The Database Development and Class Registration paper

Jacob Colburn

University of Arizona Global Campus

CST499: Capstone for Computer Software Technology

Instructor: Prof. Joseph Rangitsch

13AUG24

**Introduction**

A screenshot of a computer

Description automatically generated

Beginning in week 3 we started developing a course management system where students can register and setup a profile, browse courses, and enroll/withraw from them as needed. The goal of this project was to streamline the process of students enrolling at a college/institution. Week 3 we implemented login and registration pages, the index, and the database tables to lay the foundation for the system. Now, for week 4 we are creating the rest of the tables required, connecting them to the front end, and implementing the profile page and course registration and withdraw. Throughout this assignment I will go over the implementation of these features and functionalities implemented over the past week. Additionally, there will be screenshots throughout, and a section that will contain all screenshots to illustrate the database and pages.

**Week 3 Revisions**

Per the week 3 feedback, I didn’t update much of implemented pages and functions. I did perform a bit of work making the site a look and feel a bit better for users.

**RAR File**



**MySQL Database**

A screenshot of a computer

Description automatically generated

Implementing the database was my favorite part of the project. I know a proper db would have many more tables, but im trying to follow the requirements specifically so I only created enough to complete the task. Overall the database “course\_enrollment” contains 3 tables, a student table, a course table, and an enrollment table to track the enrollments and the many to many course. I went ahead and implemented all threes of these in week 3, but this week I added triggers. Triggers are actions that are performed when a certain condition or trigger is met. The triggers I implemented were for students who enroll or withdrawal from courses it will insert their enrollment into the enrollment table, and then the trigger would increase a student counter in the courses table (MySQL, 2024). The courses have a set max capacity of 10 students so once that trigger is activated 10 times, the course will not accept any enrollments.

**Page Development**

A screenshot of a computer program

Description automatically generated

Page development this week was a tough one. I setup the usual html/css webpages as a base for the contact us, profile/edit\_profile, transcript, and course enrollment pages. Starting with the profile page, in week 3 we implemented a connection class to connect the system to our db and used that connection to setup registrations and logins. This week the same kind of concept was used to implement profile editing and course management. All of the pages that had any db output or data related to a specific user would have a file required that sets the session id equal to a new id appended with the users id or “$\_SESSION[“id”]. An if else look or a try catch block is then implemeted with php to query the database and fetch any data relevant to the studentId that’s = to the session[“id”. That data can then be output anywhere we need (PHP, 2024). This same method was used for most of the pages this week including the profile page, index, edit profile, and courses pages. The courses page was a tough one. I was able to display all of the courses correctly and neatly but I could not figure out how to enroll students. After studying for a bit I found that I can run multiple queries and decided to query to fetch the course data, and also query to Perform the withdrawal/enrollments (Krossing, D., 2023). Once the functions were created, I added buttons inside forms to post the withdraw and enrollments. PHP has been a blast to work with, I just can’t stand making things look good with HTML. I’ve spent the last 2 days trying to figure out why my course boxes wouldn’t stay in neat boxes.

**User Experience**

The typical user experience for my site would be loading into the home page initially. You would be welcomed with a message to sign in, prompting you to click login or the menu button which contains the registration. If login was chose, a new button would appear that says need to register? When registering it would request info such as fname, lname, phone, address, etc. upon registering the user is redirected to the index where now they can login. The login page will display error messages if the user types in the wrong login information and once hey are logged in they are redirected to the index again with a new welcome message ‘Hello “user” ‘. The menu will now be a new logged in menu containing new pages like profile where they can edit their profile and manage their courses, their transcript where they can manage courses, the courses page where they can add and view all the available courses, and a logout button.

**Screenshots**

Index.php(not logged in) A screen shot of a computer

Description automatically generated

Login.phpA screenshot of a computer

Description automatically generated

Index.php(logged in)A screen shot of a course registration

Description automatically generated

Menu(logged in)A screenshot of a computer

Description automatically generated

Profile.phpA screenshot of a computer

Description automatically generated

Edit\_profile.phpA screenshot of a computer

Description automatically generated

Profile.php(after edit)A screenshot of a computer

Description automatically generated

Transcript.phpA screenshot of a computer screen

Description automatically generated

Transcript.php(after withdrawals)A screenshot of a computer

Description automatically generatedcourses.phpA screenshot of a course registration

Description automatically generated

Courses.php(already enrolled)A screenshot of a course

Description automatically generated

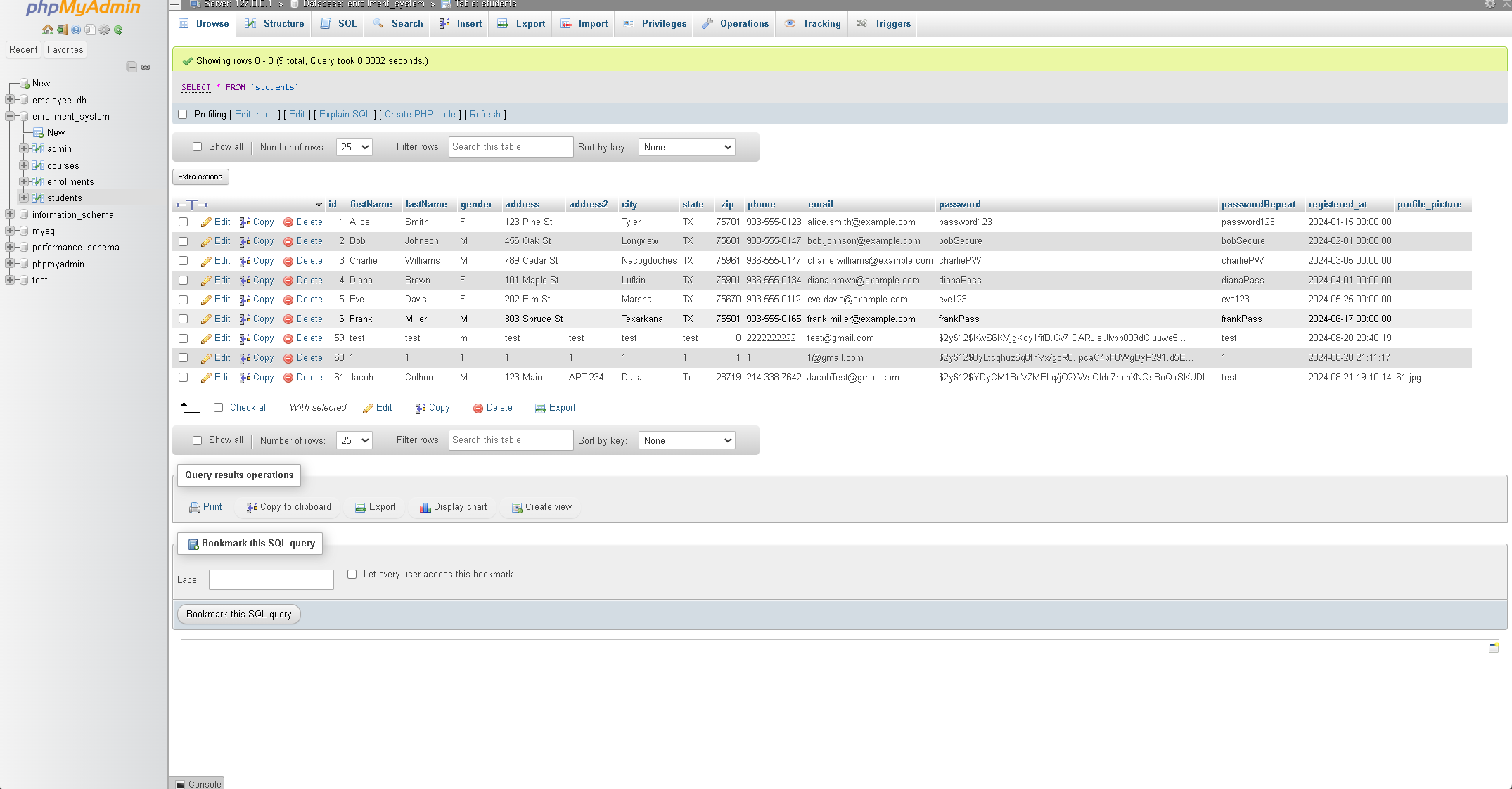
Course.php(successful enrollment)A screenshot of a course registration

Description automatically generated

Transcript.php (after enrollment again)A screenshot of a computer screen

Description automatically generatedIndex.php (after logout)A screen shot of a computer

Description automatically generated

Students tableA screenshot of a computer

Description automatically generatedcourses table

Enrollments tableA screenshot of a computer

Description automatically generated

Upload profile pic

A screen shot of a computer program

Description automatically generated

Courses.php (enrollment page)

A screen shot of a computer program

Description automatically generated

edit\_profile.inc.php (php logic for editing profile)

A screen shot of a computer program

Description automatically generated

Edit\_profile.php (page)

A screen shot of a computer program

Description automatically generated

Profile.php

A screenshot of a computer program

Description automatically generated

A black background with many colorful text

Description automatically generated

Transcript.php (manage courses from profile link)

A screen shot of a computer screen

Description automatically generated

A screen shot of a computer program

Description automatically generated

**References**

Tsui, F., Karam, O., & Bernal, B. (2018). Essentials of software engineering (4th ed.). Jones & Bartlett Learning. [Course Text](https://platform.virdocs.com/read/2348054/11/#/4/2[ch06]/6/4[ch06-sect1-002]/10/4/2)

MySQL, (2024). “27.3.1 Trigger Syntax and Examples”. <https://dev.mysql.com/doc/refman/8.4/en/trigger-syntax.html>

PHP, (2024). “PHP: session\_id – Manual”. <https://www.php.net/manual/en/function.session-id.php>

Krossing, D., (2023). “Create Sessions and Session Security in PHP”. https://www.youtube.com/watch?v=JAgd\_L3GhI0&list=PL0eyrZgxdwhwwQQZA79OzYwl5ewA7HQih&index=25