1 Make Your Site Functional

It's time to make your project dynamic. Using techniques discussed in class, add form inputs, PHP, and use PDO with MySQL to save and display data on your website. Sanitize, validate, display errors, redesign your forms and/or tables as necessary. This is your largest and possibly most difficult assignment. It will be time-consuming so start early. The requirements are as follows:

(a) **Create your forms** Your website should have a login form, and any other forms from your mockup in homework 3. Add those forms and inputs to your website. Decide whether to use GET or POST in each form. Make sure to use a form action that is a different PHP page to handle the form and redirect as necessary.

(b) **Use PHP to retrieve and sanitize form data** Use the global GET and POST arrays to get your form data in PHP, and sanitize properly. If you're going to display data in HTML, make sure it's properly escaped.

(c) **Validate form data** Validate any email addresses, dates, etc. Use at least one regular expression to validate one of the form inputs. Display a proper error statement to the user to indicate any incorrect fields. Make it obvious to the user if they missed anything required. Validate data even if options are restricted (i.e. select, radio, etc) need validation too. If any form data was incorrect when the user submits the form, *DO NOT* let the form fields clear when giving the user a chance to fix their input. Fill out the form with previously entered data. Optionally, for a better user experience, change the color of the input element with the bad data using CSS.

(d) **Use a session for peristing data across pages** Use the PHP session to carry data from one page to the next. This could include errors, notices, permissions, if you're currently logged in, etc.

(e) **Save form data to MySQL** Using PHP and PDO prepared statements, save form data to the tables you designed in homework 5 as necessary, (using INSERT, UPDATE, or DELETE). Create a separate PHP class to contain all your prepared statements. Call it Dao.php. Put every query/prepared statement in that class.

(f) **Get data from MySQL to display** Using PHP and PDO prepared statements, get form data from the tables you designed in homework 5 as necessary. Display them on the pages you designed in your mock-up (using SELECT and possibly JOIN). If you have any queries needing a WHERE clause, this is probably where you will use values from a GET form using the query string.

(g) **Login portion should work** The login part of your site should be fully functional, as well as all other forms and data-driven pages. Make it easy to use, so when I'm grading, it will be obvious how to use your site.