

UNIVERSITY ERP SYSTEM

Final Project Report

Course: Advanced Programming

Student Name: Mukul & Mayank Rana

Roll Number: 2024359 & 2024341

Project Overview

The University ERP is a desktop application developed to manage the core academic functions of a university. Built using **Java Swing** for the user interface and **JDBC** for database connectivity, the system supports three distinct user roles: Students, Instructors, and Administrators¹.

The primary goal of the system is to facilitate course registration, grade management, and administrative oversight in a secure environment. The application implements a "Shadow Database" architecture to ensure that sensitive authentication data is kept separate from academic records².

Database Design & Schema

To meet the security requirements, I implemented two separate MySQL databases. This ensures that a breach in the academic database does not compromise user credentials.

A. Authentication Database (auth_db)

This database is responsible solely for security. It stores usernames, roles, and password hashes.

- **Table:** users_auth
 - id: Primary Key (Auto Increment)
 - username: Unique identifier for login
 - password: BCrypt hash (Real passwords are never stored)
 - role: Defines access level (student, instructor, admin) ³

B. ERP Database (erp_db)

This database stores the university's operational data.

- **Table:** courses

- course_id: Primary Key
- course_name: Title of the subject
- instructor_name: The faculty member assigned
- credits: Credit value of the course ⁴
- **Table:** enrollments
 - student_name: The student enrolled
 - course_id: Foreign Key linking to courses
 - *Constraint:* Unique pair (student + course) to prevent duplicate registration⁵.
- **Table:** grades_details
 - student_name, course_id: Identifies the record
 - quiz, midterm, endsem: Raw component scores
 - final_grade: The computed letter grade⁶.

Grading Logic (Weighting Rule)

We implemented an automated grading system in the Instructor module. Instead of manually calculating totals, the instructor enters raw scores for three components. The system computes the final grade using the following weighted rule⁷:

Formula:

Total Score = Quiz (20) + Midterm (30) + EndSem (50)

Letter Grade Scale:

Based on the computed Total Score, the system assigns a letter grade:

- **A:** 90 – 100
- **B:** 70 – 89
- **C:** 50 – 69
- **D:** 40 – 49
- **F:** Below 40

This logic is handled in the InstructorPanel class. When the instructor clicks "Compute Final Grades," the system fetches the components, applies the formula, and updates the final_grade column in the database.

Enforcement of Roles & Maintenance Mode

Role Enforcement

Role-based access is strictly enforced at the Login stage.

1. When a user logs in, the AuthService validates the password hash against the users_auth table.
2. If valid, it retrieves the user's role.
3. A Factory pattern in App.java then initializes the specific Dashboard for that role.
 - **Student:** Can only see StudentPanel. They have no access to Admin or Instructor code paths.
 - **Instructor:** Can only see InstructorPanel.
 - **Admin:** Can only see AdminPanel⁸.

Maintenance Mode Enforcement

The Admin has the ability to toggle "Maintenance Mode" to prevent data changes during updates.

1. **Storage:** A boolean flag is stored in the system settings (or managed via a static service MaintenanceService).
2. **Enforcement:** Before any "write" operation is performed (e.g., a student clicking "Register" or an instructor clicking "Save Marks"), the code calls MaintenanceService.isMaintenanceOn().
3. **Result:** If the method returns true, the operation is immediately blocked, and a user-friendly warning message is displayed. This ensures data integrity is maintained⁹⁹⁹⁹.

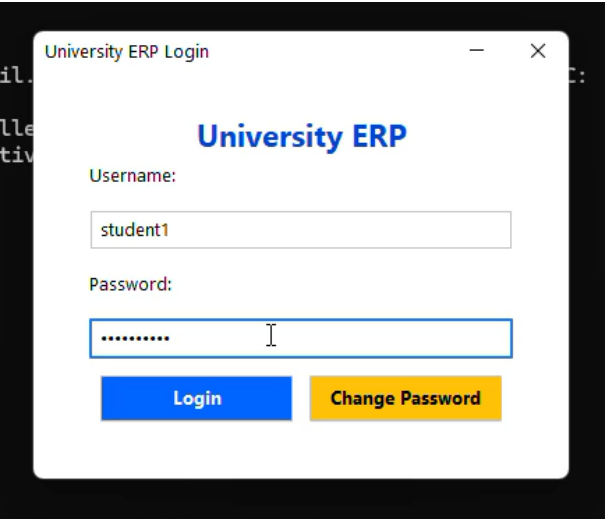
Bonus Features

To enhance the application beyond the core requirements, I added the following features:

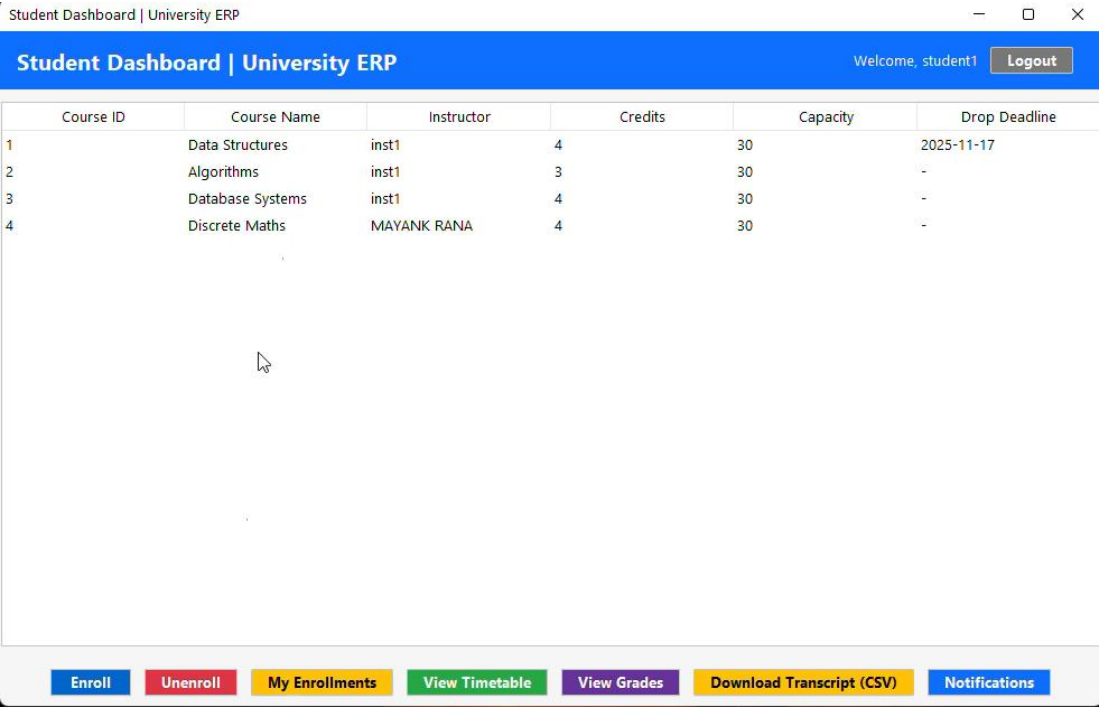
- **Data Visualization:** The Admin Dashboard features dynamic charts (Bar Chart for Enrollments, Pie Chart for User Roles) to provide quick insights¹⁰.
- **Notifications Panel:** A dedicated UI panel allowing users to view system alerts and updates.
- **Reports & Backup Window:** A summary view showing total counts of users, students, and courses¹¹.
- **CSV Export:** Instructors can export their gradebooks to CSV format¹².

Screenshots

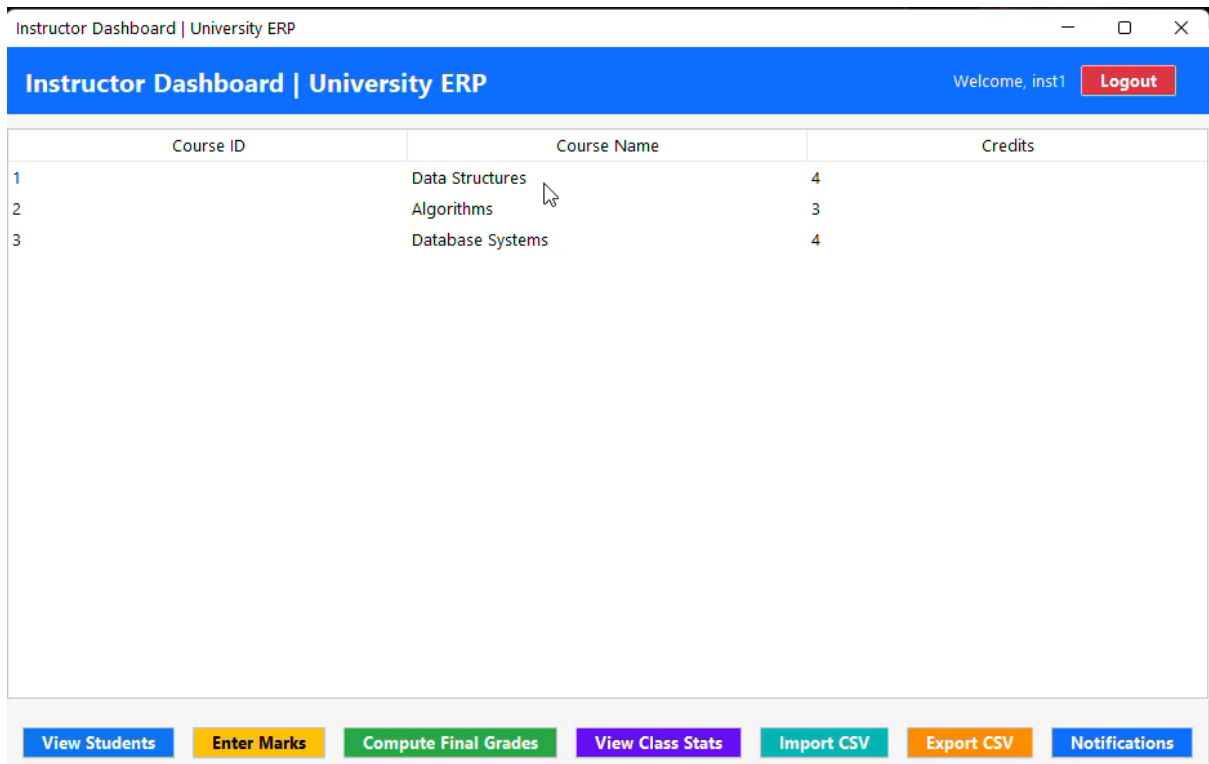
A.Login Screen



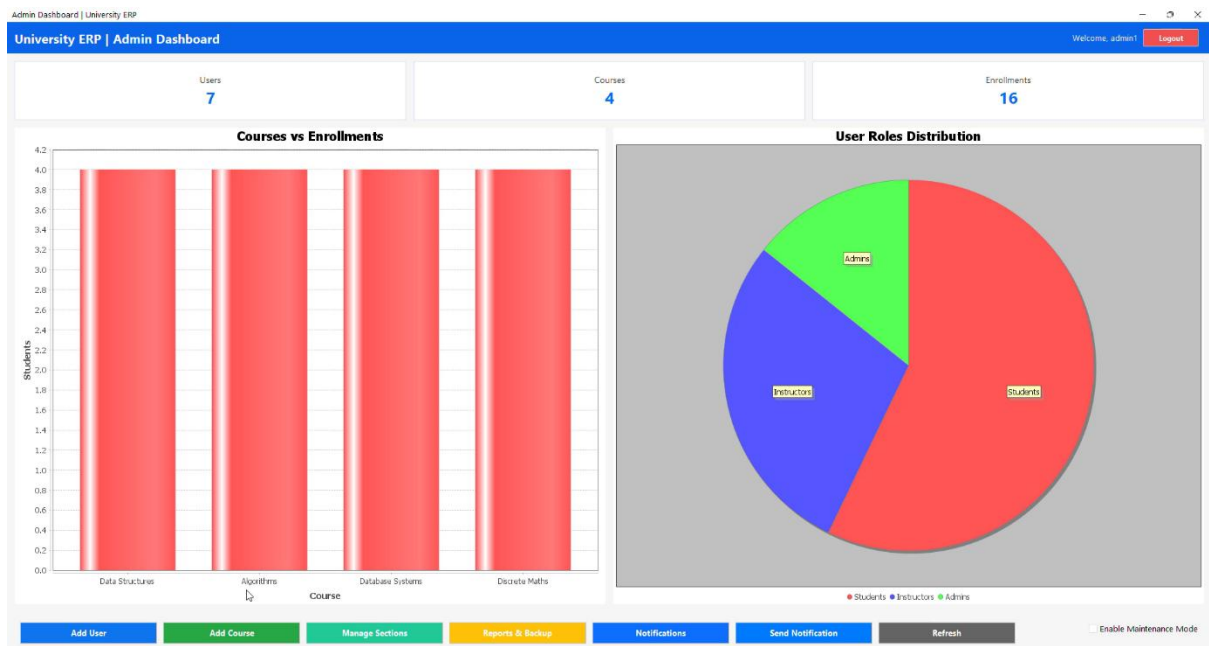
B. Student Dashboard



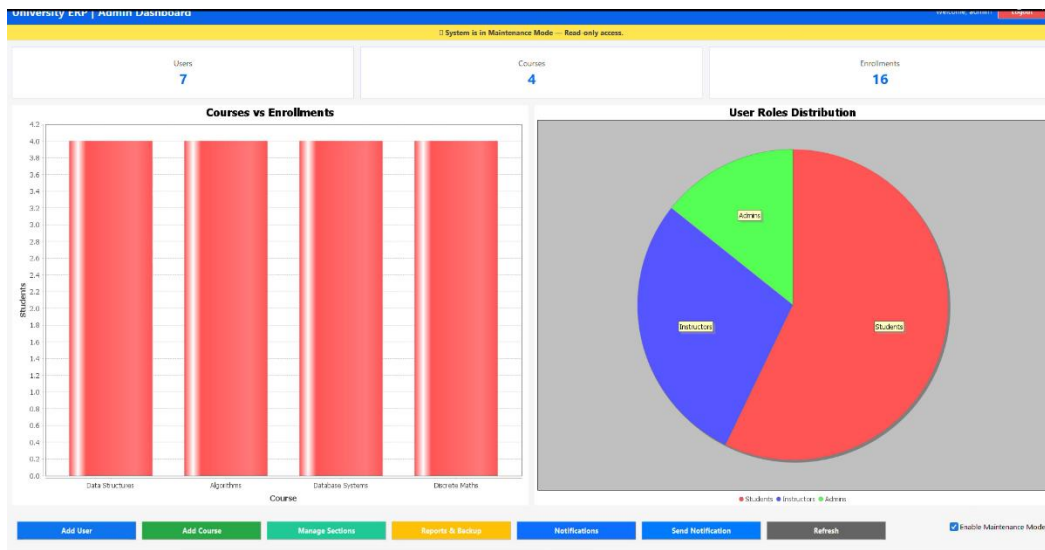
C. Instructor Dashboard



D. Admin Dashboard (with Visualization)



After Maintenance Mode On



university ERP | Instructor Dashboard

Course ID	Course Name	Credits
1	Data Structures	4
2	Algorithms	3
3	Database Systems	4



university ERP | Student Dashboard

Course ID	Course Name	Instructor	Credits	Capacity	Drop Deadline
1	Data Structures	inst1	4	50	2025-11-13
2	Algorithms	inst1	3	30	-
3	Database Systems	inst1	4	30	-
4	Discrete Maths	MOHAN RANA	4	30	-



Note: I have included screenshots of the main features below. All other screens and detailed workflows are explained properly in the submitted Demo Video.