

Relatório Final - Laboratório 03

**Gabriel Alejandro Figueiro Galindo¹, Marcelo Aguilar Araújo D’Almeida¹,
Philippe Roberto Dutra Chaves Vieira¹**

¹Instituto de Ciências Exatas e Informática
Pontifícia Universidade de Minas Gerais (PUC Minas)
Belo Horizonte – MG – Brasil

1. Introdução e Hipóteses

A prática de code review envolve a interação entre desenvolvedores e revisores para inspecionar o código produzido, garantindo sua integração à base principal sem riscos de introduzir novos erros à aplicação. Em sistemas open source, especialmente nos desenvolvidos no GitHub, esse processo ocorre por meio da avaliação de contribuições submetidas via Pull Requests (PRs). Para que um código seja incorporado à branch principal, é necessário que um colaborador do projeto analise e discuta a solicitação, decidindo por aprovar ou rejeitar o merge.

Neste relatório, o objetivo é analisar a atividade de code review em repositórios populares do GitHub, identificando variáveis que influenciam a aprovação de um PR sob a perspectiva dos desenvolvedores que contribuem para os projetos selecionados. Com base nas questões de pesquisa propostas, foram formuladas as seguintes hipóteses informais:

- RQ 01: PRs maiores tendem a receber feedbacks mais críticos ou negativos, pois são mais difíceis de revisar e têm maior chance de conter erros ou violações de estilo. Revisores podem se sentir sobrecarregados com mudanças extensas.
- RQ 02: Quanto maior o tempo de análise de um PR, maior a chance de o feedback ser negativo ou exigir mudanças, pois atrasos podem indicar que o revisor encontrou problemas ou está com dúvidas em relação à contribuição.
- RQ 03: PRs com descrições mais detalhadas e claras tendem a receber feedbacks mais positivos, pois facilitam o entendimento do propósito da mudança e demonstram cuidado por parte do autor.
- RQ 04: Um maior número de interações (como comentários e discussões) pode estar associado a feedbacks mais construtivos ou detalhados, mesmo que inicialmente negativos, pois indicam um processo colaborativo de melhoria da contribuição.
- RQ 05: PRs maiores provavelmente demandam mais ciclos de revisão, já que o volume de código aumenta a complexidade da análise e pode exigir ajustes em múltiplos pontos.
- RQ 06: Um tempo de análise maior pode indicar revisões mais aprofundadas ou múltiplas interações, o que pode estar relacionado a um maior número de revisões feitas antes da decisão final.
- RQ 07: PRs com descrições completas tendem a passar por menos revisões, pois os revisores conseguem entender rapidamente o escopo da mudança e se concentram diretamente na qualidade do código.
- RQ 08: Um maior número de interações tende a resultar em mais revisões, já que cada troca de mensagens pode levar a novas versões do código até que todas as sugestões ou objeções sejam resolvidas.

Após a coleta de dados, essas hipóteses foram analisadas e discutidas para verificar sua validade.

2. Metodologia

3. Resultados Obtidos

4. Conclusão