CSCB20, Assignment3 (HTML, CSS, JavaScript, MySQL and PHP)
Due date 5th April @ 11:59pm

1 Introduction

In this assignment, you will continue working on your CSCB20 course website from assignment2. However, in this assignment3, you can now make use of JavaScript, MySQL, and PHP also. You will also be asked to host your website on mathlab servers on UTSC network (see section 4 of this handout for more details). This assignment is challenging, and you are asked to get started early on it. We will again place similar restrictions on assignment3 as well, i.e., you cannot use any 3rd party CSS or JavaScript frameworks (i.e., copy/paste or reference them) on your website. However, you can make use of your own JavaScript and CSS framework if you so require. You must continue to work in groups of two for this assignment as well. The webpage must be usable, responsive, and visually appealing. There are some minimum requirements, but for the most part, there is a lot of creativity! This assignment gets a complete clousure on everything that we have been learning in this course starting from databases and finally concluding with web technologies.

2 What is expected in Assignment3?

Now that you have your CSCB20 course website developed for assignment2, you are asked to do the following for assignmetn3:

1. You must create a login page for CSCB20. i.e., all contents of CSCB20 (i.e., lecture material, lab material, course instructor, TA information, etc. etc.) can only be accessed after logging in. In the lecture on 19th March and on the review session also on 19th March, and 26th March, I showed/will show you how login works and you can use the in lecture code and modify it to meet your requirements. You may want to redesign your homepage, i.e., the first page of your website, such that it allows someone without a username and password to create an account. I will leave it up to you what information you like to collect from the user if they do not have a username or password. Again, you do not want to worry about security for this website. However at the very least, there should be an option on the user account creation whether the account is being created for a `student`, `T.A.` or `instructor`. All three types of user i.e. `student`, `T.A.` or `instructor` must have access to all the contents of your website. If anyone tries logging in with incorrect username and password, no content of your website must be accessible. You must instead give them an appropriate message of incorrect username/password and ask them to try again. You are free to design this homepage in any way you like. Feel free to use the University of Toronto Scarborough logo or related picture on your homepage. Keep it simple, clear, and appealing.

2. **Student Users**

   - If a user of type `student` logs into your website. You must have a special link for this student shown that displays his/her current marks from all assignments, midterm exam, labs and the final exam. You may want to debate first with your team member, what is the best way to show this information to the student? Table form (created using div elements?), or in some other form? Let us assume that a student named 'Bill' has logged in. The idea here is that there is

some link such as 'Welcome Bill, click here to see your current grades'. When I click on the above link, you pull what ever information from MySQL database you have about Bill, i.e., his grades and display it back to him. It is perfectly fine to hardcode some dummy data into your database for some students with certain marks. Make sure that Bill only sees his information and no other information about other students. This link must also be visible to users of type `students`. This information is not visible to any other kind of users i.e. `instructor` or `T.A.s`.

- The student must be able to not only view their mark but also have the option to submit a remarking request. To submit a remark request, you may want to have a small button such as 'submit remark request' next to each mark along with a small textbox that allows the student to explain why they are asking for a remarking request.
- Create a special page somewhere on your website that allows students to submit anonymous feedback back to the instructor. The anonymous feedback form can be designed in HTML/CSS/JavaScript or what ever you feel is appropriate. However, it should contain at least the following kind of questions:
  (a) What do you like about the instructor teaching?
  (b) What do you recommend the instructor to do to improve their teaching?
  (c) What do you like about the labs?
  (d) What do you recommend the lab instructors to do to improve their lab teaching?

  You can add more questions as you see fit. However, keep in mind that this link must only be visible to users of type `student`. No one else must have access to this link. There should also be an option for student to select the different instructors that may be present on the website. You should enumerate all the instructor users and display them to the students, so that the student can select the instructor for whom the feedback is meant for. You do not want the feedback going to the wrong instructor!

3. **Instructor Users**

- If a user of type `instructor` logs into your website. You must have a special link for this instructor shown that displays the grades of all his students. You may want to debate first with your team member, what is the best way to show this information to the instructor? Table form (created using div elements?), or in some other form? The idea here is that there is some link such as 'Welcome instructor Anna, click here to see all grades of your class'. When I click on the above link, you pull what ever information from MySQL database you have about the entire class. It is perfectly fine to hardcode some dummy data into your database for some students with certain marks. Make sure that this information is only visible to users of type `instructor`. **You will display this information to any user of type instructor.** This information is not visible to any other kind of users i.e. `student` or `T.A.s`.
- If a user of type `instructor` logs into your website. You must also provide this user with a special link so that the instructor can see all the anonymous feedback. It is OK to hardcode some data into your database. **However keep in mind, that when we test your website on mathlab machines, we will have complete access to your database and the data inside it.** Make sure the logged in instructor only sees the feedback that is meant for them. You do not want some feedback shown to the wrong instructor!
- The instructor must have a special link that allows them to see all remarking request.
- The `instructor` must have the option to enter mark for students. Again, you will want to think about what is the best design to do this. There are multiple ways to do this. You may want to either have a simple form like structure that allows the T.A. to enter the mark and when 'submit' button is clicked, the marks are stored in the database.

4. **T.A Users**

   - If a user of type `T.A` logs into your website. You must have a special link for this T.A. to see all the remarking requests that have been created by students. It is OK for all T.A. to see this information. Remember, at this time there is no option for the student to select the T.A. for whom they would like to direct the request to. (However, if you work on this feature such that students can direct to a specific T.A. and that the T.A. can close the remarking request when it is completed, please make sure to update the Report.pdf, so that we are aware of it and we **may** award some bonus points for this).
   - The `T.A` must have the option to enter mark for students. Again, you will want to think about what is the best design to do this. There are multiple ways to do this. You may want to either have a simple form like structure that allows the T.A. to enter the mark and when 'submit' button is clicked, the marks are stored in the database.

3 Requirements

- Before starting to code in HTML and CSS, you are required to first think about what kind of tables do you require in your MySQL database. You will also need to think about the schema, primary key, number of columns, etc. etc. This is everything that we have looked at in the first weeks of the semester.
- Once you have your schema in place, you will first want to test this out. By testing, I mean writing simple SQL queries that allow you to create a user, log in a user, view student mark, enter information about anonymous feedback, etc. etc.
- Remember the first two points on the top, do not require any PHP, or HTML or CSS or JavaScript. You can test the first two points simply by using material from the first few weeks of this semester. Once you have tested your SQL queries and you know that they work fine, at this point you can start thinking about integrating your SQL queries into your PHP code. Do not move to this step unless and until you have the first two points in this section completed and tested. Refer to Piazza on how to use MySQL installation on mathlab machines on UTSC network.
- You will want to make use of the in lecture code from 19th March and 26th March. In lecture we talked about how to fetch data from MySQL database using PHP, we also talked about how to log in a user using PHP and MySQL. You should be able to take this code and customize it to meet the requirements of this assignment.
- Remember this is a fully functional website. We should be able to login as a instructor on your website and enter some dummy marks for some student. And then logout and relogin using that student and should be able to see the mark of that student. etc etc.

4 Deliverable

There are two deliverables for this. Please make sure to read this very carefully.

1. Your assignment should be submitted as a tar (assignment3.tar) file on Markus (THIS IS TAR AND NOT RAR). MAKE SURE YOU FOLLOW THESE INSTRUCTIONS VERY CAREFULLY. IF YOU DEVIATE FROM ANY OF THESE INSTRUCTIONS, YOU WILL BE PENALIZED. When we untar your tar file, it should contain a single directory called 'Assignment3' which in turn contains the following:

- A directory called **htdocs**: In this directory, you will place all your **html** and **css** and **js** and **php** files. Make sure that your homepage is titled index.php. This will be the first page that we will click on.
- A directory called **sql** that will contain a single sql file which will allow us to create all the schema and any dummy data that you have designed at our end. I will provide more instructions on this later.
- A file called **HonorCode.txt**: You can reuse the honor code from your previous assignment. Again, just make sure that this includes yours, and your partner's information.
- A file called **Report.pdf**. In this report (1-2 pages) you must try and answer these questions:
  - What are some challenges that you and your team member faced? How did you go about addressing these challenges?
  - What are some new things that you have learnt by working on this assignment?
  - Provide simple screenshots taken directly from your website that explains how you have addressed the requirements of this assignment.

2. You will also host your entire assignment3 on this url `https://mathlab.utsc.utoronto.ca/cscb20/{UTORid}`. Only one person in the team is required to do this. However, please make sure to put this link down clearly inside the **Report.pdf** so that we can open this and mark it.

   - In order to have your website up and running on mathlab machines, I will demo the entre sequence of steps in the next review session on 26th March from 6pm to 7pm in IC 220. Please make sure to have one of your team member present. However, if you are already familiar with UNIX platform, you can continue reading.
   - **ssh** into mathlab machines by using Putty (if on windows computer) or use terminal/console (if on mac or linux computer).
   - **cd** into this directory /courses/courses/cscb20w18/utorid
   - All your html, css, javascript and php files must be placed into the directory `cscb20_space`. You will find this directory inside /courses/courses/cscb20w18/utorid
   - You must use the MySQL installation that we have running for you on mathlab machines. You used this for assignment1, and will use this again for assignment3. Refer to Piazza if you have forgotten.

Software download instructions

- As mentioned in lecture on 19th March and also in the review session on 19th March (6 pm to 7 pm in IC 220), you can download your webserver, PHP and MySQL from here: `https://www.mamp.info/en/downloads/`.
- You can use any text editor or IDE to write your code in. However, I recommend using SublimeText as your text editor, and you can get this from here: `https://www.sublimetext.com`.