Pontifícia Universidade Católica de Minas Gerais
Laboratório de Experimentação de Software
ISSN:

RELATÓRIO

Caracterizando a Atividade de Code Review no GitHub

Characterizing the Code Review Activity on GitHub

Lorrayne Oliveira [Pontifícia Universidade Católica de Minas Gerais | *lorrayne.marayze@gmail.com*]
Pedro Pires [Pontifícia Universidade Católica de Minas Gerais | *pedro.pires@gmail.com*]

Resumo. Este trabalho apresenta um estudo empírico sobre a prática de code review em repositórios populares do GitHub. A atividade de revisão de código, fundamental para assegurar a qualidade e manutenibilidade do software, é analisada sob diferentes dimensões, considerando métricas associadas ao tamanho, tempo de análise, descrição e interações de pull requests (PRs). O objetivo central é compreender como essas variáveis influenciam o resultado das revisões (merge ou rejeição) e o número de revisões realizadas. Para isso, foi construído um dataset contendo PRs dos 200 repositórios mais populares do GitHub, avaliando somente aqueles que passaram por processos de revisão humana e tiveram tempo mínimo de análise superior a uma hora. As correlações entre as variáveis foram examinadas por meio de análises estatísticas de Spearman, de modo a identificar padrões significativos no comportamento das revisões de código. Os resultados obtidos fornecem evidências quantitativas sobre fatores que impactam o sucesso das contribuições, oferecendo insights sobre as dinâmicas colaborativas de revisão em projetos open-source.

Abstract. This work presents an empirical study on the practice of code review in popular GitHub repositories. The code review activity, essential for ensuring software quality and maintainability, is analyzed through multiple dimensions, considering metrics related to the size, review time, description, and interactions of pull requests (PRs). The main goal is to understand how these variables influence the review outcomes (merge or rejection) and the number of reviews performed. A dataset was built containing PRs from the 200 most popular GitHub repositories, considering only those with human reviews and analysis time greater than one hour. Correlations among variables were examined using the Spearman statistical test to identify significant patterns in review behavior. The results provide quantitative evidence about factors influencing contribution success, offering insights into collaborative review dynamics in open-source projects.

Palavras-chave: Code Review; Pull Requests; GitHub; Revisão de Código; Engenharia de Software Colaborativa

Keywords: Code Review; Pull Requests; GitHub; Code Quality; Collaborative Software Engineering

1 Introdução

A revisão de código (code review) é uma das práticas mais consolidadas em processos ágeis de desenvolvimento de software. Ela consiste na inspeção do código por revisores antes de sua integração ao repositório principal, com o intuito de detectar defeitos, melhorar a legibilidade e assegurar a aderência aos padrões de qualidade. No contexto de plataformas open-source como o GitHub, essa atividade é operacionalizada por meio de pull requests (PRs), que representam contribuições submetidas por desenvolvedores e avaliadas por colaboradores do projeto.

Neste trabalho, busca-se caracterizar empiricamente o comportamento das atividades de code review em repositórios populares do GitHub. Especificamente, este estudo visa compreender a forma com que os fatores, como o tamanho do PR, o tempo de análise, a extensão da descrição e o volume de interações influenciam o resultado final da revisão (merge ou rejeição) e o número de revisões realizadas.

A partir da análise, pretende-se identificar padrões recorrentes e variáveis críticas que impactam o sucesso de revisões, contribuindo para a melhoria de práticas de colaboração e qualidade de código em ambientes distribuídos de desenvolvimento.

1.1 Hipóteses Informais

Com base nas observações do ecossistema de desenvolvimento open-source, foram elaboradas as seguintes hipóteses informais para orientar a investigação:

IH01: Pull requests menores, com menos arquivos e linhas alteradas, têm maior probabilidade de serem aprovados.

IH02: PRs analisados em tempo moderado (entre 1h e 48h) tendem a ser mais aceitos, enquanto análises muito longas reduzem a taxa de merge.

IH03: PRs com descrições mais completas aumentam a chance de aprovação, por facilitar a compreensão do revisor.

IH04: Um número moderado de interações (comentários e participantes) está positivamente associado à aceitação do PR, enquanto muitas interações podem indicar divergências.

IH05: PRs menores demandam menos ciclos de revisão, reduzindo o esforço necessário do revisor.

IH06: O tempo de análise está positivamente correlacionado com o número de revisões. Revisões mais longas tendem a envolver mais ciclos de feedback.

IH07: Descrições detalhadas diminuem o número de revisões necessárias, pois fornecem contexto adequado desde o início.

IH08: PRs com mais interações (comentários, revisores e discussões) tendem a passar por mais revisões até sua aprovação final.

1.2 Objetivos

O objetivo principal deste trabalho é caracterizar empiricamente a atividade de *code review* em repositórios populares do GitHub, analisando fatores que influenciam o resultado e a dinâmica das revisões de código. A partir de métricas quantitativas extraídas de *pull requests* (PRs), busca-se responder às seguintes questões de pesquisa:

- **RQ01:** Qual a relação entre o tamanho dos *pull requests* e o feedback final das revisões?
- **RQ02:** Qual a relação entre o tempo de análise dos *pull requests* e o feedback final das revisões?
- **RQ03:** Qual a relação entre a descrição dos *pull requests* e o feedback final das revisões?
- **RQ04:** Qual a relação entre as interações nos *pull requests* e o feedback final das revisões?
- **RQ05:** Qual a relação entre o tamanho dos *pull requests* e o número de revisões realizadas?
- **RQ06:** Qual a relação entre o tempo de análise dos *pull requests* e o número de revisões realizadas?
- **RQ07:** Qual a relação entre a descrição dos *pull requests* e o número de revisões realizadas?
- **RQ08:** Qual a relação entre as interações nos *pull requests* e o número de revisões realizadas?

Como objetivo secundário, pretende-se validar ou refutar as hipóteses informais elaboradas, contribuindo para o entendimento dos fatores que afetam o sucesso e a eficiência do processo de revisão de código em projetos *open-source*.

2 Metodologia

2.1 Coleta de Dados

A coleta de dados foi realizada através da API GraphQL do GitHub, utilizando um script em Python desenvolvido especificamente para esta análise. O processo de coleta contemplou informações relacionadas às atividades de revisão de código, englobando tanto características dos *pull requests* (PRs) quanto atributos do processo de *code review*. As etapas seguiram o seguinte fluxo:

Autenticação: Utilização de token de acesso pessoal do GitHub para autenticar as requisições à API e permitir a coleta de grandes volumes de dados.

Consulta e Paginação: Implementação de consultas em GraphQL para coletar dados dos 200 repositórios mais populares do GitHub, considerando o número de estrelas como critério de popularidade. Foi utilizado um mecanismo de paginação para garantir a coleta completa de todos os 200 PRs por repositório.

Filtragem dos PRs: Seleção apenas dos PRs com status *MERGED* ou *CLOSED*, que possuíam ao menos uma revisão (*review count* > 0) e tempo total de análise superior a uma hora, a fim de descartar revisões automáticas realizadas por *bots* ou pipelines de CI/CD.

Extração de Métricas de Revisão: Para cada PR foram coletadas informações sobre tamanho (arquivos e linhas modificadas), tempo de análise (diferença entre *createdAt* e *closedAt*), descrição (tamanho do texto em *markdown*), interações (número de comentários e participantes) e número total de revisões.

Tratamento de Erros: Implementação de mecanismos de controle de taxa (*rate limiting*) e de repetição automática de requisições em caso de falhas, assegurando consistência e integridade dos dados coletados.

2.2 Métricas Coletadas

Para cada *pull request* analisado, foram coletadas métricas relacionadas tanto ao processo de revisão quanto às características estruturais do PR. As métricas foram agrupadas em quatro dimensões principais, conforme descrito a seguir:

Tamanho: número de arquivos modificados e total de linhas adicionadas/removidas. **Tempo de Análise:** intervalo de tempo entre a criação e o fechamento do PR. **Descrição:** número de caracteres no corpo da descrição do PR, em formato *markdown*. **Interações:** número total de participantes e comentários na discussão. **Revisões:** quantidade de revisões formais registradas no campo *review count*. **Status Final:** estado final do PR, podendo ser *MERGED* (aceito) ou *CLOSED* (rejeitado).

Essas métricas permitem analisar as correlações entre características dos PRs e o resultado das revisões, respondendo às questões de pesquisa propostas no trabalho.

2.3 Processamento dos Dados

2.4 Exportação e Análise

2.5 Limitações

3 Resultados

Referências