Risk assessment and mitigation

Team 17: Scone Zone
Micheal Calaciuti
Sabrina Djebbar
Yaseen Khan
Tinetariro Muzunzandare
Josh Saunders
Zac Spooner

Risks and format

Risk no: A unique ID to refer to each risk
Risk name: A brief description of what the risk is
Risk Likelihood: How likely the risk is to occur. This uses the scale [“Low”, “Moderate”, “High”].
Risk Severity: How severe the effect of the risk occurring would be. This uses the scale [“Low”, “Moderate”, “Severe”].
Risk Impact: What effect the risk would have on the project should it occur
Risk Mitigation: What action we plan to take to reduce the likelihood and severity of a risk happening. E.g. backing up work frequently and often evaluating the levels of risk in the project
Risk Owner: Who is responsible for measuring and overseeing this risk and its current likelihood and severity.

Risk table

Risk no	***Risk name***	***Risk Likelihood***	***Risk Severity***	***Risk Impact***	***Risk Mitigation***	***Risk Owner***
1	Picked an unsuitable game engine	low	severe	• We will be unable to create a game as easily and efficiently as possible. This will lead to a poor game experience for the user and a bad quality game.	• Create a basic prototype early on that uses assets and implements basic movement • Make sure to always “look ahead” before implementation to see if the requirements are feasible in the game engine	Programming team
2	Group member can no longer continue working on the project	low	moderate	• The team may be left behind schedule as the remaining members must take on the added workload which may result in a late or incomplete project	• Peer reviewing project work, clear documentation. • Team members can communicate this on the discord as well as their responsibilities as soon as possible so the rest of the group can evenly and quickly split up the member leaving’s work.	Anyone
3	Group member is unable to do work due to illness	high	moderate	• The team may be left behind schedule if the ill member has not been able to complete any outstanding work. • It may be difficult to complete work as a team, if the ill member is not there to communicate their ideas for the development of the project • Or if the team is relying on the incomplete work of the ill member to move on with the project	• Make sure all work is clearly documented and completed on time. • Have regular meetings, reviewing what work has been done so that the group won't have to spend more time familiarising themselves with an ill members work • In the event of illness, make sure ill group members are kept up to date with progress made. Any outstanding tasks from the ill member should be made known and split between the group to avoid falling behind schedule.	Anyone
4	Accidental loss of project data	low	severe	• We may have to redo a lot of work if we have not backed it up regularly. • This would result in falling behind schedule	• Back up often using git, and do as much work as possible on google drive which saves automatically • Make sure each member knows how to work with git	Anyone

5	Poor design choice that has already been implemented	moderate	low	• We will be left behind schedule as we would have to rewrite all the code to implement a better design • The player will be left with a bad game experience	• Justify all design decisions fully during the design stage before implementing them, to check their appropriateness • Have periodic playtesting and checking of the various functions/ mechanics of the game	Anyone
6	Design choice that breaks one of the requirements	low	severe	• Left behind schedule as we will have to pick a new design choice that does not break our requirements before implementing it	• Decide design choices based on a set of specific goals that already have the requirements in mind • Check back to the fundamental requirements before deciding on design choices	Anyone
7	Misunderstandings about how the program connects together	moderate	moderate	• We will be unable to complete the code as it will be difficult to write code if we are unsure on what it is we're implementing.	• Document your code. • Have programmers swap the sections of code that they are working on to allow for implicit peer review whilst they work on the new section and give each individual a better understanding of the whole system • Have the programming head delegate tasks to individuals stating the input/output of what they need to make and what the task needs to do • Have someone who knows the bigger picture be explained by everyone what each section of code does.	Programming team
8	Unsuitable difficulty or AI intelligence	moderate	moderate	• This will make the game unplayable, which would lead to a poor gameplay experience for the users	• Have thorough testing with new users to get an understanding of how different "skill-levelled' people fare against it • Keep testing the AI in any new environments made or with any new features/mechanics added	Programming team