

**11139245**

CPEN 208- Software Engineering  
Project 1

## **Introduction**

This report outlines the design and implementation of a relational database for a Student Management System intended for the Computer Engineering Department. The system provides functionalities for managing student personal information, fee payments, course enrollment, and the assignment of lectures and TAs to courses. The database was developed using PostgreSQL, and a web application was built using Next.js 14 to provide an interface for users to interact with the system. This report includes details about the database schema, table creation, sample data insertion, a database function to calculate outstanding fees, and the web application development process.

## **Database Design**

### **Database Name**

student\_management\_system

### **Schemas**

The database uses a single schema named public by default for simplicity.

### **Tables and Attributes**

#### **students**

- 1.
1. student\_id (Primary Key, Integer, Auto Increment)
2. first\_name (text)
3. last\_name (text)
4. birthdate (Date)
5. email (Varchar)
6. phone (Varchar)
- 2.

#### **fees**

- 3.
1. fee\_id (Primary Key, Integer, Auto Increment)
2. student\_id (Foreign Key, Integer, References students(student\_id))

3. amount (Decimal)
4. paid\_date (Date)
5. status (Varchar)

4.

#### **courses**

5.

1. course\_id (Primary Key, Integer, Auto Increment)
2. course\_name (Varchar)
3. course\_description (Text)

6.

#### **enrollments**

7.

1. enrollment\_id (Primary Key, Integer, Auto Increment)
2. student\_id (Foreign Key, Integer, References students(student\_id))
3. course\_id (Foreign Key, Integer, References courses(course\_id))
4. enrollment\_date (Date)

8.

#### **lectures**

9.

1. lecture\_id (Primary Key, Integer, Auto Increment)
2. lecture\_name (Varchar)
3. email (Varchar)
4. phone (Varchar)

10.

#### **course\_lectures**

11.

1. course\_lecture\_id (Primary Key, Integer, Auto Increment)
2. course\_id (Foreign Key, Integer, References courses(course\_id))
3. lecture\_id (Foreign Key, Integer, References lectures(lecture\_id))

12.

#### **ta\_assignments**

13.

1. ta\_assignment\_id (Primary Key, Integer, Auto Increment)
2. course\_id (Foreign Key, Integer, References courses(course\_id))
3. lecture\_id (Foreign Key, Integer, References lectures(lecture\_id))
4. ta\_id (Integer, References lectures(lecture\_id))

## Web Application Development

### Tools and Frameworks

- **Frontend Framework:** Next.js 14
- **Database:** PostgreSQL
- **Authentication:** NextAuth.js
- **Styling:** Tailwind CSS

### Key Functionalities

1. **Login and Registration:** Secure user authentication with session management.
2. **Dashboard:** A user-specific dashboard displaying student information, courses, and outstanding fees.
3. **Course Enrollment:** Functionality for students to enroll in available courses.
4. **Fee Payment Tracking:** Display of paid and outstanding fees.

### Project Submission

The source code for the Next.js application, database scripts, and a backup of the database have been pushed to the GitHub repository. The repository URL is provided below.

### Conclusion

The project successfully demonstrates the design and implementation of a relational database for managing student information, course enrollment, and fee payments. The database is integrated with a web application built using Next.js, providing a comprehensive solution for the Computer Engineering Department's needs.

### GitHub Repository

<https://github.com/Ayisibea/University-Management-System>

### Appendices

1. **Database Backup:** A backup of the PostgreSQL database.
2. **Scripts:** SQL scripts for table creation, data insertion, and the outstanding fees calculation function.
3. **Source Code:** Source code for the Next.js application.

