

The influence of critical success factors in the implementation of an environment for the management and development of agile software

A influência de fatores críticos de sucesso na implementação de um ambiente de gestão e desenvolvimento de software ágil

- Vitor Junior de Almeida Menandro Barboza** Mestre em Sistemas de Gestão. Universidade Federal Fluminense (UFF) – Brasil. E-mail. vitor.jr.rj.2010@gmail.com
<https://orcid.org/0000-0003-0997-7662>
- Mirian Picinini Méxas** Doutora em Engenharia Civil. Universidade Federal Fluminense (UFF) – Brasil. E-mail. mirian_mexas@id.uff.br
<https://orcid.org/0000-0003-4506-7009>
- Priscilla Cristina Cabral Ribeiro** Doutora em Engenharia de Produção. Universidade Federal do Rio de Janeiro (UFRJ) – Brasil. E-mail. priscillaribeiro@id.uff.br
<https://orcid.org/0000-0003-0824-9268>
- Geisa Meirelles Drumond** Doutora em Sistemas de Gestão Sustentáveis. Universidade Federal Fluminense (UFF) – Brasil. E-mail. Gmdrumond@id.uff.br
<https://orcid.org/0000-0002-0605-2341>

ABSTRACT

This research study aims to analyze, rank, and validate the influence of Critical Success Factors (CSFs) on the implementation of agile software development environments, based on the perception of Brazilian stakeholders. First, a literature review was conducted, through which 21 CSFs were identified and grouped into three categories: Personnel, Organizational, and Technical/Process. Subsequently, a survey was administered to experts in software management and development, resulting in a sample of 101 respondents. Based on the survey results, a ranking of the factors influencing the adoption of agile methods was established, with a particular focus on the quality of the final software product. The findings revealed that Personnel factors were considered the most important, followed by Technical/Process and Organizational factors. The four highest-ranked CSFs were "Competence, Training, and Specialization", "Teamwork", "Communication Skills", and "Focus on Client Satisfaction and Engagement", all of which belong to the Personnel category. This research highlights the importance of investing in the Personnel dimension to enhance the management and development of high-quality software. Furthermore, the identified CSFs can assist organizations in improving their software development processes through the effective adoption of agile methods, ultimately contributing to the delivery of higher-quality software products.

Keywords: Agile methodologies. Quality. Management of software. Development of software. Critical success factors.

RESUMO

Este estudo tem como objetivo analisar, classificar e validar a influência dos Fatores Críticos de Sucesso (FCSs) na implementação de ambientes de desenvolvimento ágil de software, com base na percepção de stakeholders brasileiros. Inicialmente, foi realizada uma revisão da literatura, por meio da qual foram identificados 21 FCSs, posteriormente agrupados em três categorias: Pessoas, Organizacional e Técnico/Processo. Em seguida, foi aplicada uma pesquisa com especialistas em gestão e desenvolvimento de software, resultando em uma amostra de 101 respondentes. Com base nos resultados obtidos, foi elaborado um ranking dos fatores que influenciam a adoção de métodos ágeis, com foco especial na qualidade do produto final de software. Os resultados revelaram que os fatores relacionados à categoria Pessoas foram considerados os mais importantes, seguidos pelos fatores Técnico/Processo e Organizacional. Os quatro FCSs mais bem classificados foram "Competência, Capacitação e Especialização", "Trabalho em Equipe", "Habilidades de Comunicação" e "Foco na Satisfação e no Engajamento do Cliente", todos pertencentes à categoria Pessoas. Esta pesquisa destaca a importância de investir na dimensão Pessoas para aprimorar a gestão e o desenvolvimento de software de alta qualidade. Além disso, os FCSs identificados podem auxiliar as organizações a melhorar seus processos de desenvolvimento de software por meio da adoção eficaz de métodos ágeis, contribuindo, em última instância, para a entrega de produtos de software de maior qualidade.

Palavras-chave: Metodologias ágeis. Qualidade. Gestão de software. Desenvolvimento de software. Fatores críticos de sucesso.

Received on April 28, 2024. Approved on April 28, 2026. Evaluated through a double-blind peer review process.

Published in accordance with APA guidelines.

<https://doi.org/10.22279/navus.v18.1915>

1 INTRODUCTION

When analyzing the work of current organizations, according to Tabares Quiroz *et al.* (2021), it is observed that both organizational and technological change occur simultaneously, and these changes have been increasingly rapid. The incorporation of Information Technology (IT) in organizations allows better access to information, facilitating access to information that was previously difficult to reach and, on the other hand, speeds up the time needed to obtain this information, ensuring its quick and timely retrieval and delivery (Díaz Pinzón *et al.*, 2017). The implementation of these technologies leads to an expectation, which is related to the conversion of digital media into an important factor in customer decision-making in view of the organizations' need for innovation, which will result in adding value to the service provided (Cruz Estrada & Miranda Zavala, 2019).

However, the complexity and size of computing systems required the use of software engineering techniques that aim to reduce costs and effort applied in the several stages of software development and maintenance (Pressman, 2010). In this direction, methodologies converge and contribute to the reuse of successful techniques, mechanisms and experiences (Oliveira, 2014). Among the several used methods, a group stands out: agile methodologies.

In general, agile methodologies follow some basic principles: short development cycles; small and constant deliveries; cooperative, with active participation and constant presence of the client during the development of the software; and, mainly, adaptable, where the occurrence of changes is acceptable throughout the development cycle (Costinhas & Santos, 2014). The use of these methodologies with adaptations of other traditional models in the academic environment allows the process to adjust to the needs of the company and, thus, achieve the improvement of the desired development process. Besides, the agile principle of delivering solutions will often increase support from top management and decrease resistance to change (Amorim *et al.*, 2021, p. 3), knowing that these changes have their limits. Schema-flexible databases have become popular among agile developers as they ease the burden of adapting the database every time the application code changes. However, this flexibility comes at the price of a certain structural entropy, i.e., a disarray of the database (Hillenbrand *et al.*, 2022). As pointed out by Souza *et al.* (2022, p. 96), the goal of agile methods is to achieve a better performance with both the project and the final product.

This adaptation should focus mainly on the quality of the final product delivered. Currently, the agile method is being used with a focus only on delivering the final product to the customer. The problem is that this scenario raises concerns regarding how agile practices are implemented and how they may influence the quality of the final software product, especially when key success factors are not adequately addressed.

In this context, factors such as the low quality of the software developed due to the inappropriate use of tests, lack of documentation, inadequate processes for the company and low use of tools adhering to the development process bring losses mainly in the maintenance of the final product and, consequently, in its quality. This maintenance brings future discomfort to the team, to the customer and to the board, that is, the interested parties in the process. The guarantee of quality, reliability, efficiency, safety and maintainability are metrics of the development of a project in large companies (Carosia, 2004). The use of agile methods in any project to ensure quality is essential. Currently, software development companies use agile methods in their projects to reduce the production cost with high quality of services and products (Neelu & Kavitha, 2021).

Therefore, to achieve a successful implementation of a management and software development environment, one of the techniques used is the identification of the main critical success

factors (CSFs) and their influence on the implementation of this environment, according to research performed by Huerta Riveros *et al.* (2015). However, there is a lack of empirical studies that quantitatively assess and rank the influence of CSFs on agile software development environments, particularly from the perspective of practitioners, which is the originality of the present study, which aims to analyze, rank, and validate the influence of CSFs on the implementation of agile software development environments, based on the perception of Brazilian stakeholders.

Thus, it is expected that the identification of the main CSF can help companies to avoid failures during the implementation of this environment with agile methods, as well as taking the necessary actions to improve this implementation by observing the CSF that most need attention throughout this process.

Besides this introduction, Section 2 presents information on agile methods. The third section describes the used methodology and the stages of the research study, as well as the foundation of the applied survey. The fourth one presents the result analysis of the survey and the ranking of CSFs, and lastly, in the fifth section, there are the conclusions of the work.

2 AGILE METHODS

Software Engineering seeks to balance multiple and often competing project objectives, including scope, cost, schedule, and quality. In this context, software development methodologies have evolved to support organizations in managing increasing levels of technological complexity and market uncertainty. Traditionally, software projects relied on predictive approaches characterized by extensive planning and detailed documentation. However, the growing need for flexibility, rapid delivery, and continuous adaptation has contributed to the widespread adoption of agile methodologies (Ayat *et al.*, 2021; Muhammad *et al.*, 2021).

Agile Software Development (ASD) emerged as a response to the limitations of traditional approaches in dynamic environments. According to Cockburn and Highsmith (2001), agile methods were designed to address contexts characterized by constant technological and business changes, where customer requirements evolve throughout the project lifecycle. Rather than emphasizing comprehensive upfront planning, agile approaches promote iterative development, continuous feedback, and incremental value delivery (Biesialska *et al.*, 2021; Sommerville, 2007).

The literature highlights several benefits associated with agile adoption, including increased responsiveness to changing requirements, enhanced customer involvement, improved collaboration among stakeholders, and faster delivery cycles (Fronza & Wang, 2021; Pacheco *et al.*, 2020; Šmite *et al.*, 2021). These characteristics have contributed to the growing diffusion of agile practices across software organizations. Nevertheless, the literature also emphasizes that agile methods shouldn't be viewed as universally applicable solutions. Their effectiveness depends on project characteristics, organizational context, team capabilities, and management practices (Conceição & Silveira, 2018). In other words, successful implementation cannot be explained solely by the adoption of agile practices. Therefore, understanding the factors that contribute to successful agile implementation remains a relevant research challenge.

Previous studies have identified numerous CSFs associated with agile software development. However, the existing literature presents these factors in a fragmented manner, emphasizing isolated dimensions. Consequently, there remains a need to integrate these factors into a coherent analytical framework capable of supporting both theoretical understanding and empirical investigation.

Based on the literature review, this study adopts a framework composed of CSFs that were derived from the recurring factors identified in previous research and grouped according to three complementary dimensions: Personnel Factors, Organizational Factors, and Technical/Process Factors. This categorization in three dimensions follows the perspective adopted by Abdalhamid and Mishra (2017), Aldahmash *et al.* (2017), and Chiyangwa and Mnkandla (2017), who argue that successful agile adoption depends on the interaction between human, organizational and technical elements. The proposed framework establishes the theoretical basis for the empirical investigation presented in the subsequent sections and provides a structured approach for evaluating the relative importance of the identified CSFs.

3 RESEARCH METHODOLOGY

We conducted the present study by following the subsequent stages:

First stage: Aiming to provide the theoretical support to the study, we conducted a literature review, using the SCOPUS and SCIELO databases to select articles, which are relevant databases of qualified and reviewed scientific journals. We accessed the databases through the Capes Journals Portal between 2018 and 2022. Lastly, we searched the Web through the Google Scholar, and also through books, dissertations and theses that deal with the theme of this work. The search methodology of the SCOPUS and SCIELO databases was based on Costa (2010).

For the searches of the SCOPUS and SCIELO databases, we used the following terms about the theme in an isolated way: "Agile Method"; "Software Quality" and "Software Engineering". These terms were selected because of their connection with the theme of this work: CSFs in development of agile software. Initially, the referred search aimed to identify and investigate the overall number of publications in the mentioned databases, with search filter restrictions to streamline the search results. The filters were: Years: 2014 to 2022. Areas: Computer Science; Business, Management and Accounting; and Engineering; Document type: Article, Article in Press and Review.

The verification of the records found in the databases with the three keywords researched individually evidenced varied themes about the subjects Agile Method, Software Quality and Software Engineering, for example: Scalability of Agile Methods; Adopting the Principles of Agile Methods, Acceptance of Agile Methods; Quality of Algorithms; Models of Quality; Life Cycle of Software; Production Processes, among others.

Due to high dispersion and variety of the found records related to the context of this research study, an additional search strategy was adopted to increase the specificity of the results. The term "Critical Success Factors" was combined with the previously defined keywords in order to identify studies directly related to the factors influencing the quality of software developed using agile methods.

After applying the inclusion criteria (peer-reviewed articles, alignment with the research topic, publication between 2014 and 2022, and availability of full text) and the exclusion criteria (duplicate records, studies unrelated to agile software development, and publications that didn't address Critical Success Factors), 56 records were identified. Subsequently, the titles and abstracts of these studies were screened for relevance. The process resulted in the selection of eight articles for full-text analysis.

To complement the review, the references cited in the selected articles, which are highly relevant to the topic, were also considered. This supplementary search resulted in the inclusion of five additional studies that met the same eligibility criteria. Therefore, the final bibliographic portfolio comprised 13 articles, which constituted the theoretical basis for the identification and categorization of the Critical Success Factors analyzed in this study.

The search strings, databases, inclusion and exclusion criteria, and screening procedures were defined a priori to improve the transparency and reproducibility of the literature review process.

Second stage: With the selected articles, we analyzed the Critical Success Factors that were most mentioned by the authors regarding the final quality of software product developed with agile methods, by addressing such factors in this study.

Third stage: The selected Critical Success Factors were then grouped in three categories (Personnel, Organizational and Technical/Process). The authors Abdalhamid and Mishra (2017), Aldahmash *et al.* (2017), and Chiyangwa and Mnkandla (2017) addressed this categorization.

Fourth stage: Based on the Critical Success Factors identified in the literature review, a structured questionnaire was developed and applied to 101 respondents responsible for software development, from project managers, through analysts to developers. The sample of 101 respondents, obtained through convenience sampling due to the lack of a clearly defined target population, is acknowledged as a limitation of this study. To support the analysis, a ranking approach based on a Likert-scale responses and a scoring criterion ($VH + H - L - NI$) was employed.

Before sending the questionnaire, a pre-test stage was carried out with five experts in software development, to validate the questions, options of answers and streamline the questionnaire.

Table 1 presents the categories proposed based on the bibliographic review in databases and the other researched references and related to the authors who mentioned them.

Table 1

Theoretical Foundation of the Survey

Question	Author(s)
<i>Questions in the Category Personnel:</i>	
"Answer the items according to your perception of the category/theme People Management. What is the necessity for:"	
1. The company to invest in training, competence and specialization CSF1: Competence, training and specialization	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2018); Medeiros <i>et al.</i> (2015); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Abdalhamid and Mishra (2017); Pekki (2016); Mohanarajah and Jabar (2015); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)
2. Top management support for the deployment of new methodologies CSF2: Top management support	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2018); Medeiros <i>et al.</i> (2015); Digalwar <i>et al.</i> (2014); Abdalhamid and Mishra (2017); Pekki (2016); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)
3. Motivation and acknowledgement of the good work performed CSF3: Motivation and acknowledgement of good work	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2015); Digalwar <i>et al.</i> (2014); Chiyangwa and Mnkandla (2017)
4. Having communication skills to work and engage the client of the software project CSF4: Communication skills	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2018); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Abdalhamid and Mishra (2017); Gren <i>et al.</i> (2018); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)
5. The whole team to work together and be	Ghayyur <i>et al.</i> (2018); Digalwar <i>et al.</i>

committed to the final goal of the work CSF5: Teamwork	(2014); Silva and Santos (2015); Abdalhamid and Mishra (2017); Gren <i>et al.</i> (2018); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)
6. Accepting group or individual changes in order to reach improvements in the company CSF6: Group/individual resistance	Ghayyur <i>et al.</i> (2018); Silva and Santos (2015); Abdalhamid and Mishra (2017); Chiyangwa and Mnkandla (2017); Dikert <i>et al.</i> (2016)
<i>Questions in the Category Organization:</i>	
"Answer the items according to your perception of the category/theme Organization. What is the necessity for:"	
7. Cooperative cultural practice CSF7: Cooperative organizational culture	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2018); Medeiros <i>et al.</i> (2015); Silva and Santos (2015); Abdalhamid and Mishra (2017); Mohanarajah and Jabar (2015); Gren <i>et al.</i> (2018); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)
8. Human resources management practice CSF8: Human resources management	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2015); Digalwar <i>et al.</i> (2014); Abdalhamid and Mishra (2017)
9. Establishment of project management norms and procedures CSF9: Project management norms and procedures	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2018); Aldahmash <i>et al.</i> (2017); Pekki (2016); Mohanarajah and Jabar (2015); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)
<i>Questions in the Category Software Techniques and Processes:</i>	
"Answer the items according to your perception of the category/theme Software Techniques and Processes in big and widely documented projects through the use of waterfall models like RUP" and "Answer the items according to your perception of the category/theme Software Techniques and Processes in projects through the use of agile methods, like Scrum or XP"	
10. Satisfaction level of client's requirements CSF10: Focus, client's satisfaction and engagement	Medeiros <i>et al.</i> (2018); Aldahmash <i>et al.</i> (2017); Medeiros <i>et al.</i> (2015); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Mohanarajah and Jabar (2015); Gren <i>et al.</i> (2018); Tsoy and Staples (2020)
11. Probability the project will not overrun the client's budget CSF11: Poorly characterized risk planning	Ghayyur <i>et al.</i> (2018); Silva and Santos (2015); Gren <i>et al.</i> (2018); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020)
12. Capacity of the project to not miss the opportunity window or client's deadline CSF11: Poorly characterized risk planning	Ghayyur <i>et al.</i> (2018); Silva and Santos (2015); Gren <i>et al.</i> (2018); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020)
"Answer the items according to your perception of the category/theme Software Documentation. What is the necessity for:"	
13. Documentation of the project requirements, especially in big projects that require future maintenance CSF12: Efficient requirement gathering technique, traceability and scope definition.	Ghayyur <i>et al.</i> (2018); Aldahmash <i>et al.</i> (2017); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Gren <i>et al.</i> (2018); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)

14. Keeping the requirement traceability with the other project artifacts CSF12: Efficient requirement gathering technique, traceability and scope definition.	Ghayyur <i>et al.</i> (2018); Aldahmash <i>et al.</i> (2017); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Gren <i>et al.</i> (2018); Chiyangwa and Mnkandla (2017)
"Answer the items according to your perception of the category/theme Software Configuration Management. What is the necessity for:"	
15. The use of versioning tools, like SVN and Git CSF13: Tools support	Ghayyur <i>et al.</i> (2018); Digalwar <i>et al.</i> (2014); Silva and Santos (2015)
"Answer the items according to your perception of the category/theme Tests. What is the necessity for:"	
16. The Use of Test Plans, Unit Tests and Test Cases CSF14: Inspections / Reviews / Software validation	Aldahmash <i>et al.</i> (2017); Digalwar <i>et al.</i> (2014); Pekki (2016); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020)
17. The use of regression Tests, like the ones used in Selenium or JBehave CSF14: Inspections / Reviews / Software validation	Aldahmash <i>et al.</i> (2017); Digalwar <i>et al.</i> (2014); Pekki (2016); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020)
"Answer the items according to your perception of the category/theme Automation. What is the necessity for:"	
18. The use of facility, compilation and packaging management tools like Ant, Maven, Gradle, NPM CSF13: Tools support	Ghayyur <i>et al.</i> (2018); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Tsoy and Staples (2020)
19. The use of Continuous Integration tools, like Jenkins or Gitlab CI/CD CSF13: Tools support	Ghayyur <i>et al.</i> (2018); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Tsoy and Staples (2020)
20. The use of code analysis tools, like Firebugs and SonarQube CSF13: Tools support	Ghayyur <i>et al.</i> (2018); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Tsoy and Staples (2020)
"Answer the items according to your perception of the category/theme Project Management. What is the necessity for:"	
21. Evaluating the characteristics of the software project in order to choose the adequate management and development methods CSF15: Characteristics of the software project and adequate methodology	Aldahmash <i>et al.</i> (2017); Silva and Santos (2015); Abdalhamid and Mishra (2017); Mohanarajah and Jabar (2015); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020)
22. The use and deployment of agile processes CSF16: Deployment of agile development techniques	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2018); Aldahmash <i>et al.</i> (2017); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Mohanarajah and Jabar (2015); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)
23. Impacts related to changes in the planned tasks for iteration delivery CSF17: Altering iteration/ Story change	Ghayyur <i>et al.</i> (2018); Aldahmash <i>et al.</i> (2017); Silva and Santos (2015); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020); Dikert <i>et al.</i> (2016)
24. The use of task management tools, like Redmine, JIRA CSF13: Tools support	Ghayyur <i>et al.</i> (2018); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Tsoy and Staples (2020)
25. The use of frequent project follow-up meetings and team alignment meetings CSF16: Deployment of agile development	Ghayyur <i>et al.</i> (2018); Aldahmash <i>et al.</i> (2017); Digalwar <i>et al.</i> (2014); Silva and Santos (2015); Gren <i>et al.</i> (2018);

techniques	Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020)
26. Follow-up metrics to improve the delivery of software projects with quality CSF18: Metrics and Quality	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2015); Abdalhamid and Mishra (2017); Pekki (2016); Chiyangwa and Mnkandla (2017); Dikert <i>et al.</i> (2016)
27. Well-defined and well-known coding standards CSF19: Well-defined coding standards	Digalwar <i>et al.</i> (2014); Abdalhamid and Mishra (2017); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020)
28. Constant evaluation of the lessons learned for the continuous improvement of techniques and processes CSF20: Continuous improvement	Medeiros <i>et al.</i> (2015); Abdalhamid and Mishra (2017)
29. Simplicity in coding and software design CSF21: Simple Code and Design/Simplicity	Ghayyur <i>et al.</i> (2018); Medeiros <i>et al.</i> (2015); Digalwar <i>et al.</i> (2014); Chiyangwa and Mnkandla (2017); Tsoy and Staples (2020)

Based on Gray (2012), we elaborated the degree of influence of the CSFs and the value correspondences according to the following Likert scale:

Degree of influence of the Critical Success Factor: (5) – Very High (VH); (4) – High (H); (3) – Medium (M); (2) – Low (L); (1) – No Influence (NO); (0) – Do Not Know (DK).

Fifth stage: After data gathering through the survey, we treated and analyzed the data in order to verify whether the most mentioned critical success factors by authors in the literature review stage are the same ones that most influence the success of the quality of software developed with agile methods, according to the practical view of the professionals responsible for software development. The ranking of the most influential CSFs in the adoption of agile methods focused on the quality of the final product was then presented. We presented the frequency distributions of the researched CSFs, and then, we made observations in the light of the bibliographic review about the theme of the research study. We described this stage in the Section 4: Result Analysis.

Sixth stage: Finally, in the research consolidation, we described the conclusions and contributions towards the study.

4 RESULT ANALYSIS

The universe of this research study is the group of professionals responsible for the development of software in Brazil, encompassing the whole process, from project managers and analysts, to developers. Based on it, we obtained emails through the sites of technology companies, education institutions and contribution of respondents.

After the pre-test stage, we sent emails to the identified universe, with information about the research purposes and the importance of answering the survey. We conducted the field research and reached the sample of 101 respondents.

Based on the answers to the closed questions of the survey, we conducted the analysis of the gathered data and of the tabulated results, enabling data compilation. We presented the frequency distributions of the researched CSFs, and it was possible to identify convergences and gaps between the perceptions of the professionals in the sample and the authors of the bibliography used in this study. Such analyses form a relevant set of conclusions for the study.

4.1 Profile of Respondents

The research study shows that the highest number of respondents is located in the southeast region and the lowest in the north region of Brazil. In addition, some respondents live abroad. It is possible to highlight that 18% of the respondents are college graduate professionals, 56% hold Master's degrees, 21% hold doctor's degrees and 2% hold extension course degrees. Yet, the distribution of job positions of the respondents is: 35% are Project Managers or Coordinators, 23% are System Analysts, 6% are Programmers and 7% are Business Analysts. A high number stated that they have other positions—29%—the ones we identified as Engineers and Managers, among others.

We requested information about how long they have been in the profession: 9% have been for more than 25 years; 12% have been between 16 and 25 years; 31% have been between 10 and 15 years; 33% between 3 and 9 years and 15% for less than 3 years. We note that the majority of the participants of the survey are in the ranges 3 to 9 years and 10 to 15 years, with a total of 64.35%. The survey also shows that 57% of the respondents do not work in technology companies, that is, they are professionals that work in the IT department of companies in other segments.

Finally, in the profile stage, we asked how long the professional has been working in the current company: 6% for more than 25 years; 7% between 16 and 25 years in the company; 17% between 10 and 15 years; 46% between 3 and 9 years and 24% for less than 3 years. It is possible to verify a reality of the technology employee: the high turnover in companies, since 70% of the respondents have been working for a short time in the company, 9 years the most.

4.2 Perception Analysis of the CSFs

The tables that show the percentage of answers for each question present the results of the survey. Although the survey was administered to 101 respondents, the valid number of responses varies across questions because the category "Do Not Know" was excluded from the analysis, as indicated in the research design. Consequently, the totals reported in the tables represent only valid responses for each item, which explains the variation observed. This procedure was adopted to avoid distortions in the assessment of CSFs. Therefore, comparisons across items should consider differences in the number of valid responses.

The purpose is to present the main Critical Success Factors, grouped in categories (Personnel, Organizational and Technical/Process) and frequency of answers, in percentage terms.

4.2.1 Category Personnel

Table 2

Survey Answers for the Category Personnel

"Answer the items according to your perception of the category/theme Project Management. What is the necessity for:"											
	Very High		High		Medium		Low		No Influence		Total
The company to invest in training, competence and specialization CSF1: Competence, training and specialization	53.0%	53	38.0%	38	5.0%	5	4.0%	4	0.0%	0	100
Top management support for the deployment of new methodologies CSF2: Top management support	45.0%	45	45.0%	45	8.0%	8	2.0%	2	0.0%	0	100
Motivation and acknowledgement of the good work performed CSF3: Motivation and acknowledgement of good work	56.5%	57	30.7%	31	5.9%	6	6.9%	7	0.0%	0	101
4. Having communication skills to work and engage the client of the software project CSF4: Communication Skills	52.5%	52	35.4%	35	8.1%	8	3.0%	3	1.0%	1	99
The whole team to work together and be committed to the final goal of the work CSF5: Teamwork	61.3%	62	29.7%	30	5.0%	5	4.0%	4	0.0%	0	101
Accepting group or individual changes in order to reach improvements in the company CSF6: Group/individual resistance	39.3%	39	45.5%	45	9.1%	9	6.1%	6	0.0%	0	99

By analyzing the answers of the survey in Table 2, the results reinforce the central role of human factors in software project success. Teamwork, motivation, recognition, and professional competence received the highest levels of perceived influence, indicating that respondents associate project performance more strongly with interpersonal and organizational capabilities than with formal structures alone.

These findings are consistent with previous studies by Ghayyur *et al.* (2018), Medeiros *et al.* (2015), Digalwar *et al.* (2014), Silva and Santos (2015), Abdalhamid and Mishra (2017), Gren *et al.* (2018), and Chiyangwa and Mnkandla (2017), which emphasize collaboration, employee competence, stakeholder engagement, and team commitment as critical success factors in software development projects.

However, an interesting aspect emerges from the comparatively lower ratings attributed to top management support and resistance to change. While the literature presents management sponsorship as a prerequisite for organizational transformation, respondents appear to prioritize operational and team-level factors. This divergence may suggest that practitioners perceive project outcomes as being influenced more directly by day-to-day team dynamics than by executive involvement. Such results shouldn't be interpreted as evidence that these factors are unimportant, but rather as an indication that their contribution may be perceived differently from factors with immediate operational effects.

4.2.2 Category Organizational

Table 3

Survey Answers for the Category Organizational

"Answer the items according to your perception of the category/theme Organizational. What is the necessity for:"											
	Very High		High		Medium		Low		No Influence		Total
Cooperative cultural practice CSF7: Cooperative organizational culture	31%	31	53%	53	12%	12	3%	3	1%	1	100
Human resources management practice CSF8: Human resources management	37%	37	43%	43	14%	14	2%	2	4%	4	100
Establishment of project management norms and procedures CSF9: Project management norms and procedures	33%	33	44%	44	15%	15	5%	5	3%	3	100

Through the analysis of the gathered data in Table 3, the organizational culture factor emerged as the strongest element within this category, reinforcing the argument that agile adoption extends beyond technical implementation and requires alignment of organizational values and behaviors. The result is consistent with studies by Ghayyur *et al.* (2018), Medeiros *et al.* (2018), Medeiros *et al.* (2015), Silva and Santos (2015), Abdalhamid and Mishra (2017), Mohanarajah and Jabar (2015), Gren *et al.* (2018), and Chiyangwa and Mnkandla (2017) that present this CSF as the most important one for the success of the quality of agile methods projects in the category Organizational.

According to Aldahmash *et al.* (2017), the organizational culture can be defined as a set of organizational factors or variables that may influence the development of agile software in an organization. The organizational culture factor is a vital factor in the transition into agility. The organization should have a dynamic culture to respond to the different changes during the agile development cycle. Yet, Abdalhamid and Mishra (2017) mention that the adoption of agile methods by an organization requires a change in the culture and attitude of the organization. The agile idea is based on the way professionals must work to deliver new practices to team members and managers and, usually, agile methods affect the culture of the organization.

Nevertheless, the CSFs "Human resources management" and "Project management norms and procedures" had a relatively lower emphasis, if compared with other factors. The inferior number of votes for these factors is also explained by their lower number of mentions by the authors in the literature.

This finding contributes to the discussion by suggesting that organizational culture may function as an integrating factor that amplifies or constrains the effectiveness of other organizational practices. Consequently, investments in formal processes alone may not produce the expected outcomes if cultural conditions are not simultaneously addressed.

4.2.3 Category Technical/Process

Table 4

Answers regarding the Methodology for the Category Technical/Process

"Answer the items according to your perception of the category/theme: Software Techniques and Processes in big projects and widely documented projects through the use of waterfall methods like RUP"													
				Very High		High		Medium		Low		No Influence	Total
Satisfaction level of client's requirements				13.6%	11	29.6%	24	34.6%	28	16.0%	13	6.2%	81
CSF10: Focus, client's satisfaction and engagement													
Probability the project will not overrun the client's budget.				16.5%	13	21.5%	17	24.0%	19	31.7%	25	6.3%	79
CSF11: Poorly characterized risk planning													
Capacity of the project to not miss the opportunity window or client's deadline				16.0%	13	21.0%	17	32.1%	26	24.7%	20	6.2%	81
CSF11: Poorly characterized risk planning													
"Answer the items according to your perception of the category/theme: Software Techniques and Processes in agile methodology projects, like Scrum or XP"													
				Very High		High		Medium		Low		No Influence	Total
Satisfaction level of client's requirements				31.9%	29	50.5%	46	13.2%	12	1.1%	1	3.3%	91
CSF10: Focus, client's satisfaction and engagement													
Probability the project will not overrun the client's budget.				20.9%	18	27.9%	24	34.9%	30	12.8%	11	3.5%	86
CSF11: Poorly characterized risk planning													
Capacity of the project to not miss the opportunity window or client's deadline				18.8%	17	45.6%	41	25.6%	23	6.7%	6	3.3%	90
CSF11: Poorly characterized risk planning													

By evaluating the result of the two questions addressing the CSF "Poorly characterized risk planning", in Table 4, the comparison between traditional and agile approaches indicates that budget and schedule performance remain significant challenges regardless of the development methodology adopted. In reality, keeping within the budget and meeting the deadline of the project are difficulties found in all projects and approaches. "Risk planning" is an important CSF simply because it reduces alterations in the software project plan. Aldahmash *et al.* (2017) describe that it is essential to launch the project plan, which must be dimensioned correctly, during the initial stages of the agile development project. The finding suggests that project constraints associated with planning uncertainty and risk management are persistent issues that transcend methodological choices.

The notable distinction emerges in relation to customer satisfaction and engagement. Agile methodologies were perceived more positively in this regard, which is consistent with the literature emphasizing iterative delivery, continuous feedback, and closer customer involvement. However, the

findings shouldn't be interpreted as evidence of the superiority of agile methods in all contexts. Rather, they indicate that respondents associate agile practices with greater responsiveness to changing requirements and stakeholder expectations.

Table 5

Answers regarding Software Documentation for the Category Technical/Process

"Answer the items according to your perception of the category/theme Software Documentation. What is the necessity for:"												
		Very High		High		Medium		Low		No Influence		Total
Documentation of the project requirements, especially in big projects that require future maintenance. CSF12: Efficient requirement gathering technique, traceability and scope definition		54.7%	52	28.4%	27	11.6%	11	4.3%	4	1.0%	1	95
Keeping the requirement traceability with the other project artifacts. CSF12: Efficient requirement gathering technique, traceability and scope definition		38.4%	36	43.6%	41	10.6%	10	5.3%	5	2.1%	2	94

In Table 5, regarding software documentation, the results challenge the common perception that agile environments place limited value on documentation. Respondents clearly recognized the importance of documenting requirements and maintaining traceability, particularly in projects that require long-term maintenance. This suggests that practitioners perceive documentation not as a contradiction to agility, but as a complementary mechanism that assists in the maintenance and in the understanding of the software itself. It is important for the tests, and also for future maintenance, whether it be corrective or developmental.

According to Abdalhamid and Mishra (2017), agile methods are more appropriate for projects in which the requirements are not well known, because they accept changes without difficulties.

Table 6

Answers regarding Tool support and Inspections / Reviews / Software validation for the Category Technical/Process

"Answer the items according to your perception of the category/theme Software Configuration Management. What is the necessity for:"											
	Very High		High		Medium		Low		No Influence		Total
The use of versioning tools, like SVN and Git CSF13: Tools support	62.3%	48	23.4%	18	10.4%	8	2.6%	2	1.3%	1	77
"Answer the items according to your perception of the category/theme Tests. What is the necessity for:"											
	Very High		High		Medium		Low		No Influence		Total
The use of Test Plans, Unit Tests and Test Cases CSF14: Inspections / Reviews / Software validation	43.8%	39	42.7%	38	7.9%	7	4.5%	4	1.1%	1	89
The use of regression Tests, like the ones used in Selenium or JBehave CSF14: Inspections / Reviews / Software validation	23.9%	16	52.2%	35	17.9%	12	6.0%	4	0.0%	0	67
"Answer the items according to your perception of the category/theme Automation. What is the necessity for:"											
	Very High		High		Medium		Low		No Influence		Total
The use of facility, compilation and packaging management tools like Ant, Maven, Gradle, NPM CSF13: Tools support	37.2%	23	38.7%	24	17.7%	11	4.8%	3	1.6%	1	62
The use of Continuous Integration tools, like Jenkins or Gitlab CI/CD CSF13: Tools support	31.7%	20	42.9%	27	22.2%	14	0.0%	0	3.2%	2	63
The use of code analysis tools, like Firebugs and SonarQube CSF13: Tools support	21.0%	13	43.5%	27	32.3%	20	0.0%	0	3.2%	2	62

In Table 6, the findings related to the CSFs "Tools support" and "Inspections, Reviews and Software validation" reveal another relevant aspect. Although the literature consistently emphasizes these practices as contributors to software quality, the responses suggest uneven adoption across organizations. This may indicate that implementation barriers are associated not only with technical complexity but also with organizational maturity and resource availability.

In this sense, Chiyangwa and Mnkandla (2017) state that the companies need to plan the use of agile methodologies, by investing in tools that sponsor and assist in the fast development of iterations, configuration management and other agile techniques.

Table 7

Answers regarding Project Management for the Category Technical/Process

"Answer the items according to your perception of the category/theme Project Management. What is the necessity for:"											
	Very High		High		Medium		Low		No Influence		Total
Evaluating the characteristics of the software project in order to choose the adequate management and development methods. CSF15: Characteristics of the software project and adequate methodology	47.4%	46	40.2%	39	8.3%	8	1.0%	1	3.1%	3	97
The use and deployment of agile processes. CSF16: Deployment of agile development techniques	35.0%	34	47.5%	46	16.5%	16	0.0%	0	1.0%	1	97
Impacts related to changes in the planned tasks for iteration delivery CSF17: Altering iteration/ Story change	26.4%	23	59.8%	52	10.3%	9	2.3%	2	1.2%	1	87
The use of task management tools, like Redmine, JIRA CSF13: Tools support	35.3%	30	40.0%	34	20.0%	17	2.4%	2	2.3%	2	85
The use of frequent project follow-up meetings and team alignment meetings CSF16: Deployment of agile development techniques	46.9%	46	33.7%	33	15.3%	15	3.1%	3	1.0%	1	98
Follow-up metrics to improve the delivery of software projects with quality CSF18: Metrics and Quality	41.7%	40	41.7%	40	12.4%	12	2.1%	2	2.1%	2	96
Well-defined and well-known coding standards CSF19: Well-defined coding standards	41.9%	39	41.9%	39	11.8%	11	3.3%	3	1.1%	1	93
Constant evaluation of the lessons learned for the continuous improvement of techniques and processes CSF20: Continuous improvement	39.8%	39	45.9%	45	11.3%	11	1.0%	1	2.0%	2	98
29. Simplicity in coding and software design CSF21: Simple Code and Design/Simplicity	40.4%	38	47.9%	45	9.6%	9	0.0%	0	2.1%	2	94

In Table 7, project management factors received consistently high evaluations, reinforcing their importance in software development environments. Nevertheless, the predominance of respondents occupying management or analytical positions may have influenced these perceptions. Therefore, the findings should be interpreted considering the professional profile of the sample.

Overall, the results confirm the importance of people, organizational culture, and project management practices, while also revealing that respondents attribute greater influence to factors that directly affect project execution and day-to-day activities.

These findings are also supported by the literature. In particular, Aldahmash *et al.* (2017) argue that, before adopting an agile method or technique, project members must be familiar with the selected approach. Furthermore, the tools, principles, and terminology associated with the chosen agile methodology should be clearly understood by all team members to facilitate its effective implementation and contribute to project success.

4.2.4 Ranking of the Critical Success Factors

Table 8 presents the ranking of the most influential CSFs in the adoption of agile methods focused on the quality of the final product and on the use of complementary tools. Based only on the answer options NO INFLUENCE, LOW, HIGH and VERY HIGH for each factor, according to Paixão (2014), we classified the Critical Success Factors according to a specific and detailed criterion, right after the following table. This ranking organizes the CSFs from the most voted one to the least voted one. In the cases where the survey presents more than one question for the same CSF, we used the highest score for developing the ranking. This classification was simply based on the answers to the objective questions of the survey.

Table 8
Ranking of the critical success factors

Category	CSF	Very High	High	Low	No Influence	Score	Position
Personnel	CSF5: Team work Question(s): 5	62	30	4	0	88	1
Personnel	CSF2: Support from top management Question(s): 2	45	45	2	0	88	2
Personnel	CSF1: Competence, training and specialization Question(s): 1	53	38	4	0	87	3
Personnel	CSF4: Communication skills Question(s): 4	52	35	3	1	83	4
Personnel	CSF3: Motivation and acknowledgement of the good work Question(s): 3	57	31	7	0	81	5
Technical/Process	CSF15: Characteristics of the software project and adequate methodology Question(s): 21	46	39	1	3	81	6
Technical/Process	CSF20: Continuous improvement Question(s): 28	39	45	1	2	81	7
Technical/Process	CSF21: Simple Code and Design/Simplicity Question(s): 29	38	45	0	2	81	8
Organizational	CSF7: Cooperative organizational culture Question(s): 7	31	53	3	1	80	9
Technical/Process	CSF16: Deployment of agile development techniques Question(s): 22, 25	34	46	0	1	79	10
Personnel	CSF6: Group/individual resistance Question(s): 6	39	45	6	0	78	11
Technical/Process	CSF18: Metrics and Quality	40	40	2	2	76	12

Category	CSF	Very High	High	Low	No Influence	Score	Position
Technical/Process	Question(s): 26 CSF12: Efficient requirement gathering technique, traceability and scope definition.	52	27	4	1	74	13
Technical/Process	Question(s): 13, 14 CSF19: Well-defined coding standards	39	39	3	1	74	14
Organizational	Question(s): 27 CSF8: Human resources management	37	43	2	4	74	15
Technical/Process	Question(s): 8 CSF14: Inspections / Reviews / Software validation	39	38	4	1	72	16
Technical/Process	Question(s): 16, 17 CSF17: Altering iteration/ Story change	23	52	2	1	72	17
Technical/Process	Question(s): 23 CSF10: Focus, client's satisfaction and engagement	29	46	1	3	71	18
Organizational	Question(s): 10 CSF9: Project management norms and procedures	33	44	5	3	69	19
Technical/Process	Question(s): 9 CSF13: Tools support	48	18	2	1	63	20
Technical/Process	Question(s): 15, 18, 19, 20, 24 CSF11: Poorly characterized risk planning	17	41	6	3	49	21
	Question(s): 11, 12						

In order to develop the ranking of CSFs, we adopted the criterion based on Paixão (2014), which is detailed as follows:

- The sum of the answers with the degrees of influence VERY HIGH and HIGH, represented in the formula (VH + H).
- The sum of the answers with the degrees of influence LOW and NO INFLUENCE, displayed by the formula (L + NI).
- The result of the formula (VH + H) – (L + NI) will be the final score of each factor. If there is a draw among the CSFs in the result of the formula, the one with the higher number of answers VERY HIGH will prevail.

We based the adopted criterion on Paixão (2014) because it is a previously used criterion for establishing a ranking among critical success factors. It is the most adequate method for the ranking due to the fact it uses the low degrees of influence LOW and NO INFLUENCE as depreciation rates for the score of each factor (Cicco, 2017).

It is possible to verify that a distribution among the categories of CSFs happened in Table 8: the factors in the category Personnel were the most important ones, followed by a mix between the ones in the categories Technical/Process and Organizational. It shows how important it is for the company to invest in the Personnel area for the management and development of quality software.

Something curious is the fact that the CSF “Poorly Characterized Risk Planning” obtained the last place in the ranking of importance. This factor is quite important for a good project plan, and consequently important to meet the budget and delivery goals. Likewise, the CSF “Tools Support” obtained the penultimate place, evidencing the lack of concern by the companies about the necessity to invest in development tools, tests and project follow-ups. Lastly, the position of the CSF “Project Management Norms and Procedures” in the ranking should be more relevant, because many respondents of the survey belong to the project management area.

5 CONCLUSION

The purpose of this study, in addition to identifying the CSFs associated with agile software development environments, was to examine how professionals perceive the relative influence of these factors on software quality and process improvement. By integrating evidence from the literature with empirical data collected from Brazilian software professionals, this research provides a structured understanding of the factors that are perceived as most relevant to successful agile implementation.

The literature review identified 21 CSFs, grouped into Personnel, Organizational and Technical/Process dimensions. While previous studies have often discussed these factors individually, this research contributes by consolidating them into an integrated framework and empirically assessing their perceived influence within a specific national context. The survey results reinforce the central role of human-centered factors in agile environments, particularly competence, training and specialization, teamwork, communication skills, and customer engagement. The predominance of these factors suggests that organizational investments in agile transformation should extend beyond the adoption of methodologies and tools, emphasizing the development of human capabilities and collaborative practices.

A significant finding emerging from the comparison between literature and empirical evidence is the relative prominence of Personnel-related CSFs over Technical/Process factors. This result contributes to the ongoing debate in the agile software development literature by suggesting that technical practices and tool integration, while necessary, may not be sufficient conditions for achieving quality outcomes. Instead, their effectiveness appears to depend on the existence of organizational and human capabilities that support collaboration, learning and knowledge sharing.

The study also highlights the importance of integrated development environments and process automation. We can notice the necessity of an environment with tools that are integrated with the process, based on development, tests, activity control, generating products with quality. The development environment must also have the integration of the whole team, by exchanging information, content and knowledge.

From a theoretical standpoint, this study, rather than proposing new factors, refines existing knowledge by demonstrating how the perceived importance of CSFs is distributed across different dimensions and by providing evidence that human-related factors consistently occupy the highest positions in the influence ranking. This finding contributes to the maturation of the CSF literature by offering a more nuanced understanding of the relative weight of these factors in practice.

This research shows that the CSFs found in the literature and validated by the answers obtained in the survey lead to improvement in the development process with the use of agile methods, and provide quality to the delivered software. Integrating tools into the process is possible and essential for the success of the implementation. Besides, training and integrated environment for the team is quite important to meet the challenge of developing quality software.

Therefore, we expect that this research study may contribute to researchers and managers, and that they may be able to apply it and measure the CSFs and the proposed tools in development systems. As a suggestion, the identified CSFs can be used in the measurement area, or as a knowledge basis for other studies.

Nevertheless, the scope of the contribution should be interpreted within the boundaries of the research. The findings are based on the perceptions of 101 Brazilian respondents and therefore reflect a specific cultural, organizational, and professional context. Consequently, the study's contribution lies in providing contextual evidence that can serve as a basis for comparison with other national and organizational settings.

Future research should extend the investigation to other countries and organizational contexts, enabling comparative analyses that examine whether the prominence of Personnel-related CSFs remains consistent across different cultural and industrial environments.

REFERENCES

- Abdalhamid, S., & Mishra, A. (2017). Factors in agile methods adoption. *TEM Journal*, 6(2), 416–421. <https://doi.org/10.18421/TEM62-29>
- Aldahmash, A., Gravell, A. M., & Howard, Y. (2017). A review on the critical success factors of agile software development. In J. Stolfa, S. Stolfa, R. O'Connor, & R. Messnarz (Eds.), *Systems, software and services process improvement. EuroSPI 2017. Communications in Computer and Information Science*, vol. 748 (pp. 504–512). Springer. https://doi.org/10.1007/978-3-319-64218-5_41
- Amorim, A. C., Silva, M. M. da, Pereira, R., & Gonçalves, M. (2021). Using agile methodologies for adopting COBIT. *Information Systems*, 101, Article 101496. <https://doi.org/10.1016/j.is.2020.101496>
- Ayat, M., Imran, M., Ullah, A., & Kang, C. W. (2021). Current trends analysis and prioritization of success factors: A systematic literature review of ICT projects. *International Journal of Managing Projects in Business*, 14(3), 652–679. <https://doi.org/10.1108/IJMPB-02-2020-0075>
- Biesialska, K., Franch, X., & Muntés-Mulero, V. (2021). Mining dependencies in large-scale agile software development projects: A quantitative industry study. In *EASE 2021: Evaluation and Assessment in Software Engineering* (pp. 20–29). <https://doi.org/10.1145/3463274.3463323>
- Carosia, J. S. (2004). *Levantamento da qualidade do processo de software com foco em pequenas organizações* [Dissertação de Mestrado, INPE, Instituto Nacional de Pesquisas Espaciais]. <http://mtc-m16.sid.inpe.br/col/sid.inpe.br/jeferson/2003/12.03.10.17/doc/publicacao.pdf>
- Chiyangwa, T., & Mnkandla, E. (2017). Modelling the critical success factors of agile software development projects in South Africa. *SAJIM South African Journal of Information Management*, 19(1), Article 838. <https://doi.org/10.4102/sajim.v19i1.838>
- Cicco, L. R. (2017). *Fatores críticos de sucesso na implantação de electronic document management system* [Dissertação de Mestrado, Departamento de Engenharia de Produção, Universidade Federal Fluminense, Niterói, RJ]. Repositório Institucional da UFF.

<https://app.uff.br/riuff/bitstream/handle/1/9037/Dissert%20Leandro-De%20Cicco.pdf?sequence=1&isAllowed=y>

- Cockburn, A., & Highsmith, J. (2001). Agile software development: The business of innovation. *Computer*, 34(9), 120–127. <https://doi.org/10.1109/2.947100>
- Conceição, J., & Silveira, S. R. (2018). Aplicação de metodologias ágeis para desenvolvimento de software: Um estudo de caso na empresa Alliance Software. *Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica*, 9(1), 136–155.
- Costa, H. G. (2010). Modelo para webibliomining: Proposta e caso de aplicação. *Revista da FAE*, 13(1), 115–126.
- Costinhas, C. E., & Santos, G. (2014). Adoção de práticas ágeis em um ambiente Tradicional: Um estudo de caso. *Anais do XIII Simpósio Brasileiro de Qualidade de Software (pp. 239-246)*. <https://sol.sbc.org.br/index.php/sbqs/article/view/15256/15100>
- Cruz Estrada, I., & Miranda Zavala, A. (2019). La adopción de las tic en restaurantes de Puerto Nuevo, Rosarito, Baja California. *Innovar*, 29(72), 59–76. <https://doi.org/10.15446/innovar.v29n72.77932>
- Díaz Pinzón, B. H., Gómez Medina, J. S., García González, J. D., Melo Román, H. A., & Sanabria Villamizar, F. E. (2017). Contribución de las iniciativas de tecnologías de la información en las organizaciones: Una revisión de la literatura. *Innovar*, 27(66), 41–55. <https://doi.org/10.15446/innovar.v27n66.66710>
- Digalwar, A. K., Haridas, P., & Joseph, I. N. (2014). Development and validation of quality management constructs for software industries: An empirical investigation from India. *International Journal of Services and Operations Management*, 17(4), 453–478. <https://doi.org/10.1504/IJSOM.2014.060003>
- Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, 119, 87–108. <https://doi.org/10.1016/j.jss.2016.06.013>
- Fronza, I., & Wang, X. (2021). Social loafing prevention in agile software development teams using team expectations agreements. *IET Software*, 15(3), 214–229. <https://doi.org/10.1049/sfw2.12019>
- Ghayyur, S. A. K., Ahmed, S., Ali, M., Razzaq, A., Ahmed, N., & Naseem, A. (2018). A systematic literature review of success factors and barriers of agile software development. *International Journal of Advanced Computer Science and Applications*, 9(3), 278–291. <https://doi.org/10.14569/IJACSA.2018.090339>
- Gray, D. E. (2012). *Pesquisa no mundo real*. Penso.
- Gren, L., Knauss, A., & Stettina, C. J. (2018). Non-technical individual skills are weakly connected to the maturity of agile practices. *Information and Software Technology*, 99, 11–20. <https://doi.org/10.1016/j.infsof.2018.02.006>
- Hillenbrand, A., Störl, U., Nabiyevev, S., & Klettke, M. (2022). Self-adapting data migration in the context of schema evolution in NoSQL databases. *Distributed and Parallel Databases*, 40, 5–25. <https://doi.org/10.1007/s10619-021-07334-1>

- Huerta Riveros, P. C., Almodóvar Martínez, P., & Pedraja Rejas, L. M. (2015). Factores que impactan los resultados empresariales: Un estudio comparativo entre empresas chilenas y españolas. *Innovar*, 25(56), 21–41. <https://doi.org/10.15446/innovar.v25n56.48987>
- Medeiros, J. D. R. V., Alves, D. C. P., Vasconcelos, A. M. L., Schuenemann, C. T. L. L. S., & Wanderley, E. (2015). Engenharia de requisitos em projetos ágeis: Uma revisão sistemática da literatura. *Revista Principia: Divulgação Científica e Tecnológica do IFPB*, 28, 11–24. <https://doi.org/10.18265/1517-03062015v1n28p11-24>
- Medeiros, J., Vasconcelos, A. M. L., Silva, C., & Goulão, M. (2018). Quality of software requirements specification in agile projects: A cross-case analysis of six companies. *Journal of Systems and Software*, 142, 171–194. <https://doi.org/10.1016/j.jss.2018.04.064>
- Mohanarajah, S., & Jabar, M. (2015). An improved adaptive and dynamic hybrid agile methodology to enhance software project success deliveries. *Journal of Theoretical and Applied Information Technology*, 75(3), 301–325.
- Muhammad, A., Siddique, A., Naveed, Q. N., Saleem, U., Hasan, M. A., & Shahzad, B. (2021). Investigating crucial factors of agile software development through composite approach. *Intelligent Automation and Soft Computing*, 27(1), 15–34. <https://doi.org/10.32604>
- Neelu, L., & Kavitha, D. (2021). Estimation of software quality parameters for hybrid agile process model. *SN Applied Sciences*, 3(3), 1–11. <https://doi.org/10.1007/s42452-021-04305-0>
- Oliveira, B. H. (2014). *Qualidade de software no desenvolvimento com métodos ágeis* [Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo]. https://teses.usp.br/teses/disponiveis/55/55134/tde-12082014-165244/publico/BrunoOliveira_revisada.pdf
- Pacheco, D. A. D. J., Reis, C. E. D., & Jung, C. F. (2020). Método de estratégia de manufatura para terceirizadas calçadistas orientado à agile manufacturing. *Innovar*, 30(75), 99–118. <https://doi.org/10.15446/innovar.v30n75.83260>
- Paixão, T. R. (2014). *A influência dos fatores críticos de sucesso na gestão por processos de negócio do BPM* [Dissertação de Mestrado, Departamento de Engenharia de Produção, Universidade Federal Fluminense]. Repositório Institucional da Universidade Federal Fluminense. <https://app.uff.br/riuff/handle/1/20786>
- Pekki, J. (2016). How the company manages critical success factors in software process improvement initiatives: Pilot case-study in finnish software company. In C. Kreiner, R. O'Connor, A. Poth, & R. Messnarz (Eds.), *Systems, software and services process improvement. EuroSPI 2016. Communications in Computer and Information Science*, vol. 633. Springer. https://doi.org/10.1007/978-3-319-44817-6_15
- Pressman, R. S. (2010). *Software engineering – A practitioner's approach* (7th ed.). McGraw-Hill.
- Silva, K. M. B., & Santos, S. C. (2015). Critical factors in agile software projects according to people, process and technology perspective. *Proceedings of the 6th Brazilian Workshop on Agile Methods*, 48–54. <https://doi.org/10.1109/WBMA.2015.19>

Šmite, D., Moe, N. B., & Gonzalez-Huerta, J. (2021). Overcoming cultural barriers to being agile in distributed teams. *Information and Software Technology*, 138, Article 106612. <https://doi.org/10.1016/j.infsof.2021.106612>

Sommerville, I. (2007). *Software engineering* (8th ed.). Pearson Addison Wesley.

Souza, M. L. P., Souza, W. C., Freitas, J. S., Melo Filho, L. D. R., & Bagnó, R. B. (2022). Agile roadmapping: A management tool for digital entrepreneurship. *IEEE Transactions on Engineering Management*, 69(1), 94–108. <https://doi.org/10.1109/TEM.2020.3027918>

Tabares Quiroz, J., Correa Vélez, S., & Herrera Caballero, J. M. (2021). Metamorfosis organizacionales del cambio tecnológico: Integración de sistemas informáticos en una organización de salud colombiana. *Innovar*, 31(79), 103–116. <https://doi.org/10.15446/innovar.v31n79.91894>

Tsoy, M., & Staples, D. S. (2020). Exploring critical success factors in agile analytics projects. *Proceedings of the 53rd Hawaii International Conference on System Sciences*. <https://doi.org/10.24251/HICSS.2020.122>