Risk assessment and mitigation

Group 9

Group name: KOWLAAB

Aiden Sayer

Lurvïsh Polodoo

Ben Salkield

Oliver Rogers

Alex Sharman

William Roebuck

Kiyan Eiles

Risk Management Process

Introduction of the Risk Management Process:

Risk assessment is key to ensuring that this project is successful in meeting the requirements set efficiently and effectively. Risks are future foreseeable and unforeseeable events that can threaten the project. It is important to identify risks and present mitigation strategies for risks to be certain that the project runs smoothly and the final product meets the customer requirements set by the customer.

Types of Risk:

There are many different types of risks that could affect the development of a product in a project, the main four being project, product, project & product and business risks. The different types of risks allows us to categorise threats to the project and product which helps us develop and maintain an organised risk register of all the known risks to a project.

Project risks are risks that affect the schedule or resources of the project. These risks are important to identify and prepare for as they affect the productivity of the project leading to the final product being delivered late or unfinished due to delays.

Product risks are risks that affect the product's quality and overall completeness of the final product; for example, product risks take the form of code bugs and bad quality libraries. Product risks have the effect of causing the quality of the final product to be less than the user requires from the product.

Certain risks can affect both the project and the product. These risks, for example, can take the form of the misinterpreting/changing user requirements and delays in the developmental process. These risks are especially dangerous as they threaten the overall project integrity and the final product delivered to the customer.

Finally, business and technology risks are risks that affect the overall development of the software such as competition with an alternate product that fits the customer requirements and obsolete technology which may lead the final product to be obsolete by the time it is presented to the customer, threatening the entirety of the project.

Risk Register:

The risk register is a spreadsheet that allows us to effectively track potential risks using an id system. Each risk is allocated a type (Product, Project, Product and Project, Business and Technology, relating to coding problems during the development process), a short description of the potential risk, likelihood and severity and finally mitigation strategy and owner for the risk. Using this register allows us to quickly identify, analyse, plan and monitor all the potential risks that may appear throughout the developmental process of the product.

ID	Type	Description	Likelihood	Severity	Mitigation	Owner
01	Project	Person in charge of implementation becomes unavailable.	Low	High	Have 2 people working on the implementation.	Lurvïsh
02	Technology	Game engine does not support isometric development.	Low	Moderate	Switch to top-down.	Alex
03	Project	One or more members unable to commit to the repository.	Low	Moderate	Send files via discord so that someone else could commit the changes after reviewing the code.	Ben
04	Project	Too many merge conflicts leading to unresolvable code at the end.	Moderate	High	Do regular commits to resolve conflicts in a manageable manner.	Alex
05	Product	Game development is taking longer than expected.	Moderate	Moderate	Assign more people to the development process while distributing work in an effective manner.	Oliver
06	Technology	IDE (VS Code) might not be compatible with the game engine (libGDX).	Low	High	Switch to IntelliJ or Eclipse for development.	Ben
07	Project	Each team member involved in implementation might have a different coding style which can lead to inconsistent code.	High	Moderate	Establish a coding style and have regular code reviews to make sure everyone is on the same page.	William
08	Project	Lack of user/client feedback might lead to not meeting their expectations.	Moderate	High	Have regular meetups with the client to show the progress of the game while gathering feedback.	Aiden
09	Project & Product	Use of copyrighted material or licensed libraries could lead to legal issues.	Low	High	Ensure all assets and libraries used to develop the game have an appropriate licence for us to use. If time allows, create original assets where possible.	Ben
10	Project	Poor Communication	Moderate	Moderate	Hold regular meetings and	Aiden

		Between Team Members.			ensure a clear communication path is established.	
11	Project & Product	Adding new features towards the end of the implementation stage might lead to incomplete/erratic features.	Low	Moderate	Clearly define the scope from the start.	William
12	Project & Product	Insufficient testing can result in undetected bugs.	Low	Moderate	Allocate sufficient time for unit, integration and user testing to make sure things are working as expected.	Ben
13	Product	Unbalanced gameplay.	Moderate	Moderate	Conduct regular play-testing to ensure the difficulty level is set correctly to maximise user enjoyment.	Oliver
14	Product	Code with bugs being pushed to the main project. Causing features to break and wasting time fixing it.	Moderate	High	Ensure all code is tested and proven to work before it is pulled to main.	Alex
15	Project	Pushing code without authorisation from lead. Causing confusion between versions and potential for bad code to be pulled.	Low	High	Ensure all code is properly added via pull requests through gitHub	Alex
16	Product	Not having someone available to accept pull requests.	High	Low	Having another person available to accept pull requests if they are not available to review code.	Alex
17	Product and Project	Running over deadlines for coded tasks.	High	Moderate	If it is clear that someone is unable to finish a task allocated to them in time then allocated someone else to help or extend the deadline.	Kiyan
18	Product	Multiple people working on the same task causing conflicts in implementation.	Moderate	Moderate	Before any code is written, agree upon a plan for the implementation for the given task.	William
19	Product	A coded task being forgotten and not implemented.	Low	High	By planning who does what task before, we can check up and remind each other	Lurvïsh

					what tasks need to be done.	
20	Product	Revisions of code causing time loss.	Moderate	Moderate	Planning ahead of time what each piece of code needs to achieve before implementing.	Kiyan
21	Project	Revisions of tasks/requirements causing time loss.	Low	Moderate	Making a plan and agreeing what the product must do. Ensuring that requirements are clear and understood.	Oliver
22	Technology	Poorly written code affecting performance.	Low	Moderate	Product is small scale therefore it would be easy to meet performance requirements as long as code is written upon an agreed upon standard and efficiency.	Oliver
23	Product	People relying on other tasks to be completed first.	Moderate	High	Agreeing upon a set of deadlines and meeting them. Giving alternate tasks that someone can do while waiting to minimise wasted time	Lurvïsh