Risk assessment and mitigation:
Team 23

Kyle Mace (kjm560)

Josh Quinn (jtq501)

Louis Hatton (lwh506)

Faris Alblooki (fma527)

Lewis Power (lp1263)

Part A: Risk Format and Level of Detail

Our team identified three variations of risk that could possibly emerge:

- **Project** – affects project schedule or resources
- **Product** – affects product quality/completeness
- **Business** – affects the organisation procuring/developing the software

In order to account for these, our team delineated a four-step process in order to ensure that no risk could dramatically affect our project.

1. Identification of all conceivable risks
2. Analysis of each risks likelihood and severity
3. How each risk can be avoided, minimised, or mitigated

➔ An ongoing monitoring of whether any risks are taking place

Before any programming tasks were assigned, our team thought it would be logical to have a brainstorming session, wherein we suggest all risks that could possibly emerge during the project. Based upon our research, **[1]** we decided to characterise each risk along 6 dimensions.

- **Risk ID** – lends itself for easy and accurate reference during discussion.
- **Risk Description** – a description of the risk.
- **Risk Likelihood** – the probability of the risk occurring: low, medium, or high.
- **Risk Severity** – the damage that the risk will afflict to the project if it occurs: low, medium, or high.
- **Risk Mitigation** – how our group will endeavour to prevent, mitigate, or minimise a given risk.
- **Risk Owner** – who is assigned to keep track of the risk and alert the team should it emerge.

Each of these makes up a column within a risk register, sorted by the most probable and most severe risks, to the least probable and least severe risks. There is a risk register for each of the aforementioned variations of risk: project, product and business. Each risk register has been coloured differently in order to allow for easier distinction between them.

As our project is a relatively small one, developing non-critical software, we have deliberately limited the scale of risk to just three: L (low), M (medium) and H (high).

Part B: Tabular Representation of Risks

Project Risks:

ID	Description	Likelihood	Severity	Mitigation	Owner
R1	A team member falls ill and is unable to work for a period of time	M	M	Communication should be done through chats which everyone can access in case someone needs to take over another's work	Everyone
R2	A team member does not complete their task by the internal deadline set	M	M	Assess the reason behind the missed deadline. Distribute work among other team members to ensure development does not slow	Everyone
R2	A team member drops out or is unable to/won't assist in further development	L	H	Following from R1, ensure all work is well communicated so that other members can pick up where they left off.	Everyone

Product Risks:

ID	Description	Likelihood	Severity	Mitigation	Owner
R	The product is faulty	M	H	Ensure the code is subject to rigorous testing before production	Everyone
R	Player's screen size is larger than the max tested for	H	M	Ensure throughout the development stage that the game screen can be resized and it scales proportionately	Josh Q
R	Game is over-engineered with features no required	H	L	Have regular team meetings and outline exact requirements ensuring they are stuck to	Everyone
R	Player's computer does not have relevant software installed to support the game	L	H	Ensure we use the highest supported JDK	Faris A, Josh Q, Louis H
R	Features which are required are not implemented	L	H	Prioritise developing the core features in the planning states, with a focus on functionality first	Faris A
R	Player's hardware cannot support the game	L	H	Minimise intensive operations which may cause some machines to slow down	Faris A, Josh Q, Louis H
R	The hard drive where the software is stored fails	L	H	Make use of online repositories such as GitHub to ensure the code can be accessed by any team member	Everyone

Business Risks:

ID	Description	Likelihood	Severity	Mitigation	Owner
R	New requirements are introduced by the customer mid production	M	M	Follow an agile approach to the development so that new requirements can be easily implemented in a short period of time	Everyone
R	Goals are not met within the timescale planned	L	H	Have weekly team meetings and set deadlines so that the team knows at	Everyone

				what stage they are at and whether focus needs to be directed elsewhere	

Bibliography:

[1] https://www.castsoftware.com/research-labs/software-development-risk-management-plan-with-examples