|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Description of Risk | Risk Impact | Risk Likelihood | Responsibility | Evaluation | Proposed Mitigation | Response |
| Database is attacked and data is breached | High | Medium | Developer | If the URI is exploited or exposed. The database could be compromised, allowing the attacker to change, delete, steal data etc. | Regularly update/back-up database to restore data to deal with such an event. | All accounts to have password. Stopped using root user privilege. Drop test database. |
| Upload sensitive information to GitHub | Low | High | Developer | Could provide access to database or other parts of the application | Ensure not to push sensitive and/or personal data to GitHub. | Changed sensitive data and replaced with environment variables where you can. |
| Incorrect data input by user | Medium | Medium | Developer/User | Incorrect data on the application ruins it credibility and reliability | Test if the right data has been entered into the application. | Perform validation checks(e.g. no blank mandatory fields) to increase accuracy and unwanted data redundancy. |
| Natural disaster | Low | Low | Cloud service provider | Natural disaster could cause even a single region to fail causing the app to not function as required for users. | Conduct prior disaster recovery plan to run application in another region if there was to be loss of primary zonal resources | Risk still requires a response. |
| Cross-Site Request Forgery | Low | Low | Developer | Unauthorised access, GDPR breach, | Implement Challenge response mechanism to verify identity of user. E.g. 2-factor authentication. | Risk still requires a response |