UNIVERSITY OF YORK

DEPARTMENT OF COMPUTER SCIENCE

Risk Assessment and Mitigation
Cohort 2 - Group 18 - Octodecimal

Group Members:

Izz Abd Aziz

Phil Suwanpimolkul

Tom Loomes

Owen Kilpatrick

Zachary Vickers

Michael Ballantyne

The risk management process is split up into four main sections: risk identification; risk analysis; risk planning and risk monitoring.

Risk Identification

This stage is concerned with identifying all potential risks that could pose a major threat to the software engineering process, the software being developed, or the development organisation [1]. Exploring common risks to a software development project was the first step to identifying risks. This research then allowed us to adapt those risks to our own project. The team members assigned to this deliverable brainstormed possible risks together and they were added to the risk register. Each risk was given an ID so they can be referenced by other risks and documents. We also considered past experience in group projects for risks that almost or did occur. We looked into different categories of risks and added a column in the risk register to record it. Risks could either be categories as Product, Project or Business. A project risk affects the project's schedule or resources and a product risk affects the quality and completeness of the final game. Finally, a Business risk refers to something that could affect the group or company (us) that are developing, for example, the repercussions if we violated copyright law within our game. This was to assist with organisation and analysis/planning as there can be similar mitigation strategies for risks in the same category. As a result, a description of the impact was added as a column in the risk register to make it easier to complete risk planning.

Risk Analysis

Each discovered risk is considered and a judgement is made about its likelihood and severity (of impact). This relies on personal judgement and previous experience so there is no correct answer; it is just an estimation of the priority of the risk. The severity and likelihood were given a rating from 1 (low) to 5 (high) (anonymously) as this is fairly simple and allows the use of a risk matrix to calculate priority with the use of an anonymous rating system, to ensure that the true opinions of each member were recorded and not influenced by seeing the ratings of others in the team. Priority is the severity multiplied by the likelihood which results in four categories: very low (green); low(yellow); medium (orange) and high (red). The decision to make use of a colour coded system was to ensure that when looking at the risk register our eyes were naturally drawn to the more extreme risks, highlighted in more powerful colours. Once all risks were analysed and given a priority, those with lower priority due to having very low probability or only minor consequences, were removed.

Risk Planning

Once risks have been identified and analysed, strategies must be put into place to ensure these major risks do not threaten the project. These mitigation strategies can include: avoidance strategies, which aim to reduce the likelihood of a risk occurring; minimisation strategies, which reduce the impact of a risk and contingency plans, to deal with a risk when it arises. To develop these strategies we considered past experience of what did and didn't work when a risk arose. We considered what information should be collected throughout the project to uncover risks before they become serious. A mitigation strategy column was added to the risk register so when they occur appropriate strategies can be implemented. The owner of a risk is the individual who is ultimately accountable for ensuring the risk is managed appropriately [2]. To decide who should be the owner we considered who the risk will affect the most and who has the best ability to be able to prevent/manage it. To clearly indicate the risks' owners, an Owner column was added to the risk register.

Risk Monitoring

In order to appropriately monitor and review risks we encouraged team members to report new potential risks or risks that have changed throughout the project. A risk reassessment column was added to the risk register to keep a track of how often they should be checked. We dedicated time

during group meetings to talk about any potential risks that may arise and the status of already identified risks. We also created an additional document dedicated to the reassessment of risks (titled Risk Monitoring, URL.pdf). This doc contains a table which outlines the ID of the risk, the owner who is assessing the risk along with the date of the assessment and whether any action needs to be taken and the action taken if applicable. This was done in order to provide a recorded structure of when the risks had been reassessed so when a team member came to assess a risk they could see when it was last assessed and whether any action was taken and what action was taken, allowing clearer communication across the team.

b) Priority risk matrix:
X = Severity Y = Likelihood

5	5	10	15	20	25
4	4	8	12	16	20
3	3	6	9	12	15
2	2	4	6	8	10
1	1	2	3	4	5
	1	2	3	4	5

Risk Register [3]:

ID	Risk class	Risk description	Impact description	Severity	Likelihood	Priority	Prevention/Mitigation Strategy	Owner	Reassessment Frequency
1	Project	(Scope Creep) Continuous addition of new features beyond the initial scope of the project.	(Resource drain). The effort would be disproportionate to the marks given for the particular task and would be considered a waste of time and resources.	3	3	9	To make sure the team is not adding or changing features that shouldn't be added or changed. (Change control process). Document all the necessary features that are needed to be added and changes that are requested by the client.	Owen	Weekly
2	Project	(Communication failure) There has been a conflict between group members and communication has broken down.	Decreased productivity, team morale issues and dysfunctionality between team members	4	2	8	(Mitigation strategy for ID:1) Establish conflict resolution protocols, encourage open communication channels. Conduct regular team check-ins, address conflicts promptly	Tom	Weekly
3	Project	(Communication failure) Lack of communication causes multiple team members to do the same work	There will be multiple versions of the same work which will need combining/choosing between in a fair way to ensure everyone participates equally. Alongside delayed project timelines	4	3	12	Implement task tracking system, promote regular progress updates. We must establish clear task assignments, and encourage communication between members.	Tom	Weekly

4	Project	Lack of communication causes a team member to do too much of the remaining work	There has not been equal participation and there is not enough remaining work to make it even.	3	2	6	Have regular workload assessments and promote open communication regarding task allocation. Conduct regular check-ins on workload distribution, provide support for overwhelmed	Zachary	Weekly
5	Project	A team member becomes temporarily absent for a specified period of time.	Delay in task completion and redistribution of workload	4	3	12	Document handover procedures and ensure clear task delegation. Cross-training of team members across various tasks could be the contingency plan, however the risk scales with team size, therefore evaluating the work ethic of each and every member should be feasible. Establish contingency plans for temporary absences.	Izz	As needed, when a group member is absent.
6	Project	A team member becomes temporarily absent for an unspecified period of time	Uncertainty in task completion, increased workload for remaining team members and decrease in team morale	4	4	16	Regular check-ins with absent team member(s), distribute workload among remaining team members. Also (Mitigation strategy for ID:7)	Izz	As needed, while the group member is absent.
7	Project	One member permanently drops out	We would have only 5 people which may put pressure on the rest of the team due to increased workload. Also (Impact description for ID7, 8)	4	2	8	Establish contingency plans for permanent drop outs of a single team member and consult module leader. Distribute the remaining workload evenly amongst remaining team members.	Michael	As needed, if a group member/s drop out.
8	Project	Multiple members permanently drop out.	We would have a maximum number of 4 members which is not enough to complete the project. Also, (Impact description for ID:8, 9)	5	1	5	If some deliverables are dropped, the remaining team will work on the new deliverables with the remaining workload balanced across the remaining team members	Michael	As needed, if a group member/s drop out.
9	Project	A team member has been assigned too much work and reports they will be unable to complete the work on time	Delay in task completion.	4	3	12	Regular Evaluation and distribution of the workload according to the skillset of the members. Cross-training of team members across various tasks so that no concentration risk takes place. Have	Owen	Weekly

							multiple team members assigned to critical tasks so responsibilities can be shared if a team member is struggling.		
10	Project	A team member hasn't completed their work by the deadline and didn't report it	Project delays, compromised task dependencies	4	2	8	Clear reporting protocols, task tracking systems and establish reporting expectations in order to follow up on missed deadlines promptly	Zachary	Weekly
11	Product and project	There has been a drastic change in requirements	Increased workload, potential delays, scope creep	4	4	16	Robust change control process and immediate impact assessment. Also, negotiate changes with stakeholders.	Owen	As needed, if or when our requirements change.
12	Product and project	The change in requirements increases the workload by a large amount but the deadlines are not pushed back	Overworked team, Increased workload, compromises in terms of quality (lacklustre product)	4	3	12	Negotiate deadline adjustments and time extensions. Also assess the resource reallocation and workload.	Tom	Weekly (until it's no longer a problem)
13	Product	Inadequate testing leads to issues with the product	Lacklustre product and customer dissatisfaction	4	3	12	Comprehensive testing strategy alongside an intricate benchmark for quality needs to be established. Also surveys to incorporate feedback.	Phil	Weekly
14	Product	A tool relied on for a large portion of the project becomes unusable.	A new tool must be found and code rewritten which will require extra resources and increase workload.	5	1	5	Research all tools used extensively to ensure they seem reliable for the foreseeable future. Research alternates so they can be quickly implemented if necessary.	Zachary	Biweekly
15	Product	When changes are made to the code, it works on the changer's device but not on different hardware/software.	This could mean not all members of the team can run the code and develop it. It also means when the game is being marked, the module leaders cannot run the code.	5	4	20	Make use of CI to ensure that there is a central code repository that all members can access and use which would supply an identical codebase for all, with the aim of eliminating the problem entirely.	Phil	Weekly
16	Product	When merging two branches there are merge conflicts or similar issues	This means the current version of the code will not run/is incorrect or new code cannot be added to the main branch.	3	4	12	Analyse the code and git log to see how this has occurred and resolve any merge conflicts. If unsure, revert to the last commit.	Phil	Weekly

17	Business	If we used images with inappropriate copyright licences or AI art as part of our project, this may infringe on copyright law.	This would mean that a large part, maybe all of our visuals would need changing (depending on how much had been done), increasing the workload and taking time away from other areas of the project.	3	3	9	Ensure that the images usage rights are known before use and have the correct licences and permissions for their intended use. Additionally, make sure any created art does not have any likeness or similarity to any major brands' imaging or products. Each time a piece of art is to be added, it has to be passed by Tom to ensure it is suitable.	Tom	Weekly.
18	Product and Project	Code is documented poorly.	Different developers may misinterpret the code, leading to errors further down the line of the development process. This would also increase overheads as developers seek clarification with each other for ambiguous pieces of documentation.	4	3	12	Use a standardised documentation format so all developers have clear guidelines on the format and content that their documentation should adhere to.	Michael	Weekly
19	Product	The game that we have taken over has some originally undetected errors.	Depending on the severity of the errors, some may be simple fixes, taking a small amount of time. However, if the error happens to be significant, this may take up a much larger amount of time and could compromise the quality of the final product.	4	3	12	Initially after gaining access to the source code of the game, Zachary and Owen will play through the game numerous times, doing different things each time in the aim of breaking the game to identify errors. This allows the majority of errors to be spotted immediately so they can be addressed quickly, or time can be put aside in the future to address them. User evaluation will also be helpful to flag up any errors later in the development cycle.	Zachary and Owen	Weekly
20	Project	Deliverables from Assessment 1 do not meet Assessment 1 criteria.	If the given deliverables from Assessment 1 do not meet the criteria, it will require a large amount of time to correct, so they are suitable for our own	5	2	10	When first gaining access to the deliverables, they will be reviewed against Assessment 1 criteria, to ensure that they are up to standard. If they are not, time can be scheduled to	Tom and Izz	Weekly

			Assessment 2 submission.				sort this.		

References

[1] I. Sommerville.(2015, Aug. 20). Software engineering [Online]. Available: https://eu.alma.exlibrisgroup.com/leganto/readinglist/citation/52275872540001381

[2] Office of the Chief Risk Officer (Stanford University). Definition of Risk Owner [Online]. Available: https://ocro.stanford.edu/enterprise-risk-management-erm/key-definitions/definition-risk-owner#:~:text=Risk%20Owner%3A%20The%20individual%20who,his%2Fher%20risk%20management%20efforts.

[3] K. Eby. (2018, Sept. 20). Agile Risk Register Template for Information Technology [Online]. Available: https://www.smartsheet.com/risk-register-templates