ACADEMIC SECTION MANAGEMENT SYSTEM

Mihit Sreejith, Badekai Vijesh Ramachandra Bhat, Aditya Mandal, Soumyadeep Paul,
Akshit Sharma, Nitish Kumar Pinneti, Tammireddy Sri Vallabh

CS346 - Software Engineering Lab Assignment 2
Group 10B

CONTENTS

1 INTRODUCTION

1.1 Purpose

Develop a Software that manages various academic section activities of the institute. The system must cater to the requirements of different users like Dean, Registrar, Faculty, Students, Administrators and Officials for performing their assigned tasks.

1.2 Intended Audience

The intended audience for this document includes:

- Software developers and engineers responsible for designing and building the system, i.e, Group 10B members.
- Academic administrators, faculty members, and students who will use the system for their respective tasks and activities.

1.3 Intended Use

The Academic Management System is intended to be used as a centralized platform for managing various academic section activities within an institute. Users will interact with the system to perform tasks such as admissions and registration, course management, timetable generation, examination planning, grade generation, and information retrieval. The system is designed to provide user-friendly interfaces tailored to the specific needs of different user roles, facilitating efficient communication and collaboration across the academic community.

1.4 Product Scope

The product scope includes the development of a comprehensive Academic Management System that encompasses the following main modules:

- Admissions and Registration: Facilitating the enrollment process for prospective students and managing their registration details.
- Course Management: Allowing faculty members to plan and manage courses, including timetable generation and resource allocation.
- Examination Management: Handling exam scheduling, seating planning, grade generation, and result publication.
- The system will support user authentication and authorization mechanisms to ensure data security and privacy. Additionally, it will provide functionalities for information publishing and retrieval tailored to the disciplines of students, faculty members, and administrators.

2 OVERALL DESCRIPTION

2.1 User Needs

The system should mainly cater to three types of users -

1. Students
2. Faculty
3. Administrators

The Dean and Registrar role is given to certain registered administrators and have all the privileges held by any administrator.

The specific user needs are as follows-

1. Students:
   (a) Admission/Enrollment and course registration
   (b) Visibility into class schedules and exam dates
   (c) Access to Grades in enrolled courses
2. Faculty:
   (a) Float courses
   (b) Approve student course registrations
   (c) Visibility into class schedules and exam dates

(d) Give grades

3. Administrators:
   (a) Approve student admissions
   (b) Managing and creating user accounts
   (c) Timetable generation
   (d) Examination schedule, seating and other handling
   (e) Grades publication

2.2 Assumptions

The following simplifying assumptions have been taken initially. Some of the assumptions may be removed by further optimisation of the system when it is finalized.

1. All users have access to the same version of the database and the software.
2. There is only one department (CSE) and all the professors and courses and students belong to that department.
3. All admins. hold similar positions and have the same authority. Dean, registrar and examination officials are roles given to some admins and may have some extra authority on top of already existing ones.
4. All the courses are floated by the faculty before some end date and all the students register before the start of the semester.
5. The timetable generated does not take into account holidays and other interruptions and just provides a weekly schedule that is fixed for the whole semester.
6. The exam seating plan is generated once and is applicable to both midsems and end sem and any other examination taking place. All courses have the same examination structure.
7. On the system, students and faculty can register and admin have to approve their registration for the user accounts to be created. For admin accounts, they are to be added by other admin/super users.

3 SYSTEM FEATURES AND REQUIREMENTS

3.1 Functional Requirements

1. **Admissions and Registration Module:**
   (a) Allow prospective students to submit online applications.
   (b) Enable the registrar to review and process applications.
   (c) Facilitate enrollment and registration processes.
2. **Course Management Module:**
   (a) Provide tools for curriculum planning and course scheduling
   (b) Approve student course registrations
   (c) Generate timetables for classes, exams, and other academic events.
   (d) Allocate resources such as rooms, equipment, and faculty assignments.
3. **Examination Management Module:**
   (a) Enable officials to schedule exams and plan seating arrangements.
   (b) Facilitate grade generation and publication.

3.2 External Interface Requirements

Following user interfaces have been planned for now. More interfaces may be added later as per requirements -

- Login Page
- Student/Faculty Registration page
- Student Home page : For viewing timetable, exam seating plan, Course list, grades
- Faculty Home page : For uploading courses,student approvals, grades, timetable view
- Admin Home page : For viewing any database, student/faculty approval, exam schedule, timetable, permissions

3.3 System Features

1. **Authentication and Authorization:**
   (a) Secure login mechanisms for user authentication.
   (b) Role-based access control to restrict access to specific functionalities based on user roles.
2. **Data Management:**
   (a) Database management system for storing and managing academic data.

3.4 Nonfunctional Requirements

- **Security**: Implementation of data encryption mechanisms to ensure data security and privacy. Protection against unauthorized access.
- **Usability** : Intuitive user interfaces with clear navigation and consistent design.
- **Maintainability** : Systematic handling of data, and well documented code/software.
- **Reliability**: Fault tolerance to handle system failures and recover gracefully.

4 SYSTEM DESIGN AND VISUALIZATION

4.1 Data Dictionary

The database for the solution will contain the following entities in addition to possibly a few others :

Entities	**Data Description**	**Table Description**	**Usage**	**Security Permissions**		
				Student	**Faculty**	**Admin**
Sign-Up	Holds the details entered by a user when first registering on the system	Personal details, and User Type, hash of the account password, verification details	The admin can approve pending registration requests.	None	None	View, Edit, Delete
Auth	Holds the Login details for the user. Password hash is stored and matched with the hash of the entered password for secure login.	UserID, Roles (Privileges), and Password Hash	Used for login/authentication of user	None	None	Edit "Role/Privileges" only. User can change only his own password.
Student	Holds student relevant data	StudentID(derived from userID), Personal details	Used to prepare a student profile and map with other courses and relevant administrative activities.	View, Edit	View	View, Delete
Faculty	Holds faculty relevant data	FacultyID(derived from userID), Personal details.	Used to prepare a Faculty profile and map with other courses and relevant administrative activities.	View	View, Edit	View, Delete
Course	Holds course related data entered when floated by the instructor.	Course name, curriculum, credits, lecture/tut/lab hours, intake semester(like 2024 Winter - sem 6), pending approvals	For students to view course details and also register.	View	View, Add, Edit, Delete	Add, Edit, Delete
Grade	Contains students' grades obtained in particular courses	StudentID(UserID), CourseID, Grade (Letter grades AS,AA,AB...)	To declare the grades awarded by the faculty to the students	View, Edit	View	Delete
Classroom	Holds information related to classroom logistics	Table with RoomID, Capacity and type(lab/theory)	For generation of proper timetable and exam seating plan	View	View	Edit, Delete
Exam	Holds examination arrangement for all courses and students	Table with Student list, RoomID, Date-time, CourseID	For Storing the examination schedule and seating arrangement.	View	View	Edit, Delete
TimeTable	Holds the time Table for the academic semester	CourseID, RoomID, weekly time slots, etc.	To refer to for class schedule	View	View	View

4.2 Data Flow Diagrams

4.2.1 Level 0 DFD

It portrays the entirety of the academic management system as a single entity, offering an overarching view of the system.

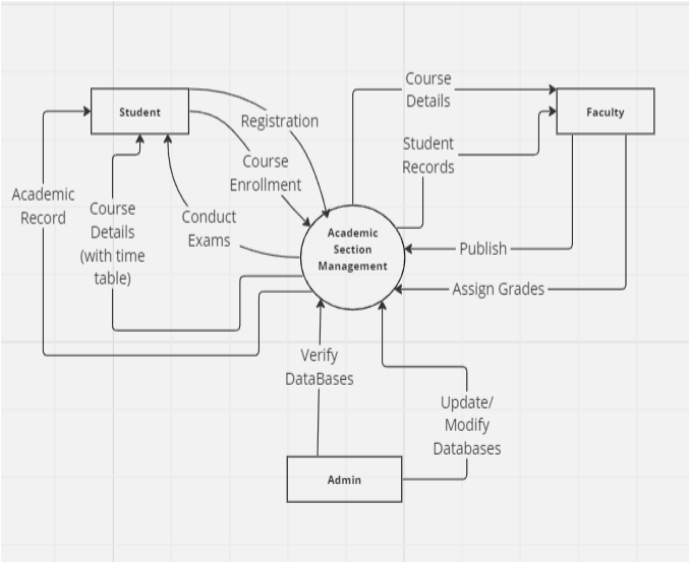


Figure 1: Data Flow Diagram (Abstract View of System)

4.2.2 Level 1 DFD

It delineates the principal functions of the system, namely student enrollment, course management, and exam conduction, along with their interactions with relevant entities.

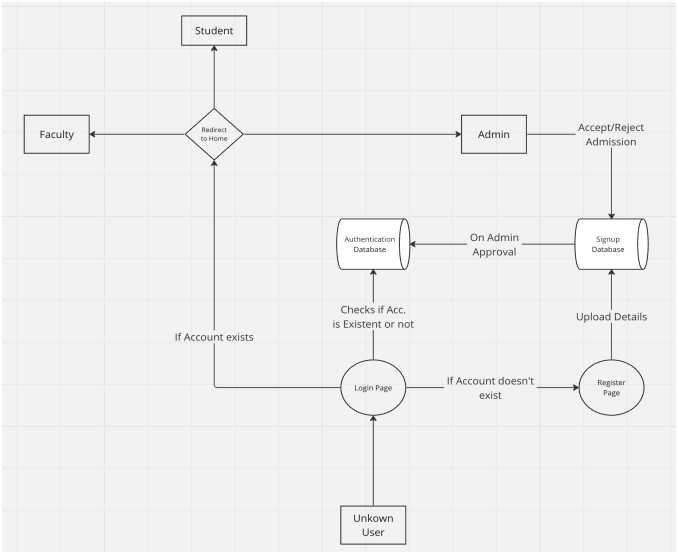


Figure 2: Login/Signup

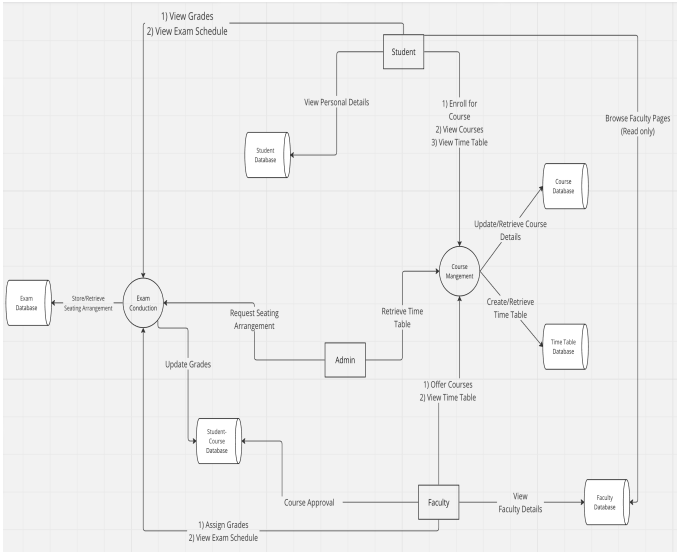


Figure 3: Post-login workflow

4.2.3 Level 2 DFD

It illustrates the data flow within each process and the manner in which data is transformed and stored. Processes such as student enrollment, course enrollment, timetable generation, grade generation, and exam conduction are depicted at this level in each diagram.

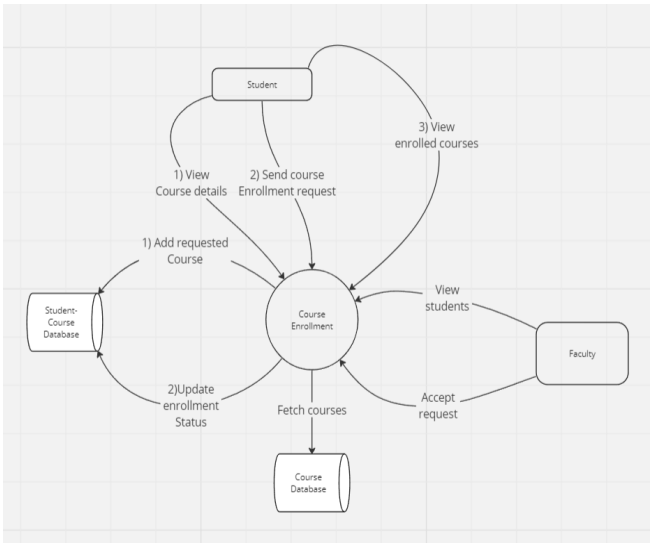


Figure 4: Course Enrollment

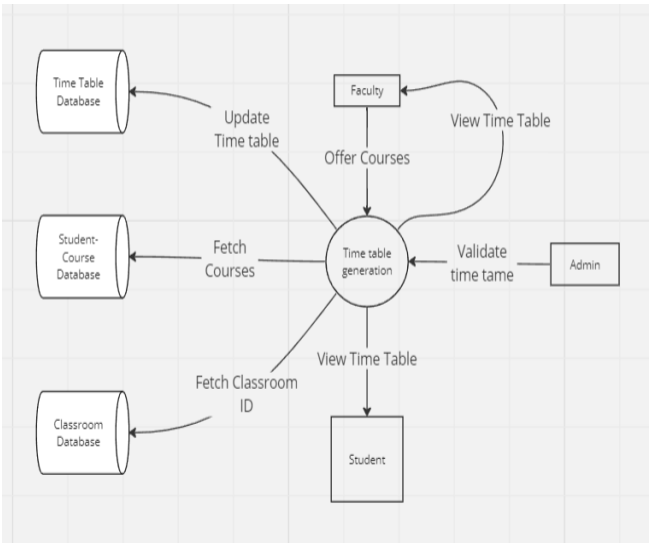


Figure 5: Generation of Timetable

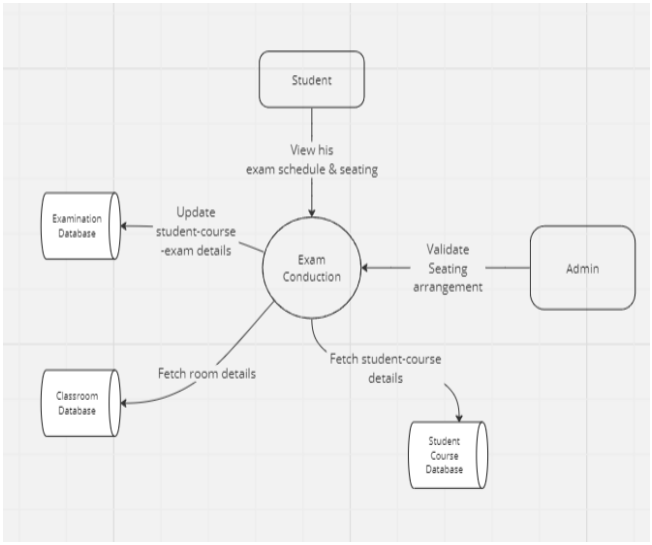


Figure 6: Conduction of Exams

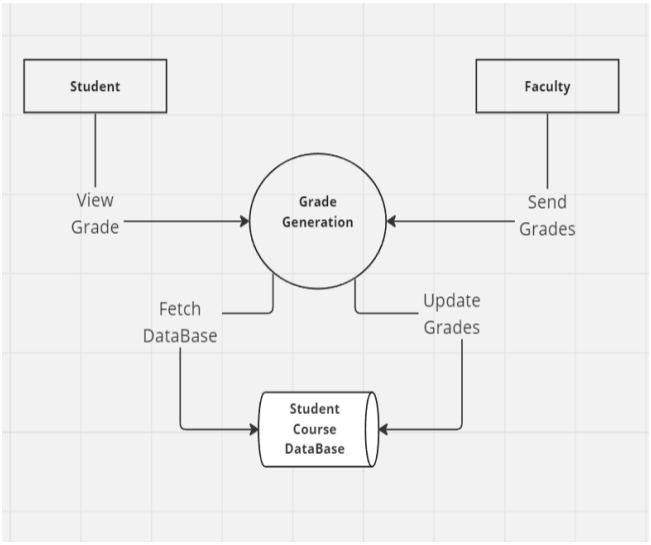


Figure 7: Generation of Grades

4.3 Entity-Relationship Diagram

The database can be represented using the notations, and these notations can be reduced to a collection of tables.In the database, every entity set or relationship set can be represented in tabular form. In the academic section management E-R diagram, students, faculty, courses, timetable,exams,etc are interconnected entities representing the core components of educational operations. Students enroll in courses, with each course being taught by a faculty member, forming a relationship between student, faculty, and course entities. Courses are organized into timetables with classrooms alloted, detailing their schedules,exams are associated with courses,grades are given to students by the faculty,etc . This allows for efficient management of academic activities, ensuring smooth coordination between students, faculty, course management,and exam management.

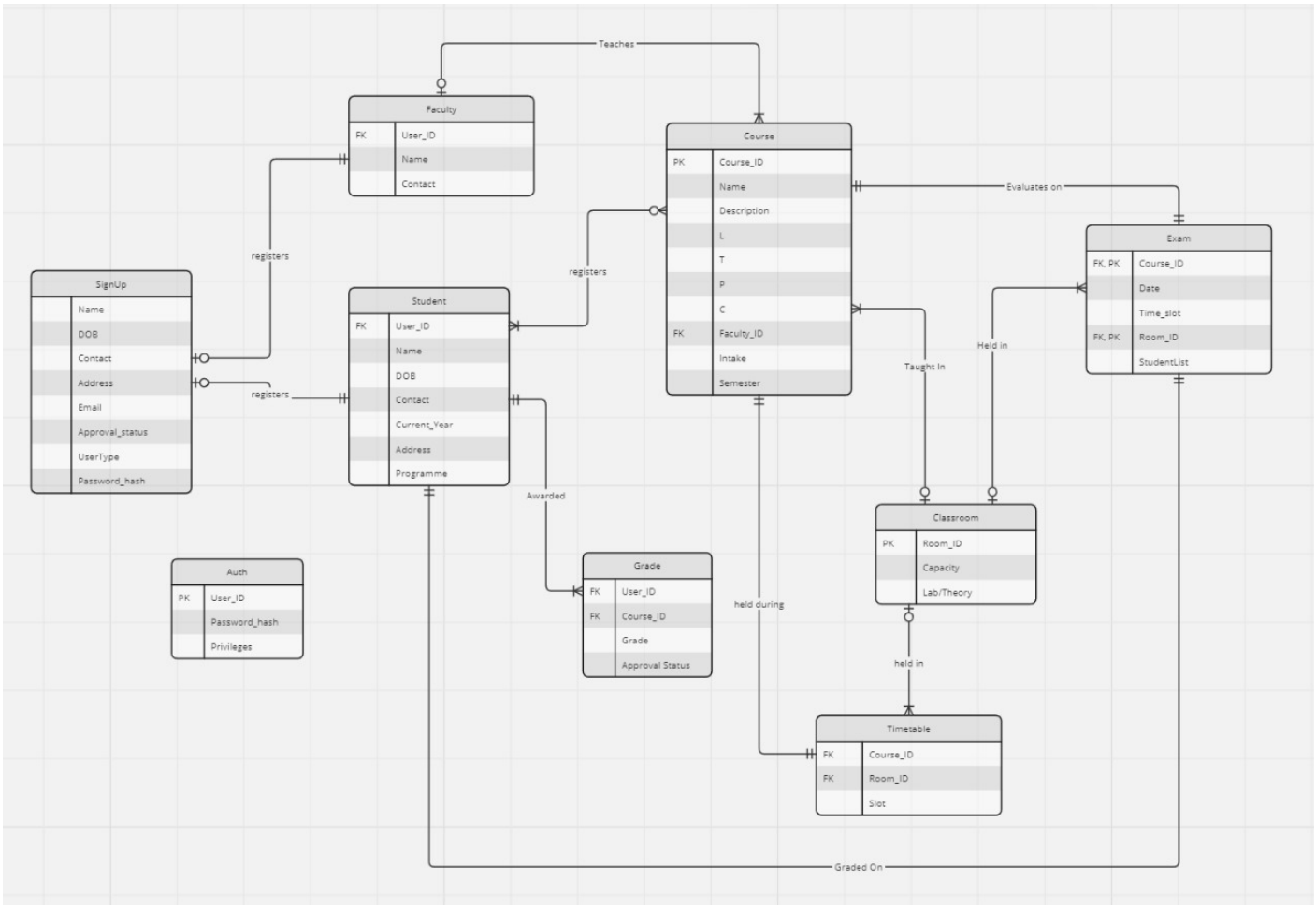


Figure 8: ER Diagram

5 SOLUTION DESCRIPTION

The proposed solution has been explained/demonstrated in the above DFDs and the ER diagram. The Data dictionary describes the databases in the ER diagram and highlights the permissions of the users on these databases.
While most of the solution is described in the DFDs, the following section will describe a possible solution for the timetable generation and examination seating plan generation.

5.1 Time table generation

Crafting an efficient timetable requires balancing various aspects like room availability and teacher/student preferences while keeping in mind the fair distribution of their time spent teaching/studying respectively. The task becomes exponentially difficult as the number of courses, faculties, and students increases. As a consequence of this, we have come up with a possible solution for the Time Table generation problem with a few assumptions (subject to change) as follows:

1. There exists only one branch in the Institute namely Computer Science and Engineering having students across 4 years in the BTech Degree.
2. 1st and 3rd yearites have classes in the evening and labs in the morning.
3. 2nd and 4th yearites have classes in the morning and labs in the evening.
4. Each year, at least 5 theory and 3 lab courses.
5. A professor can take at max 1 course for a particular batch.

To efficiently organize the scheduling of courses for our academic program, we have devised a structured approach. The allocation of time slots will be categorized into two distinct periods: morning (A, B, C, D, E) and evening (A1, B1, C1,

D1, E1). Specifically, courses for first and third-year students will be scheduled during the evening slots, while courses for second and fourth-year students will be assigned to morning slots. To facilitate parallel classes for first and third-year students, they will be scheduled in separate rooms, and the same arrangement applies to morning classes for second and fourth-year students.

	Mon	Tues	Wed	Thurs	Fri
9-10 AM	A	B	C	D	E
10-11 AM	E	A	B	C	D
11-12 AM	D	E	A	B	C
2-3 PM	A1	B1	C1	D1	E1
3-4 PM	E1	A1	B1	C1	D1
4-5 PM	D1	E1	A1	B1	C1

Our scheduling algorithm will map these time slots to corresponding theory courses for each academic year. It is imperative to adhere to the constraint that a faculty member cannot teach two consecutive classes. For example, if a faculty member instructs a course for both the first and third years, and the time slot assigned for the first-year course is, say, slot A, then the scheduler must ensure that slot E is not allocated to the third-year course.

Regarding laboratory sessions, these will be scheduled for all academic years without conflicts, and rooms will be allocated accordingly. To provide a clearer framework, the weekly schedules for each academic year are as follows:

1. First-year - Mon, Tues, Wed - 9-12 AM
2. Second-year - Mon, Tues, Wed - 2-5 PM
3. Third-year - Wed, Thurs, Fri - 9-12 AM
4. Fourth-year - Mon, Thurs, Fri - 2-5 PM

5.2 Examination Seating plan generation

The proposed solution for examination scheduling involves a structured approach to allocate fixed time slots for examinations over 5 days. Each examination day will consist of two slots: one in the morning (9:00 AM - 12:00 PM) and another in the evening (2:00 PM - 5:00 PM). To streamline the process, a distinction is made between the examination timings for first and third-year students, who will be scheduled in the evening, and the remaining years, who will have their examinations in the morning.

Given that each academic year has a maximum of 5 theory courses, a systematic mapping is established to associate each course with a specific examination day. This allocation ensures that the examination schedule is evenly distributed across the designated days. For each examination day and slot, a fair distribution of students for a particular course is achieved by dividing them equally among the available examination rooms. This approach aims to maintain a balanced and organized examination environment, optimizing the utilization of resources and providing an equitable experience for all students undergoing assessments.

5.3 Grade generation

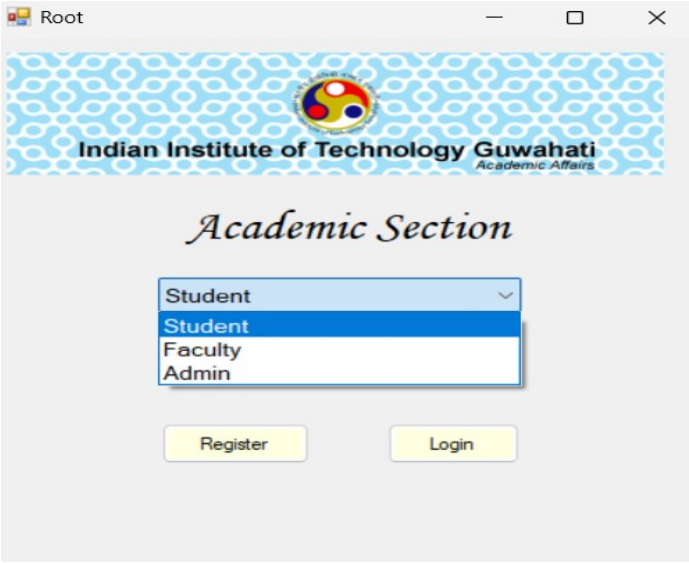
Faculty members are responsible for updating student grades in the respective courses. The system then calculates the Semester Performance Index (SPI) and Cumulative Performance Index (CPI) by employing the number of credits as weights in a weighted average. This approach ensures a comprehensive evaluation, factoring in course-specific credit values to determine the student's academic performance in each semester and overall cumulative achievement.

6 BASIC INTERFACE DESIGN

The following visualizations are merely representative of the final software product

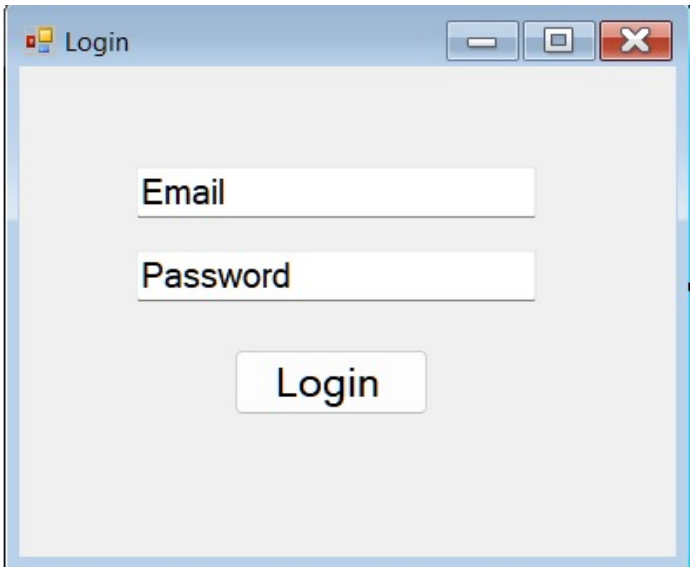
6.1 Root Page

This page redirects all users (student/faculty/admin) to appropriate login/registration page



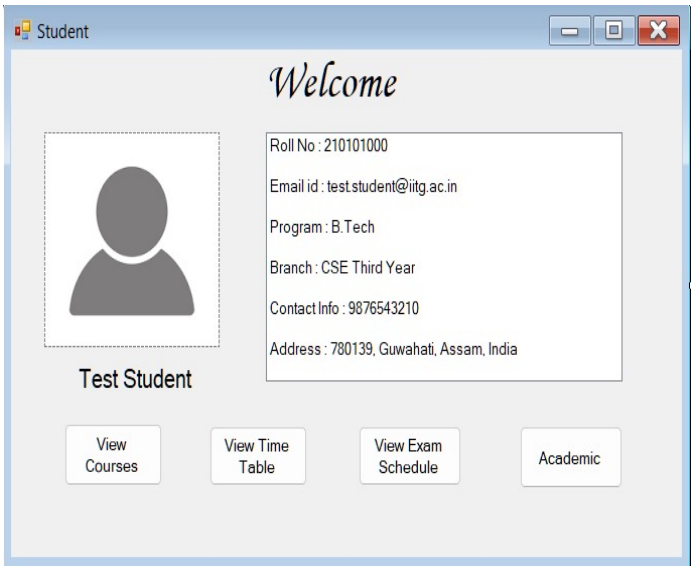
6.2 User Login Page

This page redirects all users (student/faculty/admin) to appropriate login/registration page



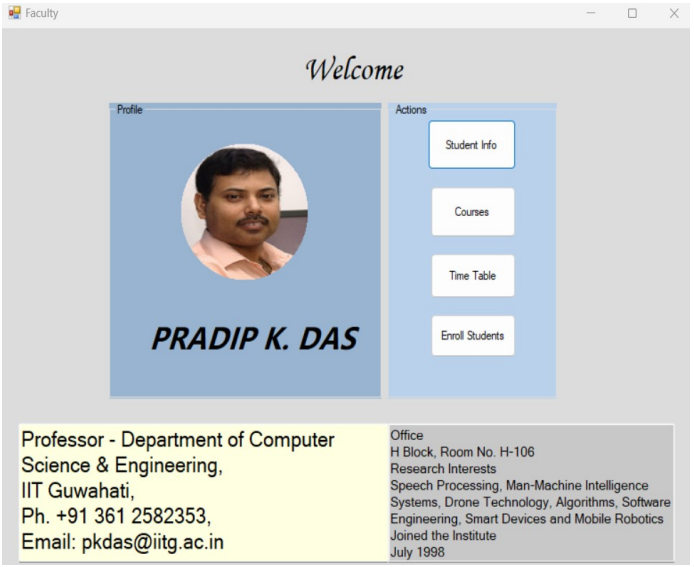
6.3 Student Home Page

Allows the student to view his profile, time table and exam schedule. Also enables him to access other academic information such as grades etc.



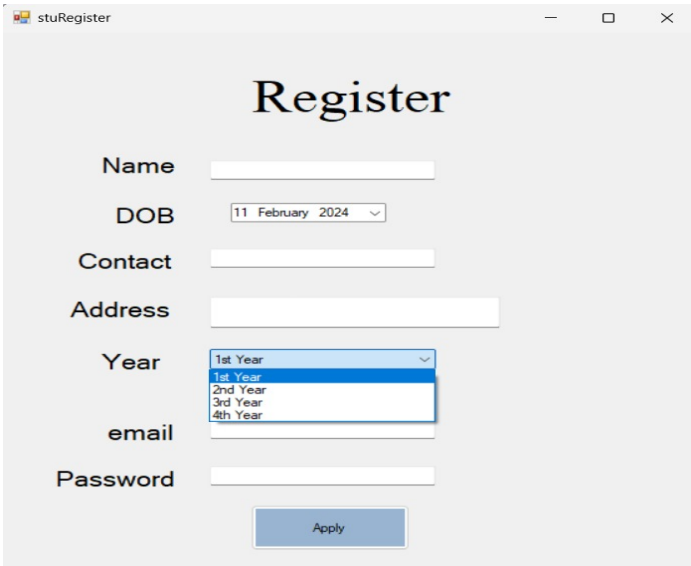
6.4 Faculty Home Page

Faculty can retrieve all relevant information and perform necessary actions from their home page (for ex: can accept course enrollment requests from students)



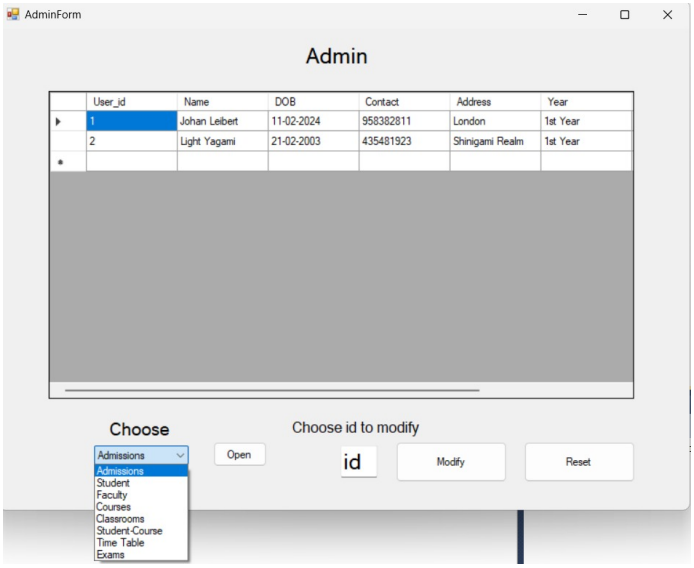
6.5 Student Registration Page

The student must fill in appropriate details during registration.



6.6 Admin Page

Admin can select the required database from the dropdown and then accordingly view and modify entries in it.



7 CONCLUSION

Our current challenge is the academic management system, a broad aspect that we've divided into various modules and are approaching each one broadly. We have established databases wherever needed,new databases can be added, and relations can be established. Database views will be added appropriately. We're working on solutions, including efficient and user-friendly time table generation and exam seating arrangements for both students and faculty. We've chosen to implement our workflow using the spiral model.

This project requires collaborative work, with all members contributing. Our solution is going to be productive and practical.Some extra assumptions have been taken in this initial solution which can be removed as we move to the coding part.The outline presented in the report is subject to change as per requirements or coding and time constraints.

Looking ahead, we are confident that our efforts will result in a successful implementation that enhances the academic experience for all stakeholders.