

Risk	Risk Statement	Response strategy	Objectives	Likelihood	Impact	Risk Level
GitHub	Any source code pushed to GitHub could potentially contain information that hackers would find useful when trying to maliciously alter the project. The source files could potentially contain hard-coded login credentials which could allow for data leaks.	Use stronger passwords and usernames than just "admin" or "root" and keep them regularly updated.	Reduce the likelihood of hacking and data leaks.	Low 4	High 11	High 15
Java code credentials	Hard coded credentials and passwords could be stolen if the code gets accessed by others.	All the credentials and passwords should be inputted by the user through a provided panel.	Reduce the risk of data leaks.	Low 4	High 10	High 14
Quality of code	Using bad practices could lead to a bad quality of code making it unreadable for other people that want to look at the code.	Make sure to avoid bad practices by utilising SOLID principles.	Reduce the complexity of code and make it more readable.	Medium 6	Low 3	Medium 9
Time management	Mismanaged time could lead to unfinished code.	Make sure to start coding early.	Improve time management.	Low 1	Low 2	Low 3
Understanding of JavaScript functionality	Not understanding JavaScript could lead to errors arising while linking the user interface to the database.	Make sure to ask for advice if there is a problem that is hard to solve.	Reduce the likelihood of losing time based on misunderstood code.	Medium 6	Low 2	Medium 8
Hardware	Hardware failure could lead to loss of work and time.	Make sure to make frequent pushes to GitHub to have a safe backup of data.	Reduce the risk for loss of data.	Low 1	Medium 6	Medium 7

Scope variation	Adding new features during the project could shift the scope too much and end up not having enough time to meet the minimum viable product.	Stick to the initial plan of the project until the minimum viable product has been reached.	Reduce the risk of not meeting a minimum viable product.	Low 2	Low 2	Low 4
Understanding of spring functionality	Not fully understanding spring code could lead to strange errors that could take more time than planned to resolve.	Ask for advice from team members or the trainers if there is a problem or a piece of code that is very hard to understand.	Reduce the risk of losing time through bugs.	Medium 4	Low 2	Medium 6

Risk Matrix

	Severity							
Likelihood	Low 0-2	Low 3-4	Low 5	Medium 6-7	Medium 8-9	Medium 10	High 11+	
High 11+								
Medium 6-10	Understanding of spring functionality	Quality of code						
	Understanding of JavaScript functionality							
Low 0-5	Time management			Hardware		Java code credentials	GitHub	
	Understanding of spring functionality							
	Scope variation							