Risk Assessment and Mitigation

TEAM 32

Team Members:

Ethan Higgins, Lakhan Mankani, Clara Hohenlohe,
David Thomson, George Burke, Adam Gurdikyan

Introduction:

A risk is simply defined as anything that could threaten your project, it is something that may or may not happen. Risks can be put into two categories, known and unknown. Risk management is the skill of being able to identify new risks, whilst also being able to monitor and mitigate risks. This skill is crucial for software projects, especially as the uncertainty of the project grows.

There are also different types of risks, firstly, we have project risks; these are risks that directly impact the project schedule and also affect the project resources, an example of this would be, academic misconduct by a team member. Secondly, there are product and project risks; these are risks that are influenced by mistakes with requirements or changes to requirements and also can be caused by specification delays. An example of this would be data loss. Thirdly, just product risks, are risks that affect the product quality and completeness, an example of this could be inconsistency within the projects coding conventions used within the code. Moreover, there is also the risk of technology, this type of risk could appear if the project does not work or runs slower on the customers' hardware. Finally there are business risks that could occur; these impact the organisation that goes into developing and procuring the software of the project. If the project is not up to standard compared to other competition in 'the market' this would be considered a business risk that has taken place.

Our group has taken into consideration that these risks exist and as a type of risk management, devised a way to measure risks and therefore, making us more capable of managing and mitigating the maximum amount of risks. We have two forms of measure one being the likelihood of the risk occurring and the latter being the severity of the risk, each given a rating out of three; either being low, medium or high. Given these measures we can organise our time and effort more efficiently when it comes to mitigating risk. For example, if a risk has a high likelihood of happening and also has a high severity rating then this particular risk should be taken into consideration with a lot more care and a lot more thought than a low likelihood risk with a low severity rating.

As a further precaution to avoid risk, we identified each risk we could think of occurring and also assigned an owner of the risk, who is in control of the risk management for that particular risk. This way, we can divide the crucial task of risk management between the group minimising the potential problems that could occur from poor risk management.

We also identified precautions to take for each risk, to decrease the severity or the likelihood of the risk happening.

With all these methods of risk reduction and management, our group would successfully mitigate the maximum amount of risk in one of the most efficient ways.

Risks:

ID	Type	Description	Likelihood	Severity	Mitigation	Owner
R1	Project	Team member unable to work for period of time	M	M	Redistribution of work	David
R2	Project	Academic misconduct by a team member	L	H	Make sure nothing is plagiarised	Ethan
R3	Project	Not being up to date and working with old code	M	L	Use GitHub for version control	Lakhan
R4	Product	Data Loss	L	H	Backups/Repositories	Lakhan
R5	Technology	Unavailability of suitable equipment to work on project	L	H	Public computers	George
R6	Business	Product is not up to standard compared to other competition	L	H	Revision of product	Ethan
R7	Technology	Game runs slower/unable to run on customer hardware	L	H	Testing of software across different PC's	David
R8	Product	Inconsistent coding conventions used within code	L	L	Following a set of prior agreed coding conventions	George
R9	Project	Project estimations are incorrect leading to being behind schedule	M	H	Periodically reviewing project estimations in accordance with the provided deadline	Clara
R9	Project	Creative disagreements among team	M	M	Have good communication between team members and	David

		members			decide which ideas we should implement as a team.	
R10	Business	Changes in requirements from customer	L	H	Stay updated with what customers want, make sure we have a good understanding of what the project is before we start coding.	Adam
R11	Business	Project development exceeds that of provided scope, leading to potential late submission	M	M	Constantly ensure that project development adheres to the provided scope.	George
R12	Project	Poor Productivity	H	H	Team members ensuring everyone else is working on the project and doing their assigned weekly tasks via Trello.	Lakhan
R13	Project	Inadequate Risk Management	M	M	Create a table containing possible risks and how to avoid them while making sure to evaluate our project against it.	Everyon e
R14	Project	Inadequate documentation and commenting of code	M	M	Ensure everyone uses the same docstring format, and comments their code adequately	David
E15	Project	Poor team management	M	H	Have regular team meetings and assign weekly tasks to team members.	Adam
R16	Business	Low stakeholder engagement	M	H	Schedule customer meetings as needed to get frequent feedback from stakeholder	David
R17	Technology	Using outdated software	L	M	Ensure software used by the team is up to date	George
R18	Project	Project is leaked to external sources	L	L	Ensure project files are only accessible by team members	

R19	Project	Disgruntled team member	L	M	Ensure team members are happy with the project	