An Adaptive Framework for Risk Management in Agile Methodology

Saba A. H. Barman (1,*)

Naseebah Maqtary (1,*)

¹ Department of Computer Science, University of Science and Technology, Sana'a, Yemen

* Corresponding authors: saba.barman2020@gmail.com, n.almaqtary@ust.edu.ye

An Adaptive Framework for Risk Management in Agile Methodology

Abstract:

Software companies face numerous risks during the software development process, particularly when using Agile methodologies due to their dynamic nature. This dynamism makes it challenging for development teams to anticipate and effectively manage potential risks. The Agile approach's focus on speed sometimes impedes a comprehensive risk assessment, making it challenging to prioritize and address risks effectively. Ignoring these risks during software development can significantly impact product quality and customer satisfaction, potentially causing severe damage to both companies and clients. In this paper, an adaptive framework for risk management in an agile method for software development has been proposed. In general, the extent of risk management application in software companies in Sana'a that use the agile method in developing software was surveyed, identifying the risks likely to occur and their impact on the project, and then proposing a risk management framework that helps in reducing risks. Through the study, it was shown that the proposed framework is important and effective for risk management in the agile methodology so it obtained a rate of 69.6% from the perspective of local software companies.

Keywords: Agile methodology, Risk management in software project, Monitoring and controlling risks, Risks management in agile, Analysis risk in projects agile, Risk identification in projects agile.

إطار تكيفي لإدارة المخاطر في منهجية أجايل

الملخص:

تواجـــه شـــركات البرمجيات العديد من المخاطـــر خلال عملية تطوير البرمجيـــات، خاصة عند استخدام منهجيات أجايل بســـبب طبيعتها الديناميكية. هذه الديناميكية تجعل من الصعب على فـــرق التطوير توقع وإدارة المخاطر المحتملة بشـــكل فعال. تركيز منهجية أجايل على الســـرعة أحيانـــا مما يعيق تقييم المخاطـــر، ويجعل من الصعب تحديد أولويات المخاطر ومعالجتها بشـــكل فعـــال. إن تجاهـــل هذه المخاطر أثنـــاء تطوير البرمجيات يمكن أن يؤثر بشـــكل كبير على جودة المنتج ورضا العملاء، وبالتالي يتسبب ذلك في أضرار جسيمة لكل من الشركات والعملاء. في هذه الورقة، تم اقتراح إطار عمل تكيفي لإدارة المخاطر في منهجية أجايل لتطوير البرمجيات. بشكل عام، تم إجراء دراسة لاستقصاء مدى تطبيق إدارة المخاطر في شركات البرمجيات في صنعاء التي تعتمـــد منهجية أجايل في تطوير البرمجيات. وقد تم تحديـــد المخاطر المحتملة وتأثيراتها على المشـــروع، بالإضافة إلى اقتراح إطار عمل لإدارة المخاطر يســـاهم في تقليل هذه المخاطر. أظهرت النتائـــج أن الإطار المقترح يعتبر مهمًا وفعـــالًا في إدارة المخاطر ضمن منهجية أجايل، حيث حصل على تقييم بنسبة %69.6 من وجهة نظر شركات البرمجيات المحلية.

الكلمـــات المفتاحيـــة : منهجيـــة أجايـــل، إدارة المخاطر في المشـــاريع البرمجيـــة، مراقبة المخاطر والسيطرة عليها، إدارة المخاطر في مشاريع أجايل، تحليل المخاطر، تحديد المخاطر.

1. Introduction

Software engineering is a process whose success and failure depends on the team, the technical environment and the organization. Developing a software project is considered a team work and the project is delivered based on different conditions and factors. Usually in traditional development, after completing the development and testing, the customer gets the final product at once and may or may not satisfied by this product. While in Agile development, the customer enjoys continuous participation in his project and receives software sprints to ensure his satisfaction [1]. Therefore, agile methodology development is the most successful development method [2]. Agile methods are processes that support the agile philosophy, i.e. agile values and principles. Each Agile method consists of a different combination of practices, which is a description of how the day-to-day work is done by the software developer [2]. It ensures that a quality software product will be delivered to the customer, using an adaptive, iterative and incremental path through cross-functional and self-organized teams [3]. Each method differs from the other by choosing its appropriate set of terminology and practices [2]. The aim of this paper is to present a proposed framework for risk management in Agile methodology. The life cycle of the agile methodology will be reviewed, and its steps will be explained in detail. The remainder of the paper organized as follows: Section 2 presents agile methodology and risk management. Types of risks are introduced in Section 3. After that Section four clarifies the study methodology. The related work is surveyed in Section 5. Section 6 depicts the proposed framework and Section 7 shows the result and discussion, Finally conclusion and future work is presented.

2. Agile Methodology and Risk Management

This section consists of two parts: the first part covers Agile methodology steps, and the second part contains risks and risk management.

2.1 Agile Methodology Development Steps

The Agile methodology life cycle consists of six steps [3]. These steps help to enhance adaptation and responsiveness to changes in accordance with the needs of the project. The six steps of the Agile methodology life cycle are depicted in the Figure 1:

1) Planning: Start by defining the project goals, setting the budget, and identifying available resources. Divide the project into small tasks and determine priorities.

2) Divided into short-term deliveries: Divide the project into small work cycles known as "Sprints," typically lasting two weeks to a month. Each Sprint aims to deliver assessable and usable results.

3) Execution: Execute the tasks identified for each Sprint by the team. Work on achieving the set goals in the Sprint with continuous interaction with customers or stakeholders.

4) Evaluation and learning: After each Sprint, review and evaluate the results. Discuss performance, identify strengths, and weaknesses. Use these insights to improve the development process in the upcoming Sprints.

5) Iteration: Repeat the process of Sprints, evaluation, and learning until the final product is achieved with the best quality and within the specified time.

6) Continuous communication: Maintain continuous communication between the team members and between the team and the customers or stakeholders throughout the development process. This allows for quick changes and product improvement based on feedback and new requirements.
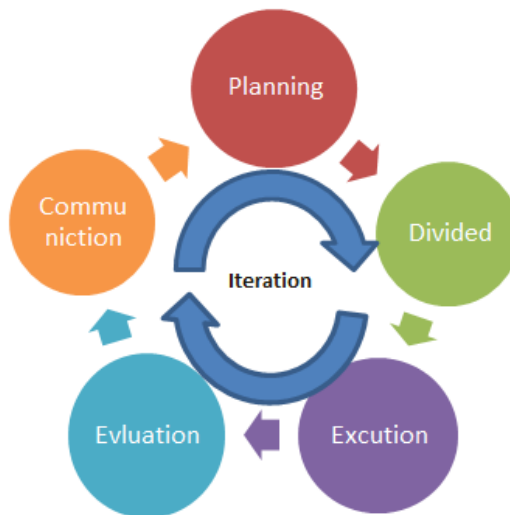

Figure 1: The Agile methodology steps [3]

2.2 Risk and Risk Management

This will define the concept of risk and what risk management is, and we will mention the importance of risk management in software projects.

A. The project risk [4] is defined as an infinite phenomenon whose presence effects of project positively or negatively.

B. Risk management [5] is a set of processes for identifying and controlling areas or events that have the potential of causing unwanted changes.

C. Importance of risk management.

Risk management strategies positively impact projects by reducing delays, lowering costs, and improving performance. Conversely, a lack of interest in risk management from managers will lead to new risks and additional costs for project sponsors, which will lead to poor relationship between the organization and the client [6,7,8,9]. And as studied in [9], Risk management has a fundamental role in project management, as it identifies the threats that arise during project implementation and manages these threats quickly, this allows software companies to avoid incurring losses that may be related to the length of time it takes to complete the project, increased production costs, meeting deadlines, product quality of the project, In addition to that, obtaining a smaller market share etc. This reflects to us that risk management is important for programming methods in general and agile in particular.

Lack of risk management or not applying it well is the most reason for the failure of software development projects [10], so approximately 11% of projects were canceled before delivering the first expected result to the client, and this is due to the lack of effective risk management, so it is necessary to pay attention to risk management during the work of software projects [11]. Therefore, one of the pillars of software project management is risk management in software development because it causes great concern to researchers [12].

3. Types of Risks

During software development, the Agile methodology may face various types of risks. In the following, we will provide an explanation of each type of risk, as they were collected from various sources and studies:

- Schedule risk refers to the potential of delays or disruptions in the planned timeline of a project [13].

- Budget risk refers to the potential of project to exceed its allocated or planned budget [13].

- Technical risk refers to the potential challenges and uncertainties related to the development and implementation of the technology or software being built [13, 14].

- Programmatic risks in Agile methodology can arise due to the complexities of coordinating various teams, aligning their work with the overall program goals, and ensuring effective communication and collaboration across teams [13].

- Lack of staff knowledge risk in Agile methodology refers to the potential challenges that arise when team members do not possess the necessary skills, expertise to effectively complete the tasks or deliverables required for a project [12, 15, 16].

- Deployment risk in Agile methodology refers to the potential challenges that arise during the process of releasing and deploying software or product increments into a production environment [13].

- Performance risk in Agile methodology refers to the potential challenges related to the performance characteristics of a product being developed. It involves the concern that the delivered solution might not meet the desired performance criteria [16, 17].

- Frequent architectural changes risk in Agile methodology refers to the potential challenges that arise when there are frequent and significant modifications to the software's underlying architecture during the development process [18].

- Lack of communication between the team and the client in Agile methodology refers to the potential challenges that arise when there is insufficient or ineffective communication between the development team and the client or stakeholders [17, 19].

- Technology risks are difficulty keeping up with the latest technology trends due to constant changes in this field [13].

4. Study Methodology

In this section, the methodology used in this paper will be discussed as follows: It begins with reviewing the relevant literature, focusing on studies related to risk management within Agile methodology. Next an adaptive framework is proposed, outlining the key phases for its development. Then this framework is validated by the software companies in Sanaa through a validated questionnaire. The results are obtained by analyzing the data collected from the questionnaire, followed by a discussion of the findings. Lastly, the conclusion and future recommendations are addressed.

5. Related Work

Risk management is considered one of the basic steps in building and developing any project, especially in the agile method. Therefore, research continues to explore the best methods and practices for risk management, and many researchers are still presenting a lot of research on risk management, In this section, some of this research will be presented as follows:

In research [5], The authors introduced a novel approach to managing risks involving PO's roles. the authors established a framework called RIsk Management PRoduct Owner (RIMPRO), which intends to support project teams to systematically manage risks related to PO activities that may arise during the project. the processes described in RIMPRO were evaluated by potential users. Through a preliminary assessment, they observed that RIMPRO is promising since it can assist teams in managing risks involving PO in a systematized and effective manner.

The authors in [20], emphasized ARMF based on development related risks by following agile development. The authors helped software organizations to prevent different project-related risks during agile development. The authors elicited the risks at two levels, (i) literature-based and (ii) IT industry-based. Finally, they merged the agile development risks from literature with standard industrial risks. They established an agile risk mitigation framework, ARMF, based on agile development and presented a groundwork established in light of empirical examination for extending it in future research.

In [21], Proposed a model that integrates risk management more seamlessly into Agile projects. Empirical results from case studies have also been provided to prove the effectiveness of the model. Lastly, a reference Risk Register (Risks and their Responses) provided for the Project Managers to use as a starting point in their project.

In this research [14], the authors proposed a framework for integrating risk management in the agile development projects while considering Scrum as an agile method and the PMBOK as a project risk management guide. The framework was to improve the risk management mechanism in Scrum and to increase the Scrum project's success rate. Limitations in this paper did not adapt a risk management framework to other agile methods such as XP, and Lean.

Other authors in [11], developed and evaluated a tool for risk management in software development projects, that use agile methods. The proposed tool provided a collection of risk management practices and an iterative lifecycle. The authors conducted an experiment with the participation of experts in agile methods. The proposed tool increased the effectiveness of risk response planning without increasing the time invested.

In [22], aimed to propose a new framework. The authors combined the conceptual framework of Global Software Development integrated with risk management in a Scrum framework. As a result, this paper showed the application of the Risk Mitigation Framework for Agile Global Software Development (RMAG) in a Scrum process model. The authors used several scrum practices and phases to mitigate risks. The authors setup an experiment to apply the RMAG in Agile Global Software Development using Scrum. Each phase has different types of risk and mitigation techniques.

Pilliang and Munawar introduced a survey in risk management to find the most exciting topics for researchers, especially in software engineering. The authors introduced a systematic approach to review a risk management in software development projects. The authors used and collected papers and journals in the international library database, then summarized them according to the phases of the PICOC methodology [6].

In the study [23], presented a guide to risk management in agile software projects. The authors defined the structure of the guide based on standards and reference models. The authors developed content based on analyzed the state of art and the traditional risk management literature. The authors also developed a web tool to facilitate access and understand of the guide's content.

This research [24], aimed to survey function of machine learning algorithms to predict the software risks. The authors review the literature and presented an analytical comparison of machine learning algorithms for predicting software risks. Also, this paper presented further presents the current challenges in the field of software risk management that needs to be addressed. Also,

This research [9], aimed to systematically review the literature (SLR) on software risks. The authors described and presented the state of the art in this field, and identified opportunities and gaps for further research.

6. The Proposed Framework

In this section, the proposed framework will be presented in this paper, which consists of seven main phases: Plan Risk Management, Identify risk, Perform Qualitative and Quantitative analysis, Prioritize the Risk, Risk Response, Monitor and Control Risk, and Risk Review. And this section will explain each phase in this framework in detail. The framework is depicted in Figure 2.

A. Plan Risk Management

In this phase, a simple risk management plan is proposed. This plan contains strategic planning to reduce the occurrence of risks by identifying the necessary procedures to limit the negative effects of risks after they occur [25]. The risk is formally added to the sprint planning meeting agenda, during the sprint planning session. proposes a simple risk management plan which includes: Risk categories, roles and responsibilities, scheduling, budget, execution timing, impacts and more. The result of this phase is the creation of a risk management plan and risk register. Defining these aspects should involve all team members [14].

Figure 2: The proposed framework for risk management in agile methodology

B. Identify Risk

In this phase, all potential risks that could occur and that may negatively affect the software development process and the final product are identified [14, 25], This takes place from the beginning of the project until its publication [26]. The identification of risks involves the members of the Agile team where ideas will be exchanged between them, based on their previous experiences and expertise [5, 26]. These can be highlighted during agile daily sessions, many of which are resolved by the team immediately during the session [25]. The outcome of this activity will be the development of a risk register. So that this will be constantly used and updated throughout the Agile lifecycle [5, 26]. The risk register updates through identifying any new risks, evaluating old risks, and implementing appropriate response strategies, in order to manage the impact of risks on the Agile project [5].

C. Perform Qualitative and Quantitative analysis

This phase involves two types of analysis, qualitatively analyzing each risk by evaluating its probability of occurrence, impact, and other characteristics, and numerically analyzing the combined effect of the identified risk and other risks on the overall project objectives. Analyze quantitatively, each identified risk is analyzed and prioritized, and each sprint's risk analysis is stored in the sprint's risk register [7]. Project managers can perform the evaluation themselves or hire experts [52]. can be rated the qualitatively analysis on a simple scale from (low=1, med=2, high=3). And can be rated the quantitatively analysis based on the percentage [25].

D. Prioritize the Risk

At this phase, the risks associated with the development of the project are evaluated and prioritized by project managers[25, 27].For identify significant risks and calculate risk exposure it is being done multiplying the probability (on a scale of 1-3) by the impact (also on a scale of 1-3). The numerical value obtained is between1 to 9. Any risk in the range of 6 to 9 is a significant risk that important and must be managed. Risks in the range of 3 to 5 are It is less important to manage it if it is inexpensive to do so. Risks in the range of 2 to 1 is ignored. The target of risk prioritization in the proposed framework is to ensure that the task with the highest risk is completed.This is an ongoing process throughout a project, and allowing for changing priorities and adding new information [25].

E. Response

This phase is done in two processes: risk response planning and risk response implementation.

i. Risk Response Planning

This process is changing the information related to risk into activities and opinions [26]. It calls for develop options, select strategies, and agree on actions to address risk exposure to maximize opportunities and minimize threats to project objectives [5, 14]. This process is often performed during sprint planning. In sprint planning, the team through its knowledge contributes to the choice of optimal option for project risk, and then a risk implementation should be executed [14, 26]. Before dealing with risks an assessment must be performed to ensure that the risk is accepted by the product owner. If stakeholders consider the risk acceptable, risk treatment alternatives should be implemented using the necessary resources and monitored through further project activities. To reduce risks, specific responses can be developed depending on the project situation, danger, required resources, and response costs. Typically, the goal of a risk response strategy is to reduce or eliminate the likelihood of a threat occurring. Even if realized, the impact on risk would be limited. Or combine them. there are four general selections for responding to any project; risk avoidance, risk transference, risk mitigation and risk acceptance [26].

ii. Risk Response Implementation

Once a strategy has been selected it is necessary of implementing the risk response plan contained in the Sprint Risk Backlog and ensuring that the risk response is executed as planned [5, 14]. Following this process will ensure that the agreed risks are addressed. The risk response implementation is done during the sprint. As the project progresses, new risks are identified and an iterative process occurs. About risk management [7].

F. Monitor and Control Risks

This phase is done in two processes: risk monitoring process and risk control process.

i. Risk Monitoring Process

In this process, the risks are monitored between project iterations that may affect Sprint [5, 11, 25], monitoring the implementation of the risk response plans contained in the Sprint Risk Backlog; Also Evaluate the effectiveness of risk management processes during the Sprint. and previously identified risks are reviewed [5]. Also, this process is responsible for identifying and updating closed risks and any new risks identified during the project [25, 26]. It is performed during a sprint and is controlled by the risk owners [5, 14]. Additionally, it is continuously supported by the agile Master [7]. A major benefit of the monitoring process is that decisions the project is based on current information [5].

ii. Risk Control Process

Controlling the risks that are discovered by defining strategies to deal with the risks and creating a solution plan for the risks if they occur during project implementation [27].

G. Risk Review

In this phase, risks should be scheduled regularly and That is, checking and rearranging them So that new risks are identified, reassessing current risks, closing out risks that are out of date [5] and it should be removed the risk which not repeat. After that, the risks that will be retained are reorganized. This means that risks that occur frequently are raised to the highest risk for easy access. As for high-priority risks, a different place is allocated to them [26].

After presenting the proposed framework and its steps, a questionnaire was created and distributed to local software companies. The data was collected and the results were obtained, which will be presented in the next section.

7. Results and Discussion

There are many results that have been obtained throughout this study, but this section will present the most significant ones based on their importance:

7.1 Results

This section presents the most prominent results of the study obtained from analyzing and interpreting the data. This section starts with data screening which focuses on the way of coding and checking the missing data on SPSS, and descriptive statistics of the items of each variable used in the study (means,

and standard deviations). Next, this section presents the results of the one sample T-test. At the end of this section, the discussion will be presented.

A. The Most Important Phases from the Companies' Perspective

In this section, the importance of each phase of the proposed framework from the perspective of the sample individual. The following Figure 3 shows the percentage obtained by the phases through the sample's response to the questionnaire.
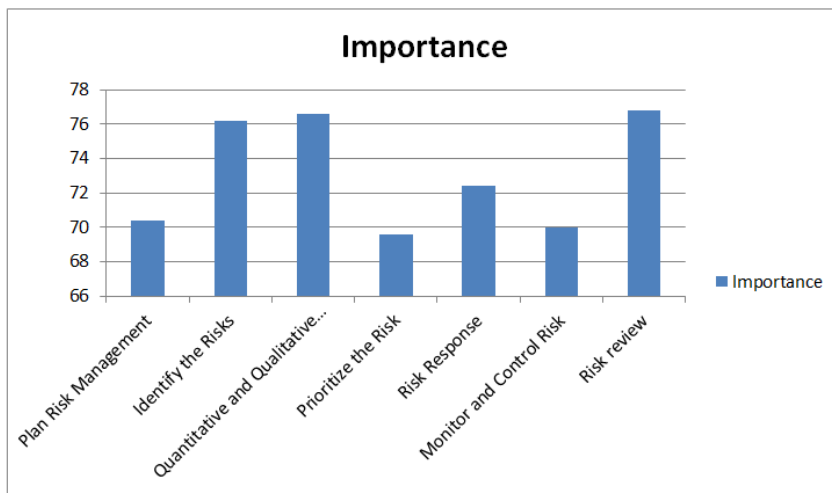

Figure 3: Important proportions of phases

B. Challenges of Implementing the Proposed Framework

The proposed framework may face several challenges during its implementation, which can be attributed to various reasons, including the need to change the company culture to enhance awareness of the importance of risk management and its integration into their work. Therefore, an attempt was made to address these challenges in the questionnaire to understand how to deal with them during the implementation of the proposed framework. The challenges in implementing the proposed framework were assessed across 7 items, with the statistical results presented in a Fig 4. The findings revealed that the phase most likely to face difficulties in implementation was the risk identification phase, with a percentage of 73.8%. This was followed by the phase of performing qualitative and quantitative analysis at 70.4%. The phase with the least potential difficulty in execution was the monitoring and control phase, with a percentage of 61.0%.
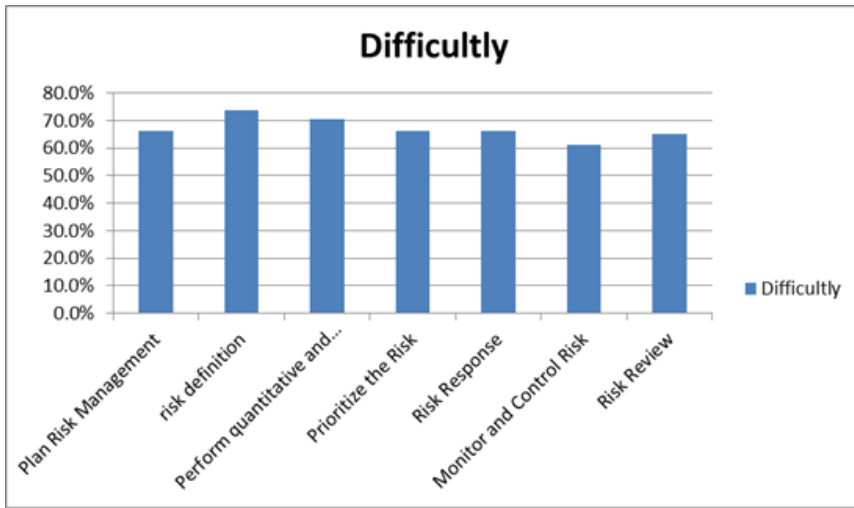
Figure 4: Difficulty integrating phases in risk management

7.2 Discussion

It became clear from the data analysis and the dimensions upon which the survey questions are based that the proposed framework is effective and adaptable, demonstrating its ability to comprehend and address risks in general.

The results also indicate that companies emphasize that all phases of the proposed framework are very important, as the risk priority phase achieved 69.6% and is considered the lowest phase among the proposed framework phase. This is higher than the acceptable level. Each stage is considered important in varying degrees, as follows:

"Plan Risk Management" accounts for 70.4% of the importance in risk management in fast operations. On the other hand, "Identify the Risks"represents 76.2% of the importance in risk management in fast operations. Based on the results provided, it appears that the "Perform Quantitative and Qualitative Analysis" phase is the most important in risk management in fast operations, with a percentage of 76.6%. Following that, the "Prioritize the Risk" phase is deemed important with a percentage of 69.6%. The "Risk Response" phase is shown to be important at 72.4%. The "Monitor and Control Risk" phase is highlighted as important at 70%.

Lastly, the "Risk Review" phase received a high percentage of 76.8% and was among the phases with high importance in risk management in fast operations.

It was shown from the Figure that all the various phases in the proposed framework, which were collected from various sources and analyzed, were very important through consensus.

It is true that all phases are important, but the results indicate that that there is phases that is more important than the others. This is because some companies are proficient in certain phases, such as plan risk management which is implemented by companies at a rate of 90.5%, and "risk response" at a rate of 88.1%. Therefore, they received low percentages in terms of importance.

The results indicate that the phase of performing Qualitative and Quantitative analysis is the most important phase in the proposed framework at a rate of 76.6%. This is because decisions about strategic matters depend on the accuracy and comprehensiveness of the analysis. This phase is considered the second phase that will face challenges when applying the proposed framework, at a rate of 70.4%, due to the difficulty of collecting quantitative and qualitative data for analysis. Furthermore, both quantitative and qualitative analysis is complex and requires specific skills and expertise in this field. Unfortunately, software companies here do not have a specialist or expert in this area, the one who manages risks is the project manager, and he obtained a percentage of 28.2%.

The results indicate that the monitoring and control phase was the second least important phase at 70.0%. This is because the majority of companies, at 35.6%, do not implement monitoring and control. There are also companies at 37.3% that do not have any strategy for tracking and monitoring risks, while another 15.7% have strategies but do not fully implement them. Additionally, the results show that 46.7% of companies work with partially monitored and controlled risks, where these companies do not use any tools or technologies. This conclusion was drawn from multiple questions.

8. Conclusion and Future Work

Risk management represents one of the most important phases that must be taken into consideration when developing any program in an agile method. As risk management makes the software development process more efficient

and effective, and through it, the progress of the project is ensured and its completion is to the satisfaction of the client and the developer, as it saves a lot of time and effort, but software companies in Yemen still lack adequate application of risk management.

This paper aimed to provide an effective framework for risk management in the Agile methodology.

In this paper, the extent to which software companies apply risk management was studied, and the study proved that most software companies do not care about risk management. Therefore, a framework was proposed for risk management in all aspects.

The proposed framework consists of several phases for risk management, including the following phases: Plan Risk Management, Identify risk, Perform Qualitative and Quantitative Analysis, Prioritize the Risk, Risk Response, Monitor and Control Risks, Risk Review. The study proved that the previous phases are very effective in managing risks to build software effectively and to the satisfaction of the customer.

This paper is concluded by making recommendations for future research directions, as follows:

-   All software companies recommend implementing risk management during software development.

-   Applying the proposed framework in corporate risk management.

-   Using tools and techniques to help manage risks.

-   Forming a risk management team.

References

[1]   S. Dhir, D. Kumar, and V. Singh, "Success and failure factors that impact on project implementation using agile software development methodology," in *Software Engineering: Proceedings of CSI 2015*, 2019, pp. 647-654.

[2]   S. Al-Saqqa, S. Sawalha, and H. AbdelNabi, "Agile software development: Methodologies and trends," I*nternational Journal of Interactive Mobile Technologies*, vol. 14, 2020.

[3]   A.-M. Gheorghe, I. D. Gheorghe, and I. L. Iatan, "Agile Software Development," *Informatica Economica*, vol. 24, 2020.

[4] P. K. Swain and B. B. Pradhan, "RISK MANAGEMENT IN TRADITIONAL AND AGILE RUNNING PROJECTS," *PalArch's Journal of Archaeology of Egypt/Egyptology*, vol. 17, pp. 5357-5365, 2020.

[5] S. d. S. Lopes, R. C. G. d. Souza, A. d. G. Contessoto, A. L. d. Oliveira, and R. T. V. Braga, "*A risk management framework for scrum projects,*" *in Proceedings*, 2021, pp. 30-40.

[6] M. Pilliang and M. Munawar, "Risk Management in Software Development Projects: A Systematic Literature Review," *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, vol. 8, 2022.

[7] E. Khanna, R. Popli, and N. Chauhan, "A Comparative Analysis of Machine Learning Techniques for Software Risk Assessment," 2022.

[8] R. Akbar, "Tailoring agile-based software development processes," *IEEE Access*, vol. 7, pp. 139852-139869, 2019.

[9] J. Masso, F. J. Pino, C. Pardo, F. García, and M. Piattini, "Risk management in the software life cycle: A systematic literature review," *Computer standards & interfaces*, vol. 71, p. 103431, 2020.

[10] R. V. Dandage, S. S. Mantha, and S. B. Rane, "Strategy development using TOWS matrix for international project risk management based on prioritization of risk categories," *International Journal of Managing Projects in Business*, vol. 12, pp. 1003-1029, 2019.

[11] B. G. Tavares, M. Keil, C. E. Sanches da Silva, and A. D. de Souza, "A risk management tool for agile software development," *Journal of Computer Information Systems*, vol. 61, pp. 561-570, 2021.

[12] A. Muayad Younus Alzahawi and M. Abumandil, "Evaluating the role of scrum methodology for risk management in information technology enterprises," *Journal of Information Technology and Computing*, vol. 2, pp. 1-8, 2021.

[13] 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)*, 2019, pp. 162-1624.

[14] S. Chaouch, A. Mejri, and S. A. Ghannouchi, "A framework for risk management in Scrum development process," *Procedia Computer Science*, vol. 164, pp. 187-192, 2019.

[15] K. Goher, A. Al-Ashaab, S. Sarfraz, and E. Shehab, "An uncertainty management framework to support model-based definition and enterprise," *Computers in Industry*, vol. 150, p. 103944, 2023.

[16] W. Qadadeh and S. Abdallah, "An improved agile framework for implementing data science initiatives in the government," in *2020 3rd International Conference on Information and Computer Technologies (ICICT)*, 2020, pp. 24-30.

[17] K. Chakravarty and J. Singh, "A study of quality metrics in agile software development," in *Machine Learning and Information Processing: Proceedings of ICMLIP 2020*, 2021, pp. 255-266.

[18] A. Salameh and J. M. Bass, "An architecture governance approach for Agile development by tailoring the Spotify model," *Ai & Society*, vol. 37, pp. 761-780, 2022.

[19] S. Ghobadi and L. Mathiassen, "Risks to effective knowledge sharing in agile software teams: A model for assessing and mitigating risks," *Information systems journal*, vol. 27, pp. 699-731, 2017.

[20] F. Marle, "An assistance to project risk management based on complex systems theory and agile project management," *Complexity*, vol. 2020, p. 3739129, 2020.

[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.

[22] Z. Podari, A. F. Arbain, N. Ibrahim, and E. Sudarmilah, "Risk Mitigation Framework for Agile Global Software Development," in *International Conference of Reliable Information and Communication Technology*, 2020, pp. 1233-1246.

[23] J. C. R. Hauck and M. Vieira, "Towards a guide for risk management integration in agile software projects," in *European Conference on Software Process Improvement*, 2021, pp. 73-87.

[24] E. Khanna, R. Popli, and N. Chauhan, "A Comparative Analysis of Machine Learning Techniques for Software Risk Assessment," *NeuroQuantology*, vol. 20, p. 3309, 2022.

[25] M. H. Zahedi, A. R. Kashanaki, and E. Farahani, "Risk management framework in Agile software development methodology," *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 13, 2023.

[26] N. Uikey and U. Suman, "Risk based scrum method: a conceptual framework," in *Proceedings of the 9th INDIACom; INDIACom-2015, IEEE Conference ID*, 2015, pp. 4.120-4.125.

[27] S. N. H. Ghazali, S. S. Salim, I. Inayat, and S. H. Ab Hamid, "A risk poker based testing model for scrum," *Computer Systems Science and Engineering*, vol. 33, pp. 169-185, 2018.