# SEPR 2019/20 Assessment 2

## Team CheatCodez


## Jonathan Grout, George Lesbirel, Cheuk Wang Wu, Lauren Quarshie, Muaz Atif, Lillian Coultas


# Risk Assessment and Mitigation

(V2 Updated From Assessment 1)

# Introduction to Risk Management

Please refer to the “Amended” column in risk assessment below for updates indication. Risk approach remained largely unchanged. Justification can be found on deliverable.

Team brainstorming session was arranged with the objective of identifying the different types of risks threatening the project. Risk Breakdown Structure (RiBS) was produced (*Appendix 4.1*), which the arrangement of potential risks in a categorized format subsequently produces the risk register.
The risk format we have chosen to implement consists of a formatted table, in which the risk management process has been condensed, and includes an analysis of product and project risks. We have also developed a risk matrix (*appendix 4.6*) through literature reviews of our documentation and product brief, as a way to characterise the level of risk by examining the degree of severity against the likelihood of occurrence. The generated matrix improves the visibility of risks, aids in decision making, and ensures that the worst case scenario can be ascertained with ease. The use of a risk matrix also facilitated the risk identification process; for example, risks with a very low probability or severity were easily identified and consequently eliminated.

- Green zone contains low-risk and no further mitigation is required.
- Yellow zone contains moderate-risk and may not be acceptable.
- Orange zone contains high risk and may be a significant threat to the project.
- Red zone contains very high risk (critical); need to reduce risk to the yellow zone.

The use of these zones improve the transparency of the results of the matrix through providing a distinct distribution in relation to the steps that need to be taken in the future.

# The Formation of the Risk Register

The risk register shown is a result of the analysis, planning and monitoring of risks. The analysis of risks has been represented by the assignment of a severity rating and a likelihood of occurrence rating, which when multiplied together produces an overall risk rating that can be used to easily identify the most critical risks. Risk planning is represented in the risk register in the form of mitigation strategies and contingency plans, where the impact of both on a particular risk is then shown in the column entitled ‘adjusted overall risk rating’.  In terms of a process for reporting on the status of risks, the ‘status’ and ‘Last review date’ columns are to be used as a form of risk monitoring, reducing the likelihood of neglecting the risk, and allowing us to re-access risks with all prior knowledge readily available. Our risk ownership strategy involves an owner for each risk being appointed, as shown in the register. Each risk owner is responsible for discussing mitigation and contingency strategies for their respective risks with other team members, during each weekly scrum meeting.

| ID | Risk name | Risk Description | Severity | Likelihood | Overall risk rating | Mitigation strategy | Contingent action | Ownership | Adjusted severity | Adjusted likelihood | Adjusted overall risk | Status | Last review date | Amended? |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TR_1 | Poorly defined requirements | Requirements are ambiguous and open to interpretation; potential miscommunicatio n between team members during meetings. | 4 | 3 | H/12 | The use agile of methods to address the issue of volatile requirements, and to facilitate frequent communication with stakeholders. | Shaper, evaluator and plant should establish new or re-establish old requirements in a from that is detailed and complete, can be tested and that is agreed to by all team members, subject to stakeholder approval. | Lillian Coultas | 3 | 1 | L/3 | Open | January 2020 | N |
| TR_2 | Requirements and developed functions do not match | Incorrect implementations from the defined requirements. Potential miscommunicatio n and overlooked. | 5 | 3 | U/15 | Unit test should be written and used over the course of implementation to prevent an erroneous outcome at the end. Implementor should make sure that the development passes unit tests, written by evaluator. | Identify missed requirements and develop into existing functions. | Cheuk Wang Wu | 2 | 3 | M/6 | Open | December 2019 | N |
| TR_3 | Inadequate design | The system design and architecture might be inefficient, non sustainable and not reliable. Performance issues, delays may occur preventing a smooth launch and issues will arise post release. | 4 | 3 | H/12 | Throughout the design process, the proposal should preferably be passed onto senior developers, in this case academic staff for review. Group discussions should also be held to find out potential issues. | Identify problematic design and the related functions. Finalise on an improved design. Develop/ improve existing function accordingly in a timely manner to minimize impact/ delay. | Muaz Atif | 4 | 2 | H/8 | Open | December 2019 | N |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TR_4 | Libraries of poor quality | Open source library chosen has various vulnerabilities and issues, potentially causing errors and delays for the final project. | 3 | 2 | M/6 | Dedicate a session for library selection, after defining the requirement. Testimonies, user feedback/ issues, and the frequency of updates should be looked at before selection and implementation. | Identify libraries of poor quality, search for good replacement and reimplement. Reduce the impact of the poor quality libraries caused. | Muaz Atif | 2 | 1 | L/2 | Closed | January 2020 | N |
| TR_5 | General lack of technical knowledge | Lack of knowledge including but not limited to: IDE, version control, programming language, data structure. | 4 | 3 | H/12 | Every team member should research and get familiar with the tools and development environment before implementation, to ensure a smooth, efficient development cycle. | Shaper should review the impact caused, then update the role assignment for the related sections to prevent the probability of future risks. | George Lesbirel | 2 | 2 | M/4 | Open | December 2019 | N |
| TR_6 | Scaling errors affecting accessibility | Incorrect scaling of the map could result in overlap, rendering certain buttons inaccessible or causing two events to be triggered when one button is pressed, due to overlapping clickable regions. | 4 | 3 | H/12 | Investigate common computer monitor resolutions, and make sure that those has been tested to ensure that the game works said resolutions. Any error shown in test should be fixed promptly to prevent the risk from happening during deployment. | Lock the game screen to a fixed resolution, and deploy after ensuring that it is working as a temporary solution, whilst the development team will be changing and adapting the game to all common computer resolutions. | Muaz Atif | 3 | 2 | M/6 | Open | January 2020 | Y (new) |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TR-7 | Large number of entities causing performance issues | If number of entities is too great FPS will drop, therefore negatively impacting the players engagement. Decreasing the resolution to counter this effect will result in a decrease in player satisfaction. | 4 | 3 | H/12 | Restrict the number of entities to a precomputed limit to avoid performance degration. Hence the resolution doesn't have to be downscaled. | Implement an entity limit as soon as possible, and recommend stakeholders to use more powerful machines when running the game at the meantime. | Muaz Atif | 2 | 1 | L/3 | Open | January 2020 | Y (new) |
| OR_1 | Poor team dynamics | Miscommunications that affect team productivity and coordination on project tasks.Conflict of opinion about the direction of the project. | 3 | 2 | M/6 | Use interactive team group sessions to identify and manage issues; the shaper should act as a facilitator to resolve the issues affecting the team.<br><br>Produce a communication plan detailing the frequency, objective, and recipients of each communication. | If issues persist then begin processes to escalate the conflict resolution to higher management with an assessment of the issues and team members involved. Shaper should be in charge to maintain a good team relationship. | Lilian Coultas | 2 | 1 | L/2 | Closed | December 2019 | N |
| OR_2 | Specification delays | Specification of the system hasn't been produced or ammended in time, causing delay for implementers, may delay the project completion. | 4 | 2 | H/8 | Team coordinator should actively follow up on the progress of specification every week during scrum meeting, and investigate the reason behind any potential disruption. | Arrange an emergency meeting, place specification as top priority. Unimportant specs may be neglected, to make up for lost times. | Cheuk Wang Wu | 3 | 1 | L/3 | Open | January 2020 | N |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| OR_3 | Scope creep | Continuously changing requirements that eventually results in an overall increase in the number of requirements over the project's life cycle - can result in inability to keep up with the number of requirements. | 4 | 5 | U/20 | Employ the use of short iteration cycles through the implementation of agile methods, as a way of handling volatile requirements. | Document every example of scope creep in a logbook and get approval from the owner of each new requirement or change before starting work. Restrict excessive changes or additions with good reasoning after the development of the game is underway. | Lauren Quarshie | 3 | 3 | H/9 | Open | January 2020 | N |
| OR_4 | Productivity issues | At the start of the project we may take it slowly when developing, spending a lot of time on some more simple tasks. However, when it comes to the end of the project we may have to rush more significant parts. | 4 | 4 | U/16 | The Gantt chart will be made, with tolerance in mind. The shaper should make sure that people are doing what their allocated tasks in the correct time frames during weekly scrum meeting. This can avoid any part of the project being rushed. | Shaper and Co-ordinator should work together, to identify the reason of reduced productivity. An updated timeline should be set based on circumstances, and every team member should follow the updated responsibility and schedule. | George Lesbirel | 2 | 2 | M/4 | Open | January 2020 | N |
| OR_5 | Constant addition of features | Throughout the project we may find multiple features that we want to add. We must ensure we don't add too many so it doesn't become too hard to manage and take too much time away from the core parts of the project. | 3 | 3 | H/9 | Implementor should work with evaluators during weekly meeting, to keep the specifications/design realistic, and to mention that the impact that constant addition of features will cause. | Evaluator should immediate evaluate the current features, and prioritize features that are critical to the success of the system. No new features should be added until the project resume normal progress. | Lauren Quarshie | 2 | 2 | M/4 | Open | January 2020 | N |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SR_1 | Overly optimistic schedule/schedule is not understood by team members | Various factors have not been taken into account and underestimating the time needed, therefore causing a potential delay, and hence a risk of missing the deadline. | 5 | 4 | U/20 | Introduce weekly scheduling workshops for the whole team to ensure the schedule is understood, therefore reducing the probability of missed tasks. | Co-ordinator should reassess the progress, then present an updated schedule with each team member and thoroughly explain upcoming tasks at each weekly scrum meeting. Stakeholders should be informed of the potential delay. | Jonathan Grout | 3 | 2 | M/6 | Open | December 2019 | N |
| SR_2 | Gold plating | We may spend too much time focusing on areas of the game that aren't of the greatest importance. This can be a waste of valuable programming hours. | 3 | 3 | H/9 | Shaper and Co-ordinator should prioritise task at the start of the project, and review team's progress weekly to prevent gold plating from happening. Evaluator opinion is also crucial on deciding the importance of tasks. | Shaper and Co-ordinator should immediately prioritise crucial areas of development, with the help of evaluator, and assign sections to team members to prevent any further delay already caused, so that the project can be successfully finished. | Jonathan Grout | 2 | 2 | M/4 | Open | January 2020 | N |
| UNR_1 | Stakeholder actions delays the project | Potentially related to scope creep. Communication/ feedback from stakeholders can potentially been delayed, due to the amount of software development they are involved in, hence disrupting our implementation timeline. | 3 | 1 | L/3 | Arrange meetings in advance with stakeholders, preferably as soon as possible. Rough timeline should be given, and impact on delayed communication and scope creep will be explained. Stakeholders will now have an understanding towards potential impacts of various actions, and communication has been established effectively. | Immediately arrange to meet stakeholders, and let them know that delays will be expected due to us not being able to obtain necessary information from them. With scope creep, an explanation of the impact will be sent to them. Make sure they are aware of the extra time, resources and cost that it might induce. | Lauren Quarshie | 3 | 1 | L/3 | Closed | December 2019 | N |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| UNR_2 | Collaboration platform becomes unavailable | Delays to the project schedule due to 3rd party platforms being used becoming unavailable. | 3 | 2 | M/6 | An offline backup should always be kept, in case of any downtime or data lost of 3rd party cloud platform, e.g. GitHub. This can be done by any team members. | As we are unable to access our project files, we should move onto the next item on gantt chart that doesn't requires the missing resources. | Cheuk Wang Wu | 1 | 2 | L/2 | Open | January 2020 | N |
| UNR_3 | Corruption of files affecting development of the game | The potential corruption of files during processes such as importation. This can lead to scheduling delays and delays in the development of the system as a whole. | 4 | 3 | H/12 | Use short and clear file names that follow file-naming conventions i.e. no symbols or special characters.<br><br>In addtion to saving work constantly, create multiple versions of the same files on a daily or weeking basis. Previous versions can be deleted once the project is submitted. | Make use of the system tools available that may be able to resolve the problem. Run anti-virus software to locate the source of the problem and resolve it.<br><br>Check for a working version of the corrupted file in the computer's temp files. | George Lesbirel | 4 | 2 | H/8 | Open | January 2020 | Y (new) |
| UNR_4 | Documentation of the new project chosen is difficult to understand or it is incomplete. | No complete explanation for certain tools used in the program or the program itself can also result in scheduling errors as time must be taken to research the program and the tools associated with it | 3 | 3 | H/9 | Ensure the project we want to use has suffient documentation before making the decision. This can be done by asking questions during the presentation of the project, or asking the developers themselves. | The coordinator and shaper should implement a revised schedule after reading through the documentation and take time for researching tools for the project into account. | Lillian Coultas | 2 | 3 | M/6 | Open | January 2020 | Y (new) |