Risk Register

The purpose of this document is to discuss about possible risks that the team would encounter during the project period. For each possible risk, one or more possible solutions will be proposed to minimise or ameliorate the risk.

Possible Risks	**Proposed Solution**
Team Dynamic	
1. Time management As a 6-man-team is relatively large and every team member has different schedules, it is difficult to have everyone to attend every meeting.	Using real-time online collaboration tools such as Google Docs and Github will help the team stay on the same page and update each team member of the current progress/latest changes. In the event of absence of team member during a meeting. Skype calls can be set up. Documents such as meeting minutes will be uploaded and shared amongst members
2. Communication Good communication with clients, mentors as well as within the team is a crucial component in producing a successful project. Since client and mentor meetings only occur every two to three weeks, there may not be immediate responses to questions. Additionally, issues that the team may encounter might not be resolved in time without the assistance of the group mentor.	More meeting sessions can be setup by appointment with client and mentor upon request.
3. Skill level Every team member has their own background knowledge. It is hard to collaborate with each other under such circumstances.	To accommodate and mitigate risks that may arise from learning new languages and frameworks, we will be splitting off into mini teams. This will result in less context switching and allow team members to help each other out for specific sections of the web application.
Quality check Lacking unit, integration and end-to-end testing.	To keep everyone's work up expectations during each stage. Cross-check system can be implemented to point out possible flaws and improvements can be made. Taking the time to implement unit, integration and end-to-end testing.
Time/ Skill Constraint Total of 60 hours are expected to be assigned towards the project. There is a possibility that certain functions of the web application can’t be achieved/or too costly to achieve.	Compromises are inevitable during the development of a project. The best possible way to resolve this is Minimum Viable Product. Using the Minimum Viable Product model, unnecessary functions can be sacrificed as long as the primary objective of the project has been achieved.