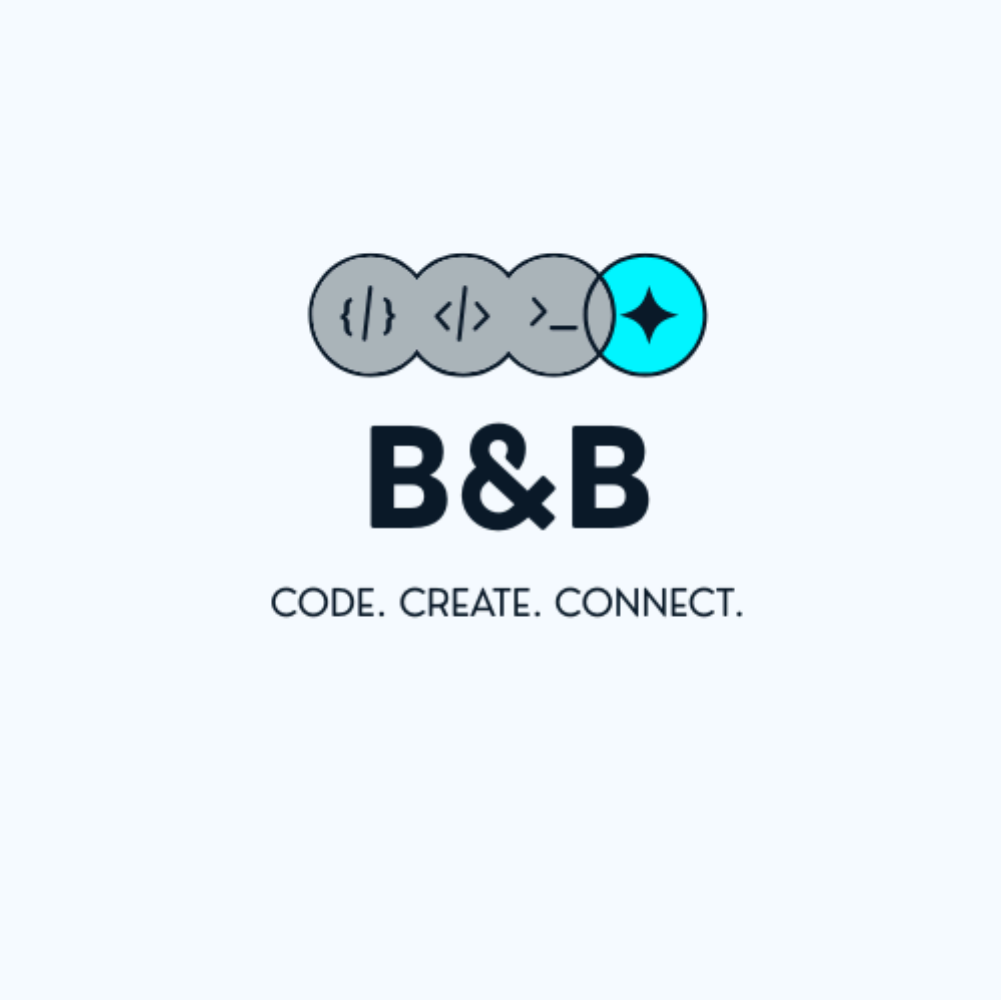
Risk Assessment and Mitigation

Team 8: Ben and Bensons



Team Members: Alyssa Skipper, Ben Senior, Benson Chow, Chloe Ward, Florian Mengkris, Hannah Thomas, James Ingram, Olivia Spencer

Introduction

The purpose of our risk management plan is to identify, evaluate and mitigate potential risks during the development of our maze game. This plan provides a structured method to deal with uncertainty, ensuring that the project can be completed on time and to the required quality. Effective team management also helps our team anticipate potential issues early and make properly informed decisions throughout our development process.

Our team's approach follows the guidance of Sommerville (2015) [1], which works at outlining effective risk management in software engineering as a process of identifying, analysing, evaluating and treating risks. Sommerville emphasises the need for risk management to be integrated into all project activities and describes how organisations should manage risks appropriately. In line with Sommerville, our process involves the categorisation of issues into a risk register to ensure consistency, clarity and traceability in the process of identifying, monitoring and mitigating risks within our project.

The risk register is broken down into **7 categories**:

- **ID** - A unique identifier for each risk (R1, R2, R3…). Used for traceability throughout the planning documents.
- **Type** - The category of the risk, broken down into three categories: *project, product and business.*
- **Description** - A concise summary describing the potential event or situation.
- **Likelihood** - The estimated probability that a risk will occur, ranked from *low to high*.
- **Severity** - The level of impact the risk would have on the project if it occurs. It is ranked from *low to high.*
- **Mitigation** - The planned actions to reduce the likelihood or impact of the risk. This could include preventive steps or contingency plans.
- **Owner** - The specific team member responsible for that risk.

Type categories are adapted from *Sommerville (2015)* [1] which classifies software engineering risks into *product, project and business*:

- **Project** - Affects project schedules or resources.
- **Product** - Affects product performance or quality.
- **Business** - Affects the organisation or customer.

Risks are managed continuously throughout the project as part of our *Scrum [2]* methodology. During each 'sprint' meeting we discuss and review ongoing risks. New risks are identified each week where necessary and mitigations are appropriately planned. Any urgent risks are communicated instantly to the team digitally to be addressed without delay. This approach further supports *Sommerville (2015) [1]* which highlights the importance of early identification and continuous monitoring for effective software risk management.

Risk Assessment Table

ID	Type	Description	Likelihood	Severity	Mitigation	Owner
R1	Project	A team member does not complete their specified task.	Medium	Medium	Reassign that piece of work to an alternative team member.	All
R2	Project	Code is not backed-up regularly.	Low	Medium	Regularly back-up code, ensure code is saved to more than one device.	Ben, Benson
R3	Project	A team member commits code to Git that conflicts with the previous code.	Medium	Medium	Establish known coding conventions to ensure all code is compatible. Ensure members are communicating changes fully.	Ben, Benson, Chloe, James, Florian
R4	Project	Project handover halfway through the project.	High	Medium	Ensure all code is well-documented so the next team can continue easily.	All
R5	Product	Lack of prior experience with frameworks (libGDX, tiled).	High	Low	Ensure all team members understand fully their roles. Confer with other team members	All
R6	Product	Not fully meeting the project requirements in time.	Medium	Medium	Focus sprint goals on core functionality. Regularly update team members on progress.	All
R7	Project	Map incomplete within a reasonable time.	High	High	Delegate design and creation to an alternative team member.	Olivia, Alyssa
R8	Business	Client unhappy with project direction.	Medium	High	Maintain regular contact with clients through Friday meetings and email to ensure the project satisfies all goals.	All
R9	Product	The project experiences performance issues which affect playability.	Medium	High	Optimise code regularly to ensure high performance. Test across different hardware to ensure consistent performance.	Ben, Benson, Florian

R10	Product	Documentation such as UML diagrams, requirement tables or design reports become inconsistent with the code.	High	Medium	Ensure documentation is updated regularly. Check through documentation when implementation is complete.	Chloe, James, Olivia, Alyssa
R11	Product	Integration issues between game components causes a runtime error.	Medium	High	Ensure regular testing is conducted, maintain consistency across modules.	Ben, Benson, Florian
R12	Project	Game files or assets are lost or corrupted during development.	Medium	High	Use Git version control and ensure there are regular backups. Verify integrity before merges.	Ben, Benson, Florian
R13	Project	The team falls behind schedule due to conflicting academic workload.	High	Medium	Adjust the volume of work aiming to be completed, redistribute work to the team for better balance and maintain open communication.	All
R14	Product	Game control or visuals confuse new users.	Medium	Medium	Conduct usability testing with non-team members to gain an outside perspective. Change design based on feedback.	Alyssa, Olivia, Ben, Benson
R15	Product	Audio features malfunction or cause a crash.	Low	Medium	Test implementation early, validate file formats and error-handling.	Florian
R16	Project	Dependencies or third-party libraries become. unavailable or incompatible	Medium	High	Identify alternative libraries early and keep local copies of dependencies.	Ben, Benson, Florian
R17	Project	Failure to identify new or emerging risks during the development process.	Medium	High	Review and update the risk register during each meeting, assign responsibility for continuous risk monitoring	Hannah
R18	Business	Misinterpretation of client requirements results in features not meeting expectations.	Medium	High	Review client meetings regularly. Discuss any ambiguity with the client.	James
R19	Project	Documents may accidentally be deleted.	Medium	High	Ensure each team member has their own individual copy of the	All

					documents they're working on.	
R20	Project	Team members do not show up to meetings.	High	Medium	Ensure talking points of the meeting are well documented. If a team member misses a meeting make sure they catch up on content.	Alyssa

References

[1] Sommerville, I., *Software Engineering*, Pearson Education, 2015.
[2] Schwaber, K. & Sutherland, J., *The Scrum Guide*, Scrum.org, 2020.
https://scrumguides.org