Sam Hirvilampi

Project Report – CSCE 310

**User Authentication and Roles – Sam Hirvilampi**

Admin Functionality

* Admin\_landing.php
* Admin\_page.php

Student Functionality

* Student\_landing.php
* Student\_page.php

Other Files

* Login.php
* Index.php
* Header.php
* Logout.php
* Process\_signup.php

Index Implemented

An index was implemented on the “user” table and applied to the “User\_Type” column. The purpose of this index is to speed up the query when looking for users that fall under a specific type. This means that you can search quicker for all the users under either “admin” or “student” types. In the project, the index can be seen on the admin side when you select to view all the admins in the system.

View Implemented

A view was created that concatenated the “user” and “college\_student” table. The purpose of this is so that when an admin wants to view all the students in the database, it will display relevant information from both tables since each “student” type has personal information in both tables. This view is seen on the admin side when you select to view all the students in the system.

Implementation Summary

The purpose of my implementation was to control access and permissions to the database.

*Login functionality*

I created a login page (login.php, index.php) in which users can signup or students can sign up. The login page verifies access permissions by looking through the user table to find an instance where the entered username and password match. Additionally, the login page will verify that the user’s “access” attribute is set to 1. Once they login the system store’s the UIN, access permissions, and user type in the session. If the system identifies that an admin just logged in, they will be redirected to the admin landing page (admin\_landing.php). This page has a navigation bar that lets you access all the features of the project on the admin side. The same goes for students where they are directed to the student landing page (student\_landing.php) if they are a student.

A screenshot of a login form

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Additionally, I implemented a “signup” feature on the login page for students to create themselves an account which gets entered into the tracking system. Upon signing up a new entry is created in both the user and college\_student tables. The “process\_signup.php” file was used for the signup logic.

A screenshot of a computer

Description automatically generated

*Admin functionality*

Once the user selects “Role Management” from the landing page when signed in as an admin, they will be taken the “admin\_page.php”. Here admins can insert, update, select, and delete other users in the system. The index and view are implemented on this page under the “view users” section. All the logic functions as intended.

When new admins are created, they are added to the “user” table. When new students are created, they are added to both the “user” and “college\_students” tables.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

Admins can edit any attributes for either user type. The admin needs to select whether they are updating a student or admin and based off that selection, you will have the option to change information in the user and/or college\_student tables. Admins can also change a user’s role. When an admin gets changed to a student user type, their “User\_Type” is changed to student a new entry is created for them in college\_student. When a student gets changed to an admin, their entry in college\_student is deleted and their User\_Type in “user” table is changed accordingly. Something to note is that the admin only needs to enter information into the cells they wish to modify information in, they do not need to re-enter all the other information.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

For the select feature, admins can choose from a drop down if they wish to see all the admins or all the students. Selecting admins will call and index that retrieves all users in the user table that have the User\_Type = “admin”. Selecting students will call a view that concatenates both the user and college\_student table and display all the users where User\_Type = “student”.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A white grid with black lines

Description automatically generated with medium confidence

Finally, for deleting there are two methods. An admin can revoke a user’s access by setting their “access” attribute to “0”. The second option lets an admin enter a UIN and press “delete” and it will wipe that user from both the user and college\_student tables.

A screenshot of a computer

Description automatically generated

*Student functionality*

Once a student selects “Profile Management” from the navigation bar in the student\_landing.php they will be redirected to student\_page.php where they have the options to view, modify, and deactivate their account.

In profile management, the student will see all of their personal information with the exception of their “access” and “User\_Type” because these are attributes, I want only admins to see.

The student can click and edit and cell to edit their information except for UIN because you will never change your UIN. Once they change a cell they can press “update” to make the changes.

The student has the option to deactivate their account as well. This functions behind the scenes by changing their “access” attribute from 1 to 0. I included a feature that asks the user to verify they really want to deactivate their account to prevent accidents.

A close-up of a computer screen

Description automatically generatedA screenshot of a computer

Description automatically generated

*Miscellaneous functions*

I added a “header.php” which is added to all the other pages. This file makes it so that the UIN of the user who is currently signed in is displayed at the top right of the page along with a logout button. The logout button will end the session and redirect the user to the login page.

A black numbers on a white background

Description automatically generated