

UNIVERSITY OF REGINA
Department of Computer Science

CS 215 – Web & Database Programming
Winter 2022

Assignment #2: Interface Mock-Up

Due: Monday ~~January 31~~ February 7, 2022 by 11:55 PM

The goal of this assignment is for you to implement the core user interface for a **collaborative note-taking** application. In the subsequent assignments, you will implement form validation in JavaScript (Assignment #3), design and implement the database (Assignment #4), implement the back-end programming in PHP (Assignment #5), and implement AJAX-based updating (Assignment #6).

Because the other assignments build upon this one, you should take care to implement a complete interface with all the required elements (listed below). However, you will have a chance (and in some cases, be required to) update your previous work in the subsequent assignments.

After having designed the interface and interaction mechanisms in the previous assignment, your goal in this one is to build the pages in HTML5 and CSS. These pages should be linked together to illustrate the features of the application. For any pages that will ultimately contain user-contributed data, they can be created with sample data to illustrate your design.

You are not permitted to use third-party libraries that simplify or abstract the HTML or CSS coding.

The following pages are to be created (the details below are the same as those listed in the first assignment):

1. Login Page

The purpose of this page is to allow a user to enter their username and password to login to the website.

2. Sign-up Page

This page will collect the information required for creating a new account in your collaborative note-taking application. At the least, it should collect an email address, screenname, avatar image/graphic, and a password. The user should be asked to provide their password twice, to ensure that it is entered properly.

3. Note List Page

This is the page that a user will be shown after they login. It will list all the notes that the user has created, as well as all the notes to which the

user has been granted access. Whether these are provided in a single list or separately is a design choice that needs to be made.

Each note should include the title of the note, the date/time it was created, the date/time the last contribution was made, the total number of contributions, and two buttons: view note and grant/revoke access. The grant/revoke access button should only be shown for the notes that the current user has created.

There should be a button placed somewhere on this page to create a new note.

4. Create Note Page

This page will provide a form in which the user can create a new note. All that is needed is a field in which the user can give the note a name and a button to submit it.

5. View/Contribute Note Page

This page will show the detailed information for the selected note. It should display the title of the note, the date/time it was created, the screen name and avatar image/graphic of the creator of the note. The current content for the note should be displayed in the order each contribution was made. For each contribution, the textual content should be displayed, along with the date/time it was added and the screen name and avatar image/graphic of the contributor.

At the bottom of the page, a form element should be provided to allow the current user to make a new contribution to the note.

6. Grant/Revoke Access Page

This page is accessed from the Note List Page, and should show the title of the note for which it was accessed. A list of all users who have an account in the system (screen name and avatar image/graphic) will be shown (except for the owner of this note), along with a checkbox for each user to grant or revoke access to the note.

All of these files should be linked together, so that it is possible to click through the pages to evaluate the design and implementation.

You must ensure that each of these pages makes appropriate use of HTML5 (following the syntax rules of XHTML) and CSS, and construct them such that there is a proper separation of the specification of the content from the specification of the presentation rules. Your submission should include seven HTML5 pages, and a single CSS file.

Grading Scheme

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

- 6 marks: HTML5 that follows XHTML syntax rules for all six pages
- 2 marks: A single CSS file to specify the presentation details
- 2 mark: Appropriate use of CSS Grid to structure the pages

If there are any missing pages, your grade will be scaled down according to the number of missing pages. For example, if you submit only 5 of the 6 required pages, your grade will be 5/6 of what is assigned based on this marking scheme.

Submissions

All the files for this assignment must 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 of your website directory).

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 the submission log file 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.