 Student Enrollment Management System

Jacob Colburn

University of Arizona Global Campus

CST499: Capstone for Computer Software Technology

Instructor: Prof. Joseph Rangitsch

13AUG24

* Explain how to run a PHP file in XAMPP.
* Create the landing page, login page, and registration page for new users.
* Create the MySQL database and tables.
* Discuss the MySQL database functions that you used and the steps you took to create the database connection custom class.
* Develop the registration page layout.
* Develop the registration page PHP source code.
* Develop the table that saves the user information in the database.
* Explain the steps taken to create the registration page and save the user information in the database.
* Provide screenshots of all developed pages, database, tables, layout, and source code.

**Introduction**

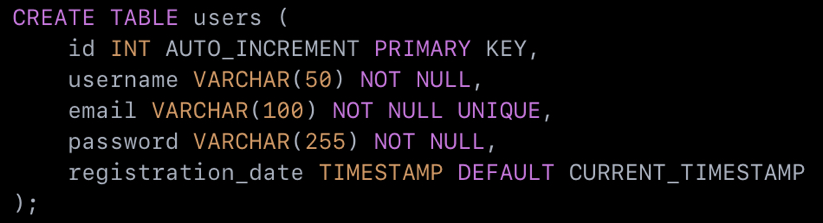
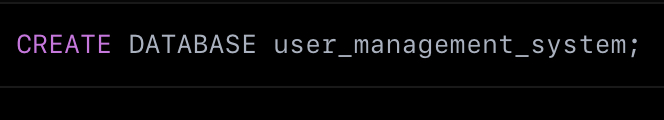
Web development has become an increasingly important skill in todays digital world that almost depends on user friendly systems and interfaces working with detailed and complete backend systems. This paper will outline my process of developing the first stage of my enrollment management system using HTML/CSS, PHP, XAMPP, and MySQL. Currently implemented pages include the landing page (index.php), login page (login.php), and the student registration page (reg.php). The system will securely store registered new users in a MySQL database and illustrate how to implement DBMS and server side scripting. Additionally this paper will provide screenshots and explanations of the developed pages, and the source code.

**How to Run PHP Files in XAMPP**

Using XAMPP to run PHP files is a pretty easy process allowing users new to development a safe and user friendly local development tool. First of XAMPP should be installed and launched. Once launched you will see the contorl panel and will need to start Apache, which is a local server, as well as starting MySQL if your going to be using a database to store any data such as login information (XAMPP, 2024). Once the server is started, navigate to the XAMPP folder and open the folder “htdocs”. This folder is where all your php files should be located. To access your PHP file through the web browser either select the admin button on the control panel to open your webpage or enter “http://localhost/yourfilename”.

A screenshot of a computer

Description automatically generated**MySQL**

A screen shot of a computer program

Description automatically generatedThe database were using for the course management system is a MySQL relational database that will eventually have relationships between students, courses, and their respective instructors. Creating a new database can be done through simple means on PHPmyadmin, but I developed mine using SQL. Currently the student table is developed to enable new students to register to the database through the login.php page. Connecting to a database from the web page is implemented through a custom php class called “dbConnect”. The dbConnect class has 4 variables, the $servername, $user, $pass, and $dbname. Once the variables were created a function “connect()” is implemented and all the variables are assigned to point to the necessary enrollment\_system database we created. A try catch method is then used to connect to the database through a PDO connection, if the connection fails the catch throws a PHP connection error message (PHP Documentation Group, 2024).

**Developing the Pages**

A screen shot of a computer

Description automatically generatedThe landing page is fairly simple, only containing a offcanvas menu with a toggle button on the top left and a brand anchor at the top right of the screen. This was implemented using bootstrap to speed up and simplify the process, but bootstrap can sometimes override your CSS attributes making things a little difficult at times. I required the landing page on the other two pages to ensure that the navbar and menu was uniform throughout the pages.

A screen shot of a computer

Description automatically generatedThe login page was created next and is also just a simple HTML/CSS page without any backend logic currently. I implemented a PHP switch function to change the nav-brand on the top right to state already registered? When on the login page, and changes depending on the page your on currently.

A screen shot of a computer

Description automatically generatedA screen shot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generatedThe registration page gave me the most trouble. For the registration page I required the index.php landing page since it contained the dbConnect class to allow my form to submit to the server. I implemented several user input sections fro attributes such as name, email, password, phone, address, city, etc.

This is where I missed a step. In the input tags for form submissions you must have the attribute “name=“ and the value must be “=“ to the id of the attribute in your database (ex. <input type=password” name= “password”>) otherwise the form wont submit to anything. The registration submission form is handled by a formhandler.ini.php file I created, with the attribute “method-“POST”, meaning it posts to the server. Following the user input areas, a button is implemented with the type set as submit so the file knows to submit the inputted information to the database.

**Source Code and Zip File**

**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)

XAMPP, (2024). “What is XAMPP”. <https://www.apachefriends.org/index.html>

PHP Documentation Group, (2024). “Connections and Connections Management”. https://www.php.net/manual/en/pdo.connections.php