

Risk management plan

PMoC

8.2.2003

Version 1.1

Johan Forsbäck

T-76.115 Risk management plan PMoC

Version	Date	Author	Description
1.1	7.2.2003	JF	
1.0	28.10.2002	JF	First version of risk management plan

Table of Contents

1. Introduction	3
2. Risk measurement	3
3. Risk identification	3
3.1 Update of risks during first and second implementation phase	4
4. Risk analysis	4
5. Risk planning	5
6. Risk monitoring	8
6.1 Comments on current risks of interest from a monitoring point of view when entering phase I3	8
References	9

1. Introduction

Risk management is one of the most essential parts of planning a software project, as it consists of many different aspects, often hard to anticipate. Such aspects are the people that work on the project, the customer, technical and technological aspects, the project organization, the tools used, requirements and estimation.¹ In this risk management plan the risk management described in Software Engineering by Ian Sommerville will be used.^[1]

2. Risk measurement

In this document the risks will be divided into two categories, probability and effect according to the following tables:

Table 1: Probability of a risk

Table 2: Effect of a risk

Probability	
<10%	Very low
10-25%	Low
25-50%	Moderate
50-75%	High
>75%	Very high

Table 1

Effect
Catastrophic
Serious
Tolerable
Insignificant

Table 2

3. Risk identification

The risks detected at early stage in this project are as follows, divided into main subcategories:

Risk type	Possible risk
Technology	SVG-module will not meet expectations
	Chosen implementation language (Java) doesn't meet efficiency requests
	Post project software integration fails
People	Person conflicts
	Staff turnover
	Sickness
Tools	IDE is unfamiliar to project group
	Other tools are unfamiliar, difficult to use
Organization	Communication fails
Requirements	Misunderstanding of customer
	Customer requests change
Estimation	Inexperienced time and project estimation

¹ Software Engineering, Ian Sommerville © 2001, page 87

3.1 Update of risks during first and second implementation phase

Risk type	Possible risk	I1-I3
Technology	SVG-module will not meet expectations	
	Chosen implementation language (Java) doesn't meet efficiency requests	
	Post project software integration fails	
People	Person conflicts	
	Staff turnover	
	Sickness	
Tools	IDE is unfamiliar to project group	
	IDE conflicts with customer ides	X
	Other tools are unfamiliar, difficult to use	
Organization	Communication fails	
	Modules to implement are delayed if other dependent modules are not ready	X
	Too much time is still needed for meetings	
Requirements	Misunderstanding of customer	
	Customer requests change	
	Project scoping fails	X
Estimation	Inexperienced time and project estimation	
Implementation	Project requires more experience than available	X

4. Risk analysis

Table includes possible changes in probability and effect for earlier detected risks

#	Possible risk	Probability	Effect	Changed/Added
1.	SVG-module will not meet needs for the project	Moderate	Catastrophic	C
2.	Chosen implementation language (Java) doesn't meet efficiency requests	Moderate	Serious	
3.	Post project software integration fails	Low	Serious	
4.	Person conflicts	Low	Serious	
5.	Staff turnover	Very low	Serious	C
6.	Sickness	Moderate	Tolerable	
7.	IDE is unfamiliar to project group	Very low	Tolerable	C
8.	Other tools are unfamiliar, difficult to use	Low	Tolerable	C
9.	Communication fails	Low	Serious	
10.	Misunderstanding of customer	Moderate	Serious	
11.	Customer requests change	Low	Serious	
12.	Inexperienced time and project estimation	High	Tolerable	
13.	IDE conflicts with customer IDEs	Very Low	Tolerable	A
14.	Project scope not limited enough	High	Serious	A
15.	Modules to implement are delayed if other dependent modules are not ready	Moderate	Serious	A
16.	Project requires more experience than available	Low	Serious	A
17.	Too much time is still needed for meetings	Moderate	Tolerable	A

5. Risk planning

The risks are named, their probability and possible effect mentioned, and plans for preventive measures as well as response in case that the risk becomes real described

Risk # 1.	SVG-module will not meet needs for the project
Probab. / Effect	Moderate / Catastrophic
Preventive measures	Exploration of SVG-module and it's capabilities
Response	1 st degree – things not in SVG must be implemented “by hand”. 2 nd degree – if not possible to implement by hand very critical

Risk # 2.	Chosen implementation language (Java) doesn't meet efficiency requests
Probab. / Effect	Moderate / Serious
Preventive measures	Sufficient knowledge of Java should prevent this to some degree.
Response	Knowing

Risk # 3.	Post project software integration fails
Probab. / Effect	Low / Serious
Preventive measures	Communication with the customer
Response	Out of project reach

Risk # 4.	Person conflicts
Probab. / Effect	Low / Serious
Preventive measures	From the beginning of the project an attitude of openness, space for different opinions, honesty and frank communication has been emphasized on, thus minimizing conflicts
Response	Any conflicts should be dealt with immediately when recognized, through open communication and an attitude of honesty!

Risk # 5.	Staff turnover
Probab. / Effect	Very low / Serious
Preventive measures	The same as to avoid personal conflict, with the addition of trying to share the tasks according to skills and wishes within the group, thus motivating through interesting tasks
Response	Taking into account the nature of this project a staff turnover wouldn't be possible. If more than one group member would leave the group, the whole project should be questioned.

Risk # 6.	Sickness
Probab. / Effect	Moderate / Tolerable
Preventive measures	Not stressing anyone beyond his limits. The group structure within the team supports the idea that several people have at least on some level knowledge of other members parts, so that absence of a few members shouldn't cause a critical problem at first hand.
Response	If the sickness occurs at a critical time someone else in the group will have to handle the undone tasks.

Risk # 7.	IDE is unfamiliar to project group
Probab. / Effect	Very low / Serious
Preventive measures	Controlled choice of IDE
Response	A too unfamiliar IDE shouldn't at any price be selected!

Risk # 8.	Other tools are unfamiliar, difficult to use
Probab. / Effect	Low / Serious
Preventive measures	Controlled selection of other tools
Response	Minimizing the amount of people that need to learn unfamiliar difficult tools.

Risk # 9.	Communication fails
Probab. / Effect	Low / Serious
Preventive measures	Enough emphasis is put on understanding each other, including the customer
Response	Within the time limits, meetings are held to get a clear enough understanding.

Risk # 10.	Misunderstanding of customer
Probab. / Effect	Moderate / Serious
Preventive measures	Enough meetings are held, where all needed issues are discussed
Response	Within the time limits, additional communication could be established.

Risk # 11.	Customer requests change
Probab. / Effect	Low / Serious
Preventive measures	Trying to question and understand the requests of the customer to get to know what the customer <i>really</i> wants.
Response	If the changed requests are reasonable and possible to implement, changes to the software will be considered.

Risk # 12.	Inexperienced time and project estimation
Probab. / Effect	High / Tolerable
Preventive measures	The estimations are based on previous registered efforts, when available.
Response	New time and effort estimations are done if needed.

Risk # 13.	IDE conflicts with customer IDEs
Probab. / Effect	Very Low / Tolerable
Preventive measures	As far as possible, the same IDEs are used
Response	Necessary porting or modifications are done.

Risk # 14.	Project scope not limited enough
Probab. / Effect	High / Serious
Preventive measures	The scope is constantly kept in mind, discussed, and worked on not to exceed the groups' efforts.
Response	An even stricter limitation to the scope must be done.

Risk # 15.	Modules to implement are delayed if other dependent modules are not ready
Probab. / Effect	Moderate / Serious
Preventive measures	A detailed enough planning of the architecture has been made, and a constant planning of the following implementations steps is made.
Response	More efforts should be put to implement the lagging module.

Risk # 16.	Project requires more experience than available
Probab. / Effect	Low / Serious
Preventive measures	Enough time is reserved for needed studies to gain enough experience.
Response	If really more experience is required than held in the group, other ways to solve such problems must be found.

Risk # 17.	Too much time is still needed for meetings
Probab. / Effect	Moderate / Tolerable
Preventive measures	Meetings participation is optimized so that only the individuals concerned will participate. Emphases are put on keeping effective meetings that are planned on beforehand when possible.
Response	If needed the time limits can be stressed to some degree.

6. Risk monitoring

Risk monitoring is done by the project manager together with the rest of the group on a regular basis, in combination with normal communication during the project

6.1 Comments on current risks of interest from a monitoring point of view when entering phase I3

1. SVG-module will not meet needs for the project

Some problems have been discovered, and if not solved, they could be very critical. One such concerns adding information into existing XML-tags using Batik. That could be very demanding to implement by hand, and would require much unallocated time and effort. At the moment the possibilities to these things using Batik is investigated.

4. Person conflicts

To my personal delight the group has come along very well, and a genuine attitude of team work has been of great importance for the group to even come this far.

7. IDE is unfamiliar to project group

We have had to work at times with several IDEs (JBuilder and AnyJ) and switching between these a few times, but this hasn't caused any bigger problems. It has mostly been a question of not wasting time on installations and such.

10. Misunderstanding of customer

On some points there has been long discussions concerning issues such as architecture, GUI and the IO-module, but so far there hasn't been any serious misunderstandings that would have caused loss of time.

12. Inexperienced time and project estimation

This is one thing that the group has had to deal with to some extent, but all members have done their best to meet up deadlines and the result has been most often good, if not surprisingly good.

14. Project scope not limited enough

Sufficient scope limitation is clearly considered the biggest threat for this project. This has been taken into account during customer meetings, both in how many features to implement, and what way to do the implementation. On the following customer meetings the final scope is to be established.

17. Too much time is still needed for meetings

This is a product of many other risks, such as #14 and #10. Due to the nature of this project a very strict limitation of the length and amount of customer meetings is not possible as it is both important for the customer to some degree to know and understand how things are implemented, and for the group to understand the requirements and needs of the customer. Because of this there is a big risk that the budget for customer (and group) meetings will be exceeded. But that is considered better than not meeting customer expectations (within realistic group efforts) or misunderstanding customers' requirements.

References

1. Software Engineering, Ian Sommerville, Addison-Wesley 2001
2. The Riskit Method for Software Risk Management, Jyrki Kontio