Risk assessment

- Corruptions in data

This is a high impact risk which has a low to medium likelihood of happening. If this occurs, the project may be buggy or completely unoperational. Avoiding this means shutting down applications correctly and avoiding code that has the potential to crash the system. The latter is not always possible which is important to regularly push the files to GitHub so that they may be retrieved later.

- Sickness or unexpected circumstances that can make a big difference when the project duration is so short.

This is a high impact risk with a low likelihood of happening to myself personally (as I do not suffer from any debilitating diseases). If this occurs, I may not be able to meet the deadline as every day for this week is full of tasks. One day of not doing any tasks is enough to jeopardise my ability to meet the deadline and/or the quality of my work. I cannot avoid getting sick however I can avoid activities that have a potential for hospitalisation, as well as not getting drunk. I have stuck to the MVP plan to account for setbacks that may occur.

- Merging the wrong files on GitHub

This can be a medium impact risk depending on what code I currently have access to. There is a low chance of this happening as merging is a multi-step process and gives me enough time to confirm the request. If this does happen, I must make sure I promptly back up the project existing on my computer into a new feature branch.

- Equipment failure

This is a high impact risk with low to medium likelihood of happening. The effects of this will include a significant loss of time due to losing important documents and applications required to facilitate the project. It may result in a loss of data; the quantity of which is entirely dependent on when I last pushed my files to the server. I cannot prevent equipment failure however I can implement practices such as monitoring system temperatures through HWMonitor. If this happens, I can resume my work on my laptop where I have all the necessary features and software however this will take an afternoon to reconfigure.

- System vulnerability to attack

This is a medium impact risk with a low likelihood of happening. Viruses and malware can result in loss of data, slowing down in computer processes and loss of personal information. This can be remedied by a good antivirus software, scanning files before executing them and ensuring that those files are from trusted sources.

- Running out of time

Running out of time is a high impact risk with a medium likelihood of happening. In such a short time scale, there is limited time to recover from setbacks. If it is just documentation that has not been completed, this is not a high impact risk as this does not take a long time to do relative to the scope of the project Running into problems, diagnosing them and fixing them is a longer process for someone inexperienced. Because of this, I have not overstretched my project in order to leave time for setbacks.

- Not being able to achieve 80% tests

This is a medium likelihood risk and the impact could be negligible or severe. If tests are not completed and would the tests have revealed issues, this would cause problems when the project is deployed. To mitigate the chances of running out of time for tests, I will leave plenty of time for tests.

- Missing dependencies

This is a high impact risk and the likelihood is low to medium. As many functions would not work without the dependencies, testing would indicate the problem. Dependencies are easily found online and can solve many issues. Inputting incorrect independencies can cause problems down the line and may result in not being able to deploy the product. To mitigate this I will follow materials online and seek guidance in case of issues.