Risk Assessment

Alysopedia

Before development begins, the risks need to be analysed and understood. It is important to know the risks as soon as possible so that they may be managed and avoided.

Created: 09/07/2020

Matrix

Each risk has a statement to specify what exactly the issue is, a response taken to manage and avoid it, and an objective to elaborate on the purpose of the response in relation to the objective.

These risks are graded on three factors:

- Likelihood - How likely is the risk to come to fruition and how difficult is it to avoid completely.
- Impact - How much will it affect the project
- Risk Level - How important is it to avoid, taking the value of likelihood and multiplying it by the value of impact.

		Likelihood				
		1): Rare	2): Unlikely	3): Possible	4): Likely	5): Almost Certain
Impact	5): Catastrophic	5): Moderate	10): High	15): Extreme	20): Extreme	25): Extreme
	4): Severe	4): Moderate	10): High	12): High	16): Extreme	20): Extreme
	3): Moderate	3): Low	6): Moderate	9): High	12): High	15): Extreme
	2): Minor	2): Low	4): Moderate	6): Moderate	8): High	10): High
	1): Negligible	1): Low	2): Low	3): Low	4): Moderate	5): Moderate

Risks	Statement	Response	Objectives	Likelihood	Impact	Risk Level
Server Issues **#New**	The application is set to run on server.port 8181, which may cause conflict related issues down the line. It also makes testing more awkward, increasing the chance of failures slipping through the cracks.	Find a solution to increase the server ports the project can use, namely for the tests. Until then, be careful with running the selenium projects	Solve the issue before it detriments the project.	5	2	High
Pre-population **#New**	The skills table within the database pre-populates each time the application is run. This is fine if the database is set to delete the contents each run, but this won't be the case with public releases. If this isn't resolved, the skills table could end up with countless duplicate skills entries.	Set the database to delete at the beginning of each run until a solution is found.	Prevent the database from being filled with duplicate entries loaded into the application each run.	2	2	Low
Coronavirus	With coronavirus growing exponentially, it's possible the project member(s) may catch it before the deadline. If not asymptomatic, a best case scenario would be being unable to present the project to peers. A worst case scenario would be death or permanent lung scarring.	Follow government lockdown procedures and avoid leaving home unless absolutely necessary.	Reduce the chance of catching coronavirus.	2	5	High
Repository Breach of Security	Source code pushed to GitHub may be vulnerable to malicious hackers.	Avoid non-unique passwords (a password manager is recommended), update them regularly and avoid cliches i.e. 'qwertyuiop'	Prevent malicious hackers from obtaining and/or altering data.	1	4	Moderate
GitHub Human Error	Any source code added to the repository via **feature** branch could unintentionally be merged into the **master** branch, leading to features	Utilise a development branch and double check to make sure that's the branch the features are being merged	Reduce unused data and prevent unfinished features from entering the latest working	1	1	Low

	being added before being tested.	into.	version of the project.			
Poor Planning	A common reason why projects fail is because people neglect the planning phase. Planning is the foundation of every successful project, it being required to know what the project member(s) need to be doing and what they are contributing to.	Plan the project accordingly. Use Kanban boards and agile working to allow for continuous planning.	Follow a variably plan to completion	3	5	Extreme
Underestimating Scope	Not keeping to the scope of the project could lead to project member(s) overestimating what they can accomplish. The result could be not meeting the deadline, possibly with an assortment of unfinished features needing to be implemented.	Keep to the scope and update as more of the things which need to be done are completed.	Meet the deadline with the project specification fulfilled, no more/no less.	3	4	High
Computer Failure	The project absolutely requires a computer to be completed. Without a computer, very little of the project can be completed.	Have a backup computer ready and use GitHub to gain a working copy of the project.	Always have a computer to complete the task with.	1	5	Moderate
Internet Failure	While some of the project can be completed without internet access (Java, documentation, etc.), there are parts which absolutely require an internet connection, such as GitHub and the presentation phase of the project. This is especially detrimental with the current coronavirus pandemic as technicians may not be readily available to resolve the issue.	Have an alternate location with internet access prepared in the event of network failure.	Always have the internet available.	2	5	High
Natural Disaster	The project is being developed remotely from the Isle of Wight. While there's no notable threat of natural disaster (i.e. Earthquake) in this location, the	Monitor the local news and move to an alternate location if this occurrence seems probable.	Prevent a natural disaster from impacting the project.	1	1	Low

	improbable shouldn't be ruled out. On February 13th 2020, a tornado in East Cowes made headlines as it injured a middle-aged man via hurled object. If this happened again and caused damage to property, it could affect the development of the project.					
Deep Vein Thrombosis	Sitting too long has been known to result in blood clots forming in the leg, known as Deep Vein Thrombosis. Symptoms of this include: shortness of breath, chest pain and death.	Take occasional breaks to move around. Keep active outside of working hours.	Prevent a life-threatening condition from affecting the scope of the project.	1	5	Moderate