Assessment 1: Greenfield Development

Risk Assessment & Mitigation

Cohort 3 Group 9

Oliver Barden

Connor Burns

Sam Goff

Milap Keshwala

Lewis Mitchell

Jeevan Singh Kang

Harry Thomas

Titas Vaidila

Our team is using a systematic table for displaying risks. Each risk has its own unique ID to ensure the table is easy to navigate and so we can clearly and simply refer to different risks during the rest of the project. The type field helps with risk identification, ideas for risks come more easily when provided with a broad category to start with. A likelihood is defined for each risk, alongside a severity rating, each denoted by a letter score (one of L, M, H for low, medium or high). A combination of these can be used to decide which risks to study in more detail (more likely risks and risks that will pose more of a problem if they occur may be worth looking at). The mitigation field outlines the actions that will be taken in the event that the risk comes to pass, this should reduce the impact of the problem as clear instructions are already written to be followed. Finally each risk is assigned an owner, this person will be responsible for keeping track of this risk factor and carrying out the mitigation steps if required.

This risk register outlines the majority of the risk management process our team is undertaking. Risks are identified and the situation in which they might occur is described in detail. Then they are analysed to determine their likelihood and severity, and a method for minimising the impact of the risk is set out. The final aspect of the risk management process is to actually use this table during the rest of the project, monitoring risks and using the mitigation instructions to help keep the development of the game running smoothly.

ID	Type	Description	Likeli hood	Severity	Mitigation	Owner
R1	Product	The Java libraries used might prove to be more limited than expected by the team, increasing the complexity of the project to be able to meet all of the requirements	M	M	Sacrifice some requirements, or do some more research, practise or put more time aside to allow for the complexity to be achieved	Jeevan Sign Kang
R2	Project	A team member is unavailable due to illness	H	L	Each task should be assigned to at least one other team member to pick up the slack, if this is not the case other team members can be assigned. Also, code should be well documented in case someone else needs to start working on it	Oliver Barden
R3	Product	A serious bug is found in the code that requires huge restructuring / rewriting of code	L	H	Get together as a team to discuss the bug, then take appropriate actions to fix it	Lewis Mitchell
R4	Project	Art that will fit with the game and has acceptable copyright cannot be found online	L	M	Use generative AI (may still have copyright issues though), or get someone in the team to make some artwork themselves	Connor Burns
R5	Project	Differing coding styles between team members	M	L	Follow Google code styling - style guide.	Harry Thomas

R6	Product	Finished game prototype doesn't meet the customer's expectations	L	H	Meet with customer regularly and share progress to ensure the team is on track	Titas Vaidila
R7	Technology	Game UI is too slow on clients chosen hardware	M	H	Early access to clients hardware to test performance, if it's too slow or buggy take appropriate action to improve it	Sam Goff
R8	Security	Data loss/corruption such as lost files	L	H	Use version control, applying regular commits and backup saves to the google share drive	Sam Goff
R9	Product	Controls feel unresponsive or lack intuitiveness	M	M	Carry out early testing of the controls to get constructive feedback, or offer custom key bindings	Milap Keshwala
R10	Product	Audio or visual assets cause performance issues	M	M	Compress assets and carry out frequent checks whilst monitoring performance	Lewis Mitchell
R11	Project	Miscommunication between team and client relating to requirements	L	H	Maintain a clear documentation of requirements and always check before making any changes	Oliver Barden
R12	Product	Chosen game framework causes development issues, i.e missing features or compatibility problems	M	H	Carry out thorough research before deciding on the final library and test early versions. If issues arise attempt to work around them or find alternatives without changing framework	Jeevan Sign Kang