Risk Assessment

Group 15

HesHus

Adam Robinson

Tsveta Ivanova

April Jack

Oliver Goodwin

Luke Jackson

Luca Hammond

Chris Rolfe

A risk assessment is crucial for planning potential issues during a project, ensuring smooth execution with minimal risk of disaster. Our plan used techniques from lectures and the book ‘Software Engineering’, consisting of four steps: Identification, Analysis, Planning and Monitoring.

Identification

Our group began the identification phase of the risk assessment by brainstorming and using past project experiences to find initial risks during one of our meetings. Then, we categorised them into three different groups:

- ***Project*** - schedule or resource impacts, *e.g. staff turnover causing delays*
- ***Product*** - quality or performance issues of the developed software, *e.g. using flaky libraries causing lack of quality*
- ***Business*** - risks that threaten our team, *e.g. team conflicts*

Analysis

Once we had finished identifying and categorising risks, we had to consider each risk’s likelihood of happening and its severity should it occur. Since this is a judgement call, the team members overseeing the risk assessment discussed these values for each risk. This was followed by a team discussion to make sure everything looked good.

Planning (Mitigation)

Now we had a list of significant possible risks and their analysis, we needed to come up with measures to avoid these risks. For some, we know that we can’t avoid them so we come up with a plan to try and minimise the impact if it does happen - for example, assigning at least two members to each task in case someone falls ill - so the other person can efficiently take over.

Monitoring

After assigning risk owners, team members continuously reassessed and reported their risks’ likelihood, severity, and new risks, using a reassessment table and new risk list (present in our group’s website). These updates were made consistent using regular WhatsApp messages and mentions in group meetings.

Register Format

Following the process from the ‘Software Engineering’ book, the risk register is a combination of the sub-tables created from each stage, with an ID column so each risk can be found more easily:

- ***Identification***: Type (Project/Product/Business) and Description (what the risk is)
- ***Analysis***: Likelihood (scale of how likely is it to occur- Low, Medium, High) and Severity (scale of the effect on the project if the risk happensLow, Medium, High)
- ***Planning***: Mitigation (measure taken to avoid/mitigate the risk)
- ***Monitoring*** (From lecture, not the book): Ownership (team member who will monitor this risk)

ID	Type	Description	Likelihood	Severity	Mitigation	Ownership

r1	Project + Product	Incompatibility issues between team members + customers with java version	L	H	Develop the game in Java 11 only, so all team members use the same version, and customers can run it too	Luke
r2	Project + Product + Business	Licensing issues with assets/libraries	L	H	Use known open source assets/libraries. Investigate into paying for licensed ones.	Chris
r3	Product	libGDX may not have the capabilities to full our requirements	L	H	Research other open source projects similar to ours that use libGDX, also consider other game engines	Tsveta
r4	Project	Underestimating the time required to develop the software	H	M	Create and follow a project schedule. Weekly meetings to make sure everyone is on track	Oliver
r5	Project	Underestimating the complexity of the software being developed	M	M	Assign multiple team members to development, so they can help each other. Generate a simple version of the complex feature for the time being	Luca
r6	Project + Business	Team members may not all have equal amounts of work allocated to them	M	M	Ensure that each person has ~13 marks of work to be completed for the deliverables	April
r7	Project + Product	Team members might get sick/leave the team/lack of involvement	M	M	Assign at least 2 team members to each task, with one being a ‘shadow’ member. This ensures that if someone can't work on it anymore, someone else can take over	Adam
r8	Product	Miscommunication/lack of communication	M	M	Establish communication channels (Whatsapp, Slack), meet as a group bi-weekly, and explain assignments clearly	Tsveta
r9	Product	Initial requirements are vague	M	M	Create an exhaustive list of questions to ask the client, and ask any	April

					follow up questions from the response. If unsure on anything, book another client meeting	
r10	Project + Product	Slow/buggy libraries used	L	M	Research libraries beforehand - check their benchmarks and issuetracker to indicate performance and amount of maintenance. Hide libraries behind a facade so if they have to be replaced we can change the facade	Chris
r11	Project	Scope creep requirements increase/change as project progresses	L	M	Ensure requirements are clearly defined at the beginning, evaluate the risk of extra requirements affecting the schedule	Tsveta
r12	Product	No budget so assets must be found for free	H	L	Search for free assets, inform stakeholders that assets will be lower quality compared to expensive ones	Luke
r13	Project	Underestimating the amount of learning it takes to understand the technology	M	L	Keep checking up on each team members understanding of what's going on, and providing help if required	Oliver
r14	Project + Product	Lack of coordination due to not having an official project manager to run the project	M	L	Have leaders within the group to check on how others are doing and maintain the workflow of the project	Adam
r15	Business	Team conflicts may arise	L	L	Intentionally design team culture from the start and periodically assess team health. Discuss any issues with the group and explore ways to resolve them	April
r16	Business	Not meeting the stakeholder's requirements	L	H	Ensure that questions are specific enough so that there is no room for ambiguity in the requirement	Adam

r17	Product + Project	Each member's equipment may be in differing conditions	L	M	Ensure all software used can be run on the oldest laptop/computer so all members can utilise the tools needed	Oliver
r18	Project	Some members may miss important deadlines	M	H	Utilise meetings to check on the progress of work so help can be provided if needed	Luca
r19	Project	Data loss due to physical or digital damage/corruption	L	H	Ensure that all files are saved on multiple devices as well as a form of cloud storage so that even if one storage device is lost, all the data can still be accessed	Chris
r20	Business	Violating copyright laws/not obtaining necessary licences for assets	L	H	Check the licencing of all assets to ensure they can be used for free in our game without violating any agreements	Luca
r21	Product + Project	User/system requirements are dissociated	M	M	Extra communication between group members working on requirements and with the client	Adam

An additional risk reassessment table is available on the website, under ‘Risk Reassessment Table’.

References:

- Somerville, I. 2015. *Software Engineering.* Pearson Education