



## INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

### Risk Analysis and Mitigation Plan in Software Development

Dr. Sheel Ghule

G Persistent Systems Limited, Nagpur, India

[sheelg@rediffmail.com](mailto:sheelg@rediffmail.com)

---

#### Abstract

Software development, often encounter many unanticipated problems, resulting in projects falling behind on deadlines, releases, exceeding budgets and result in sub-standard products due to its complex nature. Although these problems cannot be totally eliminated, they can however be controlled by applying Risk Management Plan. This can help to deal with problems before they occur. Organisations who implement risk management plan have control over the overall management of the project. This paper provides some necessary information regarding the potential risks that can occur in the software development, their impact on the projects, and some useful information of handling these risks. Moreover it can help software risk management process.

**Keywords:** Software risks, risk management, risk management software, risk management tools, web-based applications.

---

#### Introduction

There are many risks involved in creating high quality software that need to be carefully managed, require risk analysis and mitigation plan. Risk management has traditionally been an integral part of software development. In current era of software projects, security and risk management are not just something one might do if there are time and resources. This means risk management must be introduced at the beginning of the project, and risks must be evaluated and assessed during the whole development cycle. Despite new technology, innovative methods and tools, different management methods - development process still full of risks from the beginning to the end.

Therefore, to make sure a project successful we require managing specific IT risks related to our software projects: identify risks and store in a shared data storage, assess risks, using specialized tools and techniques, choose appropriate mitigation action and track that mitigated risks are lower when they were.

A software project encounters two types of risks, development process risks and product- related risks. Some of the development process risks are developer errors, natural disasters, disgruntled employees, and poor management objectives. Some project related risks are incomplete requirements, unclear project deliverables and objectives, and complexity of the product.

Risk management should begin at the earliest stages of program planning and continue throughout the total life-cycle of the program. Additionally, risk

management is most effective if it is supported with automated tool that ensures integration with the program's systems engineering and program management processes.

#### Risk Strategies

- Reactive strategies - very common, also known as firefighting, project team sets resources aside to deal with problems and does nothing until a risk becomes a problem
- Proactive strategies - risk management begins long before technical work starts, risks are identified and prioritized by importance, then team builds a plan to avoid risks if they can or minimize them if the risks turn into problems

#### Software Risks

- Project risks - threaten the project plan
- Technical risks - threaten product quality and the timeliness of the schedule
- Business risks - threaten the viability of the software to be built (market risks, strategic risks, management risks, budget risks)
- Known risks - predictable from careful evaluation of current project plan and those extrapolated from past project experience
- Geographical Distance - Geographical distance is a spatial distance between two entities (organizations, persons, cities, etc.) measured by the cost and times.

- Organizational Distance - needs a tight synchronization of the development processes of participating teams. A great problem becomes when common processes are not understood well enough by SD team members.
- Knowledge Distance - Lack of knowledge about the solution can cause bad specification of system requirements and hence bad design of the solution. Incomplete, insufficient or unclear requirements specification can be considered by a distant partner may create some serious problems and as while locally it is considered an easy task as compared to SD.
- Cultural literacy at the country and firm level - At the country level, the difference in the nature of culture may impact upon offshore outsourcing negatively and similarly at the firm level, there are also different organizational culture and practices. Issues such as business moral and norms are dominant of these types of risks.
- Unrealistic Timeline - An improbable timeline for any of the outsource/offshore process can lead to disappointing results and ultimately fail the project.
- Unknown risks - some problems simply occur without warning

**Risk Identification**

- Product-specific risks - the project plan and software statement of scope are examined to identify any special characteristics of the product that may threaten the project plan
- Generic risks - are potential threats to every software product (product size, business impact, customer characteristics, process definition, development environment, technology to be built, staff size and experience)

**Risk Prioritization**

Deal with the most important risks first. There is often a good deal of uncertainty in estimating the probability or loss associated with a risk. The amount of uncertainty is itself a major source of risk, which needs to be reduced as early as possible.

**Risk-Management Plans**

The focus of risk-management planning is to develop a plan to handle each of the high -priority risks identified during the previous activities. The plan should be documented and oriented around answering

the standard questions of why, what, when, who, where, and how.

**Risk Mitigation**

The activity of mitigating and avoiding software risks is based on information gained from the previous activities of identifying, planning, and assessing risks. Usually the predefined mitigation action categories, such as avoidance, minimization, transference, limitation, etc. are available. The types of responses can vary depending on the chosen methodology, but the main four types of responses are:

1. Mitigate the Risk – incorporate specific plans into the project scope to deal with the occurrence of, or to minimize the likelihood of, the risk occurring;
2. Avoid the Risk – remove scope that includes risk from the project;
3. Share the Risk – transfer ownership of scope to another party so they now have risk;
4. Accept the Risk – do nothing, run the chance of the risk occurring, deal with it if it does.

Risk mitigation handling options include:

1. Assume/Accept: Acknowledge the existence of a particular risk, and make a deliberate decision to accept it without engaging in special efforts to control it. Approval of project or program leaders is required.
2. Avoid: Adjust program requirements or constraints to eliminate or reduce the risk. This adjustment could be accommodated by a change in funding, schedule, or technical requirements.
3. Control: Implement actions to minimize the impact or likelihood of the risk.
4. Watch/Monitor: Monitor the environment for changes that affect the nature and/or the impact of the risk.

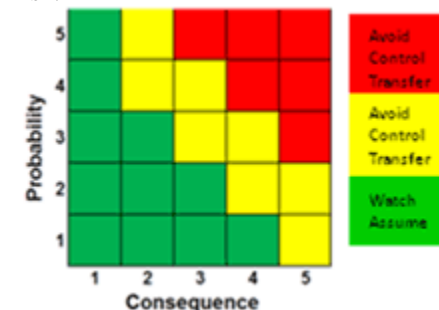


Fig. Risk mitigation handling options

Risk reporting is based on information obtained from the previous topics and compares risk status against previously identified risks. Risk reporting provides capabilities to visualize risk information in graphs and

charts that can be further exported to Excel, Word, and PowerPoint in native chart format for easy distribution to others.

Risk monitoring and documentation of lessons learned finalize the risk management processes. Usage of the risk database from past projects to plan current projects can help the managers to avoid most already known problems and lets them learn not from their own mistakes, but employ best practice experience and project expertise.

The final phase, improve and expand, starts when basic risk management practices have been implemented in the project. Improvement is needed to ensure that risk management is more and more integrated into normal project risk management and to make the processes, methods and tools more effective. Lessons learned should be documented. Also continuous training and facilitation is required. Risk management should also be expanded to other projects within the organization.

Effective risk management requires involvement of the entire program team and may also require help from outside experts knowledgeable in critical risk areas such as threat, technology, design, manufacturing, logistics, schedule, cost, etc. Overall the extended project team carries out risk management and mitigation activities. Risk management is the responsibility of the Project Manager. However, all project stakeholders should participate in the risk identification and analysis process.

### Recommendations

Unexpected events that surprise management are not tolerated in business or projects today. We suggest that a company reinforce this by applying the technique throughout the organization, from the corporate office to the engineering teams and mill maintenance shop. Using web-based systems and proprietary software allows interconnectivity of systems and users, within and outside the company to minimize the ramp-up time and to formalize the plans. With successful mitigating actions, confidence will build. With management attention and support, the actions have a high probability of success.

Without a risk management plan, the risks are there but perhaps hidden, or not well considered. You are taking the risk, so why not try to mitigate them. The return on the time, effort and costs of risk management can be substantial.

### Conclusion

Software development is the key area in software development whose objective is to provide the high quality solutions to the user in reasonable

budget and specified time deadline but as like another engineering discipline; SD is passing through an era which is full of complexities in terms of risks management. Risk factors can be controlled right after the early phase of the software development project but it must need a lot of managerial expertise. Although software project management tools and technologies exist but experience and knowledge counts much more. So it is very much important to understand the impact of different potential risk by a project manager so that the desired results are achieved. Risk management in SD is not yet fully controlled and thus there needs much more research work.

### References

1. Boehm, B., "A Spiral Model for Software Development and Enhancement," Computer, vol. 21, no. 5, May 1988, pp. 61-72.
2. Fuller, A., Croll, P., Di, L., 2002. A new approach to teaching software risk management with case studies. In: Proceedings of the 15th conference on Software Engineering Education and Training, pp.215-222.
3. G. Hunter and J.E. Beck, "A Cross-Cultural Comparison of 'Excellent' Systems Analysts," Information Systems J., Vol. 6, No. 4, 1996, pp. 261-281.
4. Kliem, R., "Managing the Risks of Offshore it Development Projects EDPACS," 32(4): p. 12- 20. 2004.
5. Gemmer, A., 1997. Risk management: moving beyond process. Computer 30 (5), 33-43.
6. Klein, S.A., 1998. Putting methodology in perspective from a project risk viewpoint. In: IEEE Power Engineering Society 1999 Winter Meeting, vol. 1,, pp. 362-365.
7. Kwak, Y.H., Ibbs, C.W., 2000. Calculating project management's return on investment. Project Management Journal 31 (2), 38-47.
8. <http://ceur-ws.org/Vol-963/paper3.pdf>
9. Kliem, R., "Managing the Risks of Offshore it Development Projects EDPACS", 2004. 32(4): p. 12 - 20.
10. Christiansen, H.M., "Meeting the Challenge of Communication in Offshore Software Development, in Software Engineering Approaches for Offshore and Outsourced Development," S. First International Conference, Zurich, Switzerland, February 5-6, 2007. Revised Papers, Editor. Springer: Berlin / Heidelberg. p. 19-26. 2007