Risk Assessment and Mitigations

Edited by Team 13

Darcy Adams

Samin Alborzi Movahhed

William Dunlop

Nam Duong

Jakub Grizmil

Davids Kacs

Team 15

Joe Wrieden

Benji Garment

Marcin Mleczko

Kingsley Edore

Abir Rizwanullah

Sal Ahmed

Risk Assessment and Mitigation v0.3

Preface

This document is intended to be read by the system engineers as well as the customer. The document is added to as the plan will evolve (refer to version history for this document's various iterations).

- v0.1 Creation of the document, outlining the planning in introduction. Risks added.
- v0.2 Risks have been expanded upon and revised. Our risk management process added to the introduction. Introduction could be expanded upon.
- v0.3 Introduction and risks final edition.

Introduction

Our Risk Management process consisted of first planning how we would carry out the next stages such as deciding the basic template we would use for our risk register. We then brainstormed as a group so as to voice everyone's concerns and systematically identified all possible risks stemming from various categories, from risks associated with time constraints and team organisation to the software itself. These were condensed, with risks with very low severities or likelihoods being eliminated, and refactored into a risk register as planned.

In the register, they were prioritised according to likelihood and severity, after we conducted qualitative analysis, since we bore in mind that this is a smaller project with negative ramifications not being critical. Thus we concluded that the level of detail did not need to be high either, and so did not focus on carrying out quantitative analysis, which may be less subjective but was too detailed and time consuming for our purposes. This prioritisation allowed us to put our focus on reducing risks we had determined to have problematic consequences. Our risks are tabulated with an unique ID for each one, the type/area, a description, likelihood, severity, mitigation and ownership. We felt all these columns are appropriate and necessary to fully be prepared in order to deal with any problems that may arise. We use three levels of likelihood and severity: low, moderate and high.

Next, we planned to reduce the impact and probabilities of risks that required attention, by laying out avoidance strategies, mitigation strategies and contingency plans. Finally, we delegated the responsibility of monitoring certain risks to every individual in the group for the duration of the project. This consisted of reassessing the likelihoods and severities of risks under an individual's jurisdiction. Each person came back to the register regularly and updated the likelihood and severity of each risk they were responsible for and if they detected a great enough change, they would have to report the change in status and raise their concerns in our frequent meetings. New risks could also emerge and are added.

Our adherence to this carefully constructed process may have taken up greater portions of our time, but allowed for us to work well together as a team and provide deliverables of good quality on time as we had prepared for any risks that may have come beforehand.

Risk Register

ID	Type	Description	Likelihood	Severity	Mitigation	Owner
R1	People	Too many people working on the coding aspect and not enough people working on the documentation, or Vice Versa.	M	H	Split the team into half. Half of the team working on the code and the other working on the documentation. Ensure there is communication between each team so that any updates can be made clear. This is done via Discord.	William Dunlop
R3	People	Team members not being sure of what tasks they will need to perform during the week, specific sections of documentations/ code implementations are not complete, leading to a delay in completion of project/documentation.	M	M	Have a weekly to do list which provides each member of the task that they will need to complete during that week and deadlines for the tasks which will help with the workflow and keeping the team on track.	William Dunlop
R5	Product	Product is over-engineered, i.e. features that are not required are implemented, unnecessarily using up resources.	L	M	Enforce that the Engineers stick to the Architecture models as defined in the documentation.	Davids Kacs
R6	Technology	User's computer has insufficient memory to deal with seven or more boats at a time.	M	H	Inspect program for optimisation opportunities each sprint, such as removing memory intensive operations from loops.	Davids Kacs
R7	Technology	User's computer has an outdated or incompatible version of Java and is unable to execute the game reliably.	L	H	Write the program in a widely supported version of Java, especially by libGDX, such as Java 8.	Davids Kacs
R8	Project	Initial schedule is greatly under-estimated and does not provide adequate time for the project.	L	H	Over-estimate length for each task to ensure that even if they are delayed, the deadline is not delayed too (i.e. the Cone of Uncertainty [1]).	Darcy Adams
R9	Project	Requirements are introduced or altered by the client and must be implemented before the deadline.	M	M	Abide by agile methodologies, in this case Scrum, with short weekly sprints in order to address changes in requirements.	Nam Duong

R10	People	Requirements implemented are not to the stakeholders/customers liking.	H	H	Providing the customers with the week's prototype/deliverable in weekly meetings to discuss any of the requirements they feel may be at odds with the current version of the implementation (as in clarification of requirements).	William Dunlop
R11	Project	Members are not accountable for any faults during the project, leaving ambiguity as to who should correct them.	L	M	Enforce ownership of each task when they are delegated in each sprint.	Jakub Grzmil
R12	Project	Collaboration tools become unavailable for the team, thus limiting the productivity for a given time period.	L	M	Guarantee each member is contactable through more secure means such as email if necessary.	Davids Kacs
R14	People	No one on the team has high enough coding skills to produce the required product.	L	H	Email the client with the problem. Ask for an extended deadline.	Darcy Adams
R15	Technology	Software Engineers are unsure on what kind of libraries can be extended from in order to implement the game.	M	H	Research is done beforehand into licencing and what libraries can be reused.	Nam Duong
R16	People	Someone on the team suddenly stops responding and ceases to complete the work that they were set.	M	M	Assign another member to cover the member's work so the project can still go as planned. Make sure the bus factor is never low enough for the project to fail.	Samin Alborzi Movahhed
R18	Product	All requirements are not implemented within the product.	H	M	A documentation is provided to the customer/stakeholder with the requirements that are not implemented in the game with a justification as to why the requirement has not been implemented.	Samin Alborzi Movahhed

R19	Product	Libraries we reuse may be flaky and may affect our product quality.	M	H	Research is done beforehand into good quality libraries in order to pick the most suitable and reliable game library for the project.	Davids Kacs
R20	People	Stakeholders/Customers becoming unavailable for some period of time during the project implementation	L	H	Change the schedule of the work so the focus for that week can become the other parts of the project which don't need immediate attention from the customers. If the unavailability is extended more than expected, ask for an extended deadline.	Samin Alborzi Movahhed
R21	Project	Inaccurate progress tracking leading up to being behind the schedule without knowing, and resulting to delay.	M	H	Assign a second person for every member to be proofreading the documentations and checking the code, making sure it is done correctly and also helps to be on top of the schedule.	Samin Alborzi Movahhed
R22	Product	Half-hearted risk management fails to detect major project risks, causing stress and increasing the probability of project failure	M	H	As mentioned in the introduction, having an adequate risk register from the beginning of the project and completing it throughout the whole implementation time.	Samin Alborzi Movahhed
R23	Product	Incomplete specification/ conflicting requirements, becoming apparent as the coding and integration begins	H	M	Use a dedicated Product Manager to raise the issue and make sure everyone (especially coders) in the group is aware of it and make a decision to change the requirements/ implementation as a group	William Dunlop

References

[1] - McConnell, S. (2006). *Software Estimation: Demystifying the Black Art*. Microsoft Press

Bibliography

- Griffiths, M. *The Top Five Software Project Risks*. Project Smart. Retrieved 11 November, 2020, from https://www.projectsmart.co.uk/top-five-software-project-risks.php

- Robertson, M. (2020, July 08). *What are the 10 biggest risks in software development?* Codebots. Retrieved November 11, 2020, from https://codebots.com/library/way-of-working/what-are-the-10-biggest-risks-in-software-development

- Naidamast, S. *Common sense software engineering – Part III: Risk analysis.* Jaxenter. Retrieved 1 February, 2021, from https://jaxenter.com/common-sense-software-engineering-part-iii-risk-analysis-121131.html

- *Risk Management in Software Development and Software Engineering Projects.*Castsoftware. Retrieved 1 February, 2021, from https://www.castsoftware.com/research-labs/risk-management-in-software-development-and-software-engineering-projects