# 

**Web Platform Development 2**

**Group Report**

# Former Group K

**Group Members:**

**Maciej Tokarski, Student ID: S1630497**

**Nikos Sinikof Student ID: S1710666**

**John Giannakis Student ID:** S1713534

Table of Contents

[1](#_Toc40726822)

[1. Introduction 2](#_Toc40726824)

[2. Link Design 3](#_Toc40726825)

[3. Persistence 3](#_Toc40726826)

[4. Security 3](#_Toc40726827)

[4.1 Password Protected 3](#_Toc40726828)

[4.2 Web Application Firewall (WAF) 4](#_Toc40726829)

[4.3 Session Token 4](#_Toc40726830)

[4.4 Vulnerabilities Prioritization 4](#_Toc40726831)

[5. Documentation of System Testing. 5](#_Toc40726832)

[6. Conclusion 7](#_Toc40726833)

# Introduction

A web-based application was asked to be created by the team which give the ability to the user to add all of their current coursework projects, modify or remove them. Besides that, an intended due date along with the actual completion date can be added for each coursework project. Also, the user will have the ability to define their own milestones along with a list of the incomplete courseworks they might have. The user can provide all the information they added into the system by sending a link and the other person can view the user’s project. This group report will explain what has been accomplished by the team giving special attention on the link design, the persistence of the data, test report for our final version of the application and last but not least the security measures that have been implemented.

# Link Design

For the application it is critical for the website URLs to be understandable by the user, in order for the user to find where they are inside the website at the particular time of period, also to allow an easy passage for any future pages and functions that might be added to the system. Furthermore, it is crucial for these URLs to avoid including any personal information of the user or other contain information that the user can potentially use to gain access to another user’s exclusive information. To avoid this occurrence to happen in this application the user will have to enter information of the projects they are currently working on, the database of the application would not incorporate any personal details about the user, as the user would not want their information to be visible to any other users and outside sources, the important data will remain protected and secured. At the top of each page the title of that page is displayed, this helps the user to be informed about the page they are at the moment. Thus, for each page it is given the corresponding URL in the page the user might be in. For instance, the login page which is used for the user to input their credentials it is included in the URL the “/login” section in the middle of the URL, this will help the user to conclude in which page they are currently on. These details made in the URLs will help perfectly for any future additions as pages within the application as features which can enhance the user experience. It would be linked to the application and then put a URL that can be navigated in the application. For example, if the team wanted to add an extra page to grand the user to edit any additional settings like changing the email or adding some more personal information like what year the student is currently on or when will graduate, it would be added by using the ‘additional settings’ at the end of the URL. As a team we wanted to make the application as easy to use and navigate as it could possibly be for the user. Firstly, for the login page it is visibly easy to distinguish where the user should enter their details in the empty fields for their username and password of their account. After the process of signing into the system the user can then progress in the coursework page. In the case where the user does not have an account and haven’t registered in the system, they can procced through a registration process. In the registration process the user will be asked to set up a new account, the user will be asked to enter their details about their new username, password, age, programme they are studying as a student and email. The programme will help what modules their projects are included in, this will help to show the related results for any of these projects. Once the registration process by the user has been finished will be directed to the coursework page, at this stage on the user can start adding their courseworks in the system. The user can also remove any courseworks they do not want to be included in their data of their account, moreover the user can update and modify the project. In general, the team feel comfortable that the URL design can allow for any future additions in the application if it is needed, as well as letting the user to know where they are.

# Persistence

Persistence data is data that is acknowledged permanent at rest with the leaving of software and devices. Data that is stable which is reclaimable form device’s memory. Data In the coursework's application is persisted among user sessions. Data persisted in database by user could include coursework projects and all its properties like project’s title, project’s due date, project’s milestones, module for which project is saved and the actual completion date of the project. One of the main functionalities of the application and at this same time its biggest value is to save effectively users data. It is relevant to prevent deleting this information between users' sessions so he will not stay without his progress saved. Every saved coursework should have unique id number which will help to assign them to proper database. Any saved progress of the user should stay as it is even while switching between sessions. In this way any of users courseworks will not be compromised in any way and student will not be in situation where he lost his milestone or the actual project.

# 4. Security

Security is an extremely important aspect of the project, because the system will store in the database information such as emails, passwords, the name of the coursework’s, deadlines and any progress that has been made by the user it is crucial not only for the user’s to fell safe and protected while using the program because it plays a vital role for the users except for the features the system offers to be able to trust the program in order to keep using it.

## 4.1 Password Protected

More specifically for the security and safety of the user we decided to implement a Sign In section where the user in order to access his/her personal account with the private information will have to type in the correct Username and Password. Also, the user will have the ability to reset their password, although the last features is not fully implemented yet. Password will be protected with encryption in order to protect them from attacks and compromises. More specifically when an attacker is able to obtain access on a person’s individual server encrypted password, they will not be able to decrypt it because they attackers will not have access to the master password. At this time the system stores the passwords on the database and does not provide password encryption.

## **4.2 Web Application Firewall (WAF)**

WAF are being used as a gateway and blocks any harmful HTTP requests by using sever heuristics to decide if the system will give access to the request or block it intime in order to prevent any malicious attacks on the system. On top of that, a benefit of Web Application Firewalls is that no changes to the system are required to be implemented.

## 4.3 Session Token

Session tokens are being used in order the browser to have as little information about the user who interacts with the program as possible. More specifically the browser will be given a text instead of the User’s actual sign in information. Additionally, that session token gives the user a form of anonymity because it does not contain any person information that can lead to him/her. When the user will try to login and sends the request to the server the session token will be included in that request, then the server will store each user’s individual session token. While the user is logged in the program, he/she can access any functionality and use it for as long as they wish for. The session token will be valid up until the time the user will log out, then the session token will expire after a certain amount of time. Although in most application when the user chooses to close the program he/she will stay connected up until they wish to log out, in our version of the system when the user closes the program they will automatically will be loges out.

## 4.4 Vulnerabilities Prioritization

On top of that the development team will prioritize the serious system vulnerabilities in order for them to be addresses ad quickly as possible and also save valuable time for less important vulnerabilities. Such serious vulnerabilities might have to do with the User’s personal Data. Some of the serious vulnerabilities which may compromise user’s personal information are SQL Injections where an attacker uses application or website code to access or corrupt database, if successful database information can be read, removed or modified. The web application should also always have Transport Layer Security (TLS) and Secure Socket Layers (SSL). These certificates provide communication security using cryptographic keys when browsing website so no one else can see and access personal information.

# Documentation of System Testing.

The table below summarizes the results of system testing:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Date Tested | Tester | Pass/Fail | Evidence |
| 1.1 | Sign in section is being displayed | IOANNIS  GIANNAKIS | Pass |  |
| 1.2 | User is able to type their Username | IOANNIS  GIANNAKIS | Pass |  |
| 1.3 | User is able to type their Password | IOANNIS  GIANNAKIS | Pass |  |
| 2.0 | URL Functionality(http://localhost:3000/) | Maciej Tokarski | Pass |  |
| 2.1 | User is able to access coursework's page from login page | Maciej Tokarski | Pass |  |
| 2.2 | User is able to add coursework | Maciej Tokarski | Pass |  |
| 2.3 | User is able to remove coursework | Maciej Tokarski | Fail |  |
| 2.4 | User is able to modify coursework | Maciej Tokarski | Fail |  |

# Conclusion

In conclusion the final application manages to provide the basic functions that are required, special attention has been given on its usability with simple layouts that help the user for an easier navigation. Also, it keeps a decent data persistence, and this has been achieved by saving all the needed information into the database. On top of it provides password protected services and despite at the moment the system is not able to encrypt the password it provides a features when someone enters their password it doesn’t show it on the actual screen for safety reasons. In addition, the system’s functionality has been tested by the development team. Last but not least we feel that we managed to implement a web-based application that has most of the asked features despite the difficulties we experienced as a team.