**CS6314 Project Part1**

|  |  |  |  |
| --- | --- | --- | --- |
| Member of Team | NetID | First Name | Last Name |
| 1. | yxw230008 | Yash Mahendrakumar | Wani |

Submission requirements.

Submit (1) a word document (this file with your answers and listing of any program & its run [log or screenshots]) and (2) a zip file of a folder which will contain all the codes (all the source codes, data and any other items related to this assignment and your work done).

This word document (this file) is your documentation (as a basis to add your answers) to be included here - all your answers, all your program-listings, instructions to compile and run [screen-shots, terminal text, or session log] to show your work done, including how to compile and run for each cases.

The word file should also have (1) in header with the course & section number, your name (Name: last name, first name), your netID (email), and Assignment #, (2) in footer with page number, (3) line number (restart from each page in left margin. Page Layout => Line Numbers). (4) Your program should have some comments (minimal or reasonable), to tell the code-reader what a segment of code is doing, and with the comment in the front/head of the program about you, this course and assignment, etc. (you may copy and paste some out of this document for your comment).

\*\* Your executable codes (that you submit) should run in cs1 without any change or recompilation.

\*\* Bring this cover sheet (this page) to TA for your demo.

Upload (attach) this document (with your answers) and a zip file (containing all the codes [source and data, etc.]).

Score-sheet (please specify what you have done by each team member has done below).

|  |  |  |  |
| --- | --- | --- | --- |
| **Member Name** | **Yash Mahendrakumar Wani** | **Demo** | **Documentation** |
| **Task1**  **course DB**  **40%** | Completed the Database Extraction (100%) |  |  |
| **Task2**  **course Web**  **40%** | Created the website using php and made changes in database |  |  |
| **Task3**  **course style**  **20%** | Styled the web page and added 3 sections for search, add and delete, and Edit |  |  |
| **Task4**  **Documentation & Demo**  **(Deduction if not done)** | Completed every documentation |  |  |

|  |
| --- |
| Deduction - Documentation (this .doc file) and upload  Max -70% if not done or poorly prepared |
| Deduction for Demo (Demo schedule or arrangement will be scheduled by TA, for your demo).  Note. You do the demo only for the part(s) that you have done and submitted.  Max -50% if not done or poorly prepared. |
| Deduction (To use of xampp - all work/project to be done using xampp frame)  Do all your work/project done using/via xampp frame. If not, there will be severe penalty (-70%). |

CLO - After successful completion of this course, the student should be able to:

1. Ability to understand web architecture, standards, protocols, tools, and technologies

2. Ability to understand HTML, HTML5 and CSS.

3. Ability to understand JavaScript, JQuery, AJAX, XML, JSON

4. Ability to understand Database Technologies and SQL

5. Ability to understand Server-side programming with PHP

6. Ability to understand Web Services SOAP and RESTful Web Services

7. Ability to understand Web Security Protocols & Standards Semantic Web

Table of Contents

[**CS6314 Project Part1** 1](#_Toc193215683)

[**Preparation** 3](#_Toc193215684)

[1. Task1 Solution 9](#_Toc193215685)

[1.1 Task1.html 9](#_Toc193215686)

[1.2 Screenshots: 10](#_Toc193215687)

[1.3 Listing of files: 16](#_Toc193215688)

[2. Task2 Solution 17](#_Toc193215689)

[2.1 task2.php 17](#_Toc193215690)

[2.2 Screenshots of Website flow 31](#_Toc193215691)

[2.3 Screenshots for Test Cases 36](#_Toc193215692)

[2.4 Listing of each file 69](#_Toc193215693)

[3. Task3 Solution 71](#_Toc193215694)

[3.1 task3.php 71](#_Toc193215695)

[3.2 task3.css 81](#_Toc193215696)

[3.3 Screenshots 87](#_Toc193215697)

[4. Task4 Solution 101](#_Toc193215698)

**Preparation**

CS Graduate Course Listing - Offering S25

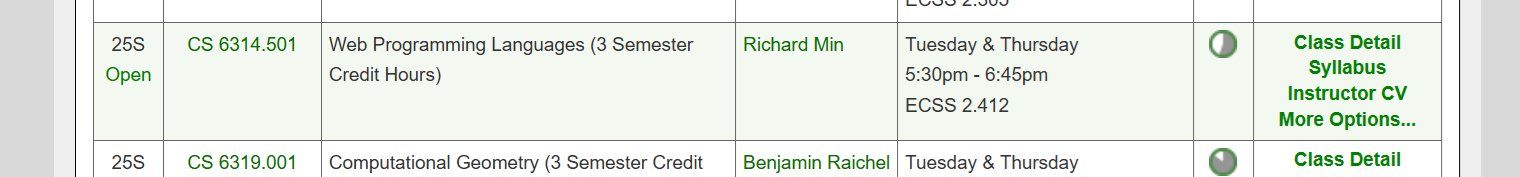
The manager of cs department hired you to be the web developer, to work on UTD CS Course management.

Your first project is to design and implement HTML/CSS with Javascript and PHP with SQL interacting with a database to do the following tasks.

For submission, upload this directory (with all its contents – all the files and all subdirectories, and SQL export file to reconstruct the database and all the tables) in the zip file for this assignment.

Create a folder (cs6314) for this course, and its subfolder for each part (e.g., cs6314/task1) in Xampp for this assignment (task). For your submission, upload the zip file of this directory (with all its contents – all the files and all subdirectories) and this document (as your project report). You will also have a few more subfolders as needed. Note that this folder should contain also a folder for all sql statements (so that all tables can be created using these sql for the project, by exporting the mysql database of all the tables for this project in sql format.

Consider the UTD CS course offering (see the attachment for the web page containing all cs course offering):



Save this web page (in cs6314 folder) to be used in the following tasks.

Name the html file: course.html

Your tasks for Project Part1 is to do:

#1. Extract for each course offering: (1) year-semester (S25), (2) Course Number & Section, (3) Course Title, (4) Instructor, (5) Instructor NetID, (6) Date & Time of the course/section, (7) Classroom Location. This is to be done by Javascript program(s) interacting with HTML document of course offering.

#2. Generate SQL statements to create table and insert the information to be run by PHP program(s)

#3. Have a PHP program(s) to generate a web page of course offerings of S25 (reading MySQL database table created and populated in #2), with functionality to list, add new entry, or to update or delete an entry, as the system is interacting with user to work with the table in the database.

#4. Do all your work/project done using/via xampp frame. If not, there will be severe penalty (-70%).

In cs6314/task1, please provide a word document (task1.docx) to show the result of each case (step-by-step or a code segment) including the screenshots (initial and/or final, and test performed). Provide a proper heading and comment for each test cases so that your reader should be able to follow the flow of your work done.

**Task1**

Create a folder for this task (cs6314/task1)

With the initial web page provided (of the course offering listing), your task is to add javascript program(s) to the web page to do the following:

(0) design database & table for course (offering) to be used for the following subtasks.

(1) to extract all the course information using javascript (task1.html and task1.js)

(2) to write each course information of (1) in sql format to a file (task1.sql)

(3) with two new fields for each course: time created and time updated, and using this sql table

(4) then to create sql table (course1) via Xampp database with sql statement (task1.sql)

(5) after the table is done, export the table in various formats (see below)

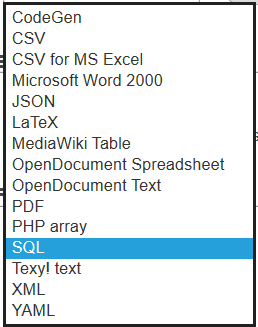
Name this html file with javascript program and sql: task1.html, task1.js, task1.sql

Name the table in database: course1

Hint. For coding hint, check www10 examples.

Note. For a file (e.g., style, javascript, etc.) as needed, use task1 prefix of each file for the file name (e.g., task1misc.html).

Note. To make your design easy, you may define each field to be: varchar (from 100 to 255, or more as needed for a certain field) with "default null" except the course id which will be a primary key of the course table. There could be more than one prerequisite courses for a course. You should include all the saved sql files (as you create or get from any tables in this assignment) in a subdirectory (cs6314/sql) so that the sql files will be also a part of the zip file to be uploaded, and also upload your documentation (this word document).



Save the initial sql statements in a file (task1.sql) and after loading the table, the table will be exported and saved in a file (in cs6314/task1/filename where the filename is specified below):

(a) to save all the course information (SQL document) in a file (task1out.sql)

(b) to save all the course information (XML document) in a file (taskout.xml)

(c) to save all the course information (PHP array) in a file (task1out.php)

(d) to save all the course information (JASON document) in a file (task1out.jason)

In cs6314/task1, please provide a word document (task1.docx) to show the result of each test case, step-by-step, including the screenshots (initial and final, and test performed). Provide a proper heading and comment for each test cases so that your reader should be able to follow the flow of your work done.

**Task2**

Create a folder (cs6314/task2) for the work for this task. Create or copy the sql for the course table (in Task1) to be used for this task.

Design and implement a web page (task2.php) with php similar to www10 example. The initial web page contains the content of cs6314 table (containing all courses for course table). It will provide a few functionalities: (1) to delete an existing course, (2) to update an existing course, and (3) to add a new course.

Name your php program: task2.php

You should have a few test cases for this task. Prepare a few sample courses (5 courses which you have taken or you are taking in this semester) for test cases to be added, to be updated, and to be deleted.

For testcase setup, provide a few buttons with a few javascript functions, respectively, to add, to delete, or to update a test case (versus manual add, delete, or update).

Name the javascript function: addTestCase1, addTestCase2, …

Name the javascript function: deleteTestCase1, deleteTestCase2, …

Name the javascript function: updateTestCase1, updateTestCase3, …

Once the program runs, it will show all the courses (from sql table via php).

Save this initial web page (source) in a file (task2out.html).

After each test case, save the resulting web pages (in cs6314/task2/filename where the filename is specified below):

(1) to save each web page in a file after each test case (test1out.html for test1, test2out.html for test2, …)

In cs6314/task2, please provide a word document (task2.docx) to show the result of each test case, step-by-step, including the screenshots (initial and final, and test performed) for each test case, sql table before and after each test case, etc. Provide a proper heading and comment for each test cases so that your reader should be able to follow the flow of your work done.

**Task3**

Create a folder (cs6314/task3) for the work for this task.

Design and implement a web page with php (from Task2 [or Task1 if you cannot complete Task2]) with a style sheet (and javascript as needed). You may also use UTD branding style for this task.

Name your php program: task3.php

Name your style sheet file: task3.css

(1) The web page should have header, footer, and navigation (as discussed in the classes and lab with adv css, layout, multi-columned web page, etc.).

(2) Initially each course will be listed in a short form with course/section number and title. By clicking or hovering (or your choice of design), the full course information will be displayed (as available).

(3) To have a button to add a new course for your choice of course(s). Once clicked, you should have a block to be shown (or a new web page for add a new course or a pop-up page, etc.). The selected course(s) will be sent to the server to be added to the database table.

(4) The web page should provide a search capability to find a course (or courses) in search string, auto-complete or suggestion-list capability given a prefix string (using course number, title, instructor name, day or time of class). Once selected, the web page will move down to the course (for the current viewport of the web page) as the user likes to navigate to a particular course. You may restrict this option to search course title (or description) only.

(5) Provide responsive web design options using bootstrap

<https://www.w3schools.com/howto/howto_website_bootstrap.asp>

In cs6314/task3, please provide a word document (task3.docx) to show the result of each case (step-by-step or a code segment) including the screenshots (initial and/or final, and test performed). Provide a proper heading and comment for each test cases so that your reader should be able to follow the flow of your work done.

**Task4**

Create a folder (cs6314/task4) for the work for this task.

Deliver your team documentation of this project part1 tasks that you have done:

(1) project technical documents (including code listing, usability assessment, …) – this word document

~~(2) project management plan~~

~~(3) use case and user-interface (of web pages with screenshots)~~

Note. If you have screenshots in the document, make sure that it is readable (font size & no need to zoom).

Hint. A sample code to extract all <p> elements in a web page.

Source: <https://www.w3schools.com/js/js_htmldom_elements.asp>

<https://www.w3schools.com/jsref/met_element_getattribute.asp>

|  |
| --- |
| <html><body>  <h2>Finding HTML Elements by Tag Name</h2>  <div id="main">  <p>The DOM is very useful.</p>  <p>This example demonstrates the <b>getElementsByTagName</b> method.</p>  </div>  <p id="demo"></p>  <script>  var x = document.getElementById("main");  var y = x.getElementsByTagName("p");  document.getElementById("demo").innerHTML =  'The first paragraph (index 0) inside "main" is: ' + y[0].innerHTML;  </script>  </body>  </html> |

|  |
| --- |
| // Source: https://www.w3schools.com/js/js\_htmldom\_collections.asp  var x = document.getElementsByTagName("p");  // The elements in the collection can be accessed by an index number.  // To access the second <p> element you can write:  y = x[1];  // The length property defines the number of elements in an HTMLCollection:  var myCollection = document.getElementsByTagName("p"); var i; for (i = 0; i < myCollection.length; i++) {   myCollection[i].style.backgroundColor = "red"; } |
| // <https://www.w3schools.com/jsref/met_element_getattribute.asp>  var classname = document.getElementsByTagName("H1")[0].getAttribute("class"); |

|  |
| --- |
| // Source: https://www.w3schools.com/js/js\_htmldom\_collections.asp  var x = document.getElementsByTagName("p");  // The elements in the collection can be accessed by an index number.  // To access the second <p> element you can write:  y = x[1];  // The length property defines the number of elements in an HTMLCollection:  var myCollection = document.getElementsByTagName("p"); var i; for (i = 0; i < myCollection.length; i++) {   myCollection[i].style.backgroundColor = "red"; } |

# Task1 Solution

## 1.1 Task1.html

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>      <title>UTD CS Course Data Extractor</title>      <style>          body {              font-family: Arial, sans-serif;              margin: 20px;              padding: 20px;          }          h1 {              color: #c75b12; /\* UTD orange \*/          }          .container {              max-width: 1000px;              margin: 0 auto;          }          .button-container {              margin: 20px 0;          }          button {              background-color: #c75b12;              color: white;              border: none;              padding: 10px 15px;              margin-right: 10px;              border-radius: 4px;              cursor: pointer;              font-size: 14px;          }          button:hover {              background-color: #a84a0f;          }          #status {              margin: 15px 0;              padding: 10px;              background-color: #f5f5f5;              border-radius: 4px;          }          #resultContainer {              margin-top: 20px;              border: 1px solid #ddd;              padding: 10px;              max-height: 400px;              overflow-y: auto;              background-color: #f9f9f9;          }          table {              width: 100%;              border-collapse: collapse;          }          th, td {              border: 1px solid #ddd;              padding: 8px;              text-align: left;          }          th {              background-color: #f2f2f2;          }      </style>  </head>  <body>      <div class="container">          <h1>UTD CS Course Information Extractor</h1>            <div class="button-container">              <button id="extractButton">Extract Course Data & Generate SQL</button>              <button id="downloadButton" disabled>Download SQL File</button>          </div>            <div id="status">Ready to extract course data from coursebook2025.html</div>            <div id="resultContainer">              <p>Results will appear here after extraction.</p>          </div>      </div>        <script src="task1.js"></script>  </body>  </html> |

1.2 Screenshots:

Screenshots with proper heading to show that you have done the work for this task. State clearly for each screenshot on what it is about (to show your work for the task is done), including the screenshot(s) of database table.

|  |
| --- |
| 1.2.1 Starting Apache and MySQL in XAMPP    1.2.2 Selecting Visual Studio Code as editor    1.2.3 Creating CS6314 folder and subfolders    1.2.4 Display if task1.html    1.2.5 After clicking Extract Course Data & Generate SQL it will run the process and display sample data    1.2.6 After clicking Download SQL File it will download the file in the same directory as the html code.    1.2.7 We can see task1.sql in the directory    1.2.8 phpMyAdmin page    1.2.9 Create CS6314 and importing task1.sql    1.2.10 Exporting course1to xml, php, json, and sql format    1.2.11 You can see json, php, sql and xml file |

## 1.3 Listing of files:

Listing of each file or (as needed or if missed)

|  |
| --- |
| 1. Coursebook2025.html – The file which has all the course data 2. task1.html – This file is to display options for extracting and downloading data in sql file 3. task1.js – Writing function to download and generate sql file 4. task1.sql – downloaded sql file 5. task1.php – To download sql file in the same directory 6. task1out.json – exported from PhpMyAdmin 7. task1out.xml - exported from PhpMyAdmin 8. task1out.php - exported from PhpMyAdmin 9. task1out.sql - exported from PhpMyAdmin |

# Task2 Solution

## 2.1 task2.php

|  |
| --- |
| <?php  // Database connection settings  $servername = "localhost";  $username = "root"; // Default XAMPP username  $password = ""; // Default XAMPP password  $dbname = "cs6314"; // Your database name  // Create connection  $conn = new mysqli($servername, $username, $password, $dbname);  // Check connection  if ($conn->connect\_error) {      die("Connection failed: " . $conn->connect\_error);  }  // Add this right after your database connection code  // Modify the savePageAsFile function to be more descriptive with operation types  // Modify the savePageAsFile function to only save test case outputs with specific naming  function savePageAsFile($operation, $identifier)  {      // Only proceed if this is a test case operation      if (strpos($identifier, 'TestCase') === 0) {          // Extract test case number from identifier (e.g., "TestCase1" → "1")          $testCaseNum = substr($identifier, 8);            // Determine file number based on operation type and test case number          // Add: 1-5, Update: 6-10, Delete: 11-15          $fileNumber = 0;          if ($operation === "add") {              $fileNumber = $testCaseNum;          } else if ($operation === "update") {              $fileNumber = $testCaseNum + 5;          } else if ($operation === "delete") {              $fileNumber = $testCaseNum + 10;          }            // Format filename as test{number}out.html          $filename = "test" . $fileNumber . "out.html";          // Get the current database state          global $conn;          $coursesResult = $conn->query("SELECT \* FROM course1 ORDER BY course\_num\_section");          // Start building the HTML content          $content = '<!DOCTYPE html>  <html>  <head>      <title>CS6314 - Task 2 Output</title>      <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">  </head>  <body>      <div class="container mt-4">          <h2>CS6314 Course Management - ' . ucfirst($operation) . ' Operation: ' . $identifier . '</h2>          <p>Generated on: ' . date('Y-m-d H:i:s') . '</p>          <table class="table table-bordered">              <thead>                  <tr>                      <th>S.No</th>                      <th>Semester</th>                      <th>Course</th>                      <th>Title</th>                      <th>Credits</th>                      <th>Instructor</th>                      <th>Date & Time</th>                      <th>Location</th>                  </tr>              </thead>              <tbody>';          $serial\_number = 1;          while ($row = $coursesResult->fetch\_assoc()) {              $content .= '                  <tr>                      <td>' . $serial\_number++ . '</td>                      <td>' . $row['semester'] . '</td>                      <td>' . $row['course\_num\_section'] . '</td>                      <td>' . $row['title'] . '</td>                      <td>' . $row['credit\_hours'] . '</td>                      <td>' . $row['instructor'] . ' (' . $row['instructor\_netid'] . ')</td>                      <td>' . $row['date\_time'] . '</td>                      <td>' . $row['location'] . '</td>                  </tr>';          }          $content .= '              </tbody>          </table>      </div>  </body>  </html>';          // Save the content to a file in the same directory          file\_put\_contents($filename, $content);      }  }  // Initialize variables  $id = "";  $semester = "";  $course\_num\_section = "";  $title = "";  $credit\_hours = "";  $instructor = "";  $instructor\_netid = "";  $date\_time = "";  $location = "";  $update = false;  // Process form submission for adding or updating a course  // Process form submission for adding or updating a course  if (isset($\_POST['save'])) {      $semester = $\_POST['semester'];      $course\_num\_section = $\_POST['course\_num\_section'];      $title = $\_POST['title'];      $credit\_hours = $\_POST['credit\_hours'];      $instructor = $\_POST['instructor'];      $instructor\_netid = $\_POST['instructor\_netid'];      $date\_time = $\_POST['date\_time'];      $location = $\_POST['location'];      $current\_time = date('Y-m-d H:i:s');      // For new record      if (!isset($\_POST['id']) || empty($\_POST['id'])) {          // Get the max ID and increment by 1          $result = $conn->query("SELECT MAX(id) as max\_id FROM course1");          $row = $result->fetch\_assoc();          $next\_id = $row['max\_id'] + 1;          $sql = "INSERT INTO course1 (id, semester, course\_num\_section, title, credit\_hours, instructor, instructor\_netid, date\_time, location, time\_created, time\_updated)                  VALUES ($next\_id, '$semester', '$course\_num\_section', '$title', '$credit\_hours', '$instructor', '$instructor\_netid', '$date\_time', '$location', '$current\_time', '$current\_time')";          if ($conn->query($sql) === TRUE) {              echo "<div class='alert alert-success'>Course added successfully</div>";              // Redirect to clear the form              header("Location: task2.php");              exit();          } else {              echo "<div class='alert alert-danger'>Error: " . $sql . "<br>" . $conn->error . "</div>";          }      } else {          // For update          $id = $\_POST['id'];          $sql = "UPDATE course1 SET                  semester='$semester',                  course\_num\_section='$course\_num\_section',                  title='$title',                  credit\_hours='$credit\_hours',                  instructor='$instructor',                  instructor\_netid='$instructor\_netid',                  date\_time='$date\_time',                  location='$location',                  time\_updated='$current\_time'                  WHERE id=$id";          if ($conn->query($sql) === TRUE) {              echo "<div class='alert alert-success'>Course updated successfully</div>";              // Redirect to clear the form              header("Location: task2.php");              exit();          } else {              echo "<div class='alert alert-danger'>Error: " . $sql . "<br>" . $conn->error . "</div>";          }      }  }  // Process test case additions (1-5)  // Process test case additions (1-5)  if (isset($\_GET['addTestCase'])) {      $testCase = $\_GET['addTestCase'];      $current\_time = date('Y-m-d H:i:s');      // Get the max ID and increment by 1      $result = $conn->query("SELECT MAX(id) as max\_id FROM course1");      $row = $result->fetch\_assoc();      $next\_id = $row['max\_id'] + 1;      switch ($testCase) {          case '1':              $sql = "INSERT INTO course1 (id, semester, course\_num\_section, title, credit\_hours, instructor, instructor\_netid, date\_time, location, time\_created, time\_updated)                      VALUES ($next\_id, '25FOpen', 'CS 6314.001', 'Web Programming Languages', '3', 'John Smith', 'jxs123456', 'Monday & Wednesday 5:30pm - 6:45pm', 'ECSS 2.306', '$current\_time', '$current\_time')";              break;          case '2':              $sql = "INSERT INTO course1 (id, semester, course\_num\_section, title, credit\_hours, instructor, instructor\_netid, date\_time, location, time\_created, time\_updated)                      VALUES ($next\_id, '25FOpen', 'CS 6363.001', 'Design and Analysis of Algorithms', '3', 'Jane Doe', 'jxd987654', 'Tuesday & Thursday 7:00pm - 8:15pm', 'ECSS 3.503', '$current\_time', '$current\_time')";              break;          case '3':              $sql = "INSERT INTO course1 (id, semester, course\_num\_section, title, credit\_hours, instructor, instructor\_netid, date\_time, location, time\_created, time\_updated)                      VALUES ($next\_id, '25SOpen', 'CS 5343.001', 'Data Structures and Algorithms', '3', 'Robert Johnson', 'rxj456789', 'Monday & Wednesday 2:30pm - 3:45pm', 'ECSN 2.110', '$current\_time', '$current\_time')";              break;          case '4':              $sql = "INSERT INTO course1 (id, semester, course\_num\_section, title, credit\_hours, instructor, instructor\_netid, date\_time, location, time\_created, time\_updated)                      VALUES ($next\_id, '25SOpen', 'CS 6320.501', 'Natural Language Processing', '3', 'Sarah Williams', 'sxw789012', 'Online Asynchronous', 'Virtual', '$current\_time', '$current\_time')";              break;          case '5':              $sql = "INSERT INTO course1 (id, semester, course\_num\_section, title, credit\_hours, instructor, instructor\_netid, date\_time, location, time\_created, time\_updated)                      VALUES ($next\_id, '25FOpen', 'CS 6360.002', 'Database Design', '3', 'Michael Brown', 'mxb345678', 'Tuesday & Thursday 4:00pm - 5:15pm', 'ECSS 4.910', '$current\_time', '$current\_time')";              break;      }      if ($conn->query($sql) === TRUE) {          echo "<div class='alert alert-success'>Test Case " . $testCase . " added successfully</div>";          // Save page state to file with operation type "add"          savePageAsFile("add", "TestCase" . $testCase);      } else {          echo "<div class='alert alert-danger'>Error: " . $sql . "<br>" . $conn->error . "</div>";      }  }  // Process test case updates (1-5)  // Process test case updates (1-5)  if (isset($\_GET['updateTestCase'])) {      $testCase = $\_GET['updateTestCase'];      $current\_time = date('Y-m-d H:i:s');      switch ($testCase) {          case '1':              // Find course with course\_num\_section CS 6314.001              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 6314.001'");              if ($result->num\_rows > 0) {                  $row = $result->fetch\_assoc();                  $id = $row['id'];                  $sql = "UPDATE course1 SET                          title='Advanced Web Programming',                          instructor='John Smith Jr.',                          instructor\_netid='jxs999999',                          time\_updated='$current\_time'                          WHERE id=$id";              } else {                  echo "<div class='alert alert-danger'>Error: Test Case 1 (CS 6314.001) not found in database</div>";                  break;              }              break;          case '2':              // Find course with course\_num\_section CS 6363.001              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 6363.001'");              if ($result->num\_rows > 0) {                  $row = $result->fetch\_assoc();                  $id = $row['id'];                  $sql = "UPDATE course1 SET                          date\_time='Monday & Wednesday 4:00pm - 5:15pm',                          location='ECSS 4.910',                          time\_updated='$current\_time'                          WHERE id=$id";              } else {                  echo "<div class='alert alert-danger'>Error: Test Case 2 (CS 6363.001) not found in database</div>";                  break;              }              break;          case '3':              // Find course with course\_num\_section CS 5343.001              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 5343.001'");              if ($result->num\_rows > 0) {                  $row = $result->fetch\_assoc();                  $id = $row['id'];                  $sql = "UPDATE course1 SET                          semester='25FOpen',                          credit\_hours='4',                          time\_updated='$current\_time'                          WHERE id=$id";              } else {                  echo "<div class='alert alert-danger'>Error: Test Case 3 (CS 5343.001) not found in database</div>";                  break;              }              break;          case '4':              // Find course with course\_num\_section CS 6320.501              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 6320.501'");              if ($result->num\_rows > 0) {                  $row = $result->fetch\_assoc();                  $id = $row['id'];                  $sql = "UPDATE course1 SET                          date\_time='Monday 7:00pm - 9:45pm',                          location='ECSS 2.415',                          instructor='Sarah Johnson',                          instructor\_netid='sxj789555',                          time\_updated='$current\_time'                          WHERE id=$id";              } else {                  echo "<div class='alert alert-danger'>Error: Test Case 4 (CS 6320.501) not found in database</div>";                  break;              }              break;          case '5':              // Find course with course\_num\_section CS 6360.002              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 6360.002'");              if ($result->num\_rows > 0) {                  $row = $result->fetch\_assoc();                  $id = $row['id'];                  $sql = "UPDATE course1 SET                          title='Advanced Database Systems',                          course\_num\_section='CS 6360.001',                          time\_updated='$current\_time'                          WHERE id=$id";              } else {                  echo "<div class='alert alert-danger'>Error: Test Case 5 (CS 6360.002) not found in database</div>";                  break;              }              break;      }      if (isset($sql) && $conn->query($sql) === TRUE) {          echo "<div class='alert alert-success'>Test Case " . $testCase . " updated successfully</div>";          // Save page state to file with operation type "update"          savePageAsFile("update", "TestCase" . $testCase);      } else if (isset($sql)) {          echo "<div class='alert alert-danger'>Error: " . $sql . "<br>" . $conn->error . "</div>";      }  }  // Process delete request  // Process delete request  if (isset($\_GET['delete'])) {      $id = $\_GET['delete'];        // Get course info before deleting for the filename      $result = $conn->query("SELECT \* FROM course1 WHERE id=$id");      $row = $result->fetch\_assoc();      $course\_info = $row['course\_num\_section'];        $sql = "DELETE FROM course1 WHERE id=$id";      if ($conn->query($sql) === TRUE) {          echo "<div class='alert alert-success'>Course deleted successfully</div>";      } else {          echo "<div class='alert alert-danger'>Error: " . $sql . "<br>" . $conn->error . "</div>";      }  }  // Process test case deletions (1-5)  if (isset($\_GET['deleteTestCase'])) {      $testCase = $\_GET['deleteTestCase'];      switch ($testCase) {          case '1':              // Find course with course\_num\_section CS 6314.001              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 6314.001'");              break;          case '2':              // Find course with course\_num\_section CS 6363.001              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 6363.001'");              break;          case '3':              // Find course with course\_num\_section CS 5343.001              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 5343.001'");              break;          case '4':              // Find course with course\_num\_section CS 6320.501 or updated to Sarah Johnson              $result = $conn->query("SELECT \* FROM course1 WHERE course\_num\_section='CS 6320.501' OR instructor\_netid='sxj789555'");              break;          case '5':              // Find course that was Database Design or now Advanced Database Systems              $result = $conn->query("SELECT \* FROM course1 WHERE title='Database Design' OR title='Advanced Database Systems'");              break;      }      if (isset($result) && $result->num\_rows > 0) {          $row = $result->fetch\_assoc();          $id = $row['id'];          $course\_info = $row['course\_num\_section'] . ' - ' . $row['title'];          $sql = "DELETE FROM course1 WHERE id=$id";          if ($conn->query($sql) === TRUE) {              echo "<div class='alert alert-success'>Test Case " . $testCase . " (" . $course\_info . ") deleted successfully</div>";              // Save page state to file with operation type "delete"              savePageAsFile("delete", "TestCase" . $testCase);          } else {              echo "<div class='alert alert-danger'>Error: " . $sql . "<br>" . $conn->error . "</div>";          }      } else {          echo "<div class='alert alert-danger'>Error: Test Case " . $testCase . " not found in database</div>";      }  }  // Get course details for edit  if (isset($\_GET['edit'])) {      $id = $\_GET['edit'];      $update = true;      $result = $conn->query("SELECT \* FROM course1 WHERE id=$id");      if ($result->num\_rows == 1) {          $row = $result->fetch\_array();          $id = $row['id'];          $semester = $row['semester'];          $course\_num\_section = $row['course\_num\_section'];          $title = $row['title'];          $credit\_hours = $row['credit\_hours'];          $instructor = $row['instructor'];          $instructor\_netid = $row['instructor\_netid'];          $date\_time = $row['date\_time'];          $location = $row['location'];      }  }  // Get all courses - ordered by course\_num\_section alphabetically  $result = $conn->query("SELECT \* FROM course1 ORDER BY course\_num\_section");  ?>  <!DOCTYPE html>  <html>  <head>      <title>CS6314 - Task 2</title>      <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">      <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>      <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>      <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>      <style>          .container {              margin-top: 30px;          }          .form-group {              margin-bottom: 15px;          }          .btn-action {              margin-right: 5px;              margin-bottom: 5px;          }          .alert {              margin-bottom: 20px;          }          .test-case-buttons {              margin-bottom: 20px;          }          .btn-group-vertical {              margin-right: 10px;          }          .test-cases-container {              display: flex;              flex-wrap: wrap;              margin-bottom: 20px;          }          .test-case-column {              flex: 1;              min-width: 300px;              margin-bottom: 15px;          }          .test-case-column h5 {              margin-bottom: 10px;          }      </style>  </head>  <body>      <div class="container">          <h2>CS6314 Course Management</h2>          <!-- Test Case Buttons -->          <div class="test-case-buttons">              <h4>Test Cases</h4>              <div class="test-cases-container">                  <div class="test-case-column">                      <h5>Add Test Cases</h5>                      <div class="btn-group-vertical">                          <button type="button" class="btn btn-info btn-action" onclick="addTestCase(1)">Add Test Case 1:                              CS 6314.001</button>                          <button type="button" class="btn btn-info btn-action" onclick="addTestCase(2)">Add Test Case 2:                              CS 6363.001</button>                          <button type="button" class="btn btn-info btn-action" onclick="addTestCase(3)">Add Test Case 3:                              CS 5343.001</button>                          <button type="button" class="btn btn-info btn-action" onclick="addTestCase(4)">Add Test Case 4:                              CS 6320.501</button>                          <button type="button" class="btn btn-info btn-action" onclick="addTestCase(5)">Add Test Case 5:                              CS 6360.002</button>                      </div>                  </div>                  <div class="test-case-column">                      <h5>Update Test Cases</h5>                      <div class="btn-group-vertical">                          <button type="button" class="btn btn-warning btn-action" onclick="updateTestCase(1)">Update Test                              Case 1</button>                          <button type="button" class="btn btn-warning btn-action" onclick="updateTestCase(2)">Update Test                              Case 2</button>                          <button type="button" class="btn btn-warning btn-action" onclick="updateTestCase(3)">Update Test                              Case 3</button>                          <button type="button" class="btn btn-warning btn-action" onclick="updateTestCase(4)">Update Test                              Case 4</button>                          <button type="button" class="btn btn-warning btn-action" onclick="updateTestCase(5)">Update Test                              Case 5</button>                      </div>                  </div>                  <div class="test-case-column">                      <h5>Delete Test Cases</h5>                      <div class="btn-group-vertical">                          <button type="button" class="btn btn-danger btn-action" onclick="deleteTestCase(1)">Delete Test                              Case 1</button>                          <button type="button" class="btn btn-danger btn-action" onclick="deleteTestCase(2)">Delete Test                              Case 2</button>                          <button type="button" class="btn btn-danger btn-action" onclick="deleteTestCase(3)">Delete Test                              Case 3</button>                          <button type="button" class="btn btn-danger btn-action" onclick="deleteTestCase(4)">Delete Test                              Case 4</button>                          <button type="button" class="btn btn-danger btn-action" onclick="deleteTestCase(5)">Delete Test                              Case 5</button>                      </div>                  </div>              </div>          </div>          <!-- Input Form -->          <div class="row">              <div class="col-md-4">                  <form method="post" action="task2.php" id="courseForm">                      <input type="hidden" name="id" value="<?php echo $id; ?>">                      <div class="form-group">                          <label>Semester</label>                          <input type="text" name="semester" class="form-control" value="<?php echo $semester; ?>"                              required>                      </div>                      <div class="form-group">                          <label>Course Number & Section</label>                          <input type="text" name="course\_num\_section" class="form-control"                              value="<?php echo $course\_num\_section; ?>" required>                      </div>                      <div class="form-group">                          <label>Title</label>                          <input type="text" name="title" class="form-control" value="<?php echo $title; ?>" required>                      </div>                      <div class="form-group">                          <label>Credit Hours</label>                          <input type="text" name="credit\_hours" class="form-control" value="<?php echo $credit\_hours; ?>"                              required>                      </div>                      <div class="form-group">                          <label>Instructor</label>                          <input type="text" name="instructor" class="form-control" value="<?php echo $instructor; ?>"                              required>                      </div>                      <div class="form-group">                          <label>Instructor NetID</label>                          <input type="text" name="instructor\_netid" class="form-control"                              value="<?php echo $instructor\_netid; ?>" required>                      </div>                      <div class="form-group">                          <label>Date & Time</label>                          <input type="text" name="date\_time" class="form-control" value="<?php echo $date\_time; ?>"                              required>                      </div>                      <div class="form-group">                          <label>Location</label>                          <input type="text" name="location" class="form-control" value="<?php echo $location; ?>"                              required>                      </div>                      <div class="form-group">                          <?php if ($update == true): ?>                              <button type="submit" name="save" class="btn btn-primary">Update</button>                              <button type="button" class="btn btn-secondary" onclick="clearForm()">Cancel</button>                          <?php else: ?>                              <button type="submit" name="save" class="btn btn-success">Save</button>                              <button type="button" class="btn btn-secondary" onclick="clearForm()">Clear</button>                          <?php endif ?>                      </div>                  </form>              </div>              <!-- Course Table -->              <div class="col-md-8">                  <table class="table table-bordered">                      <thead>                          <tr>                              <th>S.No</th>                              <th>Semester</th>                              <th>Course</th>                              <th>Title</th>                              <th>Credits</th>                              <th>Instructor</th>                              <th>Date & Time</th>                              <th>Location</th>                              <th>Actions</th>                          </tr>                      </thead>                      <tbody>                          <?php                          $serial\_number = 1;                          while ($row = $result->fetch\_assoc()):                              ?>                              <tr>                                  <td><?php echo $serial\_number++; ?></td>                                  <td><?php echo $row['semester']; ?></td>                                  <td><?php echo $row['course\_num\_section']; ?></td>                                  <td><?php echo $row['title']; ?></td>                                  <td><?php echo $row['credit\_hours']; ?></td>                                  <td><?php echo $row['instructor']; ?> (<?php echo $row['instructor\_netid']; ?>)</td>                                  <td><?php echo $row['date\_time']; ?></td>                                  <td><?php echo $row['location']; ?></td>                                  <td>                                      <a href="task2.php?edit=<?php echo $row['id']; ?>"                                          class="btn btn-warning btn-sm btn-action">Edit</a>                                      <a href="#" onclick="confirmDelete(<?php echo $row['id']; ?>)"                                          class="btn btn-danger btn-sm btn-action">Delete</a>                                  </td>                              </tr>                          <?php endwhile; ?>                      </tbody>                  </table>              </div>          </div>      </div>      <script>          // Function to clear form fields          function clearForm() {              document.getElementById('courseForm').reset();              document.querySelector('input[name="id"]').value = '';              window.location.href = 'task2.php'; // Redirect to clear the edit state          }          // Test case functions for adding courses          function addTestCase(num) {              let courseInfo = "";              switch (num) {                  case 1:                      courseInfo = "CS 6314.001 - Web Programming Languages";                      break;                  case 2:                      courseInfo = "CS 6363.001 - Design and Analysis of Algorithms";                      break;                  case 3:                      courseInfo = "CS 5343.001 - Data Structures and Algorithms";                      break;                  case 4:                      courseInfo = "CS 6320.501 - Natural Language Processing";                      break;                  case 5:                      courseInfo = "CS 6360.002 - Database Design";                      break;              }              if (confirm("Add Test Case " + num + ": " + courseInfo + "?")) {                  window.location.href = "task2.php?addTestCase=" + num;              }          }          // Test case functions for updating courses          function updateTestCase(num) {              let updateInfo = "";              switch (num) {                  case 1:                      updateInfo = "Modifying CS 6314.001 course title and instructor";                      break;                  case 2:                      updateInfo = "Modifying CS 6363.001 course location and time";                      break;                  case 3:                      updateInfo = "Modifying CS 5343.001 semester and credit hours";                      break;                  case 4:                      updateInfo = "Modifying CS 6320.501 instructor, time and location";                      break;                  case 5:                      updateInfo = "Modifying CS 6360.002 title and section number";                      break;              }              if (confirm("Update Test Case " + num + ": " + updateInfo + "?")) {                  window.location.href = "task2.php?updateTestCase=" + num;              }          }          // Test case functions for deleting courses          function deleteTestCase(num) {              let courseInfo = "";              switch (num) {                  case 1:                      courseInfo = "CS 6314.001";                      break;                  case 2:                      courseInfo = "CS 6363.001";                      break;                  case 3:                      courseInfo = "CS 5343.001";                      break;                  case 4:                      courseInfo = "CS 6320.501";                      break;                  case 5:                      courseInfo = "CS 6360.002/001";                      break;              }              if (confirm("Delete Test Case " + num + ": Remove " + courseInfo + " course?\n\nPress OK to continue or Cancel to abort.")) {                  window.location.href = "task2.php?deleteTestCase=" + num;              }          }          // Confirmation for regular delete          function confirmDelete(id) {              if (confirm("Are you sure you want to delete this course?\n\nPress OK to continue or Cancel to abort.")) {                  window.location.href = "task2.php?delete=" + id;              }          }      </script>  </body>  </html> |

## 2.2 Screenshots of Website flow

Screenshots of Website Flow

|  |
| --- |
| 2.2.1 Displaying all courses on the home page as mentioned    2.2.2 If any test case button is clicked a warning tab will open to confirm changes    2.2.3 Once the ‘Ok’ is pressed it will give a prompt that it is added and added into the database (CS 6314.001)    2.2.4 You will see task1out.html downloaded after clicking okay    2.2.5 Writing the information to add the course    2.2.6 We can see the course in data after clicking save.    2.2.7 After clicking edit button all the details will be on the input text tab we will change credit hours to 4.    2.2.8 Credit hours updated to 4    2.2.9 After clicking delete we will get warning box to confirm    2.2.10 Once we press ‘Ok’ the course will be deleted |

## 

## 2.3 Screenshots for Test Cases

Screenshots for Test Cases  
  
Note: I will be adding 2 pictures for before which will be for SQL database and website and will be adding 4 pictures for after which will include SQL database, Website, comfirmation and proof of files download

|  |
| --- |
| Before add test case 1        After add test case 1          After update case 1          After delete case 1      Before test case 2          After add test case 2          After Update test case 2          After Delete test case 2      Before test case 3        After add test case 3          After Update test case 3          After Delete test case 3      Before test case 4          After add test case 4          After Update test case 4          After Delete test case 4      Before test case 5          After add test case 5          After Update test case 5          After Delete test case 5 |

## 2.4 Listing of each file

Listing of each file or (as needed or if missed)

|  |
| --- |
| 1. task2.php – Code to add, update and delete course. Also adding test cases 2. test1out.html – Add test case 1 3. test2out.html – Add test case 2 4. test3out.html – Add test case 3 5. test4out.html – Add test case 4 6. test5out.html – Add test case 5 7. test6out.html – Update test case 1 8. test7out.html – Update test case 2 9. test8out.html – Update test case 3 10. test9out.html – Update test case 4 11. test10out.html – Update test case 5 12. test11out.html – Delete test case 1 13. test12out.html – Delete test case 2 14. test13out.html – Delete test case 3 15. test14out.html – Delete test case 4 16. test15out.html – Delete test case 5 |

# Task3 Solution

## 3.1 task3.php

|  |
| --- |
| <?php  // Database connection settings  $servername = "localhost";  $username = "root"; // Default XAMPP username  $password = ""; // Default XAMPP password  $dbname = "cs6314"; // Your database name  // Create connection  $conn = new mysqli($servername, $username, $password, $dbname);  // Check connection  if ($conn->connect\_error) {      die("Connection failed: " . $conn->connect\_error);  }  // Initialize variables  $id = "";  $semester = "";  $course\_num\_section = "";  $title = "";  $credit\_hours = "";  $instructor = "";  $instructor\_netid = "";  $date\_time = "";  $location = "";  $update = false;  $page = isset($\_GET['page']) ? $\_GET['page'] : 'courses';  // Process form submission for adding a course  if (isset($\_POST['add\_course'])) {      $semester = $\_POST['semester'];      $course\_num\_section = $\_POST['course\_num\_section'];      $title = $\_POST['title'];      $credit\_hours = $\_POST['credit\_hours'];      $instructor = $\_POST['instructor'];      $instructor\_netid = $\_POST['instructor\_netid'];      $date\_time = $\_POST['date\_time'];      $location = $\_POST['location'];      $current\_time = date('Y-m-d H:i:s');      // Get the max ID and increment by 1      $result = $conn->query("SELECT MAX(id) as max\_id FROM course1");      $row = $result->fetch\_assoc();      $next\_id = $row['max\_id'] + 1;      $sql = "INSERT INTO course1 (id, semester, course\_num\_section, title, credit\_hours, instructor, instructor\_netid, date\_time, location, time\_created, time\_updated)              VALUES ($next\_id, '$semester', '$course\_num\_section', '$title', '$credit\_hours', '$instructor', '$instructor\_netid', '$date\_time', '$location', '$current\_time', '$current\_time')";      if ($conn->query($sql) === TRUE) {          $success\_message = "Course added successfully";          // Redirect to courses page after adding          header("Location: task3.php?page=courses&message=" . urlencode($success\_message));          exit();      } else {          $error\_message = "Error: " . $sql . "<br>" . $conn->error;      }  }  // Process form submission for updating a course  if (isset($\_POST['update\_course'])) {      $id = $\_POST['id'];      $semester = $\_POST['semester'];      $course\_num\_section = $\_POST['course\_num\_section'];      $title = $\_POST['title'];      $credit\_hours = $\_POST['credit\_hours'];      $instructor = $\_POST['instructor'];      $instructor\_netid = $\_POST['instructor\_netid'];      $date\_time = $\_POST['date\_time'];      $location = $\_POST['location'];      $current\_time = date('Y-m-d H:i:s');      $sql = "UPDATE course1 SET              semester='$semester',              course\_num\_section='$course\_num\_section',              title='$title',              credit\_hours='$credit\_hours',              instructor='$instructor',              instructor\_netid='$instructor\_netid',              date\_time='$date\_time',              location='$location',              time\_updated='$current\_time'              WHERE id=$id";      if ($conn->query($sql) === TRUE) {          $success\_message = "Course updated successfully";          // Redirect to courses page after updating          header("Location: task3.php?page=courses&message=" . urlencode($success\_message));          exit();      } else {          $error\_message = "Error: " . $sql . "<br>" . $conn->error;      }  }  // Process delete request  if (isset($\_GET['delete'])) {      $id = $\_GET['delete'];      $sql = "DELETE FROM course1 WHERE id=$id";      if ($conn->query($sql) === TRUE) {          $success\_message = "Course deleted successfully";          // Redirect to courses page after deleting          header("Location: task3.php?page=courses&message=" . urlencode($success\_message));          exit();      } else {          $error\_message = "Error: " . $sql . "<br>" . $conn->error;      }  }  // Get course details for edit  if (isset($\_GET['edit'])) {      $id = $\_GET['edit'];      $page = 'edit'; // Force the page to be edit      $result = $conn->query("SELECT \* FROM course1 WHERE id=$id");      if ($result->num\_rows == 1) {          $row = $result->fetch\_array();          $id = $row['id'];          $semester = $row['semester'];          $course\_num\_section = $row['course\_num\_section'];          $title = $row['title'];          $credit\_hours = $row['credit\_hours'];          $instructor = $row['instructor'];          $instructor\_netid = $row['instructor\_netid'];          $date\_time = $row['date\_time'];          $location = $row['location'];      }  }  // Search functionality  $search\_query = "";  if (isset($\_GET['search'])) {      $search\_query = $\_GET['search'];      $search\_result = $conn->query("SELECT \* FROM course1 WHERE                                      course\_num\_section LIKE '%$search\_query%' OR                                      title LIKE '%$search\_query%' OR                                      instructor LIKE '%$search\_query%' OR                                      date\_time LIKE '%$search\_query%'                                      ORDER BY course\_num\_section");      // For autocomplete JSON response      if (isset($\_GET['autocomplete']) && $\_GET['autocomplete'] == 'true') {          $suggestions = array();          while ($row = $search\_result->fetch\_assoc()) {              $suggestions[] = array(                  'id' => $row['id'],                  'value' => $row['course\_num\_section'] . ' - ' . $row['title']              );          }          header('Content-Type: application/json');          echo json\_encode($suggestions);          exit();      }  } else {      // Get all courses - ordered by course\_num\_section alphabetically      $search\_result = $conn->query("SELECT \* FROM course1 ORDER BY course\_num\_section");  }  ?>  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>CS6314 - Course Management System</title>      <!-- Bootstrap CSS -->      <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/css/bootstrap.min.css">      <!-- Font Awesome for icons -->      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css">      <!-- jQuery UI for autocomplete -->      <link rel="stylesheet" href="https://code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css">      <!-- Custom CSS -->      <link rel="stylesheet" href="task3.css">  </head>  <body>      <!-- Header -->      <header>          <div class="container-fluid">              <div class="row">                  <div class="col-md-12">                      <div class="utd-header">                          <svg viewBox="0 0 40 32" class="utd-logo" aria-labelledby="utd-logo-title">                              <title id="utd-logo-title">UTD Logo</title>                              <path                                  d="m32.7 4.5-.7.1c-.3-.7-1.1-1.1-1.8-1h-10v27c0 1.1.5 1.5 1.1 1.5l.1.5h-6.4l.1-.5c.6 0 1.2-.4 1.2-1.5v-27h-9.8c-.8-.1-1.5.3-1.9 1l-.6-.1 1-4.1c.5.1 1 .1 1.6.1h23.8c.4 0 .9 0 1.3-.1zm-3.5 24.3c-2.8 0-6.2-.1-7-.2v-.6c.5.1 1-.4 1.1-.9v-19.4c0-.5-.4-1-.9-1.1h-.3v-.6c1.1-.1 4.6-.2 7-.2 5.1 0 7.9 2.7 7.9 10.9-.1 8.3-2.3 12.1-7.8 12.1zm.7-19.8h-2.7v16.4h2.4c2.1 0 3.1-2.5 3.1-8.9 0-5.9-1.2-7.5-2.8-7.5zm-15.2-2.4v-.6h-4.8c-.3.5-.5 1.2-.4 1.8v16.2c0 1-.8 1.9-1.8 1.9h-.2c-1 .1-1.9-.6-2-1.6 0-.1 0-.2 0-.3v-16.2c-.1-.6.3-1.1.9-1.2h.2v-.6h-4.5c-.5.5-.7 1.1-.7 1.8v16.5c0 .6 0 1.3-.2 1.9 1.5 1.9 3.8 3 6.2 2.9 2.4 0 4.8-.2 7.2-.5v-.6c-.5.1-1-.3-1.1-.9v-.1-19.3c0-.5.4-1 .9-1.1z"                                  fill="currentColor">                              </path>                          </svg>                          <h1>The University of Texas at Dallas</h1>                      </div>                  </div>              </div>          </div>      </header>      <!-- Navigation -->      <nav class="navbar navbar-expand-lg navbar-dark">          <div class="container">              <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarMain">                  <span class="navbar-toggler-icon"></span>              </button>              <div class="collapse navbar-collapse" id="navbarMain">                  <ul class="navbar-nav mr-auto">                      <li class="nav-item <?php echo ($page == 'courses') ? 'active' : ''; ?>">                          <a class="nav-link" href="task3.php?page=courses">Courses & Search</a>                      </li>                      <li class="nav-item <?php echo ($page == 'add') ? 'active' : ''; ?>">                          <a class="nav-link" href="task3.php?page=add">Add Course</a>                      </li>                      <?php if ($page == 'edit'): ?>                          <li class="nav-item active">                              <a class="nav-link" href="#">Edit Course</a>                          </li>                      <?php endif; ?>                  </ul>              </div>          </div>      </nav>      <!-- Main Content -->      <div class="container main-content mt-4">          <?php if (isset($\_GET['message'])): ?>              <div class="alert alert-success alert-dismissible fade show">                  <?php echo htmlspecialchars($\_GET['message']); ?>                  <button type="button" class="close" data-dismiss="alert">&times;</button>              </div>          <?php endif; ?>          <?php if (isset($error\_message)): ?>              <div class="alert alert-danger alert-dismissible fade show">                  <?php echo $error\_message; ?>                  <button type="button" class="close" data-dismiss="alert">&times;</button>              </div>          <?php endif; ?>          <?php if ($page == 'courses'): ?>              <!-- PAGE 1: COURSES AND SEARCH -->              <div class="row">                  <div class="col-12">                      <!-- Search Section -->                      <section id="search" class="mb-4">                          <div class="card">                              <div class="card-header bg-primary text-white">                                  <h4><i class="fas fa-search"></i> Search Courses</h4>                              </div>                              <div class="card-body">                                  <form method="get" action="task3.php" class="form-inline mb-3">                                      <input type="hidden" name="page" value="courses">                                      <div class="input-group w-100">                                          <input type="text" id="searchBox" name="search" class="form-control"                                              placeholder="Search by course number, title, instructor, or time"                                              value="<?php echo $search\_query; ?>">                                          <div class="input-group-append">                                              <button class="btn btn-primary" type="submit"><i class="fas fa-search"></i>                                                  Search</button>                                          </div>                                      </div>                                  </form>                                  <div id="searchHelp" class="text-muted small">                                      Type to search. The system will suggest matching courses as you type.                                  </div>                              </div>                          </div>                      </section>                      <!-- Course List Section -->                      <section id="courses">                          <div class="card">                              <div class="card-header bg-primary text-white">                                  <h4><i class="fas fa-graduation-cap"></i> Course Listings</h4>                              </div>                              <div class="card-body">                                  <?php if ($search\_result->num\_rows > 0): ?>                                      <div class="row course-list">                                          <?php while ($row = $search\_result->fetch\_assoc()): ?>                                              <div class="col-md-6 mb-3">                                                  <div class="course-card" data-course-id="<?php echo $row['id']; ?>">                                                      <div class="course-header">                                                          <?php echo $row['course\_num\_section']; ?>: <?php echo $row['title']; ?>                                                      </div>                                                      <div class="course-details">                                                          <div><strong>Semester:</strong> <?php echo $row['semester']; ?></div>                                                          <div><strong>Credits:</strong> <?php echo $row['credit\_hours']; ?></div>                                                          <div><strong>Instructor:</strong> <?php echo $row['instructor']; ?>                                                              (<?php echo $row['instructor\_netid']; ?>)</div>                                                          <div><strong>Schedule:</strong> <?php echo $row['date\_time']; ?></div>                                                          <div><strong>Location:</strong> <?php echo $row['location']; ?></div>                                                          <div class="course-actions mt-2">                                                              <a href="task3.php?page=edit&edit=<?php echo $row['id']; ?>"                                                                  class="btn btn-warning btn-sm">                                                                  <i class="fas fa-edit"></i> Edit                                                              </a>                                                              <a href="#" onclick="confirmDelete(<?php echo $row['id']; ?>)"                                                                  class="btn btn-danger btn-sm">                                                                  <i class="fas fa-trash"></i> Delete                                                              </a>                                                          </div>                                                      </div>                                                  </div>                                              </div>                                          <?php endwhile; ?>                                      </div>                                  <?php else: ?>                                      <div class="alert alert-info">No courses found. Please add a new course or modify your                                          search.</div>                                  <?php endif; ?>                              </div>                          </div>                      </section>                  </div>              </div>          <?php elseif ($page == 'add'): ?>              <!-- PAGE 2: ADD COURSE -->              <div class="row justify-content-center">                  <div class="col-md-8">                      <div class="card">                          <div class="card-header bg-primary text-white">                              <h4><i class="fas fa-plus"></i> Add New Course</h4>                          </div>                          <div class="card-body">                              <form method="post" action="task3.php?page=add" id="addCourseForm">                                  <div class="form-group">                                      <label>Semester</label>                                      <input type="text" name="semester" class="form-control" required>                                  </div>                                  <div class="form-group">                                      <label>Course Number & Section</label>                                      <input type="text" name="course\_num\_section" class="form-control" required>                                  </div>                                  <div class="form-group">                                      <label>Title</label>                                      <input type="text" name="title" class="form-control" required>                                  </div>                                  <div class="form-group">                                      <label>Credit Hours</label>                                      <input type="text" name="credit\_hours" class="form-control" required>                                  </div>                                  <div class="form-group">                                      <label>Instructor</label>                                      <input type="text" name="instructor" class="form-control" required>                                  </div>                                  <div class="form-group">                                      <label>Instructor NetID</label>                                      <input type="text" name="instructor\_netid" class="form-control" required>                                  </div>                                  <div class="form-group">                                      <label>Date & Time</label>                                      <input type="text" name="date\_time" class="form-control" required>                                  </div>                                  <div class="form-group">                                      <label>Location</label>                                      <input type="text" name="location" class="form-control" required>                                  </div>                                  <div class="form-group">                                      <button type="submit" name="add\_course" class="btn btn-success btn-block">                                          <i class="fas fa-plus-circle"></i> Add Course                                      </button>                                      <button type="button" class="btn btn-secondary btn-block"                                          onclick="clearForm('addCourseForm')">                                          <i class="fas fa-eraser"></i> Clear Form                                      </button>                                  </div>                              </form>                          </div>                      </div>                  </div>              </div>          <?php elseif ($page == 'edit'): ?>              <!-- PAGE 3: EDIT COURSE -->              <div class="row justify-content-center">                  <div class="col-md-8">                      <div class="card">                          <div class="card-header bg-primary text-white">                              <h4><i class="fas fa-edit"></i> Edit Course</h4>                          </div>                          <div class="card-body">                              <form method="post" action="task3.php?page=edit" id="editCourseForm">                                  <input type="hidden" name="id" value="<?php echo $id; ?>">                                  <div class="form-group">                                      <label>Semester</label>                                      <input type="text" name="semester" class="form-control" value="<?php echo $semester; ?>"                                          required>                                  </div>                                  <div class="form-group">                                      <label>Course Number & Section</label>                                      <input type="text" name="course\_num\_section" class="form-control"                                          value="<?php echo $course\_num\_section; ?>" required>                                  </div>                                  <div class="form-group">                                      <label>Title</label>                                      <input type="text" name="title" class="form-control" value="<?php echo $title; ?>"                                          required>                                  </div>                                  <div class="form-group">                                      <label>Credit Hours</label>                                      <input type="text" name="credit\_hours" class="form-control"                                          value="<?php echo $credit\_hours; ?>" required>                                  </div>                                  <div class="form-group">                                      <label>Instructor</label>                                      <input type="text" name="instructor" class="form-control"                                          value="<?php echo $instructor; ?>" required>                                  </div>                                  <div class="form-group">                                      <label>Instructor NetID</label>                                      <input type="text" name="instructor\_netid" class="form-control"                                          value="<?php echo $instructor\_netid; ?>" required>                                  </div>                                  <div class="form-group">                                      <label>Date & Time</label>                                      <input type="text" name="date\_time" class="form-control"                                          value="<?php echo $date\_time; ?>" required>                                  </div>                                  <div class="form-group">                                      <label>Location</label>                                      <input type="text" name="location" class="form-control" value="<?php echo $location; ?>"                                          required>                                  </div>                                  <div class="form-group">                                      <button type="submit" name="update\_course" class="btn btn-primary btn-block">                                          <i class="fas fa-save"></i> Update Course                                      </button>                                      <a href="task3.php?page=courses" class="btn btn-secondary btn-block">                                          <i class="fas fa-times"></i> Cancel                                      </a>                                  </div>                              </form>                          </div>                      </div>                  </div>              </div>          <?php endif; ?>      </div>      <!-- Footer -->      <footer class="mt-5">          <div class="container">              <div class="row">                  <div class="col-md-4">                      <h5>CS6314 Project</h5>                      Name: Yash Mahendrakumar Wani<br>Net ID: yxw230008</p>                  </div>                  <div class="col-md-4">                      <h5>Quick Links</h5>                      <ul class="list-unstyled">                          <li><a href="task3.php?page=courses">Course Listings</a></li>                          <li><a href="task3.php?page=add">Add Course</a></li>                      </ul>                  </div>                  <div class="col-md-4">                      <h5>Contact</h5>                      <address>                          <strong>The University of Texas at Dallas</strong><br>                          800 W. Campbell Road<br>                          Richardson, TX 75080<br>&#128222  (972) 883-2111                      </address>                  </div>              </div>              <div class="row">                  <div class="col-md-12 text-center">                      <p>&copy; <?php echo date('Y'); ?> University of Texas at Dallas. All rights reserved.</p>                  </div>              </div>          </div>      </footer>      <!-- jQuery, Popper.js, and Bootstrap JS -->      <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>      <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.min.js"></script>      <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></script>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.min.js"></script>      <script>          $(document).ready(function () {              // Initialize course cards to show only header initially              $('.course-details').hide();              // Toggle course details on click              $('.course-header').click(function () {                  $(this).siblings('.course-details').slideToggle();              });              // Autocomplete for search box              $("#searchBox").autocomplete({                  source: function (request, response) {                      $.getJSON("task3.php?search=" + request.term + "&autocomplete=true", function (data) {                          response($.map(data, function (item) {                              return {                                  label: item.value,                                  value: item.value,                                  id: item.id                              };                          }));                      });                  },                  minLength: 2,                  select: function (event, ui) {                      // Scroll to the selected course                      if (ui.item) {                          var courseId = ui.item.id;                          var courseElement = $(".course-card[data-course-id='" + courseId + "']");                          if (courseElement.length > 0) {                              // Expand the details                              courseElement.find('.course-details').show();                              // Scroll to it                              $('html, body').animate({                                  scrollTop: courseElement.offset().top - 100                              }, 500);                              // Highlight it                              courseElement.addClass('highlight');                              setTimeout(function () {                                  courseElement.removeClass('highlight');                              }, 2000);                          }                      }                  }              });          });          // Function to clear form fields          function clearForm(formId) {              document.getElementById(formId).reset();          }          // Confirmation for delete          function confirmDelete(id) {              if (confirm("Are you sure you want to delete this course?\n\nPress OK to continue or Cancel to abort.")) {                  window.location.href = "task3.php?delete=" + id;              }          }          // Auto-dismiss alerts after 5 seconds          window.setTimeout(function () {              $(".alert-dismissible").fadeTo(500, 0).slideUp(500, function () {                  $(this).remove();              });          }, 5000);      </script>  </body>  </html> |

## 3.2 task3.css

|  |
| --- |
| /\* UTD Colors \*/  :root {      /\* Updated to match official color palette \*/      --utd-orange: #c75b12; /\* Flame Orange - correct \*/      --utd-green: #008542; /\* Eco Green - corrected from #154734 \*/      --utd-gray: #e9e9e9; /\* Keeping as is, close to Warm Gray 2 \*/      --utd-dark: #333333; /\* Keeping as is, close to Black \*/  }  body {      /\* UTD's default fonts are Times New Roman (serif) and Arial (sans serif) \*/      font-family: 'Times New Roman', serif;      background-color: #f8f9fa;      color: #333;      line-height: 1.6;      padding-bottom: 0;      margin: 0;  }  /\* Header Styles \*/  header {      background-color: white; /\* Brilliance White \*/      padding: 10px 0;      border-bottom: 4px solid var(--utd-green);      box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);  }  .utd-header {      display: flex;      align-items: center;      justify-content: center;      width: 100%;      text-align: center;  }  .utd-logo {      height: 60px;      margin-right: 15px;      color: var(--utd-orange); /\* Or whatever color you want the logo to be \*/  }  .utd-header h1 {      /\* Modesto Text is used for the University wordmark \*/      font-family: 'Modesto Text', 'Arial', sans-serif;      font-size: 1.8rem;      color: var(--utd-orange);      margin: 0;      font-weight: bold;  }  /\* Navigation Styles \*/  .navbar {      background-color: var(--utd-green);      padding: 0.5rem 1rem;      border-bottom: 3px solid var(--utd-orange);  }  .navbar-dark .navbar-nav .nav-link {      /\* Sans serif (Arial) is preferred for headlines/navigation \*/      font-family: 'Arial', sans-serif;      color: white;      font-weight: 500;      padding: 0.7rem 1rem;      transition: background-color 0.3s;  }  .navbar-dark .navbar-nav .nav-link:hover,  .navbar-dark .navbar-nav .nav-item.active .nav-link {      background-color: rgba(255, 255, 255, 0.1);      color: white;  }  /\* Main Content Styles \*/  .main-content {      padding: 20px 0;      min-height: calc(100vh - 250px);  }  /\* Card Styling \*/  .card {      border: none;      box-shadow: 0 0.125rem 0.25rem rgba(0, 0, 0, 0.075);      margin-bottom: 20px;      border-radius: 0.5rem;      overflow: hidden;  }  .card-header {      padding: 0.75rem 1.25rem;      border-bottom: none;  }  .card-header h4 {      /\* Sans serif (Arial) is preferred for headlines \*/      font-family: 'Arial', sans-serif;      margin: 0;      font-size: 1.25rem;  }  .card-body {      padding: 1.25rem;  }  /\* Course Card Styling \*/  .course-list {      margin-top: 10px;  }  .course-card {      background-color: white;      border-radius: 0.5rem;      box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);      margin-bottom: 20px;      overflow: hidden;      transition: box-shadow 0.3s;      height: 100%;  }  .course-card:hover {      box-shadow: 0 5px 15px rgba(0, 0, 0, 0.1);  }  .course-header {      /\* Sans serif (Arial) is preferred for headlines \*/      font-family: 'Arial', sans-serif;      background-color: var(--utd-orange);      color: white;      padding: 10px 15px;      font-weight: bold;      cursor: pointer;      transition: background-color 0.3s;  }  .course-header:hover {      background-color: #b04d0e;  }  .course-details {      padding: 15px;      background-color: white;  }  .course-actions {      text-align: right;  }  /\* Form Styling \*/  .form-control {      /\* Sans serif (Arial) is better for UI elements like forms \*/      font-family: 'Arial', sans-serif;      border-radius: 0.25rem;      font-size: 0.9rem;      padding: 0.375rem 0.75rem;  }  .form-control:focus {      box-shadow: 0 0 0 0.2rem rgba(199, 91, 18, 0.25);      border-color: var(--utd-orange);  }  .btn-primary, .bg-primary {      background-color: var(--utd-green) !important;      border-color: var(--utd-green) !important;  }  .btn-primary:hover {      background-color: #0e3424 !important;      border-color: #0e3424 !important;  }  .btn-success {      background-color: var(--utd-orange);      border-color: var(--utd-orange);  }  .btn-success:hover {      background-color: #b04d0e;      border-color: #b04d0e;  }  .btn-warning {      background-color: #f0ad4e;      border-color: #f0ad4e;      color: white;  }  .btn-danger {      background-color: #dc3545;      border-color: #dc3545;  }  /\* Search Box Styling \*/  .ui-autocomplete {      max-height: 200px;      overflow-y: auto;      overflow-x: hidden;      z-index: 9999;  }  /\* Sticky Sidebar \*/  .sticky-top {      top: 20px;      z-index: 1020;  }  /\* Animation for Course Highlighting \*/  .highlight {      animation: pulse 1s;  }  @keyframes pulse {      0% {          box-shadow: 0 0 0 0 rgba(199, 91, 18, 0.7);      }      70% {          box-shadow: 0 0 0 10px rgba(199, 91, 18, 0);      }      100% {          box-shadow: 0 0 0 0 rgba(199, 91, 18, 0);      }  }  /\* Footer Styles \*/  footer {      background-color: var(--utd-dark);      color: white;      padding: 40px 0 20px;      margin-top: 30px;  }  footer h5 {      /\* Sans serif (Arial) is preferred for headlines \*/      font-family: 'Arial', sans-serif;      color: var(--utd-orange);      font-weight: 600;      margin-bottom: 15px;  }  footer ul li {      margin-bottom: 8px;  }  footer a {      color: white;      text-decoration: none;      transition: color 0.3s;  }  footer a:hover {      color: var(--utd-orange);      text-decoration: none;  }  footer address {      line-height: 1.8;  }  /\* Responsive Adjustments \*/  @media (max-width: 991.98px) {      .utd-header h1 {          font-size: 1.5rem;          margin-top: 10px;      }        .utd-logo {          height: 50px;      }        .sticky-top {          position: relative;          top: 0;      }  }  @media (max-width: 767.98px) {      .course-card {          margin-bottom: 15px;      }        .utd-header {          flex-direction: column;          text-align: center;      }        .utd-logo {          margin-right: 0;          margin-bottom: 10px;      }        .utd-header h1 {          font-size: 1.3rem;      }  }  /\* Font Face declarations optimized for faster loading \*/  @font-face {      font-family: 'Minion Pro';      src: url('path-to-font/MinionPro-Regular.woff2') format('woff2');      font-weight: normal;      font-style: normal;      font-display: swap; /\* Improves page load time \*/  }  @font-face {      font-family: 'FF DIN';      src: url('path-to-font/FFDIN-Regular.woff2') format('woff2');      font-weight: normal;      font-style: normal;      font-display: swap; /\* Improves page load time \*/  }  /\* Modesto Text font for wordmark \*/  @font-face {      font-family: 'Modesto Text';      src: url('path-to-font/ModestoText-Regular.woff2') format('woff2');      font-weight: normal;      font-style: normal;      font-display: swap; /\* Improves page load time \*/  }  /\* Class for professional designers who have access to official fonts \*/  .professional-typography {      --serif-font: 'Minion Pro', 'Times New Roman', serif;      --sans-serif-font: 'FF DIN', 'Arial', sans-serif;  }  /\* Optimize performance \*/  img, video {      max-width: 100%;      height: auto;  }  /\* Preload critical fonts to improve performance \*/  .preload-fonts {      position: absolute;      width: 0;      height: 0;      overflow: hidden;      z-index: -1;  }  /\* Use font-display: swap for all custom fonts to prevent FOIT \*/  \* {      text-rendering: optimizeLegibility;  } |

## 3.3 Screenshots

Screenshots with proper heading to show that you have done the work for this task. State clearly for each screenshot on what it is about (to show your work for the task is done), including the screenshot(s) of database table.

|  |
| --- |
| 3.3.1 Created Header Section    3.3.2 Created Navigation    3.3.3 Created Footer    3.3.4 The complete Page View    3.3.5 When we click on the course title the description of the course maximizes and clicking again will minimize    3.3.6 We have edit and delete options there itself.    3.3.7 Once we click Edit a new navigation tab is made to make the changes.    3.3.8 we will change the credit hours to 4 and press ‘Update Course’    3.3.9 Once we update the course, we will see the notification for time being    3.3.10 You can see the change in the credits    3.3.11 Changes can be noticed in database    3.3.12 For the search section we can see the suggestion.    3.3.13 After clicking search only valid options will be displayed.    3.3.14 We can search by Title as well.    3.3.15 All courses will be displayed which has ‘Introduction’ word in title    3.3.16 We can search by the name of the professor    3.3.17 All the courses taken by the professor will be displayed    3.3.18 We can search for the courses by Days as well    3.3.19 We can search for the courses on time as well    3.3.20 We can add the course by navigating to ‘Add Course’ section    3.3.21 Added the details in the section    3.3.22 Press ‘Add Course’ to save the course    3.3.23 We will get notification for while and we can see the course in listings    3.3.24 Database is updated    3.3.25 Now we will delete the course    3.3.26 Warning before deleting the course    3.3.27 After clicking ‘Ok’ the course will be deleted    3.3.28 The course has been deleted from the database. |