

Mapeamento Sistemático sobre Vieses Cognitivos no Desenvolvimento de Software

Bárbara Beato Ribeiro, Josué de Almeida Resende, Thiago M. R. Ribeiro,
Rodrigo Pereira dos Santos, Sean W. M. Siqueira¹

¹Universidade Federal do Estado do Rio de Janeiro (UNIRIO)
Rio de Janeiro – RJ – Brasil

{barbara.ribeiro, josue.resende, thiago.m.ribeiro}@edu.unirio.br,
{rps, sean}@uniriotec.br

Abstract. *This systematic mapping study examines the effects of cognitive biases, pointing out which are the main types, in which circumstances they occur, the issues they cause, and their potential mitigation methods in an effort to understand how cognitive biases can occur in software development. Throughout all stages of software development, more than 40 cognitive biases, some issues, and potential solutions were found.*

Resumo. *Visando entender de quais formas os vieses cognitivos podem ocorrer no desenvolvimento de software, este mapeamento sistemático da literatura trata de investigar os impactos desses vieses cognitivos, apontando quais são os principais tipos, em quais situações eles ocorrem, os problemas que eles causam e seus possíveis métodos de mitigação. Foram identificados mais de 40 vieses cognitivos, alguns problemas e possíveis mitigações no desenvolvimento de software ao longo de suas fases.*

1. Introdução

Como a tentativa de simplificação de processos feita pelo nosso cérebro pode atrapalhar a forma como tomamos decisões? Como isso pode nos levar a desenvolver sistemas e soluções que conterão erros cujos efeitos talvez nem consigamos prever a médio e longo prazo? Essas falhas são possíveis de ocorrer como resultado de vieses cognitivos nas fases do processo de desenvolvimento de software (PDS). Os estudos iniciais sobre vieses cognitivos remetem à década de 1970 [Kahneman e Tversky 1972] e, desde então, os estudos nas áreas de psicologia, sociologia e pesquisa gerencial identificaram mais de 200 tipos de vieses cognitivos [Mohanani *et al.* 2018].

No contexto do desenvolvimento de sistemas de informação (SI), especialmente os que envolvem software, esse assunto vem sendo estudado desde a década de 1990 [Silverman 1990]. Nesse contexto, o fator humano é considerado um elemento chave [Somerville e Rodden 1996], dado que o PDS é afetado por pessoas [Chotisarn e Prompoon 2013b]. O ciclo de vida de um software, por exemplo, engloba várias fases com atividades realizadas por pessoas com diversos perfis. Portanto, vieses cognitivos podem alterar a forma como cada integrante executa o seu trabalho, trazendo consequências positivas ou negativas para o resultado.

Apesar do crescente interesse sobre vieses nos últimos anos na literatura, há uma escassez de pesquisas sobre técnicas de mitigação e bases teóricas na sua compreensão

e interpretação [Mohanani *et al.* 2018]. Torna-se fundamental entender como os vieses cognitivos afetam as diferentes fases do PDS, visando identificar possíveis reforços negativos que podem gerar resultados inadequados e buscando formas de evitá-los ou de mitigá-los. Nesse sentido, este mapeamento sistemático da literatura (MSL) apresenta um levantamento dos vieses cognitivos e como eles afetam o PDS. Com isso, a questão de pesquisa principal (QP) é elaborada: *Como os vieses cognitivos afetam o desenvolvimento de software?* O objetivo é entender quais desses vieses são tratados em estudos existentes na literatura, em quais fases do PDS eles podem ocorrer, quais problemas podem surgir e quais as possíveis mitigações utilizadas para resolvê-los ou amenizá-los.

Este estudo está organizado da seguinte forma: a Seção 2 apresenta os fundamentos, conceitos e outros estudos similares ao tema; a Seção 3 traz o método de pesquisa utilizado; a Seção 4 apresenta os resultados, respondendo as questões de pesquisa propostas; na Seção 5, discute-se sobre os vieses cognitivos nas diferentes fases do PDS; na Seção 6, introduz-se as possíveis ameaças à validade; e a Seção 7 apresenta algumas considerações finais.

2. Trabalhos Relacionados e Fundamentação Teórica

De acordo com Kahneman e Tversky (1972), um viés cognitivo é um padrão de desvio no julgamento que ocorre em certas situações, causando distorção perceptiva, julgamento impreciso, ou o que se chama de irracionalidade. Gigerenzer e Gaissmaier (2011) descrevem o viés cognitivo como uma espécie de heurística - uma estratégia que despreza parte da informação para agilizar a tomada de decisão.

O estudo de Browne e Parsons (2012) focou na análise e projeto de sistemas e complementa o estudo realizado por Davern *et al.* (2012) para cognição em SI. Por sua vez, Mohanani (2016) buscou os efeitos dos vieses cognitivos sobre o levantamento de requisitos e de projeto de software. Por fim, o estudo de Fagerholm *et al.* (2022) cobriu 50 anos de pesquisa de conceitos cognitivos em engenharia de software e questionou a relevância dos conceitos, dentre outros aspectos. Como o objetivo deste estudo é identificar os vieses cognitivos no contexto do PDS, foram identificadas lacunas em relação às fases do desenvolvimento de software e como isso afeta positiva ou negativamente o PDS.

3. Método de Pesquisa

Utilizando as diretrizes de Kitchenham e Charters (2007), foi elaborado um protocolo com as 3 fases para a execução deste MSL: (1) planejamento, (2) condução e (3) análise dos resultados. O processo foi realizado entre os meses de novembro e dezembro de 2022 por três pesquisadores e revisado por dois pesquisadores com ao menos 15 anos de experiência em estudos secundários. A Figura 1 resume as etapas do processo de seleção dos estudos, que teve a ferramenta computacional Parsif.al¹ como suporte.

Seguindo as instruções de Kitchenham e Charters (2007), foi utilizado o PIO (*Population, Intervention, Outcomes*), sendo “população” como “desenvolvedores”, “intervenção” como “vieses cognitivos” e “resultados” como “efeitos/impactos”. Com isso, ocorreu a definição de uma questão de pesquisa principal (QP: “*Como os vieses cognitivos afetam o desenvolvimento de software?*”) e quatro subquestões (SQ): SQ01)

¹<https://parsif.al/>

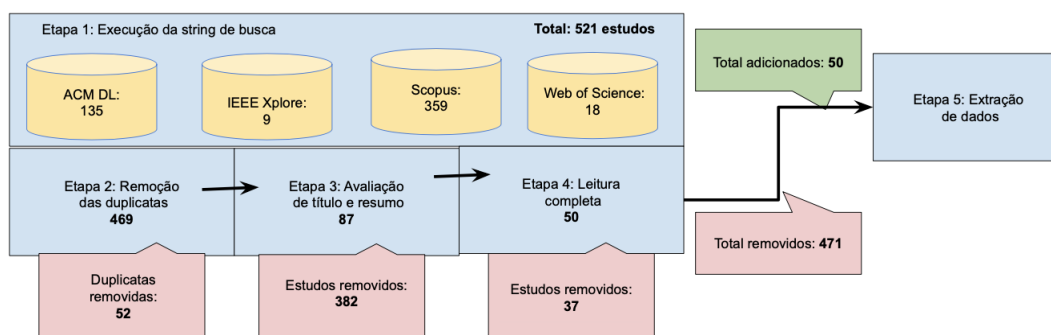


Figura 1. Etapas do processo de seleção dos estudos

“Quais são os vieses cognitivos apresentados pelos estudos?”; SQ02) “Em quais fases do PDS o viés cognitivo ocorre?”; SQ03) “Quais problemas advêm do viés cognitivo no PDS?” e SQ04) “Quais as mitigações para resolver problemas de vieses?”.

A busca eletrônica para responder às perguntas, com os termos em inglês, foi realizada nas bases Scopus², ACM DL³, IEEE Xplore⁴ e Web of Science⁵. Segundo Kitchenham e Charters (2007), a Scopus é a maior fonte para buscas em revisões de literatura. Uma *string* de busca e dois estudos de controle ([Stacy e MacMillan 1995a], [Chattopadhyay *et al.* 2020]) que foram definidos para embasar os termos deste MSL. Após alguns testes, a *string* de busca utilizada foi: (“**cognitive bias**” OR “**cognitive biases**”) AND (“**software development**” OR “**software programming**” OR “**application development**” OR “**application programming**”).

Para selecionar os estudos retornados dessas bases, foram definidos critérios de inclusão (CI): CI1) Estudo descreve em qual/quais fase(s) do PDS ocorrem vieses cognitivos; CI2) Estudo descreve problemas que advêm de vieses cognitivos e CI3) Estudo descreve mitigações para os vieses cognitivos. Critérios de exclusão (CE) também foram definidos: CE1) Estudo não trata de vieses cognitivos no processo de DS; CE2) Estudo é publicação similar (mesmo autor e temática semelhante) a outras encontradas; CE3) Estudo é do tipo “literatura cinza” (i.e., que não teve revisão por pares); CE4) Estudo não foi escrito em Inglês ou Português; CE5) Estudo cujo texto completo não pode ser analisado (indisponível) e CE6) Estudo não é descrição de editorial de periódico ou anais de evento.

Para medir a concordância do grupo de pesquisadores em relação à inclusão/exclusão dos estudos, foi utilizado o coeficiente Cohen’s Kappa. Proposto por Cohen (1960), o processo auxilia a medir a confiabilidade de itens qualitativos, de forma que o grupo entre em consenso com alguns estudos selecionados aleatoriamente e, após obter um alto índice de concordância, os pesquisadores podem ler separadamente. Ao final do processo, 50 estudos foram selecionados, sendo datados desde 1995, conforme mostra a Figura 2. A lista de estudos selecionados foi disponibilizada no Zenodo⁶.

²<https://www.scopus.com/>

³<https://dl.acm.org/>

⁴<https://ieeexplore.ieee.org/>

⁵<https://www.webofscience.com/>

⁶<https://doi.org/10.5281/zenodo.7402593>

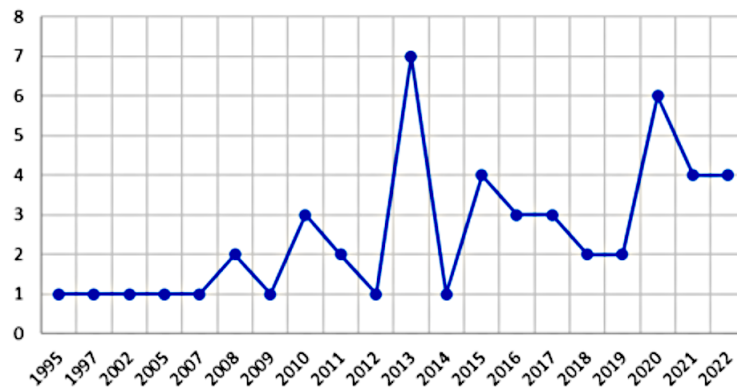


Figura 2. Quantidade de estudos por ano

4. Resultados

Nesta seção, estão descritas as respostas para as questões de pesquisa. Os resultados foram obtidos após a leitura completa dos estudos e análise conjunta dos pesquisadores.

4.1. Vieses Cognitivos Identificados

A fim de responder a SQ01, observou-se que os estudos focam em determinados vieses. Nesse sentido, o material disponibilizado no Zenodo⁶ contém uma tabela com os vieses, os grupos formados, a sua definição e quais estudos trazem quais vieses. O mais citado pelos estudos foi o **viés de confirmação** (a tendência das pessoas procurarem evidências que de alguma forma possam verificar as suas teorias ao invés de refutá-las), com 18 menções. Esse viés é seguido pelo **viés de ancoragem** (o qual acontece quando a exposição de uma informação influencia uma decisão posterior), com 13 menções, e pelo **viés de disponibilidade** (quando se utilizam exemplos que vêm à mente, entendendo que eles são mais representativos do que realmente são), com 8 menções.

4.2. Vieses Cognitivos nas Fases do Processo de Desenvolvimento de Software

Em relação a SQ02, pode-se perceber que a maioria dos estudos está no contexto de levantamento de requisitos. No total, foram contabilizadas sete diferentes fases apontadas pelos estudos, conforme a Tabela 1: tomada de decisão, levantamento de requisitos, design, modelagem, desenvolvimento (implementação/programação), testes e manutenção. Além disso, os estudos de Stacy e MacMillan (1995) e Calikli e Bener (2015) sinalizam que os vieses podem ocorrer em várias fases do PDS, como na implementação, por exemplo, ao se decidir o nome de uma variável.

4.3. Problemas Trazidos por Vieses Cognitivos

No que se refere a SQ03, no estudo de Stacy e MacMillan (1995) apontam que “representatividade” e “viés de disponibilidade” explicam situações de dados incorretos ou imprecisos (como a estimativa de tempo para determinada tarefa). Percebe-se que a maioria dos estudos contextualiza o viés e o problema relacionado. A Tabela 2 aponta os problemas encontrados e os estudos em que foram encontrados, com complementação no Apêndice. Problemas com estimativas incorretas podem ocorrer, como visto nos estudos de Aranda e Easterbrook (2005) e Shepperd *et al.* (2018).

Tabela 1. Fases do PDS citadas

Fase	Estudos
Tomada de decisão	[Rüping 2017, Shepperd <i>et al.</i> 2018]
Levantamento de requisitos	[Browne e Ramesh 2002, Aranda e Easterbrook 2005, Chotisarn e Prompoon 2013a, Shmueli <i>et al.</i> 2015, Stefi e Hess 2015, Bina e Browne 2018, Gren e Svensson 2021]
Design	[Tang 2011, de Graaf <i>et al.</i> 2014, Shmueli <i>et al.</i> 2016]
Modelagem	[Ralph <i>et al.</i> 2016]
Desenvolvimento (implementação /programação)	[Mohan e Jain 2008, Calikli e Bener 2013, Huang <i>et al.</i> 2020, Spadini <i>et al.</i> 2020, van der Linden <i>et al.</i> 2022]
Testes	[Alshraide 2008, Calikli e Bener 2013, Salman <i>et al.</i> 2020]
Manutenção	[Lavazza <i>et al.</i> 2020]

Tabela 2. Problemas relacionados a vieses cognitivos

Problemas	Estudos
Tomada de Decisão	[Stacy e MacMillan 1995b, Browne e Ramesh 2002, Aranda e Easterbrook 2005, Tang 2011, de Graaf <i>et al.</i> 2014, Ralph <i>et al.</i> 2016, Shepperd <i>et al.</i> 2018, Chattopadhyay <i>et al.</i> 2020, Lavazza <i>et al.</i> 2020, Borowa <i>et al.</i> 2021, Wyrich <i>et al.</i> 2021, van der Linden <i>et al.</i> 2022]
Problemas Técnicos	[Stacy e MacMillan 1995b, Siau <i>et al.</i> 1997, Browne e Ramesh 2002, Ralph 2013, Calikli e Bener 2013, Chotisarn e Prompoon 2013a, Brenner 2019, Salman <i>et al.</i> 2020]
Subestimação de Tempo	[Dennis <i>et al.</i> 2012, Shmueli <i>et al.</i> 2015]
Aversão à Perda	[Brenner 2019]
Retrabalho	[Mohan e Jain 2008]

4.4. Mitigações para Problemas de Vieses

Acerca da SQ04, as mitigações encontradas foram “avaliações de perfil de desenvolvedores”, “boas práticas de desenvolvimento”, “consenso de grupo”, “cuidado com requisitos” e “disponibilização de informações”. Em relação à disponibilidade de informação, ter uma base de conhecimento e treinamento foram as mitigações encontradas. As boas práticas de desenvolvimento encontradas envolvem testes, *debugging*, desenvolvimento e cuidado com o tempo. Os cuidados com os requisitos envolvem a sua rastreabilidade e não os deixarem ficar obsoletos. Consenso de grupo envolve ter um “advogado do diabo” e *planning poker*.

4.5. Como os Vieses Cognitivos Afetam o Desenvolvimento de Software?

São diversos os vieses cognitivos e cada um deles afeta o desenvolvimento de alguma forma. Isso não ocorre somente na fase de implementação/programação em si, mas também envolve as decisões previamente tomadas, cultura e regras das organizações, que afetam em conjunto e atuam colaborativamente com os vieses cognitivos. Fink e Pinchovski (2020) apontam que existem informações tendenciosas, ao passo que Stacy e MacMillan (1995) afirmam que “as inferências intuitivas e os julgamentos de probabilidade das pessoas não obedecem estritamente às leis da lógica ou da matemática”. Nesse sentido, os vieses acontecem sem os envolvidos perceberem e afetam negativamente (p.ex.: exploração inadequada, atenção equivocada, perda de contexto etc.) o PDS.

5. Discussão

Os resultados do mapeamento demonstraram que os tipos de vieses mais citados nos estudos foram o viés de confirmação, de ancoragem e de disponibilidade. Essa ordem de vieses emergiu dos próprios estudos, não existindo uma listagem prévia com esta definição. Da mesma forma, as fases do PDS apresentadas foram definidas através da leitura dos estudos, conforme foram citados. Combinando esses vieses com as fases do PDS, o viés de confirmação aparece nas fases de design, testes, levantamento de requisitos e de desenvolvimento. Seus problemas são relacionados ao retrabalho (fase de desenvolvimento), problemas técnicos (fases de testes e levantamento de requisitos) e tomada de decisão (fase de design). O viés de ancoragem foi encontrado nas fases de tomada de decisão, levantamento de requisitos, design, testes e desenvolvimento, com problemas relacionados a retrabalho (fase de desenvolvimento), problemas técnicos (fase de testes) e tomada de decisão (fases de design, levantamento de requisitos e de tomada de decisão). O viés de disponibilidade foi encontrado nas fases de manutenção, design, levantamento de requisitos e desenvolvimento, relacionados aos problemas de retrabalho, problemas técnicos e tomada de decisão. Desta forma, a partir dos resultados obtidos, foi possível responder à questão investigada neste estudo.

Outro aspecto que pode ser discutido é a **influência dos vieses em outras circunstâncias do PDS**, como no ATD⁷ [Borowa *et al.* 2021]. Assim, o viés cognitivo nem sempre está sozinho na análise de problemas e mitigações que ocorrem ao longo do desenvolvimento de um projeto de software. Ralph (2011) identifica 12 conjuntos de interconexões de vieses, separa-os em temas e, inclusive, nomeia-os como *biasplexes*, termo revisitado em seu outro estudo [Ralph 2013]. *Biasplexes* levam a antipadrões comportamentais e que, portanto, afetam negativamente o desenvolvimento de um projeto.

Apesar de geralmente estarem associados a efeitos negativos, não é apenas dessa maneira que os vieses estão presentes no PDS. Spadini *et al.* (2020) mostraram que os comentários em *code reviews* levam a vieses, mas ocorrem positivamente, como lembranças para outros problemas que podem existir naquele código que está sendo visto.

No intuito de mitigar os problemas advindos dos vieses cognitivos, diversos estudos buscaram soluções gerais ou mais específicas. Por exemplo, Calikli *et al.* (2010) realizaram uma análise empírica para medir/quantificar o nível do viés de confirmação dos indivíduos. Algumas soluções foram propostas para reduzir os vieses, tal como a rastreabilidade de requisitos [Mohan e Jain 2008] entre as fases de desenvolvimento.

Por fim, não foram encontradas “soluções finais” ou “neutralizadoras” de vieses, apesar da gama de estudos na área de psicologia, sociologia e engenharia de software. O ser humano não é neutro e, como as ferramentas computacionais repetem ou automatizam ações humanas, elas estão de algum modo enviesadas e isso também se aplica ao PDS, independente da fase.

6. Ameaças à Validade

Uma ameaça a validade é relacionada ao termo de busca que pode não conter todas as palavras-chave e sinônimos importantes, também não foram adicionados os termos em português, o que ocasionaria alteração nos resultados e perda de estudos que poderiam

⁷*Architectural technical debt* - Dívida técnica de arquitetura

ser significativos. Na tentativa de atenuar esse problema, a *string* de busca foi avaliada usando um conjunto de estudos previamente selecionados.

A reexecução deste MSL por outros pesquisadores pode trazer outras informações conclusivas por meio da seleção e classificação de outros estudos. Para mitigar esse risco, os pesquisadores utilizaram o Cohen's Kappa na fase de seleção dos estudos. Além disso, estudos previamente analisados foram aleatoriamente reanalisados por ao menos 2 dos 3 pesquisadores e os resultados foram discutidos em conjunto, com o intuito de verificar se o alinhamento obtido no Cohen's Kappa se manteve na fase de extração de dados.

7. Conclusão

Este estudo apresentou os resultados de um MSL sobre os vieses cognitivos no PDS, buscando compreender quais são, as fases em que ocorrem, os problemas decorrentes e as possíveis mitigações para tais vieses. Foram selecionados 50 estudos durante a condução deste MSL. Os vieses cognitivos representam um exemplo de questão comportamental que está sendo abordada ao longo dos anos pela comunidade acadêmica. Como contribuição para a indústria, os gestores podem explorar os vieses e, dessa forma, prevenir ou mitigá-los no PDS. O estudo se mostra relevante na academia para pesquisas relacionadas a PDS e por abranger o aspecto humano.

A partir dos resultados encontrados, possíveis trabalhos futuros consistem na elaboração de um guia de mitigações para cada fase do PDS; na construção de um mecanismo de detecção semi-automática dos vieses neste contexto; e na condução de uma pesquisa de opinião com objetivo de verificar a percepção dos desenvolvedores acerca dos vieses percebidos no seu dia a dia de trabalho.

Agradecimentos

Este estudo foi parcialmente financiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - Código 001, FAPERJ (Proc. 211.583/2019 & Proc. 210.688/2019), UNIRIO (DPq/PPQ 2022) e Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq (Proc. 305.436/2021-0).

Referências

- Alshraide, M. (2008). A complete automation of unit testing for javascript programs. *Journal of Computer Science*, 4(12):1012–1019.
- Aranda, J. and Easterbrook, S. (2005). Anchoring and adjustment in software estimation. In *Proceedings of the 10th European software engineering conference held jointly with 13th ACM SIGSOFT international symposium on Foundations of software engineering - ESEC/FSE-13*. ACM Press.
- Bina, S. and Browne, G. J. (2018). Information avoidance in requirements determination for systems development. In *24th Americas Conference on Information Systems 2018: Digital Disruption, AMCIS 2018*. Association for Information Systems.
- Borowa, K., Zalewski, A., and Kijas, S. (2021). The influence of cognitive biases on architectural technical debt. In *2021 IEEE 18th International Conference on Software Architecture (ICSA)*. IEEE.

- Brenner, R. (2019). Balancing resources and load: Eleven nontechnical phenomena that contribute to formation or persistence of technical debt. In *2019 IEEE/ACM International Conference on Technical Debt (TechDebt)*. IEEE.
- Browne, G. J. and Parsons, J. (2012). More enduring questions in cognitive is research. *Journal of the Association for Information Systems*, 13(12):2.
- Browne, G. J. and Ramesh, V. (2002). Improving information requirements determination: a cognitive perspective. *Information & Management*, 39(8):625–645.
- Calikli, G. and Bener, A. (2013). An algorithmic approach to missing data problem in modeling human aspects in software development. In *Proceedings of the 9th International Conference on Predictive Models in Software Engineering*, pages 1–10.
- Calikli, G. and Bener, A. (2015). Empirical analysis of factors affecting confirmation bias levels of software engineers. *Software Quality Journal*, 23(4):695–722.
- Calikli, G., Bener, A., and Arslan, B. (2010). An analysis of the effects of company culture, education and experience on confirmation bias levels of software developers and testers. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - ICSE '10*. ACM Press.
- Chattopadhyay, S., Nelson, N., Au, A., Morales, N., Sanchez, C., Pandita, R., and Sarma, A. (2020). A tale from the trenches: Cognitive biases and software development. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*. ACM.
- Chotisarn, N. and Prompoon, N. (2013a). Forecasting software damage rate from cognitive bias in software requirements gathering and specification process. In *2013 IEEE Third International Conference on Information Science and Technology (ICIST)*, pages 951–956. IEEE.
- Chotisarn, N. and Prompoon, N. (2013b). Predicting software damage rate from cognitive bias in software design process.
- Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.
- Davern, M., Shaft, T., and Te'eni, D. (2012). Cognition matters: Enduring questions in cognitive is research. *Journal of the Association for Information Systems*, 13(4):1.
- de Graaf, K. A., Liang, P., Tang, A., and van Vliet, H. (2014). The impact of prior knowledge on searching in software documentation. In *Proceedings of the 2014 ACM symposium on Document engineering - DocEng '14*. ACM Press.
- Dennis, A., Robert, L., Curtis, A., Kowalczyk, S., and Hasty, B. (2012). Research note—trust is in the eye of the beholder: A vignette study of postevent behavioral controls' effects on individual trust in virtual teams. *Information Systems Research*, 23:546–558.
- Fagerholm, F., Felderer, M., Fucci, D., Unterkalmsteiner, M., Marculescu, B., Martini, M., Tengberg, L. G. W., Feldt, R., Lehtelä, B., Nagyvárad, B., and Khattak, J. (2022). Cognition in software engineering: A taxonomy and survey of a half-century of research. *ACM Computing Surveys*, 54(11s):1–36.

- Fink, L. and Pinchovski, B. (2020). It is about time: Bias and its mitigation in time-saving decisions in software development projects. *International Journal of Project Management*, 38(2):99–111.
- Gigerenzer, G. and Gaissmaier, W. (2011). Heuristic decision making. *Annual review of psychology*, 62(1):451–482.
- Gren, L. and Svensson, R. B. (2021). Is it possible to disregard obsolete requirements? a family of experiments in software effort estimation. *Requirements Engineering*, 26(3):459–480.
- Huang, Y., Leach, K., Sharafi, Z., McKay, N., Santander, T., and Weimer, W. (2020). Biases and differences in code review using medical imaging and eye-tracking: genders, humans, and machines. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ACM.
- Kahneman, D. and Tversky, A. (1972). Subjective probability: A judgment of representativeness. *Cognitive psychology*, 3(3):430–454.
- Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. 2.
- Lavazza, L., Liu, G., and Meli, R. (2020). Productivity of software enhancement projects: an empirical study. In *IWSM-Mensura*.
- Mohan, K. and Jain, R. (2008). Using traceability to mitigate cognitive biases in software development. *Communications of the ACM*, 51(9):110–114.
- Mohanani, R. (2016). Implications of requirements engineering on software design: A cognitive insight. In *2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C)*, pages 835–838. IEEE.
- Mohanani, R., Salman, I., Turhan, B., Rodríguez, P., and Ralph, P. (2018). Cognitive biases in software engineering: a systematic mapping study. *IEEE Transactions on Software Engineering*, 46(12):1318–1339.
- Ralph, P. (2011). Toward a theory of debiasing software development. In *Research in Systems Analysis and Design: Models and Methods*, pages 92–105. Springer Berlin Heidelberg.
- Ralph, P. (2013). Possible core theories for software engineering. In *2013 2nd SEMAT Workshop on a General Theory of Software Engineering (GTSE)*. IEEE.
- Ralph, P., Chiasson, M., and Kelley, H. (2016). Social theory for software engineering research. In *Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering*. ACM.
- Rüping, A. (2017). Decisions among teams. In *Proceedings of the VikingPLoP 2017 Conference on Pattern Languages of Program - VikingPLoP*. ACM Press.
- Salman, I., Rodriguez, P., Turhan, B., Tosun, A., and Gureller, A. (2020). What leads to a confirmatory or disconfirmatory behaviour of software testers? *IEEE Transactions on Software Engineering*.

- Shepperd, M., Mair, C., and Jørgensen, M. (2018). An experimental evaluation of a de-biasing intervention for professional software developers. In *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*. ACM.
- Shmueli, O., Pliskin, N., and Fink, L. (2015). A position paper proposing behavioral solutions to challenges in software development projects. In *Lecture Notes in Business Information Processing*, pages 94–99. Springer International Publishing.
- Shmueli, O., Pliskin, N., and Fink, L. (2016). Can the outside-view approach improve planning decisions in software development projects? *Information Systems Journal*, 26(4):395–418.
- Siau, K., Wand, Y., and Benbasat, I. (1997). The relative importance of structural constraints and surface semantics in information modeling. *Information Systems*, 22(2-3):155–170.
- Silverman, B. G. (1990). Critiquing human judgment using knowledge-acquisition systems. *AI Magazine*, 11(3):60–60.
- Sommerville, I. and Rodden, T. (1996). Human, social and organisational influences on the software process. *Software Process*, 4:89–100.
- Spadini, D., Çalikli, G., and Bacchelli, A. (2020). Primers or reminders the effects of existing review comments on code review. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*. ACM.
- Stacy, W. and MacMillan, J. (1995a). Cognitive bias in software engineering. *Communications of the ACM*, 38(6):57–63.
- Stacy, W. and MacMillan, J. (1995b). Cognitive bias in software engineering. *Communications of the ACM*, 38(6):57–63.
- Stefi, A. and Hess, T. (2015). To develop or to reuse? two perspectives on external reuse in software projects. In *Lecture Notes in Business Information Processing*, pages 192–206. Springer International Publishing.
- Tang, A. (2011). Software designers, are you biased? In *Proceeding of the 6th international workshop on SHaring and Reusing architectural Knowledge - SHARK '11*. ACM Press.
- van der Linden, D., Williams, E., Hallett, J., and Rashid, A. (2022). The impact of surface features on choice of (in)secure answers by stackoverflow readers. *IEEE Transactions on Software Engineering*, 48(2):364–376.
- Wyrich, M., Preikschat, A., Graziotin, D., and Wagner, S. (2021). The mind is a powerful place: How showing code comprehensibility metrics influences code understanding. In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE.

A. Apêndice

A lista de estudos selecionados e a tabela completa de vieses cognitivos estão disponíveis em <https://doi.org/10.5281/zenodo.7402593>.