



DATABASE PROJECT FLEX ACADEMIC SUITE



SHAMEER ABDULLAH | MARYAM MOEED | FATIMA AAMIR
I20-0540 | I20-2470 | I20-0654

Introduction

The FLEX system is a database management system designed for FAST NUCES to manage and schedule university students' academic information. The system aims to improve student management, enrollment, course registration, scheduling, and performance tracking. It provides user-friendly interfaces for different roles such as Academic Officers, Faculty, and Students. This document highlights the design of the system, including entity relationships, ERD, schema diagram, and mappings of the relationships.

Entity Relationship Diagram (ERD)

The ERD illustrates the relationships between different entities in the FLEX system. It identifies the entities, their attributes, and the relationships between them. The following entities are part of the system:

- **User:** Represents a user in the system with attributes like username, password, and role.
- **Academic Officer:** Represents an academic office user with additional attributes.
- **Faculty:** Represents a faculty member with attributes like name, designation, and contact information.
- **Student:** Represents a student with attributes like name, registration number, and contact information.
- **Course:** Represents a course offered by the university with attributes like course code, title, and credit hours.
- **Section:** Represents a specific section of a course with attributes like section number, capacity, and assigned faculty.
- **Evaluation:** Represents an evaluation component of a course with attributes like weightage and criteria.
- **Attendance:** Represents the attendance of a student in a course section.
- **Grade:** Represents the grades assigned to students in a course.

Schema

The schema diagram provides a visual representation of the database schema for the FLEX system. It shows the tables, their attributes, and the relationships between them. The schema includes tables such as Users, Academic Officers, Faculty, Students, Courses, Sections,

Evaluations, Attendance, and Grades. Each table has its primary key and foreign key relationships based on the ERD.

For example

```
CREATE TABLE Teachers (  
    TeacherID INT PRIMARY KEY,  
    Name VARCHAR(50)  
);  
  
CREATE TABLE Courses (  
    CourseID INT PRIMARY KEY,  
    CourseName VARCHAR(50),  
    TeacherID INT,  
    FOREIGN KEY (TeacherID) REFERENCES Teachers(TeacherID)  
);  
  
create table marks(  
    evaluation_name varchar(30), weightage numeric  
);
```

Mappings of Relationships

The relationships between entities in the ERD are mapped to the database schema using foreign key constraints. For example:

- *The Users table contains user information, and the Role attribute determines the role of the user (Academic Officer, Faculty, or Student).*
- *The AcademicOfficers table references the Users table using a foreign key constraint to establish the relationship between users and academic officers.*
- *The Faculty table references the Users table to associate faculty members with their user accounts.*
- *The Students table references the Users table to associate students with their user accounts.*
- *The Courses table contains information about the courses offered, and the Section table references the Courses table to assign sections to specific courses.*
- *The Sections table references the Faculty table to assign a faculty member as the instructor for a section.*

- *The Evaluations table references the Courses table to associate evaluation components with specific courses.*
- *The Attendance table references the Students and Sections tables to record attendance for students in specific course sections.*
- *The Grades table references the Students and Courses tables to assign grades to students in specific courses.*

Audit Trail

To implement an audit trail, triggers are used to capture and log information about the operations performed in the database. Whenever an operation like insertion, update, or deletion occurs on specific tables, the trigger will record the operation details, user performing the operation, and the timestamp. This information is stored in an audit trail table for later analysis and tracking of user actions.

By implementing an audit trail, the FLEX system can maintain a history of user operations, ensuring accountability, security, and compliance with data management policies. It provides a way to track changes, troubleshoot issues, and investigate any unauthorized activities.