Risk Assessment and Mitigation

Group 8 - Delta Ducks

Yumis Zyutyu

Zac Challis

Viktor Atta-Darkua

Dandi Harmanto

Harsh Mohan

Danny Burrows

This is a risk register containing information on : different risks ordered by their Risk ID, their types, a description of the risks, likelihood of the risks occurring, Impact/ severity of the risk if it occurs and possible mitigation measures.

The Risk ID is used to reference certain risks in other parts of our project. For example, in our project's requirement document different requirements may have associated risks in this risk document, in this case, the risk ID is used to reference risks in this document. This is much faster than writing the risk information next to each requirement, especially since some risks may be associated with more than one requirement.

The risk description gives a brief overview of the risk, possibly indicating its consequences. This is required to understand what the risk is and why we don't want it happening.

The likelihood is on a scale of 1 to 10, with 1 being not likely for the risk to occur in the current situation and 10 meaning that it is all but certain that this risk will occur. Impact / Severity is also on a scale of 1 to 10, with 1 meaning that the risk is not very threatening even if it occurs and 10 meaning that the risk is very threatening and actions must be taken to mitigate it at all cost. The combination of these 2 scales allow us to estimate the danger of different risks, allowing us to know how much attention we should place on them and how thoroughly to enforce different levels of mitigation for different levels of danger. By paying more attention to more dangerous risks, we will be more likely to have a more favourable outcome.

The mitigation column records how to reduce or nullify the chance of the risk occurring and / or its danger. By following these guidelines our project can hopefully go on as problem free as possible.

The reason I have used such a format is because it is simple, easy to understand, and delivers all you need to know about different risks without adding more variables / columns. There was also a chance to add an 'Owner' column, a group member responsible for updating the likelihood and severity of different risks as the project continued but did not do so for 2 reasons. Firstly, we did not want to make the register/ graph overly complicated. Secondly, different group members may have different views on what the 2 scales of 1 to 10 represent. Each person has their own values, knowledge and thoughts, while one person may think a value of 7 means high likely to occur, another may interpret it as a mediocre chance of occurrence. To avoid such bias, having one person overlook the likelihood and severity scales of all risks seems much better.

Risk ID	Type	Description	Likelihood (/10)	Severity (/10)	Mitigation
R1	Product / Project	Specification delays, forgetting to ask certain precise requirements for the game until we come to developing them.	7	3	Continuous meetings with the customer to see what they think about the current project and what their current requirements are. Record answers and prepare questions for the meeting. Do this frequently in case the customer changes their mind on requirements and to mitigate the chance of not knowing how upcoming implementation aspects should be handled.
R2	Project / Business	Getting out of touch with the project due to the break between autumn and spring term.	5	6	Keep in touch with project over break period / just before the break ends, so even if we don't work during the break,we can get back into working on the project quickly.
R3	Technology	Different operating systems can respond to programming languages differently.	3	7	Have a few members of the group focus on programming. Discuss in group and make sure all relevant parties can use the game engine/programming language successfully.
R4	Product	The enemy boat's respective college may be hard to distinguish from each other	4	7	If different colleges are distinguished by colour, try to use distinct colours to help distinguish ship's affiliations. If too many colours are used and similar colours begin appearing, maybe include patterns or letters to help distinguish affiliations.
R5	Product	Game lasts too long or too short, leaving users unsatisfied.	5	8	Make a timer so you lose if the game takes too long, e.g. 10 minutes. Playtest the game to make sure that the objective takes a moderate amount of time, change factors such as boat speed and enemy health otherwise
R6	Product	User doesn't understand how to play	2	9	Make a tutorial or give options for pop-ups that

		the game, either not understanding the controls or game objectives / details..			explain the controls and game objectives/details. Game controls and details should be kept simple and easy to understand
R7	Product	Win / Loss condition completion doesn't end the game.	3	9	Playtest the game repeatedly to make sure bugs (like not dying at 0 health or winning when enemy collages are defeated) do not exist
R8	Product	Variables such a ship's or college's health are not updated in real time on the screen when they are affected.	5	8	Implement code for health to be updated in real time and playtest it.
R9	Product	Game too easy, little risk of player death makes the game boring for the user.	6	7	Make a suitable AI for the enemy colleges to shoot at the user. This shooting mechanism should be hard but possible to dodge and deal enough damage to make the game sufficiently challenging. Playtest to ensure this.
R10	Product	User does not feel any accomplishment in interacting with obstacles in the game, Possibly finding it boring.	4	6	Create a UI which will keep the points the player gains from sailing and gold the player gains from fighting updated in real time.
R11	Product	Player can attack a collage at an angle/ distance at which the college AI cannot shoot back, making the game boring.	6	7	Change college AI to shoot at player if attacked, even if out of the college's normal firing range / shooting angle. Playtest to ensure no workarounds.
R13	Product	Player disabilities such as blindness / dyslexia make them difficult to understand and/or play the game.	3	9	Try to ensure the game is understandable in black and white. Add audio options for those with visual impairment. Make sure the game doesn't need audio options to be understood. Try to keep letters, symbols and textual information in a sufficiently distinct format.
R14	Business	Ethics violation due to storage of player data without consent, possibly leading to legal	4	10	Do not store any user / player data. Do not store any data that carries over between games.

		problems			
R15	Product	Updated version of the code fails to meet a previously met requirement. This is possibly not noticed.	6	9	Keep a clear set of tests pertaining to the game's requirements, every time the game's code / version is updated, put the game through all the tests to make sure that it is working.
R16	Product	Values such as plunder, points, etc… are not updated as they should be when the right actions such as defeating a college occur	4	8	Add tests to check that all such variables increase by the required amount expected when doing such actions.
R17	Business	User data lost over external servers, possibly due to hacking. Ethics violation due to private data loss with possible legal problems.	2	10	Do not connect the game to the internet or any other external servers / computers.
R18	Product	Difficult to distinguish textual information while busy playing the game	5	6	All text is made large enough, clear and readable in bright distinct colours.
R19	Product	Certain game settings / information is overly difficult to understand due to overly complicated jargon.	3	7	Textual information conveyed concisely in simple and easy to understand words.
R20	Project	Group members not meeting up on time leading to project delays.	6	8	Keep a clear schedule for work to be done. Make sure all sections are shadowed in case certain group members are temporarily missing.
R21	Technology	Our commonly used coding software is incompatible with the software required for making the game	5	6	Change the software we use. Use the same combination of software throughout the group. Keep your code intact.
R22	Product	The game may not be suitable for a demonstration environment and such may turn away prospective students.	5	6	Frequent customer meetings to ensure we are on track with meeting our requirements and ensuring the game is perfect for its intended use case.