Database Systems Assignment 2

U00839259 (syrrmlla)

1. Users can be Students, Instructor, Registrations Office, ISS Advisor, Admin Office.

Students: These are main users who have many applications as mentioned below.

- Students can register Courses. (Form based Interface)
- Students can view Grades on portal.
- Students can pay to registered Courses.
- Student can check their Pre-requisite courses.
- They can choose or change their Major.

Registrations Office: (Form based/Parametric Interface)

- Make Courses available for Courses.
- Add new courses to Department or change Course details
- Add or remove Pre-requisite requirements to a Course.
- Modify Pre-requisites for a Student.
- Modify credit-hours for Courses.
- Add Student to a Course.

Instructor: (Form based/Parametric Interface)

- Post Grades for Students.
- Make changes to Grades.
- Provide permit for Courses to Students.
- Close Course registrations.

Admissions Office: (Form based/Parametric Interface)

- Admin can give admit for Students.
- Admin can allot a student number to Students.
- Admission office allows students to change their Major.
- Admission office can modify Student details.

ISS Advisor: (Parametric Interface)

- Can collect details of International Students from Student table.

2. The best DBMS Architecture to design a Web-based Airline reservation system is **Three tier Client/Server architecture,** because
    - The Web user interface can be placed in the client system.
    - The database server contains the DBMS.
    - Web/application server handle the application logic of the system.

Other architectures which are not a good choice:

- In a centralized DBMS architecture, DBMS functionality and user interface are both performed on the same system hence not appropriate for web-based system.
- In two-tier Client/Server architecture both the logic and GUI have to live on the same machine, there is a heavy load put on the server which is why the basic client/server architecture, and the two-tier client/server architecture aren't the best choices.

3. **STUDENT Table:** Student_number is the Unique column. The constraints are
    - Student number should be unique to avoid overlapping if any students have same names in same section.

COURSE Table: Cousre_number is the Unique column. The Constraints are

- Sometimes the course names might be same for graduate and PG students in that case the course number is required to distinguish between the courses.
- When a new course is added to the catalog, it must be assigned a unique number to distinguish it from the existing ones.

PREREQUISITE: Prerequisite_number is unique column. The Constraints are

- Prerequisites are unique in that they vary depending on the course in the section table.
- Few courses have prerequisites, while others do not, so it is critical to make sure that Prerequisite number is unique.

SECTION: Section_Identifier is unique column. The constraints are

- Sections offered in a given semester must be distinct in order to avoid class overlap.
- It should be unique because sometime professor may add extra section based on the number of students enrolled.

GRADE_REPORT: Student_number and Section_Identifier both should be unique, The Constraints are

- Sometimes student names may overlap.

- As stated above, Section Identified is a unique number that varies depending on the semester and year the course is offered**.**

4. **Database Schema:** A database schema is a database's design. The database description represents the structure (table names and fields/columns), the type of data that each column can hold, the constraints on the data that can be stored, and the relationships between the tables.
   Schema Construct is a schema component or an object contained within the schema.

   Database State: Database Situation Is the actual data stored in a database at a specific point in time. This includes gathering all of the data in the database. This is also known as a database instance, and the term instance can refer to individual database components. Then, using queries, we can retrieve data or manipulate it by updating, modifying, or deleting it. As a result, the database's state can change, and a database schema can have multiple instances at the same time.