ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/330456150

Risks management in software development capstone projects

Conference Paper · September 2018

DOI: 10.1145/3274005.3274024

CITATIONS
0

READS
1,844

2 authors:

Pekka Mäkiaho
Tampere University
6 PUBLICATIONS **14** CITATIONS
SEE PROFILE

Timo Poranen
Tampere University
47 PUBLICATIONS **235** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Skills, education and future of work View project

All content following this page was uploaded by Pekka Mäkiaho on 08 March 2020.

The user has requested enhancement of the downloaded file.

Risks management in software development capstone projects

Pekka Mäkiaho
Faculty of Natural Sciences
University of Tampere
Tampere, Finland
pekka.makiaho@uta.fi

Timo Poranen
Faculty of Natural Sciences
University of Tampere
Tampere, Finland
timo.t.poranen@uta.fi

ABSTRACT

Project risk management is one of the most important knowledge areas that a project manager should know well. During university level studies, computer science students participate in different kind of capstone projects where they can learn risk management in practice. In this paper, we study risk monitoring in student's software development capstone projects. We analyze risk identification, foreseen and unforeseen risks, mitigation, and reactions after a risk is realized. Most common realized risks were related to time management of the project. Main mitigation techniques were good documentation, version control usage and open communication. Teams reacted to risks by rescheduling or speeding up their work

CCS CONCEPTS

• Software and its engineering → Software creation and management → Software development process management → Risk management

KEYWORDS

Software project, Risk management, Project management, Capstone project

ACM Reference format:

Pekka Mäkiaho and Timo Poranen 2018. Risks management in software development capstone projects. In Proceedings of 19th International Conference on Computer Systems and Technologies (CompSysTech'18). *ACM, Ruse, Bulgaria, 5 pages.* https://doi.org/10.1145/3274005.3274024

1 INTRODUCTION

Project Management Body of Knowledge [1] lists risk management as one of the most important knowledge areas in project management. Proper risk management ensures that a project team and management is prepared for unfortunate events that might occur during the project. In the beginning, the project team produces a risk list and then the risks are monitored throughout the project lifecycle.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
CompSysTech'18, September 13–14, 2018, Ruse, Bulgaria
© 2018 Association for Computing Machinery.ACM ISBN 978-1-4503-6425-6/18/09…$15.00
https://doi.org/10.1145/3274005.3274024

Risk management is part of core knowledge areas of project work. In software development capstone projects, a team implements a real software product. Throughout the project, team should practice planned risk management processes to ensure that the project finishes successfully. Common risks that a team can meet are often related to personnel, requirements and schedule [2].

In this paper, we study risk management in students' software development capstone projects. We analyze risk identification, foreseen and unforeseen risks, mitigation, and team's reactions after a risk is realized. For weekly risk monitoring, Metrics Monitoring Tool -application (MMT) [3] is used. Our data was gathered from seven capstone projects during academic year 2017-18.

Underlying research questions are the following: i) How student teams identified risks? ii) What kind of risks the teams met during the project and which risks were foreseen and which were unforeseen? iii) How risks were mitigated and what were the team's reactions after a risk was realized?

The rest of this paper is organized as follows. Literature review on risk management is given next. Then risk management process of the project courses and data gathering are discussed in detail. After that, data is analyzed and discussed. Last section draws conclusions and maps future research directions

2 RISK MANAGEMENT PROCESS

Project risk management is one of the ten knowledge areas that a project manager should be aware of [1]. Bannerman's [4] definition of risk management as "a set of principles and practices aimed at identifying, analyzing and handling risk factors to improve the chances of achieving a successful project outcome and/or avoid project failure", summarizes earlier variation of definitions [2,4,5].

Project Management Body of Knowledge [1] defines a risk as "an uncertain event or condition that, if it occurs, has a positive or negative effect on a project's objectives." In practice, risks can be thought as those events that cause problems to a project.

Risks can be categorized into project, product or business risks. Project risks are related to project's schedule or resources, product risks are related to quality of the developed software, and business risks are related to the organization developing the software [6].

Risk management process includes the following activities: risk identification, risk analysis, risk planning and risk monitoring [6]. In risk identification, risks are identified, and a list of potential risks is produced. Checklists can be used to help risk identification. Sommerville [6] lists the following risk types to help identification: technology, people, organizational, tools, requirements and scheduling risks.

In risk analysis, risks are assessed based on their probability to occur and how severe (impact) consequences are. Assessment can be done in different ways, but often used probability classes are very low, low, medium, high and very high with corresponding numerical values from 1 to 5. For severities, the same classes and numerical values can be used. Based on the probabilities and severities, risks can be prioritized by multiplying probability with the severity.

Risk planning phase concentrates on planning to manage and mitigate risk events. For each risk, avoidance, minimization and contingency strategies are planned. Avoidance strategies are related to avoiding risks to occur, and minimization strategies are related to minimize risks consequences. Contingency plans are related to actions that should be done after a risk is realized.

Risk monitoring is regular activity to monitor changes in probabilities and consequences of risks. During a project, risk probabilities change often. For example, technology risks are usually higher in the beginning of a project, but after a proof-of-concept solution is implemented, probability decreases. Sometimes a risk can be even removed from the risk list.

Common risks that can occur during a software development project are misunderstanding requirements, lack of management commitment and support, lack of adequate user involvement, failure to gain user commitment, failure to manage end user expectation, changes to requirements, and lack of an effective project management methodology [2,6,7].

Student projects differ from company projects in several ways: i) Projects have a schedule but there is no real budget. Schedule is based on academic teaching periods ii) Student teams are not experienced; members are often doing their first project. iii) If there is a (student) project manager, he/she is only a little more experienced (older student) than the other team members (younger students). iv) Students are not working full time with the project and they are studying other courses at the same time. v) Student teams are working with different projects, and there is not much collaboration among the other teams. Deliverables are not dependent on other projects' outcomes. vi) The student project organization is small, in addition to the team, project manager and client, there are usually teacher responsible and course assistants. Because there are fundamental differences between industrial and student projects, also the risks are different. Ahtee and Poranen [8] reported that most common student project risks are related to tools and skills to use the tools, technological problems, scheduling and working or studying other courses during the project. Koolmanojwong and Boehm [9] reported the following main risks in software engineering class projects: architecture complexity and quality tradeoffs, personnel shortfalls, budget and schedule constraints, COTS and other independently evolving systems, customer-developer-user team cohesion, requirements volatility, user interface mismatch, process quality assurance, requirements mismatch, and acquisition and contracting process mismatches. Vanhanen and Lehtinen [10] reported that, the number of identified problems per team varied between 103 and 247, and that they were mainly related to system functionality, system quality, communication and taking responsibility.

Schneider and Visa [11] have studied agile practices in student projects. One of their remark is that in an educational setting it is important to highlight the importance of documenting as a means to reduce risks.

3 PROJECT MODEL AND DATA GATHERING

During the fall semester 2017, 35 students participated in a course named Project Work (PW) and four students participated in a Software Project Management course (SPM) at the University of Tampere. Four project groups were formed so that a student who participated in the master level SPM-course was a project manager and the bachelor level PW-course students were project members. There were not enough students on the SPM-course so the rest three out of the seven groups had no nominated project manager.

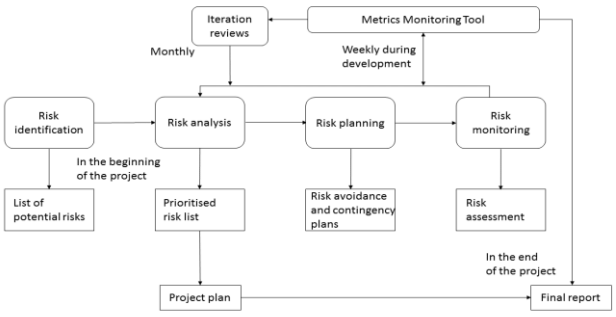

Figure 1. Risk management process, deliverables and monitoring using Metrics Monitoring Tool (adapted from Sommerville [6]).

The formed project teams had a real client to whom the teams were due to deliver a working piece of software with documentation. The students used 100-200 hours per person for the project and got credits related to the working hours. The teams had quite free hands during the project; the supervisor of each project ensured that good

project work and agile practices were followed. All the teams had five mandatory meetings with the supervisor and the client. In the first meeting, the project plan was reviewed, then there was a review meeting approximately once a month. Finally, there was a final meeting in which the final report was reviewed and feedback was given. Most of the teams had more reviews than those that were compulsory. It was also mandatory to report weekly to the supervisor by using Metrics Monitoring Tool [3]. The tool was used to monitor the metrics of each project like working hours, risks, requirements, test cases. Risk management process with related deliverables of the project teams is illustrated in Figure 1.

The groups were formed in week 36/2017, and the deadline for the project plan review was at the end of week 39. The students were instructed to identify possible risks of their project and to estimate the probability and the impact of the risks in their project plan.

The risks identified in the project plan were transferred to MMT and the students were instructed in weekly reporting to re-evaluate if new risks have risen or if the impact or the probability of any risk had changed.

The projects were due to end by the final meeting by the week two, January 2018. The teams wrote a final report, which included, among other things, information on the risks that realized during the projects and if the risks were predicted (foreseen risks) or if they just came as a surprise (unforeseen risks). There were also two Moodle questionnaires for the teams, the first in the middle of the project and the second after the project. In the Moodle questionnaires, it was asked how the risks were identified and what were the sources of the risks. Furthermore, the students were asked about monitoring the risks, what kind of actions the team had to mitigate the risks and if they had to react to the realized risks.

4 ANALYSIS OF DATA

Table 1 shows that during the nine projects' lifecycles, there were identified 3-7 risks per project (avg. 4.9). The total number of the identified (foreseen) risks was 34. On those risks, four realized (12 %). On the top of that, there were eight risks that were unforeseen, i.e. not mentioned in the project plan or in the MMT-reports.

Project Nr	Risks in the project plan	Realized risks not foreseen	Realized risks foreseen
1	Project team breaks up, single member leaves the project, project is too hard	Illness of a project member	
2	Interruption in the use of systems needed in development, project member's short sickness, longer sickness/Injury, losing documentation or source code, scope of the project increasing too much, burnout, dropout of a member	Unavailability of a member, communication problem: piece of information did not reach a group member	
3	A team member quits, client is not available to provide required support, team members could not work or are not available as expected, lack of skills, expertise or experience, a requirement feature is too complicated, some features are technically difficult to implement		Team members could not work as expected
4	Requirements set too high, falling behind the schedule, miscommunication, team member quits, time resources of a team member, technical capabilities	Lack of information from the client's partner prevented some visualizations	Time resources of a team member
5	Issues with OptiTrack system, Inaccessible lab, Sickness, Lack of motivation	Problems with time management, someone leaving group	
6	Project members are busy, inexperience in development, accidents, cases of illness	Not enough time to finish all the features.	Project members are busy, cases of illness
7	Accidents and illness, Lack of motivation, Group size changes	Problems with time management	

Table 1. Identified, foreseen and unforeseen risks

Four out of the seven groups identified risks related to a single member's health, like accidents or getting ill. The same number of groups (4/7) identified risks related to single member leaving the project and to too difficult or wide requirements. The other risks

were found in three or less of the projects. The most common unforeseen risks were related to time management or communication. All the foreseen realized risks were related to the lack of time resources of project members.

Table 2 shows how the risks were identified and monitored and which actions there were done toward the risks in these seven teams. Mostly the teams identified the risks together though one team mentioned that the risks came almost solely from the project manager. Three teams left the monitoring of the risks to the project manager, two teams discussed the risks in their meetings and two teams had no other process for monitoring the risks but reporting them to the Metrics Monitoring tool

Project Nr	Identifying risks	Monitoring	Mitigation	Reaction after the realization of a risk
1	PM's proposal	Risks were monitored by every member of the team.	Documentation, version control, open communication	Rescheduling
2	Team discussion	Team discussed on risks them in weekly meetings.	Documentation, version control, backups, decreasing the scope, keeping up good spirit	
3	Team discussion, client's comments.	Team reported having no formal process for monitoring.	More time asked, skipping other courses, chosen familiar technologies	Rescheduling
4	Team discussion	Risks are evaluated in weekly meetings by the project manager	Reporting honestly, rescheduling, dividing task equally, helping others	Rescheduling
5	Team discussion, proposals from the client and the supervisor	Group discussions: both in Slack and face-to-face meetings.		A member took the duties of the left project manager
6	PM's proposal adjusted by the project members	PM checked the situation weekly.	Studying	
7	Team discussion	PM monitored	Regular meetings,	Speeding up
	and client proposal. Project manager fine-tuned.	and edited the risks.	coding sessions, working closely.	

Table 2. Risk identification, monitoring and actions toward risks.

There were 34 foreseen risks. There were actions against fourteen risks for decreasing the probability and actions against ten risks to decrease the impact. On seven risks there were actions both for decreasing the probability and for decreasing the impact.

The most reported actions were documenting, version control and backups which both decreased the probability of risks events (e.g. losing output of a project) and decreased the impact of risks (e.g. a member leaving the project).

The eight risks that were unforeseen could not have been prepared. However, it is worth noting that on the rest (34 foreseen risks) only sixteen risks had some actions towards them. It means that 18 risks (2.6 per project) were just accepted. All the actions after a risk had realized related to rescheduling the work, which was done after the five (out of twelve) realized risks.

In Figures 2 and 3 it is shown screen captures from MMT [3] showing the probabilities and the impacts of the risks of an example project. The project was chosen as an example because it identified and assessed the risks regularly by the whole team.

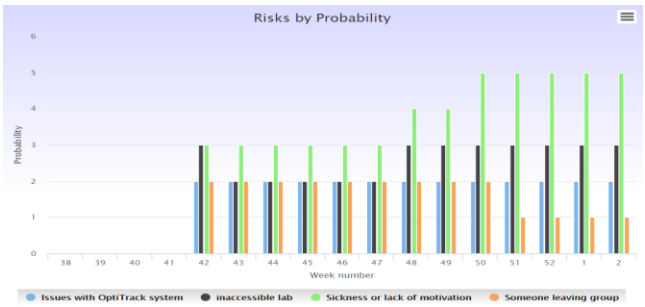

Figure 2. Risks by probability class of the team 5.

Project number 5 identified four risks. Issues with OptiTrack system was a risk of which probability sustained on a level 2 through the project. The impact of the risk was originally three but increased first to level 4 and finally to the level five as the project committed to the technology. On the end of the project, the impact decreased to level one.

Risk named 'sickness or lack of motivation' had a probability 3 in the beginning of the project but it increased to level 5; the impact changed too, being the highest on the beginning of the second half of the project.

The project had to use a laboratory, which was also used by a research staff. As there were no reservation system for the students,

there was always a risk that the team couldn't use the premises as they wanted. The probability of this risk varied between 2 and 3. The impact of this risk increased from 2 to 5 as during the project, the accessibility came more important. In the last week, the impact increased to level 1 as the lab was needed only for a demo session.

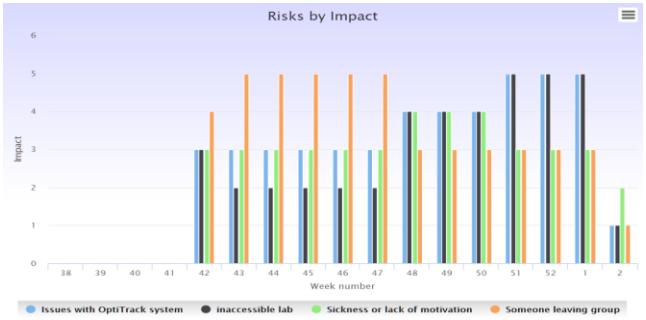

Figure 3. Risks by impact of the team 5.

The fourth risk reported by the team 5 was 'Someone leaving group'. The probability was only two in the beginning of the project and decreased to level one by the end of the project. The impact of this risk was on the level four in the 1st week and increased to the highest level 5. In the middle of the project, the impact was decreased to three and on the last week to level one.

5 CONCLUSIONS AND FUTURE WORK

In this article we studied risk management practices in students' software development capstone projects in university level studies. Risk identification was mainly done by the whole team. The checklists were not used. Most common realized risks were related to time management either the whole project or an individual member. The main mitigation techniques were good documentation, version control usage and open communication. Teams reacted to risks by rescheduling or speeding up their work.

Teaching practices of the capstone project courses can be improved by emphasizing the usefulness of risk checklists and regular monitoring of risks in team meetings.

Main weakness of this paper is the small number of observed teams. In future, this research can be repeated to cover more teams. It could also be analyzed when the risks realized in the projects.

REFERENCES

[1] A guide to the Project Management Body of Knowledge, Project Management Institute, 2017.
[2] Arnuphaptrairong, T. Top Ten Lists of Software Project Risks : Evidence from the Literature Survey, In Proceedings of the International Multiconference of Engineers and Computer Scientists, 2011, 6 pages.
[3] Mäkiaho, P., K. Vartiainen and T. Poranen. MMT - A Tool for Observing Metrics in Software Projects, International Journal of Human Capital and Information Technology Professionals 8(4), 2017, pp. 27-37.
[4] Bannerman, P.L. Risk and risk management in software projects: A reassesment. The Journal of Systems and Software 81, 2008, pp. 2118-2133.
[5] Boehm, B.W. Software Risk Management. Tutorial. IEEE Computer Society, 1989.
[6] Sommerville, I., Software Engineering 7, Addison-Wesley, 2004.
[7] Boehm, B.W. Software risk management: Principles and Practices, IEEE Software 8(1), 1991, pp. 32-41.
[8] Ahtee, T., T. Poranen. Risks in Student's Software Projects. In Proceedings of the 22nd Conf. on Software Engineering Education and Training (CSEET '09), 2009, pp. 154-157.
[9] Koolmanojwong S. and Boehm B., A look at software engineering risks in a team project course, In Proceedings of the 26th Conference on Software Engineering Education and Training (CSEET '13), pp. 21-30.
[10] Vanhanen, J. and Lehtinen, T. O. A, Software Engineering Problems Encountered by Capstone Project Teams, International Journal of Engineering Education, 30(6), 2014, 1461-1475.
[11] Schneider, J.-G. and Vasa, R., Agile practices in software development - Experiences from student projects, in Proceedings of the 2006 Australian Software Engineering Conference, 2006, 10 pages.

View publication stats