Received 21 June 2023, accepted 9 July 2023, date of publication 12 July 2023, date of current version 21 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3294823

RESEARCH ARTICLE

On Adopting Software Analytics for Managerial Decision-Making: A Practitioner's Perspective

THIAGO RIQUE[1,2], MIRKO PERKUSICH[2], EMANUEL DANTAS[3], DANYLLO ALBUQUERQUE[2], KYLLER GORGÔNIO[2], HYGGO ALMEIDA[2], AND ANGELO PERKUSICH[2], (Member, IEEE)

[1]Federal Institute of Education, Science and Technology of Paraíba (IFPB), Princesa Isabel 58755-000, Brazil
[2]Intelligent Software Engineering Group (ISE/VIRTUS), Federal University of Campina Grande (UFCG), Campina Grande 58429-140, Brazil
[3]Federal Institute of Education, Science and Technology of Paraíba (IFPB), Campina Grande 58432-300, Brazil

Corresponding author: Thiago Rique (thiago.rique@ifpb.edu.br)

The work of Thiago Rique was supported by the IFPB qualification incentive program (PIQIFPB)–public notice 109/2021.

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

ABSTRACT Organizations have used software engineering data to support decision-making by applying data-driven approaches such as software analytics. However, adopting analytics tools depends on the information they provide and the real needs of practitioners. Significant research has addressed the needs of developers, whereas the needs of managers are not well understood. Moreover, few studies have focused on the practitioners' view of data-driven decision-making. From a managerial viewpoint, this case study provides an in-depth analysis of the information needs and the perceptions of data-driven decision-making of practitioners from one software development organization. We interviewed personnel in leadership positions and used coding procedures (open and selective coding) to analyze the collected data. We identified 19 software analytics use cases and mapped them to the software life cycle processes from ISO/IEC/IEEE 12207:2017, of which organizational project-enabling and technical management processes were the most highlighted by the interviewees. We also provided a set of indicators to meet the identified use cases and shed light on critical aspects of the organization's analytics scenario. Furthermore, we identified project-related, human-related, and context-specific factors that affect managerial decision-making and organizational aspects that influence the adoption of software analytics initiatives. Although our results are particularly relevant to organizations similar to the one described herein, they aim to serve as input for implementing new analytics solutions by practitioners and researchers in general and contribute to the body of knowledge on the topic from a practitioner's perspective, helping organizations in their attempts to adopt data-driven approaches.

INDEX TERMS Case study, data-driven decision-making, managerial decision-making, software analytics, use cases.

I. INTRODUCTION

Software organizations have focused on obtaining valuable information about their products and processes. As many aspects of development can be measured throughout a project's life cycle, many advances in analyzing software engineering data and developing products through data-driven approaches have been made [1], [2]. One such approach that has gained notoriety is software analytics, which uses data, analysis, and systematic reasoning to support practitioners in making informed decisions about software projects [3]. Through analytic technologies such as data mining, machine learning, and information visualization, practitioners can explore and analyze data to obtain insightful information that they can act on while performing data-driven tasks [4].

The associate editor coordinating the review of this manuscript and approving it for publication was Claudia Raibulet.

Previous research has explored the challenging aspects of making decisions based on software data. Zhang et al. [4] advocate that the investigation of how practitioners act on the information delivered by analytics solutions should be a concern and highlight the need for providing adequate support for decision-making. In addition, Svensson et al. [5] reinforce the scarcity of studies focusing on the practitioners' view of data-driven decision-making, especially for agile organizations.

Adopting software analytics requires eliciting the relevant data for an organization and understanding the relationship between these data and the needs of its practitioners [6]. Buse and Zimmermann [1], [3] indicate that a major factor contributing to the delay or failure of many software projects is the notable disparity between the information provided by analytics tools and the real needs of project managers. They argue that existing research has provided significant findings on the developers' needs, while the managers' have not received the same attention. To leverage the potential of software analytics, some issues identified by the research community need to be appropriately addressed, such as having new in-depth studies on the needs of those who make critical decisions in software projects [1], [2], [3], [7], [8]. Moreover, understanding what influences managers' decision-making plays a relevant role in increasing the effectiveness of project management [9] and, therefore, demands more investigation in the context of data-driven approaches [5].

Most data-driven approaches have focused on supporting software developers. In contrast, this study focuses on better understanding the specific needs and perspectives of software practitioners in leadership positions. This paper reports the results of a case study conducted in one software development organization, aiming to identify use cases, decision-making factors, and organizational aspects that influence the adoption of software analytics initiatives from the perspective of managers. By delving into these areas, we contribute to the knowledge base in this domain with valuable insights useful for organizations making efforts toward becoming data-driven. Furthermore, this study differentiates itself from previous work by prioritizing practitioners in management positions who have received limited attention in data-driven decision-making [8]. To achieve our objective, we focused on the following research questions:

RQ1. What is the current scenario regarding the use of software analytics for managerial decision-making in the organization?

RQ2. What are the information needs (use cases) of practitioners for managerial decision-making, and what aspects of the software engineering process are they related to?

RQ3. What indicators (metrics) are needed for supporting managerial decision-making, and what are the challenges regarding the data required to obtain these indicators?

RQ4. What do managers take into account when making decisions about software projects?

RQ5. What organizational aspects affect the adoption of software analytics?

Preliminary results of our study have been published elsewhere [10] and [11], solely presenting software analytics use cases and related indicators. This paper builds on our previous works as follows: we improved our analysis by identifying the software engineering processes most highlighted by the practitioners, perceiving how well the use cases cover their needs from a broader perspective; we also identified factors affecting managerial decision-making in the context of software analytics and elicited organizational aspects that influence the adoption of such a data-driven approach. We believe that our findings concerning the RQs above provide organizations with valuable knowledge about critical issues that should be considered when attempting to adopt software analytics, enabling a more holistic view.

This paper has the following contributions:

(i) The results presented herein contribute to the research area of software analytics by providing use cases for managerial decision-making. The research community advocates new studies on this topic, which are crucial for developing more effective analytics solutions aligned with practitioners' needs, as new tools will be able to deliver relevant and impactful insights. Further, we mapped the identified use cases to processes from ISO/IEC/IEEE 12207:2017 *Systems and software engineering — Software life cycle processes* [12] (hereinafter referred to as ISO 12207). To the best of our knowledge, previous empirical studies have not provided such a mapping, which is a unique contribution of this paper.

(ii) It points out factors that influence managers' decisions in the context of our study. Given the need to explain and understand several software engineering phenomena, our results can be seen as a step toward adding to the body of knowledge on how managerial decisions are made in software development.

(iii) We identified a set of aspects that affect the adoption of a software analytics approach, which enables an organization to be aware not only of its potentialities but also its limitations so that the organization can take action to overcome them.

The remainder of this paper is structured as follows. Section II presents background literature to help the reader understand the fundamental aspects of our study. Section III presents related work. Section IV describes the employed research design. Section V presents our results. Section VI discusses our results in light of related work. Section VII discusses threats to validity. Finally, Section VIII presents our conclusions.

II. BACKGROUND

Software development is rife with challenges that can impede progress and result in project delays or failures [13]. When suitable data are not available to understand the complexities of a project, it becomes difficult to take effective action to increase the likelihood of success [3]. Furthermore, project and organization-specific factors can also contribute to project failure [13]. Software analytics offers a promising solution to help stakeholders make informed decisions about various aspects of a software project by converting data from different sources into actionable information. However, several research problems still need to be addressed to leverage its potential.

Martínez-Fernández et al. [14] highlighted that existing software analytics tools often fail to provide information connected with higher-quality goals. In their study, they explored the benefits of integrating quality models into analytics tools to effectively evaluate and enhance software quality. They emphasized the importance of adapting the quality models integrated into such tools to better align with companies' specific needs and development processes.

Prior studies have highlighted various challenges associated with software analytics and data-driven software engineering. For instance, Figalist et al. [2] called attention to the importance of selecting appropriate metrics tailored to a specific purpose, as well as asking relevant questions and identifying the distinct information requirements of various stakeholders. Notably, there is significant difficulty in identifying the needs of managers, which may not be readily apparent [1], [3], making it challenging to provide compelling use cases that offer tangible value to organizations [2].

The overarching goal of software analytics is to provide managers and software engineers with actionable insights. However, moving from information to insights is not easy, requiring knowledge of the domain and the ability to identify patterns involving a set of relevant indicators [3]. To obtain insightful information from software data, practitioners take advantage of analytic technologies such as machine learning, which is well-recognized for learning hidden patterns or predictive models from data, playing a relevant role in software analytics [4]. However, several challenges have to be faced along the way, for example: i) lack of transparency of machine learning models, making them difficult to understand; ii) difficulty in specifying use cases for analysis; iii) ensuring the sufficient quality of data; and iv) making teams confident about the results and their value [2]. So, in practice, the analysis of data generated from software engineering activities gets stuck at a prototypical stage, and the results are rarely used to make decisions based on data [2].

III. RELATED WORK

Existing studies have examined the needs of different practitioners in software development. Biehl et al. [15] gathered the requirements of programmers in a large software company and proposed a visualization tool to keep software teams informed of team activities. Buse and Zimmermann [1] conducted a survey to determine the data and analysis needs of managers and developers for software development analytics. Phillips et al. [16] concentrated on the specific context of integration decisions for large-scale parallel development and presented an overview of the information requirements of release managers. Begel and Zimmermann [17] compiled 145 questions about software engineering issues that data scientists could answer to address the information needs of software professionals. Treude et al. [18] researched what information developers would expect in a summary of development activity. In the context of validating and maintaining evolving software systems, Al-Nayeem et al. [19] addressed the information needs of software engineers. Lastly, Pascarella et al. [20] focused on identifying the information needs of developers in the specific context of code review.

In our study, we also focused on decision-making in the context of software analytics. Previous research has investigated the challenges and benefits of organizations implementing data-driven approaches (e.g., [21], [22]). However, few studies have looked into the practitioners' view of data-driven decision-making [5]. Furthermore, such studies did not prioritize those in management positions, who are the target of our investigation.

Drury-Grogan and O'Dwyer [23] investigated the decision-making process in agile teams, identifying three factors that influence decision-making during Sprint Planning and Daily Scrum Meetings: sprint duration, experience, and resource availability. Without being specifically interested in the perspective of managers, the authors focused solely on decisions related to task definition, task estimation, and resource allocation in the Sprint Planning Meeting and decisions on how to remove impediments in the Daily Scrum Meeting. Cunha et al. [9] aimed to understand the decision-making process in software project management, identifying factors that affect how managers make decisions, which the authors classified into two groups: contextual and individual factors. The study was conducted without considering the context of a data-driven approach (such as software analytics in our study).

Svensson et al. [5] looked into industry practitioners' view of data-driven decision-making, investigating their experiences and how data can improve decision-making in agile software companies. Their results indicated that practitioners see such an approach for making decisions as promising, although its potential is currently unfulfilled. Svensson and Taghavianfar [21] conducted an empirical study investigating the challenges and benefits faced by organizations through their attempts to become data-driven in practice. Like [5], the authors did not focus on managerial decision-making in software projects.

In contrast to previous studies, we first provided an overview of the organization's current analytics scenario to facilitate the understanding of the organizational context in which our findings take place. It is worth mentioning that many studies employed a top-down approach (e.g., surveys), using pre-defined checklists for data collection. This may prevent fundamental aspects of the research topic from emerging in collaboration with the participants. Previous studies did not focus on managerial decision-making as most participants are developers. Also, they were conducted under targeted contexts (e.g., awareness in software teams, summary of development activity, code review) or were related to specific decision scenarios (e.g., integration decisions).

TABLE 1. Overview of related work.

Reference	Participants[1]	Study scope or goal	RQ1	RQ2	RQ3	RQ4	RQ5
Biehl *et al.* [15]	Programmers	Awareness in software teams		x			
Buse and Zimmermann [1]	Developers and managers	Needs of managers and developers for software development analytics		x	x	x	
Phillips *et al.* [16]	Developers and managers	Integration decisions in the release process of large-scale parallel development		x		x	
Begel and Zimmermann	Software professionals responsible for development, testing, and program management	Questions about the information needs of software professionals that data scientists could answer regarding software engineering issues		x			
Treude *et al.* [18]	Developers	Summary of development activity		x			
Al-Nayeem *et al.* [19]	Software engineers	Information needs of software engineers for validating and maintaining evolving software systems		x			
Pascarella *et al.* [20]	Developers	Code review		x			
Drury-Grogan and O'Dwyer [23]	Individuals in different roles, mostly developers	Factors influencing the decision-making process during Sprint Planning and Daily Scrum events				x	
Cunha *et al.* [9]	Managers	Decision-making in software project management				x	
Svensson *et al.* [5]	Individuals in different roles, mostly developers	Data-driven decision-making				x	x
Svensson and Taghavianfar [21]	Individuals in different roles, mostly developers	Challenges and benefits of organizations becoming data-driven					x
Our study	Project managers, software architect and lead developer	Focus on adopting software analytics from a managerial perspective, investigating the information needs and perceptions of the decision-making process of practitioners in leadership positions	x	x	x	x	x

x indicates full or partial coverage of targeted RQs.
[1] Detailed information about the participants is available in the papers.

Table 1 provides an overview of related work, describing each study's scope or research goal and indicating how the study maps to the five RQs addressed by our paper. Different research questions are investigated in related studies, not or only partially related to our RQs. In this way, such an overview makes it more evident to what extent our study advances the state-of-the-art.

IV. RESEARCH DESIGN

This section presents the employed research methodology. We aimed to generate practical knowledge related to the practitioners' perspectives on adopting software analytics for managerial decision-making [24]. For this purpose, we conducted a case study in one organization following the specific guidelines for case study research in empirical software engineering [25], [26].

It is worth highlighting that the research community has conducted case studies with a single organization, as can be observed in Rodríguez et al. [27], Phillips et al. [16], Senapathi et al. [28], and Shahin and Babar [29], having, respectively, 10, 7, 6, and 6 participants. Despite "small" sample sizes being commonly criticized without evidence support when assessing the rigor of a study, they are effective for qualitative research and able to reach saturation [30], a feasible criterion to consider when evaluating the validity of a case study [26]. Hennink and Kaiser argue that "sample sizes in qualitative research are guided by data adequacy, so an effective sample size is less about numbers (n's) and more about the ability of data to provide a rich and nuanced account of the phenomenon studied" [30]. Moreover, our study falls into the concept of context-driven research, which should play a more substantial role in software engineering [31]. No universal solution exists for most software engineering problems since the applicability of a solution depends on contextual factors which vary across domains and industries. Briand et al. suggest that "we solve problems in context, identify commonalities and differences across contexts, adapt solutions to different contexts, and generalize over time by building a body of knowledge from concrete experience" [31].

We interviewed practitioners involved in managerial decision-making and used coding procedures available in the qualitative data analysis literature [32] to identify their needs. Thus, our approach enabled them to speak freely and researchers to focus on the knowledge being provided [27].

Next, Section IV-A presents the literature review performed to support the study, Section IV-B describes the organization under study, Section IV-C details the interview procedures and subjects' profile, and Section IV-D summarizes the employed data analysis procedures.

A. LITERATURE REVIEW

Based on a knowledgeable selection of high-quality papers on the research topic at hand, we first performed a non-commital literature review [32] to map the state-of-the-art and identify research gaps. After analyzing the data, we conducted a deeper analysis of the literature to integrate our findings into the context of existing knowledge (as discussed in Section VI). To search the literature systematically, we applied a snowballing approach having Buse and Zimmermann [1] as the seed paper. We did so because it is a seminal work on the topic and advocates new in-depth studies to better understand the information needs of stakeholders in software development analytics.

B. CONTEXT DESCRIPTION

This section presents details that help characterize our case using the context facets by Petersen and Wohlin [33].

Our case is VIRTUS,[1] a research, development, and innovation center that conducts projects in several technological domains (e.g., Web systems, mobile systems, AI, augmented reality, embedded systems, and hardware), focusing on diverse market segments (e.g., security, biometry, and business intelligence). It comprises hundreds of engineers and researchers in its headquarters in Campina Grande, Brazil.

The projects in the organization result from incentive mechanisms between academia and industry promoted by the Brazilian government and are developed with industry partners such as HP, Epson, Envision, Ericsson, and many other large, medium, and small-size technology companies, usually lasting from ten to eighteen months.

The organization is hierarchical and project-oriented from the structural perspective, having a quality department responsible for defining guidelines for its projects' quality processes and auditing them. It generally uses agile approaches such as Scrum or Kanban for project execution. The development practices and tools follow the organization's guidelines and are tailored to meet the projects' needs (e.g., programming language and type of system). A proprietary tool used by the organization to support project management integrates requirement, test, and issue management, source code repositories, and software build systems, enabling the traceability of development artifacts. For this purpose, it follows a model similar to the *Agile Traceability Information Model*, widespread in agile management tools.

C. SUBJECTS AND DATA COLLECTION

We collected data through individual interviews. For this purpose, we had the support of a champion in the organization to identify practitioners in our case that could effectively contribute to this study, as shown in Table 2. As our study focuses on managerial decision-making, we interviewed people in leadership positions. The interviewees were practitioners with a vast experience in the software industry, playing different roles in the organization: one software architect, two project managers, and one lead developer. People in the organization can play different roles depending on the project. For example, being a software architect when he was interviewed, P1 has also played the role of developer and project manager in previous projects, sharing his perceptions from the perspective of a practitioner directly involved in managerial decision-making. Moreover, the practitioners' contribution is particularly significant due to their varied perspectives, further enhanced by the opportunity to collaborate with companies operating in diverse contexts.

The interviews were semi-structured and lasted approximately one hour (we provide the interview script in Appendix A). Table 2 shows the actual length of each interview. Due to the Covid-19 pandemic, the first author of this paper conducted all interviews between October 2021 and January 2022 via *Google Meet*, voice recorded and transcribed with the consent of the participants.

D. DATA ANALYSIS

We started analyzing the data as soon as the first interview was conducted, using an iterative approach in which data collection and analysis occurred in parallel and were stopped when no new insights emerged from the data. This means that the last interviewee did not make any substantive, additional contribution to answer our research questions, which is in line with the instructions presented in Runeson et al. [26].

The interview transcripts went through a systematic multistep process using qualitative data coding procedures, i.e., open and selective coding [32]. During the analysis, we also considered the six phases described in thematic analysis by Braun and Clarke [34]. We applied the constant comparison method (CCM) [32], analyzing the data iteratively. However, aiming for greater readability, the analysis process is described in a sequential manner.

To explain the employed data analysis procedures, we focused on the factors the participants considered relevant to support managerial decision-making. Figure 1 details how we analyzed the data using one of the dimensions identified in the study as an illustrative example, *'human-related factors'*. The first and third authors (A1 and A3) participated in the analysis. To obtain an overview of the data, A1 read each transcript (step 1). In the next step, A1 analyzed the transcripts inductively, using open coding (step 2) by going through the data and attaching codes to relevant concepts. The CCM was applied to identify patterns within each interview. We clustered open codes around factors and dimensions by applying selective coding. Figure 1 shows some instances of the factor *'domain experts' opinion'* as they were coded in P1's interview transcripts. When explaining that he considers specialized opinions to make decisions, P1 used terms such as *'technical leader'* and *'quality team'*, which we aggregated under the factor *'domain experts' opinion'*. By applying steps 3a, 4a, 5a, and 6a, we identified relevant factors from each interview and emailed them back to each interviewee for validation.

[1] https://www.virtus.ufcg.edu.br/

TABLE 2. Interviewees' profile.

P#	Current role	Experience[1]	Previous roles in SW industry	Interview length
P1	Software architect	11(7)	Project manager, developer	62 minutes
P2	Project manager	16(3)	Developer	61 minutes
P3	Project manager	13(7)	Developer, software architect	75 minutes
P4	Lead developer	8(2)	Developer, technical leader	62 minutes

[1] Experience in years: in the software industry (in the target organization).

Next, A1 used the CCM to identify patterns between interviews and saturate categories (step 3b). We made revisions and modifications when necessary (step 4b) until obtaining the final factors and their respective dimensions (step 5b). Table 8 shows the complete list of factors (step 6b).

To mitigate researcher bias in the analysis process, A1 and A3 analyzed interview P1 (the first interview carried out) separately. To ensure validity, A1 and A3 compared their respective individual coding to check whether both researchers identified a similar list of factors. After this step, we agreed that, for all the remaining interviews, A1 would analyze them and prepare individual reports containing the list of factors. Then, A3 would review and validate the factors identified in each interview before emailing them to the interviewees. Aiming to keep a clear chain of evidence, besides the list of factors and their respective descriptions, the reports sent to the interviewees also included the quotations that supported each factor. We performed the analysis using the qualitative analysis tool MAXQDA.[2]

We employed the same steps described earlier to identify use cases for software analytics. However, after inductively identifying them through coding procedures, we mapped such use cases to ISO 12207 software processes using a deductive or a priori approach that can help researchers integrate concepts already well-known in the extant literature [35]. As ISO 12207 depicts each process into activities and corresponding tasks, we associated them with each use case when applicable. Such a mapping went through a peer review process in which the first two authors participated (A1 and A2) and was documented and made available as supplementary material.[3]

V. RESULTS

This section presents the results of our study by addressing the RQs. First, we present the current software analytics scenario of the organization under study, answering RQ1. Then, RQ2 is answered by identifying relevant use cases for software analytics in the context of managerial decision-making. We answer RQ3 by pointing out indicators identified in the interviews. Next, we present the interviewees' perceptions when making managerial decisions in software development, answering RQ4. Finally, we answer RQ5 by discussing aspects of our target organization that affect the adoption of software analytics.

A. CURRENT SOFTWARE ANALYTICS SCENARIO (RQ1)

We aimed to investigate the use of analytics for managerial decision-making within our target organization. Our analysis revealed the typical problems practitioners usually seek to solve, primarily associated with team productivity, risks, schedule, and quality. We also identified critical aspects that can be seen as challenges the organization must confront to leverage its analytics approach, which we summarized in Table 3. For more in-depth insights, such as quotations from the interviews, we refer to our prior publication [11].

> ***Finding 1.*** *Practitioners from our target organization are interested in solving problems mostly related to team productivity, risks, schedule, and quality. Critical aspects of the organization's current scenario can be seen as challenges to be faced to leverage its analytics approach: experts' opinion over data, adherence to the development process, need for parameterizing the development process, analytics approach limited to data visualization, data analysis based on practitioners' experience, trustworthiness of the current approach, and data format issues.*

B. SOFTWARE ANALYTICS USE CASES (RQ2)

We identified 19 software analytics use cases. Table 4 shows the list of use cases, their description, and the corresponding number of participants who mentioned them. Descriptions emerged directly from the interviews. Also, most use cases (e.g., *managing product quality*) are rather general, likely applicable to many other organizations, and in some cases, already identified in previous studies (see Section VI). They are also grounded in our empirical data, thus genuinely representing the needs of decision-makers in our case. Conversely, a few use cases are more specific to our target organization; for example, *identifying new features for projects* is directly related to the way how projects work in the organization.

Not all use cases were addressed with the same level of detail. The Totals column in Table 4 shows the number of practitioners that mentioned each use case, and the corresponding number of times (instances) that the use case was identified in the transcripts. The complete list of use cases and their respective quotations is made available as supplementary material.[4] We also refer to our previous work [11] for a more comprehensive description of use cases.

To find out what aspects of the software engineering process were highlighted by the participants, we mapped the identified use cases to the software life cycle processes from ISO 12207 [12], which establishes a common framework con-

[2] https://www.maxqda.com/
[3] https://doi.org/10.5281/zenodo.7628890
[4] https://doi.org/10.5281/zenodo.7628975

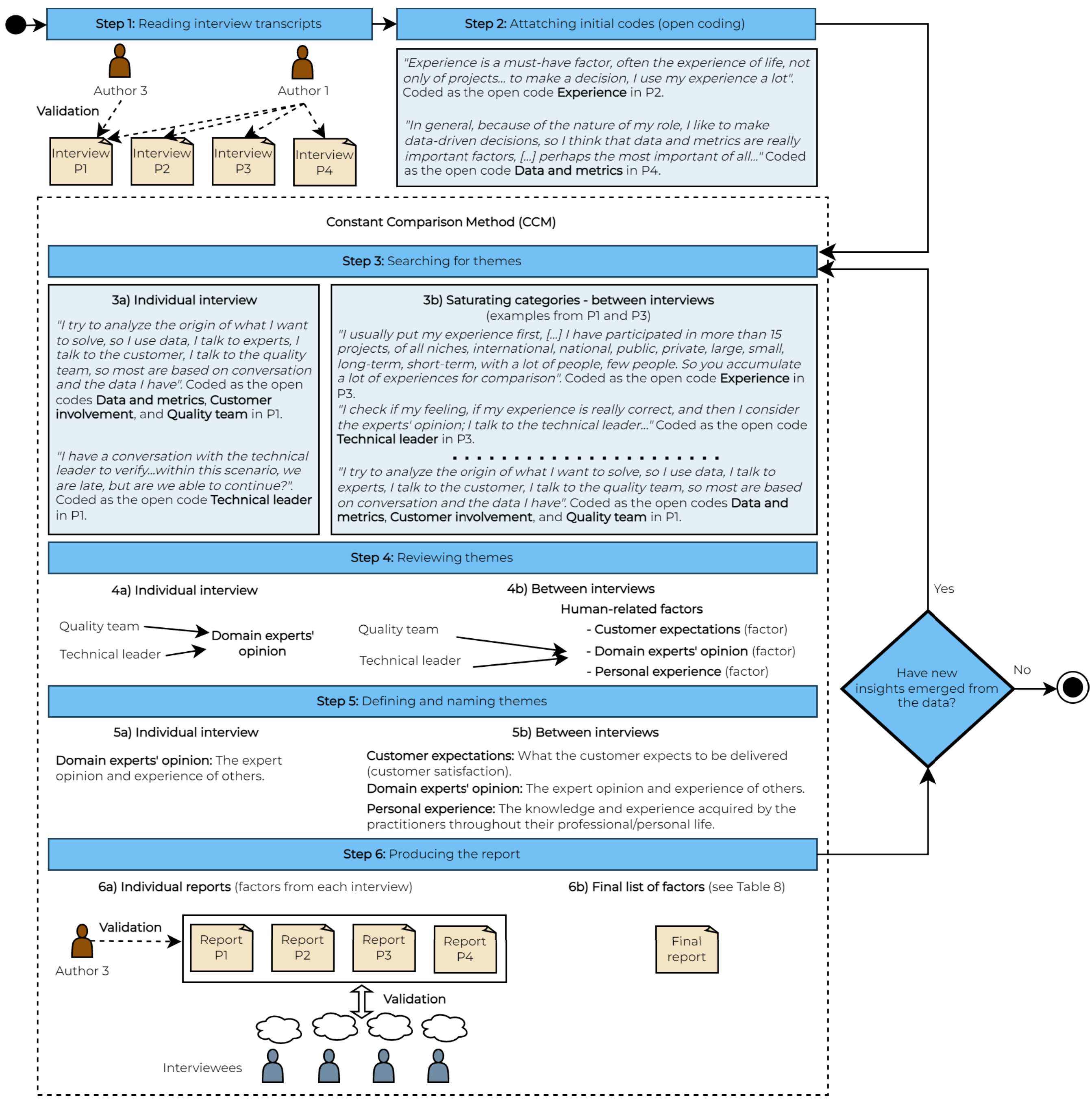


FIGURE 1. Iterative data analysis procedures.

taining processes, activities, and tasks that can be applied to the full life cycle of software systems, products, and services. It arranges those activities into four process groups:

1) Agreement processes
2) Organizational project-enabling processes
3) Technical management processes
4) Technical processes

Our mapping resulted in the association of use cases with the four process groups mentioned above. However, as our focus is on managerial decision-making, this paper only addresses the organizational project-enabling and technical management processes since they concentrated the vast majority of use cases, which points to their relevance in the software development context from the managers' perspective. The complete mapping is publicly available as supplementary material (referenced in Section IV-D). Table 5 depicts both process groups into the corresponding life cycle processes and shows their association with our use cases. Notice that a few use cases were mapped to multiple ISO

TABLE 3. Critical aspects regarding the organization's analytics scenario.

Aspect	Description
Experts' opinion over data	While trying to solve a problem, the practitioners rely mostly on the experts' knowledge and experience.
Adherence to the development process	This aspect relates to mechanisms that enable practitioners to verify the correct execution of the development process, ensuring that established practices are applied.
Need for parameterizing the development process	This aspect relates to parameters that should be systematically monitored in a project during the development process.
Analytics approach limited to data visualization	All participants mentioned limitations regarding the insights provided by the current approach, which focuses on organizing and presenting data to facilitate their interpretation. Suggestions for using alerts to notify the practitioners about unusual situations were proposed.
Data analysis based on practitioners' experience	The participants stated that they analyze the data more intuitively or based on their experience to make decisions about the projects.
Trustworthiness of the current approach	The participants recognize the usefulness of the approach currently used. However, problems related to data insertion or the fact that the organizational tool does not reflect the complete reality of the projects are critical issues.
Data format issues	Since different projects work with varying data formats, adaptations have to be made, affecting the data.

12207 processes (e.g., UC01 was mapped to *quality management process* and *quality assurance process*).

Finding 2. *The 19 software analytics use cases identified in our analysis (see Table 4) relate to the following ISO 12207 processes:* ***organizational project-enabling processes*** *(portfolio management [UC14]; human resource management [UC08, UC09, UC10]; quality management [UC01, UC03, UC04, UC06, UC07]; knowledge management [UC18, UC19]) and* ***technical management processes*** *(project planning [UC17]; project assessment and control [UC11, UC12, UC13, UC15]; risk management [UC16]; quality assurance [UC01, UC02, UC03, UC04, UC05, UC06]).*

C. INDICATORS (RQ3)

During our interviews, we asked the participants about the indicators they deemed necessary to fulfill their needs. Table 6 presents these indicators associating them with the software life cycle processes addressed in our analysis. In cases where an indicator may not be self-explanatory, we included a brief description in parentheses for clarity.

The interviewees highlighted certain obstacles related to the availability and format of the data necessary to obtain these indicators. Table 7 provides a summary of these challenges. More comprehensive insights into them have been previously published [11].

Finding 3. *Practitioners highlighted the following challenges regarding indicators (see Table 6): data from different sources, data not available, adding new features to start collecting data, need for structuring data input, and need for parameterizing the development process.*

D. FACTORS AFFECTING MANAGERIAL DECISION-MAKING (RQ4)

We captured several factors influencing managerial decision-making deemed relevant by the participants, which we classified into three dimensions given their similarities: (i) project-related, (ii) human-related, and (iii) context-specific factors.

Project-related factors refer to aspects of software projects that affect the reasoning employed by managers when making a decision. Human-related factors refer to the human aspects of software engineering, including the stakeholders' needs, expectations, expertise, and the project team's dynamics. Finally, context-specific factors include organizational characteristics and constraints that play a role in data-driven managerial decision-making.

Table 8 shows the list of factors, their description, and the corresponding number of practitioners who mentioned them. Most factors (e.g., *project data* and *customer expectations*) are rather general, likely applicable to many other organizations, and in some cases, already identified in previous studies (see Section VI). They are also grounded in our empirical data, thus genuinely representing what managers keep in mind when making decisions in our case. Other factors are more specific to our target organization; for example, *innovation level* is directly related to the type of projects it executes.

Not all factors were addressed with the same level of detail. The Totals column in Table 8 shows the number of participants that mentioned each factor, and the corresponding number of times (instances) that the factor was identified in the transcripts. Human-related factors were the ones mentioned the most, i.e., coded 26 times in MAXQDA. Next, we provide illustrative quotations from the interviews aiming to keep a clear chain of evidence for factors and their dimensions. The complete list of factors and their respective quotations is made available as supplementary material.[5]

Project data (F1): *"The second most important [factor] is obviously the quality of data, right? It's not only the data but the quality of the data. It's the real data, from the observation to the correct insertion of the data in the tool, because it is useless having the best tool in the world if the input is wrong. You will not have correct outputs with wrong inputs, so I think this is the second most important part."* (P3).

Project constraints (F2): Some constraints introduce bias into decision-making and impact the execution of projects

[5]https://doi.org/10.5281/zenodo.7628994

TABLE 4. List of identified use cases.

Id	Use case	Description	Totals[1]
UC01	Managing product quality	Assessing how quality assurance is being performed regarding a software product (e.g., do we have the OK from the quality team for the upcoming build? Does source code meet quality standards?).	2(7)
UC02	Identifying the source of bugs	Identifying what is causing bugs in a software project (e.g., who are the members responsible for most bugs?).	2(8)
UC03	Managing bugs	Managing bugs through their entire life cycle (i.e., from identifying the bug to closing the bug).	1(1)
UC04	Supporting the definition of the development/quality process	Defining the adequate practices to meet requirements (e.g., will load testing be performed? What percentage of unit test coverage is required? Will Scrum or Kanban be used?).	1(1)
UC05	Monitoring process compliance with standards	Guaranteeing that the defined practices are being followed to achieve the desired quality level (e.g., are Daily Meetings taking place daily? Are the load tests being performed before deployment in the approval environment? Are the deployment environment being monitored?).	1(8)
UC06	Monitoring technical debt	Verifying if technical debt is increasing or decreasing.	1(1)
UC07	Supporting project compliance with customer expectations	Getting feedback on customer satisfaction (i.e., if what was delivered met customer expectations).	2(3)
UC08	Identifying training needs	Identifying aspects in the project that indicate a training need (e.g., low-quality code, code with many bugs).	1(2)
UC09	Assessing how much a project depends on a certain person	Assessing the extent to which a project depends on a certain person (e.g., how much a project depends on a developer?).	1(2)
UC10	Monitoring the well-being of teams	Getting feedback on the well-being of a team (e.g., is the team satisfied? Is the team motivated?).	1(1)
UC11	Managing team productivity	Managing different aspects of team productivity (e.g., the most productive people, story points delivered, how long it takes for a developer to complete a task).	4(15)
UC12	Monitoring project schedule	Having an overview of several conditions regarding project schedule (e.g., will the team be able to deliver what was agreed for a sprint? Is the schedule delayed?).	4(9)
UC13	Supporting schedule delay management	In the face of schedule delays, enabling stakeholders to take steps that lessen the impact on the project (e.g., how much extra effort does the team need to make to mitigate the schedule delay?).	3(3)
UC14	Identifying new features for projects	Identifying new functionality for projects (e.g., to renew the contract, the customer wants new features for the project so that he/she keeps that investment in the organization).	1(3)
UC15	Managing project scope	Managing scope issues (e.g., changes in the scope agreed for a sprint or assessing if a new scope affects team velocity).	1(2)
UC16	Managing project risks	Managing the factors that compromise the progress of the project, reducing their impacts (e.g., what is the planning of the team to mitigate the impacts of changes in the project?).	4(8)
UC17	Increasing delivery value	Including a greater number of user stories into a sprint planning (i.e., a product increment that delivers direct value to the customers).	1(1)
UC18	Gathering decisions from different areas of the organization	Getting feedback on past decisions with regard to problems affecting different sectors of the organization (e.g., when a problem needs to go through different areas of the organization, the solutions proposed by each sector will help the manager make a decision).	1(1)
UC19	Solving problems with the support of lessons learned	Using the organization's experience in solving problems (e.g., solutions that worked and did not work for a problem that happened in the past so that managers will have support to solve the problem in the present).	2(2)

[1] Number of practitioners that mentioned the use case in the interview (number of instances recorded in MAXQDA for the use case).

TABLE 5. Mapping of use cases to software life cycle processes from ISO/IEC/IEEE 12207:2017.

Software life cycle processes	Use cases (UC)																		
	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19
Organizational project-enabling processes																			
Life cycle model management process																			
Infrastructure management process																			
Portfolio management process														x					
Human resource management process								x	x	x									
Quality management process	x		x	x		x	x												
Knowledge management process																		x	x
Technical management processes																			
Project planning process																	x		
Project assessment and control process											x	x	x		x				
Decision management process																			
Risk management process																x			
Configuration management process																			
Information management process																			
Measurement process																			
Quality assurance process	x	x	x	x	x	x													
Number of use cases per process	Portfolio management process: 1 Human resource management process: 3 Quality management process: 5 Knowledge management process: 2									Project planning process: 1 Project assessment and control process: 4 Risk management process: 1 Quality assurance process: 6									

in the organization, as highlighted by P4: *"sometimes the schedule makes us make some decisions, prioritize some things…So schedule brings a bias when making some decisions"*. He added: *"the schedule is a very inflexible thing in the way [the organization] makes contracts, where basically I am company X, I contract [the organization], I am paying 12 months for that team allocation in there, so I cannot fail to meet that deadline"* (P4).

TABLE 6. Indicators to meet the identified use cases.

Indicators	Organiz. project-enabling[1]				Technical management[2]			
	PM	HRM	QM	KM	PP	PAC	RM	QA
Amount of suggestions for new features and their acceptance rate by customers (the extent to which a feature suggestion was relevant to customers)	x							
Human resources/team climate indicators (it refers to indicators that allow measuring the well-being of teams, e.g., how satisfied, how motivated, etc.)		x						
Requirements coverage by tests			x					x
Amount of bugs (e.g., opened and reopened bugs)			x					x
Bug increase rate (the extent to which the amount of bugs is increasing)			x					x
Bug criticality			x					x
Extent to which the product meets customer expectations (e.g., how satisfied the customer is)			x					
Number of managers who applied a given solution				x				
Success rate of the applied solution (the extent to which a given solution was successful)				x				
Project focus (the focus of the project in which a given solution was applied, e.g., hardware project or software project)				x				
Number of communication errors (number of times a sector's request was not fulfilled because of a communication error)				x				
Progress of requests (the path a request should follow, identifying in which sector and how long it is being processed)				x				
Number of requests per sector (together with the progress of requests indicator, it aims to monitor which sectors of the organization are responsible for most bottlenecks)				x				
Possible solutions indicator (it indicates a possible solution based on past decisions made by an organizational sector)				x				
Points delivered						x		
Points stability						x		
Points per person						x		
Number of tasks with delayed schedule						x		
Number of tasks linked to risks							x	
Number of opened and mitigated risks							x	
Compliance with a quality process (the extent to which a project follows the defined practices for quality)								x
Number of times a developer's code failed								x
Number of times a developer's code failed per module								x

[1] PM: Portfolio management | HRM: Human resource management | QM: Quality management | KM: Knowledge management

[2] PP: Project planning | PAC: Project assessment and control | RM: Risk management | QA: Quality assurance

TABLE 7. Challenges regarding the availability and format of the data required to obtain indicators.

Challenge	Description
Data from different sources	Project data are scattered across multiple tools, making them difficult to share.
Data not available	Some practitioners mentioned that, for some use cases, not all data are available to obtain the indicators to meet their needs.
Adding new features to start collecting data	Implementing new features in the organizational tool was mentioned as an alternative in the interviews to make the data required to meet the practitioners' needs available.
Need for structuring data input	Some data are not recorded in a structured way (e.g., using an input text), and each manager uses his own format, which makes it difficult to gather these data.
Need for parameterizing the development process	This factor appears to be of great importance since it is pointed out again in our results. There is a need for a better parameterization of the process so that the practitioners can monitor what they want in the projects in a systematized manner.

Customer expectations (F3): *"What influences my decision? The customer expectation, I think it's the key, right? Because any problem that occurs actually occurs because the customer is expecting something, and delivering what the customer expects is the whole point"* (P1).

Domain experts' opinion (F4): *"I validate whether my feeling, my experience is really correct, and then I consider the experts' opinion, right? [...] That is how I usually validate things"* (P3). *"I have a conversation with the technical leader to verify if... within this scenario we are late, but are we able to continue?"* (P1).

Personal experience (F5): *"I usually put my experience first, [...] I have participated in more than 15 projects, of all niches, international, national, public, private, large, small, long-term, short-term, with a lot of people, few people. So you accumulate a lot of experiences for comparison. So the first thing is the feeling, 'I've already experienced this reality elsewhere', it's the first thing"* (P3). *"Experience is a must-have factor, often the experience of life, not only of projects... when making a decision, I use my experience a lot"* (P2).

Innovation level (F6): *"There is something that is important not only to me but to [the organization] as well, whether I am applying innovation. We are in a context in which we necessarily have to work with innovation. I'm not just there to make a product for the company to make money. We have to work with innovation in the context of the project, okay? So this is important for decision-making"* (P2).

We also captured key aspects that complement the identified factors in Table 8 and contribute to understand the reasoning employed by our participants in the decision-making process. Next, we provide examples of evidence in which our results are grounded.

TABLE 8. Factors affecting managerial decision-making.

Id	Factor	Description	Totals[1]
Project-related factors			**4(13)**
F1	Project data	Any information about a project provided by SE tools that impacts decision-making.	4(11)
F2	Project constraints	Any constraint that impacts the project (e.g., schedule, cost, scope).	1(2)
Human-related factors			**4(26)**
F3	Customer expectations	What the customer expects to be delivered (customer satisfaction).	2(8)
F4	Domain experts' opinion	The expert opinion and experience of others.	4(11)
F5	Personal experience	The knowledge and experience acquired by the practitioners throughout their professional/personal life.	3(7)
Context-specific factors			**1(1)**
F6	Innovation level	Given the type of projects the organization executes, innovation should be a concern when deciding the direction a project should take.	1(1)

[1] Number of practitioners that mentioned the factor in the interview (number of instances recorded in MAXQDA for the factor).

Project aspects: To make a decision, P1 stated that he analyzes whether a given problem compromises the customer's expectation. If so, he tries to answer how, looking for several project aspects that could be affected. *"At first you have the customer's expectation and then... 'Does that compromise the customer's expectation? Yes, it does'. And then, I start to analyze how it is going to compromise. 'It compromises the schedule'. So, as it compromises something within the schedule, I'll check the data and analyze... how late is it? In what part of the scope are we? Have we managed to deliver? What is the priority of the items? What are the lowest priority items?"* (P1).

Impact analysis: P1 mentioned that, when determining the direction a project should take, one key step is choosing from a set of alternatives. *"So, when you gather all this information, you have some possible decisions, analyze the impacts that they can have for that moment, and what can be worse and better in each of them. Measuring the impacts, you decide where to go"* (P1). P2 provided an example: *"if we have quality problems, then I need to work together with the team and the board at the same time, either by bringing in an expert or changing a person that may be causing a problem in the code or training the quality team"*.

Continuous quality improvement: *"You try to have a kind of continuous improvement within the project. And if a problem happens here, I don't want it to happen again from now on, and how are we going to change that? So if it's been a problem in the development process, I'll focus on the development process part, then I'll see if there was a failure in the development process"* (P1).

Awareness of the problem that leads to decision-making: *"I think the identification of the need [of making a decision] basically happens together with the data. When not, you need to look for more data, which sometimes may even involve the customer, talking to experts, you may need to consult the tool"* (P1). *"[The decision-making process] is not very clear, but... I have this input... and from this input, I'll make decisions, okay? For example, if I see that my team's productivity is decreasing, I will act on this, find out what is happening, if there was a problem with someone, if the part that people are working on... if the activity or task is complex"* (P2).

Continuous decision-making: P1 described decision-making as a continuous process in which some decisions are made quickly, while other decisions require a more detailed analysis. *"[The decision-making process] doesn't have a very clear flow, and we make decisions all the time, you know? So we can make a decision very quickly, we can make a decision that requires a deeper analysis... but [decision-making] happens all the time"* (P1).

Complexity of the problem: P1 also mentioned that the complexity of the problem being addressed affects how he makes decisions. Depending on the type of problem, making decisions may be simple or may require a thorough analysis of the problem, as he gives an example: *"I think that's how [decision-making] works, what changes sometimes is the time you spend to make a decision. For example, you saw that the customer has a test coverage expectation, which is a metric he wants, and you observed that at the end of a sprint, when the quality team brought it in, the test coverage metric was lower than expected. You can immediately say, 'hey guys, in the next sprint we'll separate a part of it, we'll have to work to do it, to achieve test coverage' "* (P1).

> ***Finding 4.*** *Practitioners highlighted project-related, human-related, and context-specific factors affecting managerial decision-making (see Table 8). In addition, the reasoning behind the decision-making process involves the following: project aspects, impact analysis, continuous quality improvement, awareness of the problem that leads to decision-making, continuous decision-making, and complexity of the problem.*

E. ORGANIZATIONAL ASPECTS AFFECTING THE ADOPTION OF SOFTWARE ANALYTICS (RQ5)

We asked the interviewees about organizational aspects that facilitate (+) or hinder (−) the adoption of software analytics. Next, we provide illustrative quotations to keep a clear chain of evidence for such aspects.

+ **Project-oriented approach:** *"We work in a project-oriented way, with a well-defined project structure, right? [...] So, one thing that I think favors the analytics approach is the good structuring and the definition of what a project within [the organization] is, right?"* (P1).

+ **Well-defined project setting:** *"[...] you have at least one proposal related to the product, how many people you will need to work with, which is information that not every company thinks about before starting to develop. [In the* *organization], you have a well-defined budget, you have a start date, an end date, a preliminary view of the scope, which is the most volatile part, and the number of people… that is good"* (P1).

\+ **Context of research, development, and innovation:** *"Aspects that favor… firstly because we are in a context of research and development, so I think this is inherent to [the organization] itself, as we always work with research and innovation, so we are in an environment conducive to the application of everything that is new to improve our processes […] we are not a software factory… so, naturally, a software factory aims to deliver the product with what the customer asked for and get its return for that. In [the organization] we are research, we are innovation"* (P2).

\+ **Organization's effort to leverage its analytics approach:** *"We have already started something similar, I mean the board has already started something similar, there is already a considerable effort, so we are still collecting data […] but we are open to that"* (P2).

\+ **Diversity of projects:** *"We have projects with different characteristics, so we have samples that favor a very diverse data analysis. So I think this is quite favorable"* (P2).

\+ **Research support:** *"I think that what favors [the application of software analytics]… I think that the support for research in [the organization] is really interesting, right? [The organization] is usually very open to that"* (P4).

– **Not well-defined project requirements:** Many projects in the organization begin without a clear definition of the parameters they should have. P1 stated that *"one difficulty that we have, within the [organization's] structure, in my view, is great volatility of requirements… let's say, a high level of uncertainty of these points at the beginning of the project. Maybe if we tried to solve these points in an initial phase, even if it took a month or two, having these parameters well-defined would help, but I think that today this fails in many projects; in the end, there is a project plan that does not reflect the project, and that is why you cannot define some points. So I think that sometimes this volatility, this flexibility can affect these analytics issues that I highlighted, related to process and product"*.

– **Low process standardization:** P1 called attention to the need for improving process standardization and monitoring the projects' compliance with such standards, as he explained: *"The points approach in Scrum, how it works… I've seen a million times the manager saying 'on my team, half a day is 3 points'. It doesn't work, you know? It's not like that… if you know the productivity calculation, you can see that this will never measure productivity increase or decrease; it's impossible, it's impossible. Mathematically, this is impossible, so… there is no standard for this kind of thing, there is no well-defined standard, a very clear guide to practices, right? And the absence of this makes you unable to perform a well-done data analysis. Now, this is the point that would need to be improved, so if I were to summarize, to have useful data, you need to guarantee the quality of the process, so there is nothing today that tells you, for example, 'how is the level of adherence to the process in this project?' It does not exist"*.

– **Low investment in the organization's tools:** *"Although everyone here has a degree in computing, it turns out that our tools don't have much investment. We invest a lot in our customers' and partners' tools, but we don't invest in our own tools. Generally, our history is always tools made in a hurry, with teams that are not mature. They are usually university students who are interns, then [the organization] hires interns to do it, the code gets bad, a lot of problems, anyway… processes are not applied properly"* (P3).

– **Project data confidentiality:** *"Projects are confidential, right? We can't share project data even with other people from [the organization], because it only involves those people on the team, […] so there's this obstacle too. How do you get data from several projects if you have data confidentiality? I don't know what that would be like […] Other institutions, for example, do not have such confidentiality issues because they do not usually work with innovation, right? They usually work with off-the-shelf projects, so confidentiality is not a big deal as it is in our case"* (P3).

– **Data availability issues:** *"On the aspects that make it difficult, I think that in general, then I'll say again that it's my perception, there's a chance of a good part of the required data not being available, so you run the risk of this happening. If you have a good idea of how to develop and everything else, but either that is not being stored, or it is not available because it is not being collected in an automated way, and it may also be that today the tool used for project management does not support this type of implementation"* (P4).

> ***Finding 5.*** *The following organizational aspects facilitate the adoption of software analytics: project-oriented approach, well-defined project setting, context of research, development, and innovation, organization's effort to leverage its analytics approach, diversity of projects, and research support. Aspects hindering the adoption of software analytics: not well-defined project requirements, low process standardization, low investment in the organization's tools, project data confidentiality, and data availability issues.*

VI. DISCUSSION

This section integrates our results into the existing body of knowledge on the topic. Our goal is to analyze to what extent the information needs identified in our study have also been considered in related work and discuss important aspects of data-driven managerial decision-making, outlining some research directions.

A. COMPARISON TO INFORMATION NEEDS IDENTIFIED IN RELATED EMPIRICAL STUDIES

It is worth mentioning that the studies presented in Section III have some particularities that made it difficult to analyze to what extent the information needs identified in our study have also been addressed in related research. Many of these studies did not specifically concentrate on managerial decision-making and were conducted within specific, targeted contexts. As a result, it can be challenging to draw direct comparisons between our findings and those presented in previous literature.

On the current analytics scenario of our target organization, our results support relevant findings in the literature. Figalist et al. [2] highlighted certain challenges within the field of software analytics, including the lack of trust in data-driven approaches and the difficulty of sharing data among teams due to different data sources and formats. These challenges align with the aspects *trustworthiness of the current approach* and *data format issues* identified in our study. Similarly, the aspect *need for parameterizing the development process* supports the call by Martínez-Fernández et al. [14] for adapting quality models integrated into analytics tools to reflect companies' needs and development processes.

Concerning the needs of practitioners, Buse and Zimmermann [1] presented information needs through decision scenarios, enabling a broader view. Our study attempted to elicit these needs in the form of clear use cases, approaching them in a more targeted manner. Although differing in terminology, the use cases UC02 and UC03 from our results relate to *targeting testing* in [1] as they involve testing aspects such as the source of bugs and the management of bugs through their entire life cycle. In Buse and Zimmermann's work [1], *targeting testing* relates to testing activities for which information on the code or bug fixes is necessary (e.g., test allocation). The same occurs with the use case UC06 and *targeting refactoring* in [1] since both technical debt and refactoring impact software quality characteristics. The use case UC08 from our results also supports Buse and Zimmermann's findings given that it also appears in [1] as *targeting training needs*. Although interviewees in our study referenced other scenarios discussed in [1] (e.g., the one related to stability), they did not emphasize or provide sufficient details to justify their inclusion as separate use cases. These scenarios were rather used by interviewees to illustrate and support the use cases that emerged from our analysis.

Biehl et al. [15] focused on a targeted context, addressing a tool for team activity awareness. Their study involved programmers whose needs were elicited to support developing such a tool, but not discussed in sufficient detail, which hindered comparing them with our results. So, we could not identify correspondences between the use cases from our work and the programmers' needs in [15]. The same applies to the studies [18], [19], [20], focusing on targeted contexts such as summary of development activity, code review, and validating evolving software systems, respectively.

Phillips et al. [16] used interviews and coding techniques to elicit the information needs of the stakeholders involved in their study. Hence, their findings appear not to be limited due to the shortcomings in the research method mentioned earlier in this paper. However, their work focused on integration decisions in parallel development, and we did not identify any use case in that specific context.

Begel and Zimmermann [17] elicited 145 questions as developers' information needs, making it difficult to find a reasonable correspondence between each question and the use cases from our results. However, such questions were grouped into categories, and an interesting fact is that the *productivity* category *"is what [many respondents] think of when they hear the term 'software data analytics' "* [17]. In our study, the most mentioned use case by the interviewees was UC11 (*managing team productivity*), which is consistent with this claim.

Most of the indicators elicited in our study were not considered in previous research on practitioners' information needs for software analytics. This is because the research method we used facilitated the emergence of these indicators from the analysis of the interview transcripts rather than relying on predefined lists provided by researchers. We could only compare our indicators with those from the study conducted by Buse and Zimmermann [1], given its focus on analytical decision-making. In their study, we found *bug reports* and *test coverage* to be the only indicators we could relate to ours, *amount of bugs*, *bug increase rate*, *bug criticality*, and *requirements coverage by tests*.

Existing studies have addressed the use of metrics in agile software development (e.g., [36], [37]). Their findings can be used to complement our set of indicators, as some metrics help meet the needs elicited in our study. We did not provide a detailed comparison to such studies because they do not focus on the specific needs of managers for software development analytics.

Finally, our analysis contributed to the literature on software analytics by mapping the use cases identified through our coding procedures to dimensions from ISO 12207 since it classifies software processes as agreement (e.g., acquisition and supply), organizational project-enabling (e.g., human resources and knowledge management), technical management (e.g., project planning and risk management), and technical (e.g., software requirements and validation). This allowed us to identify which types of processes were highlighted by the interviewees and perceive how well the use cases cover the needs of managers from a broader perspective.

Our data show that quality issues are a key concern for managers since many use cases are related to the quality management and quality assurance processes. This fact suggests that managers are interested in using analytics solutions to ensure that products and services meet organizational and project quality objectives and customer satisfaction. Also, it is essential to guarantee that the developed product is of the desired quality and follows the established procedures.

Managers are also interested in other types of processes while performing their roles. Among their purposes, we could perceive the following: monitor project status, and technical and process performance, besides directing execution to help ensure performance by plans and schedules (project assessment and control process); provide the organization with necessary human resources and maintain their competencies (human resource management process); provide the organization with the capability to exploit opportunities to reuse existing knowledge (knowledge management process); identify, analyze, and monitor risks (risk management process); sustain projects, performing their assessment to confirm they justify continued investment (portfolio management process); and coordinate workable plans to increase delivery value (project planning process).

B. DISCUSSION ON DATA-DRIVEN MANAGERIAL DECISION-MAKING

Not many studies in the literature provide a practitioner's view of data-driven decision-making [5], which hindered comparing our results with related work. Moreover, such studies do not focus on a managerial perspective.

Our results concerning what managers take into account when making decisions support or complement relevant findings in the literature. For example, data and metrics appear as the most important factors to managers in [1], followed by other factors such as customer input and personal experience. Such factors were also highlighted in our results. However, although the factor *project data* is among the most mentioned factors influencing decision-making in our study (see Table 8), we cannot claim that it plays a major role since other factors appeared to be more relevant to our practitioners.

Given its focus on the practitioners' view of data-driven decision-making, the work of Svensson et al. [5] allowed for a richer discussion of our results. Their findings indicated that most respondents disagreed or strongly disagreed that data are important and highly valued for decision-making. In addition, data are not even treated as an asset. Their results also showed that data are seldom (never or sometimes) used in decision-making. However, a vast majority of respondents believe that, in the future, data should be viewed positively and used most of the time or always in decision-making. It is also interesting to note that data do not appear (at least not explicitly) in the list of factors affecting managers' decision-making in the work of Cunha et al. [9].

In contrast, our results showed that practitioners do not have such a negative view of data-driven decision-making. Instead, their decisions are based on data, but they recognize that other factors should be strongly considered. This fact reinforces the issues that need to be addressed so that data can play a more effective role in software analytics approaches, helping practitioners make better decisions based on high-quality and reliable data [2], [11], [38].

One of the reasons given by the respondents in [5] for not using data today is that data may not be available. This is in line with one of the aspects presented in our study that make it difficult to adopt software analytics in an organization. In addition, the possibility of not being available was one of the elicited challenges concerning the data required to obtain indicators to meet the needs of decision-makers in our case.

Instead of using data, the practitioners involved in the study of Svensson et al. [5] explained that decision-making is mainly based on 'gut feeling', their experiences, or the value for customers. This is in line with our human-related factors, which were the most mentioned by our participants, pointing out the predominant subjective reasoning in decision-making. Creating and rapidly releasing software products requires that such products are based on data and customers' real-time feedback [39]. Therefore, changes and improvements in the development processes are some challenges to be faced when moving from a subjective (mainly based on practitioners' experiences) to a data-driven decision-making process [5], [39]. In this sense, our results suggest that, in a data-driven approach such as software analytics, defining which parameters to monitor in a project enables practitioners to verify the adherence of that project to the development process in a systematic manner.

Another factor that comes into play when adopting data-driven decision-making is related to the organization's characteristics. Svensson and Taghavianfar [21] investigated the challenges and benefits organizations face when moving toward becoming a data-driven organization. Our findings related to the aspects that facilitate the adoption of software analytics can be seen as drivers to overcome such challenges, whereas the aspects that hinder a software analytics approach can be seen as motivators for organizational change so that organizations can benefit from becoming data-driven.

Svensson and Taghavianfar [21] pointed out the need to promote an organizational culture to face the challenges regarding data-driven decision-making. The context of the organization investigated in this study and its support for research represent strengths in overcoming such challenges, as well as being an indicator that the organization is open to adopting data-driven approaches and investing in improving its processes. Our results also point out process issues as key challenges while drawing attention to the importance of defining parameters in a software project that enable the verification of its adherence to the development process. This confirms other findings in the literature, such as the challenge of establishing new processes aligned with data-driven needs [21] and the need to standardize processes and monitor the compliance of projects with such standards [11].

We also shed light on data issues such as confidentiality. Given the context of innovation, data restrictions inherent to the type of projects performed in our target organization represent a challenging aspect. Organizational tools are also critical when it comes to data issues. This finding is in line with the need to invest in tools and technologies for data collection, storage, sharing, and analytics presented in [21]. Finally, Svensson and Taghavianfar highlighted creativity, innovation, and growth opportunities among the benefits of being data-driven. Such benefits are strongly related to the context of the organization under study described in Section IV-B.

Data availability has been pointed out as a success factor for software metrics programs [40]. However, our study goes beyond such a finding: it claims that there are problems for which collecting data might be unfeasible, leading managers to rely on other factors. Our results corroborate Svensson et al. [5] by identifying such factors. Svensson et al. [5] identified five aspects that need to be combined with data for better decision-making: (1) own experience, (2) business value, (3) customer value, (4) input from key stakeholders, and (5) experiences from others. Such aspects can be associated with our factors: (1) and (5) with *personal experience*, (2) with *innovation level*, (3) with *customer expectations*, and (4) with *domain experts' opinion*. Regarding the context-specific factor *innovation level*, the closest we found in [5] was the high confidence of the participants in using data-driven decision-making to identify business opportunities.

In conclusion, we claim that using data is relevant but not sufficient for making managerial decisions in software projects. Past studies have discussed the presence of human factors in data-driven decision-making. For example, Minku et al. [41] discussed the value of experts' knowledge, encouraging the involvement of software engineers in the development of data mining models. Among the authors' recommendations for engaging practitioners is the collection of data from experts: not only collecting experts' knowledge "through meetings, interviews, and surveys", but also using decision-support tools through which practitioners "can organize their tasks, visualize data, record a diary of decisions, etc." and data miners "can collect data on software engineering experts decisions" [41].

C. IMPLICATIONS FOR RESEARCH AND PRACTICE

Our study has implications for both research and practice. For research, our results demonstrate that practitioners have diverse information needs, some of which are context-dependent. While some use cases are well-saturated (i.e., legitimized by many instances in the data), others require further exploration due to fewer instances. Moreover, we discovered that practitioners' needs in our case are not orthogonal, suggesting that studying their relationships is a fruitful area for future research. Therefore, we encourage additional similar studies to identify possible patterns across use cases and the underlying relationships between them. For practitioners, our results can provide input for the development of tools that deliver actionable information more connected to the real needs of managers. The list of use cases we identified in our study can be used as a comprehensive starting set to be considered when developing such tools.

Finally, our discussion points to the need for hybrid solutions for supporting managerial decision-making, combining data with expert knowledge. Developing such solutions is not new in software engineering research. For example, Bayesian networks have been heavily used to combine data with expert knowledge for solving many software engineering problems such as risk management [42], process improvement [43], and effort estimation [44]. However, factors that hinder the adoption of existing solutions for supporting managerial decision-making in practice are that they were developed for a specific context, not customizable, or relied on manual or semi-automatic data collection [45]. Thus, our data suggest that a way forward in adopting data-driven managerial decision-making is by developing solutions (e.g., tools, guidelines, and methods) to facilitate the development and use of hybrid models by practitioners, such as the one presented by Manzano et al. [45].

VII. THREATS TO VALIDITY

This section discusses the threats to validity in terms of *construct validity*, *internal validity*, *external validity*, and *reliability* [25]. Table 9 summarizes the strategies employed to mitigate each of the validity threats.

Construct validity: As our study was conducted in the context of software analytics (a data-driven approach), we organized a material describing the research context to mitigate misconceptions between interviewees and researchers. Before the interview, we ensured that all participants had read the material and were given the opportunity to express any doubts they had regarding the topic. We sent individual reports via email to obtain their feedback, which we incorporated into our analysis.

The different terminology used among papers and the lack of details in describing some important aspects of the related studies represented a challenge when integrating our results into the existing body of knowledge. Our analysis was based on the descriptions in the papers so to make a semantic correspondence between them and our use cases/factors for managerial decision-making. Also, the process of mapping the use cases to software life cycle processes was based on the descriptions of each process' activities and tasks from ISO 12207. We associated them with the descriptions of our use cases supported by the interviewees' quotations. However, our interpretation may have impacted the results.

Internal validity: The selection of participants represents an internal validity threat. Our champion helped select knowledgeable practitioners actively involved in the organization's data-driven managerial decision-making process, mitigating the risk of interviewees having an incomplete or inaccurate understanding of the topic due to a lack of expertise.

External validity: To ensure that our results are not only relevant to our organization but also useful for others, we took care to describe the context of the study so that our findings can contribute to deriving a body of knowledge that helps practitioners determine what to apply in their contexts. It should also be noted that the knowledge elicitation and data analysis techniques employed in our study are equally applicable outside a software analytics context, thus showing a broader generalizability [27].

Reliability: We followed systematic procedures to guarantee the reliability of the evidence and minimize biased views. Based on the RQs, we prepared an interview script in advance. Interviews were semi-structured, recorded, and transcribed. We used researcher triangulation, well-established

TABLE 9. Provisions for securing trustworthiness of the study.

Criteria	Description	Threats	Mitigation strategies
Construct validity	The extent to which operational measures represent the concepts under study according to the research questions.	- Relevance of the case to address the research questions. - Rigor employed in data collection (e.g., misconceptions such as different interpretations of interview questions by researchers and interviewees).	- Context of the organization conducive to adopting software analytics to support managerial decision-making. - Provision of material describing the research context to mitigate misconceptions between interviewees and researchers. - Interviews recorded and transcribed. - Individual analysis results sent to the interviewees for feedback.
Internal validity	The extent to which other aspects/factors may influence the identified results.	- Interviewees' inaccurate view of software analytics.	- Selection of participants (practitioners in leadership positions involved in managerial decision-making). - Very experienced participants.
External validity	The extent to which the findings of the case study are of interest to other people outside the investigated case.	- Appropriateness of the case. - Representativeness of the case.	- Organization's openness to data-driven approaches, key practitioners participating in the interviews consistent with the research purpose. - Description of the study's context based on [33]. - Availability of artifacts for study replication.
Reliability	The extent to which the study can be replicated obtaining the same results.	- Measurement bias (e.g., reliability of the measurement instrument/raw data). - Researcher bias (e.g., researchers' preconceptions in data collection/analysis and inappropriate use of analysis methods). - Participant bias (e.g., subject's subjectivity, willingness to provide reliable data).	- Interview script designed in accordance with the RQs. - Interviews recorded and transcribed. - Researcher triangulation. - Well-established coding techniques and tool support for data analysis (MAXQDA). - CCM helped saturate categories. - Practitioners' motivation to participate in the interviews. - Individual interviews and guarantee of anonymity.

coding techniques, and the CCM to saturate categories. Given that the information needs and factors for managerial decision-making are extensively based on tacit knowledge, we believe that the practitioners participating in the study are the best source of knowledge in our case. Additionally, we ensured that all responses were kept anonymous.

VIII. CONCLUSION

With the large amount of data produced by software engineering activities, organizations have taken steps toward data-driven approaches to support decision-making. However, there has been a scarcity of studies investigating the practitioners' view of making decisions through such approaches, especially those in leadership positions. In this study, we thus collected the needs and perceptions of practitioners involved in data-driven managerial decision-making from one software organization.

We identified 19 software analytics use cases mapped to *organizational project-enabling* and *technical management* processes from ISO 12207 and provided indicators to meet them. We also identified 6 factors affecting managerial decision-making, which we clustered around three dimensions: *project-related*, *human-related*, and *context-specific factors*. Such factors, together with the different aspects mentioned by our participants, not only confirm the literature findings but also provide new input on how managerial decisions are made, given the gap that still exists on the topic. We also elicited organizational aspects helping and hindering the adoption of software analytics.

As future research, we encourage new studies like the one presented herein so that, at some point, as a community, we can achieve a consolidated body of knowledge on the data-driven decision-making process. With our findings, we intend to cooperate with leaders from our target organization to develop tools to facilitate the adoption of software analytics for managerial decision-making.

APPENDIX A
INTERVIEW SCRIPT

A. INTRODUCTION

We are conducting a set of interviews with different stakeholders from the organization to elicit relevant use cases for these professionals in the context of a software analytics tool. Our objective in conducting this interview is to identify your main information needs regarding the use of data to understand aspects related to software development and support the decision-making process. We appreciate your participation in this activity. This interview will be recorded and transcribed (with your consent), being used as an anonymous data collection instrument. At any time, you can ask to stop recording. All information provided by you will be treated as confidential and published only with the consent of the organization. This interview should last no longer than 1 hour. If necessary, we will ask the participant for extra time (max. 30 minutes). We will ask participants to provide information related to their role/work as well as their key needs that could be met by a software analytics solution in the context of the organization.

B. SOFTWARE ANALYTICS OVERVIEW

At this point, we consider it important to provide an overview of software analytics to contextualize the participant with respect to what will be asked. [After the overview] Any questions before we start?

C. INTERVIEW QUESTIONS

A NOTE FOR INTERVIEWER
The questions in this interview are divided into the following sections:
Initial questions: demographic questions aimed at gathering information about the responsibilities of the interviewee in the context of the organization and in the software industry in general. (5 min)
Practitioners' information needs, indicators, and decision-making process: identification of the current state regarding the use of practices similar to software analytics, as well as the main information needs that could be met with the effective implementation of a software analytics solution in the organization; questions about indicators and the decision-making process. (45 min)
Final questions: closing the interview; interviewee's considerations on any missing topic deemed relevant to this research. (10 min)

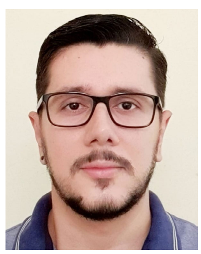
1) Initial questions
   - Q1.1: Could you tell us about your experience?
     - How long have you been working on projects in the organization?
     - What is your current role/fellowship? How long have you been in this role?
     - How long have you been working on software projects? What experiences in other organizations have you had? Same industry and domain?
     - What were your previous roles/positions?
2) Practitioners' information needs, indicators, and decision-making process
   - Q2.1: Considering the overview of software analytics we provided, can you identify any similar practices in the organization? If so, what problems do you seek to solve with this approach? How are these problems solved?
   - Q2.2: How reliable (or effective) do you consider the current approach being used in the organization? What limitations or bottlenecks could you identify?
   - Q2.3: When it comes to data-driven decision-making, what are your main needs, and what motivates them? In other words, what do you need to observe/monitor (what questions do you need answers to) and why?
   - Q2.4: Could you describe situations or scenarios in which a software analytics approach would be valuable for your work? Try to highlight what questions a software analytics approach could help answer, and what decisions such an approach could support.
   - Q2.5: For each scenario described in the previous question, what indicators (metrics) are needed to answer your questions or support your decisions? Why do you consider them necessary?
   - Q2.6: Are the data needed to obtain these indicators available (and in appropriate format) in the organization? If not, what are the main difficulties related to data availability and format? What should be done in your opinion to solve this problem?
   - Q2.7: What factors do you consider important for decision-making in the context of your role?
   - Q2.8: How does the data-driven decision-making process take place in the organization? How would you break it down into steps, main activities of each step, inputs, and outputs? You can exemplify narrating the events.
   - Q2.9: Do you consider that the decision-making process is basically the same (steps, inputs, outputs), or does it vary by project or another factor? Explain.
3) Final Questions
   - Q3.1: Considering a data-driven approach such as software analytics, what aspects of the organization do you think favor the implementation of this type of approach? What aspects make such an implementation difficult?
   - Q3.2: Is there anything related to the topic of the interview that we missed and you would like to comment on?

We would like to thank you for your willingness to participate in this activity. After the interview analysis, we will send you a summary of the findings so that you can identify any inconsistency. We hope you can get us feedback within one week. If you wish, we can also send you the full transcript of the interview.

REFERENCES

[1] R. P. L. Buse and T. Zimmermann, "Information needs for software development analytics," in *Proc. 34th Int. Conf. Softw. Eng. (ICSE)*, Jun. 2012, pp. 987–996.
[2] I. Figalist, C. Elsner, J. Bosch, and H. H. Olsson, "Breaking the vicious circle: Why AI for software analytics and business intelligence does not take off in practice," in *Proc. 46th Euromicro Conf. Softw. Eng. Adv. Appl. (SEAA)*, Aug. 2020, pp. 5–12.
[3] R. P. L. Buse and T. Zimmermann, "Analytics for software development," in *Proc. FSE/SDP Workshop Future Softw. Eng. Res.*, Nov. 2010, pp. 77–80.
[4] D. Zhang, Y. Dang, J.-G. Lou, S. Han, H. Zhang, and T. Xie, "Software analytics as a learning case in practice: Approaches and experiences," in *Proc. Int. Workshop Mach. Learn. Technol. Softw. Eng.*, Nov. 2011, pp. 55–58.
[5] R. B. Svensson, R. Feldt, and R. Torkar, "The unfulfilled potential of data-driven decision making in agile software development," in *Proc. Int. Conf. Agile Softw. Develop.* Cham, Switzerland: Springer, 2019, pp. 69–85.
[6] J. Choma, E. M. Guerra, and T. S. Da Silva, "Patterns for implementing software analytics in development teams," in *Proc. 24th Conf. Pattern Lang. Programs*, 2017, pp. 1–12.
[7] L. Guerrouj, O. Baysal, D. Lo, and F. Khomh, "Software analytics: Challenges and opportunities," in *Proc. IEEE/ACM 38th Int. Conf. Softw. Eng. Companion (ICSE-C)*, May 2016, pp. 902–903.
[8] J. Caldeira, F. E. B. Abreu, J. Cardoso, R. Simóes, T. Oliveira, and J. P. dos Reis, "Software development analytics in practice: A systematic literature review," *Arch. Comput. Methods Eng.*, vol. 30, pp. 2041–2080, Jan. 2023.

[9] J. A. O. G. da Cunha, F. Q. B. da Silva, H. P. de Moura, and F. J. S. Vasconcellos, "Decision-making in software project management: A qualitative case study of a private organization," in *Proc. IEEE/ACM Cooperat. Hum. Aspects Softw. Eng. (CHASE)*, May 2016, pp. 26–32.
[10] T. Rique, E. Dantas, M. Perkusich, K. Gorgônio, H. Almeida, and A. Perkusich, "Empirically derived use cases for software analytics," in *Proc. Int. Conf. Softw., Telecommun. Comput. Netw. (SoftCOM)*, Sep. 2022, pp. 1–6.
[11] T. Rique, E. Dantas, M. Perkusich, K. Gorgônio, H. Almeida, and A. Perkusich, "Use cases for software development analytics: A case study," in *Proc. 36th Brazilian Symp. Softw. Eng.*, Oct. 2022, pp. 388–397.
[12] *Systems and Software Engineering Software Life Cycle Processes*, Standard ISO/IEC/IEEE 12207:2017(E), International Organization for Standardization, Geneva, Switzerland, 2017. [Online]. Available: https://www.iso.org/standard/63712.html
[13] A. Nizam, "Software project failure process definition," *IEEE Access*, vol. 10, pp. 34428–34441, 2022.
[14] S. Martínez-Fernández, A. M. Vollmer, A. Jedlitschka, X. Franch, L. López, P. Ram, P. Rodríguez, S. Aaramaa, A. Bagnato, M. Choras, and J. Partanen, "Continuously assessing and improving software quality with software analytics tools: A case study," *IEEE Access*, vol. 7, pp. 68219–68239, 2019.
[15] J. T. Biehl, M. Czerwinski, G. Smith, and G. G. Robertson, "FASTDash: A visual dashboard for fostering awareness in software teams," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, Apr. 2007, pp. 1313–1322.
[16] S. Phillips, G. Ruhe, and J. Sillito, "Information needs for integration decisions in the release process of large-scale parallel development," in *Proc. ACM Conf. Comput. Supported Cooperat. Work*, Feb. 2012, pp. 1371–1380.
[17] A. Begel and T. Zimmermann, "Analyze this! 145 questions for data scientists in software engineering," in *Proc. 36th Int. Conf. Softw. Eng.*, May 2014, pp. 12–23.
[18] C. Treude, F. F. Filho, and U. Kulesza, "Summarizing and measuring development activity," in *Proc. 10th Joint Meeting Found. Softw. Eng.*, Aug. 2015, pp. 625–636.
[19] A. Al-Nayeem, K. Ostrowski, S. Pueblas, C. Restif, and S. Zhang, "Information needs for validating evolving software systems: An exploratory study at Google," in *Proc. IEEE Int. Conf. Softw. Test., Verification Validation (ICST)*, Mar. 2017, pp. 544–545.
[20] L. Pascarella, D. Spadini, F. Palomba, M. Bruntink, and A. Bacchelli, "Information needs in contemporary code review," *Proc. ACM Hum.-Comput. Interact.*, vol. 2, pp. 1–27, Nov. 2018.
[21] R. Berntsson Svensson and M. Taghavianfar, "Toward becoming a data-driven organization: Challenges and benefits," in *Proc. Int. Conf. Res. Challenges Inf. Sci.* Cham, Switzerland: Springer, 2020, pp. 3–19.
[22] M. Berndtsson, C. Lennerholt, T. Svahn, and P. Larsson, "13 Organizations' attempts to become data-driven," *Int. J. Bus. Intell. Res.*, vol. 11, no. 1, pp. 1–21, Jan. 2020.
[23] M. L. DRURY-GROGAN and O. O'DWYER, "An investigation of the decision-making process in agile teams," *Int. J. Inf. Technol. Decis. Making*, vol. 12, no. 6, pp. 1097–1120, Nov. 2013.
[24] C. Wohlin and A. Aurum, "Towards a decision-making structure for selecting a research design in empirical software engineering," *Empirical Softw. Eng.*, vol. 20, no. 6, pp. 1427–1455, Dec. 2015.
[25] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Softw. Eng.*, vol. 14, no. 2, pp. 131–164, Apr. 2009.
[26] P. Runeson, M. Höst, A. Rainer, and B. Regnell, *Case Study Research in Software Engineering: Guidelines and Examples*. Hoboken, NJ, USA: Wiley, 2012.
[27] P. Rodríguez, E. Mendes, and B. Turhan, "Key Stakeholders' value propositions for feature selection in software-intensive products: An industrial case study," *IEEE Trans. Softw. Eng.*, vol. 46, no. 12, pp. 1340–1363, Dec. 2020.
[28] M. Senapathi, J. Buchan, and H. Osman, "DevOps capabilities, practices, and challenges: Insights from a case study," in *Proc. 22nd Int. Conf. Eval. Assessment Softw. Eng.*, Jun. 2018, pp. 57–67.
[29] M. Shahin and M. A. Babar, "On the role of software architecture in DevOps transformation: An industrial case study," in *Proc. Int. Conf. Softw. Syst. Processes*, Jun. 2020, pp. 175–184.
[30] M. Hennink and B. N. Kaiser, "Sample sizes for saturation in qualitative research: A systematic review of empirical tests," *Social Sci. Med.*, vol. 292, Jan. 2022, Art. no. 114523.
[31] L. Briand, D. Bianculli, S. Nejati, F. Pastore, and M. Sabetzadeh, "The case for context-driven software engineering research: Generalizability is overrated," *IEEE Softw.*, vol. 34, no. 5, pp. 72–75, Sep. 2017.
[32] C. Urquhart, *Grounded Theory for Qualitative Research: A Practical Guide*. Newbury Park, CA, USA: Sage, 2022.
[33] K. Petersen and C. Wohlin, "Context in industrial software engineering research," in *Proc. 3rd Int. Symp. Empirical Softw. Eng. Meas.*, Oct. 2009, pp. 401–404.
[34] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative Res. Psychol.*, vol. 3, no. 2, pp. 77–101, Jan. 2006.
[35] D. S. Cruzes and T. Dyba, "Recommended steps for thematic synthesis in software engineering," in *Proc. Int. Symp. Empirical Softw. Eng. Meas.*, Sep. 2011, pp. 275–284.
[36] E. Kupiainen, M. V. Mäntylä, and J. Itkonen, "Using metrics in agile and lean software development—A systematic literature review of industrial studies," *Inf. Softw. Technol.*, vol. 62, pp. 143–163, Jun. 2015.
[37] K. V. J. Padmini, H. M. N. D. Bandara, and I. Perera, "Use of software metrics in agile software development process," in *Proc. Moratuwa Eng. Res. Conf. (MERCon)*, Apr. 2015, pp. 312–317.
[38] M. Perkusich, L. C. e Silva, A. Costa, F. Ramos, R. Saraiva, A. Freire, E. Dilorenzo, E. Dantas, D. Santos, K. Gorgônio, H. Almeida, and A. Perkusich, "Intelligent software engineering in the context of agile software development: A systematic literature review," *Inf. Softw. Technol.*, vol. 119, Mar. 2020, Art. no. 106241.
[39] W. Maalej, M. Nayebi, T. Johann, and G. Ruhe, "Toward data-driven requirements engineering," *IEEE Softw.*, vol. 33, no. 1, pp. 48–54, Jan. 2016.
[40] P. Ram, P. Rodriguez, M. Oivo, and S. Martínez-Fernández, "Success factors for effective process metrics operationalization in agile software development: A multiple case study," in *Proc. IEEE/ACM Int. Conf. Softw. Syst. Processes (ICSSP)*, May 2019, pp. 14–23.
[41] L. L. Minku, E. Mendes, and B. Turhan, "Data mining for software engineering and humans in the loop," *Prog. Artif. Intell.*, vol. 5, no. 4, pp. 307–314, Nov. 2016.
[42] N. Fenton, W. Marsh, M. Neil, P. Cates, S. Forey, and M. Tailor, "Making resource decisions for software projects," in *Proc. 26th Int. Conf. Softw. Eng.*, May 2004, pp. 397–406.
[43] M. Perkusich, G. Soares, H. Almeida, and A. Perkusich, "A procedure to detect problems of processes in software development projects using Bayesian networks," *Expert Syst. Appl.*, vol. 42, no. 1, pp. 437–450, Jan. 2015.
[44] S. Dragicevic, S. Celar, and M. Turic, "Bayesian network model for task effort estimation in agile software development," *J. Syst. Softw.*, vol. 127, pp. 109–119, May 2017.
[45] M. Manzano, C. Ayala, C. Gómez, A. Abherve, X. Franch, and E. Mendes, "A method to estimate software strategic indicators in software development: An industrial application," *Inf. Softw. Technol.*, vol. 129, Jan. 2021, Art. no. 106433.



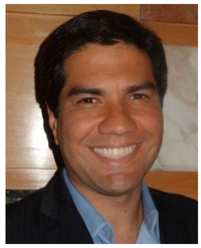
THIAGO RIQUE received the B.Sc. degree in computer science and the M.Sc. degree in informatics from the Federal University of Paraíba (UFPB), Brazil, in 2009 and 2011, respectively. He is currently pursuing the Ph.D. degree in computer science with the Federal University of Campina Grande (UFCG). From 2014 to 2021, he was a Professor with the Federal Rural University of the Semi-Arid Region (UFERSA). Since 2021, he has been a Professor with the Federal Institute of Paraíba (IFPB). He is also a member of the Intelligent Software Engineering Research Group (ISE/VIRTUS). His current research interest includes applying intelligent techniques to support software engineering.

MIRKO PERKUSICH received the Ph.D. degree in computer science. He is currently the Research Manager of VIRTUS, leading the Intelligent Software Engineering Research Group. His current research interest includes applying intelligent techniques, including recommender systems, to solve complex software engineering problems.

EMANUEL DANTAS received the Ph.D. degree in computer science from the Federal University of Campina Grande, Paraíba, Brazil, in 2021. He has been a Professor with the Federal Institute of Paraíba, since 2015. He is currently a Researcher with the Intelligent Software Engineering Research Group, VIRTUS, which is a Research, Development, and Innovation Center in Information Technology. His current research interest includes artificial intelligence applied to software engineering to solve complex problems. In the software engineering field, the main topics of interest are software project management, agile software development, effort estimation, risk management, and others.

DANYLLO ALBUQUERQUE received the M.Sc. degree in informatics from the Federal University of Paraíba, Brazil, in 2013. He is currently pursuing the Ph.D. degree in computer science with the Federal University of Campina Grande. He has been a Professor with the Federal Institute of Paraíba, since 2020. He is also a Systems Analyst with the Federal University of Campina Grande. He is also a member of the Intelligent Software Engineering Research Group, VIRTUS, which is a Research, Development, and Innovation Center in Information Technology. His current research interests include software quality, technical debt, software architecture, artificial intelligence, and intelligent techniques applied to solve complex software engineering problems.

KYLLER GORGÔNIO received the Graduate and master's degrees in computer science from Universidade Federal da Paraíba, in 1999 and 2001, respectively, and the Ph.D. degree in software from Universitat Politècnica de Catalunya, in 2010. He is currently a Professor with Universidade Federal de Campina Grande. He has experience in computer science focusing on software engineering. His research interests include petri nets, protocols, asynchronous communication mechanisms, colored petri nets, and model checking.



HYGGO ALMEIDA received the M.Sc. degree in computer science and the Ph.D. degree in electrical engineering from the Federal University of Campina Grande (UFCG), in 2004 and 2007, respectively. He has been a Professor with the Computer and Systems Department, UFCG, since 2006. He is currently the Head of the Intelligent Software Engineering Group, VIRTUS Innovation Center (VIRTUS/UFCG). He has over 20 years of teaching experience in the university and training courses for industry in the context of software engineering. He has more than 200 papers published. His current research interest includes applying intelligent techniques to solve complex software engineering problems.

ANGELO PERKUSICH (Member, IEEE) received the master's and Ph.D. degrees in electrical engineering from the Federal University of Paraíba, in 1987 and 1994, respectively. He has been a Professor with the Electrical Engineering Department (DEE), Federal University of Campina Grande (UFCG), since 2002. He was a Visiting Researcher with the Department of Computer Science, University of Pittsburgh, PA, USA, from 1992 to 1993, and developed research activities on software engineering and formal methods. He is currently a Principal Investigator of research projects financed by public institutions, such as Brazilian Agency for Research and Studies (FINEP) and Brazilian National Research Council (CNPq), and private companies. He is the Founder and the Director of the VIRTUS Innovation Center and the Embedded and Pervasive Computing Laboratory. His focus on research projects is on formal methods, embedded systems, mobile pervasive and ubiquitous computing, and software engineering. He has over 30 years of teaching experience in the university and training courses for industry in the context of software for real-time systems, software engineering, embedded systems, computer networks, and formal methods. His research interests include embedded systems, software engineering, mobile pervasive computing, and formal methods, with more than 300 papers published, and 80 master thesis and 21 doctoral dissertations advised.