

# Risk Assessment and Mitigation (2)

**Group Name**

Cohort 1, Group 4

**Group Number**

Cohort 1, Group 4

**Members**

Jude Hall

Rosie Hogg

Ishraan Ismail

Sam Wildgust

Ruby Hanson

Tom Devany

Tomas Asllani

## **Introduction:**

This document is an updated version of the original “Risk assessment and mitigation” document produced by the team we inherited from. The risks were reviewed and revised to reflect the scope, requirements, and development context of assessment 2. Existing risks were reviewed, consolidated, or removed where no longer relevant, and new risks were added to reflect the assessment-specific challenges.

## **Risk Identification**

Collectively, through discussions and brainstorming sessions the team identified potential risks. The confirmed risks were then categorised into technical, project and product related groups in order to streamline their management.

## **Risk analysis**

Each risk was evaluated and assigned a likelihood and severity rating based on collective team judgment. This process is going to significantly assist in controlling potential issues that will have a great effect on the project if not accounted for.

Likelihood measurement shows how likely is a specific risk to become an issue.

- Low - Unlikely to happen
- Moderate - Could possibly happen
- High - Very likely to happen

Severity indicates the impact a risk could have on the project if it occurs.

- Low - Minor impact that is easily managed
- Moderate - Noticeable impact requiring attention
- High - Major impact and potential disruption beyond original scope

## **Risk planning**

Each risk was followed up with analysis focusing on how an issue could be completely avoided or at least minimised. The team implemented a planning system which helped arrange deadlines and track progress in order to minimise risks across the board. Additional brainstorming sessions were held to develop specific solutions to problems in case of minimisation strategies failing. Team members responsible for the discussed aspect of the project were then assigned as the owners of the risk.

## **Risk register and risk monitoring**

To provide structure and consistency to risk management, a register containing the fields described below was constructed.

- ID - Unique identifier of the risk
- Description - What the risk is
- Likelihood - How likely the risk is to happen
- Severity - How big of an impact the risk might have on the project
- Mitigation/Avoidance - How to avoid or minimise the risk
- Owner - Team member responsible for the mitigation strategy execution and monitoring of the risk

ID	Type	Description	Likelihood	Impact	Mitigation	Owner
R1	Project	Team members getting sick, falling behind or struggling with the task	Medium	Medium	Communicate with team members early and redistribute tasks	Ishraan
R2	Project	Lost/forgotten documentation	Low	High	Store all documentation in a shared drive and maintain backups	Ruby
R3	Project	Not accurately following the game/client requirements	Medium	High	Regularly refer to the original requirements before adding new features and have clear communication with team members about client preferences and requirements	Sam
R4	Technical	Bugs and errors in the game during development may cause unexpected behaviour in the game	Medium	High	Use automated and manual testing before integrating changes	Jude
R5	Technical	Inconsistency in game versions on GitHub resulting in unaccounted-for branches	Medium	Medium	Add pull request validation, and CI build checks	Sam
R6	Technical	Difficulty understanding and modifying an existing codebase written by another team	Medium	Medium	Allocate time to review the existing codebase and make changes incrementally, supported by reviews and testing	Sam

R7	Product	Gameplay changes reduce overall enjoyment or engagement for players	Low	Medium	Gather user feedback and adjust gameplay based on evaluation results	Tom
R8	Project	Testing and bug fixing require more time than planned, impacting deadline	Medium	High	Prioritise critical test cases and high risk features	Rosie
R9	Product	Controls or interface may feel confusing to players	Medium	Low	Improve usability through iterative playtesting	Tom
R10	Technical	CI builds or tests may fail due to misconfiguration or incompatible changes	Medium	Medium	Monitor CI results and fix build failures before merging changes	Sam
R11	Project	Misinterpretation of assessment requirements	Medium	High	Review assessment guidelines regularly and clarify all the requirements through team discussions	Ishraan