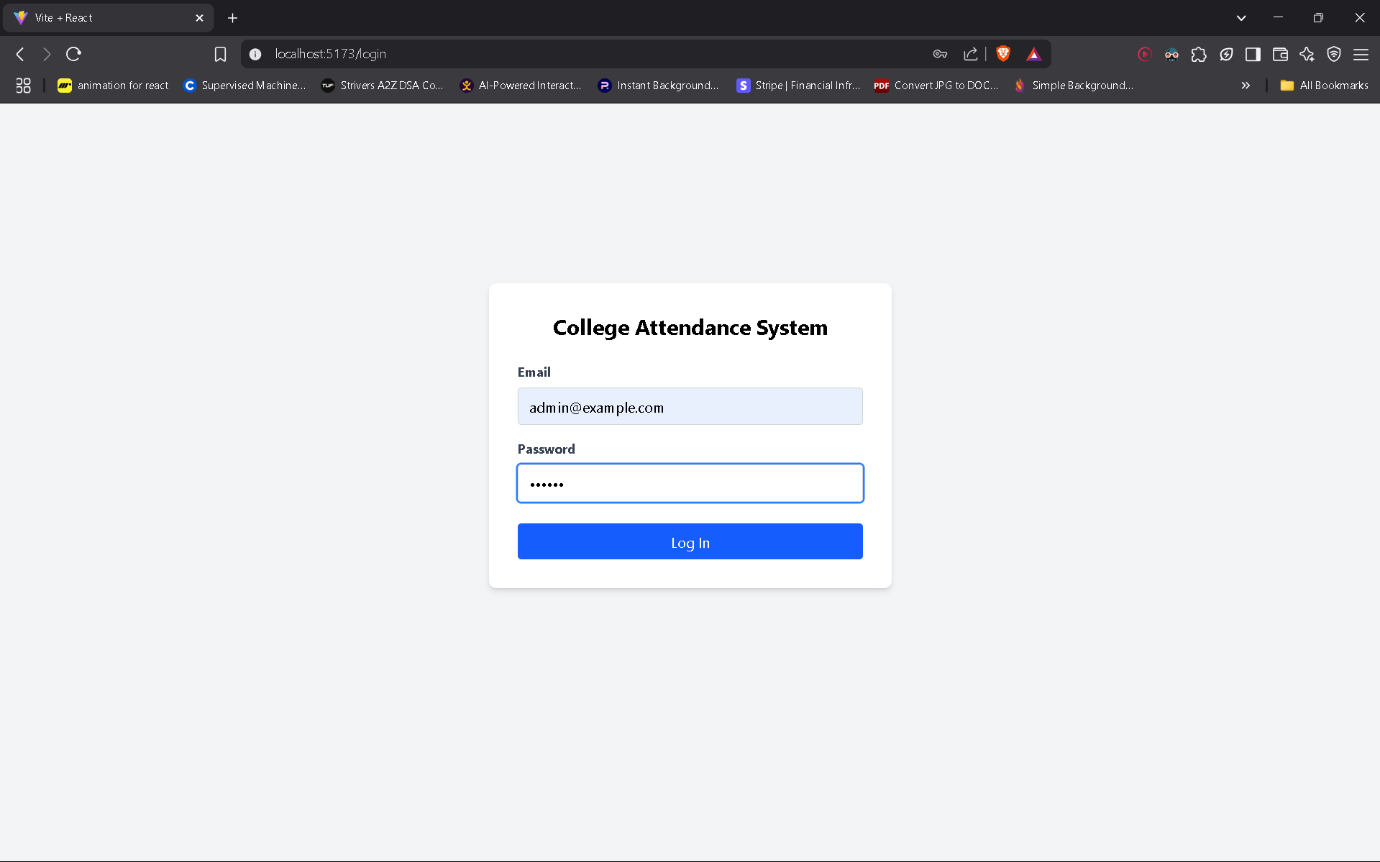
Title: Student Attendance System

Author: Samar Joshi

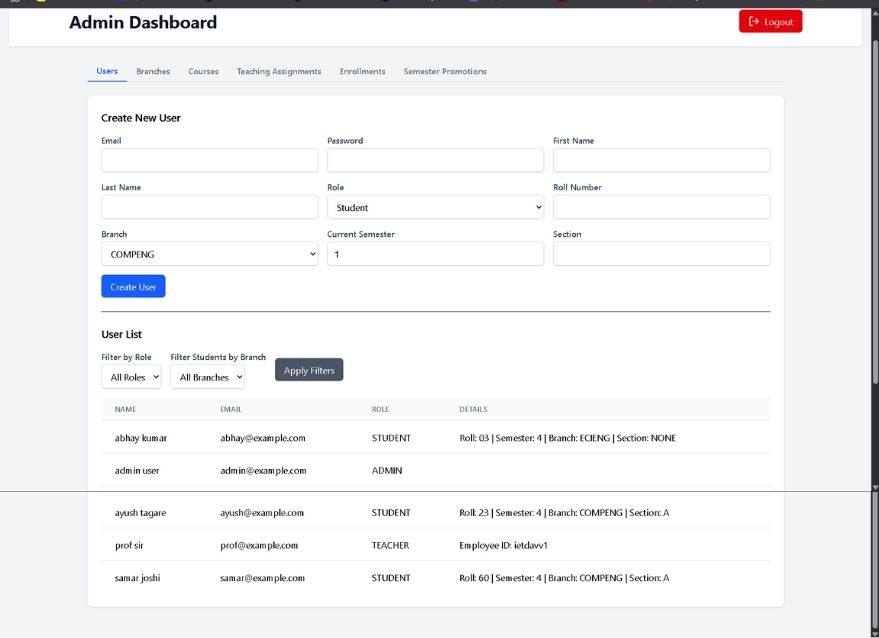
Branch and Section: Computer Engineering, Section ‘A’

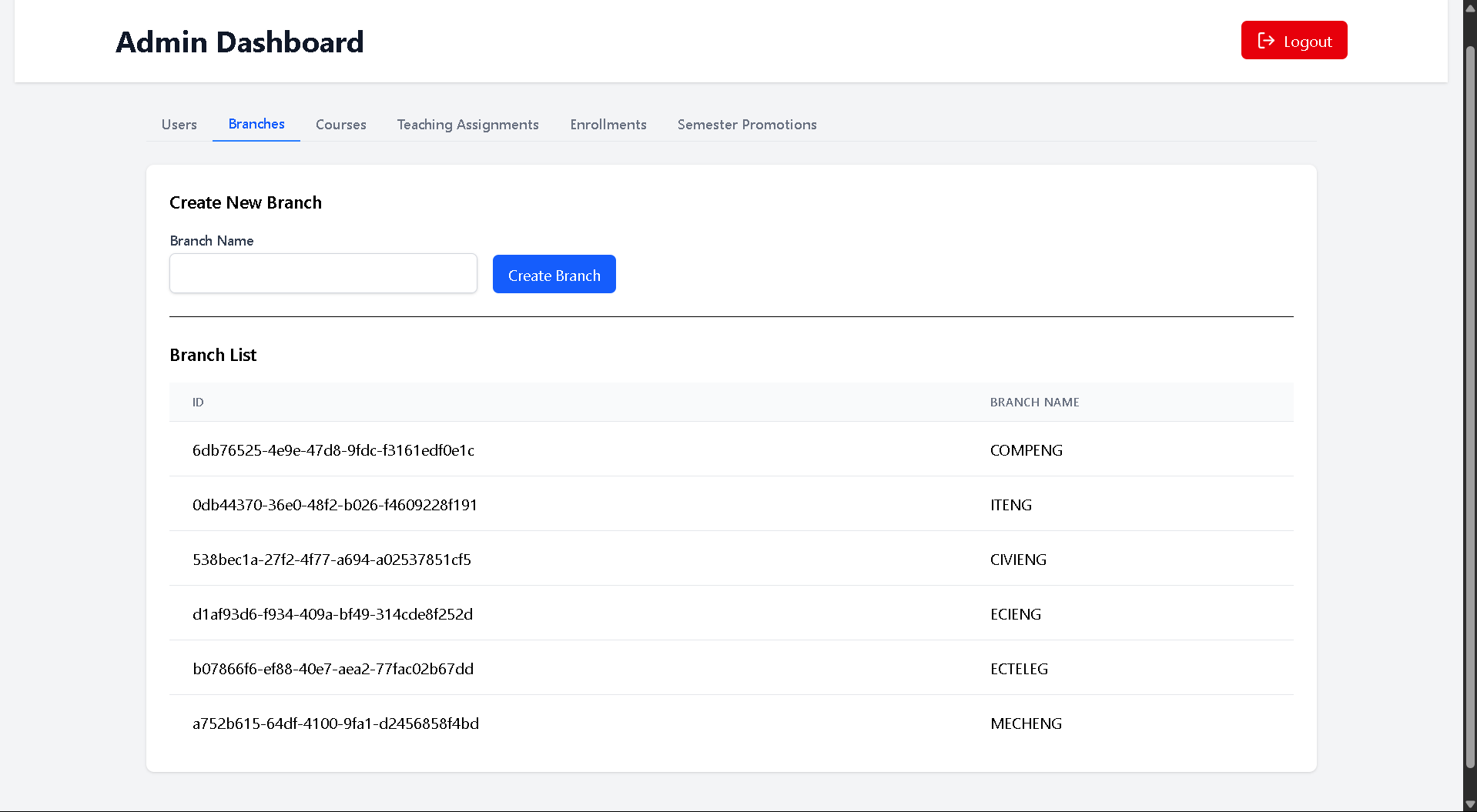
1. 🔐 Login Page:

The system begins with a secure login page. Based on credentials, users are redirected to role-specific dashboards (Admin, Teacher, or Student). This ensures controlled access to functionalities according to user privileges.

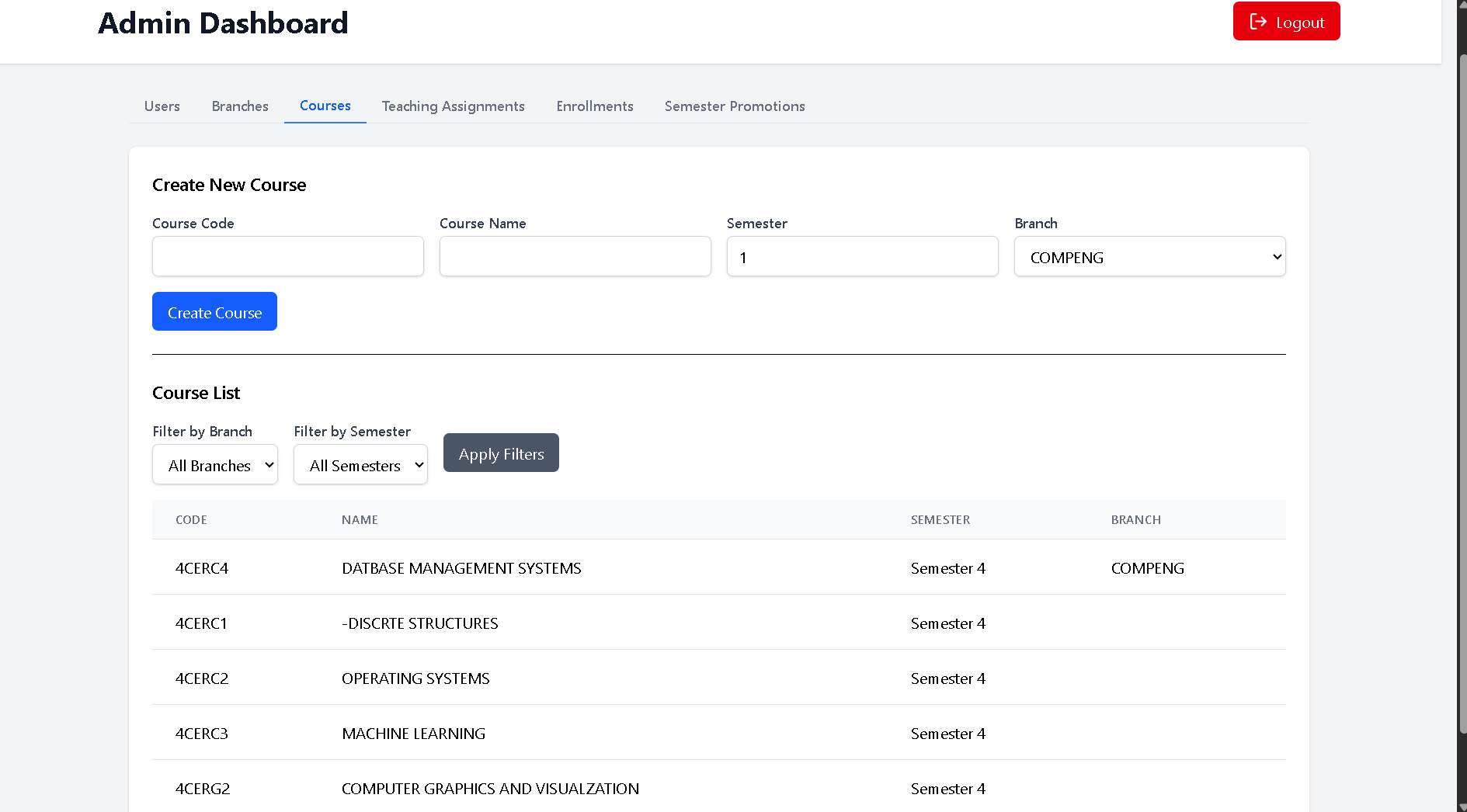
1. 👨‍💼 Admin Pages:  
   Users Section:

Admins can add, view, and manage all users—Admins, Teachers, and Students. Role-based user management ensures database access remains tightly controlled.



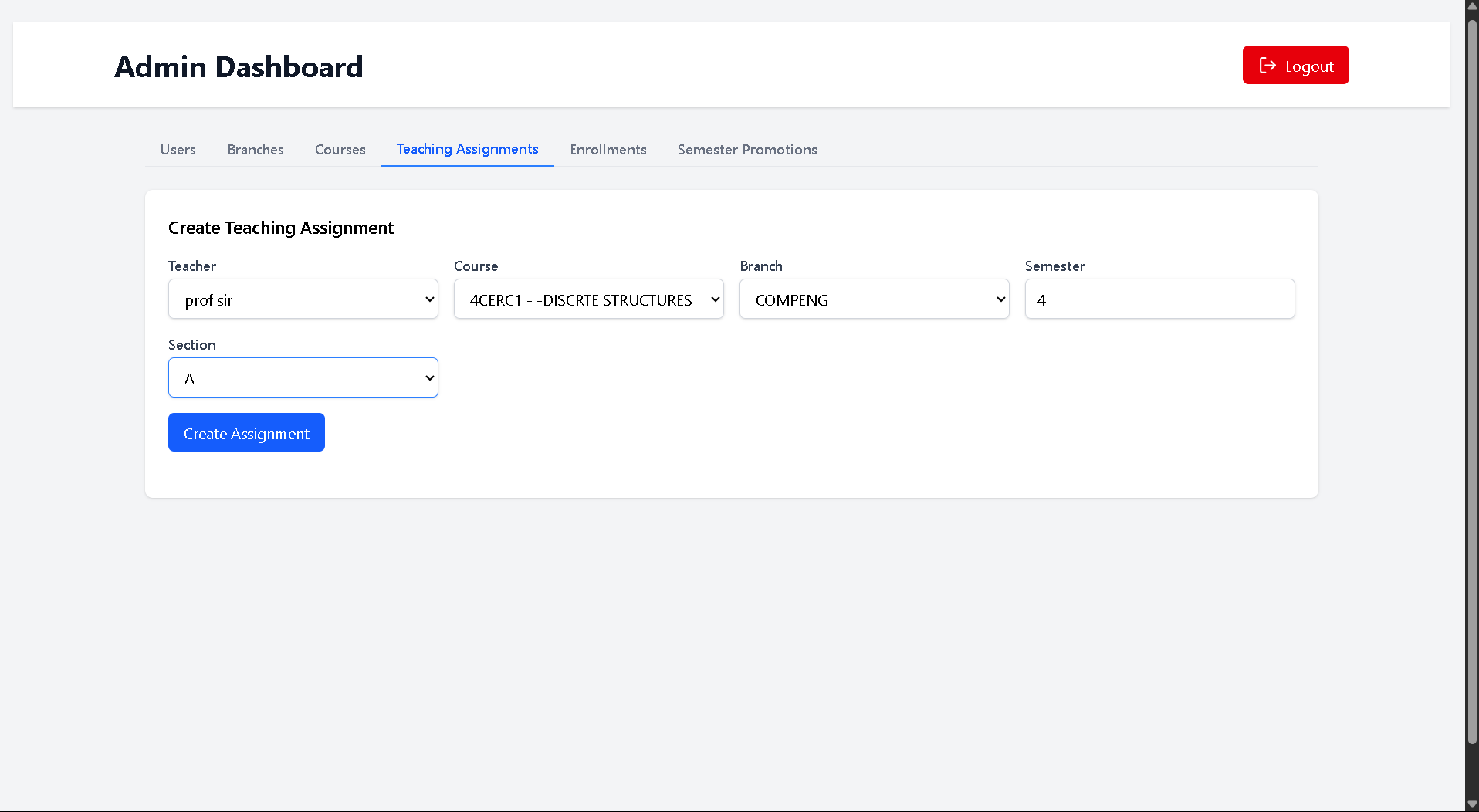
  
  
  
  
  
  
  
  
Branches:  
Admins can define and manage various academic branches available at the university.

Courses:   
Courses can be added and linked with branches, sections, and semesters for better curriculum structuring.



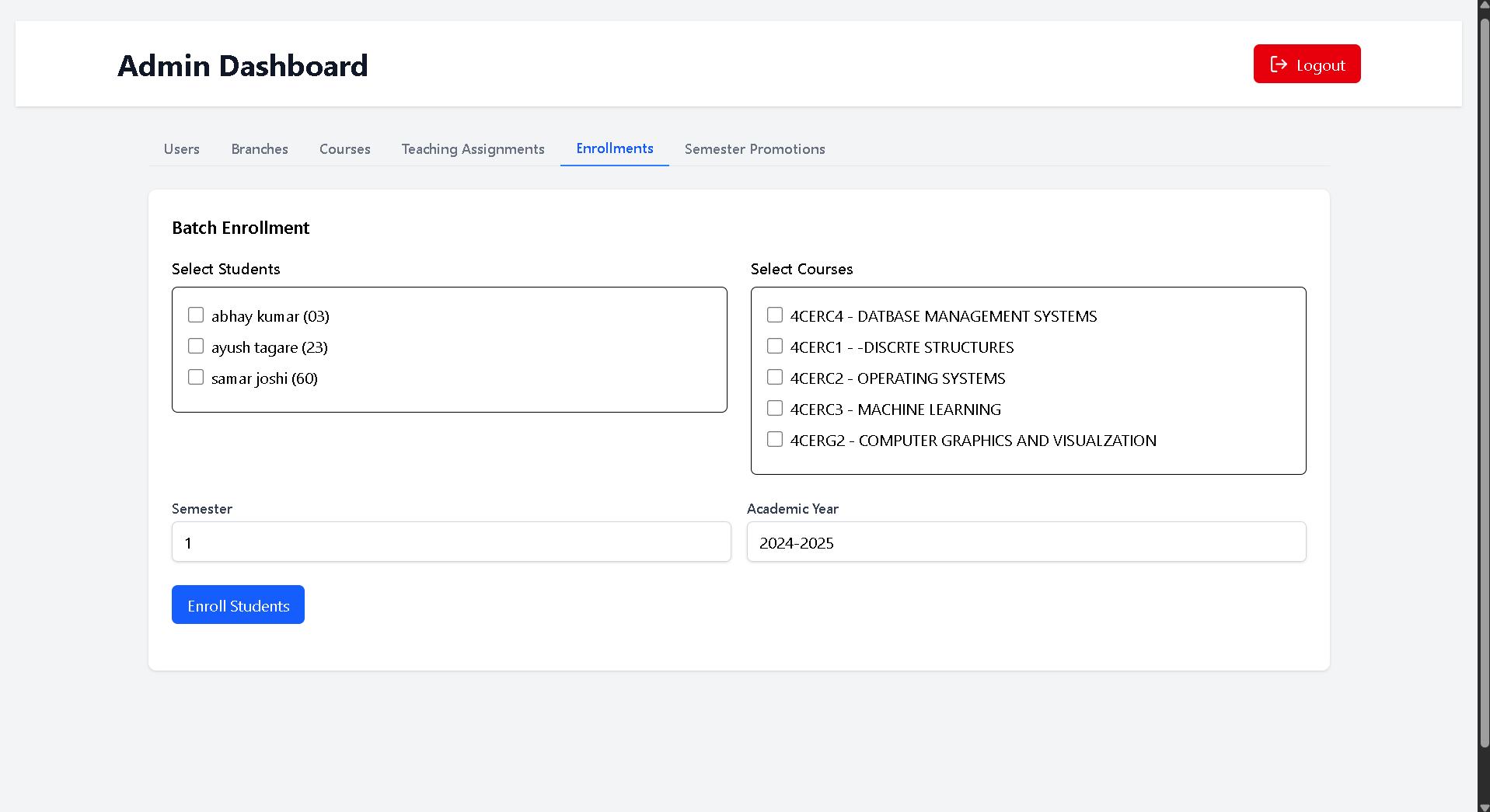
Teaching Assignments:

Admins assign teachers to specific courses, branches, sections, and semesters. This relation is used to control what data teachers can access.



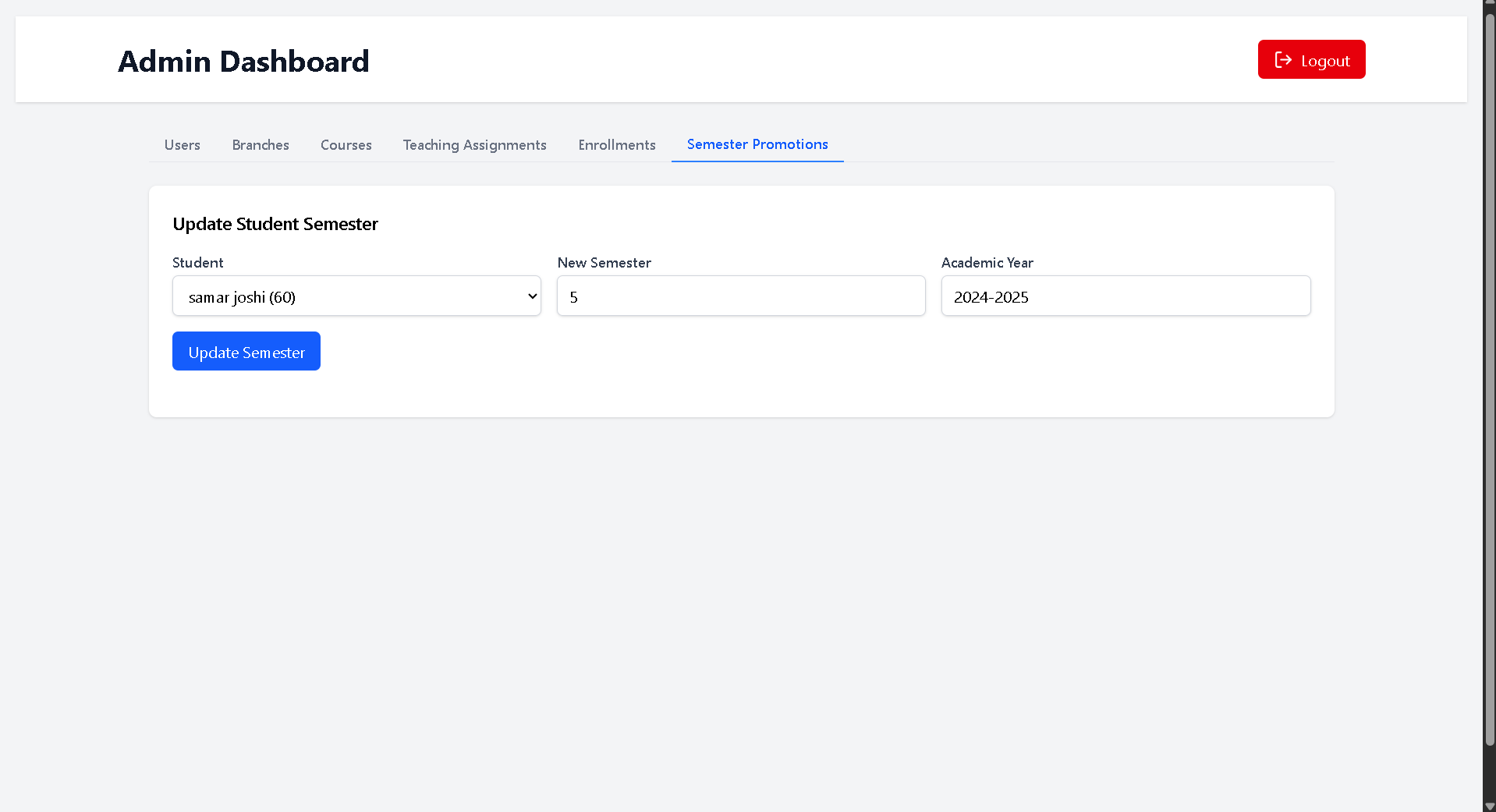
Enrolments:

Admins enroll students into specific branches, sections, and semesters. These enrollments determine what attendance data is shown to students and teachers.

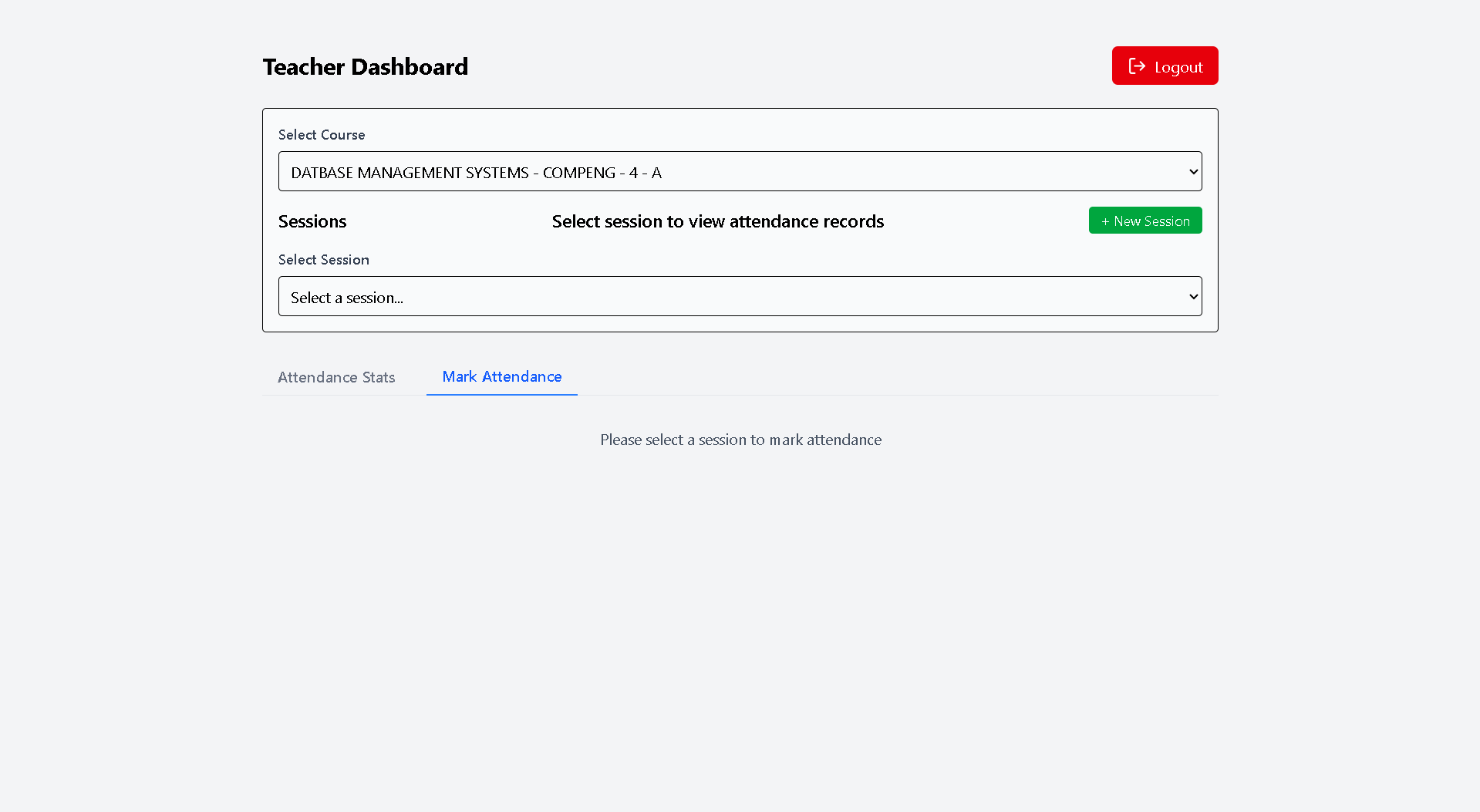


Semester Promotions:

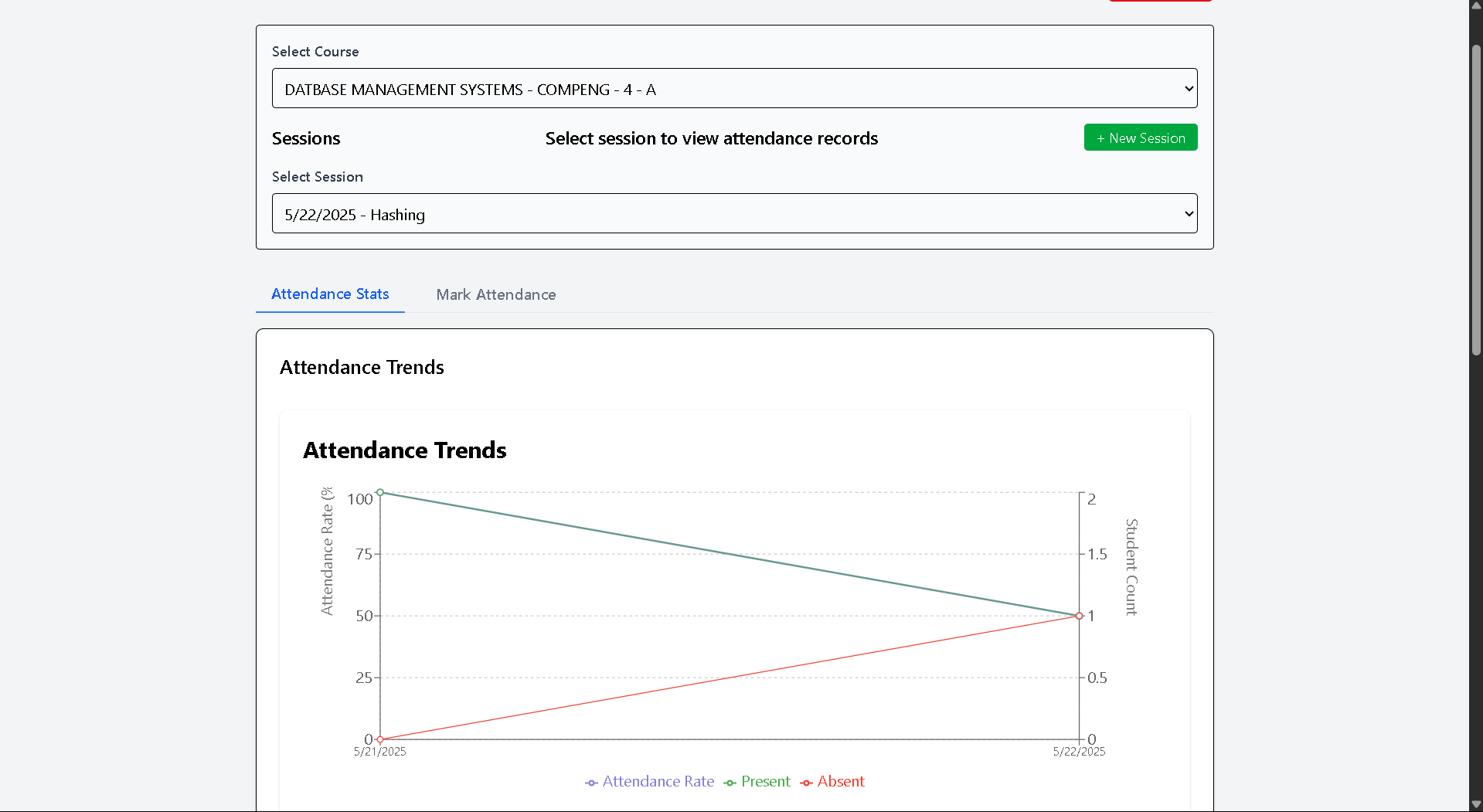
Admins can promote or retain students by updating their current semester. This simplifies academic year transitions.

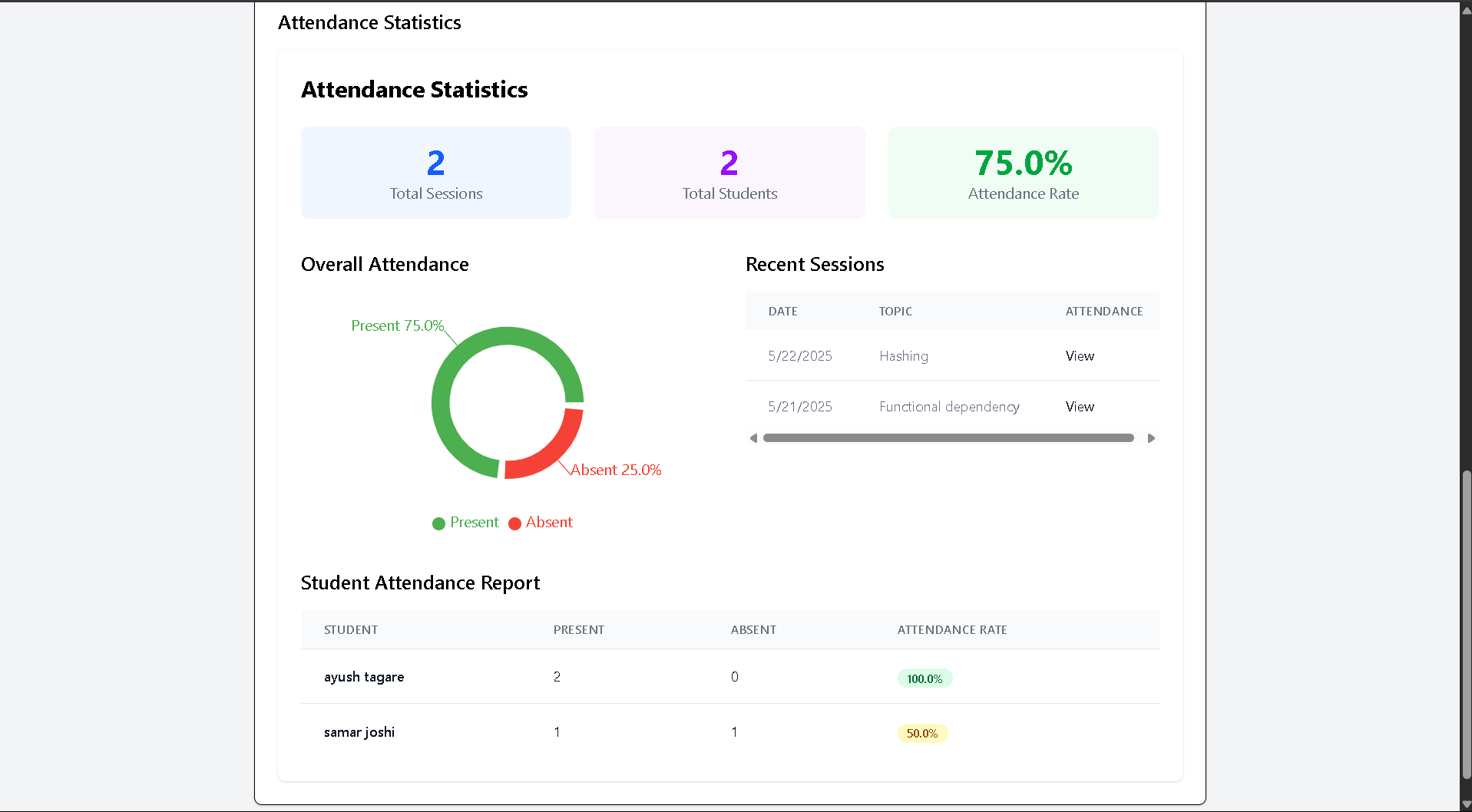


1. 👨‍🏫 Teacher Pages:  
   **Teacher Dashboard:**

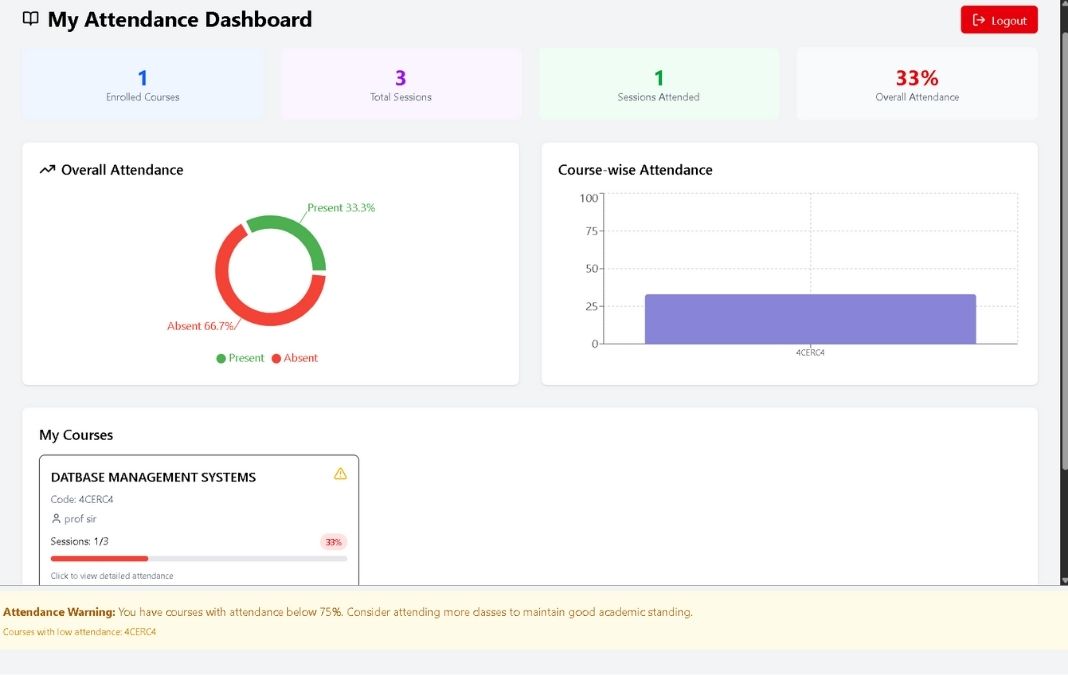
Teachers can log in and view their assigned courses and sections. This dashboard gives a quick overview of responsibilities.

Attendance Stats:   
Teachers can mark attendance for students across different sessions. Attendance records are stored per session and course, and can be reviewed at any time.





1. 🎓 Student Page:  
   Student dashboard   
   Students can view their own attendance statistics presented through clean visuals and graphs. This provides transparency and self-tracking for learners.

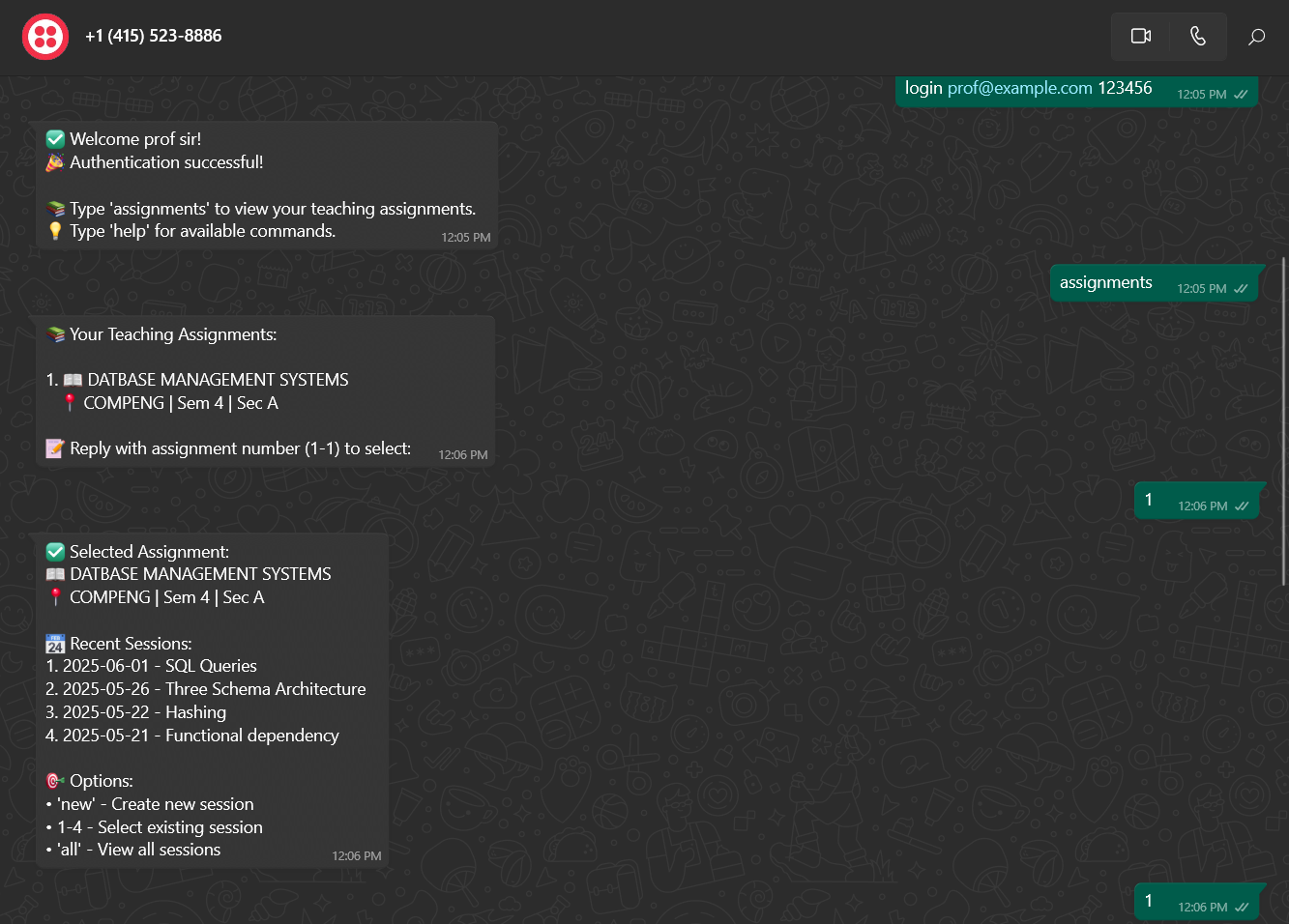


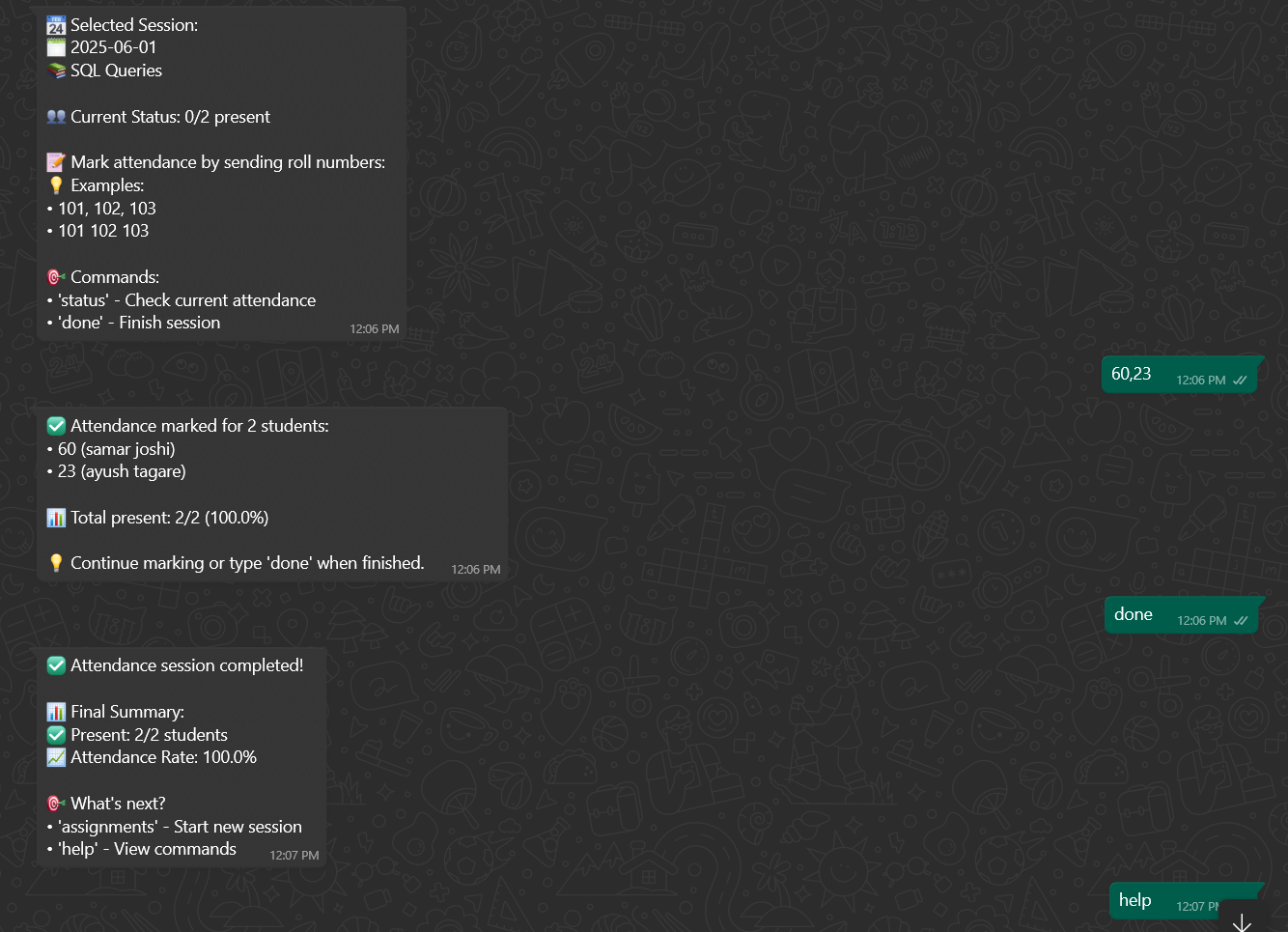
🤖 WhatsApp Bot Integration:

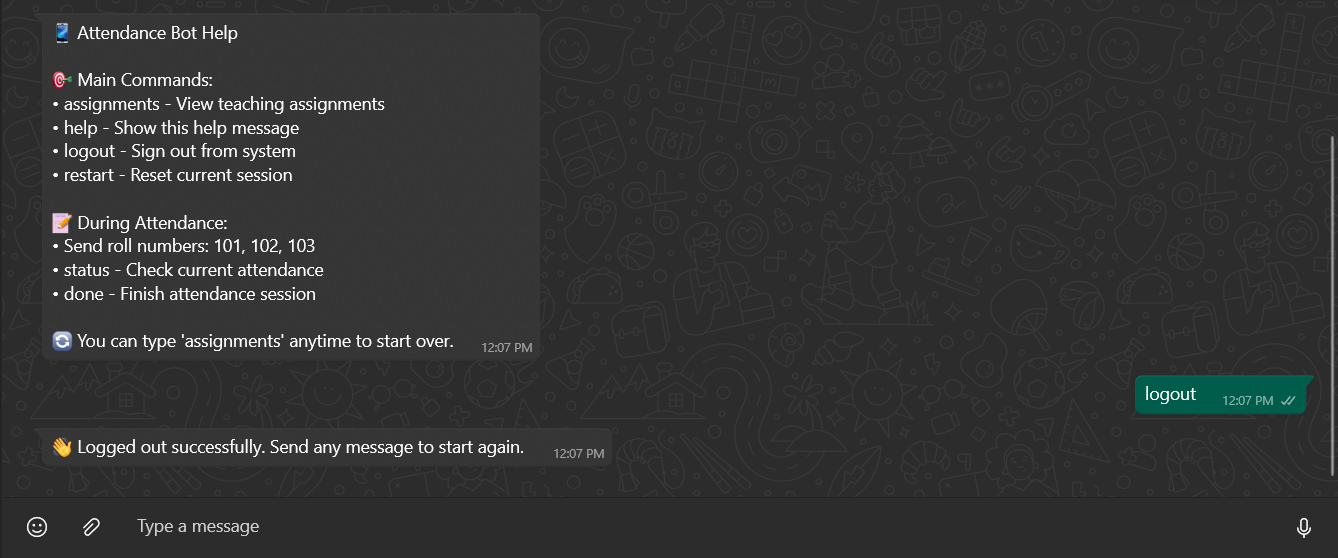
To improve accessibility and real-world usability, a WhatsApp Bot has been integrated into the system. Teachers can:

* Log in using their mobile number.
* View assigned courses and batches.
* Mark attendance for sessions via chat.
* View session-wise attendance summaries directly on WhatsApp.

This bot bridges the gap between classroom and system, allowing attendance to be handled from mobile devices with ease.







🔧 Tech Stack:

* Frontend: React.js + Tailwind CSS
* Backend: Node.js with Express
* Database: PostgreSQL (via NeonDB)
* ORM: Prisma (for efficient query handling and optimization)
* Bot Platform: WhatsApp API integration (details handled via server routes)

📌 Note:

This project was developed with the intent to apply DBMS concepts in a practical scenario. The full feature set, including role-based access control, real-time data manipulation, and mobile integration, was designed to simulate a scalable institutional solution.

(As discussed during class, project submissions like these were to be considered for evaluation — I hope this work reflects the effort and learning involved 😊.)