

Team17 - Gnocchi Games

**Henry Overton
Yousif Habib
Ben Dunbar
Lucy Newton
Adam Blanchet
Thomas Heenan**

Risk Assessment and Mitigation

Risk Assessment and Mitigation

Risk Format

We used a risk register to display risks to the project. We started by identifying potential risks to the project and inserting them into the register. Each risk has an ID, type, description, likelihood, severity and mitigation. Each risk was assigned one of three likelihoods: low, moderate and high. These were based on how likely the team thought they would occur. Similarly, each risk was assigned a severity. These were again low, moderate and high. This was based on the impact the risk would cause to the project were it to occur. Finally, we decided how to mitigate each risk. Any actions that we could take to reduce the impact a risk would have on the project were recorded.

We chose a risk register because we decided it was the most effective way that we could identify risks and ensure we had sufficient mitigation in place. The use of a colour code system allowed us to easily identify the risks that would cause the most harm and ensure we monitored them regularly.

We decided against having a more detailed rating system for likelihood and severity because our current system adequately displays the information in an efficient way and more detail was unnecessary. We opted against including an owner column in our register as once we began recording our risks we realised that it was easier to regularly re-assess risks as a group during team meetings where we discussed any changes needed to the likelihood and severity of all known risks. This ensured all team members were aware of all potential risks to the project at any given time allowing for spontaneous and robust risk mitigation. To allow for clear communication concerning risk mitigation a separate channel in our Discord server was created where group members discussed either new potential risks or the change of likelihood/severity factor that needed addressing at the next team meeting.

Risk Register

ID	Type	Description	Likelihood	Severity	Mitigation
R1	Project	One of our assigned developers becomes unable to work on the implementation	L	M	Ensure there are backup developers who are able to assume their role, also contributes towards increasing the 'bus factor'
R2	Project	A team member is not contributing	L	M	Engage proactively with said team member. Discuss how they can contribute to the team better. If the problem continues, discuss with module leaders
R3	Project	Unsure of the user requirements	M	M	The team must have a thorough discussion before the team customer meeting to discuss any questions about the requirements of the project. If an issue arises during the development of the project, organise another team customer meeting for clarification of the requirements
R4	Product & Project	New requirements are added or existing requirements are changed	M	L	Adopt an Agile software engineering method so as to keep the structure of development flexible and open to such changes
R5	Product & Project	Developers working on the implementation work on completely separate versions that are difficult to combine	H	L	Have developers frequently upload each new version to GitHub, as well as download the newest version before working on a feature

R6	Product & Project	There are issues when attempting to deliver the final product	L	H	Ensure all team members have access to all parts of the project so that any team member can submit the product
R7	Product	Poor communication and a low 'bus factor' leads to incomplete and missing parts of documentation and implementation	M	L	Conduct weekly/bi-weekly group meetings with a work plan for the following week with job allocations for each team member
R8	Product	Poor quality documentation /documentation that doesn't meet the specification	M	L	All work is peer reviewed in weekly meetings and each team member is given feedback
R9	Product	Update to software we use causes the current system version to become inaccessible or corrupt	L	M	Frequently upload versions to GitHub for version control and minimise loss should the current version become inaccessible or corrupt
R10	Product	Project not completed before deadline	M	H	Ensure the critical path is well defined and closely managed. If the completion of a task on the critical path is delayed, take corrective action promptly in order to get the project back on schedule.
R11	Project	A team member has a hardware malfunction preventing the completion of a task.	M	L	Ensure any hardware problems are communicated to the whole team so another team member can complete the task.

R12	Project	Time is wasted implementing unnecessary features.	L	L	Ensure all features added relate to a requirement as discussed with the customer.
R13	Product	Having different Java developer kits may cause issues compatibility issues	L	L	Ensure all team members are using the same Java developer kits at the start of the project
R14	Product & Project	Loss of data	L	M	Ensure that important files are not only on one machine.