

Risk Assessment and Mitigation

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 Introduction.
- v0.2 Risks have been added but require revision.
- v0.3 Risks have been revised. The additions could be revised.

Introduction

Our process of forming the Risk Assessment and Mitigation was to firstly come up with all the areas where there might be risks, for example, in team organisation or with the software. Then we went through each area one by one and came up with risks. This is a smaller project and problems with the system are not critical. Therefore the level of detail doesn't need to be high. Our risks are tabulated with a 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. This is so that we group up the risks that are most dangerous to our system and be more prepared to deal with them. We have assigned one or more people to a risk. As a group we all come up with a mitigation strategy that we're all happy with. Each person comes back to the register regularly and updates the likelihood and severity of each risk they are responsible for.

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	Marcin Mleczko
R2	People	Specific sections of documentation are not complete and specific requirements that were intended to be implemented are no longer implemented.	L	M	Have deadlines for specific sections of code/documentation to be completed for each week to ensure the team is on track and everything that was set is completed	Sal Ahmed
R3	People	Team members not being sure of what tasks they will need to perform for that work, 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 which will help with the workflow	Sal Ahmed
R4	Project	Software Engineer becomes unavailable	M	M	Involve a 2nd Software Engineer.	Joe Wrieden
R5	Product	Product is over-engineered, i.e. features that are not required are implemented, using up resources unnecessarily	L	M	Enforce that the Engineers stick to the Architecture	Kingsley Edore

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	Joe Wrieden
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	Benji Garment
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]	Marcin Mleczko
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	Abir Rizwanullah
R10	People	Requirements implemented are not to the stakeholders/customers liking.	H	H	Have weekly meetings with stakeholders/customers to discuss any issues there might be with the requirements(i.e. clarification of requirements)	Kingsley Edore
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 each sprint	Marcin Mleczko
R12	Project	Collaboration tools become unavailable for the	L	M	Guarantee each member is contactable through	Kingsley Edore

		team, thus limiting the productivity for a given time period			more secure means such as email if necessary	
R13	Product	Coders do not pay attention to the changing requirements	L	M	Have a weekly meeting where the documentation team discusses what changes have been made to the requirements due to the team-customer meeting	Joe Wrieden
R14	People	Noone 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.	Abir Rizwanullah
R15	Technology	Software engineers are unsure on what library to use in order to implement the game.	M	H	Research is done beforehand in order to pick the most suitable game library for the project	Benji Garment
R16	People	Someone on the team suddenly stops responding and cease to complete the work they were set	M	M	Make sure there are enough people covering other members so they can finish the uncompleted work	Sal Ahmed
R17	Technology	Users monitor may not have a 1920 x 1080 quality resolution which may affect gameplay quality	L	L	Provided a scalability option which allows the games to run smooth regardless of resolution	Benji Garment
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	Abir Rizwanullah

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>