# Risk Assessment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk | Probability | Impact | Before - Monitoring | After – Risk Response |
| Application Crashes | Low – Unity is fairly stable | High – Cannot use the application | Try/catch blocks, and logs of the crashes | Review crash logs and try to fix the main issues |
| Hardware (VR) failure | Low – Oculus Quest is hardly used and in great condition | High – Cannot use the application | Check for any signs of wear, and be aware of any common bugs with the oculus | Report the incident to supervisor and try to get a replacement |
| Screen Recording is Corrupted (fails) | Low – Screen recording software on the oculus is stable | Medium – No evidence for user test happening, however I would still have the output files for the user’s results | Be aware of any screen recording bugs in the oculus quest | Restart the recording (try again) |
| Results file output corrupts (fails) | Low – No reason for this to happen if properly coded and tested | Medium – Still have the screen recording which will display each scene, driving variable, and results | Try/catch blocks for creating files and logs of any failures | Check the code for any major errors (reference the error logs) |
| Main threat triggers too early | Medium – There is a lot of timing involved in simulating each scenario, hence it’s possible for triggers to go off before they should | High – Results will not be accurate or reliable | Test each scene and log the start time of each trigger | Review the timings of the triggers and approach times of the car and make changes to fix any timing issues |
| Objects (Cars, pedestrians) do not function correctly | Medium – There will be a lot of variables, hence if not properly coded (or errors not properly accounted for) then objects may not function correctly | Medium – This depends on the object and the severity of the malfunction (could be a small visual glitch, or could be complete failure of functionality) | Do lots of testing of the scenes and the individual object classes, monitor any possible hiccups | Note down the issue (For example: pedestrian walking backwards) and check the object’s code |
|  |  |  |  |  |