Risk Assessment and Mitigation

Group 5 :
Callum Watson
Jack Guan
Craig Slomski
Helene Dima
Chase Mo
Kamrul Islam

Risk Management Process:

One of the largest components of project management is risk management. Hence, we have adopted the risk management strategy outlined in [1] which decomposes the process into four main stages:
Risk Identification: To identify the risks and categorising them by type.
Risk Analysis: Weighing up and prioritising the likelihood and consequences of the risks
Risk Planning: Taking up plans to reduce the likelihood and/or severity of those risks to minimise the impact on the project
Risk Monitoring: Regularly assessing the risk and reviewing plans as the knowledge and context of the risk becomes more clear.
We believe this will be a good strategy for our team to use as it decomposes the risk management process into 4 clear steps that can be followed. Each step is described and justified in more detail below:

Risk Identification:
As project management is new to a large proportion of our group, limiting our scope for types of risk down to Project, Product and Business might limit the extent to which we can come up with risks and ways to mitigate them. Looking at [2] we can see that they have split them down into Scheduling, Budget, Operational, Technical and Programmatic risks. However, this would not be a good fit for our team as this is only a small, non-critical software and low-cost project so some aspects of that model are irrelevant to us (such as market development or running out of budget). Therefore, we have opted to use the risk checklist in [3] where it focuses on these six types of risks:
Estimation (Estimating resources required to build the system), Organisational (from the environment the software is being developed), People, Requirements, Technology and Tools.
Not only do these more accurately represent and categorise our project, but they also put us in a strong position to identify risks.

Risk Analysis:
As we are using an expanded checklist, we will have a long list of risks with little idea on the impact it has on our project. Thus we will be using a risk register where for each risk, we have broken it down the likelihood and severity to assess which risks we need to spend additional time developing risk mitigation strategies for and review regularly and which to set aside but be aware of and develop if the situation arises. We have opted to use a Low, Medium, High ranking [4] as with a smaller project we just need to know what is likely to cause large issues and it would take more effort than it's worth categorising them further.

Risk Planning:
Following the prioritisation of risks, we have decided to devote more of our energy coming up with mitigation strategies for anything that is more high risk. We use a mixture of avoidance strategies (to reduce the likelihood), minimisation strategies (to reduce the severity) [5] but due to this being a small project, we are unlikely to have as many contingency plans as it is unlikely for there to be any major factors that need to be addressed. Due to the different roles within the group, each member will own different risk mitigations.

Risk Monitoring:
We are using Google Docs so the document is open and accessible to the team and "key processes" are "well documented so that even someone with limited knowledge of the subject could execute it" [6]. It is the Team Leader's responsibility to ensure this document is 'live' and update it accordingly. We do this to ease the weight off of other team members that have their own responsibilities in the team.

ID	Type	Description	Likel-ihood	Seve-rity	Mitigation	Owner
High Impact Risks (Monitored frequently and updated accordingly)						
R1	Technology	The software does not correctly function on a operating system	M	H	Test the software on both Windows and Linux after each major change Try to implement it in Linux during early stages of development.	Main Devs
R2	People	Team member becomes unresponsive or unable to complete work	M	H	**Pair/Group Work** [6] Increasing the bus factor on each task (assigning at least 2 people to any one important task) to ensure no Single Point of Failure. See R5a and R5b about sharing files, a strategy to further increase bus factor. **Fresh Documentation** [6] To ensure that tasks can be picked up where they left off with limited understanding of context	Team Pairs
R3	Organisational	The project falls behind schedule	M	H	Have a meeting every week, schedule using gantt charts leaving a week before the assessment is due as contingency.	Meeting Chair
R4	Requirements	Customer requirements change	M	H	Taking an agile approach to development to catch any changes to requirements with the stakeholder.	Team
R5	Tools	Code Files stored on an independent device, not accessible to other team members	M	M	Push latest version any code (UML, Java, Markdown) onto Git with descriptions of changes	Team & Librarian
R5b	Tools	Documentation Files stored on an independent device, not accessible to other team members	M	M	Always use Google Docs to edit the documentation files	Team & Report Editor
R6	Organisational	Lack of communication between the customer and the team	M	M	Once the requirements and architecture is fleshed out, talk to the customer to clear up any potential problems	Meeting Chair

Low Impact Risks (Monitored infrequently but are here to be aware of)						
R7	Technology	Team members don't have the correct hardware/software to perform the tasks	L	M	In the planning stage, agree upon which software and IDEs we are using. If any of them are too powerful for our computers, we can use the University Departmental Computers / Virtual Machine.	Team
R8	People	Lack of clear quick communication with the group	L	M	Discord Server to allow communication outside of Meeting Times which are: 4 hr/week during term time set aside for meetings 1 hr/week outside of term time set aside for meetings	Team
R9	Organisational	Location & Meeting times are not suitable	M	L	Meeting/Discord poll at the start of term to figure out a suitable time and following up and checking when the labs are available	Meeting Chair
R 10	Tools	Formatting and convention irregularities between UML Diagrams	M	L	Changed from Draw.io to PlantUML to ensure consistency	Team
R 11	Organisational	Risks becoming out of date or and mitigation strategies not being followed	M	L	Team meetings to assess if there are any risks occurring and team lead ensuring the correct risk measures are taking place and updating the mitigation strategies accordingly.	Project Lead

References:

[1] I. Sommerville, “Chapter 22.1 Risk Management,” in *Software engineering*, Harlow, Essex: Pearson Education Limited, 2016, pp. 644–646.

[2] “Types of risks in software projects,” *Software Testing Help*, 05-Dec-2022. [Online]. Available: https://www.softwaretestinghelp.com/types-of-risks-in-software-projects/ [Accessed: 06-Dec-2022].

[3] I. Sommerville, “Chapter 22.1 Risk Management,” in *Software engineering*, Harlow, Essex: Pearson Education Limited, 2016, pp. 647

[4] T. Morphy, “20 common project risks - example risk register,” *stakeholdermap.com*, 2008. [Online]. Available: https://www.stakeholdermap.com/risk/register-common-project-risks.html [Accessed: 07-Dec-2022].

[5] I. Sommerville, “Chapter 22.1 Risk Management,” in *Software engineering*, Harlow, Essex: Pearson Education Limited, 2016, pp. 650.

[6] S. Dubois., “What's the bus factor of your team and how to increase it,” *Medium*, 16-Jun-2020. [Online]. Available: https://medium.com/management-matters/whats-the-bus-factor-of-your-team-and-how-to-increase-it-8bdfb63361fc [Accessed: 01-Dec-2022].