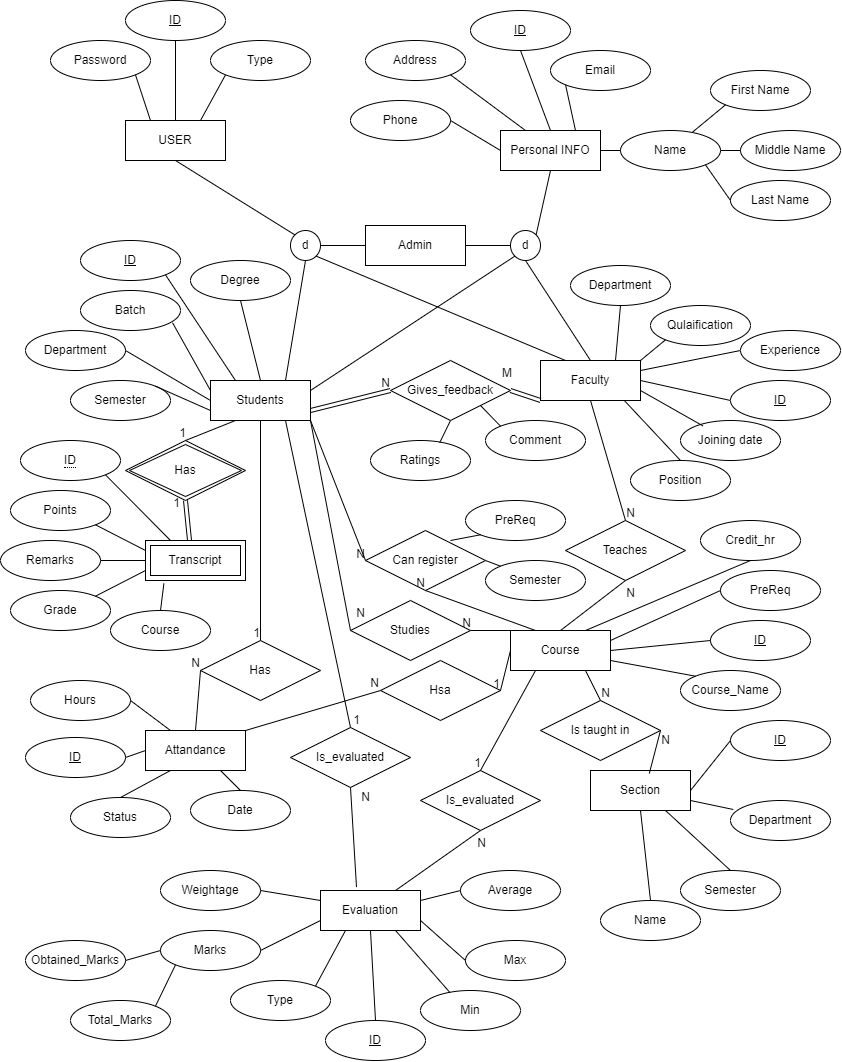
**PROJECT**

### Muhammad Humayun Malik (21I-2562)

### Zian Ahmed (21I-0503)

### Mati-ur-Rehman (21I-0872)

# Flex System



## **User**

### **Attributes**

* ID
* Password
* Type

Primary key : ID

Super class for admin, faculty and students. Data in this table is used for login.

## **Personal Info**

### **Attributes**

* ID
* Address
* Phone
* Email
* First Name
* Middle Name
* Last Name

Primary key : ID

Foreign Key(id) references User(id)

1 -> 1 relationship

A Table containing All the Basic information of a Person. Contains Data that is common in Student, Teacher and Admin.Also a super class for admin, faculty and student

## **Faculty**

### **Attributes**

* ID
* Experience
* Joining Date
* Position
* Qualification
* Department

Primary key : ID

Foreign Key : ID

This Entity Provides the specific information of teachers, their Professional information.

## 

## **Student**

### **Attributes**

* ID
* Degree
* Batch
* Department
* Semester

Primary key : ID

Foreign Key : ID

This Entity Provides the specific information of teachers, their University credentials..

## **Course**

### **Attributes**

* ID
* Name
* PreReq
* Credits

Primary key : ID

Foreign Key : ID

This Table provides Information about the courses the university has to offer.

## **Section**

### **Attributes**

* ID
* Name
* Department
* Semester

Primary key : ID

Foreign Key(Id) references Taught in(Section)

1 -> n relationship with Taught in

This table stores information about different sections in which the courses are being taught in,

## **Evaluation**

### **Attributes**

* ID
* Obtained Marks
* Total Marks
* Average
* Min
* Max
* Type
* Weightage

Primary key : ID

Foreign Key(Student) references Student(id)

Foreign Key(Course) references Course(id)

1 -> n relationship with Student

1 -> n relationship with Course

This table records the academic progress of a student in different courses. Their marks in different quizzes, assignments, projects and exams. This also keeps track of the weightages which help in generating grades.

## **Attendance**

### **Attributes**

* ID
* Date
* Status
* Hours

Primary key : ID

Foreign Key(Student) references Student(id)

Foreign Key(Course) references Course(id)

1 -> n relationship with Student

1 -> n relationship with Course

This table records the attendance of a student in different courses.

## **Teaches**

### **Attributes**

* Teacher
* Course

Foreign Key(Teacher) references Faculty(id)

Foreign Key(Course) references Course(id)

1 -> n relationship with Faculty

1 -> n relationship with Course

As the relation between course and teacher is many to many this table keeps record of different teachers teaching different courses.

## **Studies**

### **Attributes**

* Student
* Course

Foreign Key(Student) references Student(id)

Foreign Key(Course) references Course(id)

1 -> n relationship with Student

1 -> n relationship with Course

As the relation between student and course is many to many this table keeps record of different students studying different courses.

## **Taught in**

### **Attributes**

* Section
* Course

Foreign Key(Section) references Section(id)

Foreign Key(Course) references Course(id)

1 -> n relationship with Section

1 -> n relationship with Course

This table will keep a record of courses taught in different sections.

## **Register**

### **Attributes**

* ID
* Student
* Course
* Section
* PreReq
* Semester

Primary key : ID

Foreign Key(Student) references Student(id)

Foreign Key(Course) references Course(id)

1 -> n relationship with Student

1 -> n relationship with Course

The registration table will have the data that will help in registration of different courses. The constraints whether a particular student will be able to register a particular course will be present in this table.

## **FeedBack**

### **Attributes**

* ID
* Student
* Teacher
* Ratings
* Comments

Primary key : ID

Foreign Key(Student) references Student(id)

Foreign Key(Teacher) references Faculty(id)

1 -> n relationship with Student

1 -> n relationship with Faculty

This table will store the feedback provided by a student for a particular teacher . it will store the ratings and comments.

## **Transcript**

### **Attributes**

* ID
* Points
* Remarks
* Grade
* Course

Primary key : ID

Foreign Key(id) references Student(id)

1 -> 1 relationship with Student

The transcript table will store the necessary data for making a transcript of a student. It is a weak entity as it cant exist without students.