

Theoretical Saturation in Qualitative Software Engineering Studies

Antonio Pedro Santos Alves^{1,*†}

¹Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Rua Marquês de São Vicente, 225 - Rio de Janeiro, Brazil

Abstract

[Background] A central concern in qualitative research is determining when the collected and analyzed data provide sufficient support for the study's claims. In Grounded Theory, this concern is commonly addressed through *theoretical saturation*, defined as the point at which the components of a theory are sufficiently developed, and additional data no longer trigger meaningful revisions. Although this concept was proposed in sociology research, it has been widely adopted in qualitative software engineering (SE) studies, often under broader, or related, definitions, such as *data saturation*, *code saturation*, or simply *saturation*. These constructs are related but not equivalent, since they belong to different analytical approaches and require different evidence. Despite its relevance, qualitative SE studies still lack clear guidance on how saturation should be assessed, justified, and reported, which may lead to claims that cannot be assessed, or to reliance on sample size alone.

[Aims] This research aims to strengthen the empirical evidence for assessing theoretical saturation, to clarify which saturation constructs fit which qualitative SE study designs, and to provide practical guidance for SE researchers conducting, reporting, and reviewing qualitative studies.

[Method] The research will be conducted in three main phases: (i) characterizing the state of practice through a literature review of recent qualitative SE studies that claim to have reached saturation; (ii) investigating how SE researchers perceive, operationalize, and report saturation through a mixed-method study involving a survey and semi-structured interviews; and (iii) designing and evaluating a guideline to support the assessment and reporting of theoretical saturation in qualitative SE studies.

[Expected Outcomes] The main expected outcome is the *Theoretical Saturation Guideline* (TSG), an artifact intended to support planning, assessment, reporting, and review of saturation claims in qualitative software engineering research, improving transparency without reducing saturation to a fixed numerical threshold.

Keywords

Theoretical Saturation, Qualitative Studies, Empirical Software Engineering

1. Introduction

Software Engineering (SE) is a *socio-technical* discipline [1], neither exclusively social nor purely technical. It can be understood as an intricate mix of social and technical aspects, where studying one without due consideration of the other results in an incomplete investigation and understanding [2]. In this sense, qualitative methods have helped in understanding complex SE phenomena that go beyond the use of simple metrics or log analysis, thereby strengthening empirical software engineering [3].

Among the qualitative methods employed by the SE community, Grounded Theory (GT) has been one of the most widely used [4]. Originally, GT was proposed in the sociology realm in 1967 [5], with the goal of providing a systematic way to generate new theories rather than to validate existing ones. The application of a method conceived to investigate *social* phenomena poses inherent challenges for investigating *socio-technical* ones, and a decade ago, few SE studies had successfully applied GT [4]. However, since then, the methodological support available to the community has grown considerably, including *Socio-Technical Grounded Theory* (STGT) [2], its practical guide [6], and the ACM SIGSOFT Empirical Standards [7].

The effort towards mature empirical SE practices makes it possible to examine specific methodological decisions in greater depth, such as in GT (here, we use a broader definition that includes the three main variations of it [5, 8, 9]) or STGT studies. In such studies, it is common to employ isolated usage of

IDOESE'26: International Doctoral Symposium on Empirical Software Engineering, October 07, 2026, München, Germany

✉ apsalves@inf.puc-rio.br (A. P. S. Alves)

ORCID  0000-0002-3415-6201 (A. P. S. Alves)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

certain techniques. Some of them are part of the qualitative researcher's toolbox [4, 2], especially the ones related to qualitative data analysis, such as *coding*, *constant comparison*, *memoing*, and *theoretical saturation*.

Among such techniques, one of the most important for qualitative studies is the *theoretical saturation*. In the context of GT, it is the point at which a theory's components are well supported, and new data no longer triggers revisions of the theory [10, 2]. Outside theory development, SE researchers commonly invoke related constructs: *code saturation*, concerning whether new codes continue to appear, and *meaning saturation*, concerning whether the dimensions and nuances of those codes are sufficiently understood [11]. These constructs are related but not equivalent. They may be reached at different points in the same dataset, they belong to different analytical approaches, and they require different evidence.

For example, consider an interview study on code review in which *time pressure* is repeatedly identified from the first interviews onward, until new interviews no longer yield new codes. If the goal is to describe the concerns practitioners raise, this is code saturation, and it may well be sufficient for the claims the study makes. However, if the goal is to explain how time pressure shapes review behavior, the same evidence supports considerably less, since it does not show which conditions distinguish junior from senior reviewers, nor how time pressure relates to other categories, such as reviewer workload. In both cases, the evidence is the same, and what changes is the claim it is asked to support. In this sense, a saturation claim can only be assessed when we know which construct it refers to and which evidence supports it.

However, saturation lacks clear guidance on how to assess and discuss it in a qualitative SE study. A recurrent pattern is a short paragraph stating that saturation was reached, supported by a sense that the sample size was "big enough" [12, 13]. This difficulty is not exclusive to SE. The conceptualization and operationalization of saturation remain contested across qualitative research [14].

1.1. Problem Statement

Advances in qualitative methods in SE studies are relatively new compared with those in quantitative methods [15]. Despite the growing interest and wide use of some approaches, some qualitative techniques still lack evidence on how they should be assessed. *Theoretical saturation* is a concept originally imported from sociology [10] and can be understood as a point at which new data no longer trigger new insights. Despite its importance to the soundness of research, *theoretical saturation* lacks empirical evidence for assessing it, and the consequences affect various aspects.

For instance, a claim that does not state which construct it refers to cannot be assessed at all. The reader cannot tell whether the study is reporting that new codes stopped appearing, that the meaning of those codes was fully developed, or that a theory stabilized. An unqualified claim may assert more than the analysis supports, leaving unexplored variation invisible and the decision to stop collecting data difficult to audit.

Therefore, guidance is needed to support SE researchers in justifying both whether saturation applies to their study and whether their evidence supports the claim they make, as well as to help reviewers provide more constructive reviews.

1.2. Research Goal & Objectives

The goal of my research is to bring together evidence on how saturation is currently used in SE, understand how SE researchers interpret and struggle with it, gather input on how saturation is assessed in other disciplines, and produce a novel artifact to support assessing theoretical saturation in software engineering qualitative studies, called ***Theoretical Saturation Guideline*** (TSG).

It is worth noting that *theoretical saturation* is the main concept, while also considering related saturation claims commonly reported in qualitative SE studies, such as *data saturation*, *code saturation*, or *saturation* itself. This broader scope is necessary because the SE community often uses these terms without explicit distinction, and identifying which construct fits which analytical approach and data

source in SE is part of my research. In this sense, four main research questions were developed for my thesis, which are:

RQ1. What is the current state of practice regarding theoretical saturation in qualitative software engineering studies? To answer this question, the idea is to systematically analyze primary qualitative SE studies to understand how theoretical saturation is being justified and assessed within SE research.

RQ2. How is theoretical saturation perceived among software engineering researchers? To answer this question, the aim is to conduct a mixed-method study with SE researchers, consisting of a descriptive survey, informed by the literature review, followed by explanatory semi-structured interviews.

RQ3. How can we support software engineering researchers to properly consider theoretical saturation in their qualitative studies? Based on the outcomes of the studies that addressed the previous RQs, and from methodological guidelines and texts from other disciplines that discuss saturation, the Theoretical Saturation Guideline (TSG) artifact will be proposed.

RQ4. What is the perceived usefulness of the Theoretical Saturation Guideline (TSG) in qualitative software engineering research? Finally, the goal is to empirically assess the TSG through a confirmatory survey, focus groups with empirical software engineering experts, retrospective application to published studies, and to evolve it according to identified opportunities for improvement so that it can be released to the community.

1.3. Current Status and Intended Schedule

To understand the state of practice of theoretical saturation in SE qualitative studies, I am conducting a critical review of qualitative studies published between 2021 and 2025, better described in Section 3.1. From this analysis, Table 1 summarizes the procedures that will take place in my Ph.D research agenda.

Table 1

Intended research schedule.

Period	Activity
Summer 2026 & Fall 2026.	Complete the ongoing critical review and the methodological synthesis (RQ1).
Winter 2026 & Spring 2027.	Conduct the descriptive survey and the explanatory interviews (RQ2).
Spring 2027 & Summer 2027.	Synthesize the evidence and design the TSG (RQ3).
Fall 2027 & Winter 2028.	Evaluate and refine the TSG (RQ4).
Winter 2028 & Spring 2028.	Consolidate the thesis and prepare its presentation.

2. Related Work

The concept of *theoretical saturation* across qualitative studies is essentially the same as proposed in [10]. The underlying motivation for it is straightforward: when should we stop analyzing our data? If creating a theory, regardless of the main variations of Grounded Theory (GT) (*classic* [5], *Straussian* [8], or *constructivist* [9]), the definition of saturation is the point where new data no longer triggers revisions of the theory's components [10]. Beyond theory development, the term is used more broadly and with different meanings. Saunders et al. [14] distinguish conceptualizations that emphasize theoretical development, the emergence of themes, and informational redundancy, and argue that the conceptualization adopted must fit the research question and the analytical framework. Hennink et al. [11] show empirically that identifying the range of codes and developing a sufficiently nuanced

Table 2

Summary of commonly used strategies for assessing saturation, adapted from Hennink and Kaiser [17].

Strategy	Summary	Main idea
Code Frequency Counts	Tracks the number of new codes identified in each successive interview, focus group, or batch of transcripts. Saturation is considered closer when few or no new codes continue to appear. Some studies randomize the order of data to reduce sequential bias.	Saturation as diminishing emergence of new codes.
Stopping Criterion	Defines an explicit rule for stopping data collection or analysis. For example, saturation may be claimed after a predefined number of consecutive interviews or focus groups produce no new codes, or when the proportion of new codes falls below a threshold.	Saturation as a predefined decision rule.
High-Order Groupings*	Focuses not only on individual codes but also on broader analytical groupings, such as categories, themes, or meta-themes. Saturation is assessed when these higher-level groupings are sufficiently captured.	Saturation as coverage of broader conceptual categories.
Code Meaning*	Moves beyond counting codes and focuses on whether the meaning, dimensions, and nuances of each code are fully understood. Different codes may reach saturation at different points in the dataset.	Saturation as the depth of the researcher's understanding.
Comparative Method	Extends code frequency counting by comparing predefined batches of data, such as quartiles. New codes are listed for each batch and compared with previous batches until few or no new codes are identified.	Saturation as comparison across data batches.

Note. The order of each strategy in the table considered the prevalence of each strategy presented in the literature review [17]. The asterisk indicates strategies with the same prevalence.

understanding of their meaning are reached at different points in the same dataset, which cautions against treating the absence of new codes as evidence of conceptual depth.

It is worth mentioning that saturation and sampling address related but distinct concerns. Saturation is often invoked to justify, mostly, the adequacy of a sample size [16]. A saturation assessment, however, does not establish statistical representativeness and is not a substitute for generalizability. What saturation can speak to is the stability of the insights obtained from the data at hand. Moreover, saturation cannot be determined in advance [13], since planning relies on the researcher's expertise, and the adequacy of the evidence must be reconsidered as the study proceeds [10].

Despite these difficulties, some interesting findings have emerged in other research areas, such as in qualitative health research [17]. Because the concept of saturation is interdisciplinary, the findings may help clarify the *status quo* of theoretical saturation in qualitative SE research. For instance, Table 2 summarizes the most commonly used methods for assessing saturation reported in health research [17]. Building on prior work and current practices in qualitative SE research, we aim to complement existing guidance with empirical evidence that explicitly distinguishes among saturation constructs, study contexts, and evidence requirements. The intended consequence is a more transparent approach for both authors, who spend real resources collecting data, and reviewers, who can then assess the reported reasoning instead of relying on the sample size fallacy (e.g., "sample size too small") [13].

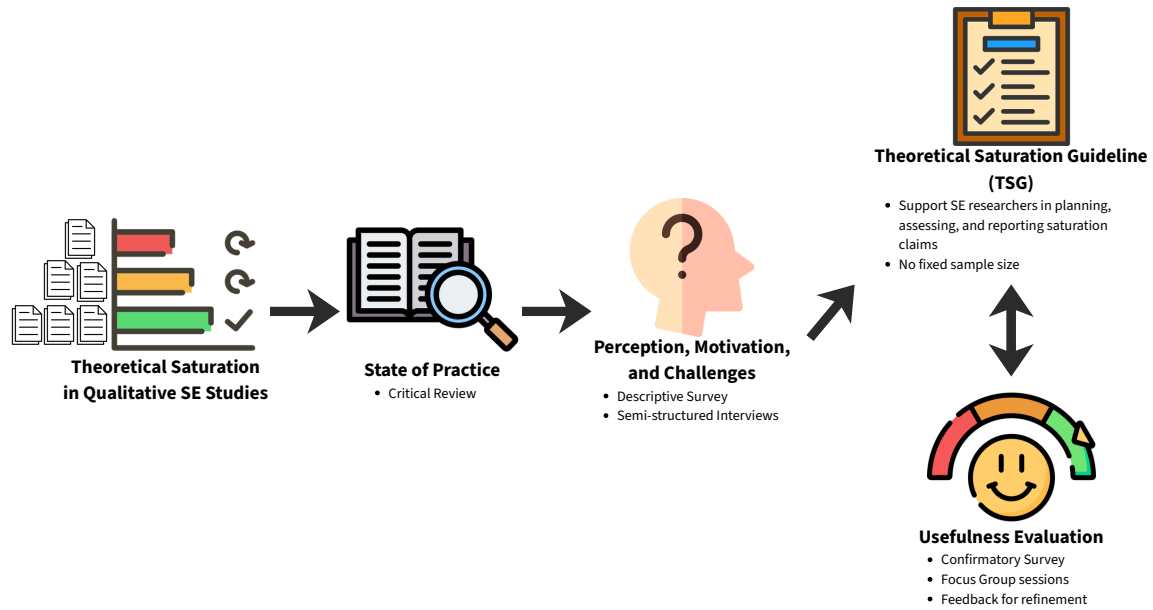


Figure 1: Overview of the proposed research methodology, from characterizing current saturation practices in qualitative SE studies to designing, evaluating, and refining the Theoretical Saturation Guideline (TSG).

3. Research Method

The proposed research method consists of three main phases, which include (i) the state of practice, (ii) perception of theoretical saturation, and (iii) the design and evaluation of a guideline for assessing theoretical saturation in qualitative software engineering studies. Figure 1 summarizes these phases.

3.1. The State of Practice of Theoretical Saturation in Qualitative Software Engineering Studies

The first phase of this research aims to characterize how theoretical saturation has been reported, justified, and operationalized in recent qualitative SE studies. This phase is addressed by conducting a critical review of qualitative studies published between 2021 and 2025 in high-ranked SE venues from the top-5 Google Scholar Journals (TSE, TOSEM, JSS, EMSE, IST), and the A*/A CORE conferences focused on SE or empirical software engineering (ICSE, FSE, ASE, EASE, ESEM). This purposive venue set combines established general SE and empirical SE outlets, similar to previous types of studies [4, 18]. The set does not include every A-ranked venue that publishes qualitative SE research, which is a delimitation of the review. Moreover, venue ranking is not treated as evidence of methodological quality.

The review will focus on studies that explicitly mention *theoretical saturation*, *data saturation*, *code saturation*, *meaning saturation*, or simply *saturation*. For each selected study, information on the research method used, the type of qualitative data collected, the number and profile of participants or artifacts, the point at which saturation was assessed, and the evidence used to claim saturation will be extracted. The analysis will be guided by concepts from prior, interdisciplinary studies, especially the strategies for assessing saturation presented in [17].

As a result of this phase, common practices for assessing theoretical saturation in SE qualitative studies will be identified, shedding light on which constructs are claimed, what evidence is offered to support them, and whether the claimed constructs fit the analytical approach and the reported data source.

3.2. Theoretical Saturation Perception among Software Engineering Researchers

The second phase complements the critical review by investigating how software engineering researchers perceive theoretical saturation, how they determine when saturation has been reached, and the challenges they face when reporting or reviewing saturation claims. A convenience sampling strategy, filtering for participants who have conducted, supervised, or reviewed at least one qualitative SE study, will be adopted and expected to target an initial set of 70 - 90 participants.

The survey will collect information about participants' experience with qualitative methods, their definitions, the criteria they use to assess saturation, and perceived difficulties. It will also include questions in which participants evaluate different saturation claims, originating from the previous step, ranging from weak to stronger justifications. Since the sampling distribution of the statistics of interest cannot be assumed, and the sample is not probabilistic, bootstrap confidence intervals will be reported alongside the descriptive statistics, similar as done in Wagner et al. [19], conveying the precision of the estimates within the obtained sample rather than prevalence in the SE community.

The interviews will take place with participants who consent to be followed up by the study. Their selection will consider different types of experience and perceptions of saturation, based on their survey responses. The goal is to provide a deeper understanding of the motivations, uncertainties, and contextual factors behind researchers' decisions. What methodological rigor is used when deciding to stop data collection? How do researchers balance publication expectations or reviewer comments? They may also reveal how researchers interpret saturation differently across study types, such as interview studies, open-ended survey analysis, or repository mining.

Given the nature of survey design, some threats might be listed, according to survey validity types [20], such as: (i) face and content validity; (ii) criterion validity; (iii) construct validity; and (iv) reliability.

3.3. Guideline Design and Evaluation

The third and final phase focuses on designing and evaluating the *Theoretical Saturation Guideline* (TSG), the main artifact of this research. The guideline will be grounded in the findings from the critical review and the perception study. The purpose is to support software engineering researchers in planning, assessing, reporting, and reviewing saturation in qualitative studies.

The design will start from the decisions and difficulties identified in the previous phases. Each candidate item will be traceable to the evidence that motivated it, whether a reporting practice observed during the review, a difficulty reported by participants, or evidence from other saturation literature in the discipline. Items that cannot be expressed as a prompt with identifiable supporting evidence will be discarded, and where the evidence points in different directions, the divergence will be recorded as conditional guidance or as an open issue, rather than resolved by majority.

The initial version of the TSG will be structured around practical decisions that researchers need to make during a qualitative study. These decisions may include (i) defining which type of saturation is relevant to the study, (ii) planning how saturation will be assessed, (iii) documenting progress, and (iv) reporting the evidence that supports the saturation claim. The final form of the instrument, whether a checklist, a set of questions, or a matrix relating decisions to study types, will be defined from the findings of the previous phases. The goal is not to provide a fixed sample size, but to emphasize transparency, traceability, and alignment between the research goal, data collection strategy, and analysis procedure.

To evaluate the usefulness of the TSG, a confirmatory survey and focus group sessions with experienced qualitative software engineering researchers will be conducted. The survey will focus on assessing and reviewing the guideline items. The population target will include those who answered the previous survey, expanded through purposive sampling and participant referrals, while filtering for those involved in more than one qualitative SE study in previous years. The focus group sessions will follow the guidelines for focus groups in SE [21] and will take place in two groups of four to six participants each. The participants are expected to be experienced qualitative software engineering researchers who were not involved in the design of the guideline items. The focus group sessions will

involve participants using the guideline to assess sample excerpts of saturation reporting and discussing whether the guideline is understandable, useful, complete, and feasible to apply. Feedback from the focus groups will be used to refine the artifact and identify open issues for future validation.

4. Expected Outcomes

The main expected outcome of this research is the *Theoretical Saturation Guideline*, an artifact to support the assessment and reporting of theoretical saturation in qualitative software engineering studies. The scientific contribution is an empirically grounded account of how saturation, a construct imported from sociology, applies to qualitative software engineering research: which of its forms fit which analytical approaches and data sources, what evidence each requires, and under which conditions its use is appropriate.

To achieve this overall objective, the research will first provide an empirical characterization of how theoretical saturation has been reported in recent qualitative SE studies. Second, it will investigate how software engineering researchers perceive and operationalize saturation, including the challenges they face when applying a concept originally developed in broader qualitative research. Finally, it will evaluate the perceived usefulness of the proposed guideline based on feedback from software engineering researchers.

This research aims to strengthen the rigor and transparency of empirical software engineering by clarifying how to assess and report theoretical saturation in qualitative studies. However, it does not intend to establish a rigid rule for all qualitative studies, nor to reduce saturation to a numerical threshold. Instead, it aims to provide structured support for researchers to justify their decisions in ways that align with their research design, qualitative methods, and goals.

5. Declaration on Generative AI

During the preparation of this work, the author used *Grammarly* in order to check grammar and spelling. During the camera-ready phase, *GPT* (OpenAI) was used to rephrase and condense the clarifications added to meet the page limit. After using this tool, the author reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

- [1] M.-A. Storey, N. A. Ernst, C. Williams, E. Kalliamvakou, The who, what, how of software engineering research: a socio-technical framework, *Empirical Softw. Engg.* 25 (2020) 4097–4129. URL: <https://doi.org/10.1007/s10664-020-09858-z>. doi:10.1007/s10664-020-09858-z.
- [2] R. Hoda, Socio-technical grounded theory for software engineering, *IEEE Transactions on Software Engineering* 48 (2022) 3808–3832. doi:10.1109/TSE.2021.3106280.
- [3] C. Seaman, Qualitative methods in empirical studies of software engineering, *IEEE Transactions on Software Engineering* 25 (1999) 557–572. doi:10.1109/32.799955.
- [4] K.-J. Stol, P. Ralph, B. Fitzgerald, Grounded theory in software engineering research: a critical review and guidelines, in: *Proceedings of the 38th International Conference on Software Engineering, ICSE '16*, Association for Computing Machinery, New York, NY, USA, 2016, p. 120–131. URL: <https://doi.org/10.1145/2884781.2884833>. doi:10.1145/2884781.2884833.
- [5] B. G. Glaser, A. L. Strauss, *The Discovery of Grounded Theory: Strategies for Qualitative Research*, Aldine, Chicago, 1967.
- [6] R. Hoda, *Qualitative Research with Socio-Technical Grounded Theory: A Practical Guide to Qualitative Data Analysis and Theory Development in the Digital World*, 1 ed., Springer, Cham, Switzerland, 2024. URL: <https://doi.org/10.1007/978-3-031-60533-8>. doi:10.1007/978-3-031-60533-8.
- [7] P. Ralph, N. bin Ali, S. Baltes, D. Bianculli, J. Diaz, Y. Dittrich, N. Ernst, M. Felderer, R. Feldt, A. Filieri, B. B. N. de França, C. A. Furia, G. Gay, N. Gold, D. Graziotin, P. He, R. Hoda, N. Juristo,

- B. Kitchenham, V. Lenarduzzi, J. Martínez, J. Melegati, D. Mendez, T. Menzies, J. Moller, D. Pfahl, R. Robbes, D. Russo, N. Saarimäki, F. Sarro, D. Taibi, J. Siegmund, D. Spinellis, M. Staron, K. Stol, M.-A. Storey, D. Taibi, D. Tamburri, M. Torchiano, C. Treude, B. Turhan, X. Wang, S. Vegas, Empirical standards for software engineering research, 2021. URL: <https://arxiv.org/abs/2010.03525>. arXiv:2010.03525.
- [8] A. Strauss, J. Corbin, Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory, 2nd ed., SAGE Publications, Thousand Oaks, CA, 1998.
 - [9] K. Charmaz, Constructing grounded theory: A practical guide through qualitative analysis, Sage Publications, London; Thousand Oaks, Calif., 2006.
 - [10] B. Glaser, Theoretical sensitivity, Advances in the Methodology of Grounded Theory (1978). URL: <https://cir.nii.ac.jp/crid/1573387450646190464>.
 - [11] M. M. Hennink, B. N. Kaiser, V. C. Marconi, Code saturation versus meaning saturation: How many interviews are enough?, Qualitative Health Research 27 (2017) 591–608. URL: <https://doi.org/10.1177/1049732316665344>. doi:10.1177/1049732316665344.
 - [12] P. Bacchetti, Current sample size conventions: Flaws, harms, and alternatives, BMC Medicine 8 (2010) 17. URL: <https://doi.org/10.1186/1741-7015-8-17>. doi:10.1186/1741-7015-8-17.
 - [13] K. Caine, Local standards for sample size at chi, in: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, CHI '16, Association for Computing Machinery, New York, NY, USA, 2016, p. 981–992. URL: <https://doi.org/10.1145/2858036.2858498>. doi:10.1145/2858036.2858498.
 - [14] B. Saunders, J. Sim, T. Kingstone, S. Baker, J. Waterfield, B. Bartlam, H. Burroughs, C. Jinks, Saturation in qualitative research: exploring its conceptualization and operationalization, Quality & Quantity 52 (2018) 1893–1907. URL: <https://doi.org/10.1007/s11135-017-0574-8>. doi:10.1007/s11135-017-0574-8.
 - [15] C. B. Seaman, R. Hoda, R. Feldt, Qualitative Research Methods in Software Engineering: Past, Present, and Future, IEEE Transactions on Software Engineering 51 (2025) 783–788. URL: <https://doi.ieeecomputersociety.org/10.1109/TSE.2025.3538751>. doi:10.1109/TSE.2025.3538751.
 - [16] J. M. Morse, “Data Were Saturated . . .”, Qualitative Health Research 25 (2015) 587–588. URL: <https://doi.org/10.1177/1049732315576699>. doi:10.1177/1049732315576699.
 - [17] M. Hennink, B. N. Kaiser, Sample sizes for saturation in qualitative research: A systematic review of empirical tests, Social Science & Medicine 292 (2022) 114523. URL: <https://www.sciencedirect.com/science/article/pii/S0277953621008558>. doi:<https://doi.org/10.1016/j.socscimed.2021.114523>.
 - [18] S. Baltes, P. Ralph, Sampling in software engineering research: a critical review and guidelines 27 (2022). URL: <https://doi.org/10.1007/s10664-021-10072-8>. doi:10.1007/s10664-021-10072-8.
 - [19] S. Wagner, D. Mendez, M. Felderer, D. Graziotin, M. Kalinowski, Challenges in Survey Research, Springer International Publishing, Cham, 2020, pp. 93–125. URL: https://doi.org/10.1007/978-3-030-32489-6_4. doi:10.1007/978-3-030-32489-6_4.
 - [20] J. Linåker, S. M. Sulaman, R. Maiani de Mello, M. Höst, Guidelines for Conducting Surveys in Software Engineering, Technical Report, Department of Computer Science, Lund University, Lund, Sweden, 2015. URL: <https://lucris.lub.lu.se/ws/files/6062997/5463412.pdf>, version 1.1.
 - [21] J. Kontio, J. Bragge, L. Lehtola, The Focus Group Method as an Empirical Tool in Software Engineering, Springer London, London, 2008, pp. 93–116. URL: https://doi.org/10.1007/978-1-84800-044-5_4. doi:10.1007/978-1-84800-044-5_4.