PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

ENGENHARIA DE SOFTWARE

Gabriella Fernanda Silva Pinto

Matheus Brasil Aguiar

Caracterizando a atividade de code review no github

Projeto apresentado à disciplina de Laboratório de Experimentação de Software do curso de Graduação em Engenharia de Software da Pontifícia Universidade Católica de Minas Gerais, como requisito parcial para avaliação da disciplina.

Orientador: Danilo de Quadros Maia Filho

MINAS GERAIS

2025

SUMÁRIO

1. Introdução

No desenvolvimento colaborativo de software, o processo de *code review* é uma etapa essencial para garantir a qualidade e a confiabilidade do código antes da integração ao projeto principal. No GitHub, esse processo é materializado por meio das *Pull Requests* (PRs), que permitem revisões, comentários e sugestões entre desenvolvedores.

O objetivo deste laboratório é **analisar as atividades de revisão de código em repositórios populares do GitHub**, identificando variáveis que influenciam no *merge* das PRs, considerando fatores como tamanho, tempo de análise, descrição e interações.
A partir dessa análise, busca-se compreender como as características de uma PR impactam sua aceitação, auxiliando na formulação de boas práticas para submissões mais eficazes.

2. Questões de Pesquisa (RQs)

As questões de pesquisa foram organizadas em duas dimensões:

A. Feedback Final das Revisões (Status do PR)

- **RQ01:** Qual a relação entre o tamanho das PRs e o feedback final das revisões?
- **RQ02:** Qual a relação entre o tempo de análise das PRs e o feedback final das revisões?
- **RQ03:** Qual a relação entre a descrição das PRs e o feedback final das revisões?
- **RQ04:** Qual a relação entre as interações nas PRs e o feedback final das revisões?

B. Número de Revisões

- **RQ05:** Qual a relação entre o tamanho das PRs e o número de revisões realizadas?
- **RQ06:** Qual a relação entre o tempo de análise das PRs e o número de revisões realizadas?
- **RQ07:** Qual a relação entre a descrição das PRs e o número de revisões realizadas?
- **RQ08:** Qual a relação entre as interações nas PRs e o número de revisões realizadas?

3. Hipóteses Informais (IH)

IH	Descrição
IH01	PRs menores têm maior chance de serem aprovados (MERGED), pois são mais fáceis de revisar e apresentam menos riscos de integração.
IH02	PRs com descrições mais detalhadas facilitam o entendimento das mudanças e têm maior chance de serem aprovados, mas descrições muito longas podem indicar mudanças complexas e polêmicas, aumentando a chance de rejeição.
IH03	PRs com mais revisões formais tendem a ser aprovados, enquanto PRs com mais comentários informais podem indicar discussões e dúvidas, aumentando a chance de rejeição.
IH04	PRs que permanecem abertos por menos tempo tendem a ser aprovados, enquanto PRs rejeitados (CLOSED) passam por revisões mais longas e discussões mais extensas.

4. Tecnologias e Ferramentas Utilizadas

- **Linguagem:** Python
- **Bibliotecas:** Pandas, Matplotlib, Seaborn
- **APIs:** GitHub GraphQL API
- **Dependências adicionais:** requests, csv, os, datetime, scipy.stats

Essas ferramentas foram utilizadas para coletar, tratar, analisar e visualizar dados das PRs de repositórios do GitHub.

5. Metodologia

5.1. Criação do Dataset

O *dataset* foi construído a partir dos **200 repositórios mais populares do GitHub**, considerando apenas aqueles com **mais de 100 PRs (MERGED + CLOSED)**.
Foram incluídas apenas PRs que:

- possuem status **MERGED** ou **CLOSED**;
- têm **pelo menos uma revisão** registrada;

- tiveram **duração superior a uma hora** entre criação e fechamento (para eliminar revisões automáticas).