

Nick Petty

COP3813 HW 7 Report

Purpose

This assignment has two parts - creating a user table, then allowing users to sign into a web application using that table. The usertable.php script, when run from a browser, clears an existing USERS table from my LAMP database, creates the table, initializes the userid and password fields, and inserts two sample users. Once this is run, users can create accounts and sign into a photo sharing app. Although the photo sharing functionality has not been implemented, the primary effort for this deliverable was the sign in and sign up feature. Additionally, sessions are used to keep accounts signed in until they sign out or sign in with another account (not possible through normal site navigation). For security, passwords are salted and hashed when stored in the database, and password input fields do not show their contents.

Design

The usertable.php file was provided as part of the assignment, and so its design was left as is. Initial app design was taken from the textbook's "Robin's Nest" example in chapter 26, and then functional components were inserted into the Bootstrap "Signin" example. The photo wall itself is still using the Bootstrap "Jumbotron" template like the rest of my portfolio, but only keeping the navbar and main content div. Colors are blue, grey, black, and white, again taken from Bootstrap's templates. Page interaction is done through the sign up/in form, so it is centered and the only element displayed. However, I was unable to program page transitions without clearing the form and presenting text, which breaks the flow of interaction. Hopefully I'll be able to fix this in future deliverables. Although not complete, the wall page uses a top-fixed navbar for action buttons, and keeps content separated with Bootstrap's well class. The layout is simple and functional, both for the sign in/up form and the wall, and is also mobile responsive.

Development

Tools

- Sublime Text 2 for coding.
- PHP for server-side data processing.
- HTML and CSS for webpage layout and styling.
- JavaScript and jQuery for DOM manipulation.
- Bootstrap for template, stylesheet, and JavaScript.
- Textbook chapters 12 and 26 for PHP code examples.
- Packetcode "Facebook Wall Design" tutorial for wall page layout.
- Favicon-generator.org for favicon.
- Cyberduck for SFTP to LAMP server.

- GitHub and SourceTree for repository management.
- Safari (OSX and iOS), Chrome, Firefox for viewing and testing.
- Nu HTML Checker for validation (no errors or warnings found).

I began this assignment by running the provided database access and initialization scripts, `db_connect.php` and `tabletest.php`. With some modification (and moving steps around), I was able to get my USERS table created with sample users and leave it available for use with my photo sharing app. The sign in/up form was at first created from textbook examples, specifically the “Robin’s Nest” application in chapter 26. Once I decided what components I could use from that app, I added the password security features from examples in chapter 12, and was able to create the sign in, sign up, and sign out use cases. This was incredibly difficult, as my PHP skill is still very limited and I do not have an IDE. All changes were tested live, and often broke on first runs. I certainly learned a lot of PHP, but I don’t feel comfortable with the language, my code is sloppy, and the user experience is not nearly as good as with my other assignments. The most glaring problem with the app is the confirmation pages that appear between actions, which I will try to remove in the next submission. I am happy that I was able to handle cases where the user navigates to the wrong page via the address bar. For example, if a user is already signed in and they access `signin.php`, they remain signed in and able to go back to the wall, sign out, or switch accounts. This is not an outwardly available feature, but the navigation system is robust enough to handle all the boundary cases I could test. A small feature that I took from the “Robin’s Nest” example in the textbook was making the app’s name a PHP variable, allowing me to rename the app and apply that name to all instances from a single line of code. For the next assignment, I like that I set up my wall in such a way that it should be relatively straightforward to pull pictures and text from the database and populate my wall. Overall, I’d estimate that this assignment took me about 15 hours to complete, and was extremely difficult.