Refatoração e Evolução de Software

GRUPO JOSÉ ALVES

Apresentação

- Bacharel em Ciências da Computação, UFG (2010)
- Mestrado em Ciências da Computação, UFG (2013)
- Campeão das regionais da Maratona de Programação em 2008 e 2009
- Trabalhei em projetos regionais, nacionais e mundiais de pesquisa na área de Ciências da Computação
- Fui professor Adjunto na Faculdade Senac por 2 anos (2015-2016)
- P&D goGeo (2014)
- Atualmente, Professor Substituto da UFG e Analista de Sistemas na Saneago

Visualização de Software

- Há muito informação hoje disponível, circulando e com um fluxo de crescimento exponencial.
- Grande parte dessa informação é útil para o ensino e a instrução.
- Dados são recursos valiosos, porém é preciso compreendê-los
- O estudo da visualização pode ser dividido em duas grandes vertentes
  - **Visualização Científica:** dados físicos e inerentemente geométricos;
  - **Visualização de Informação:** informações não físicas, por exemplo coleções de documentos, podem ser beneficiadas, porém não há nenhuma forma óbvia de se mapear tais dados em uma imagem
    - Envolve algo mais complexo.
- Visualização de Software (VS) é uma subárea da Visualização de Informação.
- E os objetivos dela são auxiliar a compreensão de sistemas complexos de software e melhorar a produtividade do processo de desenvolvimento

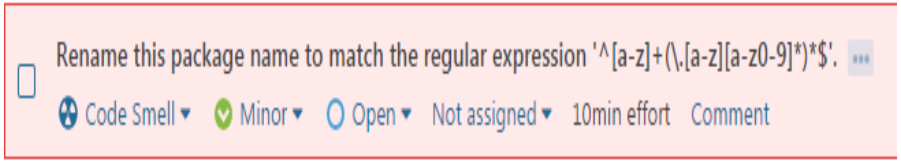
- A visualização de software pode ser levada em consideração pelas três seguintes categorias:
  - **Estrutura**: Relacionada ao que é estático em um software, sem necessidade de executá-lo. Por exemplo: o código-fonte, as estruturas de dados do programa, o grafo de chamadas estático, etc
  - **Comportamento:** O software é executado. Análise do que acontece, em tempo de execução, dada determinada entrada. Por exemplo, Análise de traces de execução, apoio visual à atividade de teste, etc
  - **Evolução**: Categoria aplicada para compreensão das modificações ao longo do tempo. A manutenção e evolução de um sistema pode chegar a 80% do custo total de desenvolvimento.

SonarQube – O que é?

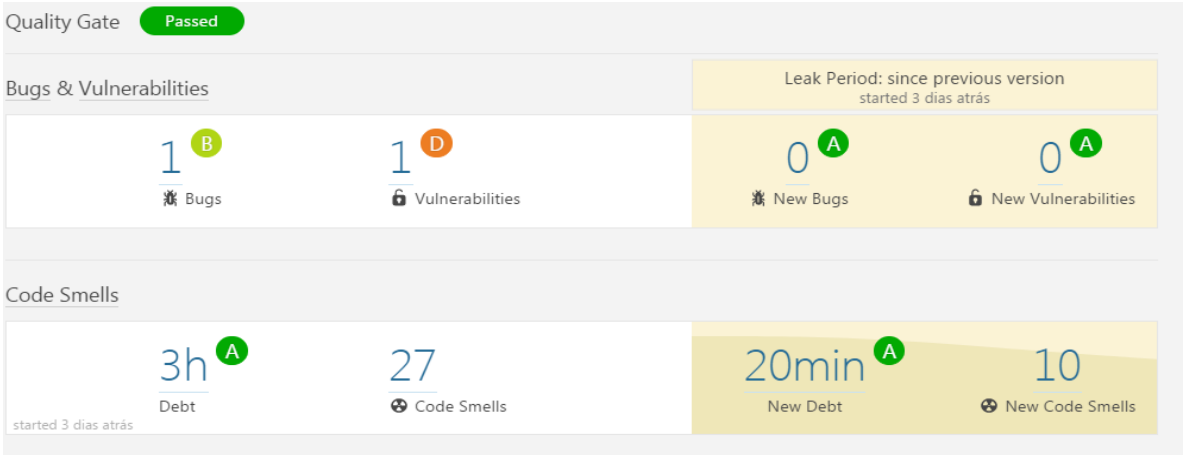
- Sistema para análise de qualidade software
  - Continuous Aspect
- Desenvolvido por Olivier Gaudin, Freddy Mallet and Simon Brandhof
- Software livre (LGPL)
- Anteriormente chamado de “Sonar”

SonarQube – O que é?

- Definição, análise, consulta e ações (PDCA)



- Dashboard



Características

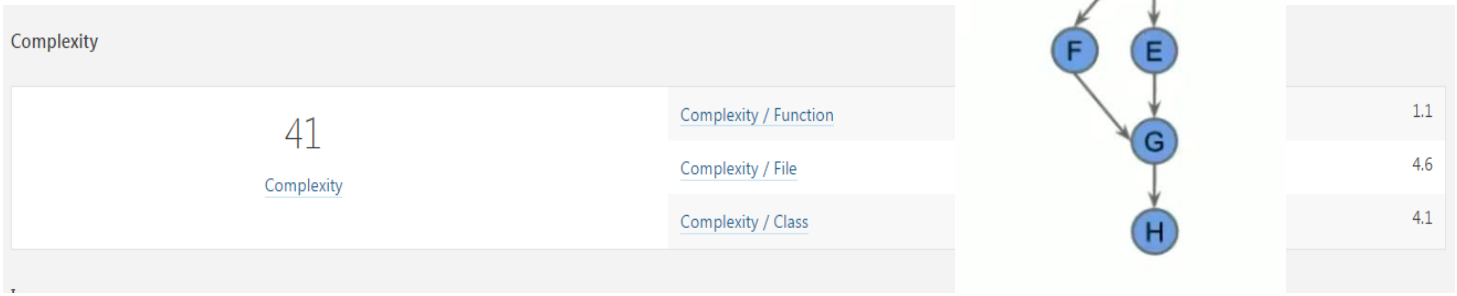
- Os pecados capitais dos desenvolvedores
- Mais de 20 linguagens de programação
  - Nasceu no Java. Hoje, tem plugin até para COBOL
- Todos os envolvidos com o projeto
  - O Dashboard é para todos (desenvolvedor, arquiteto, coordenador de projeto, etc)
- Máquina do tempo
  - Tem timeline
- Extensível
  - Vários plugins interessantes

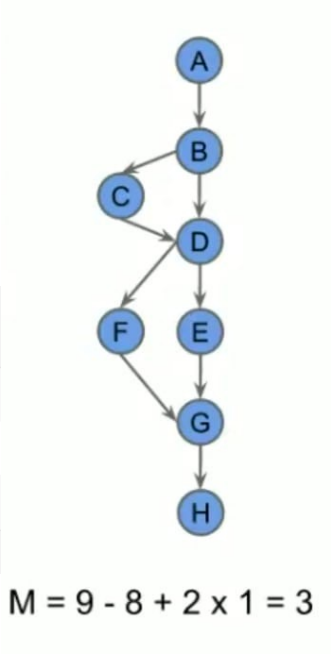
Características

- Visão do código com descrição
- Análise antes do commit
- Automatização
  - Roda dentro de uma build (Jekins por exemplo)
- Integração
  - Integração com o Jira, por exemplo
- Segurança
  - Permissão de acesso, etc

7 Pecados Capitais do desenvolvedor

1 – Alta Complexidade





$$M = A - N + 2*C$$

M: complexidade ciclomática

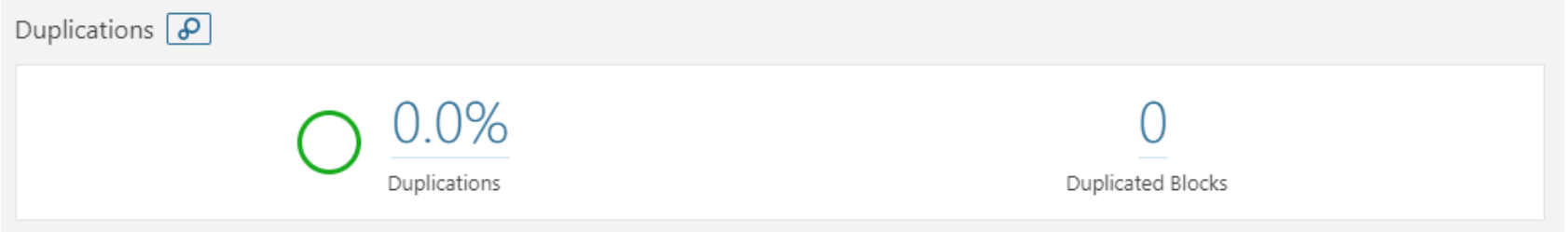
E: quantidade de setas

N: quantidade de nós

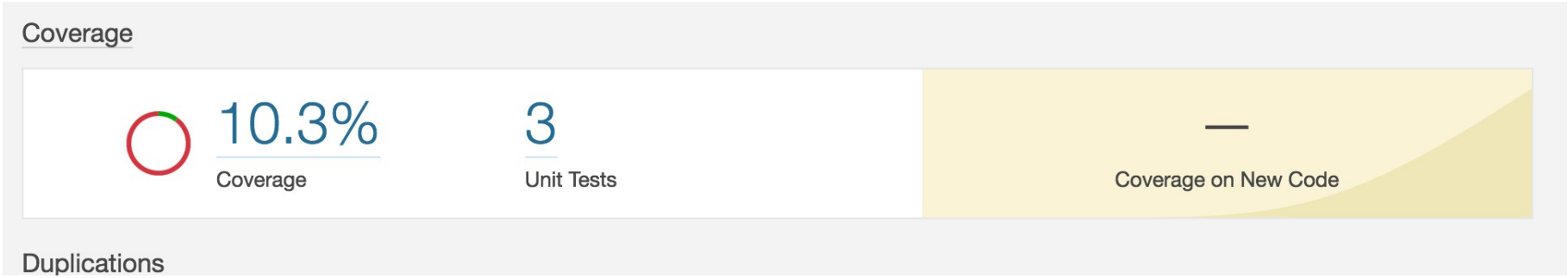
P: quantidade de componentes conectados

1 a 4	Bom
5 a 7	Ok
8 a 10	Considere refatorar
> 10	Refatorar!

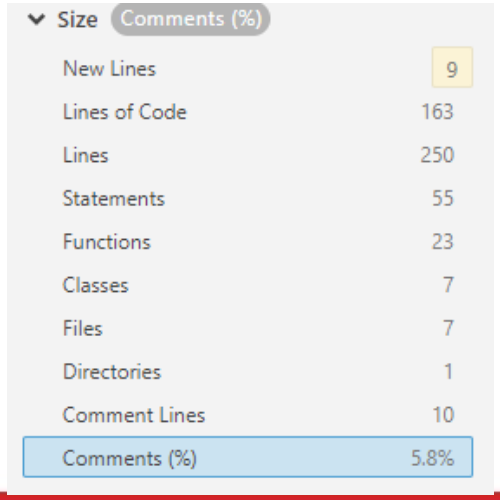
2 – Código Duplicado



3 – Falta de teste unitário



4 – Falta de documentação/comentários



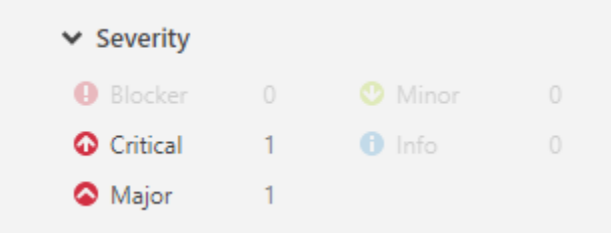
7 Pecados Capitais do desenvolvedor

5 – Design “espaguete”

- dependência cíclica

6 – Falta de padrões de codificação

7 – Erros em potencial

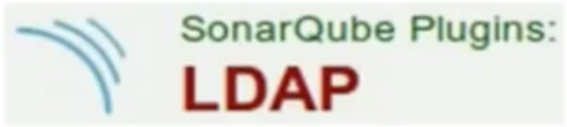


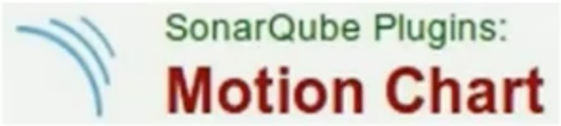
Suporte a mais de 20 linguagens

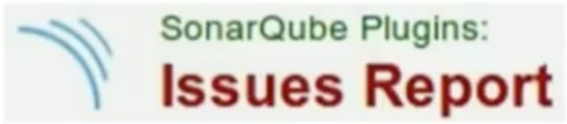
Integração


git
JIRA
SUBVERSION
Jenkins
Apache
JMeter
<APACHE ANT>
maven
eclipse
BAMBOO
mantis
BUG TRACKER

Extensível


SonarQube Plugins:
LDAP


SonarQube Plugins:
Motion Chart


SonarQube Plugins:
Issues Report

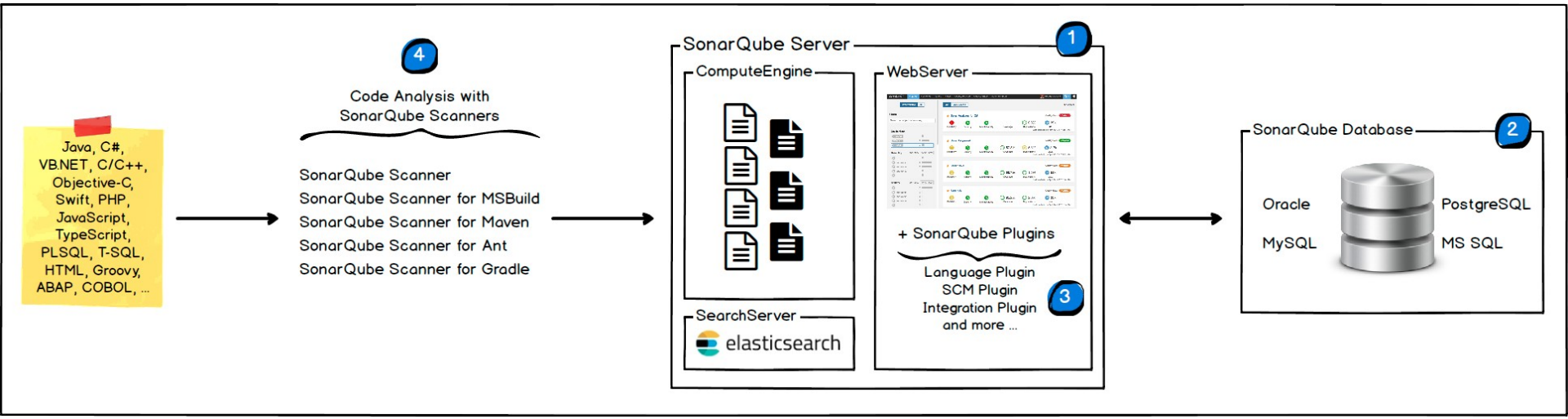

SonarQube Plugins:
Total Quality


SonarQube Plugins:
Useless Code Tracker

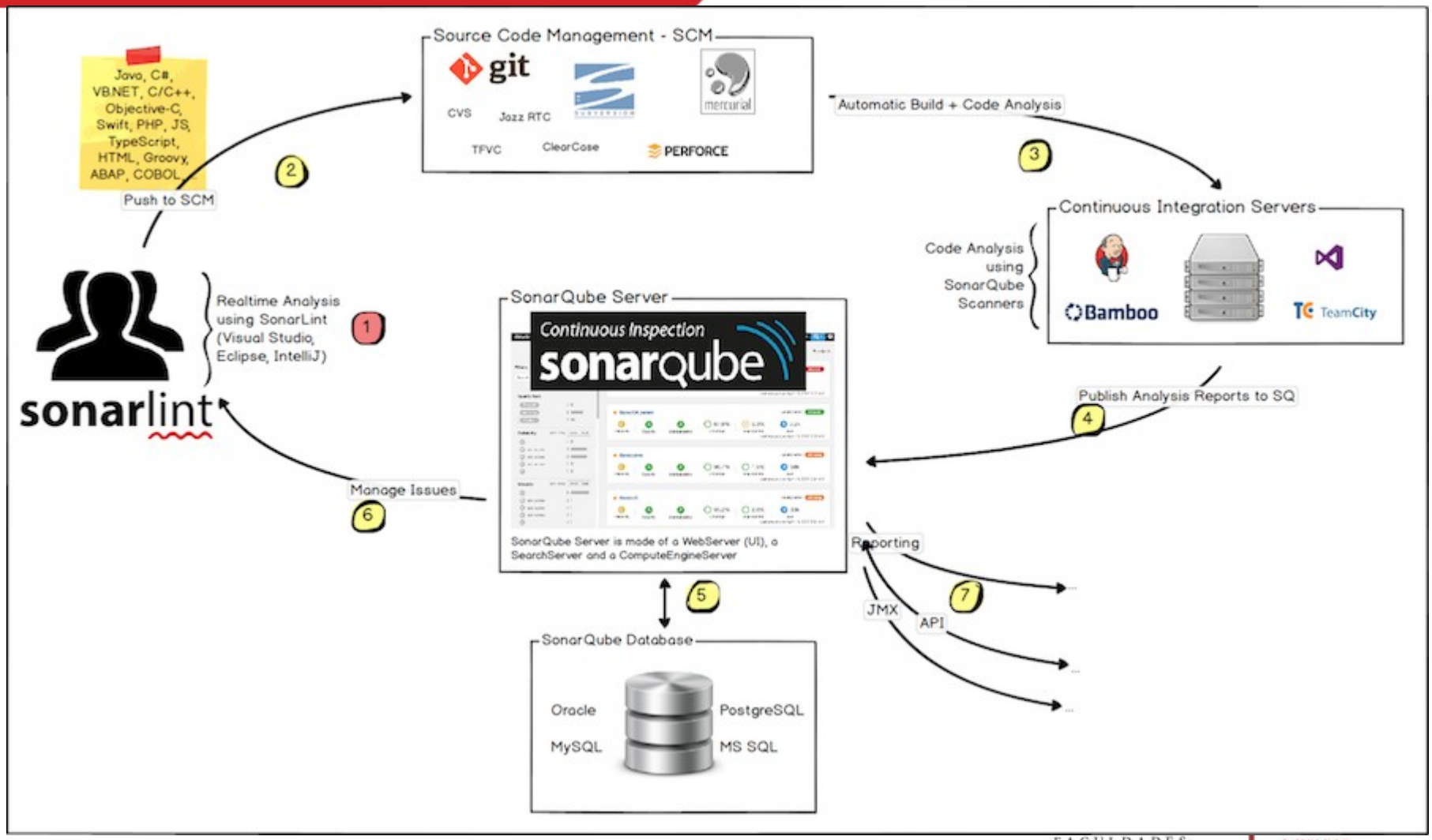

SonarQube Plugins:
PDF Report


SonarQube Plugins:
TimeLine

Arquitetura SonarQube



Integração

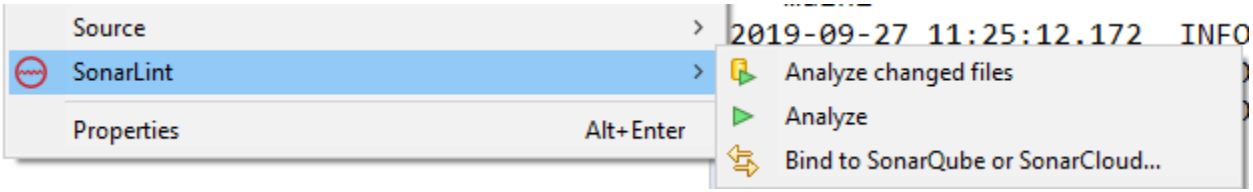

Source Code Management - SCM
git
CVS
Jazz RTC
TFVC
ClearCase
PERFORCE
mercurial
Java, C#, VB.NET, C/C++, Objective-C, Swift, PHP, JS, TypeScript, HTML, Groovy, ABAP, COBOL
Push to SCM
2
Automatic Build + Code Analysis
3
Continuous Integration Servers
Code Analysis using SonarQube Scanners
Bamboo
TeamCity
sonarlint
Realtime Analysis using SonarLint (Visual Studio, Eclipse, IntelliJ)
1
SonarQube Server
Continuous Inspection
sonarqube
SonarQube Server is made of a WebServer (UI), a SearchServer and a ComputeEngineServer
Publish Analysis Reports to SQ
4
Manage Issues
6
Reporting
7
JMX
API
5
SonarQube Database
Oracle
MySQL
PostgreSQL
MS SQL

Instalação

- Eclipse + SonarLint (plugin eclipse)
- Sonarqube (Servidor + Banco de dados)
- Sonarqube Scanner

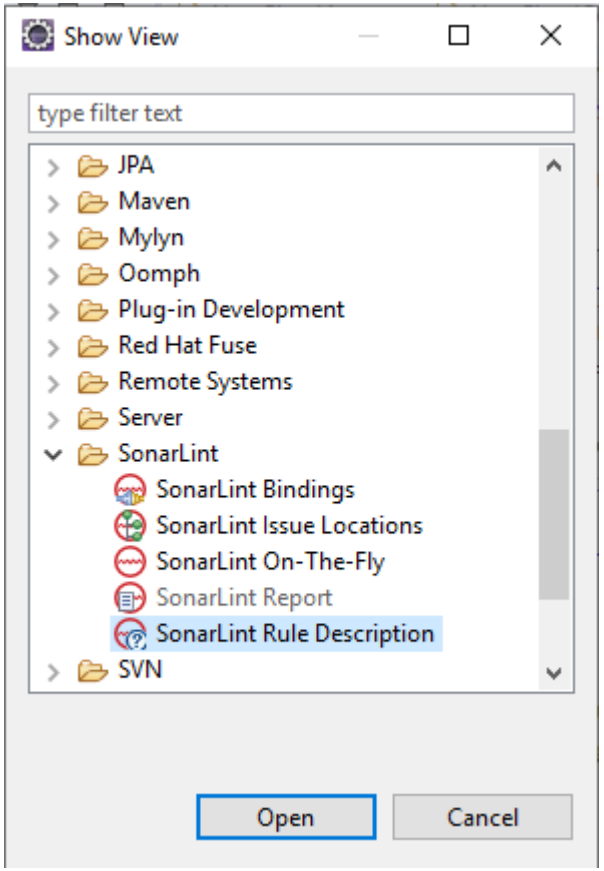
Configurando o ambiente

- Instalar Plugin SonarLint no eclipse Marketplace
- Ao clicar com o botão direito no projeto, vê-se o plugin



Configurando o ambiente

- Em *"Windows→Show View"* vê-se estas perspectivas.
  - Duas visões interessantes são "*Report*" e "*Rule Description*"



Configurando o ambiente

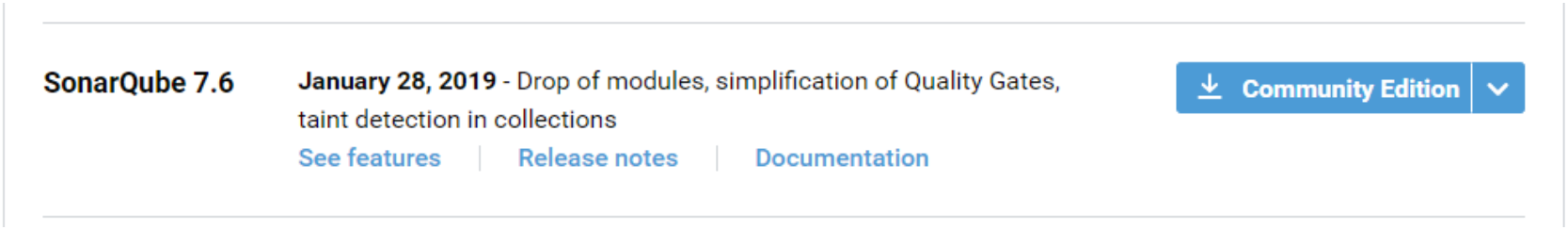
- Window “SonarLint *Report*”

Resource	Date	Description
NotaFiscalController.java		Replace "@RequestMapping(method = RequestMethod.GET)" with "@GetMapping"
NotaFiscalController.java		Replace "@RequestMapping(method = RequestMethod.POST)" with "@PostMapping"
NotaFiscalController.java		Replace "@RequestMapping(method = RequestMethod.POST)" with "@PostMapping"
NotaFiscalController.java		Replace this use of System.out or System.err by a logger.
NotaFiscalController.java		Define a constant instead of duplicating this literal "listanotasfiscais" 4 times.
NotaFiscalController.java		Add a "method" parameter to this "@RequestMapping" annotation.
NotaFiscalController.java		Add a "method" parameter to this "@RequestMapping" annotation.
WebApplication.java	20 ...	Replace this use of System.out or System.err by a logger.
bootstrap.bundle.js		Refactor this code so that this expression does not always evaluate to true.
bootstrap.bundle.js		Remove use of all comma operators in this expression.
bootstrap.bundle.js		Remove use of this comma operator.
bootstrap.bundle.js		This expression might have a value which cannot be called; it might not be a function.
bootstrap.bundle.js		This expression might have a value which cannot be called; it might not be a function.
bootstrap.bundle.js		This expression might have a value which cannot be called; it might not be a function.
bootstrap.bundle.js		This expression might have a value which cannot be called; it might not be a function.

32 files of project notafiscal (at 27/09/2019 14:49)

Configurando o ambiente

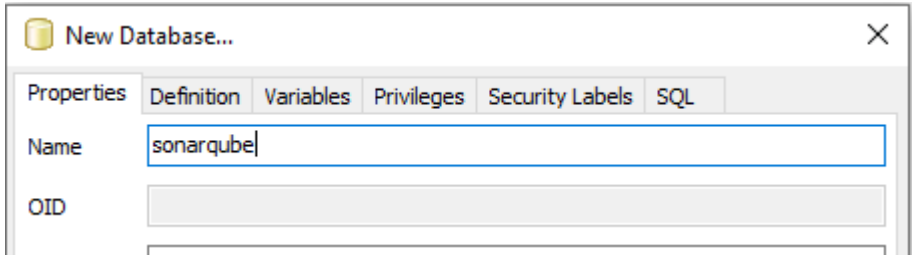
- Download SonarQube 7.6
  - https://www.sonarqube.org/downloads/



- Descompactar o .zip em C:

Configurando o ambiente

- Criar no Postgres banco de dados sonarqube

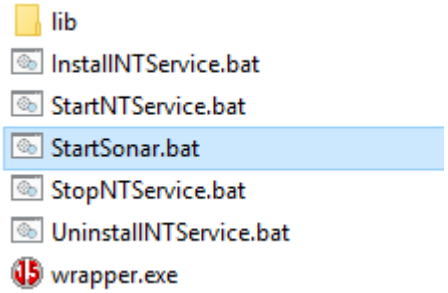


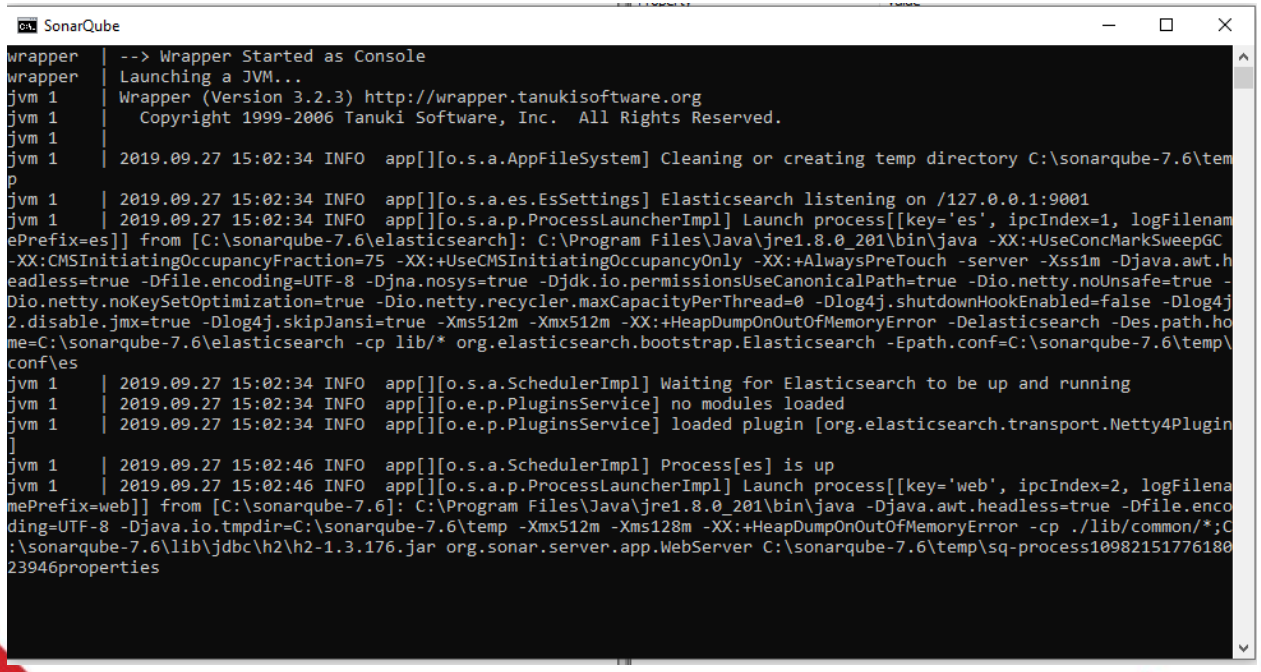
conf/sonar.properties

- Descomentar as linhas (não esqueça de mudar o banco e esquema respectivo)
  - sonar.jdbc.url=jdbc:postgresql://localhost/sonarqube?currentSchema=public
  - sonar.jdbc.username=postgres
  - sonar.jdbc.password=diego

Configurando o ambiente

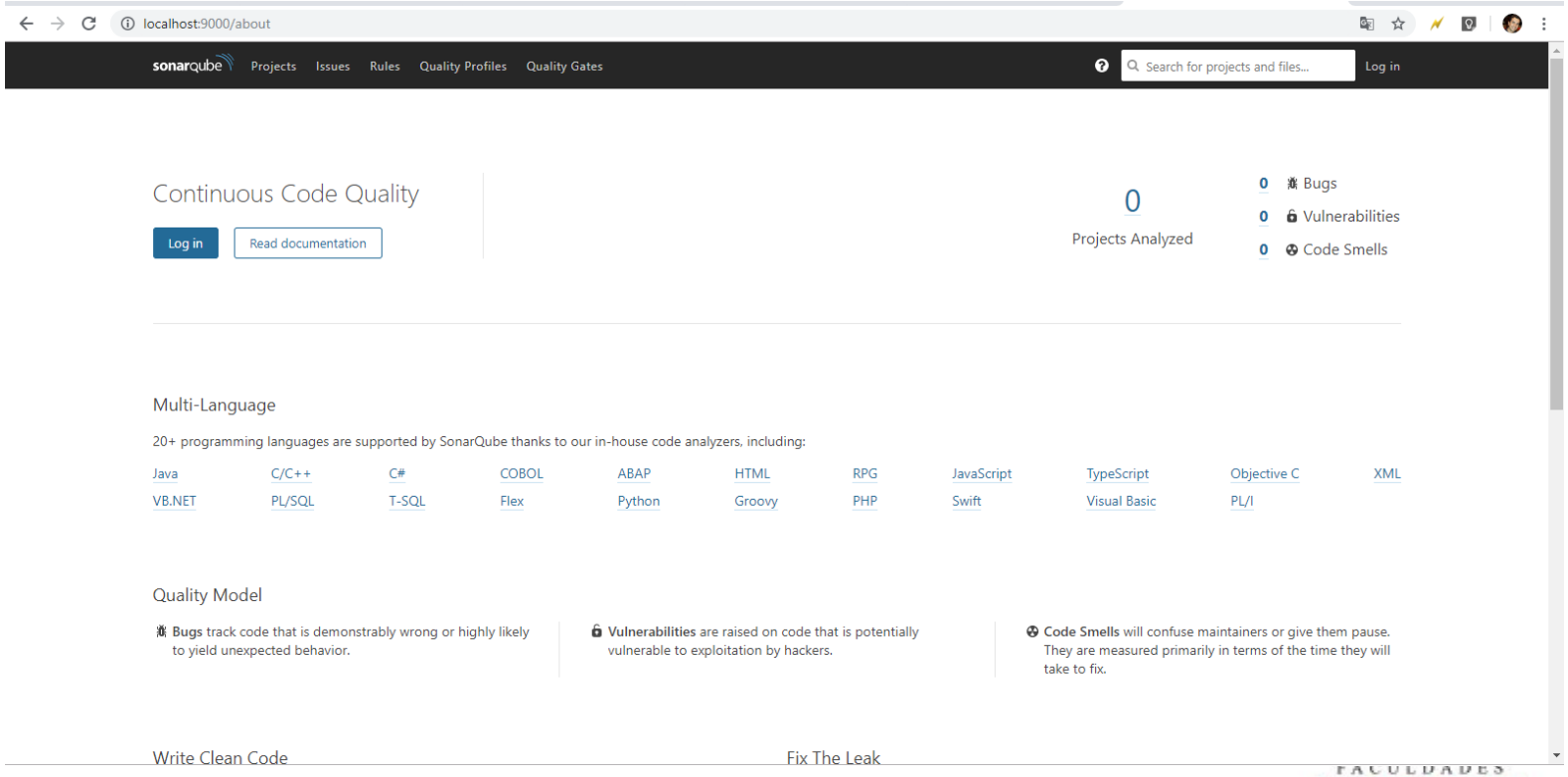
- Executar “sonarqube-7.6\bin\windows-x86-64\StartSonar.bat”





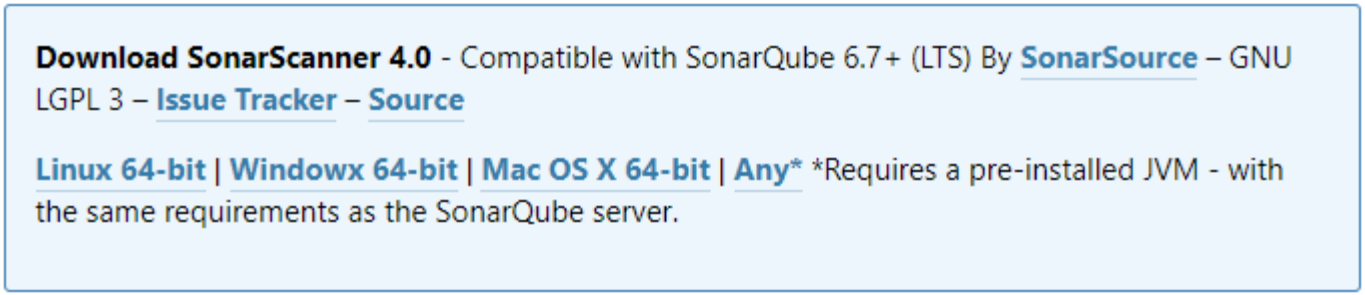
Configurando o ambiente

- Acessar http://localhost:9000
  - **Login**: admin ; **senha**: admin



Configurando o ambiente

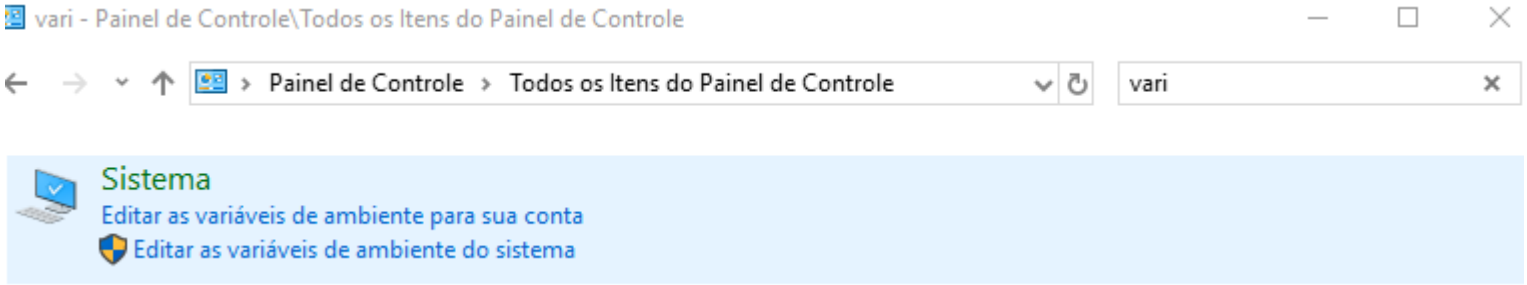
- Baixe o SonarScanner
  - https://docs.sonarqube.org/latest/analysis/scan/sonarscanner/



- Descompactar o zip em C:

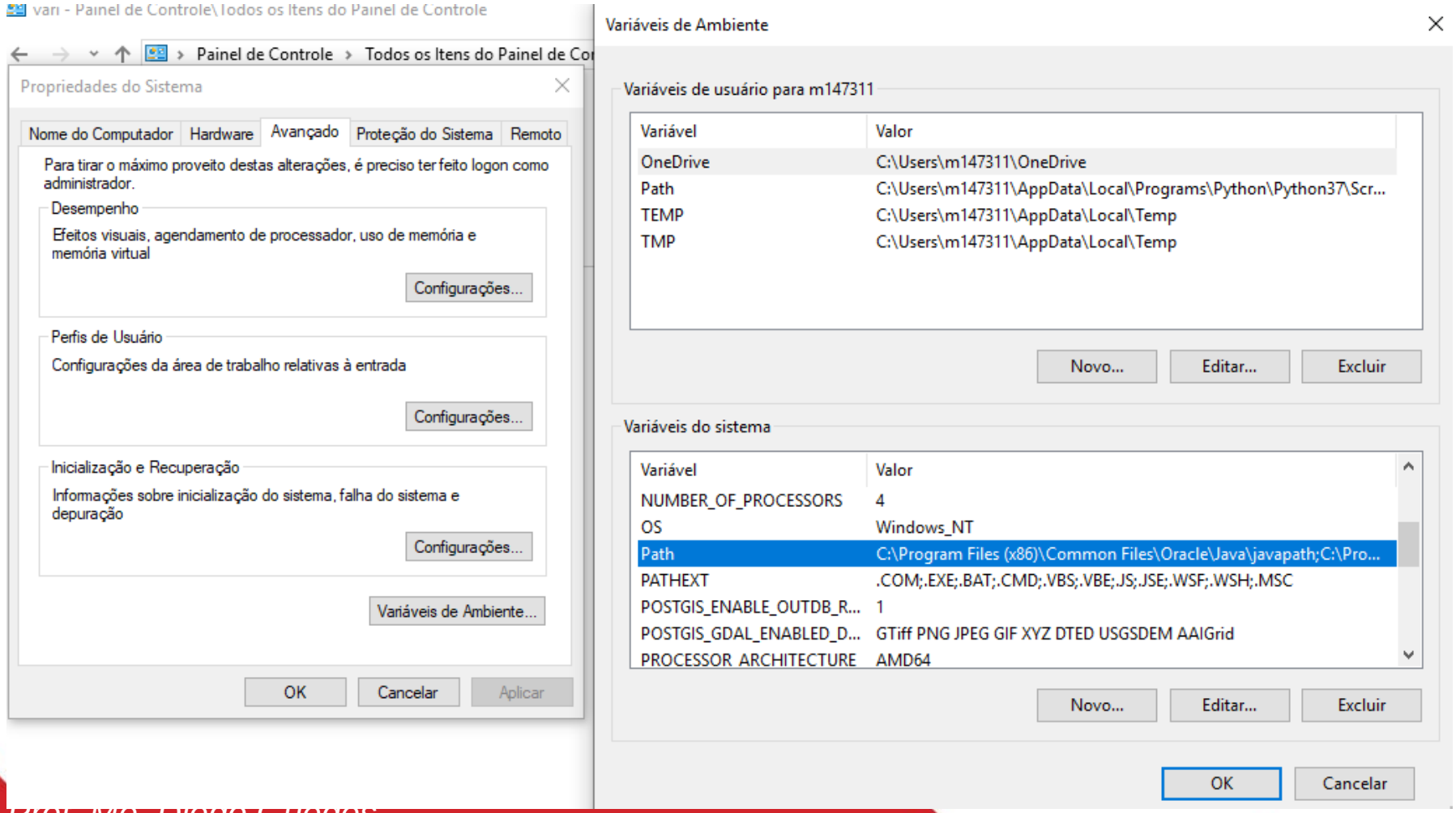
Configurando o ambiente

- Criar um variável de ambiente em
  - Painel de controle



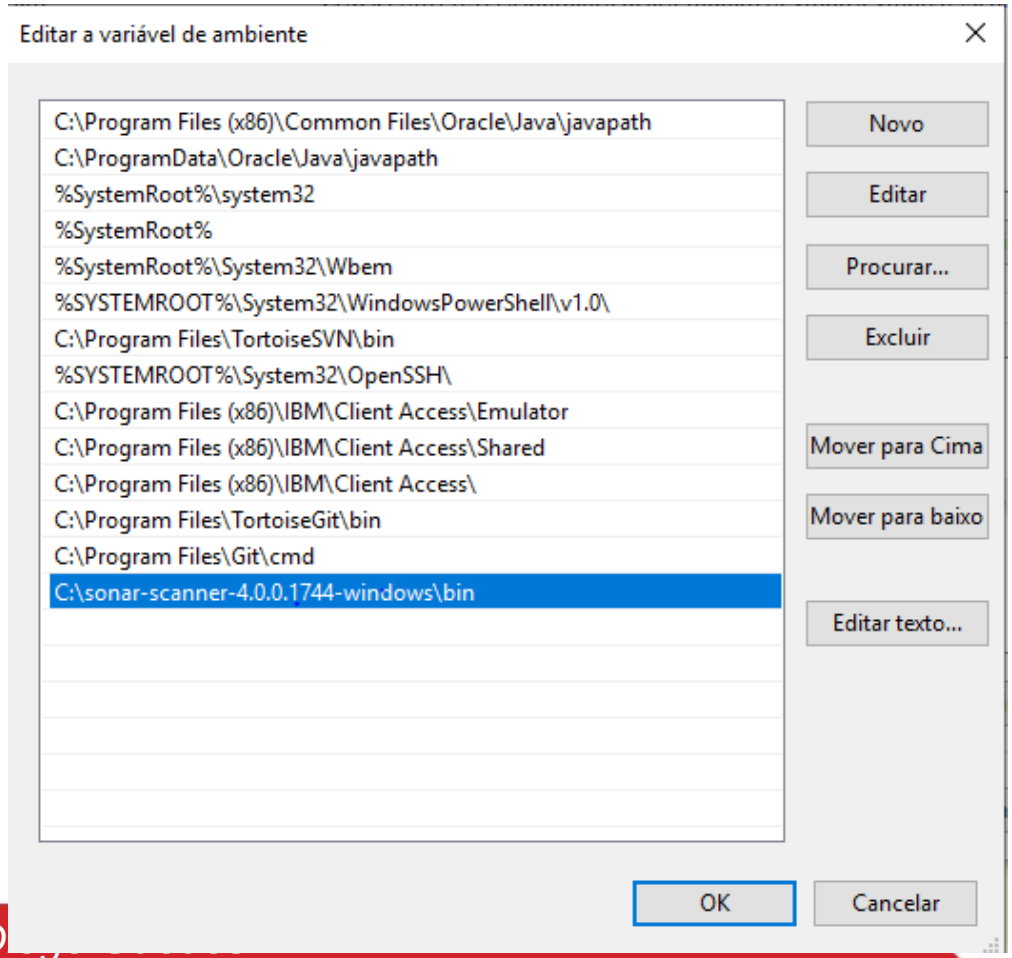
Configurando o ambiente

- Clique em Editar “Path” de “Variáveis do sistema”



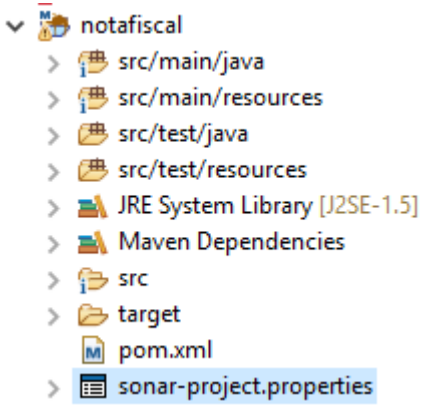
Configurando o ambiente

- Criar Novo
  - C:\sonar-scanner-4.0.0.1744-windows\bin



Configurando o ambiente

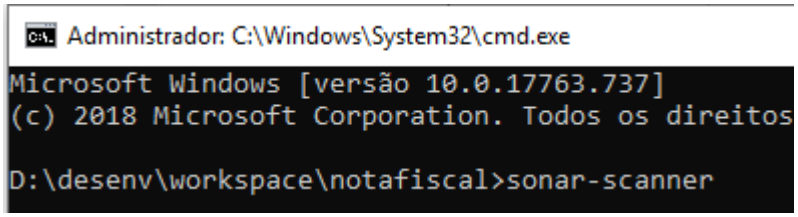
- Colocar o arquivo de propriedades na raiz do projeto
  - https://github.com/diegoguedes/refatoracaoEvolucaoSoftware_unialfa/blob/master/sonar/sonar-project.properties



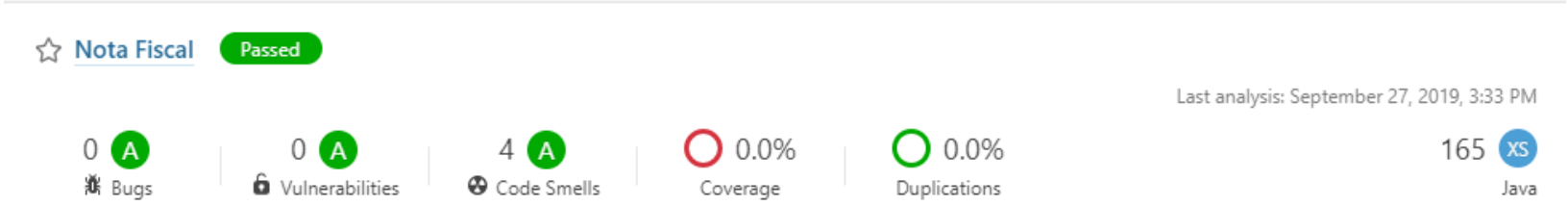

```
# must be unique in a given SonarQube instance
sonar.projectKey=notafiscal

# --- optional properties ---

# defaults to project key
sonar.projectName=Nota Fiscal
# defaults to 'not provided'
sonar.projectVersion=1.0

# Path is relative to the sonar-project.properties
sonar.sources=src/main/java

sonar.java.binaries=target/classes

sonar.language=java
```

Configurando o ambiente

- Executar o sonar-scanner na raiz do projeto



- No sonarqube já aparecerá a primeira análise



Referências

- http://www.sonarlint.org/eclipse/ - Acessado em 27/09/2019
- https://www.sonarqube.org/ - Acessado em 27/09/2019
- Mario Celso Teixeira. Ferramenta SonarQube - Continuous Inspection. 2015

- Obrigado!
- Dúvidas?

Diego Américo Guedes
www.facebook.com/professordiegoguedes
diegoamericoguedes@gmail.com