UNIVERSITY OF REGINA
Department of Computer Science

CS 215 – Web & Database Programming
Winter 2021

Assignment #5: Server-Side Programming

Due: Monday March 28, 2022 by 11:55 PM

This is the fifth in the series of assignments for building a **collaborative note-taking** application. In the previous assignments, designed and built an interface mock-up of the site, wrote some client-side code, and designed and built the database. For this assignment, you will use PHP to implement the server-side programming. The last assignment will focus on performing AJAX-based updates of the website. Because the assignments build upon each other, you may wish to made modifications to your previous assignment submission before starting this assignment.

Now that you have designed and constructed the database, the next step is to use PHP to access the database, save user-specified content, and dynamically construct specific elements of the web pages. You are not permitted to use third-party libraries that simplify or abstract the PHP coding.

This assignment is divided into two different components:

A. PHP Database Entry & File Upload

There are several pages that result in the database being updated with new information. These pages are listed below, with an explanation of what is to be done for each.

1. Sign-up Page
    - validate the form data to ensure that the required fields are present and that they do not contain illegal data; if there is a problem, return to the sign-up form with a generic error message
    - if the data is good, add it to the database and return the user to the Login Page
    - for the user avatar image/graphic upload, move the uploaded file to an appropriate location within the web application file structure, and save its location (URL) in the database
    - see http://www.php.net/manual/en/features.file-upload.post-method.php for more information on managing file uploads

2. Create Note Page
    - only allow access to this page if the user has successfully logged in
    - validate the form data to ensure that all the required elements are present and in the proper format

- if the data is good, add it to the database and return the user to the Note List Page

3. View/Contribute Note Page
    - only allow access to this page if the user has successfully logged in
    - for the note contribution section of this page, validate the form data to ensure that all the required elements are present and in the proper format
    - if the data is good, add it to the database and return the user to this same page (the contribution that was just added should now be at the bottom of the page)

4. Grant/Revoke Access Page
    - when the form is submitted, check to see what the current role settings are for this particular note
    - write the logic to determine whether a new role needs to be added for a user, or if access needs to be revoked (changed to "none" rather than deleting the role)

B. PHP Database Query

There are several pages that result in the database being queried. These pages are listed below, with an explanation of what is to be done for each.

1. Login Form
    - query the database with the login credentials to verify that the user exists and provided the correct password
    - if the information is correct, save the necessary information to know who this user is within a session variable, and redirect to the Post List Page

2. Note List Page
    - query the database for the notes to which the logged in user has access
    - List the notes with the most recently created first
    - each note should display the title of the note, the date/time it was created, the date/time of the last contribution, the total number of contributions, the screenname of the owner, the avatar of the owner, and the role of the logged-in user with respect to the specific note

3. View/Contribute Note Page
    - query the database to retrieve all the contributions to this particular note, ordered with the most recent last
    - for each, display the contribution, date/time it was added, contributor screenname, contributor, contributor avatar

4. Grant/Revoke Access Page

- query the database to get a list of all the users and their current role with respect to this specific note
- generate the page content that lists each user's screenname, avatar, and a form element to add/revoke contributor access to this note

Since the pages you have build for the previous assignments will each be modified, you should ensure that they are still HTML5 compliant (following the syntax rules of XTHML), use CSS, and that there is a proper separation of the specification of the content from the specification of the presentation rules, the specification of the JavaScript code, and the specification of the PHP processing code.

Grading Scheme

This assignment will be graded out of 10 marks, based on the following criteria:

1 mark:	Sign-up Page: account creation
1 mark:	Login Page: login and saving the state in a session variable
2 mark:	Note List Page: dynamically generate list of notes
1 mark:	Create Note Page: create new note
1 mark:	View/Contribute Note Page: dynamically generate note contributions
1 mark:	View/Contribute Note Page: add new note contributions
1 mark:	Grant/Revoke Access Page: dynamically generate list of users and roles
2 mark:	Grant/Revoke Access Page: save new role information (grant/revoke)

Submissions

All of the files for this assignment should be posted to your personal website on the Department of Computer Science servers (a separate directory for this assignment, with a link on your home page), as well as uploaded to UR Courses (a single zip file).

A simple submission log file should be provided that includes your name, student number, class number, the URL of your web application, and a listing of the files you have submitted along with a short explanation of the purpose of each file.

Failure to provide these support documents will result in delays in the grading of your assignment and possibly a deduction in your grade.

Late submissions (up to 12 hours) will be accepted with an automatic 20% grade deduction (-2 marks). If there are exceptional circumstances that kept you from submitting your assignment on-time, you should consult with your instructor as soon as you are able to do so. See the syllabus for more details on the late policy for this class.