5. Risk assessment and mitigation

Cohort Number: 1
Group Name: NeveSix (Team 6)
Group Member Names:

- Sam Savery
- Ewan Hutcheson
- Igor Smolinski
- Sanjna Srinivasan
- **Pranshu Dhungana**
- Phoebe Russell
- Zack Tyler-Kyle

Risk Management Plan

Identification process

We are going to be using a deliberate and systematic approach to identifying risks.Our risks are only going to belong to four categories:

- Product: Risks associated with user experience in regards to the product
- Project: Risks regarding project development and management
- Technology: Risks concerning hardware and software components during or after development.
- Business: Risks related to the general ongoing operations of our project once its released, which could be legal or financial.

This is a form of abstraction so we try to only focus on what matters in our case. So we aren't looking at other irrelevant details. Also not going to be focusing much on business risks due to the lack of scope for our project.

Risks will be identified through regular brainstorming sessions , and will be updated throughout the documentation process.As they are identified we will be putting them on the risk register, which is in a spreadsheet format.

There will be a column for ratings for likelihood and severity. They will be rated in terms of low to high. There will also be an id column so that each risk can be uniquely identified and easily referenced. Then there will be columns for type(category) , risk description and then a column for mitigation plan and avoidance strategy, in order to prevent and deal with risks if they do occur.

When a risk has been identified after occurrence , we will work on a way to solve the issue and add it to the risk register , with an avoidance plan in place. This is to avoid further occurrences.

Risk ownership strategy

Each of these risks will have an owner in charge of reassessing the likelihood and severity of the risk regularly. Owners should bring up the status or report on their risks every meeting.

Risk reviewing plan

When possible we should be following avoidance plans for each of the identified risks. Owners should be reminding us of this.

If a risk does cause problems then we will work together to follow through on the mitigation plans.These will have already been thought about when adding the risk to the register.

Risk Register

ID	Type	Description	Likelihood	Severity	Mitigation and Avoidance	Owner
R1	Project	Improper distribution of workload causing significant imbalance in staff contributions	M	M	Preplanning and agreement upon roles prior to beginning the project. If problems occur during the project then just address and reallocate the workload during team meetings.	Zack Tyler-Kyle
R2	Product	Mislabing of variables in code impacting code readability, increasing time taken for other staff to understand recent additions to code.	M	L	Predetermine code naming conventions and acceptable contextual names. Make sure progress during implementation is explained to the rest of the team incrementally during meetings or commit notes. If issues arise with readability then direct concerns to the group/person working on the code .	Ewan Hutcheson
R3	Project	Lack of communication resulting in staff not knowing what is to be worked on in a given session.Therefore losing track of progress.	L	H	Arrange weekly meetings outside of practical sessions and record every meeting within the logbook. If any team member is not sure what to do then, they should check the logbook and ask the rest of the team for help.	Zack Tyler-Kyle
R4	Technology	Poor internet connections/loss of connection hampering or preventing continuation of session.	H	M	Save and backup work frequently and have multiple staff in a session to cover if needed.	Pranshu Dhungana
R5	Technology	Github crashing or problems regarding access to project.	L	H	Have copies of all work on team members' computers. When issues occur then work can still be done on individual computers	Sam Savery

R6	Project	Staff member unavailable due to illness, unforseen or exceptional circumstances. Which may cause loss or lack of progress	H	L	Keep track of meetings with the logbook and promote communication and keeping the team informed of potential absences, so the session plan can be modified in advance. Then ensure the absent member is updated on details. Absent members should always check the logbook.	Pranshu Dhungana
R7	Project	Administrative issues wherein not everyone may be able to access a certain document or part of code and review it. This may lead to errors with documentation/implementation .	L	L	Ensure everyone works using a shared space such as google drive. Updates should be made to the documents stored on the team 6 drive. If anyone is unable to find or access documents that another team member is working on then this must be addressed by the owner of the documents/task.	Sanjna Srinivasan
R8	Business	Licensing issues in regard to assets used.	L	M	Inform members about each asset and let them know the source. Check licensing for each of these assets and research sources.Every asset must have documentation detailing the	Igor Smolinski
R9	Project	Bad updates which might cause errors for the rest of the project.	M	H	Make sure each potential merge and commit is peer-reviewed by the team.Ensure there is a way to revert changes by using branching and also backups.	Igor Smolinski
R10	Product	Problems during set-up or while running the program might be hard to diagnose. The program could crash and there is no clear reason for it.	H	H	Make sure that during development, have error handling procedures in place. Therefore it should be easy to find out about the source of the error. This could be done through proper class structure and modular design. So if a certain screen crashed then it may be a problem with the class pertaining to that screen. Also, have a way of ensuring the proper working of certain parts of the program during , e.g if a certain variable was true then that function worked correctly.	Igor Smolinski

R11	Technology	Updating the project may lead to certain requirements such as non-functional/functional requirements not being met.	M	M	Ensure to use a test-driven approach so each update has to pass a series of tests which are designed in regards to requirements. These could be unit tests or just general walkthroughs of the program to ensure the game functions properly.	Phoebe Russell
R12	Product	Product not working for a certain OS. Like UNIX systems.	M	M	Ensure that testing is done on multiple OS systems. If problems are found try to find solutions through research and working with the rest of the team.	Phoebe Russell
R13	Product	Game might be difficult to understand and hard to play.	M	M	Make sure to design requirements to prevent the game from being difficult or confusing for the player. This could be through implementing tutorials and intuitive features. Moreover, test the game out with those not involved in development to prevent this from occurring and if is found to be difficult then work on feedback and make some changes.	Sanjna Srinivasan
R14	Project	Conflict arising between members in regards to distribution of work. Preventing progress.	M	M	Make sure work is distributed during each meeting, members should discuss and work together to come to an agreement on who needs to do what. If two people wish to work on a task requiring only one person then the rest of the team should hear them out and decide on how to resolve the conflict.	Phoebe Russell