AcadTrack: Student CPI and Credit Management System

Rishit Bhutra (210857)

April 24, 2025

Abstract

This project is a web-based academic tracking system developed using **Flask** and **SQLite**. It allows students to enroll in courses, view academic progress (SPI/CPI), and ensures automated academic session progression. Administrators can upload new batches, add courses, manage enrollments, and upload grades.

1 Technology Stack

- **Backend:** Python (Flask)
- **Database:** SQLite
- **Frontend:** HTML, Jinja2 templating
- **Deployment:** Local VM with NAT port forwarding

2 Database Design

The database schema consists of the following relational tables:

- **Students** (student_id, name, *batch_id*)
- **Courses** (course_id, name, credits, course_type)
- **Enrollments** (*student_id*, *course_id*, year, semester, grade)
  - *student_id* references **Students**, *course_id* references **Courses**
- **Batches** (batch_id, batch_year)
- **CurriculumRequirements** (course_type, required_credits)
- **Session** (id, year, semester)
- **EnrollmentStatus** (year, semester, is_enrollment_open)

3 Core Features

Student Features

- **Student Login:** Students log in using their unique ID.

- **Dashboard:** Displays current and past enrolled courses, earned credits, and academic progress.
- **SPI/CPI Calculation:**
  - SPI (Semester Performance Index) is calculated as a credit-weighted average for each semester.
  - CPI (Cumulative Performance Index) is calculated based on the most recent attempts of all courses.
- **Enrollment:** Students can enroll in courses during open enrollment periods, provided they haven't graduated.
- **Graduation Eligibility:** Automatically determined based on
  - No failed or pending grades.
  - Credits required are fulfilled in the categories IC (Institutional Core), DE (Departmental Elective), and OE (Open Elective).

Admin Features

- **Admin Login:** Accessed using a designated admin ID.
- **Dashboard:** Allows filtering and viewing of students and courses by batch or type.
- **Batch Management:** Uploads new student batches (only in Semester 1), with auto-generated student IDs.
- **Course Management:** Add new courses with duplicate name validation.
- **Enrollment Control:** Open and close enrollments manually for each semester.
- **Grade Submission:**
  - Grades can only be submitted after enrollment is closed.
  - Admin selects a student-course pair with a pending grade.
  - Upon submission of all grades, the semester automatically advances.
- **Automatic Semester Advancement:**
  - When all grades are submitted, the system automatically advances to the next semester.
  - By default, enrollment for the new semester is open.

4 Future Improvements

- JWT-based log-in for multiple concurrent sessions.
- Role-based access through decorators.
- CSV upload for bulk course or student data.
- Dynamic AJAX interfaces for smoother UX.

5 Conclusion

This project demonstrates a comprehensive academic record system with dynamic semester progression, curriculum validation, and user-role-based access. The modular backend and scalable database structure allow easy expansion and deployment in educational institutions.