



STUDENT PORTAL WEB APPLICATION DOCUMENTATION

Kusal Akalanka Perera
15/07/2025

Table of Contents

1. Project Overview	3
Purpose	3
Key Features	3
Technologies Used	4
2. Folder Structure	4
3. Page Descriptions	5
Home page	5
Login & Registration Page(login.html)	6
Profile Page(profile.php)	6
Courses Page(course.php).....	8
Contact Page(contact.php).....	9
4. JavaScript Functions	10
5. PHP Backend Logic	10
Database Schema	10
Authentication Flow	11
Protected Pages	12
6. Challengers and Improvements	12
Challengers	12
Improvements	12
7. How to Run the Project	13
Prerequisites	13
Setup steps	13
8. Conclusion	22

1. Project Overview

Purpose

The Student Portal is a web-based application designed to provide students with a centralized platform to:

- Register and log in securely
- View and manage their academic and personal information
- View available courses
- Browse available courses with search functionality
- Contact administrators with inquiries or feedback.

The portal ensures a seamless experience for students by integrating user authentication, database management, and responsive design for accessibility across devices.

Key Features

Feature	Description
User Registration	Secure sign-up with email validation and password hashing
Login System	Session-bases authentication with optional “Remember-me” functionality
Profile Management	Edit personal and academic details (name, email, program, etc.)
Course catalog	Browse available courses with a searchable grid layout
Contact form	Submit inquiries with a star-rating system for feedback
Responsive UI	Works on desktops, tablets, and mobile devices
Light /Dark mode	Toggle between themes for better readability

Technologies Used

Technology	Usage
HTML	Structured and content layout
CSS	Styling, animations, and responsive design
JavaScript	Form validation, dynamic UI interactions and search functionality
PHP	Backend logic or authentication, database operations and form handling
MYSQL	Database storage for user data, courses, and contact submissions
Boxicons	Icon library for UI elements

2. Folder Structure

Location Path: C:\xampp\htdocs\student portal

Folder name: student_portal

This student_portal folder needs to locate inside htdocs folder.

CSS

- contact.css - styles for contact page
- course.css - styles for course page
- home.css - styles for course page
- login.css - styles for login page
- profile.css - styles for profile page
- nav.css - styles for navigation bar

PHP

- db.php - Database connection setup
- login.php - handling user login
- register.php - Handles user registration
- logout.php - ends user session
- profile.php - manage profile updates
- contact.php - handle contact form submissions

JS

- script.js - handling JavaScript functions like form validation, Dark theme

Images- Contains logo and background images

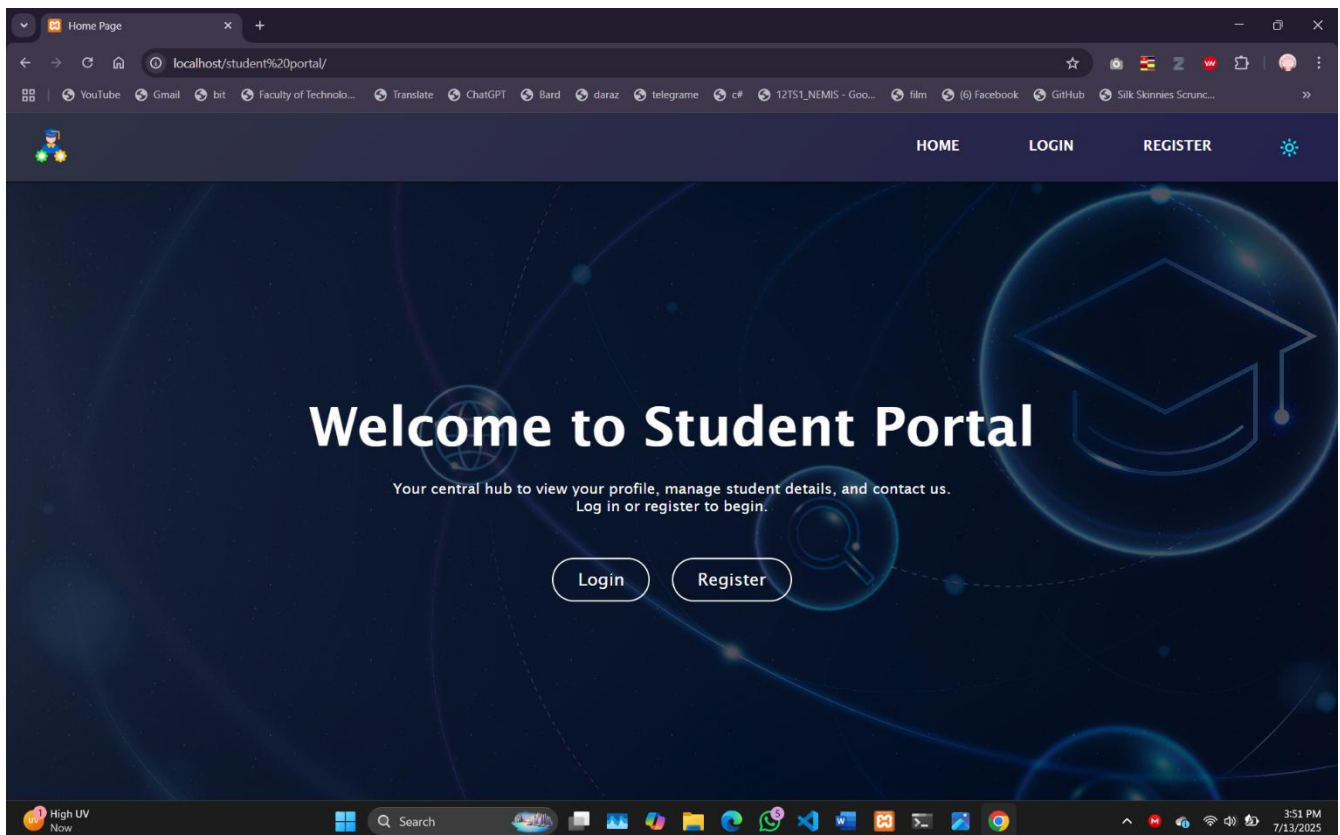
HTML

- home.html - landing page
- login.html - login and registration page
- profile.php - student profile(protected)
- course.php - course listing(protected)
- contact.php - contact form(protected)

3. Page Descriptions

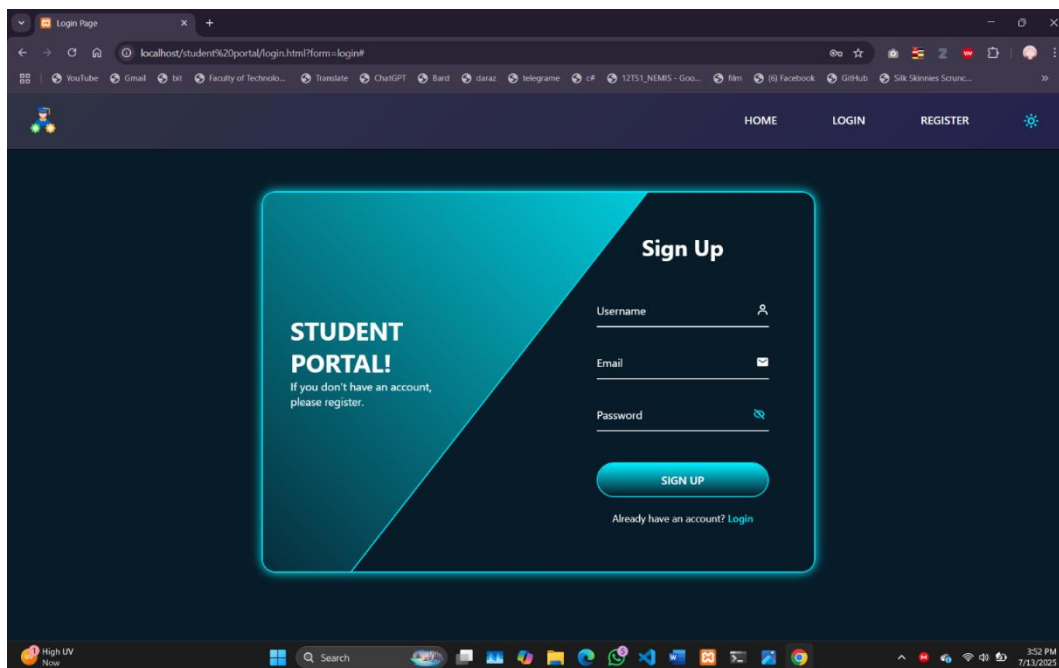
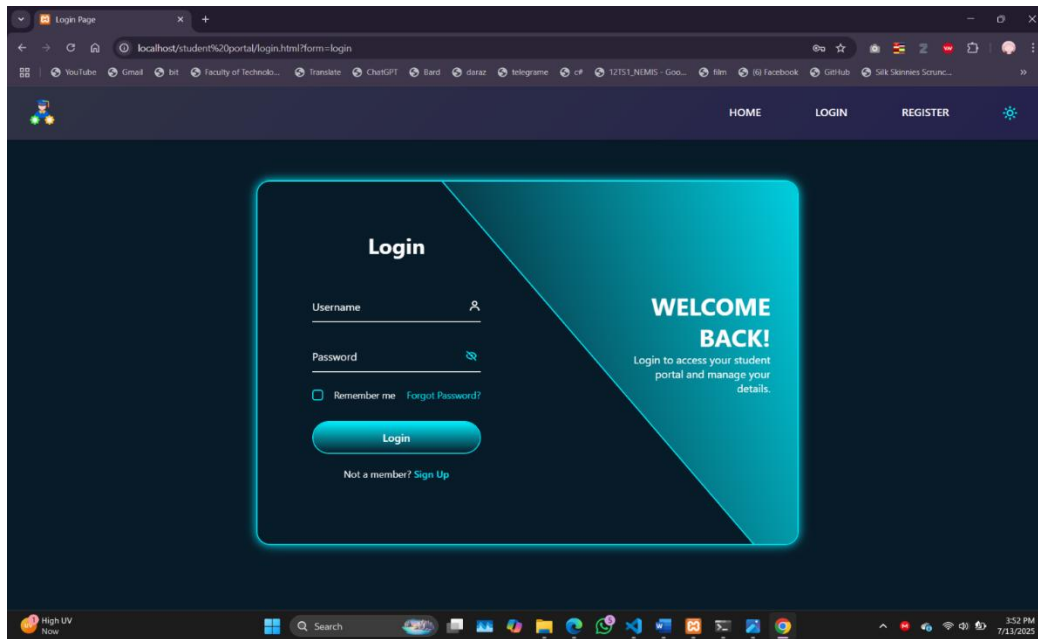
Home page

- Purpose: Landing page with a welcome message and navigation options
- Features:
 - Hero section with call-to-action buttons for LOGIN and REGISTRATER
 - Responsive design using CSS Flexbox/Grid
 - Dark/Light mode toggle



Login & Registration Page(login.html)

- Purpose: Allows students to log or create an account
- Features:
 - Animated toggle between login and registration forms
 - Form validation (required fields, email format, password strength)
 - Password visibility toggle (eye/hide icon)
 - Error handling (Invalid credentials, existing email/username)



Profile Page(profile.php)

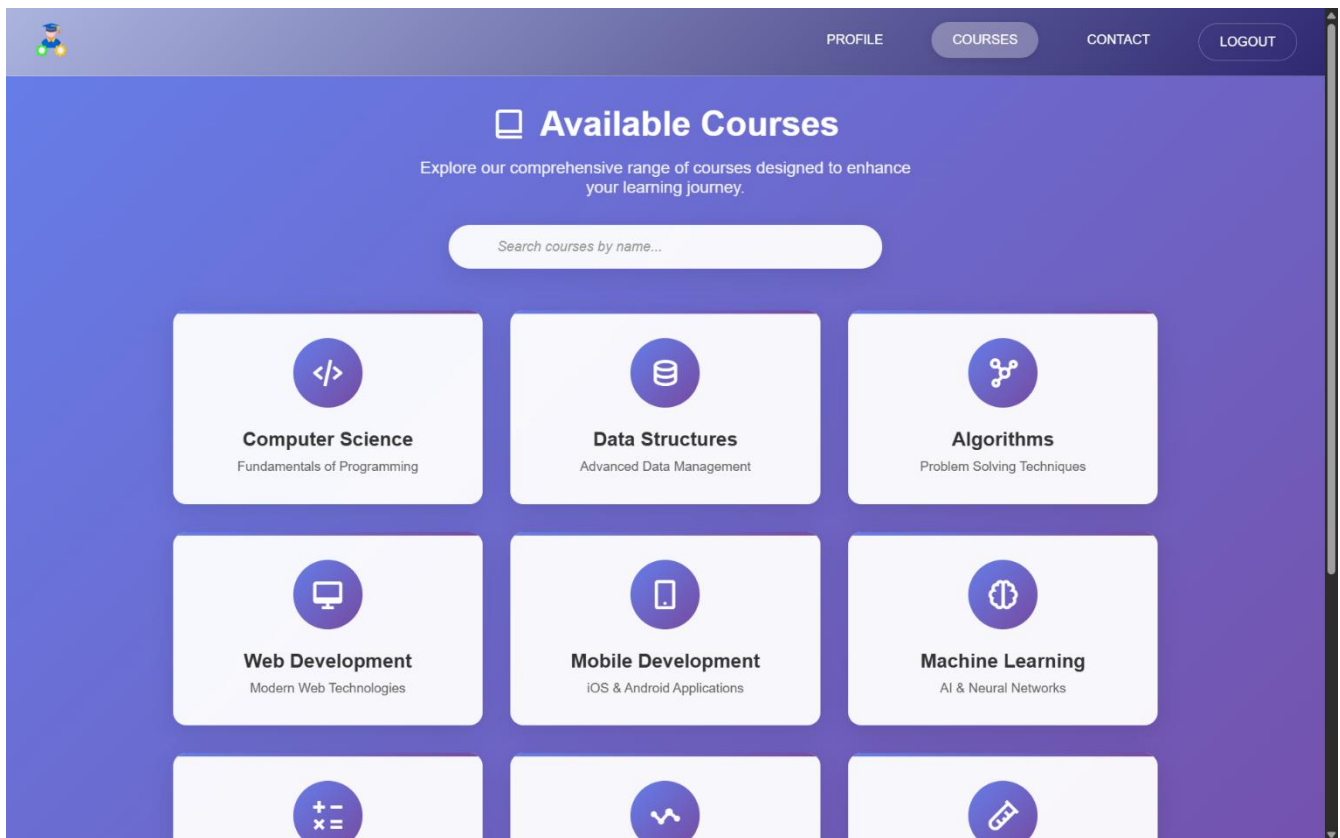
- Purpose: Display and allows editing of student information
- Features:
 - Personal info Section- Name, email, phone, address, DOB
 - Academic info section - Student ID, program, year level, enrollment date
 - Edit Mode - Toggle between view and edit modes with form validation
 - Success/Error Messages - Feedback after updates

The screenshot shows a web application interface for a student profile. At the top, there is a navigation bar with links for PROFILE, COURSES, CONTACT, and a LOGOUT button. The main content area features a profile card for Kusal Akalanka Perera, displaying a circular avatar with the initials 'KA', the student ID 'STU001', the email 'kusalakalankaun@gmail.com', and a green 'Active Student' status badge. Below the profile card are two sections: 'Personal Information' and 'Academic Information', each with an 'Edit' button. The 'Personal Information' section includes fields for Full Name, Email, Phone, Date of Birth, and Address. The 'Academic Information' section includes fields for Student ID, Program, Year Level, Enrollment Date, and Expected Graduation.

Section	Field	Value
Personal Information	Full Name	Kusal Akalanka Perera
	Email	kusalakalankaun@gmail.com
	Phone	01122336655
	Date of Birth	July 11, 2025
	Address	217/6, 'Bhagya', Kuruppumulla, Panadura
Academic Information	Student ID	STU001
	Program	Software Engineering
	Year Level	1st Year
	Enrollment Date	August 2025
	Expected Graduation	October 2025

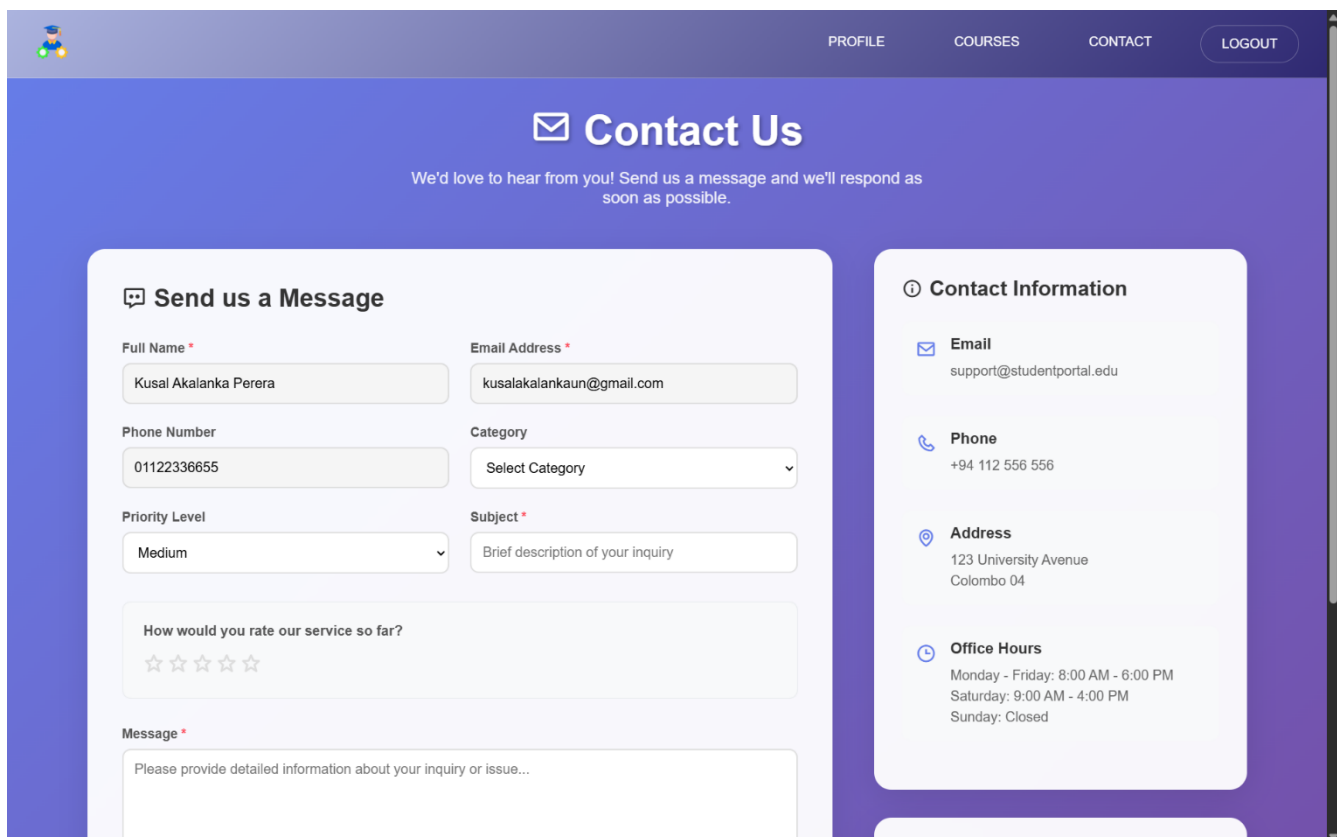
Courses Page(course.php)

- Purpose: Display available courses in a searchable grid
- Features:
 - Search bar - Filter courses in real-time
 - Course cars - each cars shows course name, description, and icon
 - No-Results message - shown if no matches are found



Contact Page(contact.php)

- Purpose - Allows students to submit inquiries or feedback
- Features:
 - Form fields - Name(auto-filled), email, phone, subject, message
 - Star Rating system – Rates service experience (1-5 stars)
 - Character count - limits message length to 1000 characters
 - Success/Error messages - confirms submission or displays errors



Contact Us

We'd love to hear from you! Send us a message and we'll respond as soon as possible.

Send us a Message

Full Name *
Kusal Akalanka Perera

Email Address *
kusalakalankaun@gmail.com

Phone Number
01122336655

Category
Select Category

Priority Level
Medium

Subject *
Brief description of your inquiry

How would you rate our service so far?
☆☆☆☆☆

Message *
Please provide detailed information about your inquiry or issue...

Contact Information

Email
support@studentportal.edu

Phone
+94 112 556 556

Address
123 University Avenue
Colombo 04

Office Hours
Monday - Friday: 8:00 AM - 6:00 PM
Saturday: 9:00 AM - 4:00 PM
Sunday: Closed

4. JavaScript Functions

Function	Description
Form validation	Check requires filed, email format, and password strength
Password Toggle	Switch between visible / hidden password input
Light/ Dark Mode	Toggle theme and save preference
Course Search	Filter course dynamically as the user types
Start Rating	Interactive rating selection with hover effects
Character Counter	Tracks message length in the contact form

5. PHP Backend Logic

Database Schema

Use XAMP SERVER for connecting database

- **Tables**

```
MariaDB [(none)]> use student_db;
Database changed
MariaDB [student_db]> show tables;
+-----+
| Tables_in_student_db |
+-----+
| contact_submissions  |
| user_info            |
| users                |
+-----+
3 rows in set (0.001 sec)
```

- users Table: Stores id, username, email, password(hashed)

```
MariaDB [student_db]> desc users;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
username	varchar(50)	NO		NULL	
email	varchar(100)	NO		NULL	
password	varchar(255)	NO		NULL	

```
4 rows in set (0.031 sec)
```

- User_info table: stores additional student details

```
MariaDB [student_db]> desc user_info;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
user_id	int(11)	NO	MUL	NULL	
full_name	varchar(100)	YES		NULL	
email	varchar(100)	YES		NULL	
phone	varchar(20)	YES		NULL	
dob	date	YES		NULL	
address	text	YES		NULL	
student_id	varchar(20)	YES		NULL	
program	varchar(100)	YES		NULL	
year_level	varchar(20)	YES		NULL	
enrollment_date	date	YES		NULL	
graduation_date	date	YES		NULL	

```
12 rows in set (0.030 sec)
```

- contact_submissions table: stores contact form entries

```
MariaDB [student_db]> desc contact_submissions;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
user_id	int(11)	NO	MUL	NULL	
name	varchar(255)	NO		NULL	
email	varchar(255)	NO		NULL	
phone	varchar(20)	YES		NULL	
subject	varchar(500)	NO		NULL	
message	text	NO		NULL	
rating	int(11)	YES		0	
category	varchar(50)	YES		NULL	
priority	enum('low', 'medium', 'high', 'urgent')	YES	MUL	medium	
status	enum('pending', 'in_progress', 'resolved', 'closed')	YES	MUL	pending	

```
11 rows in set (0.022 sec)
```

Authentication Flow

1. Registration (register.php)
 - Validates username/ email uniqueness.
 - Hashes password using password_hash ()
 - Generates a student ID (ex: STU001)
 - Sends a confirmation email
2. Login (login.php)
 - Verifies credentials against the database
 - Starts a session with \$_SESSION['user_id']
 - Redirects to profile.php on success
3. Logout (logout.php)
 - Destroys the session and redirects to login

Protected Pages

- Profile, courses, contact pages check isset(\$_SESSION['user_id'])
- Unauthorized users are redirected to login.html

6. Challengers and Improvements

Challengers

- Form validation: ensuring both client-side (JS) and server-side (PHP) validation
- Session management: preventing unauthorized access to protected pages
- Responsive design: Adapting UI for mobile devices
- Use some AI tools to fix issues

Improvements

- Added features
 - Email Notifications (registration confirmation)
 - Search functionality (course filtering)
 - Star rating system (feedback collection)
- Future enhancement
 - Password reset via email
 - Course enrollment system
 - Admin dashboard for managing students

7. How to Run the Project

Prerequisites

- XAMPP/ WAMP (for local PHP/ MYSQL server)
- Web Browser (Use Chrome for better performance)

Setup steps

- Start Apache & MYSQL in XAMPP / WAMP
- Create Database:

1. **Open phpMyAdmin**
2. **Import the provided SQL schema**

-- Create the database

```
CREATE DATABASE IF NOT EXISTS student_db;  
USE student_db;
```

-- Create users table

```
CREATE TABLE IF NOT EXISTS users (  
  id INT(11) NOT NULL AUTO_INCREMENT,  
  username VARCHAR(50) NOT NULL,  
  email VARCHAR(100) NOT NULL,  
  password VARCHAR(255) NOT NULL,  
  PRIMARY KEY (id)  
);
```

-- Create user_info table

```
CREATE TABLE IF NOT EXISTS user_info (  
  id INT(11) NOT NULL AUTO_INCREMENT,  
  user_id INT(11) NOT NULL,  
  full_name VARCHAR(100) DEFAULT NULL,  
  email VARCHAR(100) DEFAULT NULL,  
  phone VARCHAR(20) DEFAULT NULL,  
  dob DATE DEFAULT NULL,  
  address TEXT DEFAULT NULL,  
  student_id VARCHAR(20) DEFAULT NULL,  
  program VARCHAR(100) DEFAULT NULL,  
  year_level VARCHAR(20) DEFAULT NULL,  
  enrollment_date DATE DEFAULT NULL,  
  graduation_date DATE DEFAULT NULL,
```

```
PRIMARY KEY (id),
KEY user_id (user_id),
FOREIGN KEY (user_id) REFERENCES users (id)
);
```

-- Create contact_submissions table

```
CREATE TABLE IF NOT EXISTS contact_submissions (
  id INT(11) NOT NULL AUTO_INCREMENT,
  user_id INT(11) NOT NULL,
  name VARCHAR(255) NOT NULL,
  email VARCHAR(255) NOT NULL,
  phone VARCHAR(20) DEFAULT NULL,
  subject VARCHAR(500) NOT NULL,
  message TEXT NOT NULL,
  rating INT(11) DEFAULT 0,
  category VARCHAR(50) DEFAULT NULL,
  priority ENUM('low','medium','high','urgent') DEFAULT 'medium',
  status ENUM('pending','in_progress','resolved','closed') DEFAULT 'pending',
  PRIMARY KEY (id),
  KEY user_id (user_id),
  KEY priority (priority),
  KEY status (status),
  FOREIGN KEY (user_id) REFERENCES users (id)
);
```

3. Place files in htdocs:

- Copy all project files to C:\xampp\htdocs\student_portal

4. check your XAMPP/ WAMP mysql username & password :

username – root

password- ''

These are my credentials, if your mysql credentials are different with this, you need to change in db.php

5. Email changes:

1.First config PHP's mail setting in php.ini and config as shows below

```
[mail function]
; For Win32 only.
; http://php.net/smtp
SMTP=smtp.gmail.com
; http://php.net/smtp-port
smtp_port=587

; For Win32 only.
; http://php.net/sendmail-from
sendmail_from = kusalakalankaun@gmail.com

; For Unix only.  You may supply arguments as well (default: "sendmail -t -i").
; http://php.net/sendmail-path
sendmail_path = "\"C:\xampp\sendmail\sendmail.exe\" -t"
```

2. Next go to sendMail folder sendmail.ini and config as shown below

```
[sendmail]

; you must change mail.mydomain.com to your smtp server,
; or to IIS's "pickup" directory.  (generally C:\Inetpub\mailroot\Pickup)
; emails delivered via IIS's pickup directory cause sendmail to
; run quicker, but you won't get error messages back to the calling
; application.

smtp_server=smtp.gmail.com

; smtp port (normally 25)

smtp_port=587

; SMTPS (SSL) support
; auto = use SSL for port 465, otherwise try to use TLS
; ssl  = always use SSL
; tls  = always use TLS
; none = never try to use SSL

smtp_ssl=auto

; the default domain for this server will be read from the registry
; this will be appended to email addresses when one isn't provided
; if you want to override the value in the registry, uncomment and modify
```

```
;default_domain=mydomain.com

; log smtp errors to error.log (defaults to same directory as sendmail.exe)
; uncomment to enable logging

error_logfile=error.log

; create debug log as debug.log (defaults to same directory as sendmail.exe)
; uncomment to enable debugging

debug_logfile=debug.log

; if your smtp server requires authentication, modify the following two lines

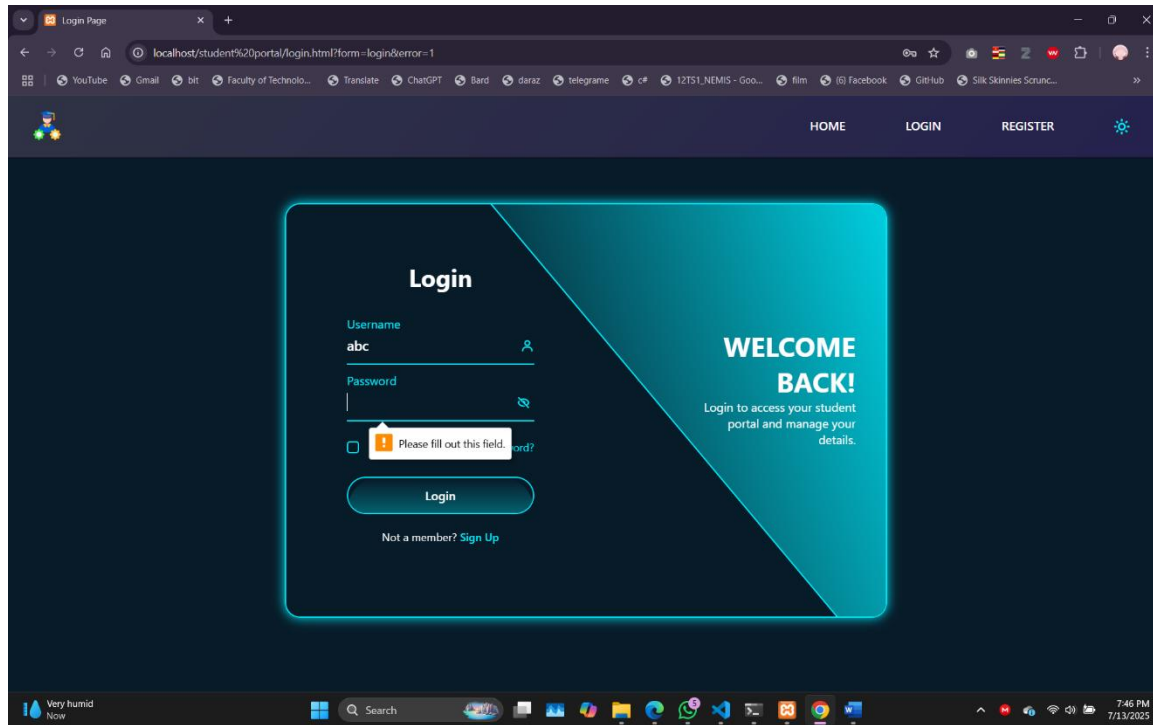
auth_username=kusalakalankaun@gmail.com
auth_password=mmqa gjkr ikli xtps
```

6. Access in Browser :

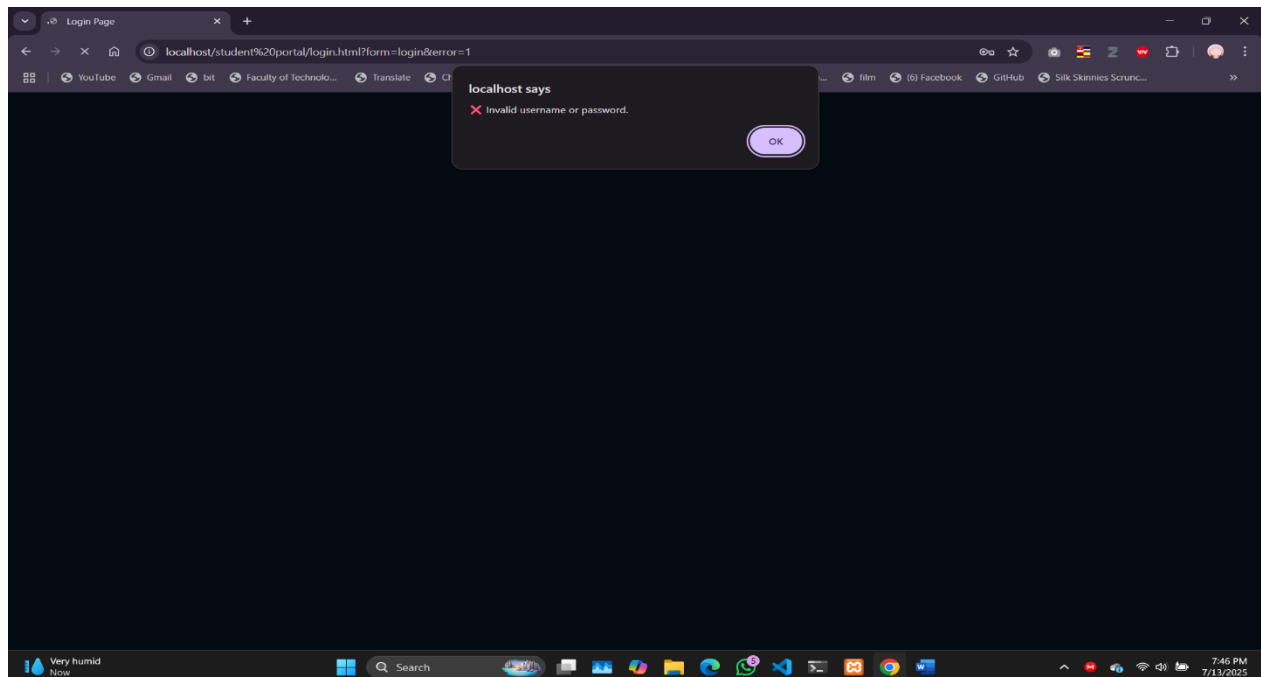
Open : http://localhost/student_portal/home.html

8. Functionalities

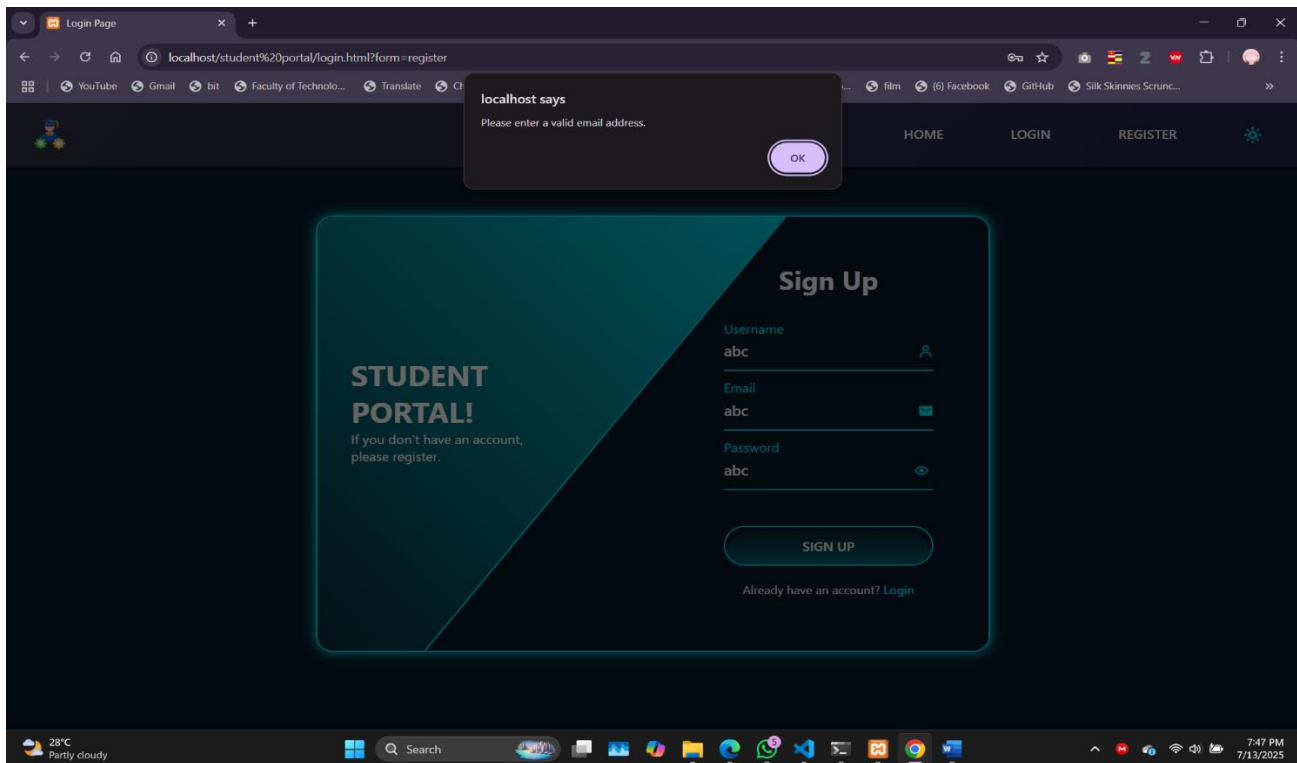
- No fields are empty error message



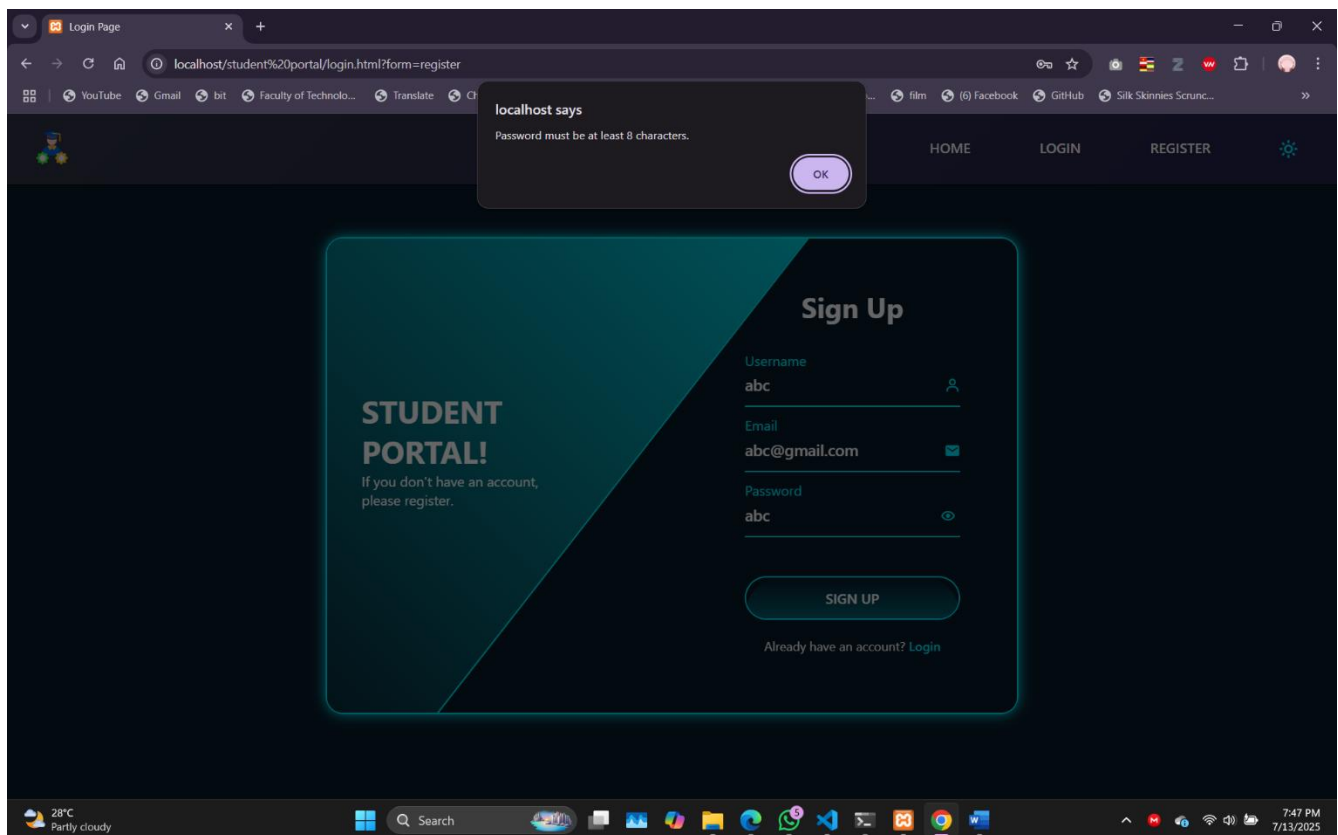
- Invalid Username or password message



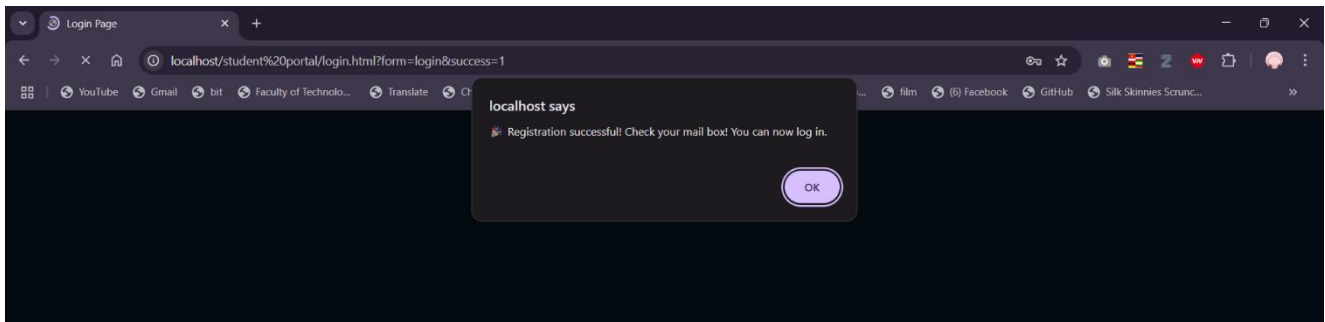
- Email address validation message



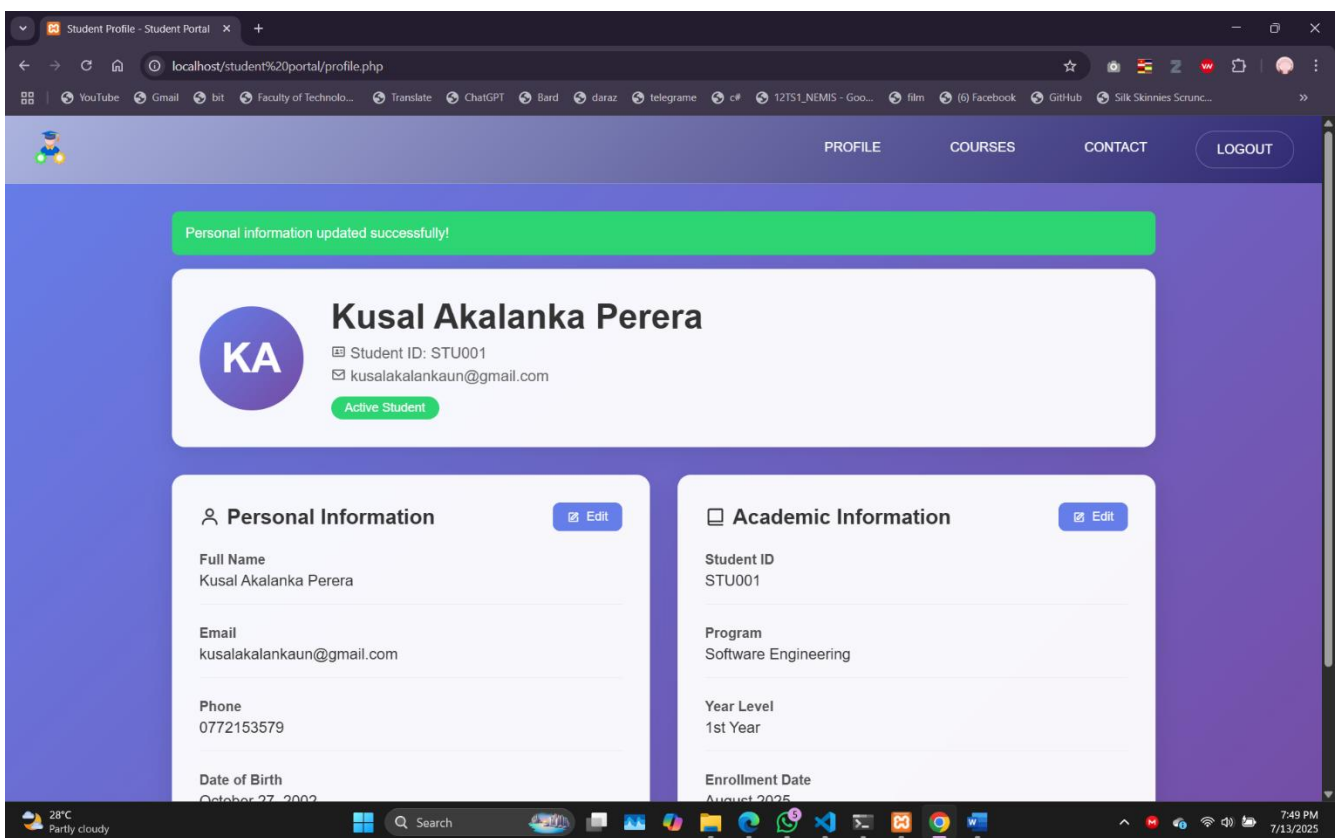
- Password validation message



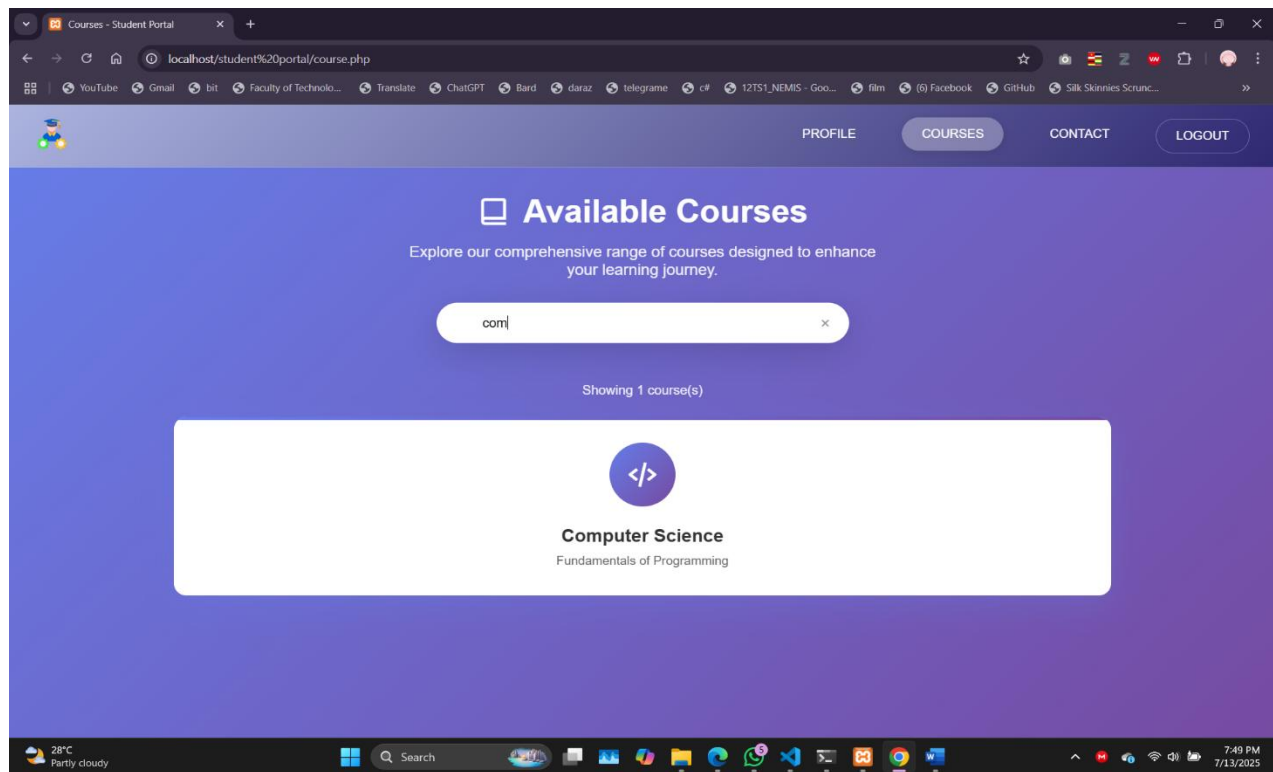
- Registration successful message



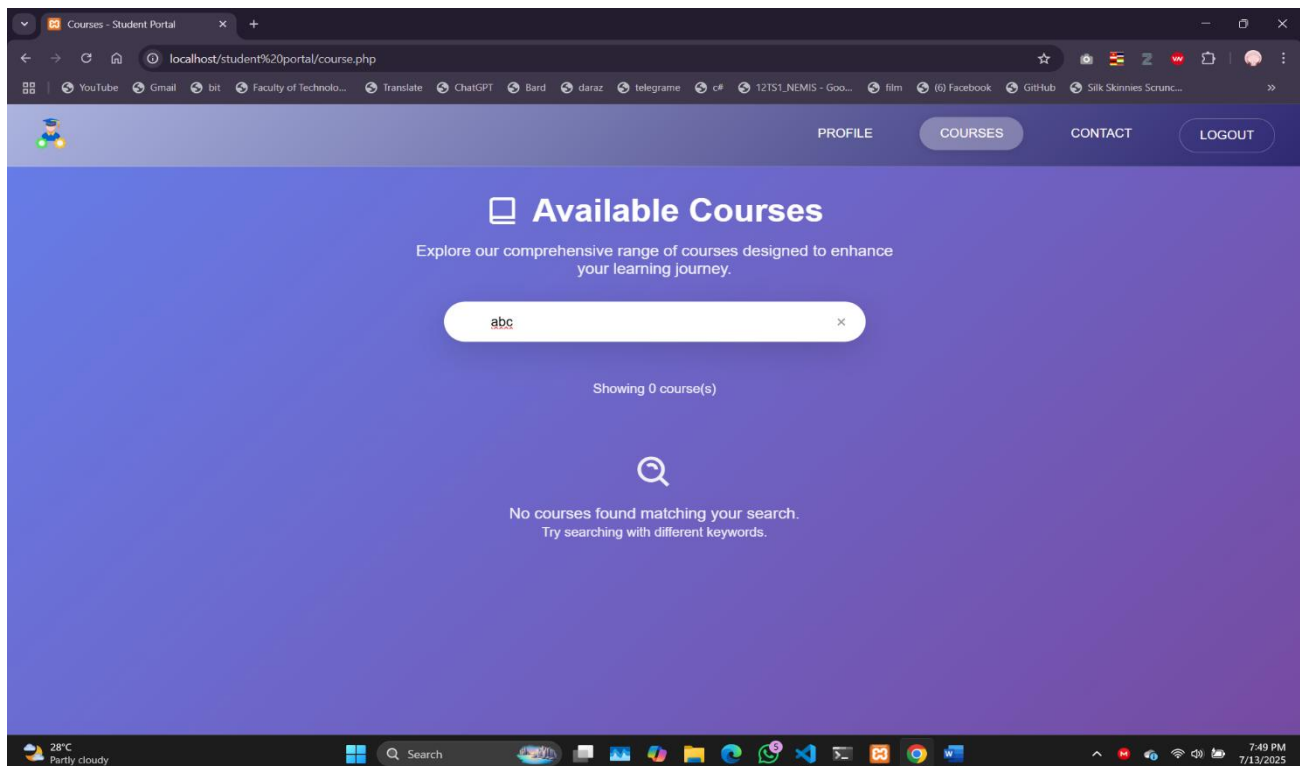
- Information updated successfully message



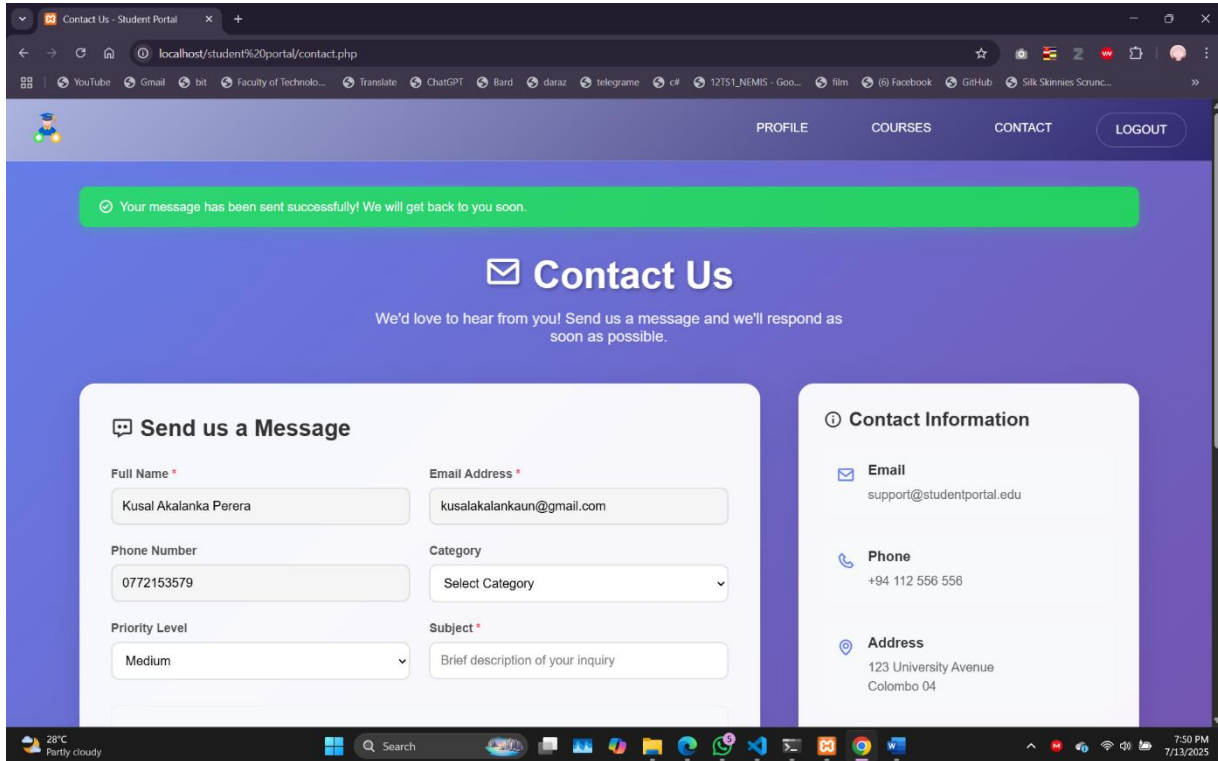
- Filter out matching search course



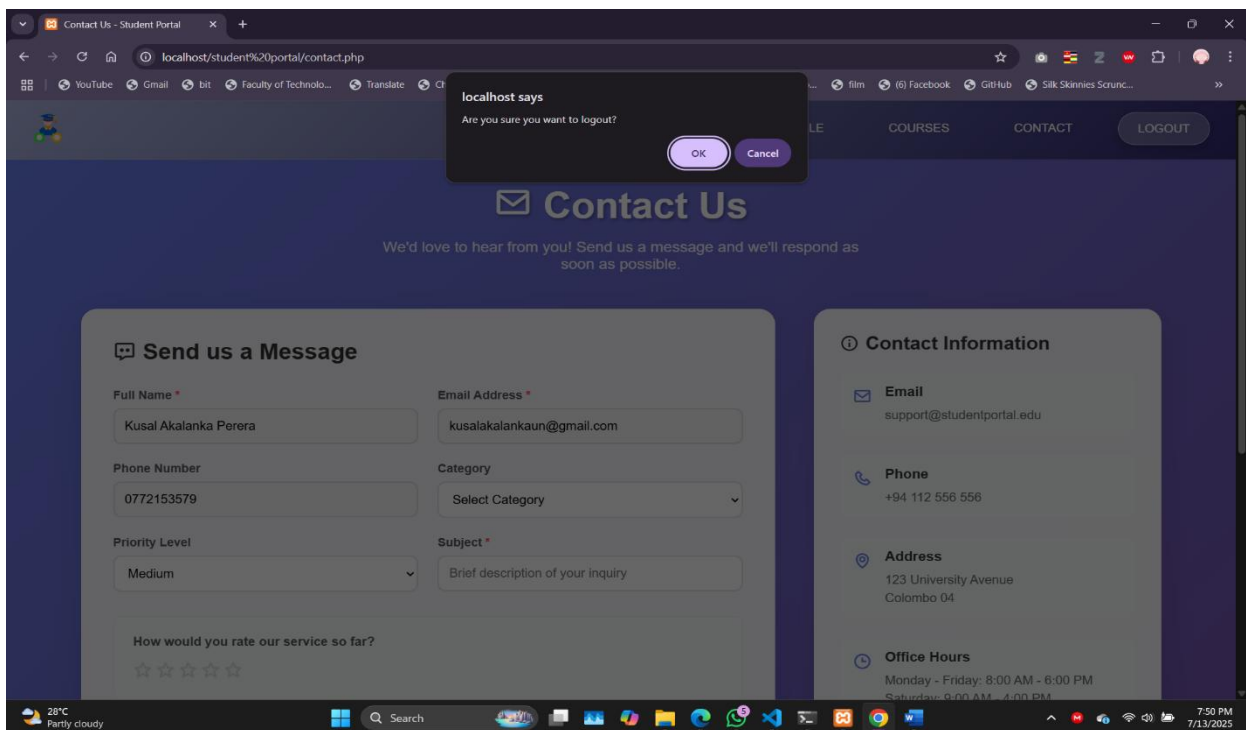
- Error message for mismatch search course



- Contact form submission saucerful message



- Logout confirmation message



9. Conclusion

This student portal application successfully delivers a comprehensive digital platform that empowers students to efficiently manage their academic journey.

Through its intuitive interface and robust backend architecture, the system provides:

- ✓ Secure authentication with password protection and session management
- ✓ Complete profile control for both personal and academic information
- ✓ Interactive course browsing with real-time search capabilities
- ✓ Direct communication channels with administrative supports
- ✓ Responsive design ensuring accessibility across all devices

Future development Roadmap:

- ✓ Password recovery system via email verification
- ✓ Course enrollment functionality with registration management
- ✓ Admin dashboard for comprehensive user management
- ✓ Analytics dashboard to track student engagement
- ✓ Mobile application version for on the go access