**Documentation-INFO263**

Sahaj Bajaj, Nuraida Lesme, William Suharyo, Haider Saeed

**Introduction**

The website designed by us consists of a login page where a user can login to the interface, a dashboard page where the user can see all the upcoming, completed and ongoing events, a create events page where the user is able to create a event filling all the necessary details, and a view events page where the events created can be seen with an option to delete any event. The details of these pages will be provided below. The purpose of designing this website is to reduce the user-unfriendliness and errors in the original access to the tserver.

**Decision making**

The tserver event definition video provided helped us to understand the background knowledge about what we were supposed to do. After understanding the underlying concepts of the problematic initial interface we did research and looked into several timetabling websites, UCLearn and UC website to get ideas for designing and the front end code outlook.

We decided to keep our approach very simplistic to let users with any level of knowledge in technology to be able to use our website. We used simple icons which everyone is aware of these days and we kept the font sizes readable enough to make user experience less time consuming and more productive.

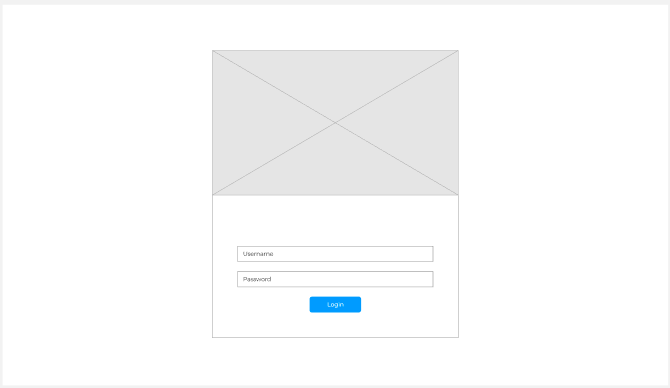
While coding for the website, we decided to divide the tasks on the basis of each person’s strength in different parts. This helped us to achieve good results in the provided time frame.

**Design layout**

Our interface was designed using PHP, Javascript, CSS and SQL and styled using HTML, CSS and AJAX.

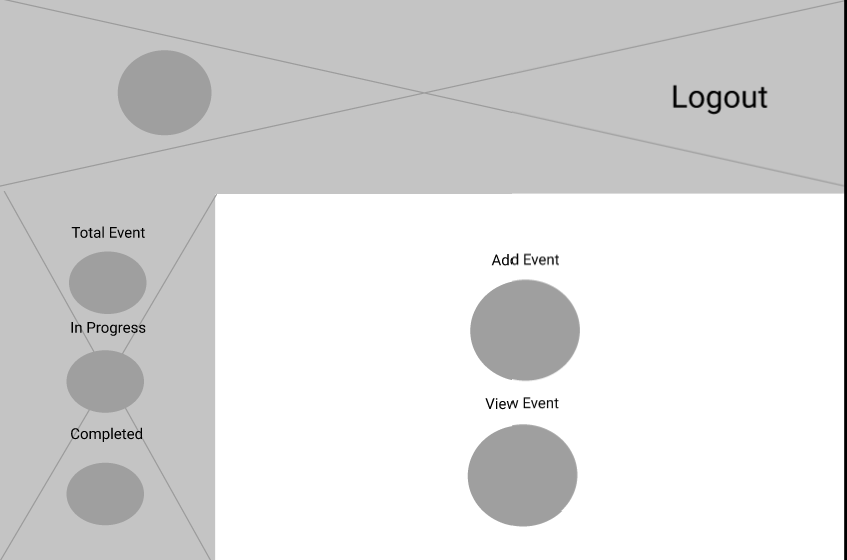
1. **Login page**

Once the folder with the project is clicked, the user will be directed to the home page. Here, there will be a simple form of asking the user to enter their username and password. The form would be in the middle of the page to ensure the main focus point of the user’s eyes. Dark and light grey are used as the background as the colours greatly contrast white, which is the form colour. A University of Canterbury logo would be on top of the form with the texts in white and the background in dark grey, matching the whole theme. This increases visibility for the user. The codes used to implement these are in the *Index.php* and *Login.php* files.



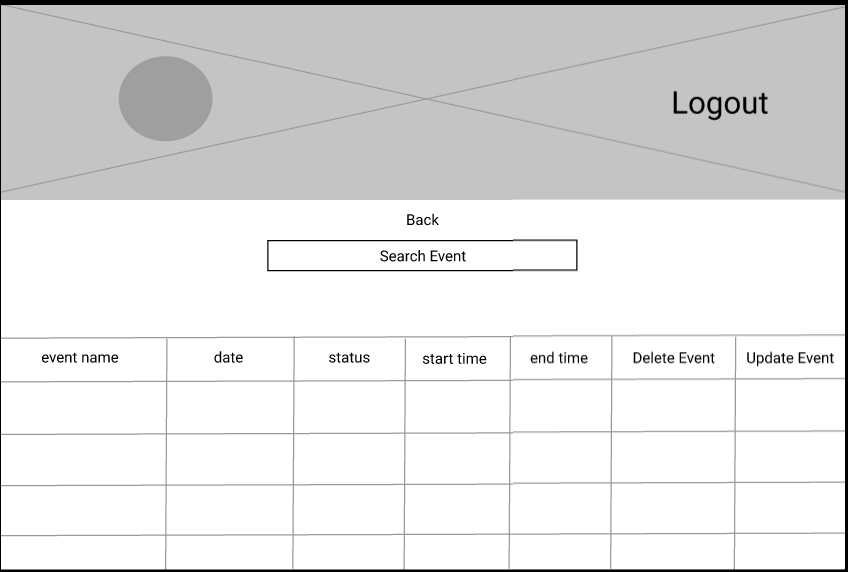
Once the user fills in their username and password, the *User\_authentication.php* file will connect with MySQL to check if their information is in the database. If successful, the user will be directed to the dashboard page and they can proceed with other actions. Whenever the user wants to logout, the *Logout.php* file would be executed and the user would be redirected to the Index.php. However, if the user's information is not in the database, an error of “The password or username is incorrect” would be displayed.

1. **Dashboard**

After the user successfully logged in, they will be presented with the dashboard which will show their profile picture; information on total, happening, and completed events; buttons that will direct the user to a page where they can add an event or view current events that are listed; and a logout button. Each button will open the selected code file while the number shown on the dashboard for events is done in the dashboard.php itself by looking at the database and placing a counter for each of the events accordingly. For the logout button however, after it is clicked it’s going to run logout.php which will destroy all the session and the user will be directed back to the login page.

1. **View event**

When the view event button is clicked, it will direct the user to *view\_event.php* where they will be presented with the same profile picture and logout button from the dashboard; however, they will also be able to see and search all of the events that have been logged to the website. Every event on the database is extracted and placed on the appropriate table. Not only that, but they will also see the delete event button for each event which when clicked will run *delete.php* and delete the event on that row. Users can also edit each event by clicking on the edit event button and they will be directed to a new page similar to the create event page where they can change the details of the event.



1. **Create event**

The *create\_event.php* is a useful php code written to create a new event or add a new class timings. The user can click on the create event button which leads the user to another page, where a new event can be created by selecting the event name and event ID. The date of the event can then be set. The start and end time of the event is then selected. The time offset is then automatically calculated. The room can be selected and by clicking on create event, this new event is then added to the database using ajax.

The *create\_event\_ajax.php* uses ajax to do some background stuff with the data, including storing, changing, and holding the database. Thus, it helps in updating the database asynchronously behind the scenes. Delete events is used to delete a specific event using ajax. It changes the database and prints out “Failed to delete event.” If it is unable to delete the event. The user can click on the delete event button for the event to disappear. This will delete the event from the database.

**Conclusion**

In a nutshell we went through heaps of libraries and resources to do the following project. The jQuery used by us is added to a folder called “js”. Our code uses object oriented programming majorly for its easy to use approach. Plenty of time has been put in each process of this entire project to achieve the best results possible.

**Setup for marking**

For setting up the environment these are the following steps:

1.Please make sure to set your database details Here:

*classes/database.class.php*

2. Set your Date Time Zone Here:

*configuration.php*

3.  Please use MySQL Workbench to execute *new\_databse.sql* to create tables used in the web pages for the procedures and data used.

4. Use the following username and password (admin / 12345) from the user table to login to the web pages.