PROJECT REPORT

PROJECT MEMBERS:
1.DIVYARAJ CHAUHAN (2024204)
2.DAKSH KUMAR (2024169)

1. Introduction

1.1 Project Overview

The University ERP System is a comprehensive desktop application designed to manage the core academic and administrative operations of a university. The system streamlines course management, student registration, grading, and transcript generation while ensuring data integrity through a dual-database architecture. It facilitates interaction between three key stakeholders: **Administrators**, **Instructors**, and **Students**.

1.2 Key Features

- **Role-Based Access Control (RBAC):** Provides distinct dashboards and permissions for Admin, Instructor, and Student roles.
- **Dual Database Architecture:** Ensures separation of concerns using university_auth_db for credentials and university_erp_db for academic data.
- **Secure Authentication:** Passwords are hashed using the **BCrypt** algorithm via the jbcrypt library to ensure no plaintext passwords are ever stored.
- **Maintenance Mode:** A global system setting that restricts critical write operations (e.g., registration, dropping, grading) for non-admin users.
- **Academic Deadlines:** Enforces configured **Registration** and **Drop** deadlines, blocking student actions outside the approved window.

1.3 Technology Stack

Component	**Technology / Library**	
Language	Java (JDK 17+)	
UI Framework	Java Swing with **FlatDarkLaf** Theme	
Database	MySQL (via JDBC)	

Security	**jBCrypt** for password hashing	
Utilities	PdfExporter (mock TXT/PDF), CsvExporter (for roster/transcript)	

2. System Architecture

2.1 Database Schema

The system employs a dual-database architecture to strictly separate authentication credentials from operational data.

- **Auth Database (university_auth_db):** Stores user login credentials (username, role) and the password_hash. This ensures no academic record (ERP DB) is exposed even if the authentication database is compromised.
- **ERP Database (university_erp_db):** Stores all academic profiles and transactional data, linked only by user_id. Tables include: students, instructors, courses, sections, enrollments, grades, and settings.

2.2 Weighted Score Computation & Grading Logic

The grading system calculates a weighted average, with instructors setting the assessment components, max marks, and weight percentages for their section.

- Weighted Grade Formula: The core calculation uses the individual component score's percentage multiplied by its weightage:
  $$\text{Total Weighted Score} = \sum \left(\frac{\text{Component Score}}{\text{Max Marks}} \times \text{Weightage \%} \right)$$
- **Letter Grade Assignment:** When the instructor **Publishes Grades**, the system automatically calculates the final score and assigns a letter grade based on configurable numeric cutoffs (e.g., A, B, C, D, F).

3. Module Details

3.1 Admin Module

The Admin dashboard focuses on system-wide control and data setup.

- **User Management:** Create new Student (requires Roll No, Program) or Instructor (requires Department) accounts. Includes a feature to **Reset Password** for any existing user.
- **Course & Section Management:** Full CRUD operations for **Courses** and **Sections**. **Deletion is restricted** for courses or sections if dependent enrollments exist.
- **System Controls:** Toggle **Maintenance Mode** (Read-Only Status) and configure **Registration/Drop Deadlines**.

- **Database Management:** Dedicated buttons for **Backup** and **Restore** of the ERP database.

3.2 Instructor Module

Instructors manage students in their assigned sections.

- **Gradebook:** Allows entry, updating, and deletion of multiple grade components per student for their sections.
- **Grade Security:** Grade submission is blocked if the system is in Maintenance Mode.
- **Class Stats:** Provides summary statistics including **Average** and **Median** scores per grade component, displayed alongside a dynamically generated bar chart for visual analysis.

3.3 Student Module

Students manage their schedule and academic records.

- **Course Catalog:** View all available sections. **Registration** is restricted by capacity, course conflicts, and the Admin-set registration deadline.
- **My Registrations:** View currently enrolled sections. **Drop Section** is restricted by the Admin-set drop deadline.
- **My Timetable:** Displays a weekly schedule of all enrolled classes.
- **Transcript:** Students can view their academic records (including final letter grades) and **Download** the official transcript as a **CSV** file or a mock **PDF** (exported as a formatted .txt file).

4. Compilation and Execution

4.1 Prerequisites

- Java JDK 17 or higher.
- MySQL Server running (default port 3306).
- Apache Maven for dependency management.

4.2 Database Configuration

The database credentials must be configured in ERP/src/edu/univ/erp/data/DatabaseConnector.java.

- **Configured Database Password:** "Admin@12"
- **Database Setup:** The two seed scripts (auth_db_seed.sql, erp_db_seed.sql) must be run prior to execution.

4.3 Build and Run

1. **Build:** Run mvn clean install from the project root.
2. **Run Application:** Execute the main class: edu.univ.erp.Main.

4.4 Default Credentials

Role	Username	Password
Admin	admin1	adminpass
Instructor	inst1	instpass
Student	stu1	stupass1
Student	stu2	stupass2

5. Authentication and Security

5.1 Password Security

The system ensures passwords are never stored in plaintext by utilizing the **BCrypt hashing algorithm**.

- **Hashing Implementation:** Uses BCrypt.hashpw(password, BCrypt.gensalt()) to generate a unique, randomly salted hash for every new password.
- **Verification:** Login is validated using the secure comparison function BCrypt.checkpw(password, storedHash).
- **Secure Password Change:** All user dashboards integrate a **Change Password Dialog**, which enforces re-hashing of the new password before updating the users_auth table.

5.2 Database Separation

The architecture strictly separates user credentials from academic data via **two databases**:

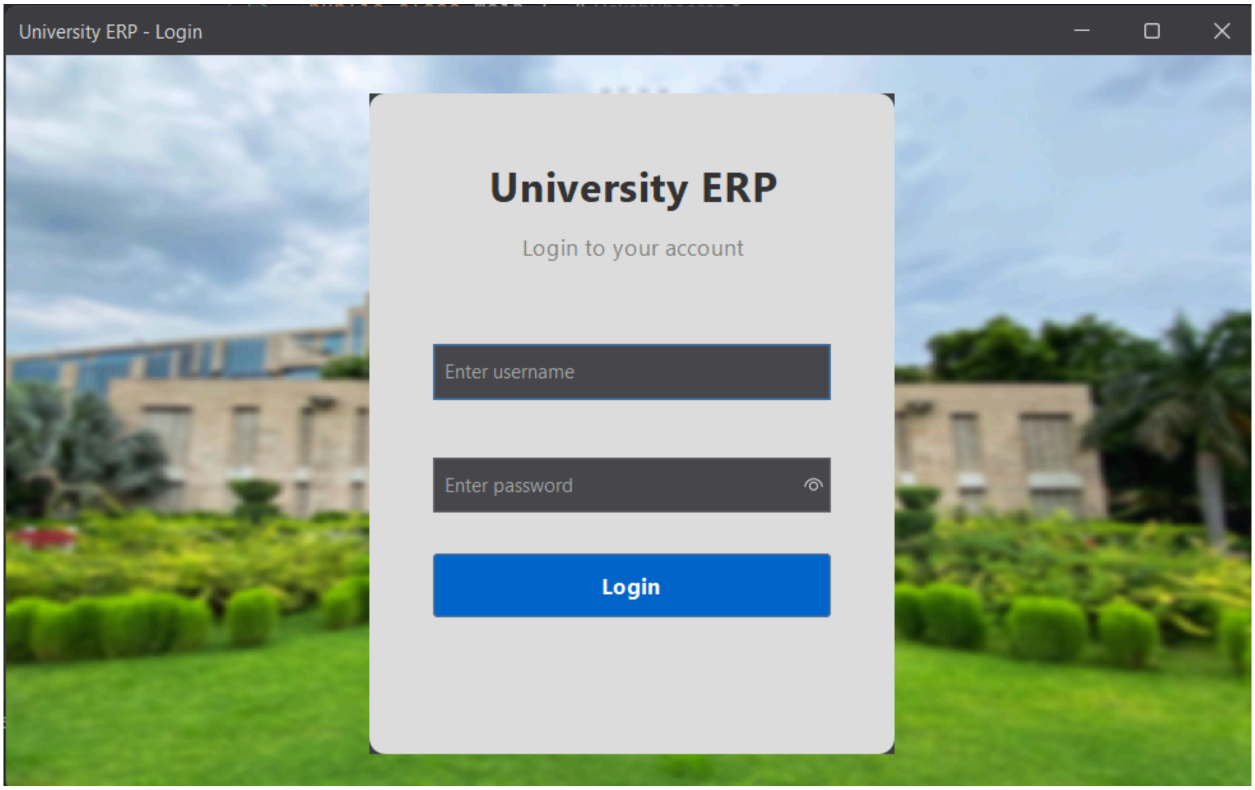
1. **university_auth_db**: Stores sensitive login data.
2. **university_erp_db**: Stores transactional data (e.g., grades, enrollment, sections).

Records are linked solely via the user_id foreign key, ensuring no credential leakage into the operational database.

6. Required Project Screenshots

1. Authentication

- **Login Window:**

- 

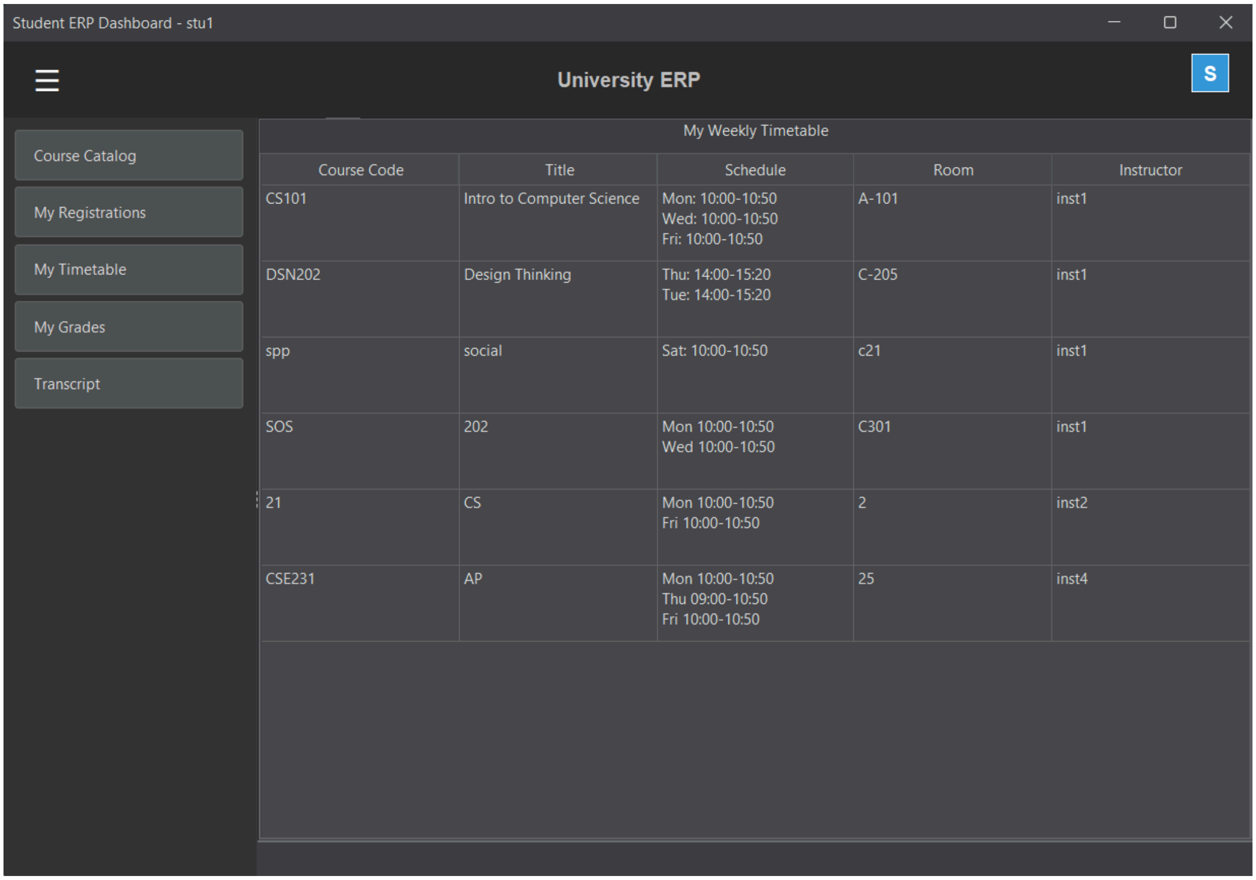
University ERP - Login

University ERP

Login to your account

2. Student Interface

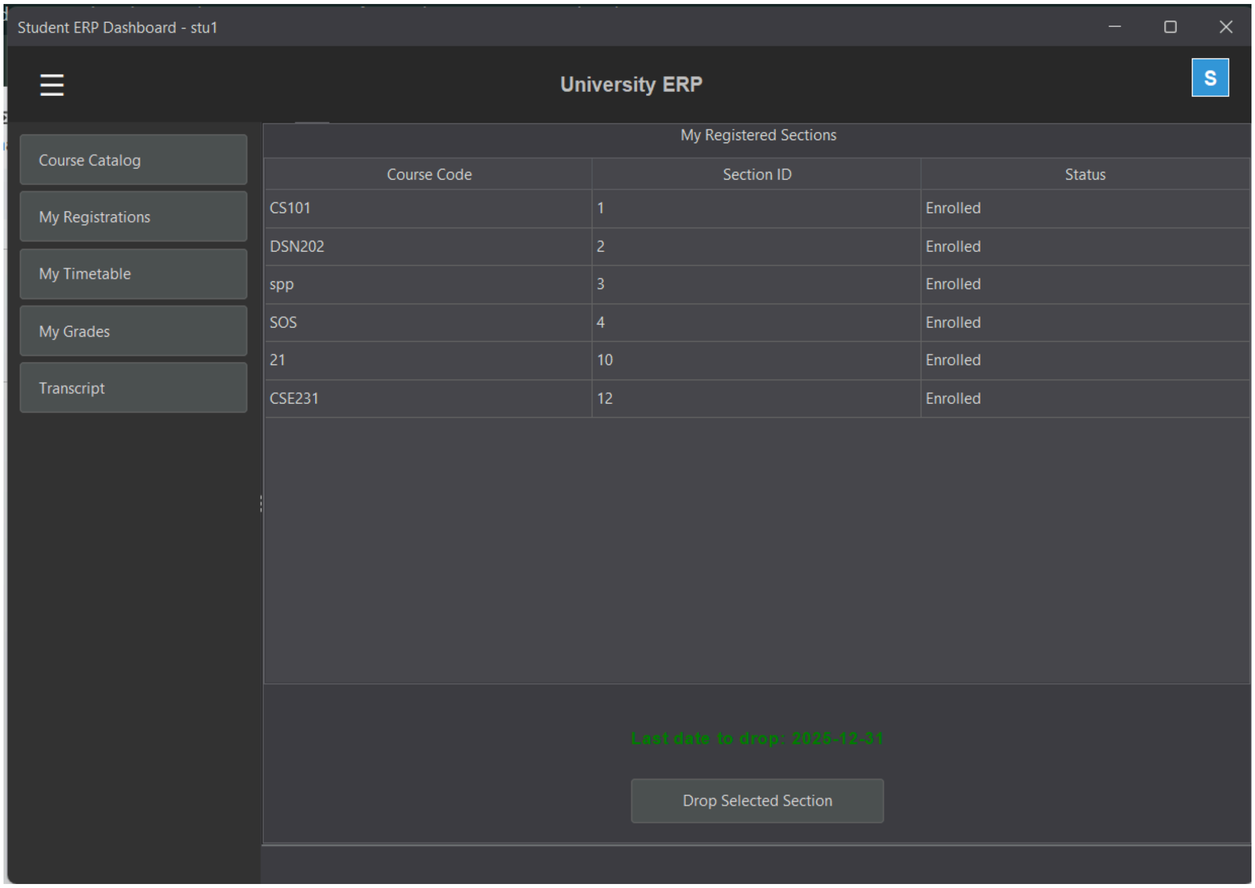
- **Student Dashboard - Timetable:**

- 

My Weekly Timetable

Course Code	Title	Schedule	Room	Instructor
CS101	Intro to Computer Science	Mon: 10:00-10:50 Wed: 10:00-10:50 Fri: 10:00-10:50	A-101	inst1
DSN202	Design Thinking	Thu: 14:00-15:20 Tue: 14:00-15:20	C-205	inst1
spp	social	Sat: 10:00-10:50	c21	inst1
SOS	202	Mon 10:00-10:50 Wed 10:00-10:50	C301	inst1
21	CS	Mon 10:00-10:50 Fri 10:00-10:50	2	inst2
CSE231	AP	Mon 10:00-10:50 Thu 09:00-10:50 Fri 10:00-10:50	25	inst4

- **Student Dashboard - Registered Sections:**

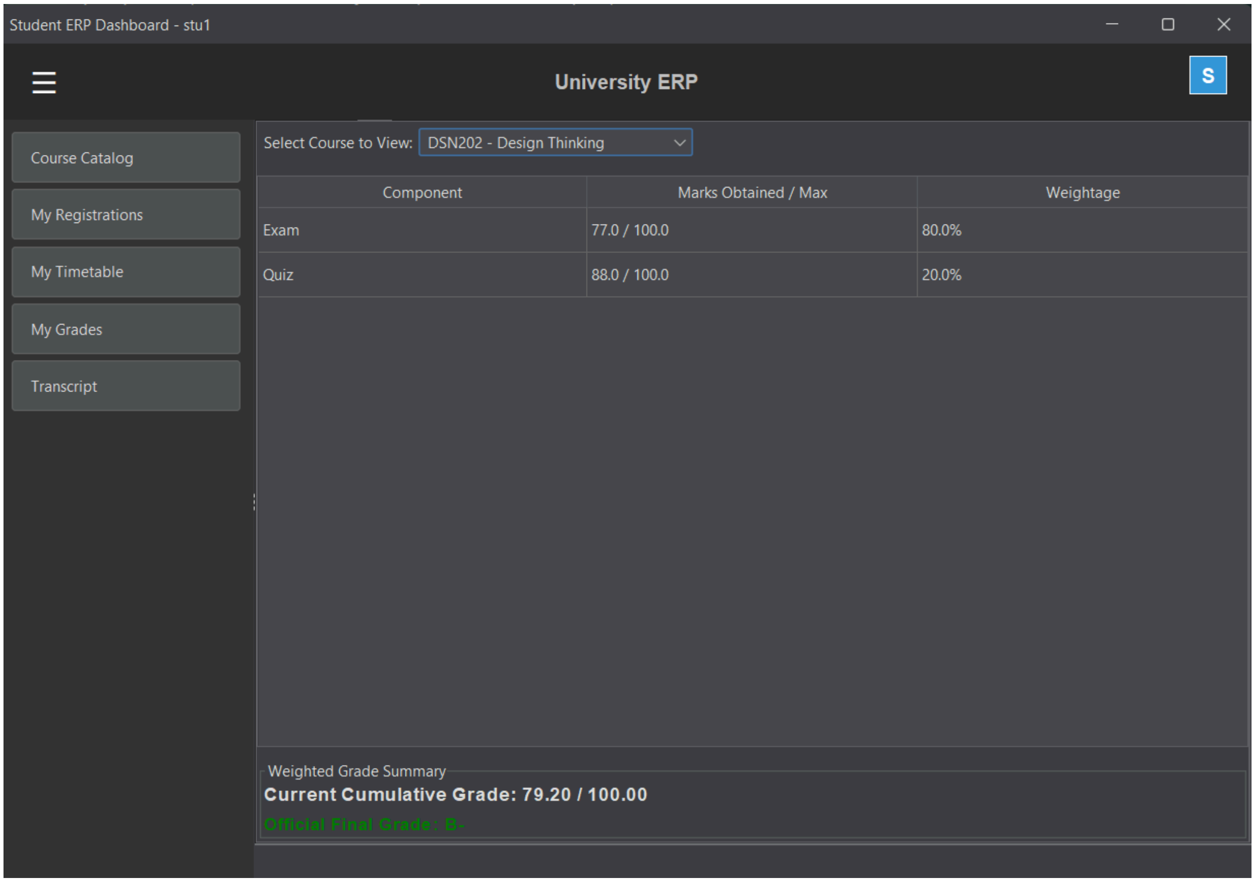
- 

My Registered Sections

Course Code	Section ID	Status
CS101	1	Enrolled
DSN202	2	Enrolled
spp	3	Enrolled
SOS	4	Enrolled
21	10	Enrolled
CSE231	12	Enrolled

Last date to drop: 2025-12-31

- **Student Dashboard - Academic Records:**

- 

Select Course to View: DSN202 - Design Thinking

Component	Marks Obtained / Max	Weightage
Exam	77.0 / 100.0	80.0%
Quiz	88.0 / 100.0	20.0%

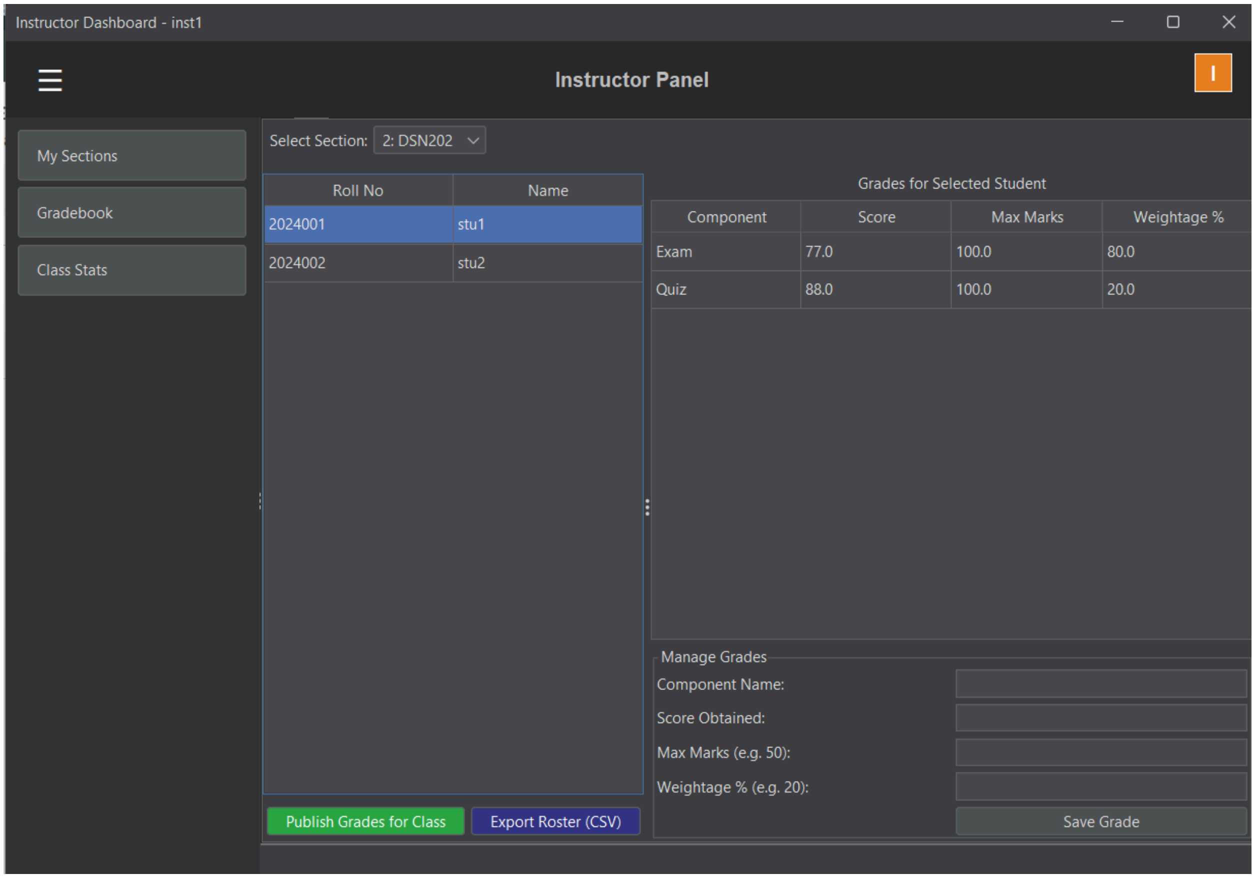
Weighted Grade Summary

Current Cumulative Grade: 79.20 / 100.00

Official Final Grade: B-

3. Instructor Interface

- **Instructor Dashboard - Gradebook:**
-
-



Select Section: 2: DSN202

Roll No	Name
2024001	stu1
2024002	stu2

Grades for Selected Student

Component	Score	Max Marks	Weightage %
Exam	77.0	100.0	80.0
Quiz	88.0	100.0	20.0

Manage Grades

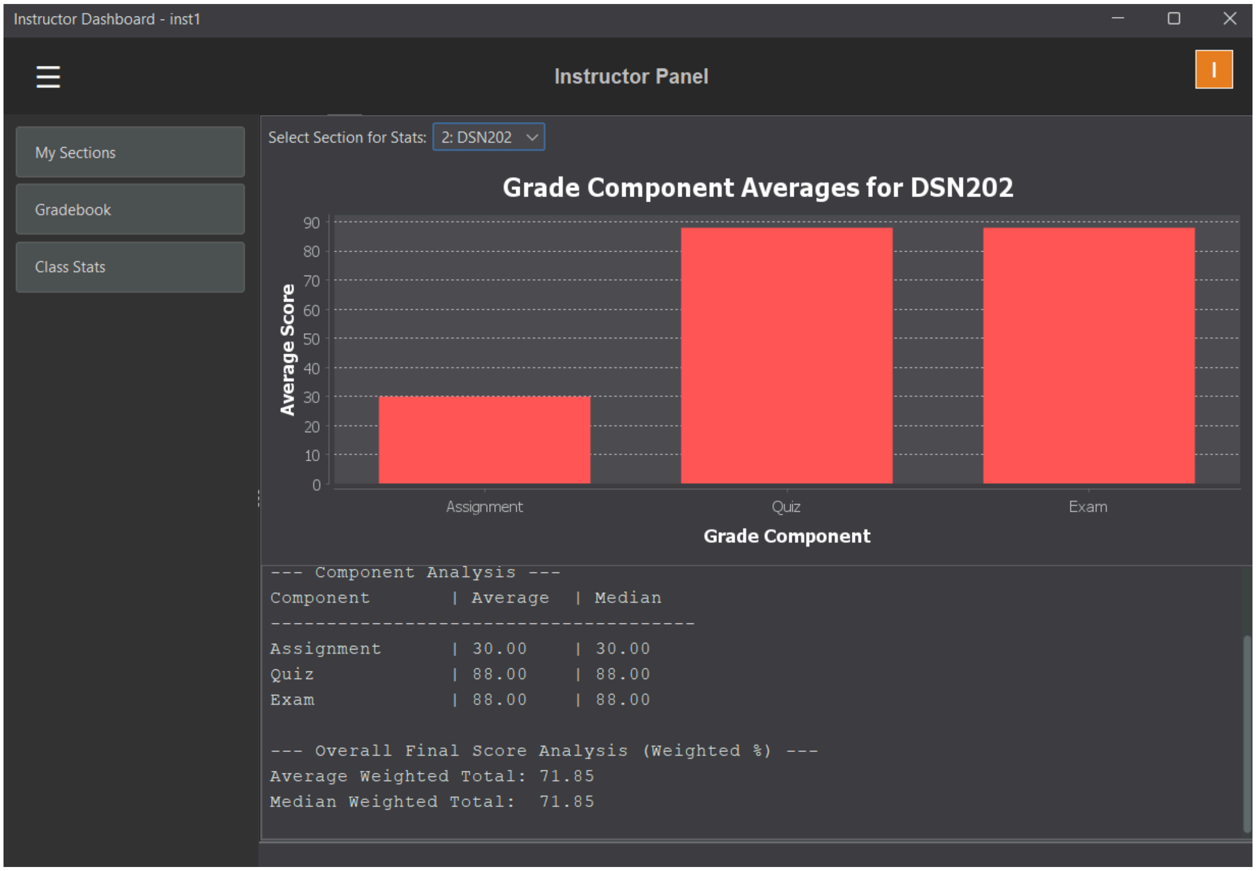
Component Name:

Score Obtained:

Max Marks (e.g. 50):

Weightage % (e.g. 20):

- Instructor Dashboard(Class Stats)-



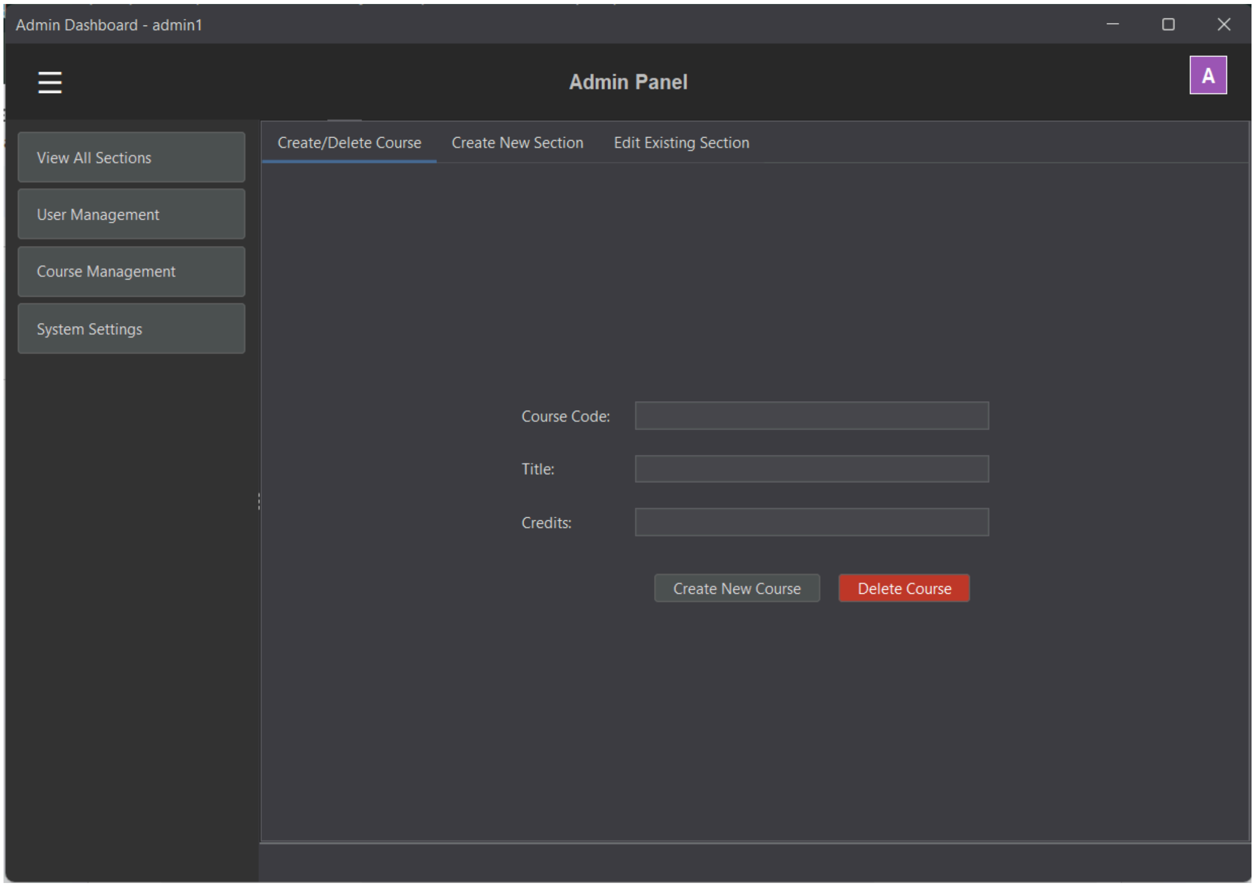
```
--- Component Analysis ---
Component        | Average   | Median
--------------------------------------
Assignment       | 30.00     | 30.00
Quiz             | 88.00     | 88.00
Exam             | 88.00     | 88.00

--- Overall Final Score Analysis (Weighted %) ---
Average Weighted Total: 71.85
Median Weighted Total:  71.85
```

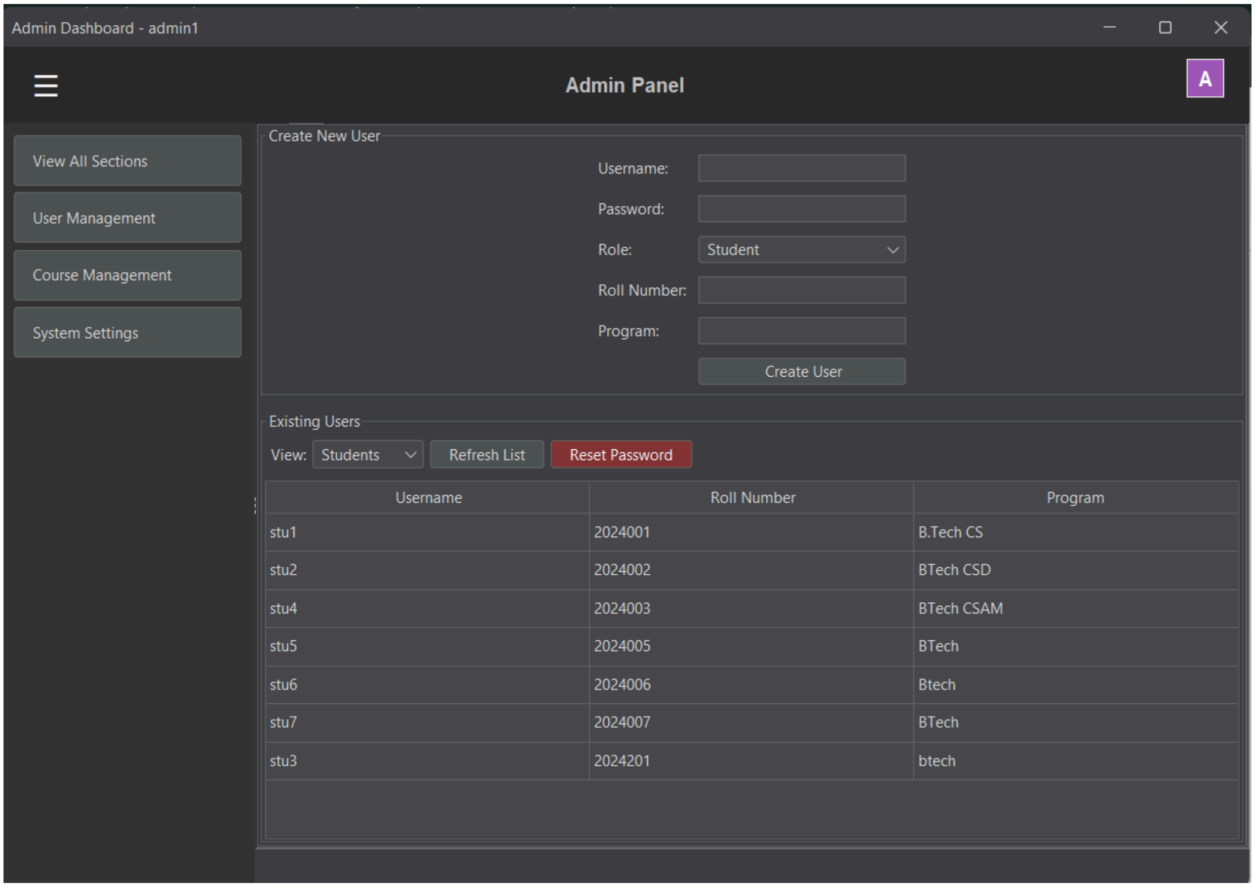
-

4. Admin Interface

- **Admin Dashboard - Course Management:**



- **Admin Dashboard - User Management:**



Create New User

Username:

Password:

Role: Student

Roll Number:

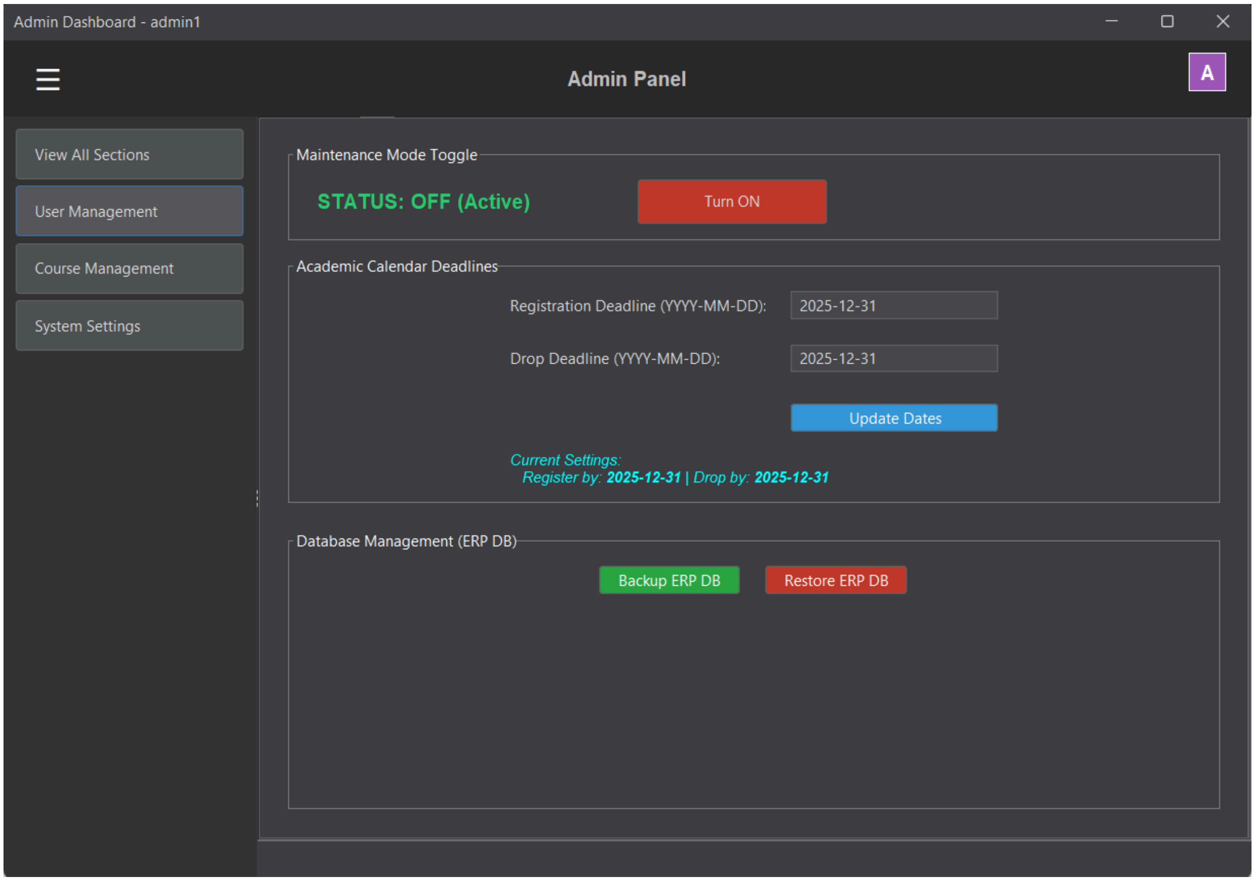
Program:

Existing Users

View: Students

Username	Roll Number	Program
stu1	2024001	B.Tech CS
stu2	2024002	BTech CSD
stu4	2024003	BTech CSAM
stu5	2024005	BTech
stu6	2024006	Btech
stu7	2024007	BTech
stu3	2024201	btech

- **Admin Dashboard - System Settings:**



STATUS: OFF (Active)

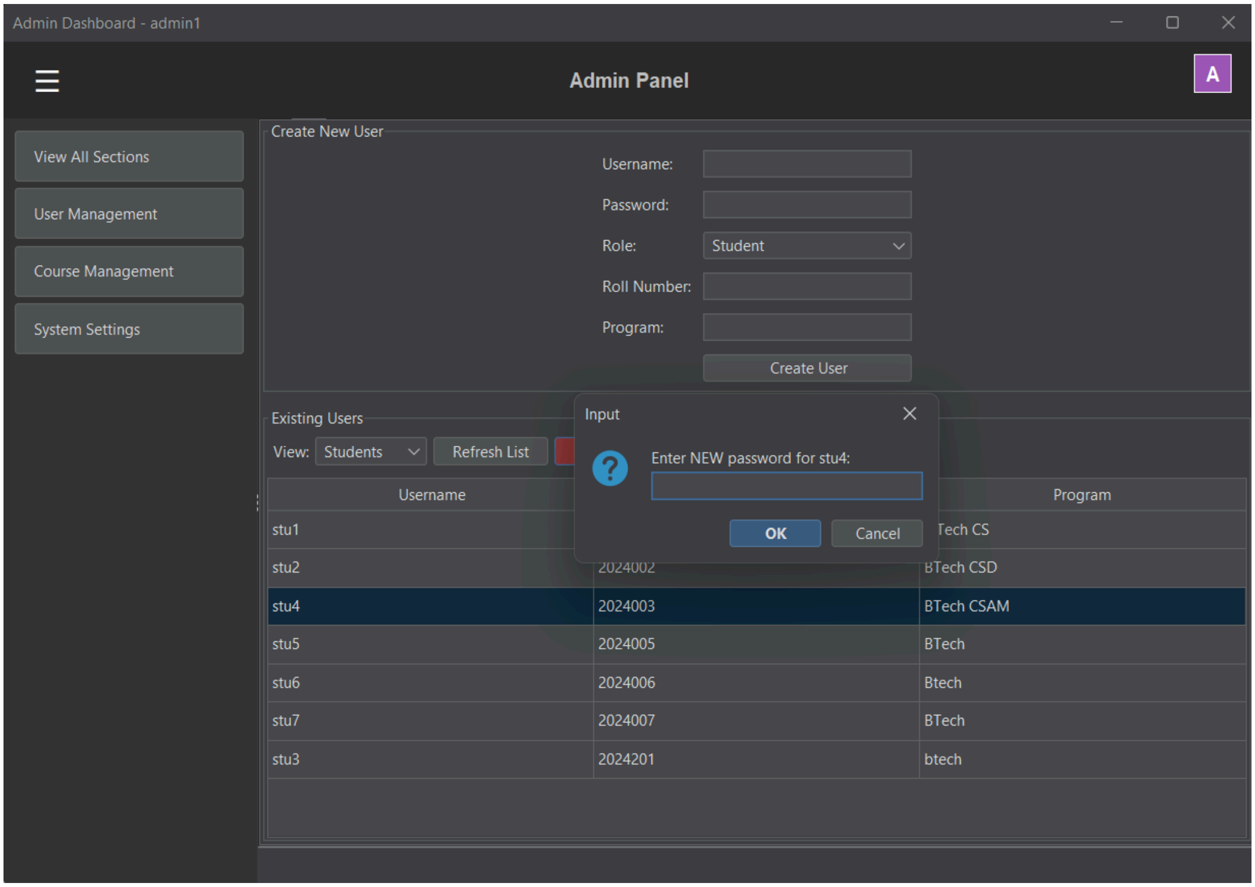
Registration Deadline (YYYY-MM-DD): 2025-12-31

Drop Deadline (YYYY-MM-DD): 2025-12-31

Current Settings:
Register by: ***2025-12-31*** | *Drop by:* ***2025-12-31***

-

- **Change Password Dialog:**



Role: Student

View: Students

Enter NEW password for stu4:

Username		Program
stu1		Tech CS
stu2	2024002	BTech CSD
stu4	2024003	BTech CSAM
stu5	2024005	BTech
stu6	2024006	Btech
stu7	2024007	BTech
stu3	2024201	btech