Description	Evaluation	Likelihood	Impact Level	Responsibility	Response	Control Measures
Internet outage	Lose access to repository and online resources	Medium	Medium	Internet Provider	Use a mobile hotspot if possible for any crucial access	Ensure local copy is kept up to date with repository
Github outage	Lose access to repository	Low	Medium	Github/Microsoft	Work on local copy and push when possible again	Ensure local copy is kept up to date with repository
Power outage	Lose any changes made since last save and any changes since last push until power is restored	Very Low	High	Electricity supplier	Work on laptop from repository if battery charged	Ensure any work is saved, commited and pushed regularly
Computer crash	Lose any unsaved work and if cannot be restarted then lose any work not pushed to repository	Low	High	Myself and Supplier of equipment	Attempt to get the PC up and running again and continue from where possible. If not possible then use a possible alternative	Ensure work is pushed to repository regularly and ensure equipment is looked after and kept up to date
Broken version pushed to repository	Application may not run until problem is resolved or the repository is rolled back	High	Medium	Myself/Developer	Roll the version back until the problem is resolved	Keep to a main/dev/feature model with regular commits
Run out of time	Features missing or MVP not met	Medium	High	Myself	Non possible	Good use of a Jira board to effectively prioritise and plan the project to help effective time management
User attempts SQL injection	If successful the user could have unwanted access or control over the database	Low	Very High	Myself	Assess and repair the damage done	Use prepared statements when user input is taken into SQL
Major issue in one of the classes	Exceptions/errors caused by a certain class which stops the application running	High	Medium	Myself	Roll the version back and try to find the root of the issue	Keep classes simple and separate where possible and use unit tests