5. Risk Assessment and Mitigation

Team 5 | Team Pending

Thom Robinson
Iris Scoffin
Ayman Elsayed
Annice Headman
Izaac Threlfall
Yihong Zhao

A. Introduction

Introduction

- The risk management process can easily be broken down into 4 simple stages. The first step is to identify all possible risks and categorising these risks with 4 different headings (technology, people, requirements, estimation)[1]
- The majority of these risks will be based on and associated with our requirements, with each requirement being paired with a risk in the requirements table.
- The risks of people will be associated with the commitment and involvement of people within the team and the people involved with the project.
- Technology risks relate to issues with software and hardware and will be based on the justifications made about our chosen architecture and the technological requirements.
- Estimation risks will deal with the deadlines not being reached and how these situations will be handled. The risks will be identified as a group using our requirements, our understanding of the architecture that we have chosen, and the overall plan for the project (the lifecycle model and work distribution) as a basis for risk identification and judgements about each risk.

Justification

- Each risk will be given a likelihood and severity (on a scale of low/medium/high) based on how likely the risk is to occur and how bad the effect of each risk will be. These variables will state how each risk is managed. This exercise will depend heavily on the personal judgements of the team members associated with the risks.
- The severity and likelihood of risk are also highlighted with a colour to visually represent these values at a glance.
- The management of risks will involve coming up with plans to avoid and mitigate each risk and also have a contingency plan in the event that avoidance and mitigation are not enough to sustain the development of the project. This will require additional planning and heavy discussion between team members in order to come up with the most optimal solution to each problem and avoid any major disruption within the overall project plan.
- Once a plan has been made for each risk, the people associated with that risk will be assigned as risk owners who will regularly make changes to the severity and likelihood of the risks depending on their judgement and involvement with the risk. The changes to the risk analysis will be reported during weekly meetings by the risk owners so that team members are inclined to not neglect monitoring risks.

B. Risks

People					
ID	Description	Likelihood	Severity	Mitigation	Owner
rP1	Website designer becomes unavailable	M	M	Attempt to inform the group and have someone else with website design skills to cover up the work	Iris
rP2	One or more programmers are unavailable	M	H	Attempt to warn the group and do as much work as possible in advance, if not temporarily assign second person to work on their code	Thom
rP3	Whole team becomes occupied/busy (e.g. due to exams)	H	M	Reduce workload for the week and redistribute work to next week	N/A
rP4	One or more people can't attend a meeting	H	L	Report the completed work in the group chat before the next meeting	Those that miss the session
rP5	Low morale/burnout	M	H	Distribute work to accommodate team members who are burnt out or reduce workload for the week	N/A
rP6	Conflicting ideas between group members	M	M	Have group members vote on which idea they prefer so it remains fair	N/A

Technology					
ID	Description	Likelihood	Severity	Mitigation	Owner
rT1	Game is overly processor dependant	L	H	Research into optimisation methods that utilise other aspects e.g. the gpu	Thom
rT2	UI/Game library doesn't support customer hardware	M	H	Check before using that software is compatible and research alternatives as backups.	Iris

rT3	Game window not mapped to monitor correctly	L	H	Know the specifications of the monitor prior to the event, or if not possible, have the game window in a fixed size	Annice
rT4	Bugs found within the game library	L	M	Work around bugs as best as possible, if they still prevail consider changing the library if necessary	Ayman
rT5	Bugs found within programs external to the game	L	M	Work around the bugs as best as possible, if they still prevail consider changing tools	Izaac

Requirements

ID	Description	Likelihood	Severity	Mitigation	Owner
rR1	Game isn’t suitable for target audience	H	M	Remove content from the game until suitable for targeted demographic, gain second opinion to help get perspective	Annice
rR2	Input that gives player unfair advantage is found	M	L	Identify the code giving players access to feature and revise/remove it	Izaac
rR4	Game does not accommodate for colour blind players	H	M	Change the colours selected for assets and sprites, ask for opinion from colour blind test users	Ayman
rR5	User finds game too difficult	M	H	Remove unnecessary content, reduce health of enemies and/or increase player health	Thom
rR6	Game takes longer than 10 minutes/less than 5 minutes	M	H	Adjust character speed (increase/decrease travel time between points on the map)	Iris
rR7	Requirements aren’t fulfilled	M	H	Get more team members to assist with the coding, else ensure that as many requirements are covered as possible	Thom

Estimation

ID	Description	Likelihood	Severity	Mitigation	Owner

rE1	Sprites aren’t completed in time	L	L	Use placeholder sprites	Ayman
rE2	Team coding ships and functions/procedures is underestimated	M	M	Ignore non essential functions, else implement a simple/abstracted replacement for it	Thom
rE3	Bugs and error handling takes longer than expected	M	H	Prioritise solving the biggest issues first and assign more people to help if available	Annice
rE4	Covid prevents team from co-operating in person	L	L	Move to an online format	N/A

References

[1] Sommerville, I., 2015. *Software engineering*. Boston: Pearson, pp.644 - 652.