# OWASP TOP10 – 2017 & PJG-PROJECT

The aim of this assignment is to try to identify any security holes we have already introduced in our system. So here is the OWASP top 10 and the implication in our project.

1. Injection

We use a MySQL database so technically SQL injection can happen. On the client-side we check the data sent by forms, it should respect a defined pattern. On the server-side, we use cookies to provide the user id. The cookie can’t be accessed by JavaScript in the browser, so it can’t be changed to try to create one and access the data.

1. Broken Authentication

For that time, we have only implemented the cookie token and the forms control.

1. Sensitive Data Exposure

In our database we store few information about our users such as email, username, level and a password. The password is, of course, stored hashed. That way even if all of the data leak, the authentication will be impossible. We also use HTTPS to encrypt every communication between the server and the client.

1. XML External Entities

To avoid that kind of attacks we don’t use XML Documents but only JSON if we need to send data.

1. Broken Access Control
2. Security Misconfiguration
3. Cross-Site Scripting
4. Insecure Deserialization

We don’t use any kind of deserialization. Sometimes we send JSON data from the server to the client.

1. Using Components with known vulnerabilities

To prevent this, we try to use only the dependencies that we need for the project and not to import all of the library.

1. Insufficient logging and monitoring

For the time we just send the error message in response of the request. But at the end of the project we should create a log file on the server and log all of the error message into.