Risk Assessment and Mitigation

GROUP 5 - BITCRUSHED BOB

Maryam Mathews
Joseph Hinde
Jacob Mace
Will Aston
Zathia Jacquesson-Ahmad
Bulganchimeg Munkhjargal
Evan Weston

Risk Management Process

For our risk management process we used a structured iterative decision process. Identification, analysis, planning and monitoring. Using this process we were able to actively monitor, recognise and assess risk.

Identification

We identified risk in a variety of ways; We conducted meetings with all our group members to brainstorm risks. We also incorporated identifying risks as part of our planning process for assigning tasks. We then classified each risk initially into one of four buckets: technical, people, process and external. Classifying them in this way helped us to identify a wider range of risks that would perhaps go unnoticed.

Analysis

We evaluated each risk based on two qualitative measures, likelihood and impact. Likelihood is the probability of the risk occurring and impact is the severity of when the risk occurs. Keeping the categories broad we used high (H), medium (M) and low (L) for both categories. Then we prioritised action against risks by working out the most critical risks (ones with the highest impact and likelihood combined, eg H × H = Critical).

This lightweight approach was chosen as it balances structure with flexibility, making it suitable for smaller projects.

Planning

For every risk we conducted mitigation actions to reduce likelihood (eg git for version control to avoid lost code) and contingency actions to reduce impact (eg local backups if repo is corrupted). Ownership was assigned to individual team members to ensure accountability and follow-up.

Monitoring and Review

We reviewed our risks regularly in a timeslot at the end of each of our meetings, that way we could reassess risk with new tasks or changed tasks. This iterative monitoring helped maintain visibility of emerging threats such as time slippage near deadlines

Risk Register

Risk ID	Risk	Description	Likelihood	Impact	Mitigation	Ownership	Status
R1	Management	Misinterpretation of customer's product brief or requests leads to missing or incorrectly implemented features	Middle	High	Constantly keep focus on requirements duringd development. Request clarification from customers when needed	Jacob ... Josep... Evan W...	In progress
R2	Software	Bugs in the game code prevent it from working	Likely	High	Test code regularly, enforce code review before merges	Jacob ... Josep... Evan W...	In progress
R3	Software	The game might not run or could be slow on low performance laptops	Middle	Normal	Minimize there sources needed to run it. Test the game in advance on low performance laptops	Jacob ... Josep... Evan W... Will As...	In progress
R4	Management	Loss of project data due to accidentally overwriting code, failing to save files correctly, etc.	Low	High	Proper version control through GitHub for the codebase and Google Drive for project-related files	All	In progress
R5	Estimation	The task may take longer than expected, and could result in the project schedule having to be altered	Low	High	Continuously communicate with group members to set realistic goals and evaluate progress periodically for feedback.	Will As... Marya... Bulgan...	In progress
R6	Software	Any 3rd party libraries may not be compatible with the rest of our code	Middle	Normal	Test 3rd party libraries before we use them and be aware of potential alternatives	Jacob ... Josep... Evan W...	Finished
R7	Software	Group members struggle to understand code	Low	Normal	Document code development and what the code does	Jacob ... Josep... Evan W...	Finished
R8	People	A group member becomes unable to continue the project	Low	Normal	Make sure key tasks are not dependant on one person	Marya... Bulgan... Will As...	In progress

R9	Management	Delay in project decisions	Middle	High	To avoid delays in project decisions, prepare alternative solutions and keep	All	Finished
R10	People	Trying to add more many features than planned, which can affect project timeline and quality	Middle	High	Keep the focus on the most important features, don't overload too many low-priority ones and check the project plan often.	Jacob ... Josep... Evan W...	Finished
R11	software	Issues with the screen or a UI element can cause discomfort for the user.	Middle	Normal	Test UI elements on as many devices as possible, and improve them based on feedback	Jacob ... Josep... Evan W...	In progress
R12	People	Lack of experience and skills could lead to low quality function or frequent errors.	Middle	High	All group members gather information about code or other research, share it together, and fill each other's knowledge gaps to improve in a better way.	All	In progress
R13	Management	Unclear assigned tasks could lead to confusion or conflicts among group members.	Middle	High	Keep the focus on the most important tasks,don't overload too many low-priority ones and check the project plan often.	Marya... Bulgan... Will As...	Finished
R14	Management	Licensing issues of project resources may cause legal problems and limit features.	Low	Critical	Make sure all team members are aware of licensing rules and have backup resources in case of possible licensing issues	Marya... Bulgan... Will As...	Finished