University of York

Department of Computer Science

SEPR - Assessment 3

Updated

Risk Assessment

Team Craig

Thomas Burroughs

Huw Christianson

Joseph Frankish

Isaac Lowe

Beatrix Vincze

Suleman Zaki

Risk Assessment

~~Removed~~ – Removed for Assessment 3

Red – Update/Addition from Assessment 3

In the course of a project, there will be a variety of aspects which are inevitable and will affect the process of making the project. Risk management is suggested to assess the potential risks at the beginning of the project and formulate certain corresponding mitigation measures. As a group, we brainstormed then listed the potential risks as suggested in [1]. Risks were also identified whilst eliciting requirements and were added in later. We analysed the risks we had listed and deleted ~~some that were unlikely to happen~~ risks will low likelihoods and impacts. Then we divided them into three categories, project risk, product risk, and business risk [2]. We chose these categories as most risks tend to either fall into one or more of said categories, allowing us to better prioritize certain risks depending on the effect they have. These categories are defined as,

- Project risk: Affect project schedule or resources.
- Product risk: Affect the quality or the performance of the software product.
- Business risk: Affect the organization developing or procuring the software.

We created a risk register table [1] with the recommended columns; risks and descriptions, responses (mitigation), and categories. After further research into risk registers we also added columns for index, likelihood [3], consequence (impact) [3], and risk owner [4]. Below are explanations of the columns in our risk register table.

- Index: Number the risks. This will be useful when referencing risks in other documents.
- Risk: The issue the team may face during the process of the project.
- Description: Details of the risk.
- Category: Places the risk into one of the three discussed categories, project, product, business.
- Impact: Classifies the impact the risk would have on the project if it were to occur. It is given one of four classifications (Figure II).
- Likelihood: Classifies the likelihood of the risk occurring. It is given one of three classifications (Figure I).
- Mitigation: Predictive measures to reduce the likelihood of the risk occurring and solutions in case these measures fail.
- Risk owner: Risks are allocated to particular members of the team who will monitor them during the project.

Members of the team were assigned to risks as risk owners if their role related to the risk. The member of the team that has that role has the responsibility of monitoring that risk. ~~At the start of each meeting each member of the team will report on the status of their risk if it has changed. See the method and planning document which outlines which team members have which role.~~ Every week the team will meet in addition to the weekly scrum meeting, where the risk register will be reviewed and potentially updated. This meeting will be organised and led by the Risk Manager. The team will be able to voice concerns over new potential risks or previous risks which have transpired during the current sprint and what mitigation plans need to be put in place. Furthermore, during this meeting the group will deliberate over whether the impact and mitigation plans of transpired risks were suitable or occurred as expected. Updates to the risk register, including changes to likelihood and severity, can then be made according to the discussion. New risks will be added to the register along with a suitable mitigation plan. Monitoring risks in this way should hopefully help to reduce the impact of unanticipated risks and prevent delays due to out of date mitigation plans which are no longer suitable for this part of the assessment.

For the likelihood of risk occurrence, we use three levels to classify, high, moderate, and low, which are denoted by red, orange, and yellow [5]. For an estimate of the degree of impact, we use four grades, critical, high, moderate, and low, expressed in black, red, orange, and yellow [5]. An impact of “Critical” means that the risk may lead to the failure of the entire project, so regardless of its likelihood we need to give it extra attention to make sure we avoid it. Risks, within their category, are in descending order of impact and if more than one risks have the same severity, then they are listed in the descending order of likelihood.

Risk Register

Category	Index	Risk	Description	Likelihood	Impact	Mitigation	Risk owner
PROJECT	2	Lost work	Work is not saved or somehow lost.	Low	Critical	Use GitHub for the whole software system and code management. GitHub has a huge user base and is very well supported so it will be reliable.	Head Developer
	5	Miss a deadline	Unable to submit an assessment in time.	Low	Critical	Set a reminder for deadlines. Reserve enough time in our project plan to complete all tasks with time to spare before any deadlines.	Team leader Secretary
	9	Bug in our software system with an unknown origin	This is more likely to happen later in the development period.	High	High	Following good practices in the design phase of the project will reduce the likelihood of this occurring. We should test the software frequently during development to find the bug as early as possible.	Test Leader
	7	Problems with software/tools	We could encounter bugs and issues with the software.	Moderate	High	Make sure we initially choose well supported and popular tools which are more likely to receive regular updates. Prepare a backup tool incase we still run into issues.	Head Developer
	6	We stray from the planned schedule	Some tasks could take longer than expected, or a disagreements could delay a task causing us to deviate from the schedule.	Moderate	High	Organize regular group meetings and assign each person some task each time. Communicate each other in a timely manner when problems arise. Redistribute tasks between team members tasks if one of them is taking longer than expected.	Manager Secretary

	4	Uneven workload distribution	Some team members feel they do more work than others.	Moderate	Moderate	We can make a summary of the amount of work done by each member. Also, we can arrange new assignments at each group meeting to ensure there is an even distribution of work.	Team leader
	1	Absent team members	Someone misses a group meeting or practical session.	Moderate	Moderate	This member should communicate with other team members in advance. A present member of the team should update the absent member on what they missed.	Team leader Secretary
	8	Struggle with the task	A team member finds a task hard to complete or is spending too much time working on it.	Moderate	Moderate	Before starting production, assign tasks to members according to their strengths. If someone hasn't used a tool before they can learn and practice in advance; team members should help each other if they can.	Team Leader
	3	Disagreement among team members	Team members have different opinions about project decisions.	Moderate	L ow	This could happen at any point in the development process. Disagreements can be voted on if necessary. Most of the disagreements will be solved before developing to reduce impact	Manager Team leader
	14	Offensive or unsuitable content	The content of the game (i.e. the story) could cause offense or unsuitable for open days and UCAS days.	Low	Critical	Offensive content should be avoided at all costs. If anyone in development team, stakeholders, or users finds any of the contents of the game offensive we will remove it immediately. The game is zombie themed which by nature is likely to include a lot of gore which could be unsuitable for open days and UCAS days. We will be using an 8-bit art style which limits the amount of gore possible and reduces realism keeping it suitable for our audience. Nevertheless, we should keep the stakeholders up to date with prototypes and concept art to verify it is still suitable.	Graphic Designer
P R O	11	Unachievable requirements	During the planning and	Moderate	High	We need to discuss in depth all requirements within the team and with stakeholders to avoid this. If we do run into this issue we should contact	Client Interface

DUCT BUSINESS			developing process, we may find some requirements are too difficult to complete.			stakeholders in a timely manner and come up with a solution or alternative together. We deleted or altered some of the requirements due to time limitations.	
	12	Requirements changed t.	The stakeholder's requirements for the project may change during the developing process. These changes can add extra unanticipated work to the project	Moderate	Moderate	The process of production should be reported to the stakeholders on a regular basis. If a requirement changes we should meet to discuss the implications of this and amend our project plan to account for any additional work. Using an agile method will allow us to quickly respond to requirement changes.	Client Interface
	13	Players do not understand how to play	The player may be confused by the controls or the mechanics of the game.	Low	Moderate	The game can introduce basic controls and mechanics into a tutorial stage at the beginning of the game. Include a controls section in the menu in case the player forgets how to play. We will invite people who are unfamiliar with the game to test it.	Head Developer
PROJECT PRODUCT	15	Code misunderstanding	People can't easily understand the code of the new project, which leads to spending much time read and understand the program.	Moderate	High	Before choosing the project, it is necessary to read some of the code to check if people can understand it.	Head Developer
	17	Critical errors in the new project	The new project contains critical errors, thus team members have to take a lot of time check-up the code and deal with the problems before further progress.	Low	High	When selecting the new project, we should run it and make sure it works with little error.	Head Developer
	16	Issues with the new system	The new project is using a different environment which team members are not	Moderate	Moderate	The team should be cautious to choose the new project, choose a project that use a system we are familiar with.	Head Developer

			familiar with, and people have to waste time to learn the new system.				
	18	Changes to new project architecture would cause errors	Changing class structure etc could cause errors in the project.	Moderate	Moderate	When changing anything in the architecture dependencies must be checked and discussed with group.	Head Developer

References

[1] S. Barati, S. Mohammadi,¨Enhancing Risk Management with an Efficient Risk Identification Approach¨, Proceedings of the 2008 IEEE ICMIT, IEEE, 2008, pp.1181-1186.

[2] Ian Sommerville, Software Engineering Ninth Edition, Boston, Pearson pp.595-602. Available: https://edisciplinas.usp.br/pluginfile.php/2150022/mod_resource/content/1/1429431793.203Software%20Engineering%20by%20Somerville.pdf

[3] Barry Boehm, Software Risk Management, European Software Engineering Conference, 1989, https://link.springer.com/content/pdf/10.1007%2F3-540-51635-2_29.pdf

[4] Vivian Kloosterman, "12 Key Elements of a Project Risk Register Template" [2014], *www.continuingprofessionaldevelopment.org* [Online]. Available: https://continuingprofessionaldevelopment.org/key-elements-project-risk-register-template/ [Accessed: 3- Nov- 2018].

[5]

	Likelihood	Impact
Low	<30 % The possibility of occurrence is low so that we do not have to worry about it.	The effect is tiny, and the problem can be solved easily.
Moderate	30% - 70% The possibility of occurrence is moderate, the prevention and mitigation measurements should be prepared in advance.	The consequences caused by the risk cannot be ignored, it won't lead to excessive adverse effects if it is dealt with quickly.
High	> 70% The possibility of occurrence is high, mitigation should be taken immediately when it happens.	If this occurs, it may greatly affect the progress of the project.
Critical	There is no risk that will 100% happen.	It may lead to the failure of the entire project.