RESEARCH ARTICLE OPEN ACCESS

A Road Map Journey of Software Risk Management for Excellence Through Simulation: A Systematic Review

Alankrita Aggarwal[1], Navdeep Kaur[2], P.K.Suri[3]
[1](Dept. of Computer Engineering, PTU University, Jalandhar, Punjab, India
Email: alankrita.agg@gmail.com)
[2](Department of CSE, SGGSWU, Fatehgarh Sahib, Mohali, Punjab, India.
Email: *drnavdeep.iitr@gmail.com*)
[3](*HCTM Technical Campus, Kaithal, India.*
Email: pksuritf25@yahoo.com)

ABSTRACT

The paper throws light on the successive journey of risk management during the previous years along with the role of simulation, concepts and tools to get optimize solution of reducing and accessing risk. The objective of the paper is to explain the concept of risk and to develop its role within the software development process and to introduce the use of risk management as a means of identifying & controlling risk in software development. There are various simulation techniques through which can be studied.

Keyword: Risk, Risk management, Assessment, Simulation, Modeling

I. INTRODUCTION

The introduction of risk starts with the basic definition of risk as –it is the possibility of suffering, harm or loss .secondly it can be a chance of danger, undesirable outcome or missed opportunity as well as the injury of software. It can be described as probability and severity of adverse effect. Question arises is why is the software world interested in risk? Next definition of Risk Management is a practice with processes, methods, and tools for managing risks in a project which is the key point of the paper. Most obvious factor is cost factor which can be reduced by the project managers so that they result in are good risk managers. The basic process of dealing with the risk is to identify the risk, analyzing its implications, determining treatment methods, monitor performance of treatment methods, Techniques & heuristics for the identification, analysis, treatment & monitoring of risk. Risk management is a project management tool to assess & mitigate events that might adversely impact a project, thereby increasing the likelihood of success [1] [2].

1.1 Sources of software risk factors or aspects which are likely to have a negative impact on the project performance or Uncertainties which affect the project performance .ranges from – technology to cost ->Software->Schedule->Hardware->People to the whole system [3].

1.2 There are various reasons of most software projects go wrong.

Some of the are Inadequate understanding of customer needs, poor requirements documents ,Poor requirements management, Poor or no architecture/design ,Code first and ask questions later, poorly understood legacy design/code, no peer reviews to catch problems early, inexperienced or incapable personnel, ineffective testing – misses serious defects

II. TYPES OF RISKS

2.1 a)**Software Project Risks** like Resource constraints, external interfaces, supplier relationships, nonperforming vendors, internal politics, interteam/ intergroup coordination problems, inadequate funding. b) **Software Process Risks** includes Undocumented software process, lack of effective peer reviews, no defect prevention, poor design process, poor requirements management, ineffective planning.c**) Software Product Risks** results from lack of domain expertise, complex design, poorly defined interfaces, and poorly understood legacy system, vague or incomplete requirements [6] [7].

Questions that must be answered to identify type of risk:

- Is the technology new to your organization?
- Are new algorithms, I/O technology required?
- Does the application interface with new software?
- Is a specialized user interface required?
- Are you using new software engineering methods?
- Are you using unconventional software development methods, such as formal methods?
- Are there significant performance constraints?
- Is there doubt the functionality requested is "do-able" [10] [15].

2.2 Software risk management Process: Software risk management Process revolves around risk knowledge base and starts from risk identification, risk analysis, risk planning, risk tracking, risk resolution, risk about risks.

2.2.1 Risk Identification: It is a process which starts from exposing risk as well as conducting walkthroughs where participants come without preparation to attend the meeting. Presenter in this case is the author of the product. Walkthroughs are mainly used for communication purpose.

Risk Exposure = Probability x Consequence

2.2.2 Risk MMM (Risk Mitigation, Monitoring, and Management : is about how we can avoid the risk and what factors can we track that will enable us to determine if the risk is becoming more or less likely and in parallel what contingency plans do we have if the risk becomes a reality respectively

2.2.3 Risk Mitigation: is all about decreasing the probability of risk event happening or the impact of it happening. For this a risk table has to be built by estimating the probability of occurrence, estimating the impact on the project on a scale of 1 to 5, where

1 = low impact on project success &

5 = catastrophic impact on project success.

In the end sort the table by probability.

Building a Risk Table:

Table 1: Building a Risk Table

Risk	Probability	Impact	RMMM
			Risk Mitigation Monitoring

Risk-exposure: quantity is an effective technique for risk prioritization, Assess risk probabilities & losses on a scale 0-10, multiply probability by loss to determine exposure.

Table 2: Building a Risk Exposure table

Unsatisfactory Outcome	**Probability of unsatisfactory outcome**	**Loss caused by unsatisfactory outcome**	**Risk Exposure**
Software error loses key data	3-5	8	24-40
Processor memory	1	7	7

RE = Probability (UO) * Loss (UO), where UO = Unexpected outcome

Given the example above, the Risk Exposure is 10% x $100,000 = $10,000 and 10% x 2 calendar week = 0.2 calendar week. Comparing the Risk Exposure measurement for various risks can help identify those risks with the greatest probable negative impact to the project or product and thus help establish which risks are candidates for further action [5] [8] [10].

2.2.4 Quantification of Risk: is to allowing solution ideas to be evaluated more critically, feedback on risks we missed, feedback on impact of risks we anticipated, us to allocate resources to deal with risks, to determine whether a risk is acceptable and encourages design awareness of risk

2.3 Analysis of risks: For analyzing the risk several questions or a questionnaire is be built to find out the exact location of the housing of risk. For Example: How severe is the consequence? How likely is the occurrence? Is the risk exposure acceptable? How soon must the risk be dealt with? What is causing the risk? Are there similarities between risks? Are there dependency relationships? What are the risk drivers?

2.3.1 Activities for analysis of risks: Grouping: Eliminate redundant risks; Combine related risks; Link dependent risks.

Determining risk drivers: Underlying factors that affect severity of consequence, may affect estimation of probability, consequence, risk exposure which increases understanding of how risks can be mitigated.

Ranking: Order of likelihood, consequence, exposure, time frame.

Determining root causes sources of risk: Old-fashion root cause analysis, identify common root causes e.g. no. of persons leaving the project.

2.2 Planning: Resolution Strategies

a) **Risk Avoidance:** Prevent the risk from occurring, reduce probability to zero

b) **Risk Protection:** Reduce the probability and/or consequence of the risk before it happens

c) **Risk Reduction:** Reduce the probability and/or consequence of the risk after it happens

d) **Risk Research:** Obtain more information to eliminate or reduce uncertainty

e) **Risk Reserves:** Use previously allocated schedule or budget slack

f) **Risk Transfer:** Rearrange things to shift risk elsewhere (to another group, for example)

2.4.1 Planning: Activities

Specify scenarios: How would we be able to tell it is really happening? Define quantified threshold for early warning what to monitor, when we consider the risk to be happening. Develop resolution alternatives: Ways to eliminate mitigate or handle the risk. Select resolution approach, what has the best Return on Investment (ROI)? Specify risk action plan, Document decisions

2.4.2 Tracking:

Monitor risk scenarios: a) Watch for signs of a risk scenario occurring, b) Compare indicators to trigger conditions: Watch indicator metrics – do they satisfy trigger conditions. c) Notify stakeholders: Let stakeholders know the risk is happening; execute action plan, Collect statistics: Update risk database. [22].

III. EVALUATING RISK MANAGEMENT WITH THE USE OF SIMULATION

3.1.1 STEPS IN SIMULATION STUDY:

First and foremost step is to optimize a solution and to formulate a problem. Former process requires construction of a base model construction .Data is collection through various methods, methods and tools are known for model programming, validation of the programming and models are to be needed, after validation simulation run and analysis is done .Documentation and implementation is to done in the end for the construction of simulation study [16] [18].

3.1.2 Simulation for Managing Software Development Risks: Researching common and significant software development risk factors and their effects, after adopting a base model, simulating the selected factors. Design simulator as a tool specifically for Risk Management.

3.1.3 Simulation features for management science: There are one or more stochastic input processes that are specified via probability and from which synthetic realization can be generated and there should be a logical model that updates the system state upon the occurrence of some discrete events. From the logical model some output processes typically ordered by some concept of time, that represent the system behavior of interest to the analyst.

Probabilistic cost/schedule estimation to assess Risk: i)Two common methods of probabilistic estimation involve inputting PERT distributions and/or using Monte Carlo simulation. ii) The PERT probability distribution used in many cost/schedule estimation tools in a form of a Beta distribution. It is specified by three parameters: a minimum value, a most likely value, and a maximum value

An approach to simulate risk factor: One of the proposed purposes for software process simulation is the management of software development risks, usually discussed within the category of project planning and management. However, modeling and simulation primarily for the purpose of software development risk management has not been explored and is quite limited. For Example - an approach to simulate Low Morale - a risk factor, to analyze its effect on certain software development risk management activities [19].

Project Risk Assessment Approach for schedule delays: Almost every large project is subject to schedule delays, and furthermore schedule delays often translate into cost overruns and a project risk assessment is performed to identify and mitigate potential contributors to cost increases and schedule delays. The assessment is used to establish schedule buffer to contain potential overruns.

IV. LITERATURE SURVEY

Survey of all the Literature starts with the review of all the previous papers published during the past years.

In [13], authors proposed that software process simulation is the management of software development risks related within the category of project planning/management. They also describe an approach to modeling risk factors and simulating their effect as a means of supporting certain software development risk management activities.

The management of risks is a central issue in the planning and management of any venture. In the field of software, Risk Management is a critical discipline. Risk management is a discipline for living with the possibility that future events may cause adverse effects. Risk management partly means reducing uncertainty [2].

According to the Technical Report published by Roger L. Van Scoy from CEI, Carnegie Mellon University, Pittsburgh, Pennsylvania states that what risk is and everyone has an intuitive understanding of risk; it surrounds us in the world. But how can understanding risk help programs be more successful. For that purpose we need to understand that a risk is not a problem.. To be technically precise, there are two factors that comprise a risk- firstly the Probability or likelihood that it will occur and secondly the loss resulting from its occurrence. Risk is a part of any activity and can never be eliminated, nor can all risks ever be known. Risk in itself is not bad; risk is essential to progress, and failure is often a key part of learning. But we must learn to balance the possible negative consequences of risk against the potential benefits of its associated opportunity. There is a possibility that the application of software engineering principles and theory will lead to product fail to yield the right product.

In [12], authors argued that a system oriented approach based on functional abstraction rather than structural decomposition is needed. Therefore task analysis focused on action sequences and occasional deviation in terms of the human errors should be replaced by a model of behavior shaping mechanism in terms of human errors should be replaced by a model of behavior shaping mechanisms in terms of work system constraints, boundaries of acceptable performance and subjective criteria guiding adaptation to change.

In [7], authors describe all the sources of uncertainty and risk, their implications for software organizations, and how risk and uncertainty can be managed. Specifically, they assert that uncertainty and risk cannot be managed effectively at the individual project level. These factors must be considered in an organizational context.

In [14], authors stated a stochastic simulation of risk factors potential effects for software development risk management so that the risk factors can be identified and can have a minimum value in the lower range i.e. starting from 0-5.

The software teams need to manage risk and she also tried to concentrate that various tools like Gantt charts, pert can be used to evaluate the risks [23].

In [12], authors taken an approach taken in this study is similar to the benchmarking methodology developed by Camp, Balm and looked at current practice in the more successful organizations and attempted to identify what it is that they do that others do not. This approach can be extended to other aspects of the risk management process, such as assignment of roles and responsibilities, timing and frequency of risk management activities, and amount of effort applied. It could also be applied to the benchmarking proponents advocate, to many aspects of the organization.

In [22], authors provide a methodological framework to identify, prioritize, access, and manage risks at different scenarios. 1. Scenario identification, 2. Scenario filtering 3. Bi-Criteria Filtering 4. Multi Criteria Evaluation 5. Quantitative Ranking 6. Risk Management 7. Safeguarding against missing critical items 8. Operational Feedback [22].

Hany H. Ammar, Sherif M. Yacoub states that there are architectural level risks based on components like connectors which calls one module to another. These risks can be described as subjective risks. Severity analysis is performed using Failure Mode and Effect Analysis (FMEA) as applied to architecture models. Based on analysis scenarios, a risk assessment model has been developed that represents components, connectors, component risk factors, connector risk factors, and probabilities of component interactions. Also develop a risk analysis algorithm that aggregates risk factors of components and connectors to the architectural level. Using the risk aggregation and the risk analysis model, the overall risk factor of the architecture as the function of the risk factors of its constituting components and connectors are identified [9].

Donald Reifer stated in his article differentiates between the latest Internet and intranet development climate with earlier releases and identifies the 10 greatest risks. He also emphasized on Software engineers need to strike a balance between the technologies that many in the Internet world depend upon, on the one hand, and the tried and true processes that promote risk management and help us achieve business goals, on the other. The just do it attitude of the past few years has done as much harm as good to the software engineering profession. While a few good practices have emerged for Internet development, many time-tested software engineering practices were jettisoned based on the faulty belief that teams don't have time to put these practices to work.

In [21], author stated that the software to be released based on the performance measurement may be considered to be in a number of states of deterioration. The performance measurement work of the software is inspected after a regular interval of time say weekly and is classified as being in one of the states. Each state is considered as functionally independent. The evaluation is carried out using Markov analysis which looks at a sequence of states and analyses the tendency of one state to be followed by another, after each release the software restored to a state having increased operating efficiency. This Markov process is stochastic in nature which has the property that the probability of transition from a given state to any future state depends only on the present state.

Many simulation methods for software engineering have been proposed in [4] [5]**.** But again all these are conventional methods and don’t take component based nature of a software into consideration. In [18], authors suggested that the Software process simulation modeling (SPSM) has been emerging as a promising approach to address a variety of issues in software engineering area, including most important is risk management Discrete event and system dynamics as well as hybrid simulation are widely used in real world management practices. In [17], authors used a simulation technique to know the potential effect of creeping user requirements on the project and the substantial progress achieved in the areas of requirement elicitation, analysis and specification etc.

In [20], authors revealed that IT projects are always over budget, behind schedule and unreliable as it is a complicated process which involves many parties with different expectations. Delay in one component can easily affect everything else. Risk management does not guarantee success but has a primary role of identifying and responding to potential issues with sufficient lead time to avoid crisis situation.

Claire Le Goues, ThanhVu Nguyen, Stephanie Forrest states the research advances in debugging include replay debugging and cooperative bug isolation. Trace localization, minimization, and explanation projects also aim to elucidate faults and ease repairs. These approaches typically narrow down a large counterexample backtrack (the error symptom) to a few lines (a potential cause).However, a narrowed trace or small set of program lines is not a concrete repair. Second, GenProg can theoretically work on any detected fault, not just those found by static analysis tools that produce counterexamples. Finally, these algorithms are limited to the given trace and source code and can thus never localize the “cause” of an error to a missing statement, adding or swapping code to address a missing statement is necessary for many of our repairs [26].

Abdullah A. Alabdulkarim, Peter D. Ball, Ashutosh Tiwari concludes that research on the application of simulation to support maintenance system design and operation. Simulation is inherently capable of providing significant insight into maintenance systems in the same way it has been proven for the value adding processes in manufacturing systems. The approach used was to examine peer reviewed literature in a structured and disciplined way to establish the connection between simulation and maintenance and often translate into cost overruns. The assessment is used to establish schedule buffer to contain potential overruns.

IV. CONCLUSION

From the above literature the acquired information is that a risk management policy is a pre-requisite of any software development process. As ever-increasing complexity and increasing demand for faster, bigger and better, the software industry is a high risk business. When teams don't manage risk, they leave projects vulnerable to factors that can cause major rework, major cost or schedule over-runs, or complete project failure. Adopting a Software Risk Management Program is a step every software manager can take to more effectively manage software development initiatives. Based on a positive, proactive approach, risk management can greatly reduce or even eliminate the need for crisis management within our software projects. Further any simulation technique can be used to know the potential effect of creeping user requirements on the project. The substantial progress has been achieved in the areas of requirement elicitation, analysis and specification.

REFERENCES

[1]. Risks to the Public in Computers & Related Systems http://catless.ncl.ac.uk/Risks

[2]. P. G. Neumann, "Computer Related Risks", *ACM Press, 1995*

[3]. B. W. Boehm, "Software Risk Management", *CS Press, 1989*

[4]. Tom Gilb, "Principles of Software Engineering Management", *Addison-Wesley, 1998.*

[5]. B. W. Boehm, "Software Risk Management: Principle and Practices", *IEEE Software, 8(1), 1991, 32-41.*

[6]. Roger Pressman, "Software Engineering: A Practitioners Approach", *McGraw-Hill, 5th edition, 2010.*

[7]. Ian Sommerville, "Software Engineering", *Addison Wesley, 6th Edition, 1999.*

[8]. E.Hall, "Managing Risk, Methods for Software Systems Development", *Addison Wesley, 1997.*

[9]. D.Karolak, "Software Engineering Management", *IEEE Computer Society Press, 1996.*

[10]. B. Boehm, "Software Engineering Economics", *Prentice Hall, 1981.*

[11]. L. Putnam, "Software Cost Estimating and Life-cycle Control: Getting the Software Numbers", *IEEE Computer Society Press, 1980.*

[12]. B. Londeix, "Cost Estimation for Software Development", *Addison-Wesley, 1987.*

[13]. Donald Reifer, "Ten Deadly Risks in Internet and Intranet Software Development", *IEEE Software, 2002.*

[14]. Dan X. Houstna, Gerald T. Mackulak, James S.,"Stochastic Simulation of risk factor potential effects for software development risk management", *Collofello in the Journal of Systems and software, 59, Elsevier , 2001, 247-257.*

[15]. P.K.Suri and Rachna Soni,"Stochastically simulating low morale- A risk factor effect on risk management", *Journal of Computer Science, 2(4), 2006, 370-372.*

[16]. Dapeng Liu,Qing Wang, Junchao Xiao,"The role of software process simulation modeling in software risk management: a systematic Review", *proceeding of 3rd international symposium on Empirical Software Engineering and Measurement, IEEE, 2009, 302-311.*

[17]. P.K.Suri ,Rachna Soni, Ashish Jolly,"Potential Effect of creeping user requirements on project management: A simulation Approach", *International journal of computer science and network security , 9(11), 2009, 255-261.*

[18]. Gillette Billy, "Operations Research", *Tata Mc Graw Hill Publishing Co.Limited, 2004.*

[19]. Roger L. Van Scoy, "Software Development Risk: Opportunity, not problem", *Technical report, 1992.*

[20]. Yacov Y. Haimes,Stan kalpan, James H.,"Lambert Risk filtering, ranking, and management framework using hierarchical holographic modeling Risk analysis ", *Wiley online library, 22(2), 2002, 383-392.*

[21]. Mohd Sadiq et al., "Software Risk Assessment and Evaluation Process (SRAEP) using Model Based Approach", *Proceedings of the International Conference on Networking and Information Technology, 2010, 171-177.*

[22]. Jens Rasmussen, Risk Management in a dynamic society: A modeling problem safety Science", *Elsvier Science Limited, 27(2/3), 1997, 183-213.*

[23]. Susan A Sherer, "The Three dimensions of Software Risk: technical, Organizational, and Environmental," *proceeding of 28th Hawaii International Conference on System Sciences, IEEE, 1995, 369-378.*

[24]. Ronald P.Higuera, David P. Gluch, Audrey J. Dorofee, Richard L. Murphy, Julie A. walker, and Ray C. William, " An Introduction to Team Risk Management (Version 1.0)", *Technical Report, Software Engineering Institute, 1994.*

[25]. Georgieva K. Et al., "Analysis of Risk Analysis Methods-A Survey", *LNCS-Springer-Verlag Heidelberg, 2009, 76-86.*

[26]. Jens Rasmussen, "Risk Management in a dynamic society: A modeling problem safety Science*", Elsvier Science Limited, 27(2/3), 1997, 83-213.*