

//	Damage	Negligible(1)	Minor(2)	Significant(3)	Major(4)	Catastrophic(5)
Likelihood	//					
Very Likely(5)		5	6	7	8	9
Likely(4)		4	5	6	7	8
Possible(3)		3	4	5	6	7
Unlikely(2)		2	3	4	5	6
Improbable(1)		1	2	3	4	5

Risk	Description	Potential Damage	Likelihood	Damage	Factor	Mitigating Actions
Testing shows minor errors	Testing results in minor parts of the program needing rewritten.	Lost time	Very Likely	Negligible	5	Test often and code according to the design document
Procrastination	Time spent not working on the project whilst the project is not yet complete.	Lost time	Very Likely	Negligible	5	Proper timekeeping to ensure all work is completed by it's deadline.
Minor illness/injury	Minor illness without time off, but affects performance, I.E. time spent dealing with symptoms.	Lost time	Unlikely	Negligible	2	Keep suitable medicine on hand.
Bad code	Time spent on code that is rewritten or scrapped.	Lost time	Likely	Minor	5	Code according to the design document. Properly plan the logic needed for the code before writing it.
Loss of internet access	Access to the internet is lost and lasts for an extended period of time.	Lost time	Unlikely	Minor	3	Ensure that a copy of all work is available offline when able to prevent loss of internet halting production
Power outage, minor damage	Power outage resulting in lost time and progress.	Lost progress and time	Unlikely	Minor	3	Save often, Surge protected power supply.
Git down for maintenance	Git repository server downtime	Lost time and potential loss of progress.	Improbable	Significant	3	Ensure that local code is up to date to minimise effect of downtime.
Serious illness/injury	Serious illness/injury resulting in hospital time	Significant loss of time	Improbable	Major	4	Do not take risks with health and safety. Avoid

						major health risks.
PC damage	PC damage rendering it unusable	Significant loss of time and progress.	Improbable	Catastrophic	5	Antivirus software, place PC in a safe location. Surge protected power supply.
Feature Creep	Additional features for the program are thought up and implemented before documentation is created for the minimum viable product.	Significant loss of time, testing and documentation fall behind.	Likely	Significant	7	Ensure the entire specification is completed with all deliverables ready before any additional functionality is attempted.