

## Understanding Developers' Work - A Visual Approach for Project Managers - Resumo Estendido - CTDSI 2022

Matheus Silva Ferreira, Heitor Costa

<sup>1</sup>Universidade Federal de Lavras (UFLA)  
Lavras – MG – 37200-900 – Brazil

matheussf95@gmail.com, heitor@ufla.br

A compreensão do trabalho dos desenvolvedores pode auxiliar na realização de “boa” gestão do projeto. Ao compreender o trabalho realizado pelos membros de um time, pode-se mensurar a distribuição do conhecimento sobre o código-fonte, sendo uma importante informação para realizar o controle de riscos (*e.g.*, rotatividade de desenvolvedores) [Ferreira et al. 2017, Ricca and Marchetto 2010]. A compreensão do trabalho dos desenvolvedores pode auxiliar o Gerente de Projetos (GP) a conhecer pontos fortes e fracos de sua equipe e favorecer, por exemplo, montagem de equipes e definição de responsabilidades [Gao and Liu 2015, Lima et al. 2015, Zuser and Grechenig 2003]. Percepções empíricas do GP sobre o trabalho dos desenvolvedores podem ser insuficientes para apoiar atividades de gestão de projetos. Nesse sentido, diferentes abordagens foram propostas para auxiliar o GP a compreender o trabalho dos desenvolvedores [Jaruchotrattanasakul et al. 2016, Feiner and Andrews 2018, Ferreira et al. 2017, González-Torres et al. 2016, Lima et al. 2015]. Apesar disso, ainda há lacunas a serem exploradas, como considerar informações evolutivas do projeto de software e diferentes níveis de granularidade sobre o trabalho dos desenvolvedores. Neste trabalho, o objetivo foi auxiliar GP a compreender o trabalho dos desenvolvedores. Para tanto, foi proposta a abordagem *Developer Tracker*, que permite ao GP interagir com elementos visuais para compreender informações quantitativas a respeito do trabalho realizado pelos desenvolvedores. Os elementos visuais incluem uma técnica de visualização de software e as informações quantitativas são obtidas a partir de dados extraídos de SCV. Este texto trata-se de um resumo estendido de "Understanding Developers' Work - A Visual Approach for Project Managers" no SBQS 2021 [Ferreira et al. 2021b].

Alguns estudos relacionados abordam formas diferentes de auxiliar o GP a visualizar e compreender o trabalho do desenvolvedor [González-Torres et al. 2016, Feiner and Andrews 2018]. Além disso, há autores que buscaram apoiar o recrutamento de desenvolvedores [Jaruchotrattanasakul et al. 2016]. De modo geral, os principais diferenciais deste trabalho são a visualização de informações com diferentes características, que inclui aspectos do projeto e de cada desenvolvedor, a comparação entre duas versões de um sistema de software, o que permite identificar características evolutivas, e a utilização de uma técnica visual, que fornece perspectivas diferentes e com múltiplos níveis de granularidade.

A abordagem *Developer Tracker* consiste em um conjunto de etapas para coletar informações registradas em repositórios de código, aplicar medidas quantitativas e apresentar visualmente os resultados aos GP. O objetivo de apresentar medidas quantitativas ([Ferreira et al. 2020]) de maneira visual é auxiliar a interpretação de

informações relacionadas ao trabalho de cada desenvolvedor e à evolução do sistema de software. *Developer Tracker* possui 5 etapas [Ferreira et al. 2021b]: i) Extrair Dados de SCV; ii) Aplicar Medidas Quantitativas; iii) Mapear Estruturas Visuais; iv) Gerar Visualizações; e v) Manipular Visualização. A última etapa pode ser considerada uma etapa complementar, pois sua execução não é obrigatória. O primeiro passo para realizar o estudo experimental foi implementar a abordagem *Developer Tracker* como um apoio computacional para web (*Developer Tracker App* [Ferreira et al. 2021a]). Em seguida, foram planejadas e realizadas avaliações individuais com voluntários. Em cada avaliação, há caracterização dos voluntários (participantes) que, em seguida, executam um conjunto de tarefas e respondem questões não utilizando e utilizando *Developer Tracker App*. As respostas das questões foram analisadas qualitativamente, permitindo identificar efeitos provocados pelo uso do *Developer Tracker App*, bem como as opiniões dos participantes. Os voluntários foram GP atuantes na indústria de software. Para condução da avaliação, foi definido um protocolo composto por termo de consentimento, conjunto de tarefas a serem executadas, roteiro de entrevistas e sequência de atividades para executar a avaliação. As respostas foram armazenadas em um formulário do *Google Forms* e os dados coletados estão disponíveis em <https://zenodo.org/record/4974757>.

Quanto aos resultados, os participantes realizaram 5 tarefas sem utilizar *Developer Tracker App*. Em seguida, responderam questões para confirmar se concluíram ou não as tarefas. De modo geral, a quantidade de tarefas concluídas foi superior à quantidade de tarefas não concluídas. Um destaque foi 12 participantes (P1 - P12) terem concluído todas as tarefas. Além disso, chama atenção o fato de eles utilizarem conhecimento empírico para realizar as tarefas. Esse fato pode ser explicado considerando a relevância de compreender os aspectos abordados nas tarefas na rotina dos participantes. Geralmente, um GP acompanha a rotina de trabalho do time e lida com a alocação de pessoas e o gestão de atividades. Assim, as tarefas podem ser consideradas essenciais para esses participantes e, portanto, eles possuem as respostas de forma quase imediata. Nesse sentido, P5 justifica "... acompanho o projeto desde o início e já são pontos de preocupação em minha rotina". Além de afirmar que as tarefas abordam "pontos de preocupação", a justificativa de P5 inclui o tempo de acompanhamento do projeto. De modo semelhante, outros participantes citaram o tempo de acompanhamento do projeto como justificativa. Em resumo, os participantes consideram que uma visão do "histórico" (P13, P15, P16), obtida pelo "prazo" (P14) de acompanhamento, muitas vezes, "desde o início" (P5, P6) do projeto, incrementado à "vivência" (P4) e à participação no "dia a dia da equipe" (P6, P9, P10, P12), fornecem insumos para compreender aspectos a respeito do trabalho dos desenvolvedores abordados nas tarefas. Por fim, os participantes expressaram sua opinião sobre o grau de utilidade da abordagem. O grau de utilidade poderia ser "Extremamente útil", "Útil", "Inútil" ou "Inútil e prejudicial". Nenhum participante considerou as opções "Inútil" e "Inútil e prejudicial". Nove participantes (56,6%) consideraram a abordagem "Útil" e a principal justificativa é conseguir informações valiosas. Além disso, 7 participantes (43,4%) consideraram essencial, escolhendo "Extremamente útil". Concluindo, *Developer Tracker App* foi considerado útil para apoiar os participantes em atividades relacionadas à compreensão do trabalho desenvolvedores. Mas, para colocá-la em produção, melhorias devem ser realizadas no uso das medidas e na geração das visualizações.

Neste trabalho, o objetivo foi apoiar GP a compreender o trabalho dos desenvolvedores utilizando uma abordagem para visualizar medidas quantitativas aplicadas sobre informações mineradas de SCV. Para tanto, foi elaborada a abordagem *Developer Tracker*, a qual foi avaliada por meio de um estudo experimental e os resultados foram apresentados e discutidos de forma qualitativa. Algumas sugestões de trabalhos futuros são: i) visualizar mais de um repositório; e ii) modificar o escopo de artefatos considerados no cálculo das medidas.

## Referências

- Feiner, J. and Andrews, K. (2018). Repovis: Visual overviews and full-text search in software repositories. In *Working Conference on Software Visualization*, pages 1–11.
- Ferreira, M., Valente, M. T., and Ferreira, K. (2017). A comparison of three algorithms for computing truck factors. In *International Conference on Program Comprehension*, pages 207–217.
- Ferreira, M. S., Júnior, P. A., and Costa, H. (2021a). Developer tracker app: Uma ferramenta para visualizar o trabalho dos desenvolvedores. In *Simpósio Brasileiro de Engenharia de Software*.
- Ferreira, M. S., Júnior, P. P., and Costa, H. A. X. (2021b). Understanding developers' work-a visual approach for project managers. In *XX Brazilian Symposium on Software Quality*, pages 1–10.
- Ferreira, M. S., Martins, L. A., Júnior, P. A. P., and Costa, H. (2020). How is the work of developers measured? an industrial and academic exploratory view. *Journal of Software Engineering Research and Development*, 8:8–1.
- Gao, M. and Liu, C. (2015). Teamwatch demonstration: A web-based 3d software source code visualization for education. In *International Workshop on Code Hunt Workshop on Educational Software Engineering*, pages 12–15.
- González-Torres, A., García-Peñalvo, F. J., Therón-Sánchez, R., and Colomo-Palacios, R. (2016). Knowledge discovery in software teams by means of evolutionary visual software analytics. *Science of Computer Programming*, 121:55–74.
- Jaruchotrattanasakul, T., Yang, X., Makihara, E., Fujiwara, K., and Iida, H. (2016). Open source resume (osr): A visualization tool for presenting oss biographies of developers. In *7th International Workshop on Empirical Software Engineering in Practice*, pages 57–62.
- Lima, J., Treude, C., Figueira Filho, F., and Kulesza, U. (2015). Assessing developer contribution with repository mining-based metrics. In *International Conference on Software Maintenance and Evolution*, pages 536–540.
- Ricca, F. and Marchetto, A. (2010). Are heroes common in floss projects? In *International Symposium on Empirical Software Engineering and Measurement*, pages 1–4.
- Zuser, W. and Grechenig, T. (2003). Reflecting skills and personality internally as means for team performance improvement. In *Conference on Software Engineering Education and Training*, pages 234–241.