**Risk Assessment and Mitigation**

Team 5

Lizzard Entertainment

Gregory Binu
Lydia Eaton
DJ Fox
Daniel Maffei
Michael Papafilippou
Nathalie Rizzo

Our team adopted a structured risk management process based on the framework taught in the risk management lecture [1]. The process involved four main stages: risk identification, analysis, planning and monitoring; this was applied proportionately with the project's size and time constraints taken into account. This ensured that mainly key risks were identified and managed without taking on too much work/overhead.

**Risk Identification:**
Within the planning stage, we discussed potential threats to our progress, focussing on categories recommended in the lecture - technology, people, requirements and estimation. This had ensured we considered a wide range of potential problems rather than just technical ones.

**Risk Analysis:**
Each identified risk was qualitatively assessed for likelihood and impact, using a simple four-level scale: Low, Medium, High, and Very High. These ratings were based on group opinion, judgement and experience, and following recommendations suggested within the lecture. Quantitative scoring was not used as it was deemed unnecessary as this is a short academic project.

**Risk Planning:**
For each risk item, we developed appropriate mitigation and contingency actions. Majority of strategies were aligned with the lectures recommendations:

- Avoidance - choosing stable tools and defining realistic workloads.
- Mitigation - keeping documentation up to date, sharing knowledge to reduce single points of failure.
- Contingency - having backup software options and redistributing tasks if a member became unavailable.

**Risk monitoring and ownership:**
Each risk was assigned an owner, responsible for checking its status and updating likelihood and severity during review meetings. Even though the project's short duration limits proper formal monitoring cycles, the team thoroughly discussed emerging issues and will continue to do so at various stages, in line with the lecture's guidance on continuous reassessment {1}.

**Risk Register Format:**
The team's risk register followed the structure shown in the lecture's example Risk Register. It uses the columns ID, Type, Description, Likelihood, Severity, Mitigation/Contingency, and Owner; the format provides a clear overview of each risk, and its corresponding response strategies. This helps make it easy to track accountability and support any necessary re-planning. Using this concise tabular structure allows the register to function as both a record and a form of communication for the whole team.

Overall, the process followed was deemed appropriate for a small scale software project such as this. It had balanced formal project management principles with flexibility, ensuring that significant risks were recognised early and managed responsibly throughout the project.

**Risk Register:**

| ID | Type | Description | Likelihood | Impact | Mitigation/Contigency | Owner |
|---|---|---|---|---|---|---|
| R1 | Project | Team unavailability - A team member unable to proceed due to illness, commitments or may even have left the project. This would disrupt progress on tasks. | High (Was medium) | High | Mitigation: Internal documentation must be up to date and shared. Single-person critical dependencies must be avoided.<br><br>Contingency: Redistribute workload among all available members fairly. | Michael (Project lead) |
| R2 | Project | Inability to adhere to schedules, perhaps due to inaccurate time-to-complete task estimation. | High | High | Mitigation: Make sure time estimates are generous. It is better to overestimate than underestimate. Also, ensure Gantt Chart is up-to-date and include dependencies.<br><br>Contingency: Re-plan immediately when lack of schedule-keeping is seen. | Michael (Planning lead) |
| R3 | Product/ Technology | Ambiguous or inconsistent product requirements can lead to having to rework designs and miss deliverables. Target customer's hardware or software could be incompatible and may not perform correctly | Low (Was Medium) | Very High | Mitigation: Conduct early reviews with various stakeholders and ensure requirements are documented clearly and concisely. Ensure early access to deployment environments. Ensure target customer's hardware is used to test prototypes.<br><br>Contingency: Build a system that can manage changed requirements to ensure stakeholder satisfaction. Try to change and use components that work universally across hardware and software to ensure universal compatibility. | Nathalie (Requirements Lead)<br><br>Michael (Project Lead) |
| R4 | Technology | Tool failure or library inaccessibility. Tools used for schedule management, project hosting etc could malfunction or become unavailable. | Low | High | Mitigation: Always maintain version-controlled backups elsewhere.<br><br>Contingency: Switch to another perhaps open-source and similar | D.J (Implementation Lead) |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | tool or library. | |
| R5 | Project | Integration issues between work packages between sub-teams could cause unintended delays when commencing the integration stage. | High (Was Medium) | Medium | Mitigation: Schedule checkpoints for integration frequently and make sure cross-team communication is on-point.<br><br>Contingency: Schedule some time for integration testing. | D.J (Implementation lead) |
| R6 | People | Lack of experience or skill gaps among team members could lead to defective or delayed deliverables. | Medium | Medium | Mitigation:<br>Time allocated must be enough to get inexperienced team members up to speed with tools and libraries used within the project. Try pairing less experienced members with more experienced members.<br><br>Contingency: Tools used, skills used and implementation choices should be simplified where possible. | Michael (Project Lead) |
| R7 | Requirements | New requests could expand workload and delay completion, leading to scope creep. | Low (Was Medium) | High | Mitigation: Deliverables must be clearly listed and defined. Ensure team wide approval is received when requesting changes.<br><br>Contingency: Decompose the project, identify and prioritise essential and non-essential features. | Nathalie (Requirements lead) |
| R8 | Documentation | Underestimating or procrastinating documentation and reporting could lead to inefficiency and progression to other subsequent phases taking longer than expected. | Medium (Was high) | Medium | Mitigation: Make sure documentation is carried out alongside technical work.<br><br>Contingency: Working sessions (such as project management and discussion sessions) could be extended as the deadline approaches to catch-up with documentation. | Michael (Project lead) |
| R10 | Technology/ Development | External dependencies could fail and libraries used could be obsolete. | Low | High | Mitigation: Ensure well-established and reputable, regularly updated APIs and libraries are used. | D.J (Technical lead) |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | Contingency: Fallback APIs could be set-up. | |
| R11 | Product | Unstable or undesirable deliverable quality. Work produced may fail to meet acceptance or testing criteria. | Medium | Very High | Mitigation: Iterative Development cycles with early testing phases could be utilised. Code reviews and Q/A sessions with team members could be held to ensure deliverable quality.<br><br>Contingency: Allocate some time window to bug-fix and test before submission as well. | DJ (Implementation lead)/ Nathalie (Requirements lead) |
| R12 | Communication | Miscommunication among team members could lead to lack of consistency with updates and overall lack of shared understanding which could further lead to duplicated or broken work. | High (Was Medium) | High | Mitigation: Schedule regular meetings and host documentation centrally and shared. This includes our biweekly meetups and our use of a shared google drive and shared github codespace.<br><br>Contingency: Perhaps assign a Communication coordinator. | Nathalie (Requirements lead)/ Michael (Project lead) |
| R13 | Project | Failure to re-assess risks and neglecting risk monitoring could lead to risk to escalate unintentionally. | Medium | High | Mitigation: Allocate time to risk review and assign clear ownership to each identified risk.<br><br>Contingency: Hold an emergency review session where new high priority risks will emerge. | Greg (Risk Manager) |

**References:**

[1] Kolovos, D (2025), *Project Planning and Risk Management*, Department of Computer Science, University of York.