**IS218 Project: Design a Personal Homework Assignments Management Web Application**

Notes for possible questions you may have:

* This is a group project.
  + You need to form a team with **up to 4 members**. Every member gets the same score. **You can work alone, but not recommended.**
  + Grading a project will be solely based on the quality of the project, and the factor of the number of members will NOT be considered in grading.
* The final project should be publicly accessible through AFS or your own server.
  + a. In rare situations, AFS is not working when approaching the deadline. You take screenshots of your local successfully running, and tune your code to local server. TA will run it using his local server.
    - i. But always, there can be a risk: because local configuration can be very different, there is a chance that the code can run on your computer, but not on TA one. You take the full responsibility and try not to use platform-dependent APIs or any very special APIs/libs/frameworks as they will cause problems when migrating the code to other systems.
    - ii. You need to inform me and TA at your earliest convenience. ***Arrangement must be made before the deadline.***

Hopefully, the above notes can answer your questions. Drop me emails if you have more questions.

**Now, the requirements:**

Please design a simple personal homework assignments management web application to manage your homework. It must have a front-end and a back-end (php) connecting to a database. ***The below requirements are the minimum. If any other design is not covered, you can use your imagination to design it.***

1. Database: please use the sql script I provided for the Project (it is on moodle) to create databases. It should create two tables: accounts and todos, after you import it.

***Note: in order to meet the requirements, you have to re-design, modify, and expand the sql to include, such as, more columns.***

2. The app must have a sign-up, sign-out, and login function. (25%)

a. Signup should take at least the following information:

i. User name: must be an email.

ii. Password

iii. First name

iv. Last name

v. College

vi. Major

b. The app should allow a user to sign out anytime and from any page. After signing out, the user should be re-directed to another page showing a msg: “You have been signed out” (or the same page, depending on your design).

c. Login: registered users can log in using user name and password. It must have input field validation capability

i. Verify whether the entered user name is an email or not (must use JavaScripts for field validation):.

ii. No empty fields are allowed for submission, meaning that you check whether a field is empty or not before submitting the form (must use JavaScripts for field validation):.

iii. Verify entered user name or password are valid or not (here requires server side validation.)

3. System aspects (65%):

1. When a user logged into the system, on the main page, user’s names (first and last) should be always shown.
2. On the home page , the user’s homework assignments must be shown under a user’s name
   1. There should be a section showing the incomplete homework assignments in chronological order. Label this section as “To-do”.
   2. There should be a section showing the completed homework assignments in chronological order. Label this section as “Completed”.
   3. Buttons
      1. For each existing incomplete homework assignment, you should have the following buttons:
         1. create a new to-do assignment. Label the button as “+”, “add”, or ”new”
         2. edit an existing to-do assignment: Label the button as “edit” or “modify” or “revise”
         3. delete an existing to-do assignment
         4. check-off an existing to-do assignment (meaning that users have completed the assignment). Label this button as “complete”
            1. Once an to-do assignment is checked-off, the section of showing completed homework assignments should be refreshed to include this new checked-off homework assignments
      2. For each existing completed homework assignments, you should have the following buttons:
         1. Edit existing finished homework assignments
         2. Delete one
         3. Un-check off: meaning that if a completed task is “unchecked off”, the section of showing “incomplete to-do homework assignments” should be refreshed to include this un-checked homework assignment.
      3. When you add/create a new to-do homework assignment, the user should be directed to a new page and ask for the following information.
         1. Due date and time
         2. A title
         3. A description of the to-do homework assignment (max 144 characters allowed, ***use JavaScript code to verify***)

After saving the newly added to-do homework assignment, the user should be automatically redirected back to the home-page and it should be added in the section of showing all incomplete to-do homework assignments.

* + 1. When you edit an existing homework assignment (complete and in-complete), all existing information of a task should be displayed to the user and editable. After saving the modified task, the user should be automatically redirected back to the home-page and the modified item should be displayed in its corresponding section.

4. Aesthetics (10%)

Style your page using CSS so that it looks polished