|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 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 a 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. Avoid putting hard-coded usernames and passwords on github if possible. | Reduce the likelihood of hacking and data leaks. | Medium | High | Medium-to-High |
| Hardware failure | If one of our PCs stops working the person won’t be able to work on the project. | If possible fix quickly, if not try to get another machine from somewhere | Ensure that no/minimal time is wasted due to not having access to the projects | Low | medium | Low-to-medium |
| Hot weather | The hot weather may lead to dehydration and make it more difficult to concentrate, leading to reduced productivity. | Drink liquids frequently, use desk fan if very hot | Remain comfortable to enable productivity | Medium | low | Low to medium |
| Loss of data | If a session working on the project is interrupted unsaved work may be lost | Work on cloud services where possible, save work frequently and push to github | Ensure that no tasks have to be repeated due loss of saved work | Low | High | Medium |
| Inconsistent internet connection | If internet is disrupted it will disrupt team communication, affecting cohesion and potentially leading to problems | Use personal hotspots on phones if needed, try to get any issue sorted quickly | Make sure our communications and cohesion are maintained | Low | Low | Low |
| Testing oversight | Insufficient user-end testing could lead to missing an important bug by focusing too much on other things | Test as many aspects of service as I can, look for volunteer system testers to make sure all aspects are covered | Make sure that there are no bugs in the final product that substantially inhibit functionality | high | medium | Med-to-high |
| Running short of time | As we approach the deadline it may be difficult to achieve everything we want to in the time allowed. | Make sure Jira board covers everything the project might involve so we know exactly what is left as all times, prioritise tasks which are crucial to functionality. | Get everything that we need to get done completed on time to help the final outcome | high | high | high |
| Files in wrong directory | If I accidentally save an important file for the project in the wrong file it will not be pushed to my github repo | Try to save things in the correct place first time, every week check contents of project folder against specification checklist | Ensure that all files are in the correct place so that they will be pushed correctly. | medium | medium | Medium |
| Git merge conflicts | If more than one of us change something in our own branches it will lead to conflict issues | Use proper protection rules to keep our master branch secure, properly allocate tasks so that such issues should not arise, pull and merge from main dev branch frequently | Make sure each of us does not do something which will screw up what someone else has done | Medium | Medium | medium |
| Application breaking error | One of us might push something which somehow breaks the application | Always push to development branch first so that master is protected, test code thoroughly on our own machines before pushing | Make sure no one breaks the whole application so that we don’t have to spend ages fixing it | Low | Low | low |