Risk Assessment and Mitigation

Mozzarella Bytes | Team 18

Assessment N°4

Daniel Benison

Elizabeth Hodges

Kathryn Dale

Ravinder Dosanjh

Callum Marsden

Emilien Bevierre

Risk Management

Risk is "an uncertain event or condition that, if it occurs, has a positive or negative effect on a project's objectives" [1]. It is important to mitigate risks as it allows more time and resources to be spent creating a product that fulfils the client's needs, rather than fixing avoidable errors.

Risk identification: Potential risks were individually mind-mapped before being discussed in a structured group brain-storming session [2]. As people had time to think of risks first, everyone had something to contribute, making it a productive session. We then read numerous top ten risk lists [1] as well as Caper Jones's software risks [2] and SEI's risk taxonomy [3] to learn about common risks and added relevant ones to our risk register.

Risk register: Based on research [4,5,6] we formatted our table with the following headings

Heading	Meaning	Reason for including
Risk ID	Unique identifier for each risk	Documents can reference the risk directly
Risk type	Project, product or technology [7]	A way to categorise the different risks
Description	Explanation of the risk	People understand the risk and its effects
Likelihood	How likely the risk is to occur	A quantitative measurement to measure the scale of the risk. Combined to give a level of action needed.
Severity	Impact the risk would have on the project if it occurred [8]	
Urgency	Level of action needed	Prioritises risks determined from matrix below
Mitigation	Strategies to prevent and lessen the effects of risks	Provides concrete action that we can take through the project to minimise that risk
Owner	Person who mitigates and manages the risk	A risk owner increases the likelihood we will make decisions to minimise the risk.

Terminology and classifications: Project risks cover risks that arise from the requirements, members leaving the group, and stakeholder's needs changing; technology risks are risks to do with languages, methods, standards and project functionality [7]. Likelihood and severity is ranked on a scale of low (L) to high (H). Low means it is either highly unlikely/has little impact, medium means that the risk might happen/ will have a noticeable but not unmanageable impact, high means the risk is highly likely/will have a significant impact on our project. Likelihood and severity combined outlines how urgently action needs to be taken to minimise the risk, and is determined from this matrix:

Risk assessment matrix	High Severity	Medium Severity	Low Severity
High likelihood	*High*	*High*	*Medium*
Medium likelihood	*High*	*Medium*	*Low*
Low likelihood	*Medium*	*Low*	*Low*

*(High: Control risk immediately, Medium: Take action to reduce risk, Low: No action needed)

How risk ownership was allocated: Ownership of each risk was allocated based on each person's role within the team (See pg.), which tasks they are planning to work on in future assignments (see pg.) and the total number of risks they have ownership of.

Risk management: Risk management is a continuous process that ends once the project comes to an end [8]. Every three weeks we will conduct a risk review where we identify new risks, look over tracked risks, close irrelevant risks and assess our risk management strategy; this ensures we are constantly aware of, and attempting to mitigate, risks.

ID	Risk Type	Description	Likelihood	Severity	Urgency	Mitigation	Owner	Updates
R1	Product	Clashing code written by multiple team members	M	M	M	- If an overlap is deemed likely (e.g. two tasks cross over in purpose) discuss with the team before starting. - Use GitHub version control to keep track of all versions of the project and to merge code edits.	EB & DB	N/A
R2	Technology	GitHub, where our source code and website is hosted, goes down.	L	H	M	- Back up code on local devices or in the cloud. - Assign a member of the team to remind everyone to back up their work weekly.	EH	N/A
R3	Project	Team member illness	H	L	M	- Edit documents on Google docs so other team members can take over if necessary. - Meeting notes will be taken and uploaded to the Google drive ASAP to allow the absent team member to catch up. - Communicate with the team via Messenger to stay updated on the state of the project.	ALL	N/A
R4	Product	Constantly changing requirements	H	M	H	- Maintain regular contact with the client to recognise change in requirements as soon as possible - Have a change management plan to modify the risk register, requirements and project plan to accommodate new requirements - Ensure code is modular and well documented so it can be easily adapted to accommodate changes.	CM and KD	N/A
R5	Project	Team member leaves the group	L	H	M	- All critical tasks are assigned to a minimum of two people so if someone were to leave there would be a team member who understood that part of the project and could teach someone what was required if needed.	ALL	N/A

R6	Technology	Features of the application framework becomes deprecated	L	M	L	- Adjust the Gantt chart to add time to learn and implement newer functions/ elements of the library - Plan time to rework the code if necessary	EB and DB	N/A
R7	Project	Misunderstanding the user's requirements	M	H	H	- Follow recommended practices to elicit requirements - Hold meetings to discuss, validate and update requirements with the client regularly - If conflicting or ambiguous requirements are noticed clarify them with the client as soon as possible - Keep the client involved in all stages of development	CM	N/A
R8	Product	Developing the wrong software functions	H	H	H	- Follow requirement engineering techniques to make sure requirements reflect the client's needs - Peer review the tests to check they test the required functionality - Frequently communicate within the development team so all members are clear on the functionality required - Implement the core functional requirements before coding additional features - If there is uncertainty whether a feature is needed clarify with the client before coding	DB and EB	N/A
R9	Project	Project is poorly managed	M	H	H	- Research methodologies to choose the one that best suits our project - Assign a team owner to oversee the project and a SCRUM master to ensure we are adopting the agile methodology - Monitor group progress using the Gantt chart every three days; modify the plan if necessary - Hold a group meeting to discuss solutions/ reallocate roles if the current management system is inadequate	KD	N/A

						- Expectation that team members will vocalise if they have been assigned too many/ few responsibilities		
R10	Project	Team members lack the technical skills to complete the project	L	H	M	- Research and practice Java in their own time - Plan plenty of time to implement the code	ALL	N/A
R11	Project	Ineffective communication between team members	M	M	M	- Use GitHub projects to allocate tasks - Expectation that every team member should participate in our virtual SCRUM every two days - Provide a way to communicate as a team (Facebook messenger) - Ensure each member is clear on their role/task before leaving meetings. - Hold regular in face meetings as well as virtual SCRUMS	KD	N/A
R12	Project	A team member could get stuck on their task, causing the project to be delayed	M	M	M	- Allocate roles according to team members' strengths and preferences - Work on tasks as a team wherever is reasonable - Keep an open dialog between group members to allow us to help one another	KD	N/A
R13	Product	Not all parts of the product are tested thoroughly, preventing the product from working as intended	M	H	H	- Requirements will be analysed prior to testing to allow suitable tests to be designed. - A Traceability Matrix will be used to ensure that all requirements are fully tested	CM and DB	N/A
R14		Team members in different locations may make communication difficult and make it harder to keep an awareness of what each team member is doing.	H	M	H	- Multiple forms of communication, including social media and shared online collaboration spaces (e.g. Google Drive) will be used to keep up with each other - An up-to-date copy of all deliverables will be stored in a shared area		
R15	Project	Global Pandemic may make it difficult to further develop the project	L	H	H	- All documentation will be edited online so all team members have access to it - Team members will be expected to keep an eye on the group chat to see any project developments	ALL	New Risk. At the time of writing this the UK has just gone into lockdown after the WHO have declared the spread of

								Covid-19 to be a pandemic. This is not a risk that we had considered before due to its extremely low likelihood however, now that it has occurred we must consider how best to work in the current situation.

References

1 T. Arnuphaptrairong, "Top Ten Lists of Software Project Risks: Evidence from the Literature Survey," *IMECS*, Vol. 1, Accessed on: [Online] Available at: https://www.semanticscholar.org/paper/Top-Ten-Lists-of-Software-Project-Risks-%3A-Evidence-Arnuphaptrairong/deaee8d192415f3facc01f224485bd9aa4b2ae34 2011

2 C.R. Pandian, *Applied Software Risk Management – A Guide for Software Project Managers,* New York: Auerbach Publications, 2007

3 M. J. Carr, S. L. Konda, I. Monarch et al "Taxonomy-Based Risk Identification", *CMUSEI-93-TR-6,* 1993, Accessed on: 12. 10.19 [Online] Available at: https://resources.sei.cmu.edu/asset_files/TechnicalReport/1993_005_001_16166.pdf

4 A. Dziadosz, M. Rejment, "Risk analysis in construction project – chosen methods", *Prodecia Engineering,* Vol. 122, pp.258-265, Accessed: [Online] Available at: https://www.sciencedirect.com/science/article/pii/S1877705815031239 2015

5 South Birmingham Community Health, "South Birmingham Community Health Emergency Planning Risk Register", NHS, Accessed on: [Online] Available at: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKEwjGs7D64-LlAhXUSsAKHbBkAokQFjAAegQIABAC&url=http%3A%2F%2Fwww.bhamcommunity.nhs.uk%2FEasySiteWeb%2FGatewayLink.aspx%3FalId%3D7582&usg=AOvVaw07GOyDm-Gyy17_9YJetU6Y 2018

6 C. Pink, "Significant Risk Register," *Surrey and Sussex NHS,* Accessed on: [Online] Available at: 2013 https://www.surreyandsussex.nhs.uk/wp-content/uploads/2013/02/5.5-Significant-Risk-Register.pdf
The four stages of managing risk according to the IAMT Cycle are: identification, analysis, mitigation and tracking.

7 D. Mankad, "Risks Management in Software Engineering," *International Journal of Advanced Computer Research,* Vol.2, No.4, pp.290 -293, Accessed on: [Online] Available at: http://www.oalib.com/paper/2564531 2012

8 J. L Mitchell, R. Black, *Advanced Software Testing Vol.3,* 2nd ed. Santa Barbara: Rocky Nook, 2015