ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

Software Project Risk Analysis using Intelligent Project metrics

Akshay Mahajan[1], Nitish Garg[2], Prof. P. Swarnalatha[3]
[1] Computer Science and Engineering (SCOPE), VIT University Vellore, India
[2]Computer Science and Engineering (SCOPE), VIT University Vellore, India
[3]Computer Science and Engineering (SCOPE), VIT University Vellore, India

Keywords: SPRM, RE, Risk Management, Data Analysis, Risk Point.

ABSTRACT

Software project administration most likely is a domain that has seen the most noteworthy rate of undertaking failures on the planet. This is not the situation with other projects of different domains because of better administration of their inalienable qualities and shortcomings. Software projects have innate instabilities and risks involved at every stage. Social Software projects endure significantly more prerequisite changes and require more thoughtfulness regarding risk administration. Risk administration is important to expand the possibility of accomplishment of any future task by investigating its instabilities. It will meet every one of the solutions for project advancement success by keeping in perspective all the future issues that may happen amid the undertaking procedure. It incorporates the recognizable proof of risks and their evaluation in the task course and tries to make changes to make the project more valuable by cutting down on the losses incurred due to the potential risks. Risk administration objectives are to capitalize on project undertaking risks that are distinguished before beginning of the task and amid the execution. This paper also aims to provide an insight to this process alongside an intelligent framework for systematic and more efficient risk elimination and handling in the software industry.

INTRODUCTION

A project can be characterized as an impermanent attempt to make an exceptional product or administration. For instance, software project activities are controlled by individuals, more often than not having constrained assets, being arranged, executed, checked and archived. Software projects, given their differing and conceptual nature, offer interesting difficulties and risks at almost all possible stages involved. Software projects are important for playing out an association's business procedure since tasks are a method by which the workflow of the organization is actualized. In this light activities are risky and managers need to take proper initiatives to keep them from this perilous status.

The software industry in recent times has moved far from the earlier and implemented methods which consisted only of single teams and structures with common administrative methods, world view to dispersed, working together groups with adaptable administration connections. In multi-authoritative development, taking an interest groups work for various associations. Multi-authoritative development can be either:

Contract based: With one focal power and different groups chipping away at particular segments with cautiously assigned determined conduct.

Cooperation based: With groups chipping away at sub-frameworks or low coupled parts with iteratively determined interfaces and conduct, frequently without a reasonable, generally acknowledged focal power for determining contrasts and clashes.

Risks are basically situations that could antagonistically influence the advancement of the activities or associated environment. Risk or risk environment variable can harm basic components, for example, spending plan, time or assets. Ordinarily it influences basic elements, for example, financial plan, time and expenses.

Risk Management comprises fundamentally, of distinguishing, breaking down, arranging and controlling events that are potential or certain threats to the success of the project in immediate or far future, intending to maintain a strategic distance from or lessen the harm of these events if there should arise an occurrence. Its administration, be that as it may, does not ensure the achievement of tasks, but rather increment the likelihood of more compelling accomplishments, regarding due dates, inside the arranged spending plan and meeting project objectives

IJESMR
International Journal OF Engineering Sciences & Management Research

According to the reports of the CHAOS Summary of 2009, only a meagre 32% of the total software projects made globally are a success, rest all are failures, owing much of it to the risks which are predominantly involved. Another research by Microsoft Inc. confirms the effectiveness of risk management in software projects which boosts 50% more deadlines being met and other cost benefits too.

This paper hence, aims to build analytical insight into the domain of risk management specifically in software projects alongside a suggested intelligent framework to support and ease this process.

RISK MANAGEMENT

Boehm and Charrette introduced risk management in SPRM for the first time. SPRM is a set of processes which if implemented improves the project's rate of success in most situations. Schedule, scope of the project, budget etc. are positively influenced by risk management processes.
There various factors which contribute to the risks involved in a project, some of which can be categorized into the following:

Common Risk Factors

- ***Project estimation and planning***- requirements change with time, prototyping, increasing complexities, etc.
- ***Cross cultural and gender***- outsourced projects or from distributed environments involves people from all over the globe, leading to disrupt in communication and clarity at times.
- ***External Factors***- could be natural disasters, or could be other artificial ones.
- ***Effectiveness of task Communication***- lack of clarity leads to major project failures at times, especially in distributive environments.
- ***Role of the user***- unclear functionalities and task knowledge.
- ***Project Manager Characteristics***- Lack of top management commitment, etc.
- ***Tools andtechnology***- New and unproven technology.
- ***Organizational Climate and Support***- Project management must plan for efficient execution of the project, and come to agreeable terms with the teams associated with developmental work and the expectations of the customers
- ***Formalization of project charter***- failure in this leads to losing on deadlines and other resources.
- ***Effectiveness of Project Monitoring***- if done keeps deadline and resources intact.
- ***Requirement stability and accuracy***- unclear and conflicting system requirements.

SPRM stages

The whole process of SPRM comprises of various stages which can be broadly classified into the following:

1. ***Risk Management Planning***- This involves planning the project in terms of resources, capital and accordingly mark every possible documentation standards to be followed.
2. ***Risk Identification***- This is a multilevel process comprising divided into the following:

   - **Application Level:** Emphasizes on risks that could be internal or external and majorly are technical or execution failure of the software project. Examples include: Viruses, competitor's strategies, natural disasters, authorized or unauthorized access and relevant system abuse comprises of internal risks.
   - **Organizational Level:** After heavy emphasis and investment on emerging technology, when organizations fail to maintain the same in the long run, it gives rise to:
     - Legal risks includes violation of customer rights through IT
     - Sustainability risks includes loss of competitive advantage
     - Increased power for bargaining amongst users
     - Data security risks which includes denial of service during important situations.
   - **Inter-organizational Level:** IT risks at inter organizational level is mainly in the network environment, which needs to be identified, risks include hacking, intrusion by unauthorized personnel and natural disasters again.

3. ***Risk Analysis-*** Analysis of all collected data from warehouse and current project scenario for risks with estimation based on probabilistic and data analysis models.
4. ***Risk Response Plan-*** Some risks may incur minimal losses and may not affect the bigger picture, hence developing plans to minimize the relevant losses and ignore others, to incorporate efficient utilization of resources.
5. ***Risk Monitoring-*** Checks and rechecks all identified risks discovered yet.
6. ***Risk Control-*** Implementing and executing the risk responses according to plans and works towards better success of project.

Risk Communicate- Strong communication structure between stakeholders and software teams to avoid communication based risks

PROPOSED FRAMEWORK

Our proposed framework is based on a layer by layer approach. The layers form in a direct or indirect way an integral part of the complete SPRM approach. Comprising mainly of three layers it includes:

1. **Data Collection and data Warehouse formation:** Involves the same methodology involved in black boxes used in aircrafts, for collecting all relevant data regarding risks at every stage of prior projects implemented and simultaneously current projects too, hence forming efficient warehouse to refer to during risk identification periods. This forms a long term and expensive process but ensures high quality in the long run.

2. **Risk Analysis under intelligent project metrics:**

The database formed in the above step is a prequel to this step. The quality of data and hence risks identified is directly proportional to further better analysis and estimation of the impact of each risks involved and hence better actions are taken.
The training data provided in the above step in fed to numerous different mathematical models which could be probabilistic like the probabilistic Bayesian network or data driven Neural networks or further randomized like the random trees.
Based on the steps discussed above, parameterized metrics have been designed to affect the final classification of the analysis as project success or failure.
Intelligent Software Metrics:

Software metrics can be characterized as the constant utilization of estimation based procedures to the product development process and to supply important management data, looking for programming and process change.

There are two categories of software metrics: base and inferred.

Base measurements is characterized regarding a solitary single property and the strategy for evaluating it. It is practically free of different measurements. The estimation technique includes the consistent grouping of operators keeping in mind the end goal to measure an attribute concerning a predetermined scale. There are two sorts of estimation strategies: subjective and objective. The first evaluates a property through human judgment, though the target technique is just in light of numerical guidelines.

Inferred metrics is characterized as a function (maybe simple or complex) of two or more estimations of base metrics.

The metrics discussed earlier are:

- The most conventional metric used which is direct is **Risk Exposure** which is numerically equal to

RE(risk) = Impact of risk * Probability(risk)

- **Risk point metric** which based on quantifiable calculation methods and is equal to:

$$\mathbf{RP} = \mathbf{URPW} * \mathbf{PCF}$$

Where RP is Risk Point, URPW is Unadjusted Risk Point Weight and PCF is Project Characteristics Factor.

Further**, PCF** is used to adjust the metrics value against a weight which is calculated based on a series of answers to a questionnaire consisting of 8 question which sums up to:

$$\mathbf{PCF} = (\mathbf{0.015} * \mathbf{CF}) + \mathbf{1.05}$$

Again **CF** here is Characteristic Factor which has a range of 0 to 4 for which question in the questionnaire which is further multiplied by the individual weight defined for each question.

$$\boldsymbol{CF} = \sum_{\boldsymbol{n}=\mathbf{1}}^{\mathbf{8}} (\boldsymbol{Question_n} * \boldsymbol{Weight_n})$$

Unadjusted Risk Point Weight (URPW): which is calculated based on Risk Exposure. The formula for k risks is:

$$\boldsymbol{URPW} = \sum_{\boldsymbol{n}=\mathbf{1}}^{\boldsymbol{k}} \boldsymbol{Weight}(\boldsymbol{Risk_n}|\boldsymbol{k} = \boldsymbol{number\ of\ identified\ risks})$$

This provides a scale of five which further estimates a project success or failure.

We propose a threshold value of 3 and set a binary analysis function which could be fed into the analysis modules as follows:

$$\boldsymbol{Predicted\ Project\ Outcome} = \begin{cases} \mathbf{1}, & Weight < 4 \\ \mathbf{0}, & Weight > 3 \end{cases}$$

- **Other Metrics** which includes those such as Exponential Risk Point and Probabilistic Risk Point.

3. Intelligent Risk Planning:

Initially, the parameters of risk arranging, including a rundown of all possible risk control activities, the relationship between many of the risks to other many control activities amongst activities and risk elements, and the execution expenses of activities are surveyed.

Second, taking the above parameters as info, the risk planning module yields the cost-insignificant activity set. Third, the project departments assess the produced activity set and change it as indicated by genuine circumstances.

Hence, the risk determination procedure is started to execute the delivered plan.

FORMULAE

$$\text{Accuracy} = \frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}} \qquad (1)$$

$$\text{Precision} = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \qquad (2)$$

$$\text{Recall} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \qquad (3)$$

IJESMR
International Journal OF Engineering Sciences & Management Research

TABLES

Table 1: Urpw Classification

Classification	RE(Risk)	Weight(Risk)
Very low	[0.0,0.2)	1
Low	[0.2,0.4)	2
Medium	[0.4,0.6)	3
High	[0.6,0.8)	4
Very High	[0.8,1.0]	5

CONCLUSION

Software Project Risk Management is a very complex and tedious task which demands immense commitment and resources in a few cases, at different stages but have potential of saving billions of dollars and time at many stages of product development. This paper provided a layered approach to the complex procedure of SPRM and most importantly provides an intelligent framework working on each layer proposed in order to automate a few processes and more importantly provide a definite and structured protocol to proceed with SPRM. Right from database collection to its utilization in a top to bottom approach with intelligent models working on it to bring down the risks to a project especially in the software industry which has been prone to major failures due to lack of a systematic protocol.

REFERENCES

1. Beranek, P. M.; Broder, J.; Romano, N.; Reinig, B.; (2005). Management of virtual project teams: Guidelines for team leaders. Communications of the Association for Information Systems, s 247−259.
2. B. Reyck, Y. Grushka-Cockayne, M. Lockett, S. Calderini, M. Moura, A. Sloper, The impact of project portfolio management on information technology projects, International Journal of Project Management 23 (7) (2005) 524–537.
3. Miler J., Górski J., Implementing risk management in software projects, Proc. of 3rd National Software Engineering Conference, Poland, 2001
4. Higuera R. P., Haimes Y. Y., Software Risk Management, SEI report CMU/SEI--96-TR-012, Carnegie Mellon University, Pittsburgh PA, June 1996.
5. J. Cumming, IT portfolio management: balancing risks and rewards of projects yields significant returns, Network World 19 (13), 2002, pp. 48.
6. Laura Girdžiūtė. 2012. Risks in Agriculture and Opportunities of their Integrated Evaluation. Procedia - Social and Behavioral Sciences 62, 783-790.
7. Selby RW, Software Engineering: Barry W. Boehm's Lifetime Contributions to Software Development, Management, and Research, John Wiley & Sons, 2007.
8. F. Reyes,N. Cerpa, A. Candia-Véjar,M. Bardeen, The optimization of success probability for software projects using genetic algorithms, Journal of Systems and Software 84 (5)(2011) 775–785.
9. W. Han, S. Huang, An empirical analysis of risk components and performance on software projects, Journal of Systems and Software 80 (1) (2007) 42–50.
10. John Organ, Larry Stapleton. 2013. Information Systems Risk Paradigms: Towards a New Theory on Systems Risk. *IFAC Proceedings Volumes* **46**:8, 116-121
11. E. Wang, P. Ju, J. Jiang, G. Klein, The effects of change control and management review on software flexibility and project performance, Information Management 45 (7) (2008) 438–443.
12. K.R. MacCrimmon, D.A. Wehrung, The risk in-basket,Journal of Business 57 (3), 1984, pp. 367–38