# Risk Assessment and Mitigation

# Group 27 - WaddleWare Studios

**James Smith**

**Amanda Ling**

**Fran Medland**

**Hannah Vas**

**James Kellett**

**Malik Tremain**

**Mischa Zaynchkovsky**

Introduction

The risk management process followed by our team is a structured approach to identify, assess and mitigate risks that could potentially impact our projects objectives. This approach ensures we address any potential issues before they become significant problems which overall reduces the likelihood of project delays and failure to meet the requirements/quality standards.

We started by thinking of potential risks across various aspects of the project. These were product and project related risks. This approach ensures us that no potential risk is overlooked.

For each risk we identified, we looked at the likelihood they would happen and ranked them on a scale of low to high. From there we also looked at the severity of the risks and ranked them using the same scale. The risks were prioritised meaning the risks that were given a high severity were of high priority. This focused our mitigation to the risks that pose the biggest threat to the project's success, ensuring efficient use of resources by addressing the most critical risks first. This was important and ensures us to know what to look out for and how drastic the consequences could be if a risk takes place.

For every risk we looked at how to mitigate them. This includes strategies to reduce the likelihood of the risk happening or could lessen the impact on the project if a risk occurs. Overall, by planning mitigation strategies, this can help to minimise potential disruptions to the project, and can reduce the likelihood and severity of these risks.

We then continuously monitor the project of new risks that can appear and the effectiveness of our mitigation strategies. This allows us to adjust our approach whenever necessary to ensure the project remains on track despite any risks we might encounter. By creating a risk status, it shows what we have done to ensure we didn't encounter any risks/minimising the amount of risks.

Our team's risk register is a spreadsheet which captures all identified risks, the likelihood, severity and mitigation plans in a structured format. This format ensures us that all relevant information is readily accessible which overall facilitates effective risk management throughout the lifecycle of the project.

| ID | Type | Description | Likelihood | Severity | Mitigation | Owner | Risk Status |
|---|---|---|---|---|---|---|---|
| R1 | Product | Libraries are defective/faulty | M | M | Check that the libraries work prior to use, have a backup | Implementation Team | Prior to starting, check made. Backups of the code at regular intervals |
| R2 | Project | Time- tasks take longer than expected to complete | M | M | Overestimate time needed to complete a task to give more time to do it or use gantt charts | Whole team | Gantt charts shows we have enough time to complete each task |
| R3 | Product | Technical issues eg. bugs | H | L | Test the game frequently, documentation | Implementation Team | Before each new commit is made a small test is run, docstrings for each method written |
| R4 | Project | Losing a team member | L | L | Having at least 2 people on every section | Whole Team | More than one person is working on each section |
| R5 | Product | System Compatibility Issues | L | H | Check before starting to code the specifications | Malik | Used gradle, deploys to all systems required |
| R6 | Project | Poor communication among team | H | M | Have regular meetings with the team | Whole Team | Weekly meetings minimum |
| R7 | Project | Not sufficent communication with stakeholders | L | H | Ensure at least one requirements elicitation interview is had | Whole Team | Meeting with customer had on the 27th feb |
| R8 | Project | Inadequate requirements gathering | M | H | Review Requirements before continuing | Hannah & Fran | In a meeting, everyone collaborated on the finished requirements document |
| R9 | Project | Lack of planning | M | M | Ensure a plan is made before starting each mini-project | James S | For each meeting, a small plan is made |
| R10 | Product | Data loss/ corruption of some of the code | L | H | Back up code regularly | Implementation Team | Is saved to a different branch in Github |
| R11 | Product | Over Engineering/ attempting to do too much | L | M | Have a base level of what is required and fulfil that, then look to do more if possible | Implementation Team | All functions are completed before starting the apperances are coded |
| R12 | Project | Poor documentation | M | L | Ensure everything is documented and easy to understand | Whole Team | Javadoc used to explain code, all documents stored where everyone can access |
| R13 | Product | Ambiguity on requirements | M | M | If the customer is ambiguous, ask follow up questions, otherwise make a decision as a team | Hannah & Fran | Meeting with customer had on 27th Feb |
| R14 | Project | Team Burnout | M | M | Split the workload evenly and equally | Whole Team | As a team assigned workloads, ensuring even split of work |
| R15 | Project | Dependancy on one individual | L | H | Ensure everyone is doing a role that has been assigned to them | Whole Team | Gannt chart shows what everyone is working on, for that week |
| R16 | Project | Insufficient detail in architecture | M | M | Ensure that the diagrams are detailed enough for all to understand | Amanda & James K | In a meeting, everyone looked at the finished diagrams to ensure understanding |
| R17 | Project | Physical Resources could be lost | H | M | Using online resources and share them with everyone | Amanda & James K | Backups made |