**OWASP Report**

1. Injection

* The application is generally safe against injection because the incoming data goes through JPArepository and not directly in the SQL query.
* Low Risk

1. Broken Authentication

* The application uses JWT authentication.
* Potential risk. Only password validation check is password length.
* How to fix: implement weak password checking according to the list of [top worst passwords](https://github.com/danielmiessler/SecLists/tree/master/Passwords).

1. Sensitive data

* Not applicable. The application database does not store user sensitive data.
* No Risk

1. XML External entities

* Not applicable

1. Broken Access Control

* The application has CORS configuration enabled.
* No POST, PUT or DELETE is allowed for non-authorized users.
* Users cannot access or modify other users data.
* No usage of url parameters on authentication.
* Low risk

1. Security Misconfiguration

* Not applicable. Application is not going to be updated regularly.

1. Cross-Site Scripting(XSS)

* ReactJS automatically escapes XSS by design.
* No Risk.

1. Insecure Deserialization

* Not Applicable

1. Components with known vulnerabilities

* Potential risk
* I am not aware of the version and stability of every component, interface or class I use in my code, although I try to use the most popular or most secure when possible.

1. Insufficient logging and monitoring

* Potential Risk.
* Not all components are tested and monitored.