

Student Course Registration System

Branch: Data Science

Section: 2

Batch No: 13

Student Roll No's : 24B11DS072, 24B11DS002, 24B11DS008, 24B11DS021

Student Name's: G BHANU TEJA, A MADHAN, A PAVAN TEJA , B SIVA SRINU

Faculty Name: K . Syamala Kalyani

Introduction

The Student Course Registration System is a desktop-based application designed to modernize and simplify the process of academic enrollment and course management. In many educational environments, managing student records and course allocations manually through spreadsheets or paper-based files is a common practice. However, these traditional methods often lead to significant challenges, including data redundancy, errors in enrollment records, difficulty tracking course capacity, and the risk of physical data loss.

To address these inefficiencies, this project introduces an automated solution that provides a centralized platform for administrators and students to interact with academic data securely and accurately. The system is built using Java Swing for the graphical user interface, providing a professional and intuitive desktop experience, while leveraging MySQL for robust, relational data storage

This application is developed using:

- Java (Core Java)
- JavaSwing (Frontend)
- JDBC
- MySQL Database

Problem Statement

In many educational institutions, student enrollment and course management are still handled through manual processes, such as paper-based records or basic spreadsheets. These traditional methods are often inefficient and lead to several critical issues.

- Data Redundancy
- Accuracy and Errors
- Course Capacity Management
- Time-Consuming Updates
- Security and Data Loss

Therefore, there is a clear need for an automated, desktop-based system that ensures data

integrity, improves administrative speed, and provides a secure environment for managing academic records.

Objectives of the Project

The primary goal of this project is to develop a reliable and efficient desktop application using Java Swing and MySQL

- To store and manage student, course and registrations data within a structured MySQL database.
- To allow admin to add and manage Students and courses
- To enable admin to register students to the courses
- To prevent duplicate enrollments
- To track enrollment count per course
- To display data in an easy-to-use interface

Scope of the Project

The system can be used in:

- Universities
- Colleges
- Training institutes
- Academic departments

The system allows authorized users (admin) to manage course and enrollment data through a Desktop application.

Modules Description

The Student Course Management System consists of the following modules:

1. Login & Authentication Module

This module serves as the entry point for the application. It provides a secure interface for administrators and users to log in with their credentials.

- **Admin Login:** Grants access to full system controls.
- **User Registration:** Allows new users to create accounts and store their details in the database.

2. Student Information Module

This module is responsible for capturing and maintaining comprehensive student records.

- **Profile Management:** Captures details such as Name, Date of Birth, Age, and Department through a specialized form.
- **Automatic ID Generation:** Ensures every student is assigned a unique identifier (e.g., STU100) automatically.

3. Course Management Module

This module handles the administrative side of the curriculum.

- **Course Entry:** Facilitates adding new courses by defining the course name, duration, and associated costs.
- **Curriculum Tracking:** Manages course IDs and detailed content descriptions for each subject.

5. Enrollment & Allotment Module

The core of the application, this module maps students to their chosen courses.

- **Registration Processing:** Links unique Student IDs to specific Course IDs.
- **Validation:** Prevents duplicate enrollments for the same student in a single course.
- **Capacity Tracking:** Maintains an enrollment count for each course to monitor availability.

6. Database Connectivity Module (JDBC)

This background module acts as the bridge between the **Java Swing** frontend and the **MySQL** database.

- **Query Execution:** Handles the execution of SQL commands like INSERT, SELECT, and DELETE.
- **Persistence:** Ensures all data entered through the GUI is stored securely and can be

User Interface Design

Login window:

This window allows authorized admin to enter their AdminID and password to access the Student Course Registration System.

The screenshot shows a login interface for the "Student Course Registration System". The title bar reads "Student Course Registration System - Login". Below the title bar, a dark header contains the text "STUDENT COURSE REGISTRATION SYSTEM" in yellow and a login hint: "Admin & User Login - (Admin: admin/admin123 | User: User/1234)". The main form area has a white background. It contains two input fields: one for "Admin ID" and one for "Password". The "Password" field includes a "Show" link to its right. Below the form is a green "Login" button. At the bottom of the window, there are links for "Forgot Password?" and "New User?".

Student Course Registration System - Login

STUDENT COURSE REGISTRATION SYSTEM

Admin & User Login - (Admin: admin/admin123 | User: User/1234)

Admin ID:

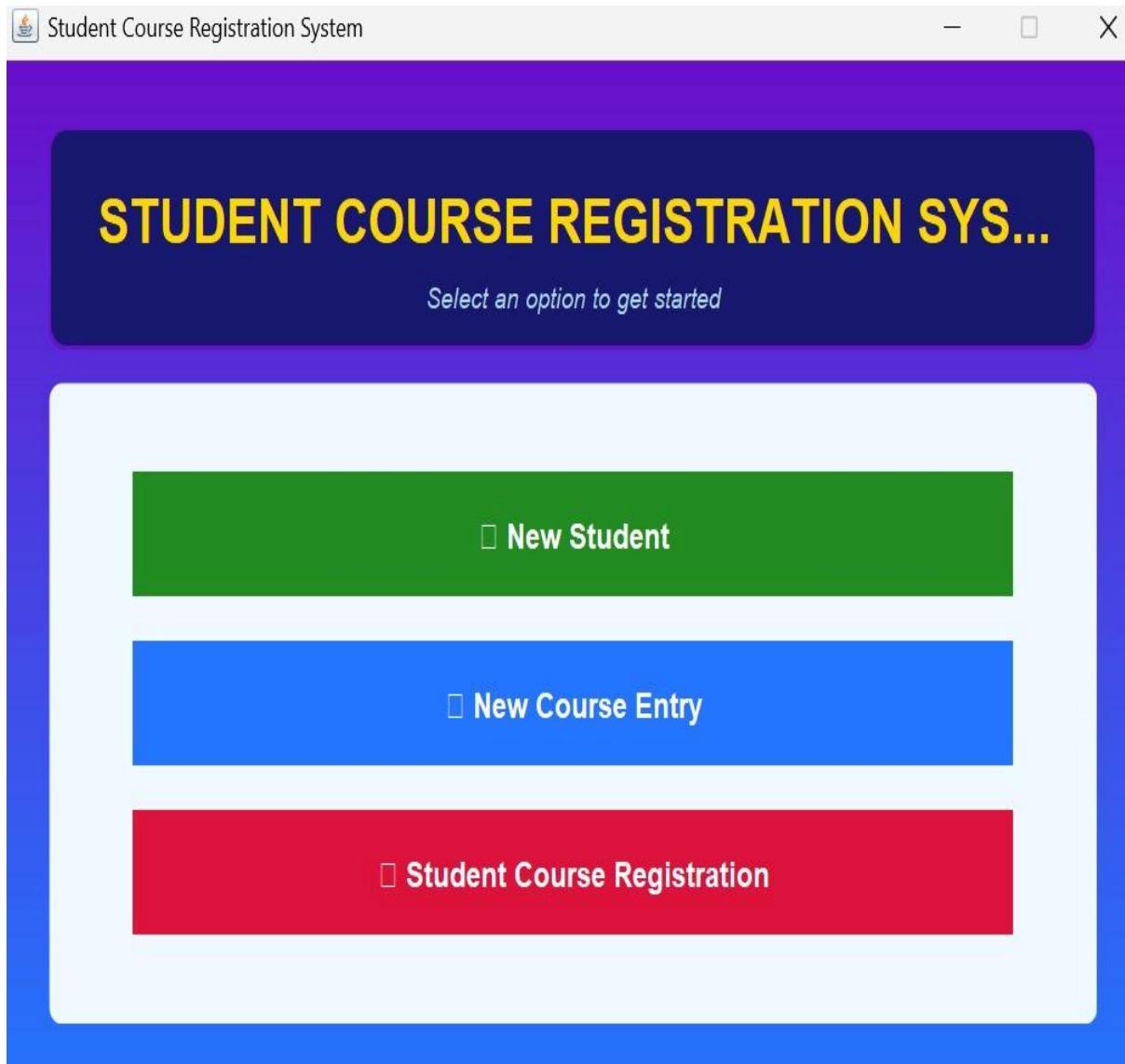
Password: Show

Login

Forgot Password? New User?

Main Dashboard:

This is the central interface of the system that provides navigation to new student entry, new course adding into the database and Course Registration operations. Only the admin can access all the three operations



Student Data Entry Window:

This window allows admin to add new students by entering personal data. The information is stored securely in the database for future Usage like registration process



STUDENT INFORMATION FORM

Manage Student Records

Student Name:

Student ID (Auto):

Date of Birth (DD/MM/YYYY):

Age:

Gmail:

Father Name:

Department:

▼

Gender:

▼

Submit

Clear

Display Info

Display Information

New Courses Entry Window:

This window helps to add new courses, so that students can enroll. In this we add course based on course ID, course name, Course content, course cost.

The screenshot shows a window titled "COURSE MANAGEMENT SYSTEM" with a subtitle "Manage Your Courses". The window has a purple header and a white body. On the left, there are five input fields with labels: "Course Name:", "Course Code (Auto):", "Duration (e.g., 6 months):", "Cost (₹):", and "Content:". To the right of each label is a corresponding input field. At the bottom, there are four buttons: "Add Course" (green), "Clear" (red), "Display All" (dark blue), and "Delete" (dark red). Below the buttons are two panels: "Courses Added" (yellow background) and "Course Information Display" (light blue background).

Student Course Registration System

COURSE MANAGEMENT SYSTEM

Manage Your Courses

Course Name:

Course Code (Auto):

Duration (e.g., 6 months):

Cost (₹):

Content:

Add Course

Clear

Display All

Delete

Courses Added

Course Information Display

Student Course Registration Window:

This window helps the user to allot the courses to the user which are available in the database and store the registration details in the data base

The screenshot shows a Windows application window titled "Student Course Registration System". The main title bar is purple, and the window itself has a white background with a dark blue header bar.

Header Bar:

- Icon: A small icon of a computer monitor with a graduation cap on it.
- Title: "Student Course Registration System"
- Buttons: Minimize, Maximize, Close (X)

Main Content Area:

STUDENT COURSE REGISTRATION

Manage Your Course Enrollments

Input Fields:

- Student Name:** An empty text input field.
- Student ID:** An empty text input field.
- Course ID (e.g., CR100):** An empty text input field.

Action Buttons:

Four buttons in a row:

- Green button: Register
- Red button: Clear
- Dark blue button: View All
- Dark red button: Delete

Information Panels:

- Registered Students:** A large yellow placeholder area.
- Registration Details:** A large yellow placeholder area.

