Software Risk Management Practice: Evidence From Thai Software Firms

Tharwon Arnuphaptrairong

***Abstract*—Software risk management has been around at least since it was introduced in mainstream of software management process, in 1989 [1]-[3] but little has been reported about its industrial practice [4]-[6]. This paper reports the current software risk management practice in Thai software industry. A questionnaire survey was designed to capture the information of the software project risk management practice. The questionnaire was sent to 141 companies and received a response rate 28 percent. The findings indicate that Thai software firms do not neglect software risk management. The results also show the discrepancy between standard risk model and industrial practice. The industrial has not implemented all the risk management activities prescribed in the standard risk model. Thai software firms seem to give more attention to risk identification, risk analysis, risk management planning, and risk monitoring and control but left out other two phrases--risk sign-off, and risk post-mortem analysis. This is similar to the findings of Kajko-Mattsson and Nyfjord [5].**

***Index Terms*—software risk, software risk management, software risk practice, software risk management practice, software risk evidence.**

I. Introduction

Software risk management is a complex activity and also a major contributor to the software project success. Since it was introduced in mainstream of software management process, in 1989 [1]-[3], both the academic and the software industry are well aware of its significance. Research about risk dimensions, risk factors, top ten risk management and a number of established standard models, frameworks and theories have been suggested. However, very a little empirical evidence about the status of its practice has been reported [4]-[6].

The objective of this research is to study the state of the practice of software risk management in Thai software industry.

Understanding the state of the practice will give incitements which hopefully will help closing the gap between theories and practices, and lead to the software project success.

This paper is organized as follows. Section II gives the review of software risks fundamental suggested in the literature. Section III discusses the research methodology. Section IV presents the findings of the survey and the conclusion and discussion are given in section V.

T. Arnuphaptrairong is with the Department of Statistics, Chulalongkorn Business School, Chulalongkorn University, Bangkok 10250 Thailand (e-mail: Tharwon@acc.chula.ac.th).

II. Overview of Related Literature

This section reviews the literature related to the proposed research objectives i.e., software risks, software risk management, software risk management process model, roles and responsibility, and empirical study in software risk management practice.

A. Software Risks

The term risk is generally used in many different domains. In the "software" context, several definitions can be found. For example, Leihman and VaanBuren [7] defines risk as "A possible future event that, if it occurs, will lead to an undesirable outcomes."

PM-BOK (Project Management Body of Knowledge) defines risk as: "an uncertain event or condition that, if it occurs, has a positive or negative effect on a project's objectives [8]."

Whereas PRINCE2, the UK government sponsored project management standard defines risk as: "the chance of exposed to the adverse consequences of future events." And in all, risks are related to 2 key elements: future events, and may cause effects [9].

Software risk management is a complex activity. It has to deal with uncertain events of the software project and their causes. Researchers have tried to overcome this obstacle by suggesting the fundamental steps or phrases to handle them. This is known as "software risk management process model."

B. Software Risk management

Software risk management can be defined as "the way to handle risks in a software project". Its objective is to reduce uncertainties and impacts associated with certain tasks in the project. The fundamental software risk management consists of 4 major processes: 1) risk identification, 2) risk analysis, 3) risk planning, and 4) risk monitoring and control [5], [8], [10].

1) Risk Identification

Risk identification deals with the process of determining which software risk factors that might affect the software project. The software risk factors can be elicited using various techniques. These include:

a) interviewing/brainstorming with project team members, experts, customers, and other stakeholders, or

b) Delphi method – a technique to reach the consensus of participants on software risk factors anonymously.

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

In the elicitation process, in order to determine the related risk factors, the process may use various tools, including risk checklists [11-13], the top ten software risks check lists [1], or risks dimensions/categories [14]. One may use the risk checklists available from the literature or from organization own repository of risk lists. Many risk checklists can be found in the literature.

In their recent experimental study, Han and Huang [11] gave a comprehensive review on software risk lists. Risks were reviewed from 12 studies. Table I shows the details of the studies and number of risks reviewed from [11].

TABLE I
SUMMARY OF SOFTWARE RISK RESEARCH [11]

AUTHOR(YEAR)	DIMENSION OF RISKS	NUMBER OF SOFTWARE RISKS
McFarlan (1981)	3	54
Boehm (1991)	0	10
Barki et al. (1993)	5	55
Summer (2000)	6	19
Longstaff et al.(2000)	7	32
Cule et al. (2000)	4	55
Kliem (2001)	4	38
Schmidt et al. (2001)	14	33
Houston et al. (2001)	0	29
Murti (2002)	0	12
Addision (2003)	10	28
Carney et al. (2003)	4	21

Finally, the software risk factors that all the parties involved agreed upon should be produced and recorded in a "risk register".

2) Risk Analysis

The next process is to analyze and prioritized the identified software risk factors. The process is to assess the impact and the probability that the identified risk will lead to the undesirable outcomes. The risk exposure is then obtained by multiplying the risk impact with its probability. The analysis may use different techniques such as risk sensitivity analysis, decision tree and scenario analysis [8]. The identified risks are then ranked according to the risk exposure calculated to create the prioritized risk list and confirmed by the stakeholders [5], [8], [10].

3) Risk planning

The following step is the process of developing a risk response or risk management plan. The risk response plan consists of strategies, options or alternative actions and actions in response to the prioritized risks. Generally the risk response strategies aim at reducing or eliminating the probability of the prioritized risks, minimize the impact of the risks if it is realized. There are four common strategies in response to the software risks --acceptance, avoidance, mitigation, and transference.

Risk acceptance is to accept or do nothing to deal with a particular risk.

Risk avoidance is to take action to prevent risk events from occurring so that if it occurs there will be little impact.

Risk mitigation is to take early action to reduce the risk probability or to protect from its impacts.

Risk transference is to shift the responsibility of the consequences of a risk to a third party.

Besides the risk response plan, control and monitoring plan and contingency plan may be included in the risk planning process. The control and monitoring plan describes relevant procedures and measures in order to control and monitor the risks. Contingency plan defines a secondary or alternative course of action to be taken in the event that the primary approach fails to function as it should.

4) Risk monitoring and control

Risk monitoring and control is the process of keeping track of the registered risks according to the control and monitoring plan. The purpose is to make sure that all risk responses have been implemented, observe the risk status and take action as specified in the risk response plan and record the risk status in the risk register.

However, in addition to these 4 steps above, two more process are also suggested --5) risk sign-off and 6) risk post-mortem analysis [5].

5) Risk sign-off

The status of the risk likelihood and impact should also be monitored onto the risk register. For the risk that is mitigated, this process is to update the status and removes it from the risk list and sign it off. Sometimes, this step may be seen as a part of the risk monitoring and control.

6) Risk Post-Mortem Analysis

This process is to evaluate the risk management process and its results when a project has been completed. Review should be conducted to see the effectiveness on how the risks identified, analyzed, planed, managed and monitored. The lessons learned can then be used on other projects to aid their risk management.

C. Software risk management Process Model or Framework

Software risk management process models specify stepwise tasks in order to manage risk of the software project [4]. There are variations in software risk management models which usually centered around the principle and practice of four major processes mentioned before –1) risk identification, 2) risk analysis, 3) risk planning, and 4) risk monitoring and control. Whilst the software risk management process model in [5] comprises of 6 phrases -- risk identification, risk analysis, risk planning, risk monitoring and control, risk sign-off and risk post-mortem analysis. Well known risk management model or framework includes Boehm [1], SEI's software management model [15] and Kontio's Riskit methodology [16, 17].

According to Boehm [1] risk management consists of two steps –risk assessment and risk control. Risk assessment contains risk identification, risk analysis, risk prioritization whereas risk control involves risk management planning,

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

risk resolution procedure, and risk monitoring. Riskit [17] consists of risk management mandate, goal review, risk identification, risk analysis, risk control planning, risk control and risk monitoring. SEI's software management model [15] encompasses identify, analyze, plan, track, control, and communicate. These frameworks also recommend different techniques, for example, identifying risks for software project Boehm [1] recommended risk checklists, decision drivers, assumption analysis, or decomposition. Riskit [17] recommended brainstorming, checklist or benchmarking whereas SEI recommended risk taxonomy questionnaire method [15].

There are many prominent risk management standards, models, or guidelines available in literature and practice. Example models are CMMI (RSKM model), Continuous risk management (CRM), ISO/IEC guide, ISO 9000, ISO 9001:2000, Project Management Body of Knowledge (PMBOK), Prince 2, and IEEE [4, 5].

D. Roles and Responsibility

Project managers are generally responsible for the whole software risk management process. After risks are identified and prioritized they may be assigned to the responsible persons or risk owners [5].

E. Empirical Study in Software Risk Management Practice

Lack of empirical study in software risk management practice was discussed in the literature [4]–[6], [16].

In their review of literature of different techniques for risk management in software engineering, Misra et al. [16] concluded that "there is a lack of understanding of the area amongst the software engineering practitioner" and "many of the approaches discuss in this article are limited by the lack of empirical study"

Kajko-Mattsson and Nyfjord [5] stated that "Despites the fact that risk management has been with us for some time, little has been reported about its industrial status." Bannerman [4] and Odzaly [6] also called for more empirical software risk management practical evidence.

In Kajko-Mattsson and Nyfjord [5], by using a convenience sampling, international master program students were asked to choose to interview an organization that has risk management process in place, in their home country. Data from 37 organizations were collected and analyzed. The results show discrepancies between industrial practices and the standard models prescribed in the literature. Organizations studied did not implement important process as prescribed in the literature. On the other hand, standard model fails to identify some important risk management activities. Only a few have implemented the entire process of software risk management. Organizations mainly implemented risk identification and risk analysis process. Many problems were indicated. The first mentioned problem was with employees' attitude toward risk management. Employees were described as do not take the risk management seriously. Other problems were related to experience of risk managers, tools, resources, formal procedure, process standardization, knowledge management, and documentation. Suggestions regarding risk categorization, roles, risk activities and phrases, risk recording, risk for specific type and organization were introduced.

Bannerman [4] studied risk management practice in government sector in an Australian state. Structured interview with 23 informants from 17 organizations on 17 projects were conducted. The findings were similar to the study of Kajko-Mattsson and Nyfjord [5], as he put "software risk management is under-performed in practice." The findings challenge some conventional concepts of risk management and project management. For example, it was found that software projects do not conform to a uniform structure, as assume in much of the literature and as they mentioned "Risk management research lags the needs of practice, and risk management as practiced lags the prescription of research".

Odzaly [6] showed evidence from reviewing of the literature that risks are not well understood, there are too many risks to manage, risks management is difficult due to complexity and there is a lack of motivation to perform risk management. They used an on-line questionnaire to study the barriers of software risk management. The perception data of 18 project managers from 12 companies was collected. The research showed a good awareness of software risk management but with low tool usage. The main barriers of software risk management are related to perception of its high cost but low value. Risk identification and risk analysis are especially perceived as effort extensive and costly. They suggested the values of cost ratio for software risk management needed to be proved.

III. Research Methodology

A. Survey Design

The survey method was used to obtain the information of the software risk management practice from the Thai software firms. About 200 software companies that joined Software industry club of The Federation of Thai Industries (FTI) were used for the survey frame. In the data collection process, names, addresses and contacts of software firms were obtained from FTI. An officer at The Federation of Thai Industries (FTI) was asked to help in contacting and solicitation in order to increase the response rates. The software firms will first be contacted by e-mail and asked to participate in the research. If the software company agreed to participate, the questionnaire was sent for the software project risk management data needed.

141 companies agreed and 40 questionnaires were returned. This is a response rate of 28 percent.

B. Questionnaire Design

General information about the software firms and the respondents were obtained from the first part of the questionnaire. The second part of the questionnaire was designed to obtain the information regarding the software

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

risk management practice of the software firms. 13 questions included in the questionnaire are shown in Table II.

TABLE II
QUESTIONS

Part 1: General information
1. Organization Name:
Organizational Size (Number of employee)
(Number of developers)
2. Respondent Position
Experience (number of year) in project management
Part 2: Software Risk Management Practice
3. Does your organization follow/ use/ have a software risk management process?
4. Is there any standard Risk Management Model in place?
5. Does your organization carries out (please rate how widespread in your in your projects?
6. If you perform risk identification in you organization, in the Risk Identification process, which of the following techniques your organization utilizes (can check more than one item)?
7. If you perform risk analysis in you organization, in the Risk analysis process, which of the following techniques your organization utilizes?
8. Who is responsible for software project risk management?
9. Does your organization assigned software to risk owners?
10. Does your organization follow/ have any risk management standard or model?
11. Does your organization use any tool to support the following step?
12. Does your organization use risk register?
13. Does your organization record risk management at the following step?

C. The Profile of the Respondents

As shown in Table III, of the 40 questionnaires returned, 31 companies (77.5%) answered that their organizations have a software risk management process. Therefore the other 9 organizations that answered that they do not have software management process will be excluded from further analysis.

TABLE III
THE NUMBER OF FIRMS WITH RISK MANAGEMNT PRACTICE

Risk management Practice	Frequency	Percentage
Risk management process is embedded in the project management process	29	72.5
Risk management process is maintained as a separate process	2	5.0
Do not have risk management process	9	22.5
Total	40	100.0

Profile of the 31 companies and respondents are given in Table IV. Most of the companies are of small to medium size. 48.39 percent of the companies have the number of employees of 1 to 16 and 29.03 percent of the companies have the number of employees of 17 to 32. The average number of employee is 70.26.

48.39 percent of the companies the companies have the number of developers: 1-6, and 25.81% percent of the companies the companies have the number of developers of 7-12. The average number of developer is 9.7.

Most of the respondents are project managers (45.16%). 54.84 percent have the experience in project management from 1 to 5 years and 29.03 percent have the experience in project management from 6 to 10 years. The average years of work experience is 5.54 years.

TABLE IV
THE COMPANIES' and RESPONDENTS' PROFILE

	Frequency	Percentage
Number of Employees		
1 - 16	15	48.39
17 - 32	9	29.03
more than 32	6	19.35
Missing	1	3.23
Number of Developers		
1 - 6	15	48.39
17 - 12	8	25.81
more than 12	8	25.81
Position		
Manager	14	45.16
Committee	1	3.23
Consultant	2	6.45
Employee	13	41.94
missing	1	3.23
Work Experience (Years)		
1 – 5	17	54.84
6 - 10	9	29.03
More than 10	2	6.45
missing	3	9.68

IV. FINDINGS

This section discusses the findings of the state of practice of software risk management, which includes the adoption software risk management processes --the risk identification, risk analysis, risk prioritization, risk management planning, risk resolution, risk monitoring, risk sign-off and risk post-mortem analysis; risk roles and responsibility; risk owner; risk management standard or model; risk management tools; and risk documentation.

A. The Software Risk Management State of Practice

Table V and Figure 1 shows the state of practice software project risk management process of all of the 31 companies.

TABLE V
THE SOFTWARE RISK MANAGEMNT PRACTICE

Phrase	Frequency	Percentage
Risk Identification	30	96.8
Risk Analysis	31	100.0
Risk Prioritization	24	77.4
Risk Management Planning	30	96.8
Risk Resolution	25	80.6
Risk Monitoring	26	83.9
Risk Sign-off	20	64.5
Risk Post-Mortem Analysis	15	48.4

Fig. 1. Software Risk Management Practice

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

From observation of the frequency, the state of practice can be divided of three groups. The first group --risk identification, risk Analysis and risk management planning, the frequency is about 30 out of 31 while the second group -- risk prioritization, risk resolution and risk monitoring, the frequency is about 25 out of 31. The last group --risk sign-off and risk post-mortem analysis the frequency are 20 and 15 out of 31 respectively.

Table VI, where: a is every project (100%), b is almost all (80 – 99 %), c is some (60 – 79 %), d is a few (40 – 59 %), and e is very few (less than 40 %), shows that the robustness of the practice of software risk process. Most of the answered of these phrases fall into a (every project), b (almost all), and c (some) except the practices of risk sign-off and risk post-mortem are spread out.

TABLE VI
SOFTWARE RISK MANAGEMNT PRACTICE

Phrase	a	b	c	d	e
Identification	12	7	6	2	1
Analysis	11	9	4	2	2
Prioritization	9	4	5	2	1
Management Planning	9	9	4	3	3
Resolution	7	6	5	1	4
Monitoring and Control	8	8	3	2	4
Sign-off	6	5	5	-	3
Post-Mortem Analysis	2	2	3	4	3

B) Risk Identification Practice

To identify software risks, 22 and 21 out of the 30 respondents answered that they use brainstorming and check lists techniques respectively while the least used techniques are Delphi method and risk dimensions respectively (Table VII).

TABLE VII
THE USE OF IDENTIFICATION TECHNIQUES

Technique	Frequency	Percentage
Check Lists	21	70.0
Top Ten Lists	5	16.7
Risk Dimensions	3	10.0
Interview	7	23.3
Brain Storming	22	73.3
Delphi Method	1	3.3

C) Risk Analysis

To performing risk analysis, decision analysis is the most use method (22 out of 31) while risk exposure is second (14 out of 31) (Table VIII).

TABLE VIII
THE USE OF RISK ANALYSIS TECHNIQUES

Technique	Frequency	Percentage
Risk Exposure	14	45.2
Decision analysis	22	71.0
Others	4	12.9

D) Risk Management Planning

Regarding risk management planning process, risk plan and contingency plan are the two most popular planning tools used (Table IX). 15 companies (48.4%) used risk plan and 14 companies (45.2%) used contingency plan.

TABLE IX
THE USE OF RISK MANMENT PLANNING TECHNIQUES

Technique	Frequency	Percentage
Risk Plan	15	48.4
Risk resolution/ Strategy	10	32.3
Contingency Plan	14	45.2
Others	1	3.2

E) Risk Roles and Responsibility

Regarding risk roles, 25 out of 31 respondents (80.6%) identified project manager as the person responsible for software project risk management.

The other 6 respondents answered that there are more than one person responsible for the software project risk management. They are project manager and client manager, project manager and teamwork, project manager, project coordinator and developer, and project manager, executive and development managers.

F) Risk Owner

Concerning risk owner, 19 out of 31 respondents (61%) identified that they assigned software to risk owners while 12 (38.7%) did not have risk owners.

G) Risk management standard or model

12 out of 31 respondents (38.7%) identified that they followed some risk management standards or models while 18 out of 31 respondents (58%) did not have any risk standard or model. 5 out of the 12 respondents reported that they used CMMI (RSKM) and 4 respondents used ISO/IEC 29110, 1 respondent reported that it used CRM, 1 answered that it used all ISO/IEC guide, ISO9000, and ISO 9001:2000 and 1 answered that it used both CRM and PMBOK.

H) Risk Management Tools

Table X shows that about 13 out of 31 the respondents (41.94%) used some tools in managing risks except for risk sign-off and risk post-mortem analysis there are only 8 (or 25.8%) and 6 (or 19.4%) out of 31 respondents reported the use of risk management tools.

Reported tools are vary. Microsoft excel is the most frequent reported tools (frequency of only 2) for every phrase of the software risk management phrases.

TABLE X
RISK MANAGEMNT TOOLS

Phrase	Frequency	Percentage
Risk Identification	13	41.9
Risk Analysis	13	41.9
Risk Prioritization	13	41.9
Risk Management Planning	14	45.2
Risk Resolution	13	41.9
Risk Monitoring	14	45.2
Risk Sign-off	8	25.8
Risk Post-Mortem Analysis	6	19.4

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

I) Risk Documentation

16 out of 31 respondents (51.61%) answered that they used risk register in their companies. Table XI shows the frequency and percentage of the risk recording for each software risk management phrases. The frequency varies from 20 to 26 except at the risk sign-off and risk post-mortem analysis phrase.

TABLE XI
RECODRING RISK

Phrase	Frequency	Percentage
Risk Identification	24	77.42
Risk Analysis	26	83.87
Risk Prioritization	20	64.52
Risk Management Planning	24	77.42
Risk Resolution	23	74.19
Risk Monitoring	23	74.19
Risk Sign-off	17	54.84
Risk Post-Mortem Analysis	14	45.16

V. Conclusion and Discussion

In 2003, Deldolph [18] discussed a number of reasons why software risk management is neglected. However, this study uncover a different story. From the 40 questionnaires returned, 31 companies (77.5%) answered that their organizations have a software risk management process. After 10 years, this may indicate that software risk management is not anymore ignored.

From the data analysis above, the general picture of the software risk management of Thai software firms can be concluded as the followings:

1. Of the 40 questionnaires returned, 77.5% answered their organization have a software risk management process.
2. The state of practice are of three groups. About 30 out of 31 perform risk identification, risk analysis and risk management planning while about 25 out of 31 practice risk prioritization, risk resolution and risk monitoring and the least practice phrases are risk sign-off and risk post-mortem analysis with the frequency of 20 and 15 out of 31 respectively.
3. 22 and 21 out of the 30 respondents (73.3% and 70%) identified that they use brainstorming and check lists techniques respectively.
4. In performing risk analysis, decision analysis is the most used method (71.0%).
5. Regarding risk management planning process, risk plan and contingency plan are the two most popular planning tools used (48.4% and 45.2% respectively).
6. 80.6% of the respondents identified project manager as the person responsible for software project risk management.
7. 61% of the respondents identified that they assigned software to risk owners.
8. 12 out of 31 respondents (38.7%) identified that they followed some risk management standards or models while 18 out of 31 respondents (58%) did not have any risk standard or model.
9. Only 13 out of 31 the respondents (41.94%) used some software tools in managing risk.
10. 16 out of 31 respondents (51.61%) answered that they used risk register in their companies.

The general picture above shows that Thai software firms seem to give more attention to risk identification, risk analysis, risk management planning, risk monitoring and control and left out other two phrases -risk sign-off, and risk post-mortem analysis.

Figure 2 shows that it is more or less similar to the findings in Sweden of Kajko-Mattsson and Nyfjord [5].

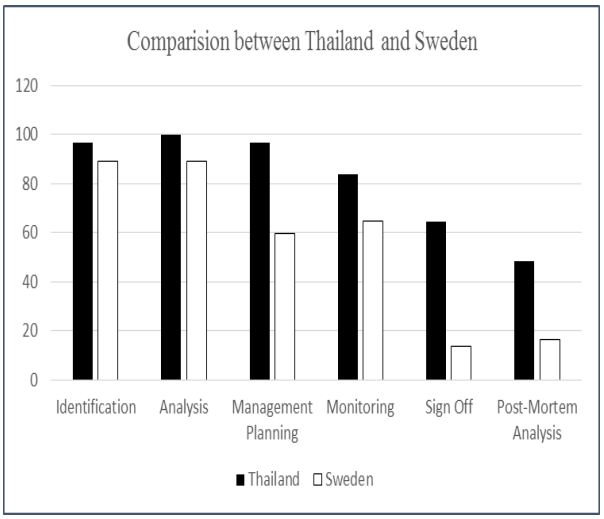

Fig. 2. The percentage comparison between Thailand and Sweden

This indicates the discrepancy between theory and practice as suggested in Kajko-Mattsson and Nyfjord [5], as they put it that the industry studied has not implemented all the activities prescribed by model found in the literature.

To explain this phenomenon, it is hypothesized that the two phrases -risk sign-off, and risk post-mortem analysis are seen from the industry as less significant. This is because most of the literature covered only four major processes –1) risk identification, 2) risk analysis, 3) risk planning, and 4) risk monitoring and control and similarly left out risk sign-off, and risk post-mortem analysis. However, Kajko-Mattsson and Nyfjord [5] indicated that these two processes are of great important for the long term effective software risk management.

References.

[1] B.W. Boehm. "Software Risk Management: Principles and Practices," *IEEE Software*, vol. 8, number 1, pp.32-41, 1991.
[2] R.N. Charette. *Software Risk Analysis and management*, McGraw-Hill, New York, 1989.
[3] B. Freimut, S. Kartkopf, J. Kontio, and W. Kobitzsch. "An Industrial Case Study of Implementing Software Risk Management," *ESEC/FSE 2001*, Vienna, Austria, 2001.
[4] P.L. Bannerman. "Risk and Risk Management in Software Projects: A reassessment," *The Journal of Systems and Software*, vol. 81, pp.2118-2133, 2008.
[5] M. Kejko-Mattson and J. Nyfjord. "State of Software Risk Management Practice," *IAENG International Journal of Computer Science*, vol. 35, number 4, November 2008.

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[6] E. E. Odzaly, D.Greer, and P. Sage. “Software Risk Management Barriers: an Empirical Study,” *Third International Symposium on Empirical Software Engineering and Measurement 2009*, pp. 418-42, 2009.
[7] R. Leihman, and J. VaanBuren. "The Risk of Not being Risk Conscious: Software Risk Management Basics,” *STSC Seminar Series*, Hill AFB, UT.
[8] Project Management Institute, *A Guide to the Project management Body of Knowledge (PMBOK)*, 3rd Ed. ANSI/PMI 99-001-2004, PMI, Newton Square, PA, 2004.
[9] B. Hughes, and M. Cotterell. *Software Project Management 5th ed*, pp. 163, McGraw-Hill (UK), 2009.
[10] IEEE 1540, *IEEE 1540 Standard for Lifecycle Process-Risk Management*, IEEE New York, NY, 2001.
[11] W.M. Han and S.J. Huang. “An Empirical Analysis of Risk Components and Performance on Software Projects,” *The Journal of Systems and Software*, vol. 80, number 1, pp. 42-50, 2007.
[12] L. Wallace and M. Keil. “Software Project Risk and their Effect on Outcomes,” *Communication of the ACM*, vol. 47 number 4, pp. 68-73, 2004.
[13] R. Schmidt, K. Lyytinen, M. Keil, M. and P. Cule. “Identifying Software Project Risks: An International Delphi Study,” *Journal of Management Information Systems*, vol. 17, number 4, pp. 5-36, 2001.
[14] K. Lyytinen, L. Mathisen, and J. Ropponen. "A Framework for Risk Management," *Journal of Information Technology*, vol. 11 number 4, pp.275-285, 1996.
[15] Software Engineering Institute/Canegie Melon University (CMI/SEI), *"Software Risk Management,"* URL:http://www.sei.cmu.edu/library/abstracts/reports/96tr012.cfm, Access May 2011.
[16] S. C. Misra, V. Kumar, and U. Kumar. "Different Techniques for Risk Management in Software Engineering: A Review," SAC Conference, pp.196-205, 2006.
[17] J. Konito. *Software Engineering Risk Management: A Method Improvement Framework, and Empirical Evaluation*, Ph.D. Thesis, Department of Computer Science and Engineering, Hensinki University of Technology, Finland, 2001.
[18] F.M. Dedolph. “The Neglected Management Activity: Software Risk Management,” Bell Labs Technical Journal, vol. 8, Issue 3, pp.91-955, 2003.

Tharwon Arnuphaptrairong is An Assistant Professor in Business Information Technology at the Department of Statistics, Faculty of Commerce and Accountancy, Chulalongkorn University, Thailand. He received a B.Sc. Degree in Statistics from Chulalongkorn University, a M.Sc. in Computer Applications from Asian Institute of Technology, Bangkok, Thailand, and a Ph.D. Degree in Management Sciences from University of Waterloo, Canada. His research interests include Software Project Management, Software Risk Management, Software Cost Estimation and Empirical Software Engineering.

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)