Risk assessment and mitigation

5.1 Introduction

Risk management is an integral part of managing a team. It is important to recognise any potential risks before they arise in the development of the project. Once a risk is identified there should be a plan in place to resolve it if it does occur.

Before deciding which format to use, we conducted research on Principles and Practices of Risk Management [1]. From this, we identified that we should use a Risk Register format.

We have used a Risk Register format, in the form of a spreadsheet, as a central repository for all of this information. We decided to use this format because it represents the risks in a clear, easy to read manner and we ordered them according to Risk Factor (Risk Likelihood * Risk Severity).

As this is a small project, and the software we are developing is non-critical, we decided to keep our measures simple. Therefore, Risk Likelihood and Risk Severity are labelled with either 1, 2, or 3 (Low, Medium, and High respectively). They are colour-coded to increase readability.

Using a Risk Register format also means we can assign an Identifier to each risk allowing us greater traceability when we monitor them, while also giving us the opportunity to store a description, impact, mitigation, and action alongside them. We also considered adding a Risk Responder to describe who would respond to each risk, but decided it would be most appropriate if we decided to tackle them as a team.

5.2 Risk Declaration

Every identified risk has the following properties:

- **ID Number :** A unique number to each risk allowing a risk to be identified quickly within the table.
- **Description:** Outlines the details of the risk and who it involves.
- **Likelihood:** Given a value 1,2 and 3, (representing Low, Medium, and High respectively) the likelihood is how likely it is that a risk may occur. It is colour coordinated to draw attention to the more likely risks and make it more readable.
- **Impact:** A description of how it will impact the project and who it will affect.
- **Severity:** A value between 1-3 to represent how severe the impact of the risk is on the project, (again representing Low, Medium and High).This is also colour coordinated on a scale, bringing attention to the more punishing risks
- **Risk Factor:** Determined by *Likelihood * Severity*. Risks with a high Risk Factor are likely to be more important, and to require more monitoring than risks with a low Risk Factor.
- **Mitigation:** The mitigation of the risk is what the stakeholders / development team must do to try to reduce the likelihood of the risk happening.
- **Action:** The action is what the stakeholders/development team must do once the risk has occurred to counter it.

5.3 Risk Register

Risk ID	Description	Likelihood (1-3)	Impact	Risk Severity (1-3)	Risk Factor (Likelihood * Severity)	Mitigation	Action
1	There is a change in requirements by the end user	3	Could affect the whole design of the project	2	6	Agile principles are being used to reduce the effect of a requirements change on the rest of the project.	Try to implement the requirement, but if need come to compromise, allowing for the requirement as much as possible.
2	Time estimates are inaccurate and deadlines are not met	2	Failure to meet deadline delaying other areas of the project and potential deterioration of design quality	3	6	Make sure deadlines allow extra time and include time for catching up in the schedule	If the whole team misses a deadline they must work as quick as possible to finish still producing quality product. The design has to be questioned to see if it is implementable in the time frame
3	Stakeholder conflict - disagreement between stakeholders	3	Disturbed development process	2	6	All conflicts are sorted as quickly as possible with minimum impact on the rest of the development	A smooth decision making process should be in place to resolve any conflicts and help avoid any further disagreements later in development
6	Software library issues	3	Time spent searching for a replacement for the library/ the code used from there	2	6	Ensure all libraries are tested before using them in the code.	Have an alternative method/library to implement the section of code as backup
4	Requirements are not complete	2	If not dealt with during design, may have a project missing features that the client wanted that have not been discussed	2	4	Have multiple meetings with the client regarding the requirements until both sides are completely happy.	If discovered later in the project that a requirement is missing, treat it as a requirement change and try to implement it as close as possible.
5	Requirements are not achievable	2	Team could spend time attempting to implement a requirement that is impossible, affecting the schedule. Client might feel that the project is incomplete.	2	4	Make sure all requirements are objective and there is a method in place to test to make sure they have been met	Discuss with clients and compromise its removal or discuss a new requirement which is achievable to replace the previous.
7	Requirements are ambiguous	2	Time spent outlining the requirement(s) at a later date	2	4	Ensure all requirements are clear, concise and objective	The team discuss how to make the requirement unambiguous, getting input from the client/other stakeholders if necessary
8	A team member is ill and falls behind on their work	2	Reduced team progress. Other team members might have to pick up their work	2	4	All team members know what each other member is doing and all work is saved in an accessible place.	The other member of the sub-group must help them on their section. If it is too much work for one person, then the rest of the group shall help out.
9	Sub-groups do not perform to the same standard	2	One sub-group might produce inferior work in their section, reducing the quality of the whole project	2	4	Sub-groups will be balanced in terms of skills and commitment. Reviews take place when work is ready to check the standard	In a review, if a subgroup is found to be producing sub-standard work, the review leader must explain to them what needs to be done and offer any help. The group will be given another deadline to meet with an update which again shall be reviewed.
10	Misunderstood requirements	2	Time must be taken clarifying	2	4	Ensure every stakeholder is clear on	The requirement must be clarified and then the clear

			the requirements and doing any required design			the requirements before implementation	requirement should be implemented
11	A team member leaves the team	1	Higher workload for the remaining team members	3	3	All work is spread out evenly between the rest of the team and all members know what another is doing.	The member's work should be shared out between the remaining team fairly.
12	Communication breakdown within the development team	1	Agile method efficiency reduced. Project could come to a halt	3	3	Have scrum meetings often and Reviews whenever artefacts are ready	Emergency team meeting to discuss where each member is with their work and a review of the current communication system in place.
13	Stakeholders become disengaged	1	Project progress reduced, at risk of coming to stand still	3	3	Ensure all stakeholders are clear on their role and the expectations of them. As soon as a problem may arise deal with it immediately with a meeting.	Emergency team meeting to discuss why this has occurred and how to get the stakeholders back engaged in the project
14	Compatibility issues between team member's codes	3	Error when merging the codes on the VCS	1	3	Ensure code-writers stick to the class diagrams, naming methods and variables correctly.	If error occurs, the writers solve the conflicts using Github's error system.
15	Design is not achievable/maintainable	1	Delay whilst redesigning the project. Ensure that the principles are met in the new design but that it is also more practical to implement	2	2	Ensure all stakeholders are happy with the design throughout and that it is viable	Remove any unnecessary/complicated features that are causing issues. If needed replace them with more viable options which still meet the requirements.
16	Communication breakdown with client(s)	1	Project may start to deviate from client's vision	2	2	Have regular communication with clients, updating them on the status of the project. This can be via email or face to face meetings	Call a face-to-face meeting with the client as soon as possible to discuss the project and how the breakdown has occurred.
17	A team member misses meetings and has a vital piece of work	2	Stop the team from progressing in that area	1	2	All documentation work is saved on google drive and all code on Github, using decentralised git, therefore all team members can access it without relying on another person	Retrieve work from google drive or latest backup if it has not been saved correctly.
18	Issues with VCS system or Github	2	Cannot assess or upload the project for others to use	1	2	Have a backup method of project collaboration which does not involve Github	Use the backup system indefinitely until Github is ready to be used again
19	Team members lack skills to do their role	2	Delay in progress until another team member can help or that team member learns the skills	1	2	Make sure each team member is happy with their role and they understand what they have to do.	Team members should ask for any help if they need it. If no-one can provide help from within the team the member should use other resources eg Lectures/Internet tutorials.
20	Team members do not work together effectively	1	Reduced productivity and potential conflicts	1	1	Ensure sub-groups within team work effectively together. Resolve any conflicts that arise immediately.	Emergency meeting to discuss any issues between the group and to deal with them in a professional manner. Adjustment to sub-groups if necessary.

5.4 Bibliography

[1] Boehm, B.W. "Software Risk Management: Principles And Practices". *IEEE Software* 8.1 (1991): 32-41. Web. Available: http://scholar.google.co.uk/scholar_url?url=http://csse.usc.edu/TECHRPTS/1991/usccse91-500/usccse91-500.doc&hl=en&sa=X&scisig=AAGBfm24XKw04S1LBVNvGIIQcp8w_39nwA&nossl=1&oi=scholarr&ved=0ahUKEwium9Kcq4zQAhWCxhoKHbnZCPAQgAMIGigAMAA [Accessed: 17- Oct- 2016].