![A blue stripe on a white background

AI-generated content may be incorrect.](data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAOEAAADhCAMAAAAJbSJIAAAAElBMVEX///8Aqv/0+/8Ao/8AqP9Iuf+HVr03AAAAnElEQVR4nO3cQRGEAAzAwHKAf8vngQ+TsusgBjIDAAAAAAAAAAAAAAAAAAAAsNBvuzm3m2M7hX0K+xT2KexT2KewT2Gfwj6FfQr7FPYp7FPYp7BPYZ/CPoV9CvsU9insU9insE9hn8I+hX0K+xT2KexT2KewT2Gfwj6FfQr7FPZ9oPDabu7t3t5yAAAAAAAAAAAAAAAAAAAAAPDEHycxS5flcq0rAAAAAElFTkSuQmCC) **University Flex System**

**DELIVERABLE IV**

**GROUP MEMBERS:**

M. HARIS FAYYAZ 23L-0767

MOEEZ IJAZ 23L-0750

ALI YOUSAF 23L-0737

**SECTION:** BCS-4C

**SUBMITTED TO:** SIR MUHAMMAD KAMRAN

**COURSE:** DATABASE MANAGEMENT SYSTEMS LAB

Blue logo with a white background

AI-generated content may be incorrect.

**1. Initial Database Diagram (Deliverable I)**

**Tables:**

* Users
* Departments
* Faculty
* Students
* Courses
* CourseEnrollment
* FacultyCourses
* Attendance
* Grades
* Announcements

A computer screen shot of a computer

AI-generated content may be incorrect.

**2. Identified Functional Dependencies (FDs) and Issues**

**Users**

* UserID → FullName, Email, Password, Role, CreatedAt

**Departments**

* DepartmentID → DepartmentName

**Faculty**

* FacultyID → UserID, DepartmentID
* UserID → FacultyID

**Students**

* StudentID → UserID, DepartmentID, EnrollmentYear
* UserID → StudentID

**Courses**

* CourseID → CourseName, CourseCode, DepartmentID, Credits

**CourseEnrollment**

* EnrollmentID → StudentID, CourseID, EnrollmentDate, Grade
* (StudentID, CourseID) → EnrollmentDate, Grade

**FacultyCourses**

* AssignmentID → FacultyID, CourseID, Semester
* (FacultyID, CourseID, Semester) → AssignmentID

**Attendance**

* AttendanceID → StudentID, CourseID, AttendanceDate, Status
* (StudentID, CourseID, AttendanceDate) → Status

**Grades**

* GradeID → StudentID, CourseID, Grade
* (StudentID, CourseID) → Grade

**Announcements**

* AnnouncementID → PostedBy, DepartmentID, Message, CreatedAt

**Identified Issues:**

* No partial dependencies.
* No transitive dependencies.
* Foreign keys correctly enforce referential integrity.
* Minor duplication prevented through unique constraints (example: FacultyCourse assignments, Student enrollments).

**Conclusion:**

* Already highly normalized.

**3. Step-by-Step Normalization Process**

**First Normal Form (1NF)**

* All attributes atomic.
* No multivalued attributes.
* Every relation has a primary key.
* 1NF satisfied.

**Second Normal Form (2NF)**

* No partial dependencies.
* Every non-prime attribute is fully functionally dependent on the whole primary key.
* Composite keys properly handled using unique constraints.
* 2NF satisfied.

**Third Normal Form (3NF)**

* No transitive dependencies.
* Non-prime attributes are only dependent on primary keys.
* 3NF satisfied

**Boyce-Codd Normal Form (BCNF)**

* Every functional dependency has a candidate key on the left.
* Minor adjustments made to ensure uniqueness (such as adding unique constraints in composite keys).
* Hence, the schema is already in BCNF, Proved.

**4. Final Normalized Database Structure (BCNF)**

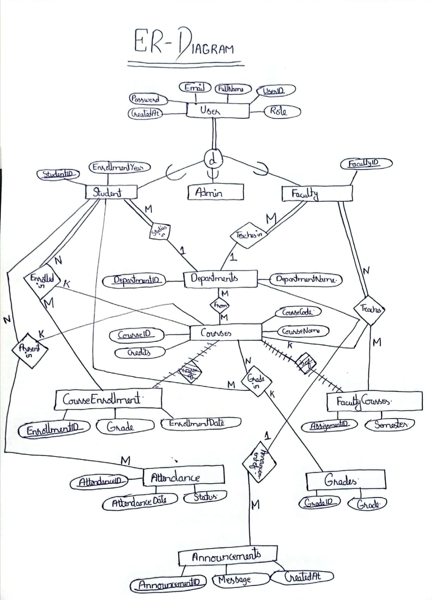
**Tables Overview:**

* **Users** (UserID PK)
* **Departments** (DepartmentID PK)
* **Faculty** (FacultyID PK, UserID FK Unique)
* **Students** (StudentID PK, UserID FK Unique)
* **Courses** (CourseID PK)
* **CourseEnrollment** (EnrollmentID PK, (StudentID, CourseID) Unique)
* **FacultyCourses** (AssignmentID PK, (FacultyID, CourseID, Semester) Unique)
* **Attendance** (AttendanceID PK, (StudentID, CourseID, AttendanceDate) Unique)
* **Grades** (GradeID PK, (StudentID, CourseID) Unique)
* **Announcements** (AnnouncementID PK)

**Relationships:**

* Users linked to Students, Faculty, and Announcements.
* Departments linked to Faculty, Students, Courses, Announcements.
* Courses linked to Enrollments, Faculty Assignments, Attendance, Grades.

**5. ER Diagram (Final Normalized Structure)**



**6. Observations and Remarks**

* The database was well designed from the beginning with minimal redundancy.
* Only minor uniqueness constraints were necessary to ensure full BCNF.
* Relationships between entities are clear and properly enforced.
* High data consistency and integrity due to good use of foreign keys.
* Easy to scale with additional features (e.g., adding more user roles, departments, or course types).
* The structure supports transactional operations efficiently and prevents anomalies.