Software Project Risk Management

Shailesh S. Limkar\textsuperscript{1} and Vaishali P. Bhosale\textsuperscript{2}
\textsuperscript{1,2}Bharati Vidyapeeth Deemed University, Institute of Management, Kolhapur
\textsuperscript{1}shaileshlimkar@gmail.com and \textsuperscript{2}vaishali.p.bhosale@gmail.com

ABSTRACT

The aim of this research paper is to find some tools and techniques for software project risk management recommended by different journals and articles. We have gone through different approaches in context of risk management. The literature review explains top ten software risk items and risk management techniques for the same enlisted by Dr. Barry Boehm. Conceptualization of risk management process is elaborated with two major phases. Then we have taken risk management paradigm introduced by Software Engineering Institute as our standard to analyze different techniques and tools available for risk management. Finally the discussion ends with the concluding remark that collective approach is needed to deal with software project risk.

KEYWORDS

SEI, SRE, Softrisk, Riskit.

1. INTRODUCTION

Webster’s dictionary defines “risk” as “the possibility of loss or injury.” This definition can be translated into the fundamental concept of risk management: risk exposure, sometimes also called “risk impact” or “risk factor.” Risk exposure is defined by the relationship\cite{5}

\[ RE = P(UO) \times L(UO) \]

where RE is the risk exposure, P(UO) is the probability of an Unsatisfactory Outcome and L(UO) is the loss to the parties affected if the outcome is unsatisfactory. To relate this definition to software projects, we need a dehtion of “unsatisfactory outcome.” [Barry Boehm] e.g. budget overruns and schedule slips, products with the wrong functionality, poor-quality software is unsatisfactory outcome. This varies with the perspective of the participants in the project viz. developers, customers, maintainers etc.

2. LITERATURE OVERVIEW

We have gone through literature available on software project risk management and found the table below as a summarized presentation of the same from the article “Software Risk Management : Principles and Practices” Dr. Barry Boehm\cite{1}.

3. CONCEPTUALIZING RISK MANAGEMENT PROCESS

All projects have some level of risk associated with them. Even if the product under development is imply another version of an existing system or product, risks may appear in areas such as:

- changes in development personnel (and resulting experience levels with the product)
- changing market conditions and customer expectations
- changing business conditions for the development organization

The more you understand the risks, the better equipped you are to manage them.

Figure shows, a general practice of risk management which involves two primary steps each with three subsidiary steps.\cite{4}

<table>
<thead>
<tr>
<th>Risk item</th>
<th>Risk-management technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personnel shortfalls</td>
<td>Staffing with top talent, team building, key personnel agreements.</td>
</tr>
<tr>
<td>Unrealistic schedules and budgets</td>
<td>Detailed multisource cost and schedule estimation, software reuse.</td>
</tr>
<tr>
<td>Developing the wrong functions and properties</td>
<td>User surveys and user participation, prototyping, early users’ manuals.</td>
</tr>
<tr>
<td>Developing the wrong user interface</td>
<td>Prototyping, scenarios, user participation.</td>
</tr>
<tr>
<td>Gold-plating</td>
<td>Requirements scrubbing, cost-benefit analysis.</td>
</tr>
<tr>
<td>Continuing stream of requirements changes</td>
<td>Information hiding, incremental development</td>
</tr>
<tr>
<td>Shortfalls in externally furnished components</td>
<td>Inspections, reference checking, compatibility analysis.</td>
</tr>
<tr>
<td>Shortfalls in externally performed tasks</td>
<td>Competitive design or prototyping, team-building.</td>
</tr>
<tr>
<td>Real-time performance shortfalls</td>
<td>Simulation, benchmarking, modeling.</td>
</tr>
<tr>
<td>Straining computer-science capabilities</td>
<td>Technical analysis, cost-benefit analysis.</td>
</tr>
</tbody>
</table>

4. CONCEPTUALIZING RISK RESOLUTION TECHNIQUES

4.1. SEI RISK MANAGEMENT PARADIGM

Risk management is a process that is systematic and continuous and it can best be described by the SEI (Software Engineering Institute) risk management paradigm. SEI thinks all risk
management technique will follow their prescribed paradigm for risk management of software project. Figure shows the SEI paradigm.

![Fig 1: General Practice of Risk Management](image)

**Fig 1: General Practice of Risk Management**

We have summerized the Elements and their Purpose in SEI model.

**Identify:** makes all known project risks explicit before they become problems

**Analyze:** transforms risk data into decision-making information

**Plan:** translates risk information into decisions and mitigating actions (both present and future) and implements those actions

**Track:** monitors risk indicators and mitigation actions

**Control:** corrects for deviations from the risk mitigation plans

**Communicate:** enables the sharing of all information throughout the project and is the cornerstone of effective risk management [3]

5. DIFFERENT TOOLS AND TECHNIQUES FOR SOFTWARE RISK MANAGEMENT:

5.1 SOFTWARE RISK EVALUATION (SRE)

The SRE addresses the identification, analysis, planning, and communication elements of the SEI Risk paradigm. An SRE is used to identify and categorize specific project risk statements emanating from product, process, and constraint sources.

![Fig. 3: SRE covering three elements of SEI’s paradigm](image)

5.2 SOFT RISK 2000

The first step in Softrisk model is Risk Identification. Softrisk model identifies not only general kind of risk that can be occurred for any kind of project but also specific type of risk that only can be happened for some specific type of project. After risk identification in second stage of Softrisk model each identified risk is addressed in term of its probability and magnitude. Probability and magnitudes of the risks are estimated by a special checklist. Negligible, low, medium, high, very high and extra high are the categories under which risks are estimated.

In risk documentation stage Softrisk documents all generic and specific risks data. This document is used for tracking projects situation, statistical operations and future risk predictions. Afterwards, risk assessment is done based on risk’s probability and magnitudes.

5.3 RISKIT ANALYSIS GRAPH

The Riskit method provides precise and unambiguous definitions for risks. It is aimed at modeling and documenting risks qualitatively. It uses the concept of utility loss to rank the loss associated with the risk. [2]

6. CONCLUSION

The concluding remarks on the software project risk management are:

- No single method to deal with the risk management in software projects and to maintain its quality.
- Collective approach is needed to deal with software risks.
- Quality software ensures all the risk management steps considered as an umbrella activity during the software development process.
REFERENCES
[3]. D. Chatterjee, VC Ramesh, Real Options For Management In Information Technology Projects, IEEE, 1999.

Fig. 4: The Soft Risk 2000 Model

Fig 5: