
A Systematic Review of Risk Management Tools and Techniques in Software Projects

Fadhli Luthfiansyah¹, Aji Prasetyo², Teguh Raharjo³

fadhli.luthfiansyah@ui.ac.id, aji.prasetyo21@ui.ac.id, teguhr2000@gmail.com

^{1,2,3}Faculty of Computer Science, Master of Information Technology, University of Indonesia, Jakarta

Article Information

Submitted : 18 Jan 2024
Reviewed: 20 Jan 2024
Accepted : 19 Feb 2024

Keywords

Project Risk
Management; Risk Tools
and Techniques;
Software Development;
Systematic Literature
Review

Abstract

The crafting of software is a continual procedure, and the success of each step of that process is contingent on effective management. Despite this, numerous organizations need help developing e-service systems, frequently dealing with budget constraints and tight deadlines. The lack of focus on risk management in software projects is likely to blame for these failures. Management of risks is crucial to ensuring the success and efficacy of software development projects, as it assists in identifying areas of vulnerability and provides valuable insights into the project's most important aspects. This study identifies and analyzes tools and techniques to support software development projects risk management activity. A systematic literature review (SLR) methodology was employed to collect and evaluate relevant research articles. The findings highlight various risk management tools and techniques, including brainstorming, root cause analysis, risk probability assessment, artificial intelligence, and risk response planning. These tools and techniques contribute to identifying, analyzing, planning, and controlling risks in software projects. The research provides insights into the state of the art in risk management. It complements previous studies by offering practical guidance on software development project risk management tools and techniques.

A. Introduction

The development of software is an iterative process, and the success of each step of that process is contingent on effective management [1]. Many organizations struggle with difficulties in their e-service system development initiatives [2]. The challenge commonly faced by organizations is increasing budget needs and passing a predetermined schedule [3]. This is due to a lack of awareness or sufficient time for risk management in the IS / IT project [3], [4].

Risk management is essential for a software development project to be effective and efficient. Analyzing risk makes it possible to identify vulnerable points in the schedule and provide information regarding the sensitivity of the entire schedule's components [5]. The pioneer of software project risk management, Barry Boehm, said, "Risk management is important especially because it helps people to avoid disasters, rework, and cancellation of projects and helps to stimulate a situation of success in software projects." [6].

Furthermore, employing a comprehensive management approach to handle risks in a concurrent project setting facilitates the identification of major risks on an enterprise scale rather than solely focusing on individual projects. This endeavor results in a streamlined allocation of resources, ensuring a more effective pursuit of the company's overarching goals. [7].

Four primary steps make up the core software risk management [8][9]: Risk identification is the process of finding any possible occurrences that might have a negative impact on the project; Risk analysis is conducted to determine the severity of the potential consequences and the likelihood that the hazard in question will result in unfavorable outcomes; Risk planning is the process of establishing strategic choices, determining actions, improving opportunities, and minimizing risks to the project's goals; The implementation of risk monitoring and control measures is crucial in order to identify and address potential issues proactively.

Although risk management is essential to software development, this stage is often skipped over in software development [10]. According to a Project Management Institute (PMI) study conducted worldwide, only 49% of projects are completed on schedule [11]. In addition, according to a report released by Wellington, that only 64% of project managers involve risk management in their projects [12]. As the world's largest professional organization specializing in project management, PMI has recognized risk management as one of the eight main focus areas in the Project Management Body of Knowledge (PMBOK) and a research topic frequently discussed by researchers in project management [4], [13], [14].

Previous research has been conducted to identify and analyze risk management in software development [1]. However, the exploration of tools and techniques used in risk management remains scarce as little to no concrete explanations are available. Therefore, this study aims to identify and analyze tools and techniques that can support the risk management process that can be applied to software development. This study provides state-of-the-art and complements previous research that still lacks project software risk management tools and techniques.

Risk Management

The management of risks remains an essential component of project management, and it plays an important part in ensuring that projects ultimately achieve their intended goals. The major purpose of this strategy is to identify and eliminate any possible risks that might have an effect on the targeted outcomes. For example, small and medium-sized businesses (SMEs) use preventative measures to reduce the impact of these risks and ensure that the project will be successful overall [15]. Risk management activities include [8], [9], [16]:

1. Identify the Risk

The process of identifying risk factors involves recognizing potential opportunities and threats that could affect different stages of a project and documenting their characteristics. Assessing the impact of these risk factors entails estimating the probability and consequences of their occurrence and prioritizing them based on their potential impact on project implementation. The identification and assessment of risk factors occur at specific points in time, such as the beginning and end of an iteration, considering both project-related and manufacturing process-related factors. In a hybrid approach combining traditional and agile methods, project-specific risk factors are identified during the planning phase, while the continuous process focuses on identifying and evaluating factors arising directly from the product manufacturing process within each iteration. The team leader should record and report issues raised by the team during daily meetings. The outcome of this identification and assessment process is a detailed list of identified risk factors, described in a manner that makes sense for the project team in their specific situation.

2. Analyze the Risk

In this phase, particular risks are prioritized for further investigation or action. The likelihood of occurrence, potential impact, and other relevant factors are considered. It may also involve quantitative analysis to assess how specific risks interact with other sources of uncertainty and their combined effect on project goals.

3. Plan Risk Response

The process of identifying choices, choosing methods, and deciding on actions to tackle particular project hazards in addition to addressing the overall risk exposure of the project.

4. Control Measure and Monitor The Risk

The process of putting into action risk response strategies that have been mutually agreed upon. In addition, monitoring risks is a critical aspect of risk management, involving the continuous evaluation and tracking of implemented risk response plans, identification and analysis of new risks, and assessment of the overall effectiveness of the risk management process throughout the project.

B. Research Method

The method used in this study is the Systematic Literature Review (SLR) method. The SLR method is a specific research methodology or research conducted to collect and evaluate research related to a particular topic focus. We have used the PRISMA guidelines provided by Matthew J. [17] to ensure a comprehensive and unbiased systematic review (Figure 1).

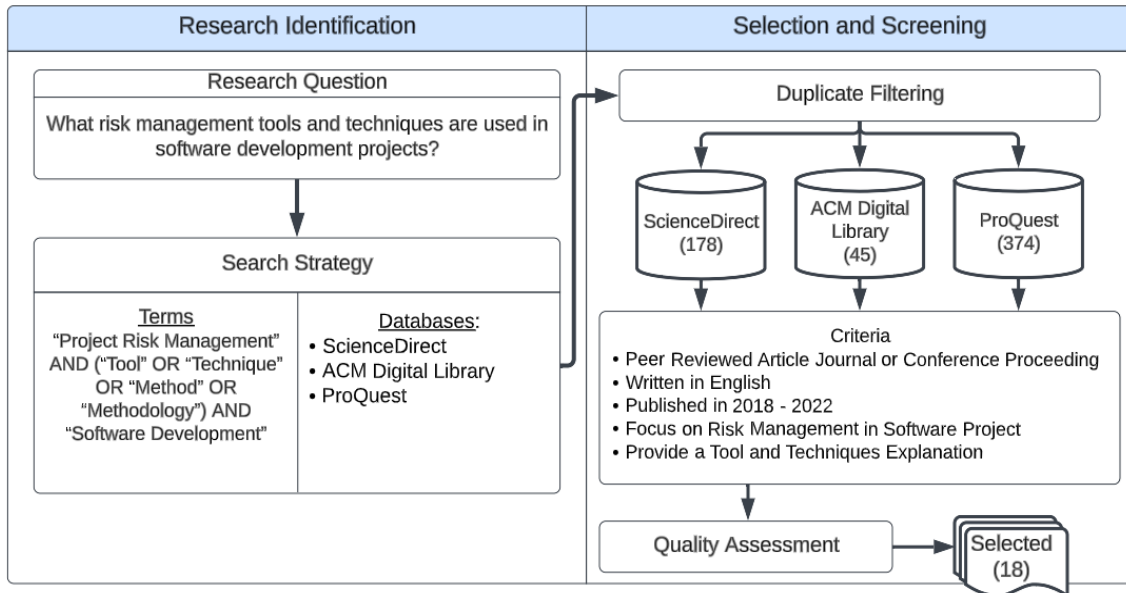


Figure 1. An overview of the PRISMA process for locating and choosing appropriate literature on software project risk management techniques and tools.

This SLR consists of four main steps: primary study planning and search, study collection, data extraction, and data synthesis. Part III-A identifies research objectives and questions as the first step. In parts III-B and III-C, search strategy steps involve studying selection criteria, study selection procedures, keyword formulation for research, and search queries. In III-D, the final step requires quality assessment.

1. Research Questions and Objectives

The primary purpose of this SLR is to explore risk management implementation in software development projects. We create research questions to be able to focus on the objectives of this research.

RQ: What risk management tools and techniques are used in software development projects?

Based on these research questions, the focus of this study's objectives is to review the trends that have occurred in recent years, especially the use of tools and techniques for risk management in software development projects.

2. Search Strategy

In this study, we conducted a search using the electronic databases ScienceDirect, ACM Digital Library, and ProQuest. We prepared several lists of

keywords to search for relevant literature on project risk management in software development from selected electronic databases. **“Project Risk Management” AND (“Tool” OR “Technique” OR “Method” OR “Methodology”) AND “Software Development”**. Electronic database searches that utilize article titles, abstracts, and keywords to retrieve relevant articles.

3. Selection Criteria

We analyze the query results that have been obtained by removing duplicate articles. Filtering is also done based on inclusion and exclusion criteria.

The study's inclusion criteria encompassed empirical journal articles and conference proceedings in English, specifically those that were peer-reviewed and published between 2018 and 2022. The articles could be qualitative or quantitative in nature, but they had to address the topic of risk management. However, any articles that were not in English or were published in 2017 or earlier were excluded.

Any articles that were in the form of posters with only graphics and lacked complete textual content, surveys, or systematic reviews were also excluded from consideration. Furthermore, articles that did not explicitly mention any software project in their title, abstract, or keyword section, and failed to address the topic of risk management, were also excluded from the analysis.

4. Quality Assessment

Quality assessment is used to assess the quality of the selected article. Quality assessment (QA) is also used to assess whether the selected article is fully accessible and can answer our review. To determine consistency, we formulated some QA questions.

QA1: Does the article discuss any risk management on the software development project?

QA2: Does the article use tool or technique of risk management clearly stated?

QA3: Are the aims of the research clearly stated without ambiguity in the paper?

QA4: Are the limitations of the study mentioned?

Each question has a yes/no response option with a 1-point or 0-point value. After a thorough evaluation of the whole article's quality, the findings are considered. All research publications should go through the QA procedure in accordance with the QA questions. Therefore, this review includes all 18 selected articles.

C. Result and Discussion

This evaluation of the relevant literature includes 18 articles that focus on risk management in software projects between 2018 and 2022. Figure 2 provides a concise summary of the distribution of the different kinds of publications that were subjected to the selection and screening procedure. The figure shows that journal articles make up 66.7 percent of the total, while conference proceedings make up 33.3 percent.

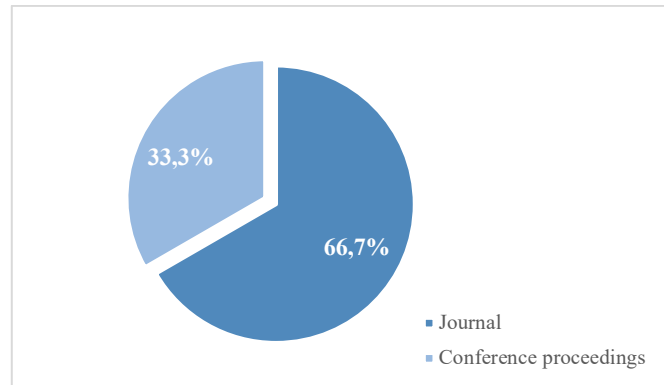


Figure 2. Distribution of Publication by Type.

According to Figure 3, the total number of articles consists of the following: three articles in 2018, six articles in 2019, two articles in 2020, five articles in 2021, and two articles in 2022. There was a rise of three articles in the total number of publications in 2019, making it the year with the largest number of publications in the last five years. In addition, the years 2020 and 2022 had just two articles each, which was the lowest number of articles ever recorded.

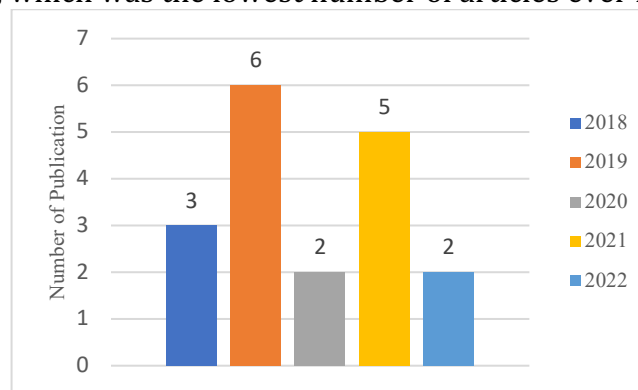


Figure 3. Annual frequency of risk management in software project-related publications.

Risk Management Tools and Techniques in Software Project

This section describes the tools used for comprehensive risk management in software projects. We summarize the results obtained from the entire literature, as shown in Table I. It entails the identification, assessment (qualitative and quantitative analysis), response plan, monitoring, and controlling of project risks.

Table 1. Risk Management Tools and Techniques in Software Project

Relate Studies	Tools and Technique	Activity
[18][19][20][21]	Brainstorming	Identify Risk
[22]	Root Cause of the Problems	
[23]	Learning Cycle	
[24]	Periodic Risk Reporting	
[22][21][24]	Risk Probability Assessment	Analyze Risk
[25][23][26][27][28][21][24]	Risk Impact Assessment	
[7][21]	Risk Classification	
[25][23][26][27][29][21]	Ranking of Risk	
[18][30]	Neural Network	

[23][31][32][33]	Artificial Intelligence	
[22][26][21][24][34]	Planning for risk mitigation	Plan Risk Response
[19]	Project Rescheduling	
[18][26][21]	Looks for Risk triggers and implement Contingency Plan	Control Measure and Monitor the Risk
[19][20][24]	Periodic risk status reporting	

1. Identify the Risk

The process of identifying risks holds immense significance in ensuring effective project management, as it entails a systematic exploration of potential hazards and threats that may emerge throughout the project's lifecycle.

Within the perspective of software project development, multidisciplinary teams comprising project managers, software engineers, and esteemed academics engage in a dynamic and collaborative brainstorming process to meticulously identify and analyze various risk factors and sources. This interactive brainstorming approach allows participants to gain a comprehensive understanding of the project's complexities, while categorizing and evaluating a wide array of variables and incidents associated with risks [18]–[21]. To streamline this intricate process, affinity diagrams are employed, enabling the consolidation and categorization of the identified risks into an initial risk list [18].

To ensure a seamless integration within Agile methodologies, it is crucial to seamlessly incorporate risk identification efforts. The output of this process is a Risk Register, overseen by the Project Manager or Scrum Master, wherein every team member can contribute by utilizing root cause analysis to identify potential risks [22]. Moreover, the team consistently reports risks during daily scrum meetings, ensuring that the Risk Register remains up-to-date and reflects the evolving project landscape [24].

2. Analyze the Risk

The achievement of goals such as scope, timeline, cost, and quality is vital for the success of software development projects. Therefore, project managers and teams must acquire comprehensive knowledge about the underlying risk factors [25].

A detailed analysis was conducted to identify the causes and potential effects of risks at the company level. Risks are assessed thoroughly and given a score based on the likelihood and potential impact, ranging from low, medium to high levels. Using this information, a risk matrix was created to provide a visual representation of the risks, considering their probability and severity [21], [22], [24]. In an agile context, if a risk is deemed urgent and impacts the current Sprint, immediate assessment and response via ad hoc meetings is required. Otherwise, the risk can be postponed until the next official Risk Management meeting, which will be held at the start of the Sprint [22].

In addition, a number of studies have reviewed, screened, and determined a collection of risk variables that significantly influence each project goal by using rank-ordering techniques of risk [21], [23], [25]–[27], [29]. To analyze the data, Shrivastava [25] used the Kendall Concordance test, a statistical tool that allows quantification of consensus among responses with regard to the effect of each risk factor on project objectives. This test facilitates determining the statistical

significance of the average rating associated with each project objective for the identified risk factors. By utilizing the average rating, the team was able to distinguish project targets that experienced the greatest impact from each risk factor.

Neural networks have become valuable tools for risk analysis in software projects. Li et al. [18] has introduced a novel risk assessment model that integrates the rough set and backpropagation neural network (BPNN). This model specifically aims to enhance the monitoring of software project risks. The underlying concept of this joint approach involves using the rough set for attribute reduction of software project risk factors' sample data. Subsequently, the reduced set is fed into a BP neural network for training, resulting in a refined classification model. Ultimately, the model's output is utilized to evaluate the level of risk in software projects, thus facilitating informed decision-making for project stakeholders.

Furthermore, Sembiring et al. [30] presents novel risk assessment approaches built exclusively for application development and deployment. This method combines Bayesian networks with Boehm's software risk ideas. The network's risk probabilities are rigorously analyzed using numerical simulations and important feedback from domain experts and application development teams. Using the ideas established in the Boehm risk framework, the derived Bayesian network model is utilized to quantify the amount of risk exposure. In Indonesia, a government agency successfully applied this strategy.

Other techniques such as artificial intelligence can also be used for risk analysis in software projects [23], [31]–[33]. The incorporation of fuzzy theory in calculating the weight of risk factors brings about enhanced scientificity and accuracy in risk assessment, leading to a more effective reduction in the likelihood of risk occurrence [32]. This technique enables the evaluation of the degree of uncertainty and associated risks and provides a buffer to safeguard the current Sprint and the entire project against the adverse effects of underestimation. Additionally, it allows for analyzing potential advantages and risks associated with overestimation. By embracing fuzzy numbers and embracing the inherent uncertainty in estimation, organizations can make more informed decisions and effectively manage risks throughout the project lifecycle [31].

3. Plan Risk Response

The process of risk response planning plays a vital role in addressing the underlying causes of identified and assessed risks. It is imperative to develop comprehensive strategies that encompass various elements, including well-defined plans, effective risk mitigation techniques, and contingency plans, among other considerations. The primary objective of these strategies is to minimize the likelihood of unfavorable risks materializing during the course of a project [22]. Furthermore, the implementation of risk response strategies should be aligned with the perceived level of risk, cost constraints, and realistic timeframes, taking into account the expectations and perspectives of the stakeholders involved [26].

When it comes to risk response, it involves the proactive formulation of measures or actions that can be swiftly deployed in the event that risks manifest themselves. Regardless of the project methodology employed, if a risk management strategy other than acceptance is adopted, adjustments to the project

plan become necessary. This holds true even in agile project management approaches that emphasize adaptability and iterative development processes. For instance, opting for a risk avoidance strategy entails the identification of associated activities and their prioritized execution in the nearest iteration. By doing so, the aim is to reduce the likelihood of the specific risk factor materializing and causing potential disruptions [24].

Agile methods, renowned for their short iteration cycles, frequent team meetings, and continuous customer engagement, inherently integrate risk mitigation strategies with prompt responsiveness. These methodologies provide a conducive environment for teams to proactively address risks and swiftly adapt to changing circumstances. Through ongoing collaboration and communication, teams can identify potential risks early on, develop appropriate mitigation measures, and implement them in a timely manner. This agile approach ensures that risks are managed effectively throughout the project lifecycle, promoting a higher level of project success [22], [24].

On the other hand, if a risk transfer strategy is deemed appropriate, decision-making and actions should be escalated to the project management level. This involves the careful consideration of potential risks that could be transferred to external parties or stakeholders who possess the necessary capabilities and resources to handle them effectively. It is worth noting that decisions regarding the specific risk mitigation actions to be utilized should be made during iteration planning, allowing sufficient time for the team to prepare and implement these measures effectively. By incorporating risk response considerations into the iteration planning process, teams can proactively manage risks and enhance the overall project outcomes [24].

4. Control Measure and Monitor the Risk

studies have incorporated the succeeding crucial stages in attentive risk monitoring and management after developing a well-crafted reaction plan. It demands the team's active execution of the planned risk responses, as well as continual monitoring for triggers with the potential to develop into actual consequences [18], [21], [26]. Furthermore, when deemed necessary, contingency plans should be promptly executed to mitigate the adverse effects. To facilitate this essential process, it is highly recommended that all identified risks and their corresponding responses are meticulously documented on individual "risk cards." These cards should be prominently displayed and made easily accessible to the team during official meetings, serving as a constant reminder of the ongoing risk management process. By adopting this approach, the risk management process remains dynamic and visible throughout the project's lifespan, eliminating any possibility of it being relegated to a static and disregarded plan [22].

In lightweight methodologies such as Scrum, risk factors directly linked to the project's requirements and scope, encompassing vital aspects such as product quality, end-users, customers, corporate culture, and decision-making time, primarily fall within the domain of the astute Product Owner [24]. Conversely, the dexterous Scrum Master assumes the pivotal responsibility of addressing risks associated with the project's timeline, budgetary constraints, team dynamics, technical and technological environment, and the overarching organizational

context. The boundary management process, an integral component of agile project management, encompasses diligent and meticulous recording and reporting of risks, alongside the strategic direction and guidance of the project [20]. It serves as the compass that steers the project towards its objectives while effectively mitigating potential risks.

In both lightweight and more rigid agile approaches, a similar emphasis is placed on the allocation of roles and responsibilities. However, in the stricter agile methodology, significant risks carry the obligation of prompt communication to the astute project manager for comprehensive monitoring and expeditious resolution [19], [24]. This rigorous protocol ensures that potential risks are promptly identified, closely monitored, and diligently managed to safeguard the project's overall success and mitigate any adverse consequences that may arise along the way.

D. Conclusion

This research delved into a comprehensive analysis of 18 scholarly articles spanning 2018 to 2022, with a primary focus on risk management in software projects. The findings highlighted a discerning distribution of publication types, with journal articles comprising 66.7% of the corpus, while conference proceedings accounted for the remaining 33.3%.

The thorough literature review unveiled many invaluable tools and techniques employed in the risk management domain for software projects. These encompassed a wide array of practices, including but not limited to identifying risks, meticulous analysis, formulating effective risk response plans, and continuously monitoring and controlling potential risks throughout project lifecycles. The research successfully identified and explored methodologies such as brainstorming, root cause analysis, periodic risk reporting, risk probability assessment, employment of neural networks and artificial intelligence, and contingency planning. A salient emphasis was placed on early risk identification, proactive planning of risk responses, and continuous vigilance and control to mitigate potential risks, ultimately ensuring enhanced project success effectively.

This study also revealed that agile methodologies, renowned for their inherent adaptability, collaborative nature, and prompt responsiveness, provide an ideal environment for fostering effective risk management practices. Additionally, the allocation of roles and responsibilities, effective communication channels, and the seamless integration of risk management within the fabric of project planning and execution emerged as pivotal factors,

Limitations

This study identifies several limitations in the current state of risk management in software development projects. These limitations include the lack of standardized risk management practices, inadequate integration of risk management into project management processes, and the need for more research on effective risk management techniques.

Future Works

Regarding future research, the authors suggest developing standardized frameworks and guidelines for risk management, incorporating risk management into agile and DevOps methodologies, and investigating emerging technologies such as machine learning and data analytics to improve risk identification and analysis. Additionally, the article emphasizes the importance of empirical studies to evaluate the effectiveness of different risk management approaches and to identify best practices for mitigating risks in software development projects.

E. Acknowledgment

We would like to give special thanks to the Indonesian Ministry of Communication and Informatics Scholarship, and The Education and Training Center for The Indonesian Agency for Meteorology, Climatology and Geophysics for their financial support of this study. Furthermore, support from the University of Indonesia's Faculty of Computer Science.

F. References

- [1] J. Masso, F. J. Pino, C. Pardo, F. García, and M. Piattini, "Risk management in the software life cycle: A systematic literature review," *Comput. Stand. Interfaces*, vol. 71, p. 103431, Aug. 2020, doi: 10.1016/j.csi.2020.103431.
- [2] S. Pekkola, M. Ylinen, and N. Mavengere, "Consortium of Municipalities Co-tailoring a Governmental e-Service Platform: What could go wrong?," *Digit. Gov. Res. Pract.*, vol. 3, no. 1, pp. 1–16, Jan. 2022, doi: 10.1145/3511889.
- [3] W. Sardjono and A. Retnowardhani, "Analysis of Failure Factors in Information Systems Project for Software Implementation at The organization," in 2019 International Conference on Information Management and Technology (ICIMTech), Aug. 2019, pp. 141–145. doi: 10.1109/ICIMTech.2019.8843725.
- [4] J. T. Marchewka, *Information Technology Project Management: Providing Measurable Organizational Value*. Wiley, 2016. [Online]. Available: <https://books.google.co.id/books?id=rsLlBQAAQBA>
- [5] A. Martens and M. Vanhoucke, "A buffer control method for top-down project control," *Eur. J. Oper. Res.*, vol. 262, no. 1, pp. 274–286, Oct. 2017, doi: 10.1016/j.ejor.2017.03.034.
- [6] B. Boehm, "Software risk management," in *ESEC'89: 2nd European Software Engineering Conference University of Warwick, Coventry, UK September 11--15, 1989 Proceedings*, 2005, pp. 1–19.
- [7] I. M. Alharbi, A. A. Alyoubi, M. Altuwairiqi, and M. A. Ellatif, "Enhance Risks Management of Software Development Projects in Concurrent Multi-Projects Environment to Optimize Resources Allocation Decisions," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, 2021, doi: 10.14569/IJACSA.2021.0120626.
- [8] Jingyue Li, O. P. N. Slyngstad, M. Torchiano, M. Morisio, and C. Bunse, "A State-of-the-Practice Survey of Risk Management in Development with Off-the-Shelf Software Components," *IEEE Trans. Softw. Eng.*, vol. 34, no. 2, pp. 271–286, Mar. 2008, doi: 10.1109/TSE.2008.14.
- [9] S. Chaouch, A. Mejri, and S. A. Ghannouchi, "A framework for risk management in Scrum development process," *Procedia Comput. Sci.*, vol. 164, pp. 187–192, 2019, doi: <https://doi.org/10.1016/j.procs.2019.12.171>.

- M. Hammad, I. Inayat, and M. Zahid, "Risk Management in Agile Software Development: A Survey," in 2019 International Conference on Frontiers of Information Technology (FIT), Dec. 2019, pp. 162–1624. doi: 10.1109/FIT47737.2019.00039.
- [10] M. A. Langley, "The High Cost of Low Performance How will you improve business results? 8th Global Project Management Survey," Proj. Manag. Inst., 2016.
- [11] V. Hines, "Annual Report - The State of Project Management 2021," Wellington Ltd., 2021, [Online]. Available: <https://wellington.co.uk/wp-content/uploads/2021/03/The-State-of-PM-2021.pdf>
- [12] T. Raz and E. Michael, "Use and benefits of tools for project risk management," Int. J. Proj. Manag., vol. 19, no. 1, pp. 9–17, Jan. 2001, doi: 10.1016/S0263-7863(99)00036-8.
- [13] M. Padalkar and S. Gopinath, "Six decades of project management research: Thematic trends and future opportunities," Int. J. Proj. Manag., vol. 34, no. 7, pp. 1305–1321, Oct. 2016, doi: 10.1016/j.ijproman.2016.06.006.
- [14] K. Vichova, P. Taraba, and T. Belantova, "Risk Management of the Project and the Use of Software in SME," WSEAS Trans. Bus. Econ., vol. 17, pp. 551–559, May 2020, doi: 10.37394/23207.2020.17.54.
- [15] Project Management Institute, A guide to the project management body of knowledge (PMBOK guide), 6th ed. Project Management Institute, Inc, 2017.
- [16] M. J. Page et al., "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," Syst. Rev., vol. 10, no. 1, pp. 1–11, 2021, doi: 10.1186/s13643-021-01626-4.
- [17] X. Li, Q. Jiang, M. K. Hsu, and Q. Chen, "Support or Risk? Software Project Risk Assessment Model Based on Rough Set Theory and Backpropagation Neural Network," Sustainability, vol. 11, no. 17, p. 4513, 2019, doi: <https://doi.org/10.3390/su11174513>.
- [18] P. Mäkiäho and T. Poranen, "Risks Management in Software Development Capstone Projects," in Proceedings of the 19th International Conference on Computer Systems and Technologies, 2018, pp. 160–164. doi: 10.1145/3274005.3274024.
- [19] M. Mousaei and T. J. Gandomani, "A New Project Risk Management Model based on Scrum Framework and Prince2 Methodology," Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 4, 2018, doi: <https://doi.org/10.14569/IJACSA.2018.090461>.
- [20] I.-C. Costea-Mareu and G. Militaru, "Managing Business Risks for a Software Development Company," FAIMA Bus. Manag. J., vol. 9, no. 3, pp. 52–59, Sep. 2021, [Online]. Available: <https://www.proquest.com/scholarly-journals/managing-business-risks-software-development/docview/2582863090/se-2?accountid=17242>
- [21] L. G. Cuong, P. D. Hung, N. L. Bach, and T. D. Tung, "Risk Management for Agile Projects in Offshore Vietnam," in Proceedings of the 10th International Symposium on Information and Communication Technology, 2019, pp. 377–384. doi: 10.1145/3368926.3369718.
- [22] R. G. Jimoh, O. O. Olusanya, J. B. Awotunde, L. I. Agbotiname, and L. Cheng-Chi, "Identification of Risk Factors Using ANFIS-Based Security Risk

- Assessment Model for SDLC Phases,” *Futur. Internet*, vol. 14, no. 11, p. 305, 2022, doi: <https://doi.org/10.3390/fi14110305>.
- [23] M. Trzeciak, “Sustainable Risk Management in IT Enterprises,” *Risks*, vol. 9, no. 7, p. 135, 2021, doi: <https://doi.org/10.3390/risks9070135>.
- [24] S. V. Shrivastava and U. Rathod, “A Goal-driven Risk Management approach for Distributed Agile Development Projects,” *Australas. J. Inf. Syst.*, vol. 23, 2019, doi: <https://doi.org/10.3127/ajis.v23i0.1843>.
- [25] C. M. Tae, P. D. Hung, and L. D. Huynh, “Risk Management for Software Projects in Banking,” in *Proceedings of the 2020 The 6th International Conference on E-Business and Applications*, 2020, pp. 65–69. doi: 10.1145/3387263.3387268.
- [26] N. A. H. Ghazali, S. S. M. Fauzi, R. Gining, W. Sobri, and A. J. Suali, “Visualizing Software Risks in Software Engineering Projects using Risk Sensitivity Analysis Approach,” *J. Phys. Conf. Ser.*, vol. 1529, no. 2, Apr. 2020, doi: <https://doi.org/10.1088/1742-6596/1529/2/022074>.
- [27] T. Karimi and Y. Yahyazade, “Developing a risk assessment model for banking software development projects based on rough-grey set theory,” *Grey Syst.*, vol. 12, no. 3, pp. 574–594, 2022, doi: 10.1108/GS-05-2021-0074.
- [28] B. A. Jnr, “Validating the usability attributes of AHP-software risk prioritization model using partial least square-structural equation modeling,” *J. Sci. Technol. Policy Manag.*, vol. 10, no. 2, pp. 404–430, 2019, doi: <https://doi.org/10.1108/JSTPM-06-2018-0060>.
- [29] J. Sembiring, J. P. Sipayung, and A. A. Arman, “Application Development Risk Assessment Model Based on Bayesian Network,” *TELKOMNIKA*, vol. 16, no. 3, pp. 1376–1385, Jun 2018, doi: <https://doi.org/10.12928/TELKOMNIKA.v16i3.7288>.
- [30] D. Kuchta, “Application of Fuzzy Sets to the Expert Estimation of Scrum-Based Projects,” *Symmetry (Basel)*, vol. 11, no. 8, p. 1032, 2019, doi: <https://doi.org/10.3390/sym11081032>.
- [31] Y. Zhou and Y. Hu, “Research on Software Risk Assessment Model Based on AHP- Fuzzy Comprehensive Evaluation,” in *Proceedings of the 2020 4th International Conference on Management Engineering, Software Engineering and Service Sciences*, 2020, pp. 16–20. doi: 10.1145/3380625.3380648
- [32] A. Tlili and S. Chikhi, “Risks Analyzing and Management in Software Project Management Using Fuzzy Cognitive Maps with Reinforcement Learning,” *Informatica*, vol. 45, no. 1, pp. 133–141, Mar. 2021, doi: <https://doi.org/10.31449/inf.v45i1.3104>
- [33] E. Dantas et al., “Investigating Technological Risks and Mitigation Strategies in Software Projects,” in *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, 2022, pp. 1527–1535. doi: 10.1145/3477314.3507062.

Appendix A: Selected Articles

No	Title	Name of Journal / Conference proceedings	Year of Publication	Quartile	Country	Type of Publication	Last SJR Rank
----	-------	--	---------------------	----------	---------	---------------------	---------------

No	Title	Name of Journal / Conference proceedings	Year of Publication	Quartile	Country	Type of Publication	Last SJR Rank
1	A Goal-driven Risk Management Approach for Distributed Agile Development Projects	Australasian Journal of Information Systems	2019	Q2	Australia	Journal	0,423
2	A New Project Risk Management Model based on Scrum Framework and Prince2 Methodology	International Journal of Advanced Computer Science and Applications	2018	Q3	United Kingdom	Journal	0,258
3	Application Development Risk Assessment Model	Telecommunication, Computing, Electronics and Control	2018	Q3	Indonesia	Journal	0,286
4	Application of Fuzzy Sets to the Expert Estimation of Scrum-Based Projects	MDPI - Symmetry	2019	Q2	Switzerland	Journal	0,483
5	Developing a risk assessment model for banking software development projects based on rough-grey set theory	Grey Systems	2021	Q2	United Kingdom	Journal	NA
6	Enhance Risks Management of Software Development Projects in Concurrent Multi-Projects Environment to Optimize Resources Allocation Decisions	International Journal of Advanced Computer Science and Applications	2021	Q3	United Kingdom	Journal	0,258
7	Identification of Risk Factors Using ANFIS-Based Security Risk Assessment Model for SDLC Phases	MDPI - Future Internet	2022	Q2	Switzerland	Journal	0,773

No	Title	Name of Journal / Conference proceedings	Year of Publication	Quartile	Country	Type of Publication	Last SJR Rank
8	Investigating Technological Risks and Mitigation Strategies in Software Projects	Proceedings of the ACM Symposium on Applied Computing	2022	NA	United States	Conference proceedings	NA
9	Managing Business Risks for a Software Development Company	FAIMA Business & Management Journal	2021	NA	Romania	Journal	NA
10	Research on Software Risk Assessment Model Based on AHP- Fuzzy Comprehensive Evaluation	International Conference on Management Engineering, Software Engineering, and Service Sciences	2020	NA	China	Conference proceedings	NA
11	Risk Management for Agile Projects in Offshore Vietnam	Proceedings of the International Symposium on Information and Communication Technology	2019	NA	Vietnam	Conference proceedings	NA
12	Risk Management for Software Projects in Banking	Proceedings of the International Conference on E-Business and Applications	2020	NA	Malaysia	Conference proceedings	NA
13	Risks Analyzing and Management in Software Project Management Using Fuzzy Cognitive Maps with Reinforcement Learning	Informatica	2021	Q2	Netherland	Journal	0,624
14	Risks Management in Software Development Capstone Projects	Proceedings of the International Conference on Computer Systems and Technologies	2018	NA	Bulgaria	Conference proceedings	NA
15	Support or Risk Software Project Risk Assessment Model Based on Rough Set Theory and Backpropagation	MDPI - Sustainability	2019	Q2	Switzerland	Journal	0,664

No	Title	Name of Journal / Conference proceedings	Year of Publication	Quartile	Country	Type of Publication	Last SJR Rank
	Neural Network						
16	Sustainable Risk Management in IT Enterprises	MDPI - Risk	2021	Q3	Switzerland	Journal	0,439
17	Validating the usability attributes of AHP-software risk prioritization model using partial least square-structural equation modeling	Journal of Science and Technology Policy Management	2019	Q2	United Kingdom	Journal	0,647
18	Visualizing Software Risks in Software Engineering Projects using Risk Sensitivity Analysis Approach	Journal of Physics: Conference Series	2019	Q4	United Kingdom	Conference proceedings	0,183