**SUMÁRIO**

[1 Introdução 3](#_Toc8051398)

[1.1 Introdução 3](#_Toc8051399)

[1.2 Motivação 4](#_Toc8051400)

[1.3 Delineamento do problema 5](#_Toc8051401)

[1.4 Objetivo e Questão da Pesquisa 6](#_Toc8051402)

[2 Revisão da Literatura e Metodologia 8](#_Toc8051403)

[2.1 Breve histórico da qualidade de produto de software e trabalhos relacionados 8](#_Toc8051404)

[2.2 Diferentes necessidades de informação e semelhantes necessidades de tomar decisões sobre o produto de software 13](#_Toc8051405)

[2.3 Ciclo de Vida de Desenvolvimento de Sistemas Contemporâneos 15](#_Toc8051406)

[2.4 Metodologia 17](#_Toc8051407)

[2.4.1 - Fase Estratégica 18](#_Toc8051408)

[2.4.1.1 - Resultados da Pesquisa 18](#_Toc8051409)

[2.4.1.2 - Lógica da Pesquisa 18](#_Toc8051410)

[2.4.1.3 - Propósito da Pesquisa 19](#_Toc8051411)

[2.4.1.4 - Abordagem da Pesquisa 19](#_Toc8051412)

[2.4.2 - Fase Tática 20](#_Toc8051413)

[2.4.2.1 - Processo de Pesquisa 20](#_Toc8051414)

[2.4.2.2 - Métodos de Pesquisa 20](#_Toc8051415)

[2.4.3 - Fase Operacional 21](#_Toc8051416)

[2.4.3.1 - Métodos para Coleta de Dados 21](#_Toc8051417)

[2.4.3.2 - Métodos de Análise de Dados 22](#_Toc8051418)

[Referências Bibliográficas 24](#_Toc8051419)

**ÍNDICE DE FIGURAS**

[Figura 2: Relacionamento entre as características de qualidade de produto da ISO 25010 14](#_Toc8057101)

[Figura 3: Pontos de observação da qualidade de versões de produtos de software 19](#_Toc8057102)

[Figura 4: Estrutura e Desenho da Pesquisa 20](#_Toc8057103)

# MeasureSoftGram: o modelo matemático

## Metodologia

A partir de um compêndio de estudos metodológicos, Wohlin e Aurum (2015) propuseram uma estrutura conceitual que fundamenta e auxilia a tomada de decisão em relação ao desenho (*design*) de uma pesquisa em engenharia de software experimental. A escolha do desenho de pesquisa mais apropriado para um determinado estudo tem sido problemática, principalmente, em função: i) da quantidade de diferentes métodos experimentais disponíveis e a falta de um completo entendimento dos pesquisadores sobre tais; ii) das atividades da engenharia de software serem tipicamente desempenhadas por pessoas. Segundo os autores, muitos pesquisadores da área de engenharia de software experimental têm relatado essas dificuldades nas duas últimas décadas.

A estrutura conceitual proposta Wohlin e Aurum (2015) é composta por um conjunto de pontos de decisões, que o pesquisador deve se atentar e perceber no decorrer de um projeto de pesquisa, que são organizados em três grandes fases: estratégica, tática e operacional.

Essa estrutura não segue uma abordagem linear, sequencial, sendo comum que diferentes pontos de tomada de decisão ocorram em diferentes momentos do projeto da pesquisa. Dessa maneira, a estrutura é sucessivamente refinada, na medida em que a pesquisa evolua. O importante é que o pesquisador tenha em mente quais são os pontos de tomada de decisão e que esses pontos implicam em diferentes caminhos a serem seguidos no percurso da pesquisa, de forma a guiar a escolha das diferentes estratégias e métodos em uma investigação experimental. Apresentamos o desenho desta pesquisa, ilustrado por meio da Figura 4.

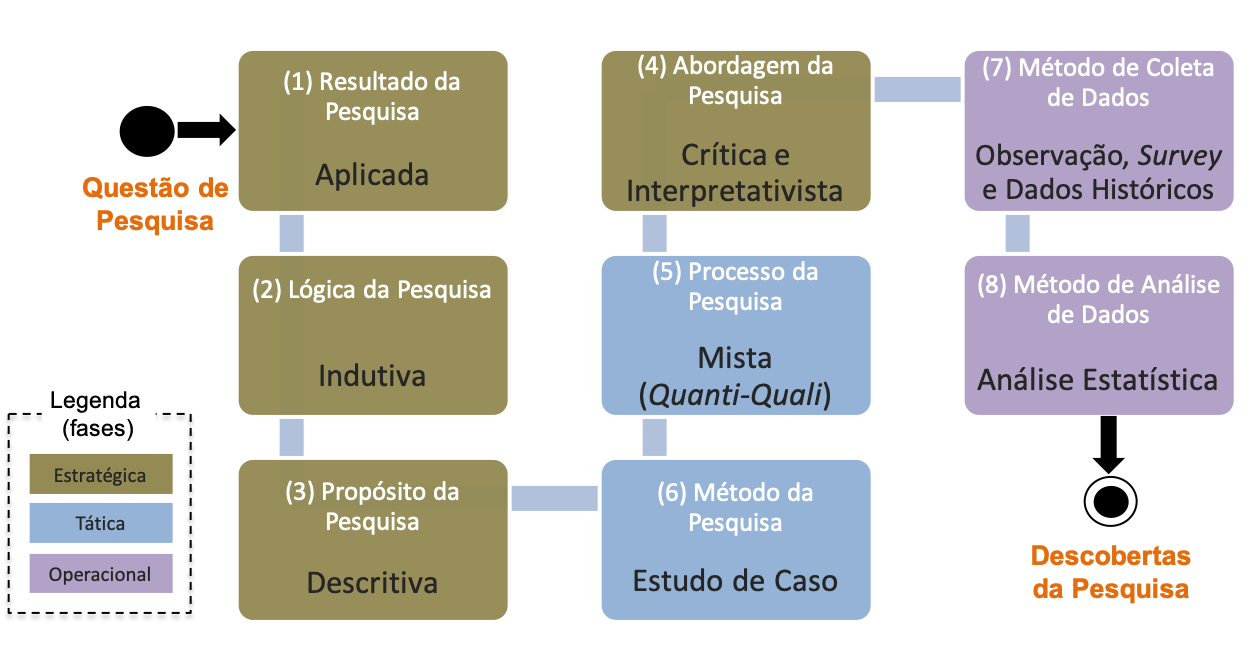


Figura 4: Estrutura e Desenho da Pesquisa

## - Fase Estratégica

Esta fase está relacionada: i) aos resultados da pesquisa; ii) a lógica de construção da; iii) ao propósito e iv) a abordagem a ser utilizada na pesquisa.

O ponto de partida, é a definição da questão de pesquisa, considerado crítico, uma vez que, endereça um problema a ser resolvido e influencia o restante do processo. A questão de pesquisa deste trabalho, bem como o objetivo foram apresentados na subseção 1.7.

## - Resultados da Pesquisa

Os resultados podem ser do tipo pesquisa básica ou aplicada. Neste trabalho temos uma pesquisa aplicada, uma vez que procuramos propor uma solução para o problema apresentado na subseção 1.6, ou seja, um problema específico. Futuramente, esperamos contribuir para melhorar o estado da prática da análise da qualide de produto de software. Cabe ressaltar, que, tipicamente a pesquisa aplicada se beneficia dos conhecimentos construídos na pesquisa básica.

## - Lógica da Pesquisa

Refere-se a direção a ser seguida na pesquisa, podendo partir de um problema geral para um específico e vice-versa. Destaca-se a lógica dedutiva e a indutiva. Percebemos essa pesquisa como do tipo indutiva, uma vez que, fazemos uso de uma abordagem bottom-up, já que, pretendemos estabelecer conceitos teóricos e padrões, a partir da observação de dados reais de projetos de software.

## - Propósito da Pesquisa

Pode ser classificado como exploratório, descritivo, explanatório ou de avaliação. Consideramos este trabalho como uma pesquisa descritiva. Pesquisas descritivas são aplicadas para descrever um fenômeno ou características de um problema. São mais objetivas que as pesquisas exploratórias e normalmente procuram entender ‘O quê’ ou ‘Como’ ao se descrever um fenômeno. Nesta pesquisa, pretendemos entender como modelar e analisar a qualidade de produto de software de forma multivariada e multiespacial, de forma a identificarmos padrões entre os dados que nos forneçam *insights* para entender as relações entre as características e subcaracterísticas que descrevem a qualidade do produto de software. Esperamos que as observações e análise nos forneçam subsídios para que o modelo matemático definido no MeasureSoftGram possa apoiar a decisão sobre a aceitação de versões de produto de software, por parte do time de desenvolvimento e gestores de produtos de software.

## - Abordagem da Pesquisa

De acordo com (Easterbrook et al., 2007) as principais correntes de pensamento com influência em engenharia de software são: positivisma; interpretativista; teoria crítica e pragmatismo. Ainda há o destaque de que o positivismo e interpretativismo são as abordagens que maior ocorrência na literatura.

O **interpretativismo**, tem o objetivo de entender as atividades realizadas pelos seres humanos, na perspectiva dos participantes das investigações científicas. Nessa abordagem há um claro direcionamento de se considerar o ser humano, exceto os observadores, como parte do fenômeno observado. Com isso, aspectos culturais e de contexto são profundamente estudados de forma a subsidiar as conclusões em uma investigação científica. Há uma tendência de uso de métodos qualitativos. Já a **teoria crítica**, se baseia na premissa que variáveis sociais e culturais influenciam diretamente o comportamento de um sistema e, portanto, as interconexões não podem ser ignoradas. Essa abordagem objetiva relevar conflitos e contradições observadas em um fenômeno e difere-se do positivismo e interpretativismo na medida em que, a observação de um fenômeno procura prever ou explicar a situação atual do comportamento de um sistema em observação. Tendem a usar métodos qualitativos e estudos longitudinais. Entendemos que essas duas correntes de pensamento são as de principal influência neste trabalho, embora, haja forte influência da corrente positivista, percebida pela utilização de processo quantitativo na pesquisa.

Em essência, o MeasureSoftGram opera com um modelo matemático que possui um conjunto de regras de transformação, normalização e agregação de medidas de qualidade. Entretanto, o produto de software é desenvolvido por pessoas, em diferentes contextos. Assim, as regras de interpretação das medidas são fornecidas por especialistas, e, principalmente, a decisão de aceitação das versões de produto são tomadas por engenheiros de software e gestores de desenvolvimento de produto, influenciados e dependentes das características do contexto de desenvolvimento. Logo, possui um caráter intrínseco de subjetividade.

## - Fase Tática

Uma vez endereçadas as questões associadas a fase estratégica da pesquisa, a fase tática foca em definir o processo e metodologia de pesquisa a serem utilizados.

## - Processo de Pesquisa

Os dois processos de pesquisa mais reconhecidos na literatura são chamados de pesquisa qualitativa ou quantitativa. O processo qualitativo fornece subsídios que favorecem observar um fenômeno a partir das características sociais e culturais associadas a este. É utilizado quando o pesquisador procura entender a perspectiva dos participantes envolvidos no fenômeno. Já os processos quantitativos, são mais apropriados para apoiar a construção de teorias, procurando descrever de maneira objetiva detalhes relacionados ao objeto de estudo. Há uma forte ênfase na utilização de métricas, medição numérica e análise de dados fazendo uso de técnicas estatísticas.

Dessa forma, consideramos que o processo de pesquisa deste trabalho é misto. Além da descrição das características sociais e culturais serem mandatórias (ex: linguagem de programação, experiência dos engenheiros de software, experiência dos usuários com o uso do produto de software); há a sensibilidade dos diferentes contextos de desenvolvimento (ex: web, mobile, embarcado) que influenciam as decisões no ciclo de vida de um produto. Por outro lado, provemos um modelo matemático que representa numericamente o fenômeno da qualidade.

## - Métodos de Pesquisa

Uma parte fundamental em um projeto de pesquisa é a escolha do de pesquisa. Nessa etapa o pesquisador decide qual a metodologia de pesquisa, ou combinação delas, é mais indicada para o fenômeno que deseja observar. Na verdade, é comum a combinação de metodologias em uma investigação científica na área de engenharia de software. Destacam-se: estudo de caso; pesquisa-ação; entrevista com especialistas; estudos etnográficos e experimentos controlados.

O método que julgamos melhor nos apoiar na investigação desta pesquisa é o estudo de caso de múltiplos casos. O estudo de caso consiste na observação de um fenômeno em seu ambiente natural, ou seja, onde ele ocorre. Não há intervenção do observador ao longo da execução do estudo de caso. Faz uso de variados métodos de coleta de dados afim de se obter informações oriundas de diferentes fontes. Pode ser conduzido como um caso único, quando é um caso revelador, informativo, crítico ou único; ou de múltiplos casos, quando é possível a replicação em diferentes locais. Faz uso de processos qualitativos, quantitativos, ou mistos, como é o caso desta pesquisa.

Inicialmente, pretendemos caracterizar a modelagem multidimensional da qualidade de produto de software com a escolha de duas ou três medidas, de maneira a compreendermos essa modelagem e, principalmente, como caracterizar e analisar as relações mútuas entre as características e subcaracterísticas da qualidade. Para tanto, utilizaremos repositórios de software-livre como fonte primária de informação, na qual, utilizaremos técnicas de mineração e recuperação de informação, que nos auxilie a caracterizar os projetos a serem analisados, bem como as métricas extraídas desses projetos.

Pretendemos reforçar a confiança de nossas análises e conclusões iniciais, triangulando as fontes de informação, em diferentes contextos. Em particular, em *startups* e em contratos de prestação de serviço de desenvolvimento software, em organizações públicas federais brasileira.

Travassos, gostaria de citar algum trabalho seu que expressasse muitas das discussões que já tivemos sobre sua experiência com o uso de estudo de caso em investigações em engenharia de software.

## - Fase Operacional

O foco dessa fase é a escolha das técnicas de coleta de dados, bem como a análise e interpretação destes, de forma a produzir conclusões sobre a pesquisa.

## - Métodos para Coleta de Dados

A coleta de dados é guiada pela questão de pesquisa, bem como pela metodologia de pesquisa escolhida, onde são usados métodos qualitativos e quantitativos. Segundo Wohlin e Aurum (2015), os métodos mais relevantes para coleta de dados em engenharia de software são: entrevistas; observação participante; pesquisa em dados históricos; questionários; experimentos e simulação.

Neste trabalho, inicialmente, utilizaremos pesquisa em dados históricos de projetos de software livre, disponíveis na plataforma Github. Posteriormente selecionaremos, por conveniência, diferentes organizações, especialmente startups de desenvolvimento de software e organizações públicas federais que desenvolvem ou contratam serviço de desenvolvimento de software. Nesses contextos, pretendemos utilizar entrevistas, questionários e observação participante.

## - Métodos de Análise de Dados

De acordo com o processo de pesquisa adotado os dados coletados são do tipo qualitativo ou quantitativo. A depender desse tipo, diferentes métodos de análise podem ser utilizados para interpretação e análise dos dados. Considerando a área de engenharia de software e, o tipo de processo de pesquisa utilizado, podemos entender que os métodos de análise mais comuns são:

* para dados de natureza qualitativa destacam-se os métodos hermenêutica, análise temática e *grounded theory*. Vale salientar que esses métodos fazem uso da técnica de *coding* que é inicialmente aplicada.
* para dados de natureza quantitativa destaca-se o uso de estatística descritiva (média, moda, mediana, análise de percentis, desvio padrão e variância) ou estatística de inferência (teste de hipótese, regressão, mineração de dados). É importante ressaltar que, de acordo com a distribuição dos dados, os métodos ainda podem ser classificados como paramétricos ou não-paramétricos, o que influencia, por exemplo, a escolha do teste de hipótese.

Como nesta pesquisa definimos um processo do tipo misto, para dados de natureza qualitativa, podemos realizar análise temática de dados obtidos em entrevistas ou em questões abertas em questionários. Já para os dados de natureza quantitativa, utilizaremos estatística descritiva e de inferência.

As medidas, que dão significados às métricas, são normalizadas em nosso modelo. Para analisar as distribuições tanto de métricas, quanto de medidas coletadas, utilizaremos estatística descritiva. Já para interpretar numericamente as medidas, utilizaremos estatística de inferência e interpretações trigonométricas.

O mecanismo de normalização proposto no modelo matemático visa endereçar a transformação tanto das medidas qualitativas, quanto das quantitativas, em uma representação numérica com escala razão que varia entre 0 e1, sendo 0-pior e 1-melhor. Uma vez normalizadas, as medidas são modeladas em tensores no espaço, onde operações algébricas, multilineares, são utilizadas para agregar e reduzir o multiespaço a um espaço Euclidiano[[1]](#footnote-2). Isso nos dá capacidade de utilizar interpretações sobre a reta (ex: regressão numérica) e, é dessa maneira que propomos analisar o comportamento da qualidade de software ao longo do tempo.

Nosso *surrogate* para comparar a medida agregada no último nível hierárquico (qualidade total), é suportado por aplicações trigonométricas em espaços vetoriais, especialmente a diferença de cossenos[[2]](#footnote-3), medida utilizada para calcular a similaridade entre vetores. Essa medida se tornou consagrada na área de buscas e recuperação de informação, utilizada no algoritmo *PageRank*, proposto por Lawrence Page e Sergey Brin. (Page et al., 1999).

# Referências Bibliográficas

ABDELLATIF, T. M.; CAPRETZ, L. F.; HO, D. Software Analytics to Software Practice: A Systematic Literature Review. Proceedings of the First International Workshop on BIG Data Software Engineering. Anais...: BIGDSE ’15.Piscataway, NJ, USA: IEEE Press, 2015 Disponível em: <http://dl.acm.org/citation.cfm?id=2819289.2819300>

ACAR, E.; Canan, A-B; Haluk, B.; Rasmus, B., Bülent, Y. Multiway analysis of epilepsy tensors. Proceedings 15th International Conference on Intelligent Systems for Molecular Biology {(ISMB)} & 6th European Conference on Computational Biology (ECCB), Vienna, Austria, July 21-25, 2007. Disponível em: <https://doi.org/10.1093/bioinformatics/btm210>

AGRAWAL, R.; GOLSHAN, B.; PAPALEXAKIS, E. A Study of Distinctiveness in Web Results of Two Search Engines. Proceedings of the 24th International Conference on World Wide Web-WWW ’15 Companion.New York, NY, USA: ACM, 2015 Disponível em: <http://doi.acm.org/10.1145/2740908.2743060>

ALDAAJEH, S. H.; AL-QUTAISH, R. E.; AL-QIREM, F. A tactic-based framework to evaluate the relationships between the software product quality attributes. International Journal of Software Engineering, v. 5, n. 1, p. 5–26, 2012.

ALDAAJEH, S. et al. Communing Different Views on Quality Attributes Relationships’ Nature. European Journal of Scientific Research, v. 68, n. 1, p. 101–109, 2012.

AKIYAMA, F. An Example of Software System Debugging. Inform Process Congress. p. 353-359, 1971

BAKSHY, E.; ECKLES, D.; BERNSTEIN, M. S. Designing and Deploying Online Field Experiments. Proceedings of the 23rd ACM conference on the World Wide Web. Anais...2014

BARRETO, E. L. Cálculo tensorial. [s.l.] Instituto de Matemática Pura e Aplicada do Conselho Nacional de Pesquisas, 1965.

BASILI, V. R. A Personal Perspective on the Evolution of Empirical Software Engineering. In: Perspectives on the Future of Software Engineering. [s.l.] Springer Berlin Heidelberg, 2013. p. 255–273.

BECK, K. et al. Manifesto for Agile Software DevelopmentManifesto for Agile Software Development, 2001. Disponível em: <http://www.agilemanifesto.org/>

BIRD, C.; MENZIES, T.; ZIMMERMANN, T. The Art and Science of Analyzing Software Data. 1st. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2015.

BOEHM, B. W. Characteristics of Software Quality. [s.l.] North-Holland, 1978.

BOEHM, B. W.; BROWN, J. R.; LIPOW, M. Quantitative Evaluation of Software Quality. Proceedings of the 2Nd International Conference on Software Engineering. Anais...: ICSE ’76.Los Alamitos, CA, USA: IEEE Computer Society Press, 1976Disponível em: <http://dl.acm.org/citation.cfm?id=800253.807736>

BOSCH, J. Building Products as Innovation Experiment Systems. (M. A. Cusumano, B. Iyer, N. Venkatraman, Eds.)Software Business. Anais...Berlin, Heidelberg: Springer Berlin Heidelberg, 2012

BOSCH, J. et al. The Early Stage Software Startup Development Model: A Framework for Operationalizing Lean Principles in Software Startups. (B. Fitzgerald et al., Eds.)Lean Enterprise Software and Systems. Anais...Berlin, Heidelberg: Springer Berlin Heidelberg, 2013

BUSE, R. P. L.; ZIMMERMANN, T. Information needs for software development analytics. Proceedings - International Conference on Software Engineering-ICSE. 2012 Disponível em: <http://www.scopus.com/inward/record.url?eid=2-s2.0-84864194945&partnerID=40&md5=c3501a80b7d9cdd917a665867484ee78>

BUSE, R. P. L.; ZIMMERMANN, T. Analytics for Software Development. Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research. Anais...: FoSER ’10.New York, NY, USA: ACM, 2010 Disponível em: <http://doi.acm.org/10.1145/1882362.1882379>

CHOMA, J.; GUERRA, E. M.; DA SILVA, T. S. Patterns for Implementing Software Analytics in Development Teams. Proceedings of the 24th Conference on Pattern Languages of Programs-PLoP ’17.USA: The Hillside Group, 2017 Disponível em: <http://dl.acm.org/citation.cfm?id=3290281.3290308>

CZERWONKA, J. et al. CODEMINE: Building a Software Development Data Analytics Platform at Microsoft. [S.l.]: IEEE Computer Society. July 2013

DE FRANÇA, B. B. N.; JERONIMO JUNIOR, H.; TRAVASSOS, G. H. Characterizing DevOps by Hearing Multiple Voices. Proceedings of the 30th Brazilian Symposium on Software Engineering. Anais...: SBES ’16.New York, NY, USA: ACM, 2016Disponível em: <http://doi.acm.org/10.1145/2973839.2973845>

DE MENDONCA, Vinicius Rafael Lobo et al. Static analysis techniques and tools: A systematic mapping study. 8th International Conference on Software Engineering Advances-ICSEA, 2013.

DEBOIS, P. Devops: A software revolution in the making. The Journal of Information Technology Management, v. 24, p. 3–5, 2011.

DIMITRI N., Kleanthis N. Mokios, Nicholas D. Sidiropoulos, and Alexandros Potamianos. 2010. Batch and adaptive PARAFAC-based blind separation of convolutive speech mixtures. IEEE Transactions on Audio, Speech, and Language Processing, 18, 6 (2010), 1193–1207. DOI: http://dx.doi.org/10.1109/tasl.2009.2031694

DINGSØYR, T. D. T. Agile Project Management: From Self-Managing Teams to Large-Scale Development. 37th IEEE/ACM International Conference on Software Engineering, {ICSE} 2015, Florence, Italy, May 16-24, 2015, Volume 2. Anais...2015Disponível em: <https://doi.org/10.1109/ICSE.2015.299>

DROMEY, R. G. (1995). A model for software product quality. IEEE Transactions on Software Engineering, 21(2), 146–162. doi:10.1109/32.345830.

EASTERBROOK, S. et al. Selecting Empirical Methods for Software Engineering Research. [s.l.] Springer, 2007.

ELBERZHAGER, Frank; MÜNCH, Jürgen; NHA, Vi Tran Ngoc. A systematic mapping study on the combination of static and dynamic quality assurance techniques. Information and software technology, v. 54, n. 1, p. 1-15, 2012.

EVANGELIA Pantraki and Constantine Kotropoulos. 2015. Automatic image tagging and recommendation via PARAFAC2. In 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP’15). IEEE, 1–6. DOI:http://dx.doi.org/10.1109/mlsp.2015.7324363

ACAR E., Canan Aykut-Bingol, Haluk Bingol, Rasmus Bro, and Bu ̈lent Yener. 2007. Multiway analysis of epilepsy tensors. Bioinformatics 23, 13 (2007), i10–i18. DOI:http://dx.doi.org/10.1093/ bioinformatics/btm210

FAGERHOLM, F.; GUINEA, A.S.; Mäenpää, H.; Munch J. The RIGHT Model for Continuous Experimentation. Journal of Systems and Software, v. 123, p. 292–305, jan. 2017.

FABIJAN, A.; Pavel D.; Olsson, H. H.; Bosch, J., The Evolution of Continuous Experimentation in Software Product Development: From Data to a Data-driven Organization at Scale. Proceedings of the 39th International Conference on Software Engineering. Anais...: ICSE ’17.Piscataway, NJ, USA: IEEE Press, 2017 Disponível em: <https://doi.org/10.1109/ICSE.2017.76>

FELDERER, M. et al. Industrial Evaluation of the Impact of Quality-driven Release Planning. Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. ESEM, New York, USA, 2014, Disponível em: <http://doi.acm.org/10.1145/2652524.2652579>

FENTON, N. & B. J. Software Metrics-A Rigorous and Practical Approach, Third Edition. Third ed. [s.l: s.n.].

FITZGERALD, B.; STOL, K.-J. Continuous software engineering: A roadmap and agenda. The Journal of Systems & Software, v. 123, n. Complete, p. 176–189, 2017.

HANSEN, S.; PLANTENGA, T. D.; KOLDA, T. G. Newton-based optimization for Kullback-Leibler nonnegative tensor factorizations. Optimization Methods and Software, v. 30, n. 5, p. 1002–1029, 2015.

HAOUES, M. et al. A guideline for software architecture selection based on ISO 25010 quality related characteristics. International Journal of System Assurance Engineering and Management, v. 8, n. 2, p. 886–909, 2017.

HENNINGSSON, K.; WOHLIN, C. Understanding the Relations Between Software Quality Attributes - A Survey Approach, 12th International Conference on Software Quality

HOOVER, C. L.; ROSSO-LLOPART, M.; TARAN, G. Evaluating Project Decisions: Case Studies in Software Engineering. [s.l.] Addison-Wesley, 2010.

INSTITUTE, P. M. Um Guia do Conhecimento em Gerenciamento de Projetos (Guia PMBOK). Pennsylvania, 2014.

ISO/IEC 25010 System and software quality models. [s.l.] 2010

ISO/IEC 9126 Software engineering -- Product quality. [s.l.] ISO/IEC, 2001.

ISO/IEC:15939 Software Engineering - Software Measurement Process. International Organization for Standardization and International Electrotechnical Commission. [S.l.]. 2002

JURISTO, N.; MORENO, A. M. Basics of Software Engineering Experimentation. 1st. ed. [s.l.] Springer Publishing Company, Incorporated, 2010.

KAI-WEI Chang, Wen-tau Yih, and Christopher Meek. 2013. Multi-relational latent semantic analysis. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP’13). 1602–1612

KEVIC, K.; Kevic, K.; Murphy, B.; Williams, L.; Beckmann, J.; Characterizing Experimentation in Continuous Deployment: A Case Study on Bing. Proceedings of the 39th International Conference on Software Engineering: Software Engineering in Practice Track-ICSE-SEIP ’17.Piscataway, NJ, USA: IEEE Press, 2017 Disponível em: <https://doi.org/10.1109/ICSE-SEIP.2017.19>

KITCHENHAM, B. What’s up with software metrics?--A preliminary mapping study. Journal of systems and software, v. 83, n. 1, p. 37–51, 2010.

KOHAVI, R. et al. Controlled Experiments on the Web: Survey and Practical Guide. Data Min. Knowl. Discov., v. 18, n. 1, p. 140–181, fev. 2009.

KOHAVI, R.; LONGBOTHAM, R.; WALKER, T. Online Experiments: Practical Lessons. IEEE Computer, v. 43, p. 82–85, 2010.

KOHAVI, R. et al. Online Controlled Experiments at Large Scale. Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining-KDD, New York, NY, USA: ACM, 2013 Disponível em: <http://doi.acm.org/10.1145/2487575.2488217>

LAVAZZA, L.; MORASCA, S. An Empirical Evaluation of Distribution-based Thresholds for Internal Software Measures. Proceedings of the The 12th International Conference on Predictive Models and Data Analytics in Software Engineering. Anais...: PROMISE 2016.New York, NY, USA: ACM, 2016Disponível em: <http://doi.acm.org/10.1145/2972958.2972965>

LIMA, P. et al. A Metrics Suite for code annotation assessment. Journal of Systems and Software, v. 137, p. 163–183, 2018.

LINDGREN, E.; MÜNCH, J. Software Development as an Experiment System: A Qualitative Survey on the State of the Practice. Lecture Notes in Business Information Processing. Anais...2015

LÓPEZ, L. et al. Q-Rapids Tool Prototype: Supporting Decision-Makers in Managing Quality in Rapid Software Development. CAiSE Forum. Anais...: Lecture Notes in Business Information Processing.Springer, 2018

LUZ, W. P.; PINTO, G.; BONIFÁCIO, R. Building a Collaborative Culture: A Grounded Theory of Well Succeeded Devops Adoption in Practice. Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement-ESEM, New York, NY, USA: ACM, 2018 Disponível em: <http://doi.acm.org/10.1145/3239235.3240299>

MARINHEIRO, A.; BERNARDINO, J. OpenBRR Evaluation of an Open Source BI Suite. Proceedings of the International C\* Conference on Computer Science and Software Engineering. Anais...: C3S2E ’13.New York, NY, USA: ACM, 2013Disponível em: <http://doi.acm.org/10.1145/2494444.2494463>

MCCABE, T. J. A Complexity Measure. IEEE Trans. Softw. Eng., v. 2, n. 4, p. 308–320, jul. 1976.

MCCALL, J.; K. RICHARDS, P.; F. WALTERS, G. Factors in Software Quality. Volume I, II and III. Concepts and Definitions of Software Quality. US Rome Air Development Center Reports, US Department of Commerce, USA, p. 168, 1977.

MENEELY, A.; SMITH, B.; WILLIAMS, L. Validating Software Metrics: A Spectrum of Philosophies. ACM Transactions on Software Engineering and Methodology (TOSEM)., v. 21, n. 4, p. 24:1--24:28, fev. 2013.

MENZIES, T. et al. Sharing Data and Models in Software Engineering. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2015.

MENZIES, T.; ZIMMERMANN, T. Software Analytics: So What? {IEEE} Software, v. 30, n. 4, p. 31–37, 2013.

MIGUEL, J. P.; MAURICIO, D.; RODRIGUEZ, G. A Review of Software Quality Models for the Evaluation of Software Products. CoRR, v. abs/1412.2977, 2014.

MORDAL-MANET, K. et al. The squale model - {A} practice-based industrial quality model. ICSM. Anais...{IEEE} Computer Society, 2009

MORGAN, J.; LIKER, J. K. The Toyota Product Development System: Integrating People, Process, and Technology. [s.l.] Taylor & Francis, 2006.

NEELY, S.; STOLT, S. Continuous Delivery? Easy! Just Change Everything (Well, Maybe It Is Not That Easy). 2013 Agile Conference, {AGILE} 2013, Nashville, TN, USA, August 5-9, 2013. Anais...2013Disponível em: <https://doi.org/10.1109/AGILE.2013.17>

OLSSON, H. H.; ALLAHYARI, H.; BOSCH, J. Climbing the “Stairway to Heaven” - A Mulitiple-Case Study Exploring Barriers in the Transition from Agile Development towards Continuous Deployment of Software. (V. Cortellessa, H. Muccini, O. Demirörs, Eds.)EUROMICRO-SEAA. Anais...IEEE Computer Society, 2012Disponível em: <http://dblp.uni-trier.de/db/conf/euromicro/euromicro2012.html#OlssonAB12>POPPENDIECK

OUHBI, S. et al. Evaluating Software Product Quality: A Systematic Mapping Study. 2014 Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement. Anais...out. 2014

PAGE, L.; Brin, S.; Motwani, R.; Winograd, T. The PageRank Citation Ranking: Bringing Order to the Web**.** [s.l.] Stanford InfoLab, nov. 1999. Disponível em: <http://ilpubs.stanford.edu:8090/422/>.

PANTRAKI, E.; KOTROPOULOS, C. Automatic image tagging and recommendation via PARAFAC2. 25th {IEEE} International Workshop on Machine Learning for Signal Processing, {MLSP} 2015, Boston, MA, USA, September 17-20, 2015. Disponível em: <https://doi.org/10.1109/MLSP.2015.7324363>

PAPALEXAKIS, E. E.; FALOUTSOS, C.; SIDIROPOULOS, N. D. Tensors for Data Mining and Data Fusion: Models, Applications, and Scalable Algorithms. ACM Trans. Intell. Syst. Technol., v. 8, n. 2, p. 16:1--16:44, out. 2016.

POPPENDIECK, M.; POPPENDIECK, T. Implementing Lean Software Development: From Concept to Cash (The Addison-Wesley Signature Series). [s.l.] Addison-Wesley Professional, 2006.

Rakesh Agrawal, Behzad Golshan, and Evangelos Papalexakis. 2015. A study of distinctiveness in web results of two search engines. In 24th International Conference on World Wide Web, Web Science Track. ACM. DOI:http://dx.doi.org/10.1145/2740908.2743060

REPÚBLICA, P. Lei na 8.666/93, de 21 de Junho de 1993, 1993. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/L8666cons.htm>

Samantha Hansen, Todd Plantenga, and Tamara G. Kolda. 2015. Newton-based optimization for Kullback- Leibler nonnegative tensor factorizations. Optimization Methods and Software 30, 5 (April 2015), 1002– 1029. DOI:http://dx.doi.org/10.1080/10556788.2015.1009977

SAMOLADAS, I. et al. The SQO-OSS quality model: measurement based open source software evaluation. Open source development, communities and quality. Anais...2008

SCHERMANN, G. et al. We’re doing it live: A multi-method empirical study on continuous experimentation. Information and Software Technology, v. 99, p. 41–57, jul. 2018.

SJØBERG, T. D. D. I. K.; CRUZES, D. S. What works for whom, where, when, and why?: on the role of context in empirical software engineering. 2012 {ACM-IEEE} International Symposium on Empirical Software Engineering and Measurement, {ESEM} ’12, Lund, Sweden - September 19 - 20, 2012. Anais...2012Disponível em: <http://doi.acm.org/10.1145/2372251.2372256>

SIAVVAS, M. G.; CHATZIDIMITRIOU, K. C.; SYMEONIDIS, A. L. {QATCH} - An adaptive framework for software product quality assessment. Expert Syst. Appl., v. 86, p. 350–366, 2017.

SLTI/MP. Instrução Normativa No 04, Secretaria de Logística de Tecnologia da Informação, Ministério do Planejamento 2014.

SVAHNBERG, M.; HENNINGSSON, K. Consolidating different views of quality attribute relationships. WoSQ@ICSE. Anais...DBLP:conf/icse/SvahnbergH09: {IEEE} Computer Society, 2009

TAHIR, A.; MACDONELL, S. G. A systematic mapping study on dynamic metrics and software quality. Software Maintenance (ICSM), 2012 28th IEEE International Conference on. Anais...2012

TANG, D. et al. Overlapping Experiment Infrastructure: More, Better, Faster Experimentation. Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Anais...: KDD ’10. New York, NY, USA: ACM, 2010, Disponível em: <http://doi.acm.org/10.1145/1835804.1835810>

TCU2471 TRIBUNAL DE CONTAS DA UNIÃO. Acórdão 2.471/2008- Plenário, 2008. Disponível em:<http://portal3.tcu.gov.br/portal/page/portal/TCU/comunidades/tecnologia_informacao/boas_praticas/TCUacordao2471.pdf>

WOHLIN, C. et al. Experimentation in Software Engineering. [s.l.] Springer Publishing Company, Incorporated, 2012.

WOHLIN, C.; AURUM, A. Towards a Decision-making Structure for Selecting a Research Design in Empirical Software Engineering. **Empirical Softw. Engg.**, v. 20, n. 6, p. 1427–1455, dez. 2015.

Yilun Wang, Yu Zheng, and Yexiang Xue. 2014. Travel time estimation of a path using sparse trajectories. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD’14). ACM, New York, NY, 25–34. DOI:http://dx.doi.org/10.1145/2623330.2623656

ZHANG, D. et al. Software Analytics As a Learning Case in Practice: Approaches and Experiences. Proceedings of the International Workshop on Machine Learning Technologies in Software Engineering. Anais...: MALETS ’11.New York, NY, USA: ACM, 2011 Disponível em: <http://doi.acm.org/10.1145/2070821.2070829>

WANG, Y.; ZHENG, Y.; XUE, Y. Travel Time Estimation of a Path Using Sparse Trajectories. Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining-KDD ’14.New York, NY, USA: ACM, 2014 Disponível em: <http://doi.acm.org/10.1145/2623330.2623656>

WAGNER, S. et al. The Quamoco Product Quality Modelling and Assessment Approach. Proceedings of the 34th International Conference on Software Engineering. Anais...: ICSE ’12.Piscataway, NJ, USA: IEEE Press, 2012Disponível em: <http://dl.acm.org/citation.cfm?id=2337223.2337372>

WASSERMAN, A.; PAL, M.; CHAN, C. Business Readiness Rating for Open Source. Proceedings of the EFOSS Workshop. Anais...Como, Italy: [s.d.]

1. <http://mathworld.wolfram.com/EuclideanSpace.html> [↑](#footnote-ref-2)
2. <https://www.themathpage.com/aTrig/sum-proof.htm> [↑](#footnote-ref-3)