International Journal of Computers and Informatics (IJCI)
Vol. (3), No. (3)

IJCI

المجلة الدولية للحاسبات والمعلوماتية
الإصدار (3)، العدد (3)

March 2024

Monte Carlo Simulation for Risk Management in Agile Software Development

Ali Hamzah Obaid
Al-Furat Al-Awsat Technical University, Babylon Technical Institute
inb.ali210@atu.edu.iq

Khansaa Azeez Obayes Al-Husseini
Al-Furat Al-Awsat Technical University, Babylon Technical Institute
inb.khanssa@atu.edu.iq

Abstract

Agile development methodologies are becoming increasingly popular in agile development projects due to their iterative nature and adaptability. The Monte Carlo method is distinguished by its statistical technique of using random samples to obtain many satisfactory results for solving uncertainty. It can be used in agile software development by building different scenarios and measuring their impact on the project results through statistical operations. Its ability to meet significant challenges in managing risk and uncertainty mitigation.

In this study, we propose a Monte Carlo simulation-based approach to measure and analyze risks in the agile development process. It provides a probabilistic framework for risk assessment by integrating the Monte Carlo simulation methodology to model the various development process variables. It was addressed by identifying the main risk factors within agile projects, such as tasks, availability of resources, and external dependencies. The proposed approach contributes to mitigating potential risks in agile software development using Monte Carlo theory, which provides a systematic framework that enables work teams and management to overcome risk and

uncertainty associated with the agile development process in dynamic project environments.

Keywords: Monte Carlo, Agile, Software Development, Risk, Simulation.

1. Introduction

Recently, automation of the decision-making process has played an essential role in the development of intelligent systems. As software development companies face particular possibilities of risks, the need arises to predict the risks that these companies face during the development process. In the agile development process, there are many uncertainties and challenges that stakeholders and project managers may face, the most important of which are the changing requirements, especially by the end users of the program. These requirements significantly affect the planned timetable for completing the project and the material costs allocated for the process. Development, which in turn negatively affects the quality of the final product [1].

Therefore, we see many development teams tending to reduce risks by using several methods, and one of these methods is the use of Monte Carlo simulation because of its remarkable ability and effectiveness in managing risks and uncertainty in software development projects. This method identifies ways and opportunities to achieve the plans and goals set for modelling and evaluating alternative scenarios for developing volatile events and changing requirements during the development period [2].

Monte Carlo simulation is based on statistical calculations that take a set of repeated random samples to obtain numerical results and apply them to estimate the risks of an agile development project by building different scenarios and analyzing their potential effects on the development process results. The Monte Carlo method can be used in agile development projects to assess risks through:

1- Identifying potential risks due to changing requirements and uncertainties related to costs or work teams.

2- Identifying the main requirements and variables during the development process that could affect, in one way or another, the quality of the final product.
3- Identify and set frequency distributions for the selected variables.
4- Building a simulation model that includes the introduction of variables and their probability distributions, which in turn represents the dynamic model of the project.
5- Create a generator to generate random samples, as the Monte Carlo method allows the generation of many random samples from the probability distributions of variables.
6- Evaluating and analyzing the results of the simulation process.
7- Iterate and improve the model and plan to mitigate potential risks.

Monte Carlo simulation is a powerful tool for improving the efficiency of project management due to its approach based on risk assessment by generating many series of mathematical statistics and predicting potential risks and its effectiveness in mitigating those risks because it is of great importance in dealing with uncertainty, which allows decision makers to have confidence in Make bolder decisions by dealing with the potential for risk and uncertainty. [3]

Purpose: The purpose of this work is to study the theoretical aspects of using risk simulation in agile software development using the Monte Carlo method.

Objectives: Identify risk factors, integrate probability distributions, and develop a Monte Carlo simulation model to calculate risks in agile software development.

2. Risk of Agile Software Development

Risk is an uncertain event that hurts the project's success [4]. Agile development is considered one of the methodologies used for software management and development, which focuses on building the program in multiple stages and within

specific periods. Each stage produces a program distinguished from the previous one by several additional tasks, and each stage is considered an actual program [5].

What distinguishes agile software is that it is an iterative software development process, and each iteration has a specific scope and period. These periods may cause an increase in the possibility of risk due to unequal priorities for delivering the project in a short time. Adapting to changing requirements is a feature of agile software, and this feature makes uncertainties affect the project results, which increases the probability of risk. Hence, risk management in agile software can include many risk factors that must be considered, including [6].

1- Teamwork: This leads to the inability to communicate with work teams, which causes a delay in delivering the project on time.
2- Participation: Decision-makers' participation in the project life cycle is necessary to ensure their requirements are met. Understanding the requirements and meeting their needs accurately leads to the risk of the product being delivered and working efficiently, as the involvement of stakeholders throughout the project period is essential to meet their needs and feedback consistent with the project objectives.
3- Final delivery: The deadlines for delivering the product are among the most essential and significant risks facing the agile software development process due to the large number of variables.

Software development based on agile methodologies is subject to many potential risks due to fluctuations in requirements, technical complexities, the possibility of restrictions on the resources used, and the dynamics of the teams' work. All of these factors contribute to uncertainty in project implementation management.

From the above, development process management teams must prioritize teamwork practices and effective communication [7]. This effectiveness and risk reduction can

be achieved by holding meetings between development teams and stakeholders. Participation strategies can help stakeholders mitigate risks and ensure that The development process is moving toward good delivery. [8]

3. Risk factors within Agile projects

Many software organizations tended to adopt agile methods for software development, but they noticed failures in the development process that were reported using the agile approach [9]. Due to the iterative nature of the agile approach, a lack of requirements and unclear scope are considered to be among its main risk factors. Therefore, these institutions sought to mitigate the risks associated with agile development and management of the development process, as it was noted that there are multiple risks, for example, risks related to the time of the development process, the time of the delivery process, the financial resources allocated to the development process, the interaction of work teams, and the dynamics of communication between them or with Stakeholders. Here, we explain a group of essential factors that may affect the development process of agile methods.

1- Risks related to stakeholders: Through the cooperation of work teams with project owners or end users in conducting interviews or workshops and listening to the project requirements.
2- Risks associated with time management: One of the most critical areas in the development process is the process of delivering the product on time and evaluating work levels and their impact on the various performance measures of the work teams by running the simulation process with different requirements scenarios and time limits. Stakeholders can obtain A clear vision of the potential risks in time management. This allows them to make better decisions in determining the time required to complete the task by the work teams to balance the material costs and the period required to complete the task.

3- Risks associated with changing requirements: By considering the uncertainty of requirements for stakeholders or end users, Monte Carlo simulation is used to improve the process of predicting changing requirements during the development process, where stakeholders can evaluate the likely results with work teams and develop more robust and realistic forecasting models. This allows for better forecasting of requirements and improves planning for managing the development process and allocating the necessary resources.

By considering these factors that may affect the development process, agile development teams can build a clear and comprehensive understanding of the potential risks and predict them, which enables them to address these risks and challenges that they may face early and proactively and mitigate them during the development process period, [10]as The iterative nature of the agile approach allows for continuous adaptation and improvement of its vision and strategy in challenging potential risks.

4. Risk calculation simulation model

The Monte Carlo method for any project's success depends on the input data's accuracy and the simulation model's rigor, precision, and realism. To create any model that simulates risks, we take into account cases of uncertainty and variation in project requirements, and this is what the Monte Carlo method provides us with to calculate software development risks. Figure (1) shows a guide to applying the Monte Carlo method to analyze software development risks.

Building a simulation model first requires identifying the variable requirements in the development process that contribute to the success of any project and that may be subject to uncertainty, such as cost variables, lack of resources, changing teams, etc.). [11] Probability distributions are then determined to estimate the probability of different outcomes for each variable stage. To build a model that represents the flow

of project tasks for the development process using Monte Carlo theory, Triple statistical data are used based on previously recorded data or expert judgment. This model simulates the development process, which depends mainly on the specified variables and their probability distributions. It is characterized by generating random samples of probability distributions from the variables specified during the simulation and repeating them. For each simulation process, the results are evaluated based on random samples, which may include evaluating the project period, cost, or scope... When looking at the distributions of project results, the results are analyzed from variables that are characterized by uncertainty, such as standard deviation calculations and percentages; through the results obtained in the analysis process, management efforts focus on reducing risks and uncertainty, and high-risk areas are identified through variation in results. Depending on the results obtained, improving the simulation model by adjusting the probability distributions or incorporating new requirements to reach the best results is possible. Communication and iteration in agile software development allow project teams to participate with stakeholders in discussing the results obtained from the simulation model built by the Monte Carlo method. It facilitates reducing risks through the project's progress and repeating the model to make the best decisions [12].

To enhance the accuracy of risk assessment and reduce the possibility of uncertainty, the simulation model is constantly updated with each development in terms of requirements or variables in the life of the development process.

International Journal of Computers and Informatics (IJCI)
Vol. (3), No. (3)
المجلة الدولية للحاسبات والمعلوماتية
الإصدار (3)، العدد (3)
March 2024

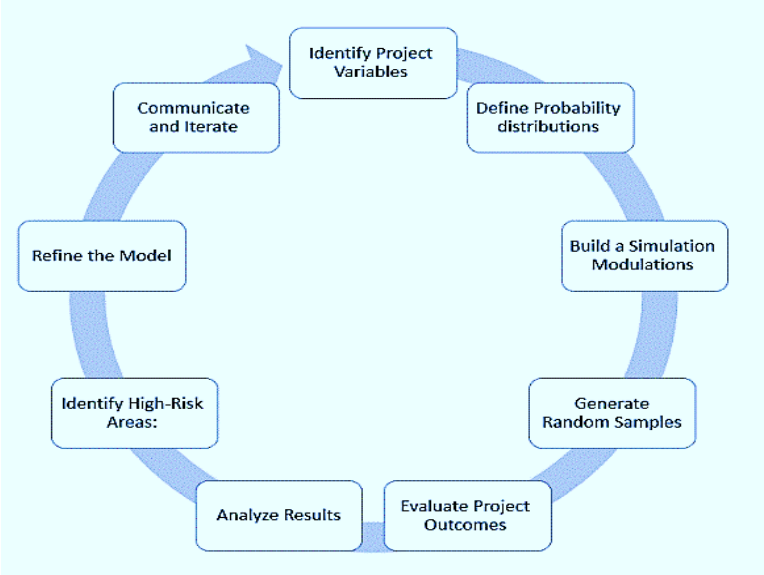


Figure [1]: Monte Carlo method to analyze software development risks

Risk Distribution according to the Monte Carlo Model

To distribute risk calculations using Monte Carlo theory in agile software development, a hypothetical table that includes multiple iterations and focuses on task duration variables using the triangular distribution can be built. See table (1)

Since the task duration includes (optimistic, most likely, and pessimistic), these estimates include the minimum, maximum, and most likely time each task may take.

Random value: It is calculated by a random number generator.

Project results: represents the sum of all task durations for each iteration.

After applying the simulation for sufficient iterations, the results can be analyzed to understand the distribution of potential project outcomes and calculate statistical metrics.

Table (1): Calculation of risks in agile software development according to Monte Carlo theory

Iteration	Task Duration (Optimistic)	Task Duration (Most Likely)	Task Duration (Pessimistic)	Random Value	Project Outcome
1	5 days	8 days	15 days	Randomly sampled	Sum of all task durations
2	6 days	9 days	14 days	Randomly sampled	Sum of all task durations
...	...	...	...	...	...
1000	4 days	7 days	12 days	Randomly sampled	Sum of all task durations

5. Uncertainty Model

In any Monte Carlo simulation process, it is necessary to develop a model of uncertainty that is related to new requirements or variables that arise during the period of the development process that contribute to uncertainty, such as the availability of resources, project scope variables, or any other process related to the development process. [13]

Through Monte Carlo simulation, random samples are selected from the probability distribution within a specific range of possible, most likely, and pessimistic estimates. To calculate the project results, these samples are taken, which may include collecting periods of individual or team tasks, and these results are recorded to complete the process of analyzing the results.

After completing the simulation.

1- Mean (Average)

$$\text{Mean} = \frac{\text{Sum of all outcomes}}{\text{Number of iteration}}$$

2- Standard Deviation

$$\text{Standard Deviation} = \sqrt{\frac{\sum(\text{Squared differences from the mean}}{\text{Number of iterations}}$$

3- Percentiles

Examine percentiles (eg,10^{th},50^{th} ...) to understand the range of potential outcomes.

The various project variables (task durations, availability of resources, speed of work) or any other factors related to the development process that contribute to uncertainty are calculated, and the probability distributions for each specific variable are determined according to mathematical equations to model uncertainty solutions according to the following:

- Triangular Distribution

$$\frac{2(x-a)}{(b-a)(c-a)} & \text{for } a \leq x \leq c \\ \frac{2(b-x)}{(b-a)(b-c)} & \text{for } c \leq x \leq b \\ 0 & \text{otherwise} \end{cases}$$

Where a,b, and c are the minimum, maximum, and most likely values .

- Normal Distribution

$$P(\times) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\times-\mu)^2}{2\sigma^2}}$$

Where μ is the mean and σ is the standard deviation

- Uniform Distribution

$$P(x) = \frac{1}{b-a}$$

International Journal of Computers and Informatics, London Vol (3), No (3), 2024
https://doi.org/10.59992/IJCI.2024.v3n3p3 E-ISSN 2976-9361

6. Monte Carlo Simulation Model for Risk Calculation in Agile Software Development

The Monte Carlo method has gained significant traction in evaluating software development risks because of its importance in managing and mitigating challenges effectively. There are several steps to follow a Monte Carlo simulation loop to evaluate risks in agile software development. In the beginning, variables and their probability distributions, representing aspects of uncertainty, must be identified. Certainty, random samples are generated for each variable based on probability distributions. The results are calculated based on the random samples entered into the simulation process and the generation and calculation process is repeated several times to reach the best results.

By generating random samples, we assume that we have variables (X1,

PO= X1+X2+X3+…….. +Xn.

Designing accurate and comprehensive mathematical equations to calculate risks in agile software development is difficult due to the dynamic and iterative nature that characterizes agile development methodologies. However, we provide a representation of certain aspects of the process in the form of mathematical equations:

Triangular distribution of task duration

Triangular distribution of task duration:

$$P(x) = \frac{2(x-a)}{(b-a)(c-a)}$$

$$F(x) = \begin{cases} 0 & \text{for } x \leq a \\ \frac{(x-a)^2}{(b-a)(c-a)} & \text{for } a < x \leq b \\ 1 - \frac{(c-x)^2}{(c-b)(c-a)} & \text{for } b < x \leq c \\ 1 & \text{for } x > c \end{cases}$$

Where: a is the optimistic estimate, b is the most likely estimate, and c is the pessimistic estimate.

Expected Task Duration:

Expected Duration=$\frac{a+4b+c}{6}$

Standard Deviations = $\frac{c-a}{6}$

Project Outcome$_i$= $\sum_{j=1}^{n} TriangularDistribution(a_j, b_j, c_j$

Where: i is the iteration index, n is the number of project variables, and aj,bj,cj are the parameters of the triangular distribution for the j-th project variable.

Mean project Outcome= $\frac{1}{N}\sum_{i=1}^{N} Project\ Outcome_i$

Standard Deviations of Project Outcome = $\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\text{Project Outcome}_i - \text{Mean Project Outcome})^2}$

Where:

N= It is the number of simulation iterations.

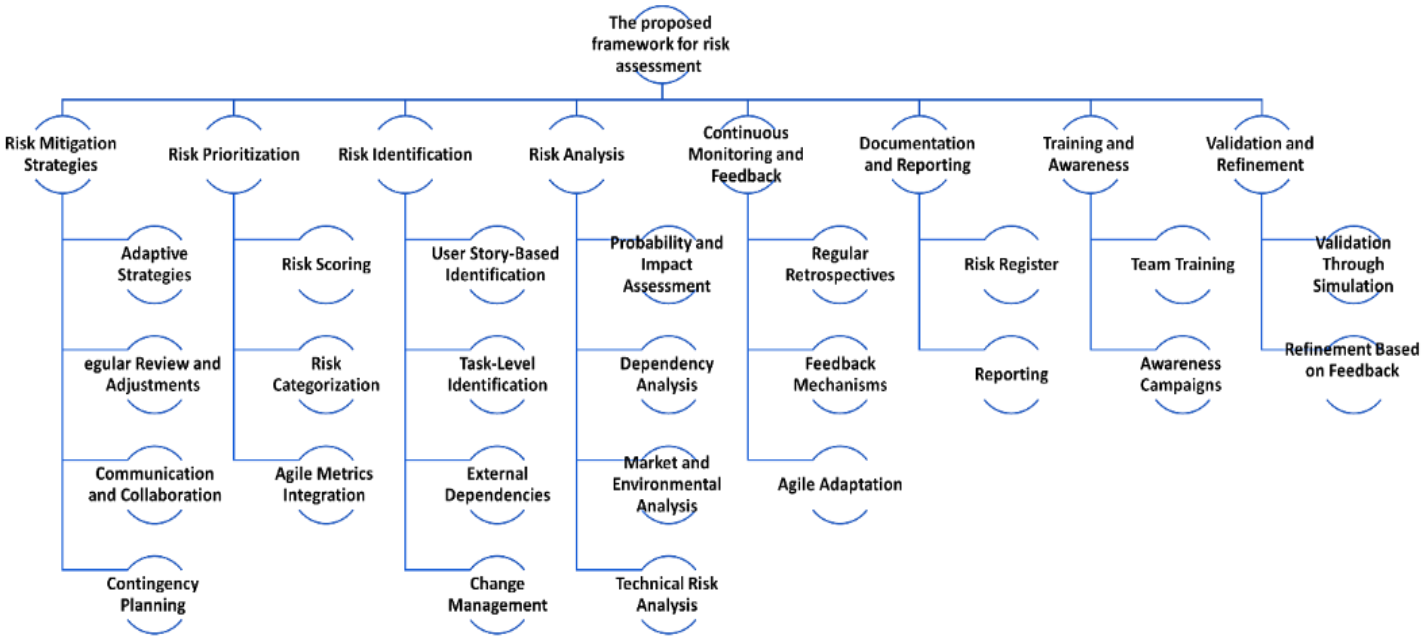
These equations provide a simplified representation of some of the nuances in the overall approach to risk calculation. In practice, performing Monte Carlo simulations for agile software development often involves detailed modeling, coding, and simulation techniques tailored to the specific context of a software development project, which may require context-specific conditions such as probability distributions and different modeling techniques.

7. Proposed Framework

Software development faces many risks, which result from unexpected events or circumstances that may positively or negatively impact the development process. Therefore, when starting any project, it is necessary to develop scenarios to manage potential risks [14]. In the form of (2), we put together a proposal for managing a set of scenarios for facing risks in agile software development. They are distributed according to specific strategies to mitigate risks, including strategies for adapting to changing requirements and regular review of all requirements and modifications required by stakeholders through communication and agreement to develop plans. A well-studied scenario to avoid the risk. A scenario for developing a prioritization of risks based on the size of the impact of these risks and the probability of their occurrence, recording the potential risks, and classifying them according to their priority based on the degree of their impact. A scenario for knowing the risks based on determining the type of potential risk, studying the history of these risks, and determining their levels of importance. And the possibility of managing to avoid these risks. Risk analysis scenarios are used to evaluate situations of risk and uncertainty to reduce their potential impact on the development process and analyze the dependency of these risks. Feedback scenarios through an organized mechanism to review the opinions of users and stakeholders, monitor them continuously, and adapt them to the requirements of agile methodologies. Documentation and

International Journal of Computers and Informatics (IJCI)
Vol. (3), No. (3)
IJCI
March 2024
المجلة الدولية للحاسبات والمعلوماتية
الإصدار (3)، العدد (3)

preparation of reports on potential risks and recording them. Training and awareness of work teams on the importance of mitigating risks in the development process and verifying them. Improvement process based on user feedback perspectives, and changeable requirements.



Figer [2]: Proposed framework for risk assessment

8. Conclusion

In the agile development process, Monte Carlo simulation provides a systematic approach to assessing and managing risks in managing the chain of changing requirements during the development process by considering uncertainties in requirements management and predicting variables during the agile development cycle, allowing stakeholders to make crucial decisions and develop their outlook and its strategy to improve the efficiency of the development process. The Monte Carlo

method and its use in developing agile software projects also provide its efficiency in managing changing and recurring requirements, assessing the risks associated with the physical aspects, distributing the tasks of work teams, and predicting the requirements and goals set for the development process to reach the best final product.

Using the Monte Carlo method in the agile development approach helps better understand the uncertainties associated with the development process and mitigate risks very effectively if new data and information are available while updating ongoing requirements as the work progresses.

References

1- Tawunwoot, W., & Chomngern, T. (2023, November). Prototype Structure Design Process Using Information Flow Diagram (IFD) for Mobile Application Learning System. In 2023 7th International Conference on Information Technology (InCIT) (pp. 492-497). IEEE.
2- Magennis, T. (2012). Managing software development risk using modeling and Monte Carlo simulation. In Lean Software and Systems Consortium Conference, Boston.
3- Mahadevan, S. (1997). Monte Carlo simulation. Mechanical Engineering-New York and Basel-Marcel Dekker-, 123-146.
4- Marle, F. (2020). An assistance to project risk management based on complex systems theory and agile project management. Complexity, 2020, 1-20.
5- Williams, L. (2010). Agile software development methodologies and practices. In Advances in computers (Vol. 80, pp. 1-44). Elsevier.
6- Al-Husseini, K. A. O., & Obaid, A. H. (2020). Interaction between project tasks and risk management tasks in software development. Periodicals of Engineering and Natural Sciences, 8 (4), 2300-2308.
7- Linderman, K., Schroeder, R. G., & Sanders, J. (2010). A knowledge framework underlying process management. Decision Sciences, 41(4), 689-719.
8- Al-Husseini, K. A., & Obaid, A. H. (2019). Analysis and risk management in software development using the logical-algebraic model. In CEUR Workshop Proceedings (Vol. 2475, pp. 241-248).

9- Shrivastava, S. V., & Rathod, U. (2017). A risk management framework for distributed agile projects. Information and software technology, 85, 1-15.
10- Tam, C., da Costa Moura, E. J., Oliveira, T., & Varajão, J. (2020). The factors influencing the success of on-going agile software development projects. International Journal of Project Management, 38(3), 165-176.
11- Jonkers, R. K., & Shahroudi, K. E. (2020). A design change, knowledge, and project management flight simulator for product and project success. IEEE Systems Journal, 15(1), 1130-1139.
12- Mun, J. (2006). Modeling risk: Applying Monte Carlo simulation, real options analysis, forecasting, and optimization techniques (Vol. 347). John Wiley & Sons.
13- Arunraj, N. S., Mandal, S., & Maiti, J. (2013). Modeling uncertainty in risk assessment: An integrated approach with fuzzy set theory and Monte Carlo simulation. Accident Analysis & Prevention, 55, 242-255.
14- Al-Husseini, K. A. O., & Obaid, A. H. (2018). Development of Risk Management Tools in Question-Answering Based Software Design Environment. International Journal of Computer Science and Mobile Computing-IJCSMC, 7(6), 165-174.