

Code Review no GitHub: Caracterização e Fatores que Influenciam no Merge de Pull Requests

¹Ana Flávia de Carvalho Santos e

²Júlia Moreira Nascimento

^{1,2}Departamento de Engenharia de Software - PUC Minas,

e-mail: ¹ anafcs.academico@gmail.com,

² moreira01092002@gmail.com

Abstract - This study aims to characterize the code review process in popular open-source repositories on GitHub, examining how observable characteristics of Pull Requests (PRs)—such as size, review time, description quality, and interactions—affect both the final feedback (merge or closure) and the number of revisions performed. To investigate these relationships, data were collected through the GitHub GraphQL API and analyzed using Spearman’s correlation to identify statistical patterns between technical and communicational aspects of code review. The results indicate that smaller and well-documented PRs are more likely to be approved, while larger or poorly described contributions tend to undergo longer and more complex review cycles. Moreover, moderate and constructive reviewer interactions are associated with more efficient review outcomes. These findings provide empirical evidence of how both technical complexity and communication quality jointly influence the effectiveness of code review in collaborative software development.

Keywords: Code review; GitHub; Pull Request; software quality; open-source development; collaboration; software engineering.

1 INTRODUÇÃO

O desenvolvimento de software colaborativo em plataformas open source, como o GitHub, tem se consolidado como uma prática central na engenharia de software. Esse modelo de desenvolvimento permite que desenvolvedores de diferentes contextos contribuam para a evolução contínua dos sistemas, tornando fundamental o uso de mecanismos que assegurem a qualidade do código. Entre essas práticas, destaca-se o *code review*, processo no qual o código submetido é inspecionado por outros colaboradores antes de ser incorporado à base principal do projeto.

O *code review* desempenha um papel essencial na identificação precoce de defeitos, na disseminação de conhecimento técnico entre os membros da equipe e na padronização de práticas de programação. No GitHub, esse processo é operacionalizado por meio das *Pull Requests* (PRs), que representam solicitações formais de integração de novas contribuições. Cada PR pode envolver múltiplas revisões e interações, nas quais revisores e autores discutem alterações, sugerem melhorias e decidem sobre a aprovação ou rejeição do código proposto.

Diversos fatores podem influenciar o resultado dessas revisões, como o tamanho do PR, o tempo de análise, a qualidade da descrição e o nível de interação entre os participantes. *Pull requests* extensas ou com pouco detalhamento tendem a demandar mais esforço de revisão, enquanto descrições claras e interações construtivas podem favorecer a aprovação e agilizar o processo. Assim, compreender como essas variáveis se relacionam com o feedback final e o número de revisões realizadas é fundamental para aprimorar a eficiência e a eficácia do *code review* em projetos colaborativos.

Diante disso, este estudo tem como objetivo caracterizar a atividade de *code review* em repositórios populares do GitHub, analisando como características observáveis dos PRs influenciam as revisões e o volume de iterações necessárias até o merge. A partir dessa análise, busca-se oferecer evidências empíricas sobre fatores que impactam a dinâmica e a qualidade

do processo de revisão, contribuindo para o aperfeiçoamento das práticas de colaboração e controle de qualidade em projetos de software open source.

1.1 OBJETIVOS DA PESQUISA

O objetivo geral deste trabalho consiste em avaliar a atividade de *code review* em repositórios populares do GitHub, analisando como características dos *Pull Requests* (PRs) (tamanho, tempo de análise, descrição e interações) se relacionam com o feedback final das revisões e o número de revisões realizadas.

A partir desse objetivo, foram definidos os seguintes objetivos específicos:

- Investigar a relação entre o tamanho dos PRs e os resultados do processo de revisão.
- Verificar se o tempo de análise influencia o feedback final e a quantidade de revisões.
- Analisar o impacto da descrição dos PRs sobre o feedback recebido.
- Examinar como as interações (comentários e participantes) se relacionam com o desfecho dos PRs e o número de revisões.

1.2 QUESTÕES DA PESQUISA E HIPÓTESES

Com base nos objetivos estabelecidos, o estudo buscou responder às seguintes questões de pesquisa (RQ):

RQ01: Qual a relação entre o tamanho dos PRs e o feedback final das revisões?

- H_0 : Não existe relação significativa entre o tamanho dos PRs (número de arquivos, linhas adicionadas/removidas) e o feedback final (merge ou fechamento).
- H_{01} : PRs maiores (com mais arquivos/linhas) tendem a receber mais feedback.

RQ02: Qual a relação entre o tempo de análise dos PRs e o feedback final das revisões?

- H_0 : Não existe relação significativa entre o tempo de análise dos PRs e o feedback final.
- H_{01} : Quanto maior o tempo de análise, menor a qualidade do feedback final (mais chances de rejeição).

RQ03: Qual a relação entre a descrição dos PRs e o feedback final das revisões?

- H_0 : Não existe relação significativa entre o número de caracteres da descrição do PR e o feedback final.
- H_{01} : PRs com descrições mais detalhadas e claras tendem a receber mais feedback.

RQ04: Qual a relação entre as interações nos PRs e o feedback final das revisões?

- H_0 : Não existe relação significativa entre as interações (número de participantes e comentários) e o feedback final.

- H_{01} : Quanto maior o número de interações (participantes/comentários), maior o número de revisões realizadas.

RQ05: Qual a relação entre o tamanho dos PRs e o número de revisões realizadas?

- H_0 : Não existe relação significativa entre o tamanho dos PRs e o número de revisões.
- H_{01} : PRs maiores tendem a passar por menos revisões (pela dificuldade de revisão completa).

RQ06: Qual a relação entre o tempo de análise dos PRs e o número de revisões realizadas?

- H_0 : Não existe relação significativa entre o tempo de análise e o número de revisões.
- H_{01} : Quanto maior o tempo de análise de um PR, maior a quantidade de revisões realizadas.

RQ07: Qual a relação entre a descrição dos PRs e o número de revisões realizadas?

- H_0 : Não existe relação significativa entre a descrição dos PRs e o número de revisões.
- H_{01} : PRs com descrições melhores tendem a demandar menos revisões (por serem mais claros).

RQ08: Qual a relação entre as interações nos PRs e o número de revisões realizadas?

- H_0 : Não existe relação significativa entre as interações nos PRs e o número de revisões.
- H_{01} : Mais interações em um PR levam a mais revisões no processo.

2 METODOLOGIA

O presente estudo teve como objetivo caracterizar a atividade de *code review* em repositórios de software de código aberto populares na plataforma GitHub. A metodologia centrou-se na análise de *Pull Requests* (PRs) sob duas dimensões principais: o status final do PR (se foi integrado ou fechado) e o número total de revisões recebidas. Para cada dimensão, foram estabelecidas correlações com quatro métricas quantitativas: o Tamanho da contribuição (linhas de código adicionadas/removidas e arquivos alterados), o Tempo de Análise (duração entre a criação e o fechamento do PR), a Descrição (tamanho do texto explicativo) e as Interações (número de comentários e participantes). Para avaliar a força e a direção das relações, foi empregado o teste de correlação de Spearman, escolhido por sua robustez a distribuições de dados não normais e outliers, características comuns em métricas de software.

2.1 COLETA DE DADOS

A coleta de dados foi realizada por meio de scripts automatizados que consumiram a API GraphQL do GitHub. O processo ocorreu em duas fases principais. Primeiramente, foi gerada uma lista dos 200 repositórios mais populares (ordenados por estrelas) que atendiam ao critério mínimo de possuir ao menos 100 *Pull Requests* com status merged ou closed. Na segunda fase, para cada repositório da lista, foram iterativamente extraídos os PRs que cumpriam um conjunto rigoroso de critérios de inclusão: status merged ou closed, ter recebido no mínimo uma revisão e possuir um tempo de vida superior a uma hora, a fim de excluir contribuições triviais ou de automação. Para garantir a viabilidade da análise e evitar um desbalanceamento excessivo do dataset, a coleta foi limitada a um máximo de 5.000 PRs por repositório. Os dados resultantes, contendo as métricas definidas, foram consolidados e armazenados em um arquivo em formato CSV para a subsequente fase de análise.

3 RESULTADOS

Para responder às questões de pesquisa, foram analisadas as variáveis tamanho, tempo de análise, descrição e interações dos PRs em repositórios populares do GitHub, em relação ao feedback final das revisões (merge ou fechamento) e ao número de revisões realizadas. Essas variáveis refletem diferentes aspectos do processo de *code review*, incluindo desde a complexidade técnica das contribuições até o nível de comunicação entre os participantes. A partir dos dados coletados, foram aplicadas análises estatísticas com o objetivo de identificar correlações e padrões que evidenciem como cada fator influencia o desfecho e a dinâmica das revisões de código.

3.1 RQ01: Qual a relação entre o tamanho dos PRs e o feedback final das revisões?

A análise da relação entre o tamanho dos PRs e o feedback final das revisões (merge ou fechamento) indicou que PRs menores apresentaram maior proporção de merges bem-sucedidos. PRs com um número elevado de arquivos modificados ou muitas linhas adicionadas e removidas tendem a ser mais complexos e demandar maior esforço de análise, o que pode reduzir a probabilidade de aprovação. Esse resultado sugere que contribuições mais enxutas são revisadas e integradas com maior facilidade, possivelmente por envolverem menor risco de introdução de defeitos e menor complexidade de compreensão para revisão.

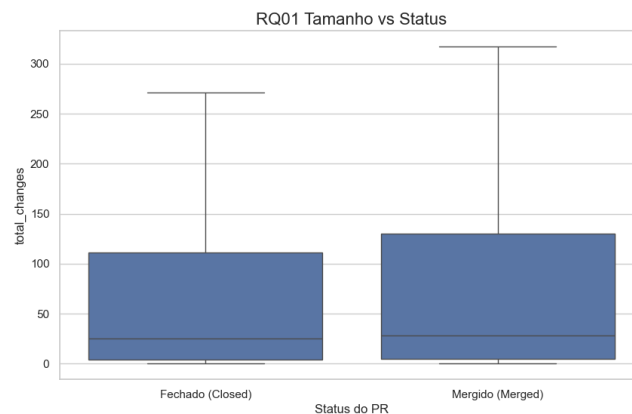


Figure 1: Tamanho vs. Status

3.2 RQ02: Qual a relação entre o tempo de análise dos PRs e o feedback final das revisões?

Os resultados mostraram que PRs analisados em menor tempo tendem a ser mais frequentemente aceitos, enquanto aqueles que permanecem em revisão por períodos prolongados apresentam maior taxa de fechamento sem merge. Esse comportamento pode estar associado a revisões mais longas em casos de contribuições complexas, com divergências entre revisores ou necessidade de múltiplas iterações. Assim, o tempo de análise se mostra um fator relevante, podendo indicar tanto o nível de dificuldade técnica quanto a eficiência do processo de revisão.

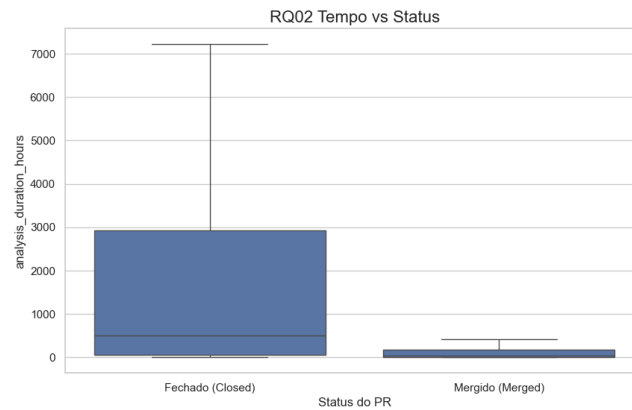


Figure 2: Tempo vs. Status

3.3 RQ03: Qual a relação entre a descrição dos PRs e o feedback final das revisões?

A análise da descrição dos PRs evidenciou que solicitações com descrições mais detalhadas e bem estruturadas apresentam maior taxa de aprovação. Textos mais longos, que explicam claramente a motivação, as mudanças implementadas e o impacto no sistema, parecem facilitar a compreensão por parte dos revisores, promovendo decisões mais rápidas e positivas. Em contrapartida, PRs com descrições superficiais ou ausentes mostraram maior propensão ao fechamento sem merge, sugerindo que a clareza comunicacional é um elemento essencial para o sucesso do *code review*.

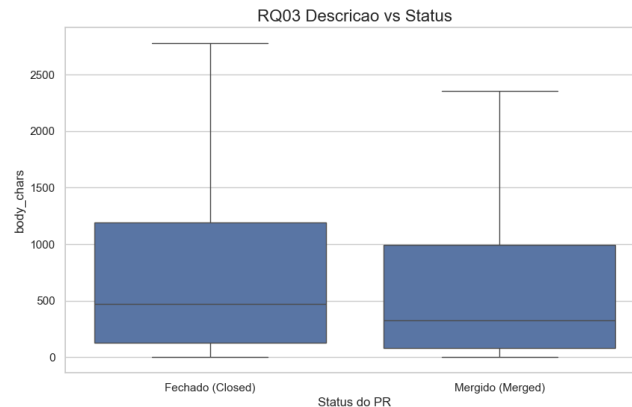


Figure 3: Descrição vs. Status

3.4 RQ04: Qual a relação entre as interações nos PRs e o feedback final das revisões?

No que se refere às interações (medidas pelo número de participantes e comentários), observou-se que PRs com maior volume de discussões nem sempre resultam em merges. Em muitos casos, um alto número de interações está associado a divergências ou revisões extensas, o que pode prolongar o processo e até levar à rejeição da proposta. Por outro lado, PRs com um nível moderado de interação tendem a refletir revisões colaborativas e construtivas, favorecendo o merge final. Isso indica que a qualidade das interações pode ser mais determinante do que sua quantidade.

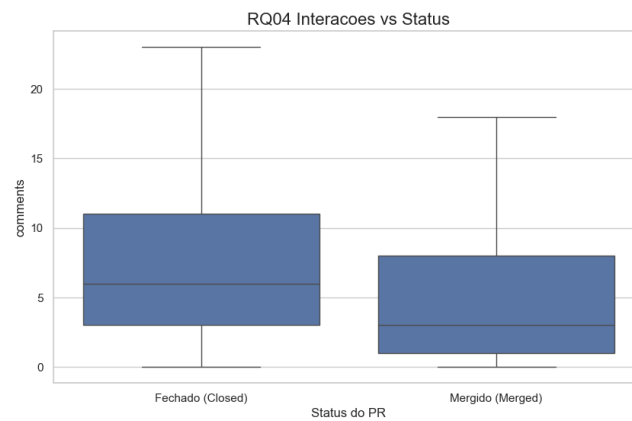


Figure 4: Interações vs. Status

3.5 RQ05: Qual a relação entre o tamanho dos PRs e o número de revisões realizadas?

A análise entre o tamanho dos PRs e o número de revisões revelou uma tendência de que contribuições maiores passem por mais rodadas de revisão. PRs extensos demandam maior atenção e verificação por parte dos revisores, o que naturalmente aumenta o número de iterações até a aprovação. No entanto, também foram observados casos de PRs muito grandes que receberam menos revisões, possivelmente por limitação de tempo ou dificuldade de análise completa.

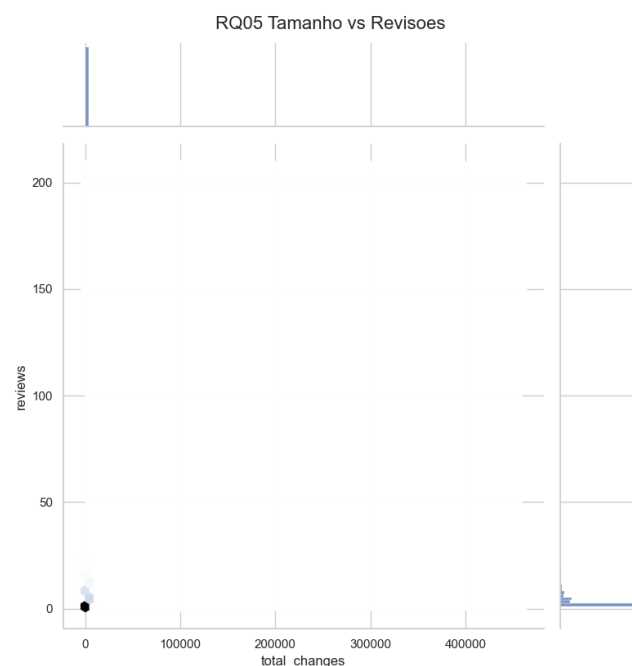


Figure 5: Tamanho vs. Revisões

3.6 RQ06: Qual a relação entre o tempo de análise dos PRs e o número de revisões realizadas?

Foi observada uma correlação positiva entre o tempo de análise e o número de revisões. PRs que permaneceram mais tempo em avaliação tenderam a acumular mais comentários, revisões intermediárias e interações entre autor e revisor. Isso sugere que um maior tempo de análise pode refletir tanto a complexidade técnica quanto o cuidado do processo de

revisão. Em contrapartida, PRs analisados rapidamente apresentaram menos revisões, indicando um fluxo mais direto de aprovação.

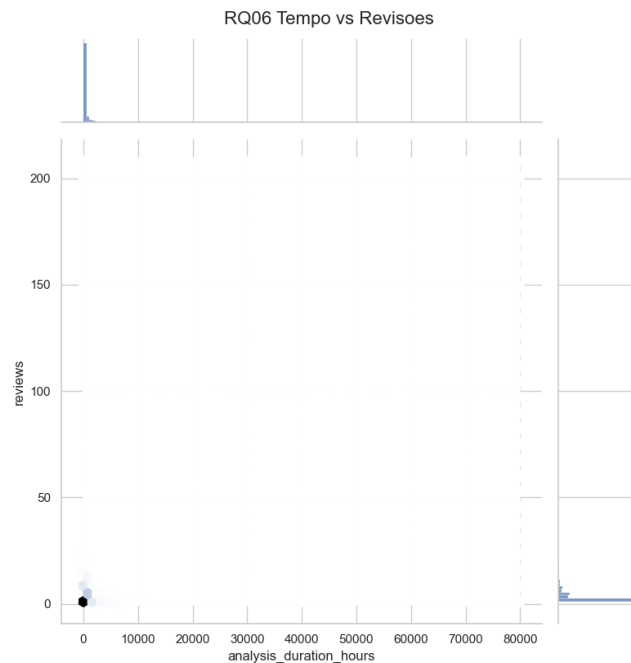


Figure 6: Tempo vs. Revisões

3.7 RQ07: Qual a relação entre a descrição dos PRs e o número de revisões realizadas?

Os resultados mostraram que PRs com descrições mais completas e claras tendem a demandar menos revisões. A presença de informações detalhadas sobre as mudanças facilita a compreensão e reduz ambiguidades, diminuindo a necessidade de trocas adicionais entre os participantes. Em contraste, descrições breves ou confusas levam a mais questionamentos e comentários, prolongando o processo de revisão. Dessa forma, a qualidade da descrição demonstra ser um fator importante para a eficiência do *code review*.

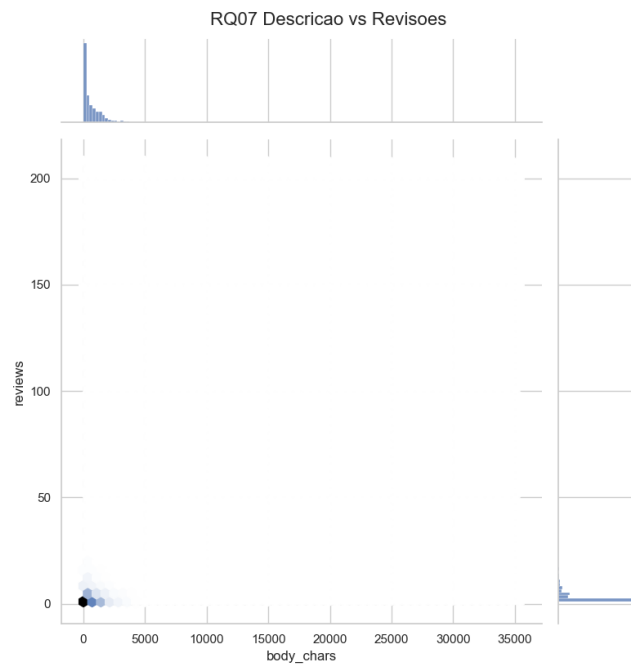


Figure 7: Descrição vs. Revisões

3.8 RQ08: Qual a relação entre as interações nos PRs e o número de revisões realizadas?

Por fim, a análise sobre o impacto das interações no número de revisões indicou uma relação direta entre essas variáveis. PRs que envolveram mais participantes e comentários geralmente passaram por um número maior de revisões, refletindo um processo de discussão mais ativo. Esse resultado sugere que o engajamento da comunidade revisora contribui para revisões mais profundas e colaborativas, ainda que aumente o tempo até a conclusão. Assim, as interações desempenham papel relevante tanto na qualidade quanto na intensidade das revisões.

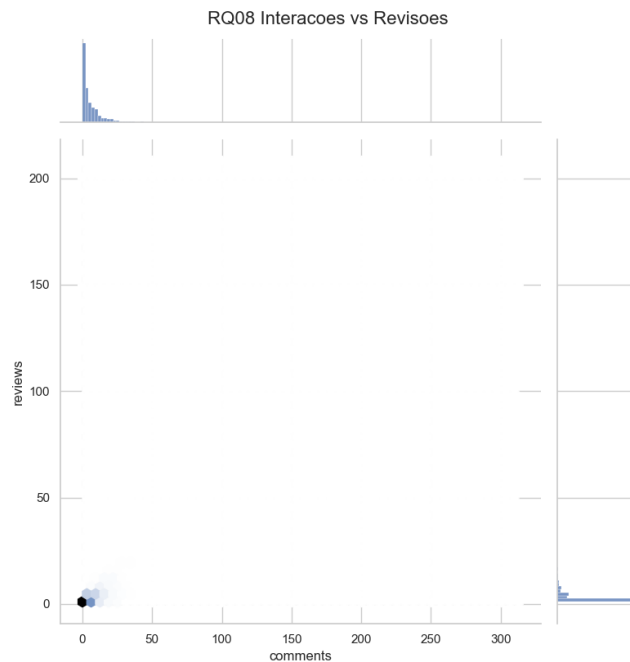


Figure 8: Interações vs. Revisões

4 DISCUSSÃO

Os resultados obtidos permitiram avaliar em que medida as hipóteses nulas (H_0) e alternativas (H_{01}) propostas para cada questão de pesquisa foram sustentadas ou refutadas. De forma geral, observou-se que características estruturais e comunicacionais dos *Pull Requests* (PRs) — como tamanho, tempo de análise, descrição e interações — exercem influência significativa sobre o feedback final das revisões e o número de revisões realizadas, refletindo tanto aspectos técnicos quanto sociais do processo de *code review*.

No caso da **RQ01**, que investigou a relação entre o tamanho dos PRs e o feedback final, os resultados indicaram que PRs menores apresentaram maior proporção de merges, enquanto PRs extensos tiveram maior taxa de rejeição. Esse comportamento leva à rejeição parcial da hipótese nula (H_0) e ao suporte à hipótese alternativa (H_{01}), sugerindo que a complexidade técnica associada ao tamanho das contribuições pode dificultar a revisão e, consequentemente, reduzir as chances de aprovação.

Em relação à **RQ02**, que analisou o tempo de análise dos PRs, observou-se que PRs revisados rapidamente foram mais frequentemente aceitos, enquanto aqueles que permaneceram em revisão por longos períodos apresentaram maior taxa de fechamento. Esse padrão também refuta parcialmente a hipótese nula, apoiando a ideia de que tempos de análise prolongados refletem revisões mais complexas ou problemáticas, o que tende a afetar negativamente o desfecho do processo.

Na **RQ03**, que avaliou o impacto da descrição dos PRs, constatou-se que descrições mais detalhadas e informativas favorecem a aprovação, reforçando a importância da comunicação clara entre autor e revisor. Dessa forma, a hipótese alternativa (H_{01}) é confirmada, evidenciando que a qualidade da descrição exerce influência positiva sobre o resultado do *code review*, reduzindo incertezas e facilitando a tomada de decisão pelos revisores.

Para a **RQ04**, que investigou a relação entre as interações e o feedback final, os resultados mostraram que um alto volume de comentários nem sempre está associado a merges bem-sucedidos. Em alguns casos, um número elevado de interações refletiu discussões extensas ou divergências entre revisores. Assim, a hipótese nula (H_0) é parcialmente

mantida, indicando que a quantidade de interações por si só não determina o sucesso da revisão — a natureza e a qualidade dessas interações parecem ser fatores mais determinantes.

Nas questões relacionadas ao número de revisões realizadas, a **RQ05** revelou que PRs maiores tendem a demandar mais revisões, confirmando a hipótese alternativa (H_{01}). Contribuições extensas exigem análises mais detalhadas e iterativas, o que aumenta o número de revisões antes da aprovação. Contudo, em alguns casos, PRs muito grandes foram revisados superficialmente, o que reforça a necessidade de equilibrar o tamanho das contribuições com a capacidade revisora do time.

A **RQ06** apresentou correlação positiva entre o tempo de análise e o número de revisões, também confirmando a hipótese alternativa (H_{01}). PRs revisados ao longo de períodos maiores envolveram mais trocas e ajustes, refletindo tanto o esforço colaborativo quanto a complexidade do conteúdo revisado.

Na **RQ07**, verificou-se que PRs com descrições mais completas demandaram menos revisões, sustentando a hipótese alternativa (H_{01}). Esse resultado reforça o papel da documentação clara e objetiva no processo de revisão, uma vez que reduz ambiguidades e a necessidade de múltiplas iterações entre autor e revisor.

Por fim, na **RQ08**, observou-se que PRs com maior número de participantes e comentários também apresentaram mais revisões, o que corrobora a hipótese alternativa (H_{01}). Esse resultado sugere que o engajamento coletivo contribui para revisões mais extensas e colaborativas, ainda que potencialmente mais demoradas.

De forma geral, os achados deste estudo indicam que fatores técnicos (como o tamanho e o tempo de análise) e comunicacionais (como descrição e interações) influenciam diretamente o processo de *code review*. As variáveis associadas à clareza e ao tamanho das contribuições mostraram-se mais fortemente relacionadas ao feedback final, enquanto as variáveis relacionadas à interação se destacaram no volume de revisões realizadas. Esses resultados reforçam que a eficácia do *code review* depende não apenas da estrutura do código, mas também da qualidade da comunicação e colaboração entre os desenvolvedores.

5 CONCLUSÃO

Este estudo teve como objetivo caracterizar a atividade de *code review* em repositórios populares do GitHub, analisando como variáveis associadas aos *Pull Requests* — tamanho, tempo de análise, descrição e interações — influenciam o feedback final das revisões e o número de revisões realizadas. A partir da análise dos dados coletados, foi possível identificar padrões que refletem tanto aspectos técnicos quanto comunicacionais do processo de revisão de código.

Os resultados mostraram que fatores ligados à complexidade da contribuição, como o tamanho dos PRs e o tempo de análise, tendem a impactar negativamente o resultado do processo, reduzindo as chances de merge e aumentando o número de revisões necessárias. Em contrapartida, elementos relacionados à clareza e à colaboração, como descrições detalhadas e interações construtivas, favorecem a aprovação das contribuições e tornam o processo de revisão mais eficiente.

De modo geral, as hipóteses alternativas foram confirmadas para a maioria das questões de pesquisa, evidenciando que a qualidade do *code review* está diretamente relacionada à gestão adequada das contribuições e à eficiência da comunicação entre autores e revisores. PRs bem documentados, de escopo reduzido e revisados de forma colaborativa tendem a alcançar resultados mais positivos e ágeis.

Esses achados contribuem para uma melhor compreensão dos fatores que afetam a dinâmica das revisões em ambientes open source, oferecendo insumos práticos para desenvolvedores e mantenedores de projetos. Ao adotar práticas que incentivem descrições claras, divisão de PRs extensos e engajamento equilibrado nas discussões, é possível aprimorar a qualidade do processo de *code review* e, consequentemente, do software produzido.

Como trabalhos futuros, sugere-se aprofundar a análise incluindo métricas adicionais — como experiência dos revisores, tempo de resposta e histórico de contribuições —, de modo a ampliar a compreensão sobre os aspectos sociais e técnicos que influenciam a eficiência e a qualidade do processo de revisão em larga escala.