Module	SEPR
Year	2019/20
Assessment	1
Team	MiKroysoft
Members	Daniel Crooks, James Rand, Irene Sarigu, Alfie Jennings, Charlotte Clark, Jasper Law
Deliverable	Risk assessment and mitigation
Website	https://mikroysoft.github.io/

Section 5a

We held a group meeting and brainstormed the possible risks that threatened our project based on our own knowledge and experience. We brainstormed because it's beneficial for quick construction of a risk register, as one idea can encourage the identification of another related risk [1]. Group meetings are often guided by a leader, which may prevent others from voicing their opinion [1]; we felt it was important that everyone contributed and was aware of the possible risks and so each team member was encouraged to name at least one potential risk. We then considered worst case scenarios resulting from these risks, from this we came up with our own management plan to avoid and mitigate the risks. Lastly, we judged the probability of the risk happening and how severe the impact would be. Likelihood and severity were both divided into 5 categories, their descriptions are shown in figures Risk.1 and Risk.2. To decide a severity score, the scrum master suggested each team member held a score on their fingers between 1 and 5, which were revealed at the same time (removing bias), and a mean was calculated for the final result. In cases of a decimal number being the result, the scrum master made the final decision. From the brainstorming session the risk description, likelihood of occurrence, severity of impact and management became the key parts of our risk register. Additionally, each risk was given an ID, categorized into types and given an owner, who is responsible for carrying out the management of the risk. Owners were assigned to risks based on their role assigned in the team, as they will be most experienced to do so. The four types of risks were: Project (for example management change), Product (e.g. tool bugs), Business (e.g. budget), Technological (e.g. hardware capability)

Below **Figure Risk.1** shows the definition of each likelihood scoring and its corresponding probability of occurring.

Figure Risk.1

Likelihood	Probability of happening (1)	Definition
Very High	80% - 100%	Very likely that the risk will occur, the risk is expected to occur during the project period.
High	60% - 80%	It is likely that the risk will occur during the project period, the likelihood of the risk occurring is probable.
Medium	40% - 60%	The risk may occur occasionally and might occur during the project period.
Low	20% - 40%	The probability of the risk occurring is remote and is an unusual occurrence, it is unlikely that the risk will occur during the project period.
Very Low	0%-20%	Very unlikely that the risk will occur during the project period, it is predicted that the risk will not occur during this period of time.

Below Figure **Risk.2** shows the meaning behind each severity score and its definition.

Figure Risk.2

Severity rating	Descriptor (1)	Meaning
1	Negligible	Insignificant impact, there will be little disruption to the project development and we will still be able to provide a product on time that meets the user requirements.
2	Minimal	Minor impact, so disruption to the project development but is easily managed.
3	Moderate	Slight impact, modification will be needed for the project development for a desired outcome.
4	Serious	Major impact, the product development is significantly affected and will require high levels of monitoring and abnormal operations to ensure the product is delivered on time at a high standard.
5	Critical	Major impact, project development is completely disordered and there is a high chance the product will not be delivered on time or at a high standard. High levels of monitoring will be required and development may operate abnormally.

Section 5b

Risk ID	Type	Description of Risk	Likeli-hood	Seve-rity	Management	Owner
RA_DIFF	Project	Management tool is too difficult to understand and we are unable to use it.	High	2	**Avoid:** Familiarised ourselves with the management tool before deciding to use it. **Mitigation**: Read documentation to guide how we use the tool or seek another management tool if reading the documentation does not work.	Irene
RA_NEED	Project	Management tool doesn't meet our needs.	Low	1	**Avoid:** Looked at what the management tool had to offer before deciding to use it. **Mitigation:** Read the documentation to see if there are any other features that can be used or seek another management tool that meets our needs.	Irene
RA_LIB	Product	Graphic libraries for the project are too complex to understand and use.	High	4	**Mitigation:** Research into the graphic libraries we use before relying on them.	Daniel
RA_ABSENT	Project	A team member is absent and cannot present their work that other members may be dependent on.	Medium	1	**Avoid:** Comment code and share work on GitHub. **Mitigation:** Frequent meetings will minimise the severity of single absences. Absentee must communicate with present team members to inform others of their absence and catch-up.	Daniel
RA_REQ_CHANGE	Project	Requirements change at the last minute.	High	3	**Mitigation:** have continuous communication with stakeholders and frequent meetings. As well as referring to the requirements documentation regularly through the project.	Charlotte

ID	Type	Description	Likelihood	Severity	Mitigation	Owner
RA_TOOL_CONFUSION	Project	Using too many management and communication tools causing confusion and poor communication. Meaning team members don't receive important information.	Medium	2	**Mitigation:** Pre-assign tasks and have a stand-up in each meeting. Also agree on a preferred method of communication, with which all team members are familiar.	Irene
RA_TEAM	Project	Team dynamics don't work as expected leading to conflict and poor communication.	Medium	3	**Avoid:** Underwent a team building session in SEPR Lab 1. **Mitigation:** Remain professional and communicate about the project. If a dispute distracts from the project, a group discussion should be held to resolve the issue.	Irene
RA_EFFICIENT	Project	Team does not work efficiently under pressure.	Medium	4	**Mitigation:** Assign internal deadlines to manage workload. Try and avoid absences, as well as give extra time if needed	Irene
RA_KNOWLEDGE	Project	Team lacks technical knowledge to work on a given task.	Low	4	**Mitigation:** Attend all labs and lectures available and follow up with independent research.	Daniel
RA_HARDWARE	Technological	Experience hardware problems whilst working on a task, resulting in work being lost.	Medium	3	**Mitigation:** Use GitHub to allow access to all points in development. Ask for advice to fix hardware problems and use university computers so work can be accessed from server.	Daniel
RA_REQ_USER	Project/ Product	The team misunderstands the user requirements, or the requirements are not detailed enough.	Low	5	**Mitigation:** Hold regular meetings with the stakeholders and regularly refer to the requirements. Code may need to be refactored to meet requirements.	Charlotte
RA_MAINTAIN	Product	We write unmaintainable code and cannot develop the code.	High	4	**Mitigation:** Follow coding conventions and comment efficiently.	Alfie

ID	Type	Description	Likelihood	Severity	Mitigation/Avoidance	Owner
RA_BASIC_FT	Product	Focus too much on extra features, without completing important basic features.	High	5	**Avoid:** Determine the basic features needed before programming. **Mitigation:** Remodel code so that basic features are included, team members may need to put in extra time to ensure this.	Alfie
RA_INT_DEADLINE	Project	Not meeting internal deadlines set by the team.	High	3	**Mitigation:** Team members should work independently to meet deadlines agreed upon.	Irene
RA_COM_USER	Project	Poor communication between the team and user.	Medium	3	**Mitigation:** Communicate frequently with the user and refer regularly to the user requirements throughout project development.	Charlotte
RA_CRASH	Product	Game crashes whilst user is playing, progress is lost and user cannot complete the game.	Low	5	**Mitigation:** Test to find bugs and fix accordingly.	Jasper
RA_ARCH	Technological	The designed architecture cannot support what we want to implement.	Low	3	**Mitigation:** design architecture for new requirements and adapt if needed.	James
RA_DIFF	Product	Inappropriate difficulty.	High	1	Adjust the difficult game elements as required.	Jasper
RA_GOAL-MET	Product	Game goal (or methods to reach goal) are unclear to the user.	High	3	Highlight the relevant tools or game elements. Refactor tutorial mode to include the required information.	Jasper
RA_UI	Product	The UI is too complex, lacks features, or interferes with the gameplay.	High	4	Adapt the UI as required.	Jasper
RA_STORY	Product	Storyline is unclear or interferes with gameplay.	Medium	2	Adapt as required.	Jasper

Our team came up with more potential risks, however these were all predicted to have a very low likelihood of occurring or lowest severity. The very unlikely risks are available on our website.

Bibliography for risk assessment:

[1] Murray, S., Grantham, K. and Damle, S. (2011). Development of a Generic Risk Matrix to Manage Project Risks. *Journal of Industrial and Systems Engineering*, [online] 5(1), pp.35-51. Available at: http://www.jise.ir/article_4040_b99470f366ec6a2ff0658df670d86ca0.pdf [Accessed 11 Nov. 2019].