



INSTITUTO DE CIÊNCIAS EXATAS E INFORMÁTICA - PRAÇA DA LIBERDADE

CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

Matheus Santos Rosa Carneiro

Raíssa Carolina Vilela da Silva

Vitor Augusto Alves de Jesus

**TRABALHO PRÁTICO DE LABORATÓRIO DE EXPERIMENTAÇÃO DE
SOFTWARE**

BELO HORIZONTE

2021

1. INTRODUÇÃO

O GitHub é uma plataforma de armazenamento e versionamento de códigos que permite a criação, gestão e compartilhamento de repositórios gratuitamente a partir de qualquer máquina que tenha conexão à internet. Essa ferramenta oferece uma API gratuita que permite a obtenção de dados sobre qualquer repositório público utilizando queries com parâmetros bem definidos. Dessa forma, utilizaremos essa ferramenta para buscar nossos dados e analisar os resultados encontrados.

A partir do desenvolvimento de sistemas open-source, temos o fator que diversos desenvolvedores contribuem em partes diferentes do código e um dos riscos a serem gerenciados diz respeito à evolução dos seus atributos de qualidade interna. Isto é, ao se adotar uma abordagem colaborativa, corre-se o risco de tornar vulnerável aspectos como modularidade, manutenabilidade, ou legalidade do software produzido. Para tanto, diversas abordagens modernas buscam aperfeiçoar tal processo, através da adoção de práticas relacionadas à revisão de código ou à análise estática através de ferramentas de CI/CD.

Neste contexto, o objetivo deste projeto é analisar aspectos da qualidade de repositórios desenvolvidos na linguagem Java, correlacionando eles com características do seu processo de desenvolvimento, sob a perspectiva de métricas de produto calculadas através da ferramenta CK ([Links para um site externo.](#)).

Este trabalho prático foi proposto pelo professor orientador da disciplina de Laboratório de Experimentação de Software, com o intuito de fazer com que respondêssemos as seguintes perguntas:

1.1.1 Qual é a relação entre a popularidade dos repositórios e as métricas de qualidade?

1.1.2 Qual é a relação entre a maturidade dos repositórios e suas métricas de qualidade?

1.1.3 Qual é a relação entre a atividade dos repositórios e as suas métricas de qualidade?

1.1.4 Qual é a relação entre o tamanho dos repositórios e as suas métricas de qualidade?

As hipóteses levantadas inicialmente são:

1. Quanto mais populares os repositórios, piores são as métricas de qualidade, pois fica mais difícil de se controlar os diferentes aspectos dos sistemas, dessa forma é mais difícil de se manter um padrão de qualidade elevado.
2. Quanto mais novo o repositório, melhores são suas métricas de qualidade, visto que o repositório possui padrões de projeto mais novos e atualizados.
3. Quanto mais atividade no repositório, melhores são suas métricas de qualidade, visto que possuiu mais correções de bugs e melhorias gerais no código.
4. Quanto menor o repositório, melhores são as métricas de qualidade, visto que os componentes do software tendem a trabalhar de forma mais independente e respeitando os aspectos de qualidade.

2. METODOLOGIA

Este é um estudo da disciplina de Laboratório de Medição e Experimentação de Software, que consiste em realizar uma abordagem quantitativa. Esta abordagem, ocorre a partir da migração dos mil repositórios mais bem avaliados utilizando a linguagem de programação Java no *GitHub*. Os dados coletados foram analisados baseando-se em métricas e questões pré-definidas pelo professor orientador da disciplina em questão.

Para finalidade de estudos experimentais através dos dados coletados, foi elaborado um *script* na linguagem de programação *Python* para que fosse possível realizar a coleta e tratativa dos dados. Além disso, foi-se utilizada a ferramenta *CK* para analisar aspectos de qualidade dos repositórios, relacionando então, características do processo de desenvolvimento sob perspectiva das métricas de produto calculadas.

Algumas métricas foram propostas para que facilitasse a análise dos resultados obtidos. A coleta dos dados foi feita através da *API* do *GitHub* conhecida como *GraphQL* e em seguida os dados foram exportados para um arquivo *.csv* para que fosse possível analisar os dados em conjunto. As métricas e as *questions* utilizadas nesse estudo estão descritas a seguir:

1.1 Métricas

- **Popularidade:** número de estrelas;
- **Tamanho:** linhas de código (LOC) e linhas de comentários;
- **Atividade:** número de *releases*;
- **Maturidade:** idade (em anos) de cada repositório coletado;

1.1.1 Métricas de qualidade

- **CBO:** *Coupling between objects* (Acoplamento entre objetos);
- **DIT:** *Depth Inheritance Tree* (Árvore de herança de profundidade);
- **WMC:** *Weight Method Class* (Classe de método de peso);

1.2 Questions

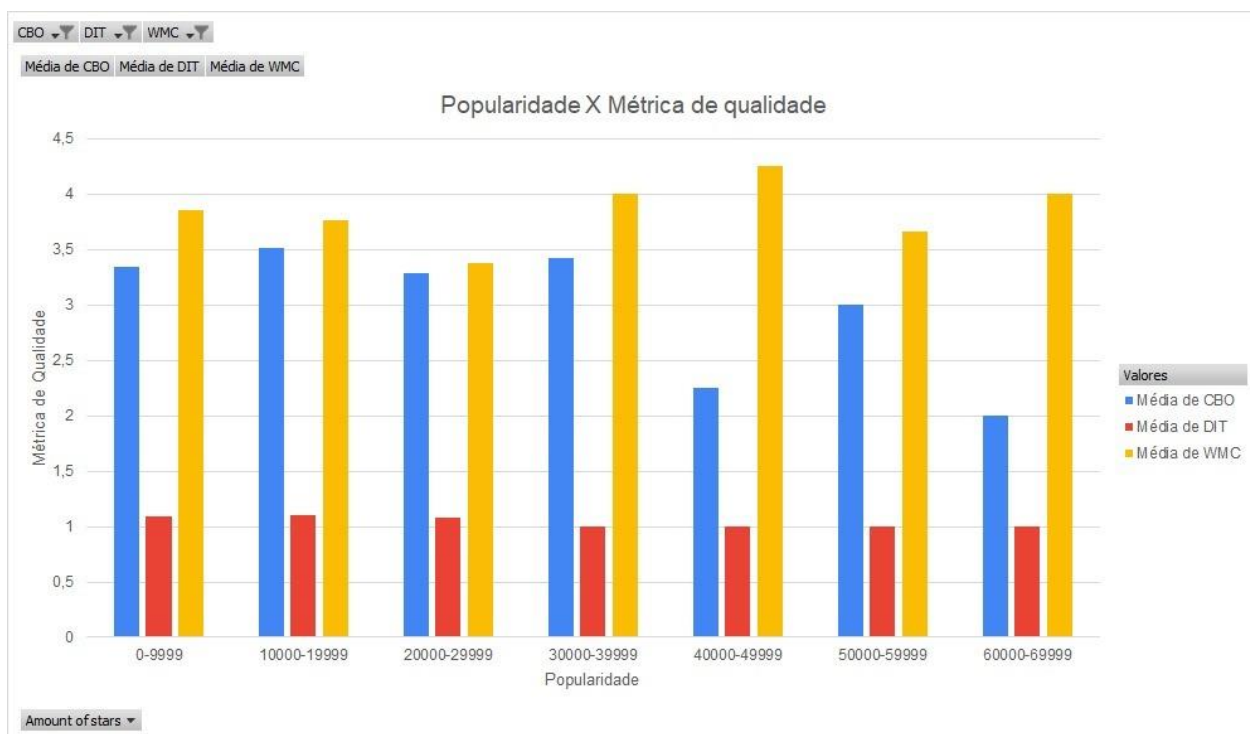
- 1.2.1 Qual é a relação entre a **popularidade** dos repositórios e as métricas de qualidade?
- 1.2.2 Qual é a relação entre a **maturidade** dos repositórios e suas métricas de qualidade?
- 1.2.3 Qual é a relação entre a **atividade** dos repositórios e as suas métricas de qualidade?
- 1.2.4 Qual é a relação entre o **tamanho** dos repositórios e as suas métricas de qualidade?

3. RESULTADOS OBTIDOS

Os resultados obtidos estão diretamente relacionados às métricas citadas e a conclusão da análise está sendo realizada em resposta às *questions* propostas.

➤ Popularidade X Métricas de Qualidade

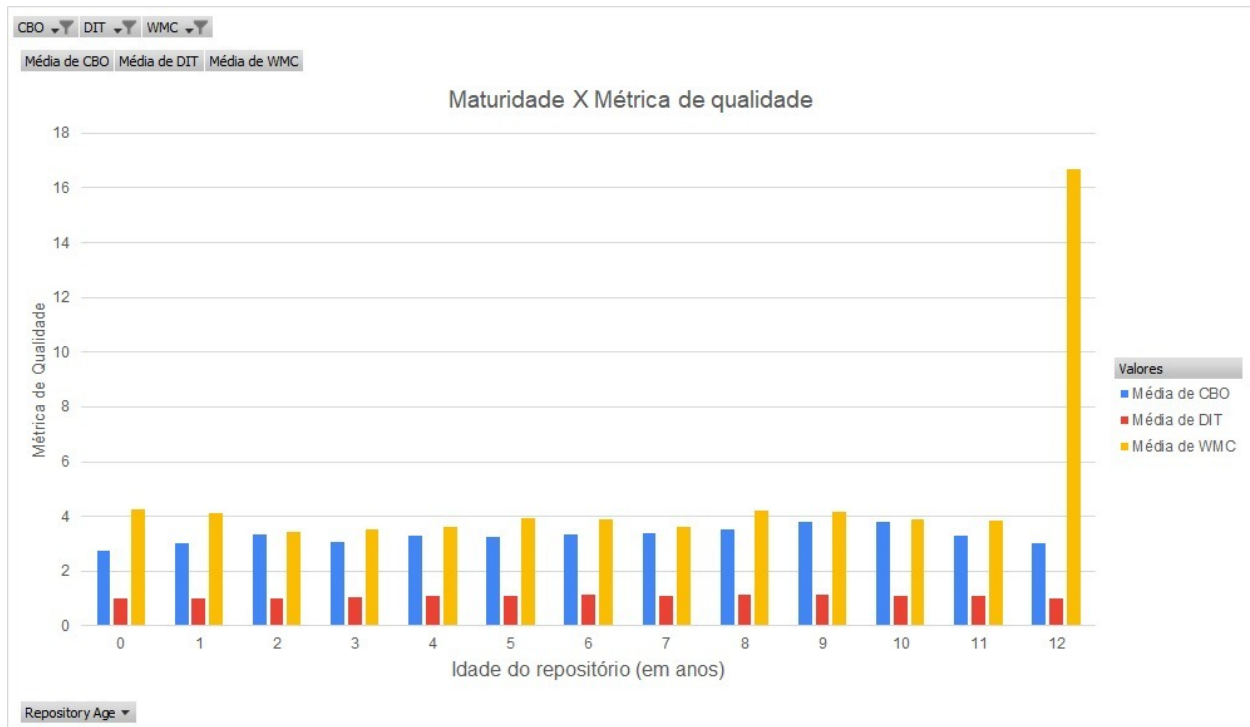
Em análise aos resultados obtidos, podemos perceber que repositórios que tem avaliação entre quarenta mil e cinquenta mil estrelas, ou seja, repositórios de popularidade com nível médio, são os que possuem maior média de **WMC**. No entanto, os repositórios que contém entre dez mil e vinte mil estrelas, são os repositórios que possuem maior média de **CBO**. Pode-se perceber também, que os repositórios entre zero e trinta mil estrelas, possuem maior média de **DIT**. Desta forma, é possível visualizar que em média, não há repositórios que estejam com todas as métricas de qualidade niveladas, ou seja, independente da popularidade do repositório, pode-se ver que alguns vão ter a média de uma métrica de qualidade mais alta ou mais baixa do que as outras métricas.



➤ Maturidade X Métricas de Qualidade

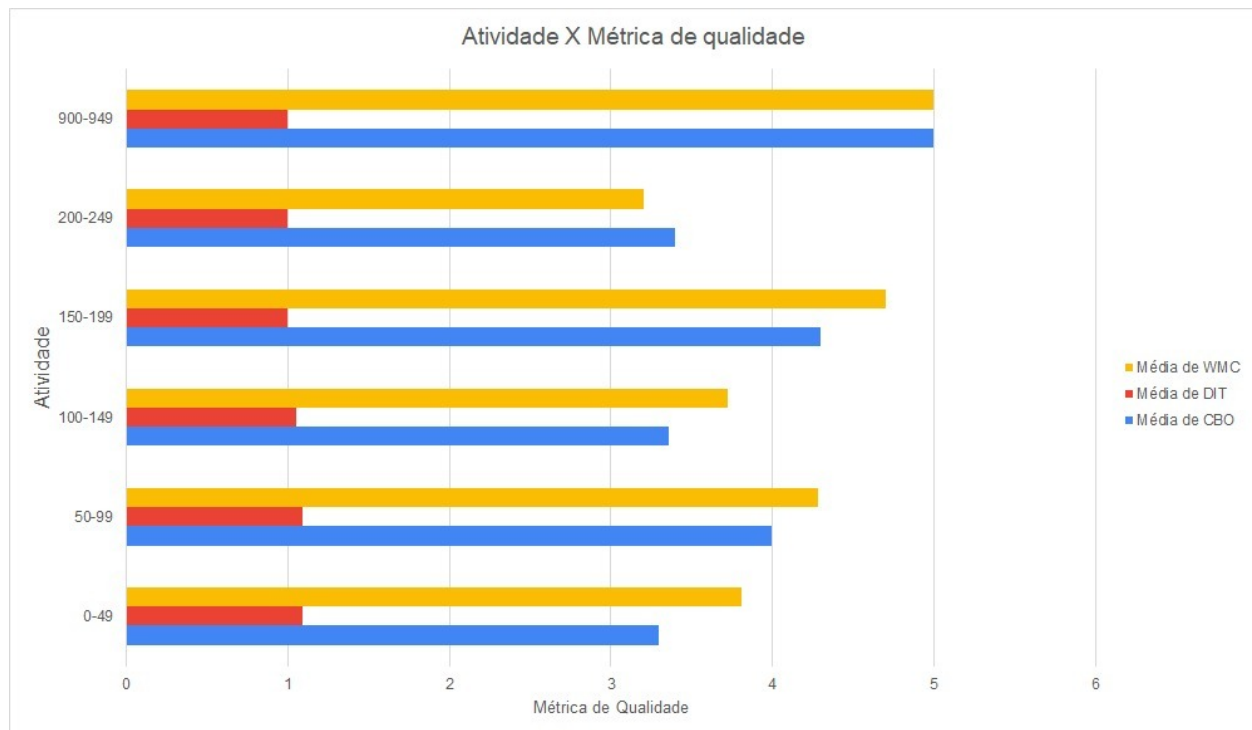
Analisando em relação a maturidade dos repositórios e as métricas de qualidade, há uma discrepância enorme, ao vermos o gráfico, pois, é nítido que os repositórios mais antigos,

que possuem 12 anos de idade, são os repositórios que em média possuem mais classes de métodos de peso (**WMC**). Outra coisa que é possível analisar através da visualização do gráfico, é que os repositórios mais novos, são os repositórios que em média possuem menos acoplamento entre os objetos (**CBO**). No entanto, uma coisa interessante a ser analisada, é que independente da idade do repositório, o nível médio da árvore de herança e profundidade (**DIT**) dos repositórios, não varia de acordo com a idade destes.



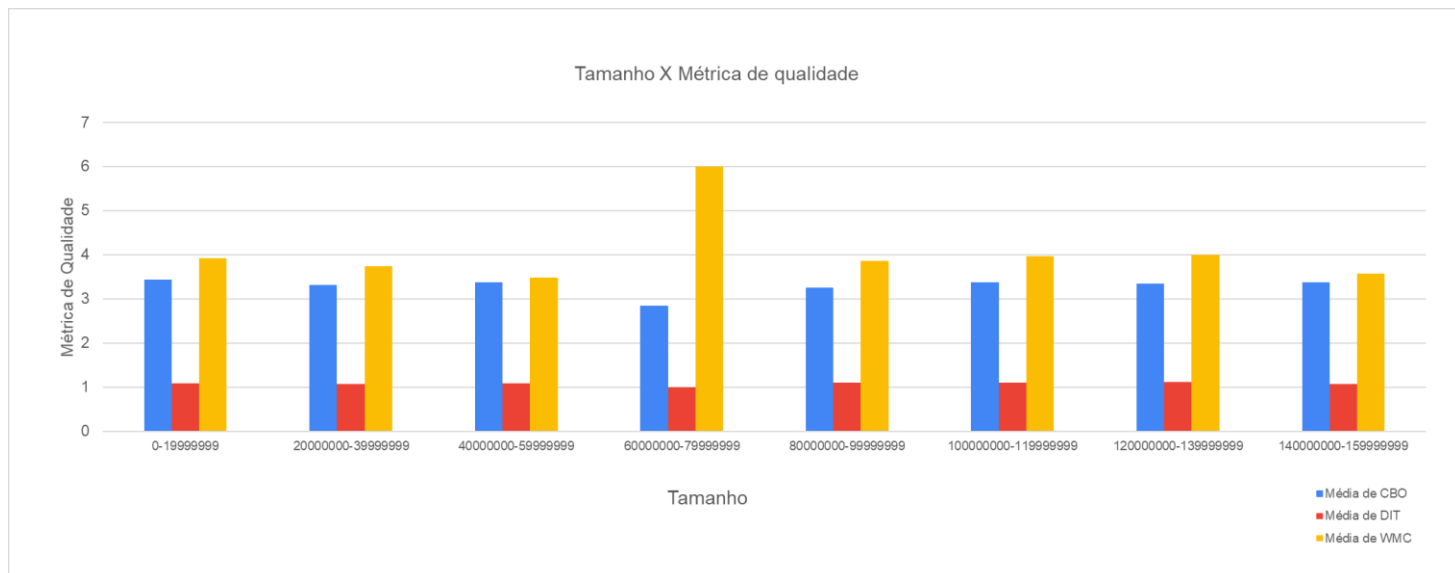
➤ Atividade X Métricas de Qualidade

Relacionando as atividades do repositório com as métricas de qualidade, é possível perceber que os repositórios que mais contém *releases* são os repositórios que tem maior média das métricas **CBO** e **WMC**, no entanto, algo que é interessante a ser observado, é que essa média não cresce de forma proporcional a quantidade de *releases* que possuem nos repositórios. Uma forma de perceber isso, é que os repositórios abaixo de 250 *releases*, tem grandes variações entre as métricas de acoplamento entre objetos e classes de métodos de peso. Além disso, é possível ver que a métrica **DIT** possui variação apenas nas primeiras 150 *releases*, após isso, o **DIT** não sofre mais variações.



➤ Tamanho X Métricas de Qualidade

Ao realizar uma análise do tamanho do repositório em relação às métricas de qualidade, podemos observar que os repositórios que possuem entre sessenta milhões e oitenta milhões de linhas de código, possuem a média do **WMC** mais alta em relação aos outros repositórios, no entanto, possui também a média mais baixa de acoplamento entre objetos (**CBO**). Já em relação ao **DIT**, é possível perceber que ele não varia em relação a quantidade de linhas presente nos repositórios analisados.



4. CONCLUSÃO

A partir da pesquisa realizada pudemos concluir que as medidas de qualidade, relacionadas com outras métricas, podem nos gerar importantes dados e relações obtidas através dos repositórios. A primeira hipótese proposta foi que, quanto mais popular o repositório, menor seriam os números relacionados às métricas de qualidade. Analisando o gráfico pudemos perceber que não há relação direta entre nas métricas de qualidade do WMC e CBO com popularidade. Porém analisando a métrica CBO, (Coupling between objects - Acoplamento entre objetos), quanto mais popular o repositório menor é seu acoplamento com outros objetos, negando a hipótese apresentada.

A segunda hipótese, quanto menos maduro o repositório, melhores são suas métricas de qualidade, se mostrou verdadeira, tendo em vista, há uma discrepância enorme, ao vermos o gráfico, pois, é nítido que os repositórios mais antigos, que possuem 12 anos de idade, são os repositórios que em média possuem mais classes de métodos de peso (WMC). E também é possível analisar através da visualização do gráfico, que os repositórios mais novos, são os repositórios que em média possuem menos acoplamento entre os objetos (CBO) e provavelmente adotaram melhores práticas de projeto.

A terceira hipótese que dizia que quanto mais atividade no repositório, melhores são suas métricas de qualidade, se mostrou falsa, visto que repositórios com menos atividade (mais recentes) possuem melhores práticas de programação e, conseqüentemente melhores indicadores de qualidade.

Por fim, a respeito da última hipótese que diz que, quanto menor o repositório, melhores são as métricas de qualidade, não conseguimos chegar a uma relação direta, visto que as médias de qualidade calculadas a partir dos tamanhos dos repositórios não nos mostrou relação entre esses dois fatores.