

# Risk Management

Cohort 4: Group 2

## Greyhounds

Olwenn Gillespie

Glenn Asher

Praise Konji

Jennifer Oakley

Alice Cavens

David Naji

Igor Dylewski

## Risk Management Process & format of Risk Register

### Our process

We run a lightweight, iterative cycle tuned to the project:

1. **Identity:** In Week 1 we brainstormed by **category** (People, Requirements, Technology, Product, Schedule, Handover etc). We pulled risks from the product brief, our architecture and past student mistakes (eg. Assignment Handover)
2. **Analyse:** For each risk, we assigned **Likelihood** and **Impact** on a **simple L/M/H scale** (low/moderate/high).
3. **Plan:** Each risk gets; a **mitigation** (how we reduce the likelihood/impact now), and a named **Owner** accountable for keeping it under control.
4. **Monitor:** We review the register weekly in the meetings. Owners update Likelihood and Impact if circumstances change, and we watch early indicators (eg failing CI). If indicators trigger, we act immediately: re-plan scope, swap tasks rather than rely on last-minute adjustments.

### How we put this into operation

- The register lives in the team google drive so team members can easily find it.
- We keep the **bus factor** > 1 on critical areas
- Risk discussion is a **standard agenda** item at team meetings

### Register Format

We use a compact table that matches the marking cues ("likelihood, impact, mitigation, ownership") while remaining easy to check through:

- **ID:** For traceability
- **Category:** People/Tech/Product/Requirements/Schedule etc
- **Description:** statements about the risk
- **Likelihood:** L/M/H
- **Impact:** L/M/H
- **Owner:** Named team member

### Justification

- **Ownership + weekly review** ensures risks are actively managed, not just listed
- **Categories** ensures balanced coverage of team, process, etc rather than a code-only focus
- **Traceable IDs** makes it easier to reference risks in issues
- **Clarity& shared understanding:** Simple-language risk statements

## RISK REGISTER

ID	Category	Description	Likelihood	Severity	Mitigation	Owner
R1	People	Low participation / non-participation from a team member(s)	L	M	Pair work, track evidence of participation, intervene early, talk to lecturer early	Jennifer
R2	People	Interpersonal conflict	M	M	Use 'six thinking hats' in debates, encourage reconciliation	Jennifer
R3	People	Forced takeover lowers bus factor	M	M	Assign shadow for each role; enforce code reviews	Praise
R4	People	Time management issues with coursework from other modules	M	M	Weekly review; maintain documentation; escalate early if issues arise	Jennifer
R5	People	Creative blocks	M	M	Take breaks; seek inspiration from other sources	Jennifer
R6	Technology	Chosen Java game library is a poor fit or under-documented	L	H	Make prototype earlier; carry out research prior to choosing game library	Alice
R7	Technology	Low performance on lab PCs	L	H	Target 60fps on lab PCs	David
R8	Product	Audio issues on lab PCs (drivers/latency)	M	M	Use a widely supported backend (JavaSound or OpenAL-Soft via LWJGL); avoid platform specific drivers	Alice
R9	Product	Collision/physics bugs	M	H	Unit tests on collision utils, deterministic step: fixed-timestep physics(eg 60 Hz) decoupled from render; clamp dt.	Igor
R10	Product	Assets (art/music) consistent style/quality	M	L	Establish style guide early	Alice

R11	Product	Accessibility/usability below expectations	M	M	Color-blind mode enforced, readable fonts	David
R12	Handover	Poor handover docs for A2 takeover team	M	H	Produce clear and understandable architecture	Praise
R13	Handover	Team inherits poor quality code from previous team	M	H	Early audit; peer review of other team's code	Jennifer
R14	Testing	Insufficient time for playtesting and polishing	M	M	Schedule playtest sessions, polish checklist + severity triage	Asher
R15	Gameplay	Difficulty not tuned; players cannot complete in ~ 5 minutes	M	M	Playtest with random users; add hints	Asher
R16	Requirements	Misread product brief/customer	L	H	Confirm assumptions in minutes; early playback to customer	Olwenn
R17	Requirements	Late requirement changes	L	H	Change log with rationale	Asher
R18	Project	Cross platform packaging issues	M	M	Test on lab PCs, provide a safe mode build(reduced effects)	Alice
R19	Project	Scope creep (A2 features leaking into A1)	L	M	Confirm in practicals; use MoSCoW; defer to A2 when possible	Jennifer
R20	Project	Computer damaged/broken	L	H	Constantly back up on individual computers; preferably work on the cloud ie google sheets	Jennifer
R21	Project	Input handling inconsistent across OS/keyboards	M	M	Abstract to actions (e.g Jump, Pause) instead of binding to raw keycodes, test on windows/macOS/Linux in CI and on a lab PC	David

R22	Project	Long load times due to 'asset pipelines'	M	L	Bundle many small files into a few packs to reduce disk seeks, cache decoded assets	Igor
R23	Project	Project deadlines missed/late	M	M	Project/Team lead constantly checking in on progress and timelines	Jennifer
R24	Project	Frame pacing inconsistent	M	M	Cap FPS/vsync, fixed-timestep	Igor
R25	Project	Device-specific paths/permissions break saves on lab machines	M	M	Use user-home-app-dat a directories; cross platform paths	Alice
R26	Traceability	Gaps between requirements - architecture - code - tests (marks loss)	M	H	Maintain ID scheme; reference IDs in code/tests names	Olwenn
R27	Project	Timer tied to frame rate - game speed varies with FPS	M	M	Use fixed timestep for logic; test across 30/60/120 FPS profiles	Alice
R28	Build	Release blocks failing hours before deadline	M	H	Project needs to be fully done 2 days before deadline	Asher
R29	Website	Submitted URL or website structure incorrect - markers cannot find material(marks at risk)	M	H	Website checklist; peer review before submission	Igor
R30	UX	Aspect-ratio/resolution scaling issues (projector/ultra wide) cause UI clipping	M	M	Auto-scale UI; test on multiple resolutions	David
R31	Docs	Over page-limit for Risk1/Req1/etc - excess content ignored	L	H	Page-counter review	Jennifer
R32	Legal	Using 3rd party assets like music & art without proper permissions	L	H	Verify permissions before using any external assets	Igor