As our project consits of web application and database we might experience some attacks by crackers (a.k.a. hackers). Most common attacks that we might experience are :

* SQL injection (SQLi)
* Cross-Site Scripting (XSS)
* Unrestricted file upload
* Resource manipulation

One of the most common attacks are SQL injection which can do a lot of damage to sensitive and overall data, in case if it’s being stolen or deleted from database. All the cracker does is enters SQL code in user name or password fields and the code is executed when the variable is passed. To prevent that we should use prepared statements and make sure that when the information is passed from the client to server it’s not directly executed.

Another threat is Cross-Site Scripting, which might affect current application. It works on the same principles as the SQL injection but mostly affects only the client. The attacker might send some malicious script that can access any cookies, session tokens, or other sensitive information retained by the browser and used with that site. These scripts can even rewrite the content of the HTML page. This mostly could affect other users. The same as SQL injection we must make sure that all the information passed in the login fields are checked before executed.

As the current system features file upload there might be threat with malicious files that is being uploaded to the server and might affect users who are opening and reading those files. This should be handled by server side security software which monitors the incoming traffic and identifies malicious threats.

Resource manipulation contains attacks like forced browsing and path traversal. These attacks does not require a lot of hacking knowledge and can be made by average user. It might be dangerous and can get access to sensitive data. Just a manual input or editing address in address bar can cause a lot of problems. To prevent these kind of attacks it is important that access-control settings are accurate and up to date for every page and application on the site, it is important that testing occurs prior to launch in order to verify that pages are protected.

In current project it might not be the most sensitive data that is stored on the server but still it might store some personal information and pictures of a child that studies in the school. As the data that this project is mostly concentrated on is a home works it should not be that desirable for attacks. We chose not to implement high security level for this project.