Risk Assessment and Mitigation

Cohort 1, Team 1:

Connor Blenkinsop
David Luncan
Emily Webb
Muhidin Muhidin(MO)
Sam Laljee
Sooyeon Lee

1. Introduction

In terms of software engineering, risks can be defined as any sort of existing threat factors that could negatively impact the development of an ongoing project. The early model of risk management was initially conceived by Boehm in the late 1980's (Kontio, 2001) [1]. With further additions to Boehm's prototype model by Charette in 1989, risk management was successfully implemented in software engineering (Kontio, 2001).

There have been more recent discussions and suggestions in the software engineering community [2], however, considering that Boehm's risk management method still remains effective in practice, we decided to adopt Boehm's risk management model as our primary design choice.

2. Risk Assessment

2.1. Risk Identification

Boehm's model suggests the catalogue of top-10 risk items when it comes to software engineering (Stern and Arias, 2011) [3]. Bearing Boehm's list in mind, we classified the risks with the following 6 categories: technology, requirement, product, organization, personnel, and external environment.

Category	Description
Technology	Any risks arising from constraints in technical capabilities; software-wise, it could mean certain libraries or codes fails to serve their purpose; hardware-wise, it could mean the device in use can be subjected to various issues
Requirement	Any risks associated with user requirements and system requirements; this could arise from changes in requirements proposed by customers; often some requirements could be very hard to meet
Product	Any risks regarding the final outcome of the project; the product must be give the customer complete satisfaction; the performance of final product might be disappointing and of bad quality
Organization	Any risks related to the overall management of the project; poorly managed scheduling, lack of leadership and cooperation; unorganized project could get lost in process
Personnel	Any risks due to personnel shortfalls; team members could have insufficient knowledge and be unsuitable; unexpected circumstances leading to lack of participation by people
External Environment	Any risks originating from unforeseeable changes in external landscape; sudden announcement by the department or government

Table 1 : Risk Category and Descriptions

2.2. Risk Analysis

Having identified the categories of risks, we further specified the expected risks for each category and analysed their 2 attributes: severity and likelihood. By giving both attributes 3 levels of status, each labelled as *Low (L), Moderate (M), and High (H),* we decided to explain its severity and likelihood. As this is a fairly small-sized project, we figured that having 3 levels would be the most suitable option.

Level(s)	Severity	Likelihood
Low	The damage caused by risk is marginal and can be fixed easily, affecting the project marginally	The chance of risk occurring is very low, and the risk is unlikely to happen
Moderate	The damage caused by risk is tolerable but should be fixed, affecting the project moderately	The chance of risk occurring is moderate, and it is reasonable to expect the risk to happen
High	The damage caused by risk is more than serious and must be urgently fixed, affecting the project significantly	The chance of risk occurring is considerably high, and the risk is very likely to happen

Table 2 : 3 Levels of Severity and Likelihood

Below is the risk register with following columns: risk ID, risk category, risk description, risk severity, risk likelihood, mitigations, and risk owner. It is ordered by risk ID and some of the cells are colour-coded to visually aid understanding.

Risk Assessment and Mitigation - Eng1 Team1

ID	Category	Description	Severity	Likelihood	Mitigations	Owner
R1	Technology	Any Github issues related to the project resources can disrupt the progress	H	L	Regularly download files locally and keep copies of the repository	Muhidin & Conner
R2	Technology	Any other software we use such as Google Drive and Facebook Messenger could have problems, making it unable to check our progress	M	L	Prepare an emergency solution that can be always available, such as contacting via university email and phone number	Everyone
R3	Technology	Any electronic devices or networks in use fails, and possibly results in loss of data	H	M	Frequently save your work and utilize various storage applications. When a device or network goes down, use the university's laptop and PC	Everyone
R4	Technology	Any 3rd party libraries and tools utilized behaves unexpectedly during project	H	M	Always keep your knowledge in tools up-to-date and test it thoroughly	Connor & Muhidin
R5	Technology	Upon testing, errors and bugs are discovered within the product's code	M	H	Anyone available should frequently check the Github repository and look into the code to find errors	Connor
R6	Technology	Upon building the website, links to deliverables don't work due to IT reasons	M	M	Add URLs one at a time and check whether it is broken or not for each update on website	Muhidin
R7	Technology	Customer's technical environment isn't compatible with the product, resulting in customer dissatisfaction	H	L	Acquire prior knowledge in customer's IT settings and make sure the product can be run smoothly	Sooyeon
R8	Requirement	To meet the requirements is beyond the team's capacity and cannot be accomplished	H	L	Discuss requirements at the initial stage of product development and try best to enhance the team's capability with training	David
R9	Requirement	Customer continues to change the set of requirements, making it harder to progress	M	M	Outline how those changes would impact the project, then devise a new plan. Convey to the customer that as the project gets closer to its end, changing it could not be easy to do so	David
R10	Product	The end product fails to function properly,	H	L	Test the final product on various occasions and seek	Sam

		for instance, it is very slow and difficult to use			how the problem could be solved. Keep coming back to architecture and coding.	
R11	Product	Customer is dissatisfied with the end product for unforeseen reasons	H	M	Arrange a meeting with the customer and ask for feedback. Implement amendments as soon as possible.	Sooyeon
R12	Product	Users of the product reports errors that were not discovered upon development	H	H	Aside from test codes, try the final product from a user's perspective. Gain insights in user experience.	Sam
R13	Organization	Unrealistic scheduling and inadequate task delegation leading to missing deadlines	H	L	Make sure we understand our capabilities. Implement Gantt charts to keep up.	Emily
R14	Organization	Lack of communication reduced the productivity and resulted in personal conflicts	M	L	Keep the meeting log and record the content of each meeting.	Everyone
R15	Personnel	High number of absence by team members negatively affect the progress of project	H	L	Flexibly arrange the meeting time and ensure the tasks are distributed evenly. Find alternative ways to participate e.g. Zoom call	Everyone
R16	Personnel	Members do not understand the project workflow and needs assistance	L	M	Communication between team members is essential. Help each other out.	Emily
R17	Personnel	Members forget to deliver their job	H	M	Assign more members to take over certain tasks. Members at issue shall receive additional training.	David
R18	External Environment	Department announces any changes that could possibly be related to the project	L	L	Frequently check VLE and university emails. Enquire the department for further details if necessary.	Sooyeon

Table 3 : Risk Register

2.3. Risk Prioritization

Using a risk matrix with severity as the x axis and likelihood as the y axis, we can visualize what risks have high severity and likelihood at the same time and easily agree that those risks deserve to be prioritized. On the contrary, risks with low severity and likelihood could be handled later. This step will make it easier for the team members to judge what to do first.
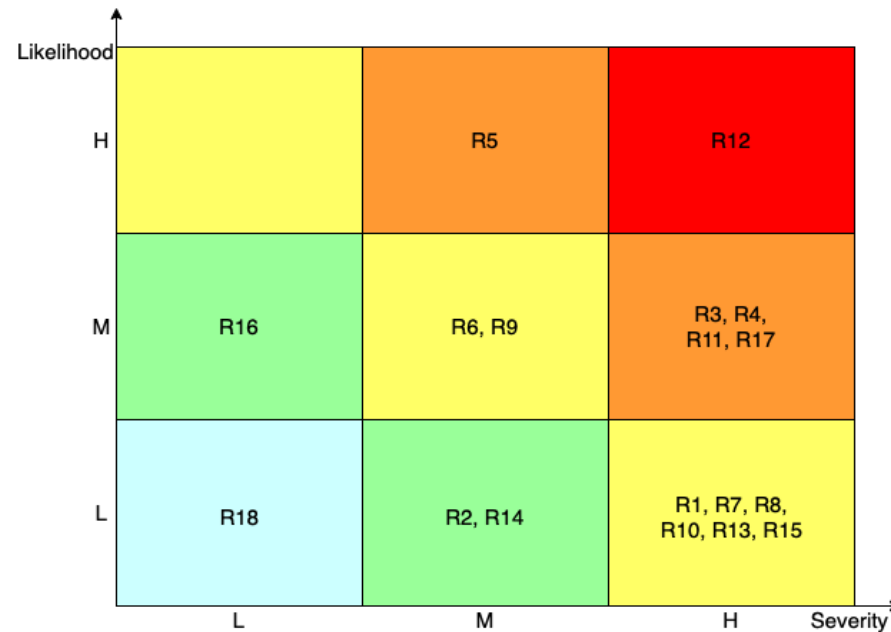

Figure 1 : Risk Matrix

It is clear that R12, which is a product type risk, needs to be prioritized the most and that R18, which is an external environment type risk, can be the least of the team's concern.

3. Bibliography

1. J. Kontio, *Software Engineering Risk Management: A Method, Improvement Framework, and Empirical Evaluation* (Doctoral dissertation, Helsinki University of Technology, Espoo, Finland). September 2001. Accessed on: January 28, 2022. [Online]. Available: https://aaltodoc.aalto.fi/bitstream/handle/123456789/2350/isbn951225655X.pdf?sequence=1
2. *A Review of Two Software Risk Management Methods,* SAPM: Course Blog. Accessed on: January 28, 2022. [Online]. Available: https://blog.inf.ed.ac.uk/sapm/2014/02/13/a-comparison-of-two-software-risk-management-methods/
3. R. Stern and J. C. Arias, Review of Risk Management Methods. *Business Intelligence Journal* 4(1), 60-61. January 2011. Accessed on: January 28, 2022. [Online]. Available : https://condor.depaul.edu/dmumaugh/readings/handouts/SE477/SERIM_Article_3.pdf