

A Relative Performance Evaluation of Risk Analysis and Assessment Factor For Software Development Projects

T. Balaji ¹

¹ Assistant professor, Dept. of Computer Science, Govt. Arts College, Melur - 625106, India
bkmd_gacm@rediffmail.com

Abstract - The primary goal of this paper is an initiative approach for minimizing the **uncertainty and potential loss** associated with a software project. Another goal is to provide insights to support decision-making. The risk analysis and assessment is beneficial approach and often very important approach for corporate endeavors and smaller scale or very large scale projects development industry. Research paper can be reviews to analyses and assessments of risk in software development projects. It describes how to analyze and assessment of software risks on their projects. The risk management is very important for all software projects and so that the overall impression of different risks in design is minimized. Although the risk management is a best practice in the software development industry for reducing the surprising factor.

Keywords: Risk Analysis, Risk Assessment, Generic Risk Analysis, REF, Quantitative Analysis, Qualitative Analysis and SDIS.

1 Introduction

Risks are simply potential problems in various development organizations. A probability or threat of damage, injury, liability, loss, or any other negative occurrence that is caused by external or internal vulnerabilities, and that may be avoided through preemptive action. *David Hillson* says, “**Risk can be defined as uncertainty of matters**”. It can damage a project, but some uncertainties can help to solve the projects. According to *Gilbs* risk principle, “**If you don’t actively attack the rises, they will actively attack you**”. In order to successfully to manage a software development, we must learn to identifying, analyzing and controlling these risks. In software development concern, risk is a measurable the inability to achieve overall program objectives with defined cost, schedule, operational resources, budget and other technical constraints. A top-level approach to interact a risk analysis should be deeply integrated the software development design.

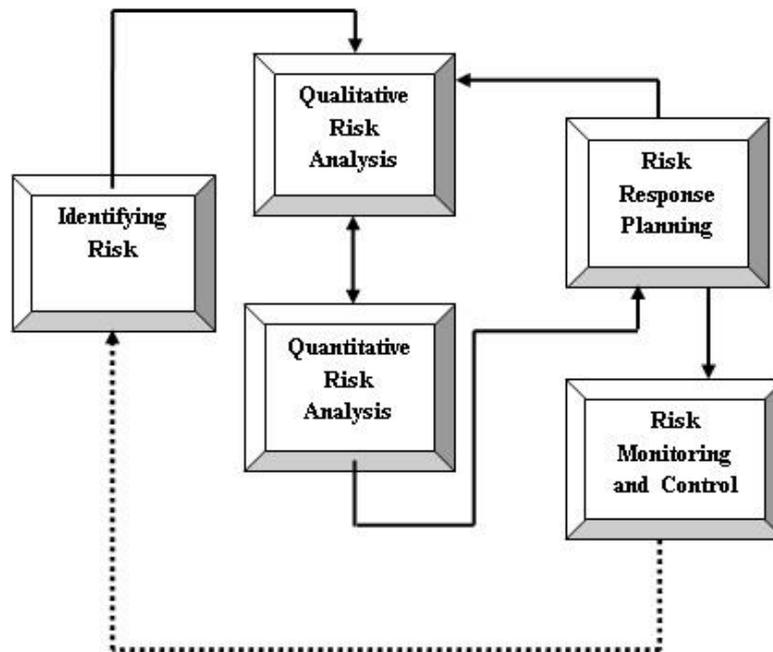


Figure 1. Risk Management System

The large software concern can't avoid the risks associated with the design and operation of a major software development or deployment project. The current definition of risk is "noun". They include:

- *Possible for suffering, harm or loss – danger.*
- *An element, factor, statistics or course involving uncertain danger.*
- *Possible of loss, injury, disadvantage or destruction.*

2 A Different Perspectives of Risk

It is very important to analyze software risk from the beginning to ending software development design. The whole idea of software risk management design is to innovative and think outside contained in the above. A software development project may be classified into various types of risk

2.1. Technological Risks

It includes problem with language, project size, project functionality, platforms, methods, standards or processes. This risk may result from the lack of experience and poor defined parameters, etc.

2.2. Project Risks

It includes inadequate configuration control cost, poor quality and fails in operation repeatedly.

2.3. Financial Risks

It includes cash flow, P/L account, capital and budgetary control, Income-Tax preparation, returns on investment constraints.

2.4. Commercial and Management Risk

The commercial risk is a final project may have lower user satisfaction. Lower user satisfaction means the product has low quality, function inadequately and has complex structures.

Management Risk includes a lack of planning, management experience and training, communications problems, organizational issues, lack of authoring and control problems.

2.5. Nature of Contract and Legal Risks

It includes changing requirements, market-driven schedules, health and safety issues, government regulation and product warranty issues, and registration issues, etc.

2.6. Personnel Risks

It includes staffing lags, experience and training problems, ethical and moral issues, staff conflicts and productivity issues, ego and emotional issues, etc.

2.7. Other Resources of Risks

In includes unavailability or late delivery of equipment and supplies, inadequate tools, inadequate facilities, distributed locations, unavailability of computer resources, slow response time, educational difficulties, lack of time, misunderstanding and misguidance, etc.

Some other areas are focused to risk other than software development industry considered the following problems:

- *Unavailability of development or testing equipment and facilities.*
- *Inability to acquire resources with critical skills.*
- *Unachievable performance of requirements.*
- *Problems with language communication and translations.*
- *Problems of products and internationalization.*

- *Technological approaches that may not work.*
- *Mixing any proportions of data in product design.*
- *The frequency is lot absolute for your conditions.*

3 Risk Analysis Factors

The purpose of analysis is to convert the data in to decision-making information. Analysis is a process of examine the risks in detail to determine the extent of the risks. Risk analysis is very important and tailoring in the software development design. Examining the change of outcomes with the modification of risk system. The risk analysis and practice in software development projects vary on several dimensions.

- Context and Size
- Circumstances and Complexity
- Uncertainty and Inadequate Issues
- Critical factor and lack of efficiency, etc.

In the risk analysis step, each risk is assessed to determine the following:

- **Likelihood** –the probability that the risk will result in a loss.
- **Impact** – The size or cost of the loss if the risks turn to the problem.
- **Timeframe** – The risk needs to be addressed.

A typical risk analysis involves several major activities that often include:

- Learn as much as possible about the analysis aimed.
- Discuss production issues surrounding the software.
- Perform impact analysis.
- Ranking and priorities of Risks.
- Develop a mitigation strategy.
- Report findings, etc.

Another important level of analysis is design level analysis in software development project. The design level analysis is knowledge intensive. A central activity in design level risk analysis is to build up a consistent view of the target system as a reasonable level of cost. The nature of software systems leads many developers and analysis to assume that software is sufficient for spotting design problems. Anyone involved with software development project knows to bringing successful completion can be difficult to achieve. The most frequently factors that contribute to the challenge include as follows:

- Poorly defined requirements.
- Time creep and priorities.
- Scope creep
- Unrecognized development process and methodology.
- Unrealistically low budgets, inadequate resources and limited time schedule.

The job risk management analysis is to identify, address and eliminate sources of risk before they become as to success of the project. Risk can be addressed at different levels:

1. **Crisis Management** - firefighting, address risks only have some problems.
2. **Fix on Failure** - Detect and react to risks quickly.
3. **Risk Mitigation** - Plan ahead of time to provide resources to cover risks.
4. **Prevention** - Implement and execute a plan of a project to identify risks and prevent the problems.
5. **Elimination of root causes** – Identify and eliminate factors that make it possible for risks.

The problems of risk analysis can also be addressed in several ways including:

- Expanding the list of generic risks in project development.
- Maintaining and focusing on the larger project goals and extending list of stakeholders or developers.

4 Generic Risk Analysis

Generic risk analysis is limits to the developer's perception. To identify the reasons of failure, understand and address the risks associated with different types of software projects. It is a key factor to project failure by constraining the developer's perceptions of the real project risks. For software development experts classifies the software such as system software, end-user software, commercial software, application software and real-time software, etc. The software development projects differ in number of significant ways. For example, real-time military system differs from commercial applications in their technical risks. The software development risks differ by the types of software projects.

Boehm defines the Risk Exploration Factor equation to help the quantitatively establish risk priorities. This measures the impact of risk in terms of the expected value of the loss. The Risk Exploration Factor(REF) is defined as the probability of an undesired outcome times the expected loss.

$$\text{REF} = \text{Probability of Uncountable Outcome} \times \text{Loss of Uncountable Outcome}$$

The list of risks is prioritized based on the results of our risk analysis. Since the prioritized list of risks is used to identify risks requiring additional planning and action. The extended risk factors remain "largely un expected areas in the software project risk management". For a software development project to succeed, the risk resolution analysis should consider the following:

- Delivered project types, consisting of sectors and applications.
- Developer's opinions.
- Different developer's expectations.

- Size and complexity of the software types and monitoring tools used by the project manager.
- Simply categories projects by the size of the code or duration in order to guide their risk management approach.
- Categories by the technological aspects.

These approaches do not much light on the risks of the delivered software development project. The risk analysis should also be outward looking and taking into account the overall environment within the software will be used and target application area. For example internet filter project and context of school changes types of risk that needs to be addressed in this software development. The result of this analysis must be added to development risk analysis for complete risk analysis for system development project.

5 Quantitative and Qualitative Risk Analysis

The quantitative risk analysis is a critical and also good judgment in software development. A quantitative risk developing matrices that can be used to describe the risks in terms of money lost, days over schedule, number of technological functionalities to meet. These quantities can be measured periodically to ascertain the risk in terms of Risk Exploration and Risk Leverage. There are two problems with this emphasis on quantifiable risk are:

- Exclusion of quantitative risk.
- Emphasis changes the risk perception of the developers.

The problem of addressing both quantitative a qualitative risk is global placement. For example, software development on research project, we recommend that best practice for balanced risk analysis based upon the following:

- Properly aligning the project goals with strategic intentions.
- Appropriate requirement analysis involvement in project developers.
- Equal emphasis on all aspects of analysis.

The quantitative and qualitative risk analysis recognized and assessed in software development projects. These analyses are different and both are necessary. The quantitative risk analysis methods can be reduced the number of software development failures in different perceptions, it is called Software Development Impact Statement (SDIS). The SDIS process was tested on software development in project organizations with different location, size, function, scope, development, methodology and technology level, etc. The quantitative risk analysis can be applied at every stage of software development and SDIS was tested successfully against every phase of software development. A SDIS risk analysis can be applied to any work analysis can be applied to any work product such as a work breakdown structure in a system's development.

6 Risk Assessment Technique

Risk management is the practice of assessing and controlling risk that affects the software project, process or product. Risk assessment is the systematic study of various elements of risks. The assessment of risk is often carried out by experts in the chosen field of activity. The basic concepts of assessing the risk managements are:

6.1. Goal

Risk is managed in reduction to a specific goal. It is clearly defined goal with measurable access criteria that the acceptable risk.

6.2. Uncertainty

There is always a degree of uncertainty in risk occurrence. The probability of risk is always greater than zero.

6.3. Loss

The loss can be either an undesirable outcome or a lost the opportunity.

6.4. Choice

Unless there is a choice, there is no risk management.

6.5. Intelligent Decisions

It made on awareness, the risks associated with choices and visibility into the risks at a project level.

6.6. Resolving Risk

The key is done by developing and executing a risk action plan to resolve the risks.

6.7. Preventing Problems

The resolution of risk prevents the problems of costly rework.

Role of risk assessment is to continuously identify, analyze, plan, track, control and communicate the risks associated with a software development project. Many techniques for risk assessment and analysis have been inspired by or adapted from the more software developed projects. The risk assessment is the possibility of suffering a loss but it can manage the organizations. The risk assessment is essential to progress

and failure is often a key part of learning. The software risk assessment technique based upon functionality in the very large design stages. Many attributes characterize value in the software development projects. Some examples are given:

- Customer satisfaction
- Software computation speed
- Software coding length
- Data of delivery
- Number of software defects
- User friendliness
- Easier comprehension

Another important assessment factor is to managing risk as a team. The risk team management defines the organizational structure and operational activities for collectively managing risks throughout the software development projects.

- Shared different visions of projects
- Forward looking and thinking
- Open comments and communications
- Individual perception
- Integrations
- Acceptable strategies
- Systematic and suitable methodological aspects
- Routine and continuous processes.

In order to be successful assessment, the software development based system must meet the technical and business requirements within cost and schedule constraints. Another risk assessment of software development deals with the team risk management. It defines the organizational structure and operational activities for collectively managing the development organization.

Risk assessment describing risk evaluation criteria, their measures within the software development organization and steps to be taken to mitigate the risks by the risk management software system. We will discuss how SDP (Software Development Projects) can be used as a part of continuous risk management model to provide early identification will then be used to analyze the risk information such that the planning can be carried out in an informed state. Another step for tacking information is acted upon to control the risk. The final step is to communicate and document risk and mitigation for future access and awareness. The sophisticated risk assessment techniques to access the requirements, completeness, perform static code analysis, aggregate software matrices or even formally prove the corrections and creative

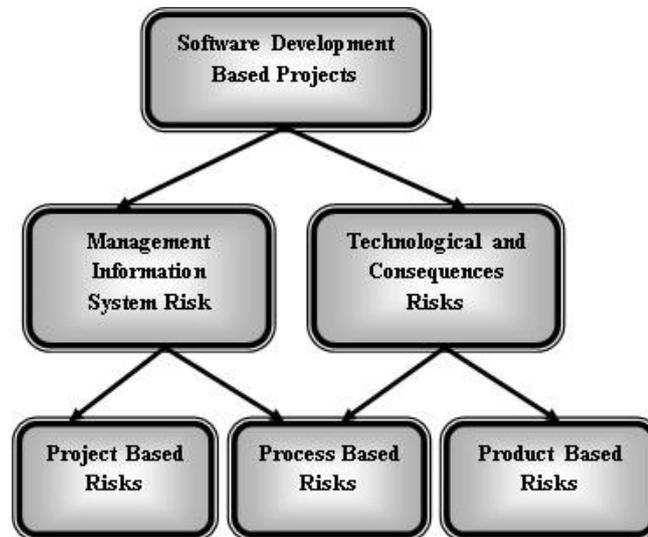


Figure 2. Risk Assessment Hierarchy in SDP

7 Conclusion

The risk analysis and assessment techniques are widely used in software development industries when the projects have progressed to the implementation stages and enough data is available to perform these analyses. Furthermore it is closely focused on a structured risk management process is vital to success, but it also important to align the structure to the size, and type of project in software development organizations. This type of risk process should be scalable. For example, “a small project” probably requires a light touch and meeting every couple of weeks. On the other hand, “a large project” probably requires a heavy touch and meeting at a very large developing company with many developers and stakeholders could require a highly structured, detailed approach to managing risk.

The resulting SDIS provides a snapshot of the risk potential can be identified and planned tasks before undertaking the system development. The use of quantitative and qualitative test is a best practice questions associates a full range of developers with the project tasks. The risk management is an ongoing process that is implemented as a part of the initial project planning activities and utilized throughout all of the phases of the software development project life cycle.

Risk management requires a fear-free environment where risks can be identified and discussed openly. Risk experts focus two principles for the success to software development projects. The first is to avoid an unfocused, free for all discussions. It must be facilitated and structured. Another useful tool for measuring the internal and external treats and opportunities like weakness and strengths. This is a useful and valuable point to achieving a successful software development design rather than by accident

References

- [1] L. Sarigiannidis and P. Chatzoglou, "Software Development Project Risk Management: A New Conceptual Framework," *Journal of Software Engineering and Applications*, Vol. 4 No. 5, 2012, pp. 293-305.
- [2] James Collofello, "Software Risk Management", Arizona State University, 2012.
- [3] J. H. Iversen, L. Mathiassen and P. A. Nielsen, "Managing Risk in Software Process Improvement: An Action Research Approach," *MIS Quarterly*, Vol. 58, No. 13, 2012, pp. 395-433.
- [4] Don Gotterbarn and Simon Rogerson, "Responsible Risk Analysis for Software Development", *Communications of the Association Systems* vol. 75, pp 730-750, 2012.
- [5] Alex Down, Michael Coleman, Peter Absolon, "Risk Management for Software Projects", London, McGraw-Hill Book Company, 2012.
- [6] Carnegie Mellon SEI, "Risk Management Overview", 2011.
- [7] Geoffrey G. Roy and Terence L. Woodings, "A Framework for Risk Analysis in Software Engineering", IEEE 2011.
- [8] Karl E. Wieggers, "Know Your Enemy: Software Risk Management", *Software Development Magazine*, 2011.
- [9] [Gilb-88] Tom Gilb, "Principles of Software Engineering Management", Wokingham, England: Addison-Wesley, 2011.
- [10] [Boehm-89] Barry W. Boehm, Tutorial: "Software Risk Management", Les Alamitos, CA, IEEE Computer Society, 2011.
- [11] B. Boehm, "Software Risk Management", IEEE CS Press, 2010.
- [12] Dale Karolak, "Software Engineering Risk Management: Finding Your Path Through the Jungle", IEEE Computer Society.
- [13] Elaine Hall, "Managing Risk", Addison Wesley Publications.
- [14] Roger L. Van Scoy, "Software Development Risk: Opportunity", *Software Engineering Institute*, CMU/SEI-92-TR-30.

Other Web References

- [15] Software Risk Management Magazine (<http://www.sei.cmu.edu>)
- [16] Software Risk Management Resources (<http://www.rdware.com>)