Practical 5

System Analysis and Design for the Selected System

Aim: Perform system analysis on selected system. 1)Systems analysis (what the system should do) 2) Systems design (how to accomplish the objective of the system.)

(Hint: Flowcharts/ER diagrams)

5. System Analysis and Design

The College Management System (CMS) is a software solution designed to facilitate the management of a college by automating various operations, including student enrollment, faculty management, course registration, examination scheduling, and result processing. The system improves efficiency, reduces manual errors, and enhances the overall user experience.

5.1 System Analysis

The College Management System aims to achieve the following objectives:

- **Efficient Student and Faculty Management**: Maintain an organized record of students and faculty, ensuring easy access and tracking.
- **Automated Course Registration**: Reduce manual intervention in course enrollment and schedule management, ensuring real-time updates.
- **User Authentication and Role-Based Access**: Provide a secure access mechanism for different users (students, faculty, administrators) with appropriate privileges.
- **Examination and Result Processing**: Automate the scheduling, conduction, and evaluation of examinations, ensuring transparency in result generation.
- **Reporting and Analytics**: Generate detailed reports on student performance, course enrollments, and faculty activities, aiding in better decision-making.
- **Scalability**: Support an increasing number of students, faculty, and courses over time, ensuring long-term sustainability and growth.

5.1.1 Gather System Requirements

Functional Requirements:

- **User Management Module**: Administrators can create, update, and delete user accounts. Users have different roles such as Admin, Faculty, and Student, each with distinct privileges.
- **Student Management Module**: Enables administrators to enroll students, update details, and track academic progress.
- **Faculty Management Module**: Allows faculty members to manage assigned courses, update personal information, and access student records.
- **Course Management Module**: Facilitates adding, updating, and removing course records, ensuring proper classification based on department and credits.

- **Course Registration Module**: Allows students to enroll in courses based on prerequisites and availability.
- **Examination Management Module**: Automates exam scheduling, hall allocation, and student attendance tracking.
- **Result Processing Module**: Handles the evaluation of student performance and generation of results, ensuring transparency.
- **Report Generation Module**: Produces reports on student performance, faculty workload, and course enrollments for administrative decision-making.
- **Security and Authentication**: Implements encryption-based secure login and session-based access control.

Non-Functional Requirements:

- **Performance**: Ensures fast response times for course registration, exam scheduling, and result processing.
- **Scalability**: Accommodates growing numbers of students, faculty, and courses efficiently.
- **User-Friendly Interface**: Provides an intuitive and easy-to-use UI for seamless access.
- **Security**: Implements encrypted storage of sensitive data and role-based access controls.
- **Maintainability**: Supports easy updates and feature enhancements.
- **Reliability**: Ensures uninterrupted operation without failures.

5.1.2 Analyze the Current System

Existing Issues

- Manual student enrolment and faculty management result in errors and inefficiencies.
- Course registration and scheduling are time-consuming and prone to conflicts.
- Examination scheduling lacks automation, leading to mismanagement.
- The absence of real-time result processing causes delays.
- Lack of automated reporting makes performance tracking difficult.

Gaps Identified

- A centralized database is required for effective record management.
- Course registration and scheduling should be automated.
- Secure authentication and access control mechanisms must be implemented.
- A reporting system is essential for performance tracking and decision-making.

5.2 System Design

System Design defines the architecture, modules, interfaces, and data structure of the College Management System. It ensures efficiency, scalability, and security while transforming functional requirements into a structured solution.

5.2.1 Architectural Design

The College Management System follows a three-tier architecture:

1. **Presentation Layer:** A web-based user interface for user interactions.
2. **Business Logic Layer:** Handles core functionalities such as student enrollment, course management, and result processing.
3. **Database Layer**: Stores essential records, including student details, faculty information, course data, and examination results.

5.2.2 User Interface Design

The UI is designed to be intuitive and user-friendly, with dashboards for different roles:

- **Login Page:** Secure login with role-based authentication.
- **Admin Dashboard:** Displays system statistics, student and faculty management options, and report generation.
- **Faculty Dashboard:** Includes options for course management, examination scheduling, and result entry.
- **Student Dashboard:** Allows course registration, exam schedule viewing, and result tracking.
- **Course Registration Interface:** Enables students to enroll in available courses.
- **Examination Management Screen:** Displays exam schedules and student attendance records.
- **Result Processing Screen:** Provides examination results and performance analysis.
- **Notification System:** Sends alerts for enrolment deadlines, exam schedules, and results.

5.2.3 Module Description

- **User Module:** Manages user registration, authentication, and role-based access control.
- **Student Module:** Maintains student records, course enrolments, and academic history.
- **Faculty Module:** Handles faculty information, assigned courses, and examination responsibilities.
- **Course Module:** Facilitates course creation, updates, and categorization.
- **Course Registration Module:** Allows students to enroll in courses and ensures prerequisites are met.
- **Examination Module:** Schedules exams, assigns exam halls, and tracks attendance.
- **Result Processing Module:** Evaluates student performance and generates results.
- **Report Generation Module:** Provides analytical insights into student performance, course enrolments, and faculty workload.

5.2.4 Database Design and ER diagram

The database design ensures efficient storage and management of student, faculty, course, and examination data. It follows a relational model with multiple interconnected tables representing different entities and relationships.

Entities and Relationships:

1. **Student (Student_ID, Name, Email, Enrollment_No, Department) –** A student enrolls in multiple courses.
2. **Faculty (Faculty_ID, Name, Email, Department, Designation) –** A faculty member teaches multiple courses.
3. **Course (Course_ID, Course_Name, Credits, Department) –** A course is taught by multiple faculty members and taken by multiple students.
4. **Enrollment (Enrollment_ID, Student_ID, Course_ID, Semester) –** Links students and courses, tracking enrollments.
5. **Examination (Exam_ID, Course_ID, Faculty_ID, Exam_Date, Exam_Type, Total_Marks) –** A course has multiple examinations.
6. **Result (Result_ID, Student_ID, Exam_ID, Score, Grade) –** Tracks student performance in exams.

The ER diagram visually represents the relationships among students, faculty, courses, and examinations, facilitating structured data management and retrieval.

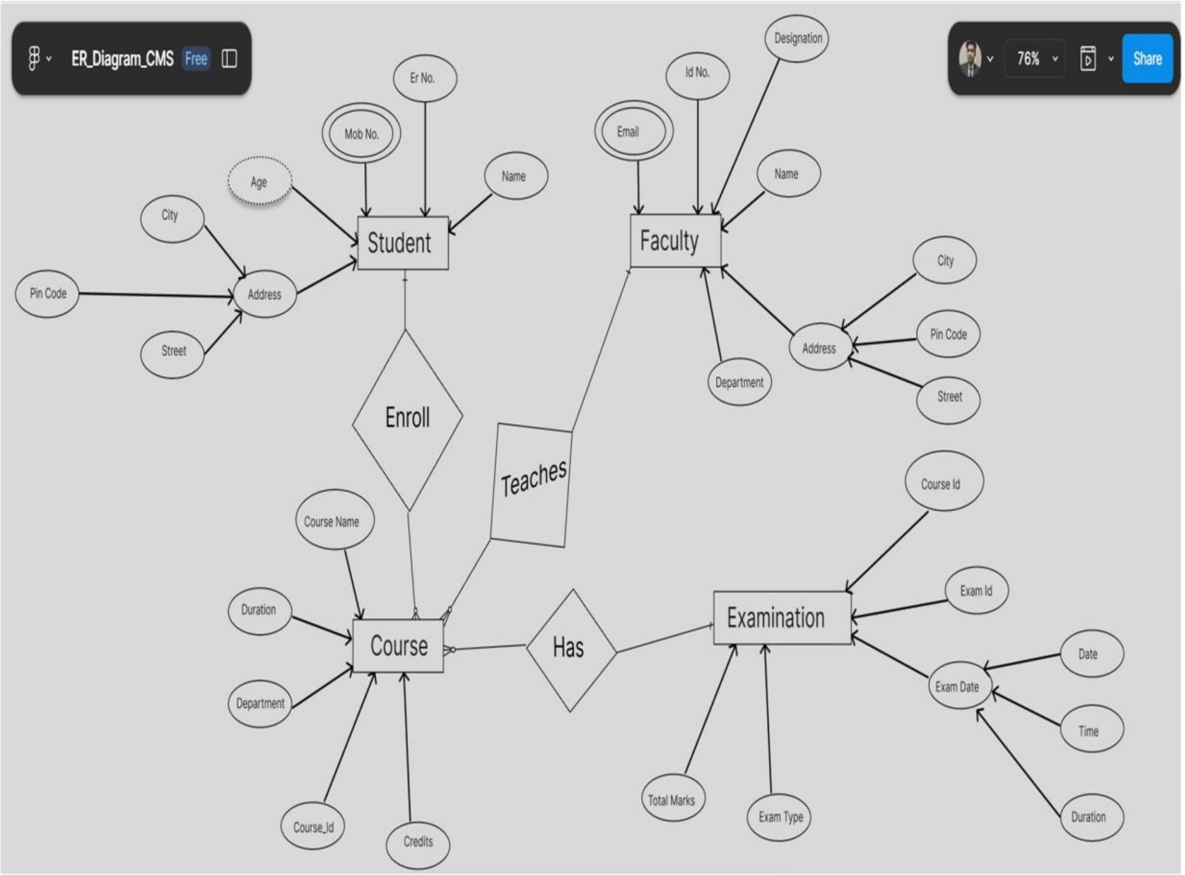

Figure 5.1 ER Diagram