

# **REPORT**

## **Advanced Programming (AP) ERP-Project**

**Group 99** – Vaibhav Samrat Waghaye (2024597) and Mahi Yadav (2024326)

### **University ERP System Documentation**

#### **1. Project Overview**

This Role-Based Academic Management System is a desktop application built with Java Swing. It supports three distinct workflows Student, Instructor, and Admin. The system is architected with a dual-database design separating authentication from academic data to ensure security. When login users are greeted by a visually engaging interface and routed to a role-specific dashboard.

#### **2. Role Enforcement & Access Control**

The system enforces strict access control across two architectural layers-

- **Mainframe (UI) Layer**- Controls visibility. For example, a Student cannot physically access or see Admin dashboards.
- **Service (Business) Layer**- Enforces logic. Even if UI restrictions are bypassed, service methods validate permissions (e.g., ensuring an Instructor is actually assigned to a section before updating grades) before committing changes.

#### **Role Permissions-**

- **Admin**- Full control over users, courses, sections, and system settings (Maintenance Mode).
- **Instructor**- Restricted to managing their assigned sections, viewing rosters, and entering grades.
- **Student**- Read-only access to their specific enrollments, timetable, and final grades.

#### **3. Maintenance Mode**

Maintenance Mode ensures system integrity during updates. It is implemented redundantly-

- **UI Level**- When enabled by an Admin, the interface for Students and Instructors switches to "View-Only." Input fields and action buttons are disabled, and a banner is displayed.

- **Service Level-** A global interceptor checks the settings table before any write operation. If Maintenance Mode is active, the database rejects the request, preventing unauthorized data modification.

## 4. Database Architecture

The application utilizes two distinct databases to separate security credentials from domain data.

### A. Authentication Database (`auth_db`) Handles secure login and user identity.

- **users**- Stores usernames, secure **BCrypt** password hashes, roles (Admin/Instructor/Student), and account status.
- **password\_history**- Maintains a history of password changes for security auditing.

### B. Academic Database (`erp_db`) Manages the core university operations.

- **User Profiles**- students, instructors (Linked to `auth_db` via `user_id`).
- **Curriculum**- courses (Global catalog), sections (Specific instances with time/room).
- **Academic Records**- enrollments (Links students to sections), grades (Legacy support).
- **Grading System**- assessment\_components (Max marks/weights), assessment\_scores (Individual scores), final\_grades (Final letter grades).
- **System Config**- settings (Stores global flags like `maintenance_mode`).

## 5. Key Features

- **Security**- Passwords are never stored in plain text (BCrypt hashing used).
- **User Experience**- Custom "Mountain" background login, distinct dashboards per role, and intuitive navigation.
- **Reliability**- Centralized service logic prevents code duplication and ensures consistent validation.
- **Lifecycle Management**- inactive users are strictly blocked from login.

## 6. Use-Case Summary

- **Admin**- Manage Users, Create Courses/Sections, Assign Instructors, Toggle Maintenance Mode.
- **Instructor**- View Teaching Schedule, Grade Students (granular assessment scores), Compute Final Grades.
- **Student**- Browse Courses, Register/Drop Sections, View Timetable, Download Transcript.

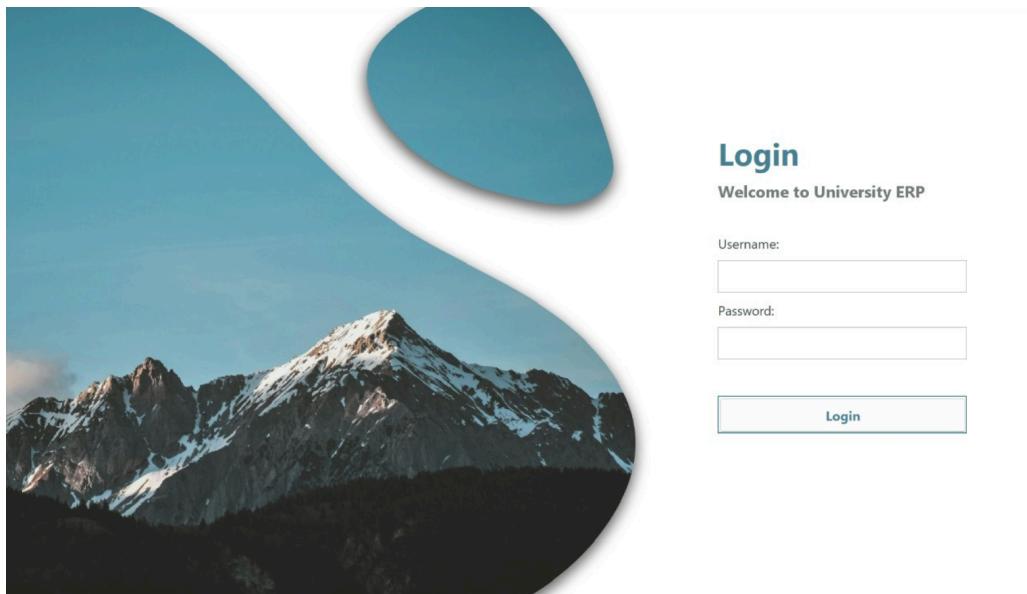
## 7. Entity Relationships

The data model relies on four core entities- **Student**, **Instructor**, **Section**, and **Enrollment**.

- A **Section** is an instance of a Course, taught by one Instructor.
- **Enrollment** acts as a many-to-many bridge, linking Students to Sections and serving as the container for their academic performance (Grades).

**Below are screenshots of our app-**

Login page-



## Login

Welcome to University ERP

Username:

Password:

Incorrect password. Attempts left: 4

When an incorrect password is entered , it shows the amount of attempts left , when out of attempts it locks the ERP

## Login

Welcome to University ERP

Username:

Password:

Too many failed attempts. Login disabled.

## Admin dashboard

The screenshot shows the Admin dashboard with a dark sidebar on the left containing navigation links: Admin Panel (System Management), Dashboard, Users, Courses, Reports, and Settings. The main area features a welcome message "Welcome back, admin1!" and a "Quick Actions" section with six buttons: Add Student (Create new student), Add Instructor (Create instructor), New Course (Add course), New Section (Create section), Settings (System settings), and Maintenance Mode (Turn on). Below this is a "Management Dashboard" section with three cards: User Management (Manage all users, with sub-points: View users, Add users, Edit profiles, Reset passwords, and an Open button), Course & Section Management (Manage courses and sections, with sub-points: Create courses, Manage sections, Assign instructors, Enrollments, and an Open button), and System Administration (System tools, with sub-points: Maintenance mode, Backup DB, System settings, Logs, and an Open button).

## Admin functionalities-

### Course Management

View and manage all courses in the system

1	Calculus	1
2	Operating Systems	1

[Add Course](#)[Edit Course](#)[Delete Course](#)[Manage Sections](#)

### User Management

View and manage all users

admin1	admin
inst1	instructor
stu1	student
stu2	student
test1	student

[Add User](#)[Delete User](#)[Back to Dashboard](#)

## Maintenance mode in admin-

The screenshot shows a dark-themed Admin Panel interface. On the left is a sidebar with options: Restore, Move, Size, Minimize, Maximize, Dashboard (selected), Close, and Alt+F4. The main area has a yellow header bar with the text "MAINTENANCE MODE ACTIVE". Below it is a welcome message "Welcome back, admin1!" and a "System Administrator" title. A sub-header says "Manage your university system". There are four cards in a grid: "Total Students" (3 Enrolled), "Instructors" (1 Active), "Courses" (2 Available), and "Sections" (1 This semester). Below these are "Quick Actions" buttons for "Add Student", "Add Instructor", "New Course", "New Section", "Settings", and "Disable Maintenance". At the bottom is a "Management Dashboard" section with three buttons: "User Management" (green), "Course & Section Management" (purple), and "System Administration" (grey).

## Backup (Bonus)

A modal window titled "System Administration" with a close button (X) at the top right. The main content area contains the text "Choose an action:". Below it are three buttons: "Backup Database" (highlighted with a dashed border), "Restore Database", and "Cancel".

## Student Dashboard-

The dashboard features a sidebar with 'Quick Actions' including 'Browse Courses', 'Register/Drop', 'My Timetable', 'My Grades', and 'Transcript'. The main area displays a welcome message, roll number, year, and three summary cards: 'Enrolled Courses' (1), 'Total Credits' (4), and 'Progress' (40%). A large button links to 'View All Enrolled Courses'.

Enrolled Courses	Total Credits	Progress
1	4	40%

Things students can do and see like timetable , grades , courses etc-

A table lists courses by code, title, credits, section, instructor, schedule, room, enrollment status, and a 'Status' column. A dropdown menu allows sorting by course code.

Sort Courses By:		Course Code (Prefix + Number)						
Code	Title	Credits	Section	Instructor	Schedule	Room	Enroll	Status
CS202	Operating Systems	3						
MATH101	Calculus	4						

## Available Courses

Search:

Course Code	Course Title	Credits	Section	Instructor	Day/Time	Room	Enrolled/Capa...	Status
CS202	Operating Systems 3		2	inst1	Wed 2:00 PM	Room 202	0/50	Available
MATH101	Calculus	4	1	inst1	Mon 10:00 AM	Room 101	2/60	Already Enrolled

Add to Timetable

Close

## My Weekly Timetable

08:00 - 09:00								
09:00 - 10:00								
10:00 - 11:00	<b>MATH101</b> Calculus Room 101 • inst1							
11:00 - 12:00								
12:00 - 13:00								
13:00 - 14:00								

Close

## □ My Grades & Performance

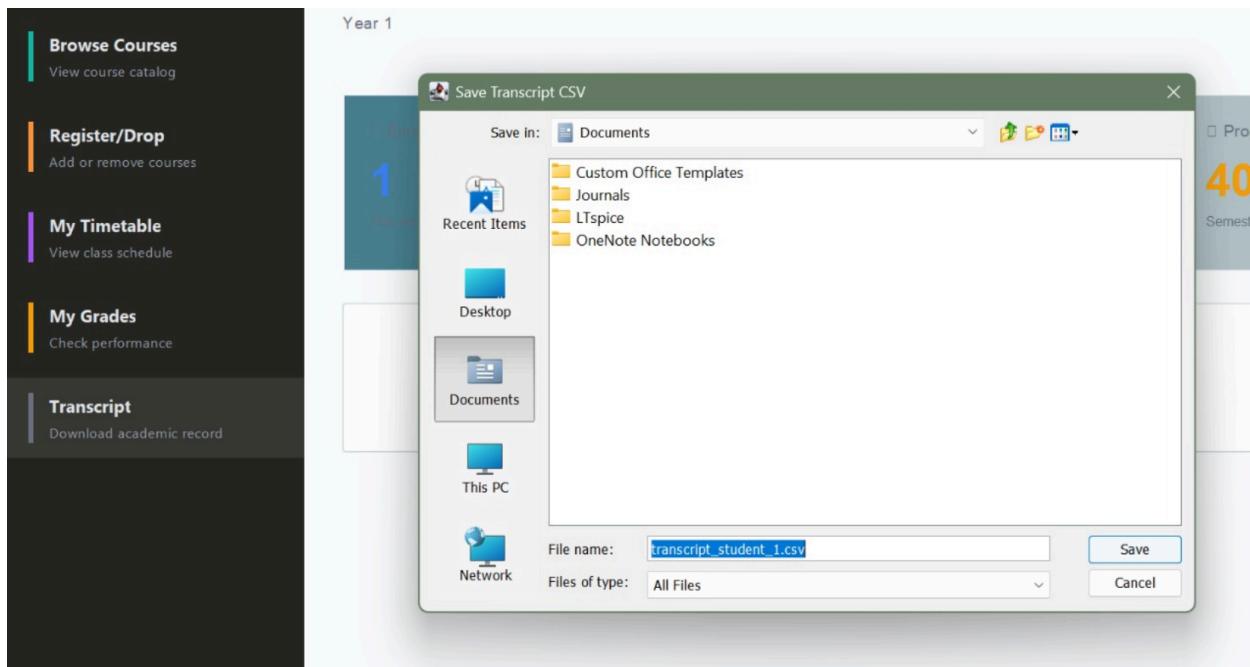
### MATH101 - Calculus

4 Credits • inst1

Assessment	Score	Weight %	Contribution
No assessments defined	-	-	-
Final Score	Not Computed	-	-

**Close**

## Bonus - CSV



## Instructor dashboard-

Welcome, inst1!

Computer Science Department

2 Active Sections

My Sections 2 This semester	Total Students 2 Across all sections	Avg. Class Size 2 Students per section	Pending Grades 0 Need attention
-----------------------------------	--	--	---------------------------------------

### My Teaching Sections

Course Code	Course Title	Section ID	Schedule	Room	Enrolled	Avg Grade	Action
CS202	Operating Syste... 2		Wed 2:00 PM	Room 202	0/50	—	<a href="#">Manage</a>
MATH101	Calculus	1	Mon 10:00 AM	Room 101	2/60	—	<a href="#">Manage</a>

Once we click manage , we can manage whatever we want to

### My Teaching Sections

Course Code	Course Title	Section ID	Schedule	Room	Enrolled	Avg Grade	Action
CS202	Operating Systems	2	Wed 2:00 PM	Room 202	0/50	—	<a href="#">Manage</a>
MATH101	Calculus	1	Mon 10:00 AM	Room 101	2/60	—	<a href="#">Manage</a>

### Assessment Weight Rules

Configure assessment components and their weights

Component Name	Weight %	Max Marks

[Delete Selected](#) [Add Component](#) [Save Rules](#)

Assigning grades of students-

Grade Entry	
Enter scores for each assessment component	
Student	Final Grade
stu1	0.0
stu2	0.0

[Save All Scores](#)

Instructor can check class statistics-

Class Statistics			
Performance overview and grade distribution			
Students	Average	Highest	Lowest
2	0.00	0.00	0.00
Grade Distribution			
No scores available			

### **How maintenance and roles are enforced-**

In our ERP application, maintenance control and academic rules are enforced through a layered architecture of access, service, and UI. The **MaintenanceAccess** class stores the current maintenance state in memory and syncs it with the database. The **AdminService** ensures only administrators can toggle this state, updating the settings table and propagating changes system-wide. UI panels (Student, Instructor, Admin) check the shared flag at runtime to enable or disable action for example, blocking course registration or restricting instructor operations. Academic rules, such as semester ranges and enrollment limits, are enforced in the service layer, while the UI simply calls service methods. This design ensures consistent application of maintenance restrictions and academic policies across all modules.