

Ref	Risk Description	Cause	Risk Event	Likelihood (1-5)	Impact (1-5)	Risk Rating	Action
1	Broken Repository	Pushing broken code to the Git hub	Issue rolling back to previous version	3	3	9	Ensure all code compiles and works as expected before pushing to Git hub.
2	Hardware failure	Old computer	Stops a team member working on the code	2	5	10	Ensure to keep computer off when not in use, be careful around it as to not spill drinks etc
3	Motivation/ mental health	Burnout, worry from Covid-19 effecting family etc.	Assignment stalls if family member is affected by COVID-19, or not completed to best standards due to burnout / complacency	3	4	12	Keep sprints as simple as possible, do best not to worry about things outside of your control and ensure you follow sprints as set out on the management board. Do not get stressed when issues arrive while coding.
4	Lack of time	Not using time correctly	Project only gets finished to MVP rather than having could have would have too.	3	5	15	Split tasks to sub tasks on Kanban, ensure you delegate enough time for each task, ensure MVP is the priority and work on extras later.
5	Not understanding technologies used completely	Having issues with some of the technologies required	Slows project completion and may not allow for a full finished project that meets specification.	3	5	15	Refer back to previous exercises when needed, always refer to specification, check Kanban board and most importantly, ask for help when needed.
6	Internet issues	Internet going down	Stops uploading to github, access to tutorials and QA-community slows the ability to merge to developer for the team	2	2	4	Download required tutorials/files where needed so they do not require internet access, push to remote when internet is stable.
7	Server/hosting issues	Server not working correctly	Doesn't allow testing to pass, halts development	3	4	12	Ensure server is running correctly and all drivers are closed after every test.
8	Incorrect project structure	Packaging incorrect	Makes finding files / setting paths harder slows project down	2	2	4	Ensure you follow enterprise architecture model and previous projects and ensure project follows these
9	Miscommunication between members	People misunderstanding each other	People work on the same part at the same time, could also mean someone creating the wrong class or the wrong parameters for a class	2	3	8	Ensure the team all know what each other are doing, explain your current and previous day at standups clearly, constant communication is key
10	Merge conflicts	Two files being worked on at the same time	If the wrong parts of code are kept, can cause issues with the overall file structure	3	3	9	Ensure everyone knows who's working on what, communicate merge conflicts with team
11	Project schedule is not clearly defined or understood	Incorrect spring tasks/ Jira board issues	Someone works on a feature which is reliant on another that isn't implemented yet	2	4	8	Ensure Jira is kept up to date and correct, communicate tasks between team clearly
12	Scope creep	Added features or requirements added	Extends time needed to publish final product	2	3	6	Ensure team works towards MVP at all times, with added features only added when full functionality complete
13	Unreliable estimates	Team think certain tasks will take less time than they do	Project time extends, miss MVP	2	5	10	Ensure estimates are short and give enough time to finish all story points
14	Lack of team planning sessions	Don't have stand ups everyday	Team gets lost on whos doing what, project stalls/	2	4	8	Ensure stand ups are completed everyday and sprint meetings every week.

releases with
missing features