

# Relatório Final – Lab02

## Um estudo sobre qualidade em sistemas Java

Pedro Franco  
Gabriel Pongelupe

Professor: Danilo de Quadros  
Curso: Engenharia de Software  
Disciplina: Laboratório de Experimentação de Software  
6º Período

### 1 Informações do Grupo

- Integrantes: Pedro Franco e Gabriel Pongelupe
- Professor: Danilo de Quadros
- Curso: Engenharia de Software
- Disciplina: Laboratório de Experimentação de Software
- Período: 6º

### 2 Introdução

Este trabalho busca compreender como certas características de repositórios de software — como popularidade, idade, nível de atividade e dimensão — se relacionam com métricas de qualidade em sistemas desenvolvidos em Java. Para isso, foram analisados os 1000 projetos Java mais estrelados do GitHub, e calculadas métricas de código com auxílio da ferramenta CK.

### 3 Hipóteses

- H1: Projetos mais famosos apresentam classes mais coesas (menores valores de LCOM).
- H2: Softwares mais antigos acumulam maior acoplamento (CBO superior).
- H3: Repositórios ativos tendem a manter a qualidade de forma mais consistente.
- H4: Sistemas grandes (com mais linhas de código) apresentam árvores de herança mais profundas (DIT elevado).

## 4 Tecnologias e Ferramentas Utilizadas

- Linguagens: Python e Java
- Bibliotecas: os, pandas, subprocess, tempfile, shutil, time, concurrent.futures, date-time, argparse, requests
- Ferramenta de métricas: CK
- API: GitHub REST API

## 5 Metodologia

### 5.1 Coleta de Dados

Os dados foram obtidos através da GitHub REST API, selecionando os repositórios mais populares em Java, organizados por número de estrelas.

### 5.2 Consolidação

As métricas extraídas por classe foram reunidas em arquivos CSV, posteriormente agregados por repositório, formando um conjunto de dados consolidado para análise.

### 5.3 Análise

Aplicou-se estatística descritiva (médias, medianas e desvios padrão), além de cálculo de correlações entre variáveis. Também foram elaborados gráficos exploratórios para examinar as relações entre popularidade, maturidade, atividade e métricas de qualidade.

## 6 Questões de Pesquisa

- RQ1: Qual a relação entre a popularidade dos repositórios e as suas características de qualidade?
- RQ2: Qual a relação entre a maturidade dos repositórios e as suas características de qualidade ?
- RQ3: Qual a relação entre a atividade dos repositórios e as suas características de qualidade?
- RQ4: Qual a relação entre o tamanho dos repositórios e as suas características de qualidade?

## 7 Resultados

### 7.1 Estatísticas Descritivas

As estatísticas sumarizadas encontram-se em `results/complete_analysis.csv`.

## 7.2 Gráficos por Questão de Pesquisa

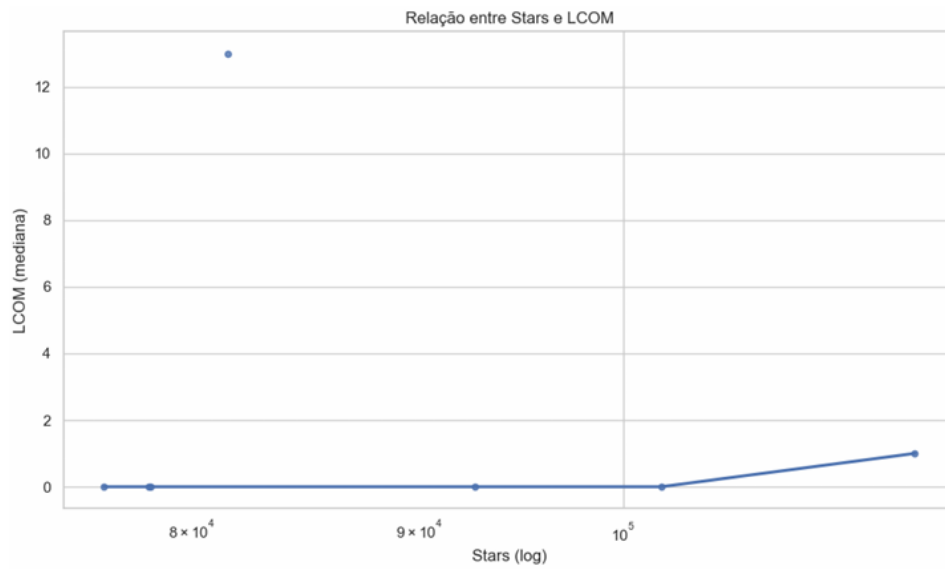


Figura 1: Relação entre popularidade (estrelas) e LCOM(Lack of Cohesion in methods).

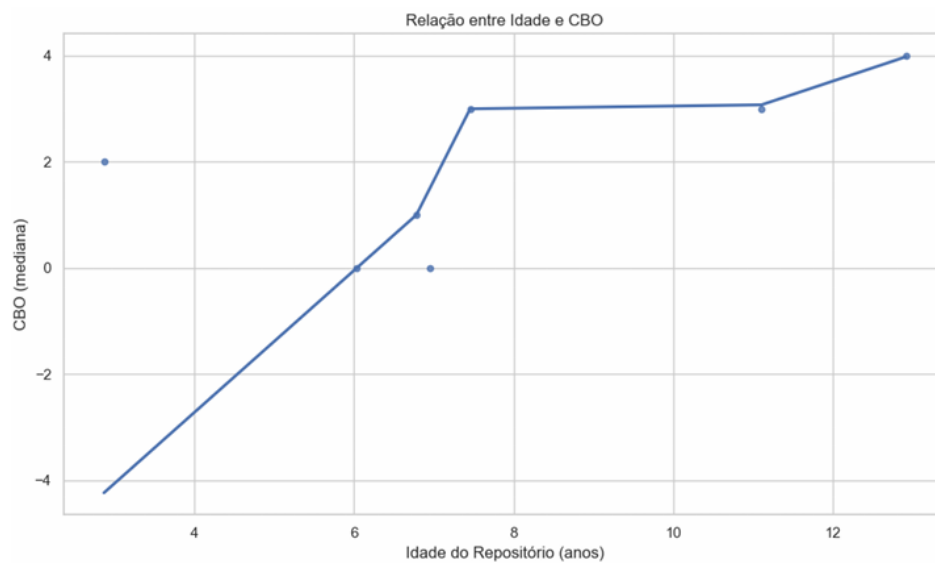


Figura 2: Relação entre maturidade e CBO (Coupling Between Objects).

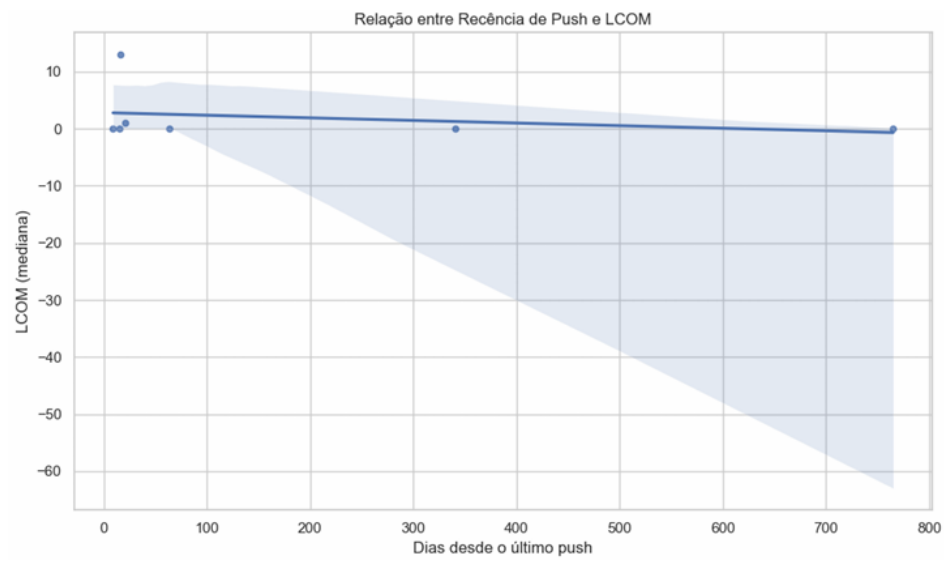


Figura 3: Relação entre recência de push e LCOM(Lack of Cohesion in methods)

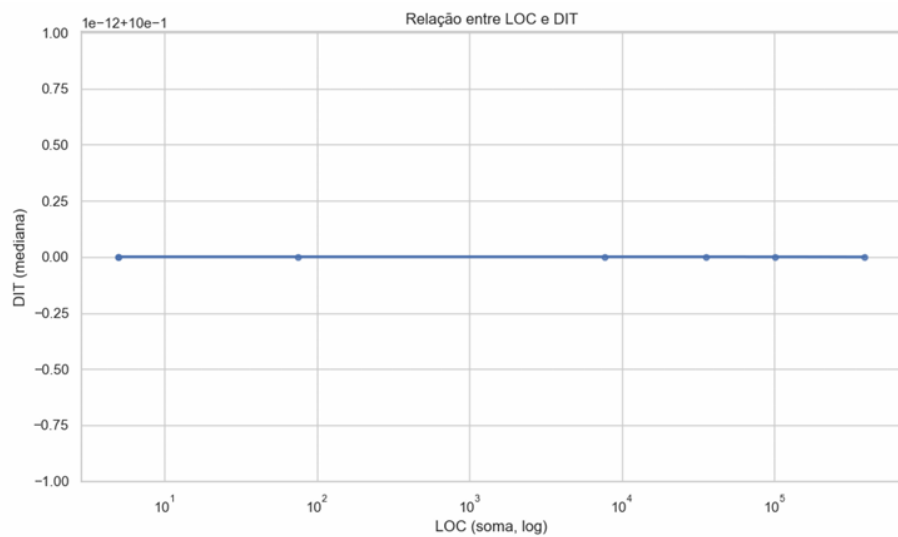


Figura 4: Relação entre tamanho e profundidade da árvore de herança.

## 8 Discussão

As análises sugerem indícios de associação entre algumas características de processo e as métricas de qualidade. De forma geral, as hipóteses foram apenas parcialmente confirmadas:

- H1: houve sinais de relação inversa entre popularidade e LCOM.
- H2: projetos mais antigos apresentaram acoplamento ligeiramente maior.
- H3: a atividade não demonstrou forte impacto sobre as métricas.
- H4: sistemas mais extensos mostraram heranças mais profundas em certos casos.

Entre as limitações, destacam-se falhas da ferramenta CK em projetos muito grandes e a alta variabilidade nos dados.

## 9 Conclusão

O estudo revelou padrões importantes acerca de como popularidade, idade e tamanho podem influenciar a qualidade de software em sistemas Java. Apesar de limitações, foi possível identificar relações significativas, ainda que não absolutas. Pesquisas futuras podem incluir novas métricas, análises temporais e até mesmo a utilização de dashboards interativos.