**OWASP Report**

1. Injection-In The Java application is applied JPA, which is Object Relational Mapping tool (ORM), which prevents the SQL injection and untrusted data can not be sent to interpreter as part of query.
2. Broken Authentication- In the React application is applied regex for the username and password, in order to prevent automated credentials. And the application doesn’t have any default credentials for the admin and the users.
3. Sensitive Data Exposure- The Java and the React application are passing Json Web Token between them for the authentication and also the passwords, which are stored in the database are encrypted.
4. XML External Entities- This is an attack for applications, which parse XML input and the Java application is not vulnerable to this security risk.
5. Broken Access Control- The permissions for the managers and for the users are different. In the Java application is applied preauthorization for every controller it has. The JWT token is extracted in the application to see weather the user is manager or common client.
6. Security Misconfiguration- This risk is prevented by properly configuring the application security.
7. Cross- Site Scripting- React is framework, which prevents this risk automatically. In the Spring boot application is added restriction, which allows only the React application to make HTTP requests to it.
8. Insecure Deserialization- The project doesn’t have any deserialization.
9. Using Components with known vulnerabilities- All unused dependencies, components, files are removed. It has a plugin, which makes sure that unused libraries are removed.
10. Insufficient Logging and Monitoring- The project do not log any failed logins in it.