Risk Assessment & Mitigation

Risk Declaration & Formatting

A risk is defined as anything that can threaten your project in the future, which is not confirmed to occur. Because of this, Risk Management is a crucial part of Project Management. Without accurate and consistent risk assessment, the project may be developed to a low quality, or not at all. The Risk Management Process is split up into the following sections; Identification, Analysis, Planning, Monitoring.
Each section will be addressed at some point in this document.

Risk Identification & Analysis

Each risk has been identified through group brainstorming and research, using software engineering pages such as [1], identifying multiple risks which could occur during each task or to a person. Consequently, each was assigned a unique alphanumeric ID, a Likelihood, Severity, Type and an Owner, who be responsible for the monitoring and handling of that risk. Some risks were decided to be insignificant or extremely unlikely to occur, and therefore were not included in the planning. If an alternative or additional risks are discovered or mentioned during the project, the plan will be updated by the Project Officer.

Likelihood, the probability of such risk occurring, was scaled as:
- Low - very unlikely to occur during the duration of the project.
- Moderate - medium chance of occurring during the duration of the project.
- High - extremely high probability of occurring during the project

Severity, the impact of if the risk were to occur, was similarly scaled as:
- Low - little impact on the completion of the project, insignificant effect.
- Moderate - some impact on the completion of the project, troublesome but manageable.
- High - serious impact on the completion of the project, potentially fatal.

The different types of Risks are defined and coded as below:
- Project - PJ: risks that can affect project schedule, management and team members.
- Product - PD: risks that would affect the quality or performance of the product, whether technical or implementation based, or documentation.
- Business - B: risks that affect the organisation developing or procuring the software.
- Technology - T: risks relating to the software or hardware used.
- People - P: risks relating to the individuals in the project team.
- Organisational - O: risks relating to the planning and environment.
- Tools - TL: risks relating to the software and tools used by the team members.
- Requirements - R: risks relating to changes in customer requirements and management of such.
- Estimation - E: risks relating to the estimation of the system characteristics and system resources.

During this brainstorming session, it was decided that we would apply our choice of development method to our risk monitoring. As we chose the Agile Development Method [2], the risk monitoring would occur on a weekly basis, with the owners of the risks coming together for a meeting to discuss any issues or current concerns with the current risks and potential ones we had missed initially. We will change the plan accordingly to ensure all risks are considered. The choice of multiple risk owners was made so to ensure that each could be considered carefully, in case one person found themselves overlooking some of those deemed less likely, causing cracks in the monitoring, and potentially allowing a problem to occur.
The following roles were assigned:
- Eleanor Bracegirdle - Project Officer, OO
- Emma Phillips - Product Officer, PO
- Merry Boyes - Technology Officer, TO

Due to the limited number of Business risks, the next large impact area was decided to be Technology and Tools. The Project Officer would monitor the business risks.

Risk Planning

ID	Risk	Type	Likelihood	Impact Severity	Mitigation & Monitoring	Owner
PJ1	Time required to develop software is underestimated.	PJ O E	Moderate	Moderate	Stick to requirements and set deadlines. Frequent reminders to complete work and of upcoming deadlines. Use of Agile development and management software such as ASANA.	OO
PJ2	Effort required to develop software is underestimated.	PJ P O E	Moderate	Moderate	Allow the estimated 300 student hours per team member. Regular meetings to ensure all collaborating sufficiently. Check effort going into required details, rather than extra requirements.	OO
PJ3	Weekly schedule pressure reduces productivity on the project.	PJ P O E	High	Moderate	Weekly meetings to ensure at least a certain amount of time in the week is given to work on the project. Project members who have fewer constraints can offer to take some work, as long as contribution continues.	OO
PJ4	Half-hearted risk assessment fails to identify major project risks.	PJ PD P O	Low	Low	Regular upkeep of risk assessment through Agile method of development. Weekly meeting of Risk Owners to discuss issues and/or occurrences.	OO
PJ5	Team members are permanently unavailable, through dropping out or long-term health problems, etc	PJ P O	Low	High	Avoid assigning tasks to a singular person. Regular updates and explanations of current work to rest of team members. Simplify product to reduce strain on remaining members.	OO
PJ6	Team members are temporarily unavailable, due to illness, family conditions, etc.	PJ P O	Moderate	Moderate	Same mitigation strategy as above risk. Keep regular contact, if possible, to keep unavailable member informed for when they return. If able, request team member to continue their work to reduce delays.	OO
PJ7	Team member fails to contribute/misses meetings on a regular basis.	PJ P O	Low	High	Regular team wellbeing and mentality checks. Contact person in question about situation. If problem continues, contact module supervisor.	OO
PJ8	Teams members have low motivation and so are not productive	PJ P O	Moderate	Moderate	Regular team wellbeing and mentality checks. If unmotivated or loss of interest, call a group meeting to provide support. If the situation is extreme, contact module supervisor.	OO

PJ9	Conflict between team members cause low productivity and miscommunication	PJ P O	Low	High	Team activities to encourage good, friendly communication. Frequent communication to keep all team updated. Attempt to resolve conflict once arises. If too severe, contact module supervisor as soon as possible.	OO
PJ10	Lack of communication between the members causes irregularity in the documentation and code.	PJ P O	Low	High	Regular communication about current progress in each document. Document review forms ensure at least one other team member checks a finished document and gives feedback. Collaboration between members working on code.	OO
PJ11	Team needs extra time to learn new software or programming languages.	PJ O TL E	Moderate	Low	TO will be responsible for vetting new software to be used, as well as introducing team members to the functionality. Additional meetings could be called if members are having trouble learning individually.	TO
PJ12	Excessive schedule, other university deadlines and pressures cause a reduction in productivity.	PJ O	High	Moderate	Regular team wellbeing and mentality checks to ensure team do not get overwhelmed with the pressures of the project and University. If extreme pressure is on a certain member, suggest they get support. Ensure collaborative group so no one member has more work.	OO
PJ13	Team members disregard main requirements to focus on parts of the design they are interested in.	PJ O R	Moderate	High	Regular requirement checking to ensure requirements are being completed.	OO
PJ14	Changes in project requirements.	PJ O R E	Moderate	Moderate	Clear, concise documentation and code to allow for easy changes as required, via modular code and flexible product design.	OO
PJ15	Files are lost, misplaced or deleted.	PJ T O E	Low	High	Regularly backup documents on different software or hardware. Use of the reliable Github software, making website addition simple. Version control system maintained throughout project duration.	PO
PD1	Failure to meet requirements.	PD R	Low	High	Regular monitoring of progress and requirement completion. Weekly meetings to ensure all requirements met.	PO
PD2	Inadequate or undetailed documentation causes errors in code and requirements.	PD O	Moderate	Moderate	Complete documentation review forms as soon as a team member says a section is complete to avoid long-standing inconsistencies.	PO

PD£	Complex/New technical feature cannot be implemented in the time remaining in the project.	PD T T E	High	Moderate	Prepare for such a feature by implementing it as soon as possible, and predict the time taken to adjust other tasks.	TO
PD4	The client is not happy with the final project.	PD B R	Low	High	Ensure an open line of communication is kept with the client. Weekly checks to ensure meeting all requirements. Agile development to allow for changes in all documentation if client changes a requirement mid-project.	PO
PD5	Project cannot be continued by other teams or the University.	PD O T TL	Low	Moderate	Clear, commented code with a modular structure. Detailed class and inheritance diagrams. Clear documentation and direction of project.	PO
PD6	Changes in requirements requires major design and code changes.	PD O R	Moderate	Moderate	Clear commented code with a modular structure, so code changes are easy to implement. Detailed class and inheritance diagrams with clear documentation to ensure ease of additional tasks.	PO TO
PD7	Limited testing causes simple and/or large bugs to be present in final product.	PD T TL	Moderate	Moderate	Frequent group testing and resolution to solve large and small-scale bugs, consulting module supervisor if a bug cannot be solved with the team's current knowledge.	TO
B1	Final product is not approved of by the University, or by the University's target market.	B R	Moderate	High	Weekly checks of the project with the requirements. Open line of communication with client to ensure all desires met. Ensure product is fun, attractive and not insulting through requirement checks.	OO PO
B2	Project is not finished and/or buggy at final deadline.	B O T TL	Low	High	Frequent testing and maintenance of the code to remove large scale bugs, additionally smaller, insignificant ones if time allows. Regular team meetings and monitoring of product development in relation to deadlines. 'Overtime' if deadlines arrive with some documentation not ready, to ensure meeting deadline requirements.	OO TO

Bibliography

[1] S. Naidamast, "Common sense software engineering - Part III: Risk analysis," JAXenter, 2 October 2015. [Online].
Available: https://jaxenter.com/common-sense-software-engineering-part-iii-risk-analysis-121131.html
[Accessed 1 November 2018].

[2] I. Sommerville, Software Engineering, Pearson, 2015.
Available online at:
https://course.kennysoft.kr/2017_winter/%5B21000555-001%5D%20%EC%86%8C%ED%94%84%ED%8A%B8%EC%9B%A8%EC%96%B4%EA%B3%B5%ED%95%99/textbook/Ian%20Sommerville-Software%20Engineering-Pearson%20(2015).pdf