OWASP

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Code | Category | Likelihood | Impact | Risk | Actions Possible | Planned |
| A01 | Broken Access Control | HIGH | SEVERE | HIGH | Implement check to see if the user login before giving the access. | YES |
| A02 | Cryptographic Failures | LOW | LOW | LOW | Nothing, fixed | YES |
| A03 | Injection | LOW | LOW | LOW | Nothing, fixed | YES |
| A04 | Insecure Design |  |  |  |  |  |
| A05 | Security Misconfiguration |  |  |  |  |  |
| A06 | Vulnerable and Outdated Components |  |  |  |  |  |
| A07 | Identification and Authentication Failures |  |  |  |  |  |
| A08 | Software and Data Integrity Failures |  |  |  |  |  |
| A09 | Security Logging and Monitoring Failures |  |  |  |  |  |
| A10 | Server-Side Request Forgery |  |  |  |  |  |

**Reason**

* *A01 Broken Access Control* –
* *A02 Cryptographic Failures* – I use bcrypt for my application. Bcrypt is one of the strong adaptive and salted hashing functions with a work factor.
* *A03 Injection* – I use hibernate ORM for my application.
* *A04 Insecure Design* -
* *A05 Security Misconfiguration* -
* *A06 Vulnerable and Outdated Components* -
* *A07 Identification and Authentication Failures*-
* *A01 Broken Access Control* -