ETF Assignment 2 Report

Introduction

The purpose of this report is to provide an overview and an evaluation of a web-based student record management system which acts as a comprehensive platform to record and manage import details for educational institutions. The report will focus on the methodologies used, functions of the web-based system etc.

Methodologies & languages used.

* Front End (HTML + Bootstrap)
* jQuery (5 different uses in the site)
* Ajax (the data fetch from the server)
* JSON (for communication)
* Back End (REST API)

Home Page

A screenshot of a computer

Description automatically generated with medium confidence

The front of the homepage is developed with html and bootstrap that includes a register form with a theme of blue buttons and a light gray and white background. 

Also, it consists of a navigation bar that consists of a link to the home and login pages.

These are some snippets of the code.

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Student Management System - Home</title>

  <!-- Bootstrap CSS -->

  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

</head>

<body>

  <!-- Navigation Bar -->

  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">

    <a class="navbar-brand" href="#">Student Management System</a>

    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

      <span class="navbar-toggler-icon"></span>

    </button>

    <div class="collapse navbar-collapse" id="navbarNav">

      <ul class="navbar-nav ml-auto">

        <li class="nav-item active">

          <a class="nav-link" href="home.html">Home</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" href="login.html">login</a>

        </li>

      </ul>

    </div>

  </nav>

  <!-- Hero Section -->

  <section class="jumbotron jumbotron-fluid text-center">

    <div class="container">

      <h1 class="display-4">Welcome to Student Management System</h1>

      <p class="lead">Manage student records in a convenient and efficient manner.</p>

      <a href="login.html" class="btn btn-primary btn-lg">Login</a>

    </div>

  </section>

  <!-- Registration Section -->

  <section class="container">

    <div class="row justify-content-center mt-5">

      <div class="col-md-6">

        <div class="card">

          <div class="card-header">

            <h4>Register</h4>

          </div>

          <div class="card-body">

            <form id = "registrationForm">

              <div class="form-group">

                <label for="username">Username</label>

                <input type="text" class="form-control" id="username" placeholder="Enter your username">

              </div>

              <div class="form-group">

                <label for="password"> Password</label>

                <input type="password" class="form-control" id="password" placeholder="Enter your password">

              </div>

              <div class="form-group">

                <label for="email"> Email</label>

                <input type="email" class="form-control" id="email" placeholder="Enter your email">

              </div>

              <button type="submit" class="btn btn-primary">Register</button>

            </form>

          </div>

        </div>

      </div>

    </div>

  </section>

The backend of the home webpage consists of Ajax codes and rest API that sends entered values by the user to the server that then stores those values in a table called admin and if the process is a success the webpage will display a message saying so or it will display an error message. And once the process is complete it will redirect the user to the login page.

Here are some snippets of the code.

$(document).ready(function () {

      // Submit event handler for the registration form

      $('#registrationForm').submit(function (event) {

        event.preventDefault();

        // Get the form input values

        var username= $('#username').val();

        var password = $('#password').val();

        var email = $('#email').val();

        var user = {

          username: username,

          password: password,

          email: email,

        };

Sending AJAX request to register the users.

 $.ajax({

          url: '/register-user',

          type: 'POST',

          contentType: 'application/json',

          data: JSON.stringify(user),

          success: function (response) {

            console.log('Data successfully submitted:', response);

          },

          error: function (error) {

            console.error('Error submitting data:', error);

          }

        });

Resetting form fields

$('#username').val('');

        $('#password').val('');

        $('#email').val('');

Student Registration Page

The user interface of the student registration page includes text input fields and select options to enter data of the student such as first and last names nearest city, course etc. And a table at the end displays student data so the user can see the data of all students entered. The same theme used in the home page is continued.

A screenshot of a registration form

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with low confidence

Html and bootstrap was used to develop the webpage, here are some snippets of the code

<section class="container mt-5">

    <h2>Student Registration</h2>

    <form id="registrationForm">

      <div class="form-group">

        <label for="sid">SID:</label>

        <input type="text" class="form-control" id="sid" placeholder="Enter SID" required>

      </div>

      <div class="form-group">

        <label for="firstName">First Name:</label>

        <input type="text" class="form-control" id="fName" placeholder="Enter First Name" required>

      </div>

      <div class="form-group">

        <label for="lastName">Last Name:</label>

        <input type="text" class="form-control" id="lName" placeholder="Enter Last Name" required>

      </div>

      <div class="form-group">

        <label for="email">Email:</label>

        <input type="email" class="form-control" id="email" placeholder="Enter Email" required>

      </div>

      <div class="form-group">

        <label for="city">Nearest City</label>

        <input type="text" class="form-control" id="city" placeholder="Enter the nearest city" required>

      </div>

      <br>

      <div class="form-group">

        <label for="city">Guardian</label>

        <input type="text" class="form-control" id="Guardian" placeholder="Enter the guardians name" required>

      </div>

      <br>

      <div class="form-group">

        <label for="Course">Course</label>

        <select class="form-group" id="Course">

          <option value="found">Foundation</option>

          <option value="hnd">HND</option>

          <option value="degree">Degree</option>

        </select>

      </div>

      <br>

      <div class="form-group">

        <label for="sub">Subject</label>

        <select class="form-group" id="subject">

          <option value="etf">ETF</option>+

          <option value="oop">OOP</option>

          <option value="dbms">ORDBMS</option>

          <option value="hci">HCI</option>

          <option value="ip">IP</option>

          <option value="se">SE</option>

        </select>

      </div>

      <br>

      <button type="submit" id="submitBtn" class="btn btn-primary">Submit</button>

    </form>

    <br>

    <a href="/showData.html">View Registered Students</a>

  </section>

Registered Students Table e

<!-- Registered Students Table -->

  <br>

  <h2>Registered Students Table</h2><br>

  <table id="studentTable" class="table">

      <thead>

          <tr>

              <th>ID</th>

              <th>First Name</th>

              <th>Last Name</th>

              <th>Email</th>

              <th>City</th>

              <th>Guardian</th>

              <th>Course</th>

              <th>Subject</th>

          </tr>

      </thead>

      <tbody id="studentTableBody">

          <!-- Table rows will be dynamically added here -->

      </tbody>

  </table>

same as the registration page ajax and rest API was used at the back end to pass entered values to the database.

$(document).ready(function () {

      // Function to fetch and display student data

      function fetchStudentData() {

        $.ajax({

          url: '/fetch-students',

          type: 'GET',

          success: function (response) {

            // Clear existing table rows

            $('#studentTable tbody').empty();

            // Iterate over the student data and populate the table

            response.forEach(function (studentData) {

              var row = `

                            <tr>

                                <td>${studentData.Sid}</td>

                                <td>${studentData.firstname}</td>

                                <td>${studentData.lastname}</td>

                                <td>${studentData.email}</td>

                                <td>${studentData.city}</td>

                                <td>${studentData.guardian}</td>

                                <td>${studentData.course}</td>

                                <td>${studentData.subject}</td>

                            </tr>`;

              $('#studentTable tbody').append(row);

            });

          },

          error: function (err) {

            console.log('Error fetching Student Data', err);

          }

        });

      }

// Submit event handler for the registration form

      $('#registrationForm').submit(function (event) {

        event.preventDefault();

        // Get the form input values

        var sid = $('#sid').val();

        var firstName = $('#fName').val();

        var lastName = $('#lName').val();

        var email = $('#email').val();

        var city = $('#city').val();

        var guardian = $('#Guardian').val();

        var course = $('#Course').val();

        var subject = $('#subject').val();

        var student = {

          sid: sid,

          firstname: firstName,

          lastname: lastName,

          email: email,

          city: city,

          guardian: guardian,

          course: course,

          subject: subject

        };

        // Send AJAX request to register the student

        $.ajax({

          url: '/register-students',

          type: 'POST',

          contentType: 'application/json',

          data: JSON.stringify(student),

          success: function (response) {

            console.log('Data successfully submitted:', response);

            // Fetch and display updated student data

            fetchStudentData();

          },

          error: function (error) {

            console.error('Error submitting data:', error);

          }

        });

        // Reset form fields

        $('#sid').val('');

        $('#fName').val('');

        $('#lName').val('');

        $('#email').val('');

        $('#city').val('');

        $('#Guardian').val('');

        $('#Course').val('');

        $('#subject').val('');

Login Page

Front end of this webpage is developed using html, CSS and bootstrap it consists of a navigation bar and a login form and footer that has been added to all pages. The same theme is continued here as well as on all pages.

A screenshot of a computer

Description automatically generated

The login form consists of two text fields for the user to enter the username and password respectively.

The back-end of the login page is made using php

<?php

// Database configuration

$servername = "localhost";

$username = "root";

$password = "";

$database = "studentmangemt";

// Establish a database connection

$connection = mysqli\_connect($servername, $username, $password, $database);

// Check if the connection was successful

if (!$connection) {

    die("Connection failed: " . mysqli\_connect\_error());

}

// Check if the form is submitted

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    // Retrieve the submitted username and password

    $username = $\_POST["username"];

    $password = $\_POST["password"];

    $sql = "INSERT INTO admin (username, password) VALUES ('$username', '$password')";

    if (mysqli\_query($connection, $sql)) {

        header("Location: index.html");

    } else {

        echo "Error inserting data: " . mysqli\_error($connection);

    }

    mysqli\_close($connection);

}

?>

Th code connects to a MySQL database using the provided server name, username, password, and database name.

After establishing the database connection, the code checks if the form has been submitted using the $\_server… variable. If the form is submitted via the POST method, it retrieves the entered username and password from the form fields.

// Establish a database connection

$connection = mysqli\_connect($servername, $username, $password, $database);

// Check if the connection was successful

if (!$connection) {

    die("Connection failed: " . mysqli\_connect\_error());

}

// Check if the form is submitted

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    // Retrieve the submitted username and password

    $username = $\_POST["username"];

    $password = $\_POST["password"];

    $sql = "INSERT INTO admin (username, password) VALUES ('$username', '$password')";

    if (mysqli\_query($connection, $sql)) {

        header("Location: index.html");

    } else {

        echo "Error inserting data: " . mysqli\_error($connection);

    }

    mysqli\_close($connection);

}

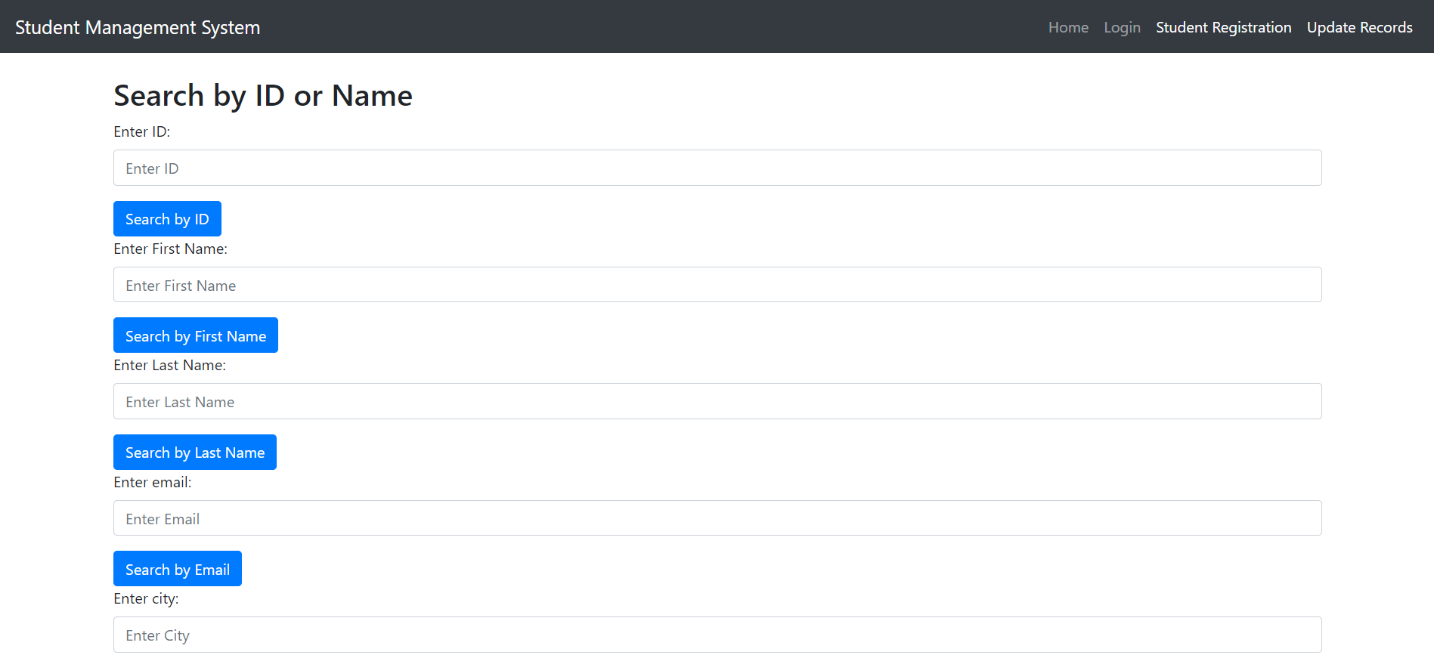
?>

The code then constructs an SQL query to insert the username and password into the “admin”

Table of the database. If the query execution is successful, the user is redirected to “index.html” Otherwise, an error message is showed.

Search Page

The user interface of the search page consists of text fields to insert different data of a record so the user can search records by different data. The UI is developed using html CSS and bootstrap. The same theme is continued using bootstrap.



A screenshot of a computer

Description automatically generated with medium confidence

A table is designed at the end of the page to display recorded data.

The back end of the page consists of ajax codes and rest API that passes the values entered by

the user to the database table named “studentdata”.

cript>

    $(document).ready(function () {

        // Submit search by ID form

        $('#searchIdBtn').click(function (event) {

            event.preventDefault();

            var searchId = $('#searchId').val();

            searchStudent('id', searchId);

        });

        // Submit search by fname form

        $('#searchFirstNameBtn').click(function (event) {

            event.preventDefault();

            var searchName = $('#searchFirstName').val();

            searchStudent('fname', searchName);

        });

        // Submit search by lname form

        $('#searchLastNameBtn').click(function (event) {

            event.preventDefault();

            var searchName = $('#searchLastName').val();

            searchStudent('lname', searchName);

        });

        // Submit search by email form

        $('#searchEmailBtn').click(function (event) {

            event.preventDefault();

            var searchEmail = $('#searchEmail').val();

            searchStudent('email', searchEmail);

        });

        // Submit search by city form

        $('#searchCityBtn').click(function (event) {

            event.preventDefault();

            var searchCity = $('#searchCity').val();

            searchStudent('city', searchCity);

        });

$.ajax({

                url: '/get-student/' + type + '/' + value,

                type: 'GET',

                success: function (response) {

                    console.log('Student details retrieved:', response);

                    // Display student details in the table

                    if (response.length > 0) {

                        for (var i = 0; i < response.length; i++) {

                            var student = response[i];

                            var row = '<tr>' +

                                '<td>' + student.Sid + '</td>' +

                                '<td>' + student.firstname + '</td>' +

                                '<td>' + student.lastname + '</td>' +

                                '<td>' + student.email + '</td>' +

                                '<td>' + student.city + '</td>' +

                                '<td>' + student.guardian + '</td>' +

                                '<td>' + student.course + '</td>' +

                                '<td>' + student.subject + '</td>' +

                                '<td><button class="deleteBtn" data-sid="' + student.Sid + '">Delete</button></td>' +

                                '</tr>';

                            $('#studentTableBody').append(row);

                        }

                    } else {

                        var row = '<tr><td colspan="4">No student found</td></tr>';

                        $('#studentTableBody').append(row);

                    }

                },

                error: function (error) {

                    console.error('Error retrieving student details:', error);

                }

            });

 // Attach click event handler to delete buttons

        $('#studentTableBody').on('click', '.deleteBtn', function () {

            var sid = $(this).data('sid');

            var rowElement = $(this).closest('tr');

            // Send AJAX request to delete student record

            $.ajax({

                url: '/delete-student/' + sid,

                type: 'DELETE',

                success: function (response) {

                    console.log('Student record deleted:', response);

                    // Remove the deleted row from the table

                    rowElement.remove();

                },

                error: function (error) {

                    console.error('Error deleting student record:', error);

                }

            });

        });

    });

Update Page

The purpose of the update page is for the user to change or update the records mainly by searching the records by either the ID or the first name. The updated records will be saved back in the studentdata table. The interface consists of text fields for the id and the first name and the individual records such as last name, city, guardian etc. will be displayed on the text fields below so the user can edit them the same theme used for the other pages that is made using bootstrap is continued.

A screenshot of a computer

Description automatically generated with low confidence

A picture containing text, screenshot, line, number

Description automatically generated

Ajax and JavaScript codes are used for the backend alongside with rest API. The backend facilitates the updating displaying of records via the connection of the database.

$(document).ready(function () {

        // Submit search by ID form

        $('#searchIdBtn').click(function (event) {

            event.preventDefault();

            var searchId = $('#searchId').val();

            searchStudent('id', searchId);

        });

        // Submit search by fname form

        $('#searchFirstNameBtn').click(function (event) {

            event.preventDefault();

            var searchName = $('#searchFirstName').val();

            searchStudent('fname', searchName);

        });

        function searchStudent(type, value) {

            // Clear table body

            $('#studentTableBody').empty();

            // Send AJAX request to fetch student details by ID or Name

            $.ajax({

                url: '/get-student/' + type + '/' + value,

                type: 'GET',

                success: function (response) {

                    console.log('Student details retrieved:', response);

                    // Display student details in the fields

                    if (response.length > 0) {

                        var student = response[0];

                        $('#updateId').val(student.Sid);

                        $('#updateFirstName').val(student.firstname);

                        $('#updateLastName').val(student.lastname);

                        $('#updateEmail').val(student.email);

                        $('#updateCity').val(student.city);

                        $('#updateGuardian').val(student.guardian);

                        $('#updateCourse').val(student.course);

                        $('#updateSubject').val(student.subject);

                    } else {

                        alert('No student found');

                    }

                },

                error: function (error) {

                    console.error('Error retrieving student details:', error);

                }

            });

        }

        $('#updateBtnById').click(function (event) {

            event.preventDefault();

            var updateId = $('#updateId').val();

            var updateFirstName = $('#updateFirstName').val();

            var updateLastName = $('#updateLastName').val();

            var updateEmail = $('#updateEmail').val();

            var updateCity = $('#updateCity').val();

            var updateGuardian = $('#updateGuardian').val();

            var updateCourse = $('#updateCourse').val();

            var updateSubject = $('#updateSubject').val();

            var updatedStudent = {

                firstname: updateFirstName,

                lastname: updateLastName,

                email: updateEmail,

                city: updateCity,

                guardian: updateGuardian,

                course: updateCourse,

                subject: updateSubject

            };

            $.ajax({

                url: '/update-student/' + updateId,

                type: 'PUT',

                data: JSON.stringify(updatedStudent),

                contentType: 'application/json',

                success: function (response) {

                    alert('Student record updated');

                    // Handle success response here

                },

                error: function (error) {

                    alert('Error updating student record:', error);

                    // Handle error response here

                }

            });

        });

        $('#updateBtnByFirstName').click(function (event) {

            event.preventDefault();

            var updateFirstName = $('#updateFirstName').val();

            var updateLastName = $('#updateLastName').val();

            var updateEmail = $('#updateEmail').val();

            var updateCity = $('#updateCity').val();

            var updateGuardian = $('#updateGuardian').val();

            var updateCourse = $('#updateCourse').val();

            var updateSubject = $('#updateSubject').val();

            var updatedStudent = {

                firstname: updateFirstName,

                lastname: updateLastName,

                email: updateEmail,

                city: updateCity,

                guardian: updateGuardian,

                course: updateCourse,

                subject: updateSubject

            };

            $.ajax({

                url: '/update-student-by-first-name/' + updateFirstName,

                type: 'PUT',

                data: JSON.stringify(updatedStudent),

                contentType: 'application/json',

                success: function (response) {

                    alert('Student record updated by name');

                    // Handle success response here

                },

                error: function (error) {

                    alert('Error updating student record: ' + error);

                    // Handle error response here

                }

Here’s a snippet of the backend.

jQuery effects

Four jQuery effects were added to the home page.

When the user clicks on the button click here to see features the section of the page slides down and when it is clicked again it appears.

 $(document).ready(function() {

    $("#cbtn").click(function(){

  $("#feature").slideToggle("slow");

});

});

When the user clicks on the about us button the background of the relevant section of the webpage will be highlighted in grey color.

$(document).ready(function(){

   $('#aboutbtn').click(function () {

               $('#ab').css('background-color', 'lightgray');

               $('#ab').css('border', '2px solid #333');

           });

         });

When the user clicks the contact us button the relevant section will slide up and down while highlighting the border of the webpage in red.

<script>

  $(document).ready(function(){

  $("#contactbtn").click(function(){

    $("#con").css("color", "grey").slideUp(2000).slideDown(2000);

  });

});

</script>

<script>

   $(document).ready(function(){

  $("#contactbtn").click(function () {

                var parent = $('#con').parent();

                parent.css('border', '2px solid red');

            });

          });

</script>

When the user clicks the register button when the process is successful the register form will hide it-self showing just the login button and the other contents on top.

<script>

  $(document).ready(function() {

    $("#btn").click(function(){

  $("#reg").slideUp("slow");

});

});

</script>

A single jQuery effect was added to the student registration page when the user clicks the view student data button the table below will fade in showing the updated records.

    $(document).ready(function (slideDown) {

      $("#viewDataBtn").click(function () {

        $("#studentTable").slideToggle("slow");

      });

    });