Top Right Corner

Risk presentation description and justification

We have decided to base our risk table on a format described by Laurie Williams. We feel this decision is well justified as Williams is a respected researcher and teacher in the field of Agile software engineering, teaches a highly-rated professional agile software development course and has over 9 years of industrial experience[1].

The items in our table are ranked (and ordered) based on their severity and likelihood, with the most likely and severe risks at the top of the table[2]. The severity is determined by the amount of work required if the risk were to occur and the likelihood by the probability of the risk occurring. Sommerville recommends a similar system, but only sorts risks by their severity, without taking into account the likelihood[3], we felt that this ordering was inferior as a risk that is very severe but very unlikely should not be seen as more important than one that is slightly less serious but far more likely to occur. Separate columns for likelihood and severity are included[2] making our choice of rank clear. We opted for a categorial system for determining risk severity and likelihood as opposed to a numerical one (Williams' system allows for either). We felt, as Sommerville does[3], that making precise numerical assessments wouldn't really be possible as, for example, we can't know exactly how much work will be involved if a team member drops out as it depends partially on the time at which they do so. The categories we chose for likelihood are: frequent (over a 74% chance of occurring), probable (40%-74% chance of occurrence) and improbable (up to a 39% chance of occurrence). The categories chosen for severity are: catastrophic (over 6 hours worth of work), critical (between 1.5 and 6 hours of work), marginal (up to 1.5 hours of work).

The risk descriptions are specified in the CTC format described by David P. Gluch[4] and encouraged by Williams[2]. The template for each description is: Given that *condition*, then there is a concern that (possibly) *consequence*. Where the condition section specifies the conditions that exist that have made us concerned about the consequence, and the consequence specifies any potential losses that we may suffer due to the occurrence of the risk[4]. Separating the condition from the consequence gives a clear structure to the description, making it clear to the reader what may go wrong and why we feel that it may happen. This format made it easier for us to write up the risks ensuring that only relevant details were included and any irrelevant details omitted. Also when we talk about risk mitigation we may talk separately about how we may alleviate the concerns that we have due to the condition or reduce the effect of the consequences. As such the mitigation column of our table contains (as recommended by Williams[2] and Sommerville[3]):

- Strategies for avoiding the risk (dealing with the condition)
- Strategies for protecting ourselves from the consequence if the risk does occur, by minimizing the impact of the risk and having contingency plans in place

The only place where we chose to deviate from Williams' template was in the risk monitoring process, she recommends that we have a column that records the Rank last Week/weeks left[2]. We decided against this in this report as it is the initial specification of the risks and therefore no meaningful information can be included in this column (as we haven't been able to monitor the risks as of yet). Instead we shall follow the recommendations of Sommerville and simply update the table at regular intervals[3] (keeping older copies for reference).

Risk table

Rank	Risk	Likelihood	Severity	Mitigation
1	Given that the customer will issue a requirements change, then there is concern that (possibly) this change will require a major redesign of the system	Probable	Critical	As soon as the changes are issued we will come together as a group to estimate the impact of the changes and develop an action plan so that we know the steps to take in order to make the necessary changes on time
2	Given that we are all students mixing with a large number of other students of a daily basis, then there is concern that (possibly) team members catch minor illnesses during the project	Frequent	Marginal	We have emergency contacts for all group members so that communication should never be an issue. We will allow for some flexibility in our plan so that individuals can catch up on work they were meant to do. All documentation shall be kept on/uploaded to a shared folder on the Google drive and all code uploaded to GitHub so that it is always available to all of the other group members.
3	Given that this is the first time we have carried out such a project, then there is concern that (possibly) we may not fully understand the requirements	Improbable	Catastrophic	We will meet up with the customer to discuss any requirements that we feel we do not fully understand and to show them our implementations of certain requirements to ensure that we have met their expectations.
4	Given that this is the first time we have carried out such a project, then there is concern that (possibly) we end up generating incorrect requirements	Improbable	Catastrophic	The team needs to ensure that they meet up with the customer to ensure that the requirements we come up with matches their expectation
5	Given that we have to take on another team's project, then there is a concern that (possibly) we may pick a poor project	Improbable	Catastrophic	Ensure that we inspect other teams code and documentation fully and then discuss our findings before picking up their project at the start of phases 3 and 4
6	Given that we are unable to foresee any serious events that may occur in our team members lives, then there is concern that (possibly) a team member could drop out, go on a leave of absence or develop a major illness	Improbable	Catastrophic	We have emergency contacts for all group members so that communication should never be an issue. We have all agreed to let the other team members know ASAP if we are completely unable to complete any work for an extended period of time. All documentation shall be kept on/uploaded to a shared folder on the Google drive and all code uploaded to GitHub so that it is always available to all group members. As we

				are following an Agile methodology we shall not have to alter the plan too much, but make sure that the work that the team member was supposed to do gets reallocated.
7	Given that the customer can't be too involved with our project (as they have other commitments and are a part of many other SEPR projects), then there is a concern that (possibly) they may not like our final product	Improbable	Catastrophic	Ensure that we do not insult or offend the customer with the final system by ensuring there is no controversial or offensive content in the final game. Meet up with the customer whenever possible to discuss major design decisions.
8	Given that we are going to be using libraries that we may have never used before, then there is concern that (possibly) some of these libraries may be poorly coded and/or poorly documented	Improbable	Catastrophic	We shall meet up to research libraries before development and ensure that they are well documented and supported. We will also try to use libraries that team members have had previous (and positive) experience with.
9	Given that this is our first major software development project, then there is concern that (possibly) we may underestimate the complexity of certain tasks	Improbable	Catastrophic	When planning each Scrum we need to ensure that we plan to do as much work as possible so that it doesn't build up. We've allocated catch up time in the plan at the end of major sections of work.
10	Given that some of our team members may be less competent programmers than others, then there is concern that (possibly) the quality of the code they produce may affect the quality of our final system	Improbable	Critical	We shall have discussions on how to allocate roles such that members with weaker coding abilities are allocated only the programming tasks that they feel comfortable with.
11	Given that many of our team are planning to do a year in industry, then there is concern that (possibly) interviews may clash with group meetings or SEPR lectures and practical sessions	Probable	Marginal	We have agreed to catch up on any missed lectures ASAP and always upload a copy of the notes from any meetings we have to our shared folder on Google Drive so that team members can catch up on work that they would have completed during that meeting and can see what we discussed.
12	Given that we haven't used Git before, then there is concern that (possibly) we may forget to commit the latest version.	Probable	Marginal	At every Scrum meeting we shall ensure that the latest versions of all aspects of the system are available. If any group member notices code that should be on GitHub isn't they should contact the

				group chat to let the relevant person know.
13	Given that some team members choose to work on their laptop and there is a chance that it could be damaged or lost, then there is concern that (possibly) work may be lost	Improbable	Marginal	All group members should regularly backup their work to the Google Drive / GitHub so any data lost will not have a large impact on the project.
14	Given that we are using web based services to store some of our work (GitHub and Google Drive), then there is concern that (possibly) we may be unable to access it if the services are unavailable	Improbable	Marginal	Ensure that whenever possible we download the latest version and store a backup copy else where (especially near deadlines).
15	Given that the decision making system for the AI in the auction system could be very complex and hard to implement, then there is a concern that (possibly) it may be not work very well	Probable	Critical	We have alternative requirements in place: allow players to offer other players resources for a set price and to offer money for a given number of resources.

Bibliography

[1]"Laurie Williams: Software Engineering Research", *Collaboration.csc.ncsu.edu*, 2016. [Online]. Available: http://collaboration.csc.ncsu.edu/laurie/index.html. [Accessed: 05- Nov- 2016].

[2]L. Williams, *Risk Management* 2004, pp. 4-12. Available: http://agile.csc.ncsu.edu/SEMaterials/RiskManagement.pdf. [Accessed: 18-Oct- 2016].

[3]I. Sommerville, *Software engineering*, 10th ed. Harlow, England: Addison-Wesley, 2015, pp. 641-652.

[4]D. Gluch, *A Construct for Describing Software Development Risks*, 1994, pp. 13,14. Available: https://resources.sei.cmu.edu/asset_files/TechnicalReport/1994_005_001_16313.pdf. [Accessed: 05- Nov- 2016].