Department of Information and Communication Technology

Faculty of Technology

University of Ruhuna

Database Management Systems

Practicum ICT 1222

Assignment 02 – Mini Project

Group 19

Submitted to: Mr.P.H.P. Nuwan Laksiri

Submitted by:

Tg/2023/1753- Harshana Prabath

Tg/2023/1759- Shonali Galpihilla

Tg/2023/1737- Yasiru Nimsara

Content

Introduction to the Group Project

The Faculty of Technology of University of Ruhuna requires an efficient and centralized system to manage student information, including personal details, attendance, marks, and results. This Student Management System aims to automate and streamline these operations using a robust Database Management System. This group project is to build a useful, manageable system to handle, store and output accurate, consistent and timely information. Also, it provides authorized users with access to the information they need.

Introduction to the Solution

The proposed solution is a Database Management System developed using MySQL to manage all student-related data. We have built a Database to store students' relevant data and output the required information by performing calculations.

The system provides functionalities for:

- Managing student and staff details
- Recording and analyzing attendance
- Managing assessment and exam marks
- Calculating grades, eligibility, SGPA, and CGPA
- Handling user-specific access through role-based privileges

Objectives of the proposed Student Management System:

- Centralized Data Storage
- Role-based Access Control
- Enhanced Reporting and Decision support
- Better student performance monitoring
- Smooth result Management
- Scalability and Reliability
- Reduced Workload
- Data Accuracy

Data Requirement Document [DRD]

Introduction

The Data Requirement Document defines all necessary data items, their structures, relationships and constraints, ER diagram and the relational mapping used for building the Student Management System database. This defines the data requirements needed to support functionalities such as student enrollment, attendance tracking, mark recording, eligibility checking, and result generation.

Data Requirements

Main entities to be included in the database:

- User → Student, Lecturer, Technical Officer
- Course
- Enrollment
- Department
- Attendance_session
- Attendance_Records
- Medical
- Mark
- Assessment_Type
- Result_summary

Detailed Data Description

User → Represents all system users including Students, Lecturers, and Technical Officers, each assigned with specific access privileges.

Course → Stores details of each course such as course code, title, credits, and lecturer in charge.

Enrollment → Links students to the courses they register for in a semester.

Department → Contains information about departments within the faculty offering the courses.

Attendance_Session → Records details of each class session (date, time, type: theory/practical) conducted for a course.

Attendance_Records → Keeps individual attendance data of students for each session, indicating presence, absence, or medical status.

Medical → Stores information about medical certificates submitted by students to justify absences.

Mark → Maintains the marks obtained by students for different assessments in each course.

Assessment_Type → Defines types of evaluations such as quizzes, assignments, mid-semester, and final exams.

Result_Summary → Summarizes results including grades, eligibility, and GPA calculations for each student.

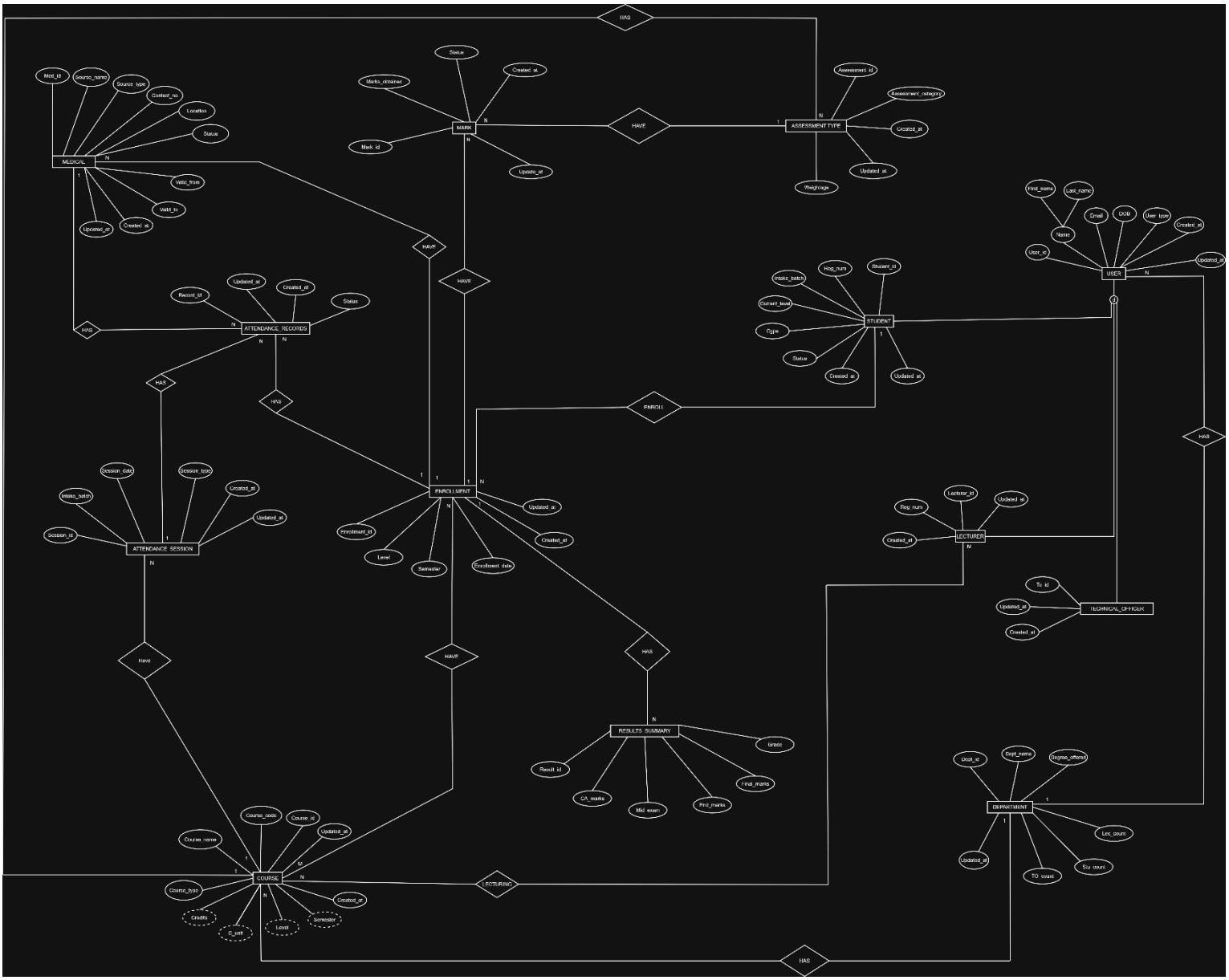
Relationships Between Entities

- A Department has many Courses and many Lecturers
- A Lecturer teaches many courses
- A Student can enroll in many Courses, and each Course can have many Students.
- Each Course consists of multiple Attendance_sessions
- Each Attendance_session has multiple Attendance_records, each linled to Student
- A Student may submit multiple Medical records related to Attendance_records
- Each Course has multiple Assessment_types
- Marks are recorded per Student, Course, and Assessment_type
- Result_summary aggregates the final marks and grades for each student per course

Data Constraints and Assumptions

- Attendance eligibility must be > 80%
- All marks are stored as percentages
- Students with medical approval will have “MC” displayed in their results
- Repeat students have a maximum achievable grade of “C”
- Suspended students have results displayed as “WH”

EER Diagram


HAS
Med_id
course_name
Source_type
Contact_no
Location
Status
MEDICAL
Valid_from
Valid_to
Updated_at
Created_at
Status
Marks_obtained
Created_at
MARK
Mark_id
Updated_at
HAVE
Assessment_id
Assessment_category
ASSESSMENT TYPE
Created_at
Updated_at
Weightage
HAVE
HAVE
Record_id
Updated_at
Created_at
Status
ATTENDANCE_RECORDS
HAS
HAS
HAS
First_name
Last_name
Gmail
DOB
User_type
Created_at
Name
User_id
Updated_at
USER
Intake_batch
Reg_num
Student_id
Current_level
Cgpa
Status
Created_at
Updated_at
STUDENT
ENROLL
HAS
Session_date
Session_type
Created_at
Intake_batch
Session_id
Updated_at
ATTENDANCE SESSION
ENROLLMENT
Updated_at
Enrollment_id
Created_at
Level
Semester
Enrollment_date
Lecturer_id
Updated_at
Reg_num
Created_at
LECTURER
To_id
Updated_at
Created_at
TECHNICAL_OFFICER
Have
HAVE
HAS
RESULTS SUMMARY
Grade
Result_id
Final_marks
CA_marks
Mid exam
Final_marks
Dept_id
Dept_name
Degree_offered
DEPARTMENT
Lec_count
Updated_at
TO_count
Stu_count
course_code
Course_id
Updated_at
Course_name
COURSE
Course_type
Created_at
Credits
Level
Semester
C_unit
LECTURING
HAS

Relational Mapping

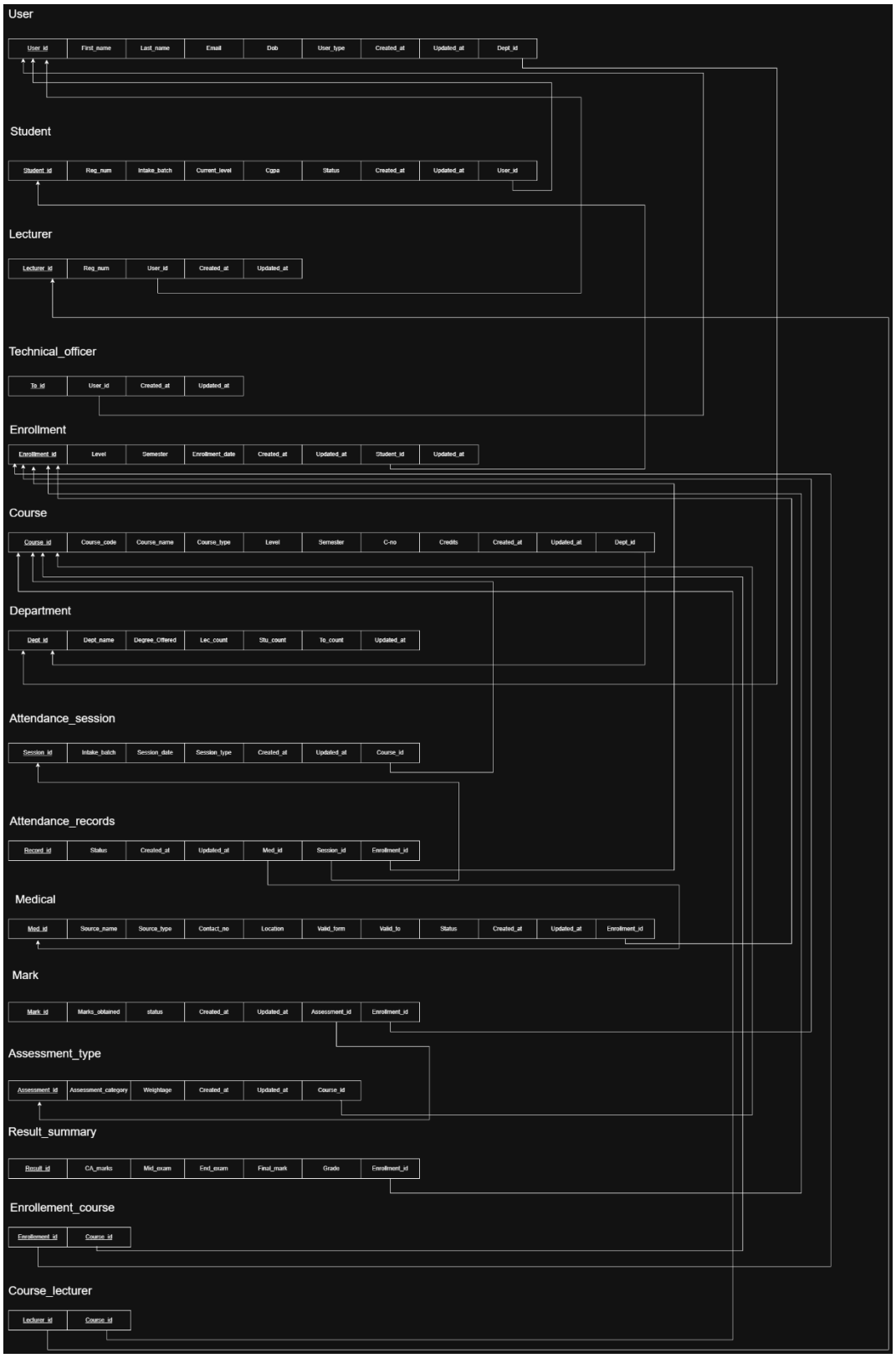


Table Structures

User

```
mysql> desc user;
+------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
| Field      | Type                                    | Null | Key | Default           | Extra                                         |
+------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
| user_id    | int                                     | NO   | PRI | NULL              |                                               |
| firstname  | varchar(20)                             | YES  |     | NULL              |                                               |
| lastname   | varchar(20)                             | YES  |     | NULL              |                                               |
| email      | varchar(50)                             | NO   | UNI | NULL              |                                               |
| dob        | date                                    | YES  |     | NULL              |                                               |
| user_type  | enum('student','lecturer','t_officer')  | YES  |     | NULL              |                                               |
| dept_id    | int                                     | YES  | MUL | NULL              |                                               |
| updated_at | timestamp                               | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at | timestamp                               | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
9 rows in set (0.04 sec)
```

Student

```
mysql> desc student;
+--------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
| Field        | Type                                    | Null | Key | Default           | Extra                                         |
+--------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
| student_id   | int                                     | NO   | PRI | NULL              | auto_increment                                |
| reg_num      | varchar(20)                             | YES  | UNI | NULL              |                                               |
| user_id      | int                                     | YES  | MUL | NULL              |                                               |
| intake_batch | int                                     | YES  |     | NULL              |                                               |
| current_level| int                                     | YES  |     | NULL              |                                               |
| cgpa         | decimal(4,2)                            | YES  |     | NULL              |                                               |
| status       | enum('active','graduated','suspended')  | YES  |     | active            |                                               |
| created_at   | timestamp                               | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
| updated_at   | timestamp                               | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
+--------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
9 rows in set (0.09 sec)
```

Lecturer

```
mysql> desc lecturer;
+-------------+-------------+------+-----+-------------------+-----------------------------------------------+
| Field       | Type        | Null | Key | Default           | Extra                                         |
+-------------+-------------+------+-----+-------------------+-----------------------------------------------+
| lecturer_id | int         | NO   | PRI | NULL              | auto_increment                                |
| reg_num     | varchar(20) | YES  |     | NULL              |                                               |
| user_id     | int         | YES  | MUL | NULL              |                                               |
| updated_at  | timestamp   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at  | timestamp   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+-------------+-------------+------+-----+-------------------+-----------------------------------------------+
5 rows in set (0.06 sec)
```

Technical Officer

```
mysql> desc technical_officer;
+------------+-----------+------+-----+-------------------+-----------------------------------------------+
| Field      | Type      | Null | Key | Default           | Extra                                         |
+------------+-----------+------+-----+-------------------+-----------------------------------------------+
| to_id      | int       | NO   | PRI | NULL              | auto_increment                                |
| user_id    | int       | YES  | MUL | NULL              |                                               |
| updated_at | timestamp | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at | timestamp | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+------------+-----------+------+-----+-------------------+-----------------------------------------------+
4 rows in set (0.01 sec)
```

Department

```
mysql> desc department;
+----------------+-------------+------+-----+-------------------+-----------------------------------------------+
| Field          | Type        | Null | Key | Default           | Extra                                         |
+----------------+-------------+------+-----+-------------------+-----------------------------------------------+
| dept_id        | int         | NO   | PRI | NULL              |                                               |
| dept_name      | varchar(50) | YES  |     | NULL              |                                               |
| to_count       | int         | YES  |     | 0                 |                                               |
| stu_count      | int         | YES  |     | 0                 |                                               |
| lec_count      | int         | YES  |     | 0                 |                                               |
| degree_offered | varchar(20) | YES  |     | NULL              |                                               |
| updated_at     | timestamp   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at     | timestamp   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+----------------+-------------+------+-----+-------------------+-----------------------------------------------+
8 rows in set (0.05 sec)
```

Course

```
mysql> desc course;
+--------------+--------------+------+-----+-------------------+-----------------------------------------------+
| Field        | Type         | Null | Key | Default           | Extra                                         |
+--------------+--------------+------+-----+-------------------+-----------------------------------------------+
| course_id    | int          | NO   | PRI | NULL              | auto_increment                                |
| course_code  | varchar(20)  | NO   | UNI | NULL              |                                               |
| course_name  | varchar(100) | NO   |     | NULL              |                                               |
| department   | varchar(15)  | YES  |     | NULL              |                                               |
| level        | int          | YES  |     | NULL              |                                               |
| semester     | int          | YES  |     | NULL              |                                               |
| c_unit       | int          | YES  |     | NULL              |                                               |
| credits      | int          | YES  |     | NULL              |                                               |
| updated_at   | timestamp    | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at   | timestamp    | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+--------------+--------------+------+-----+-------------------+-----------------------------------------------+
10 rows in set (0.19 sec)
```

Enrollment

```
mysql> desc enrollment;
+-----------------+-------------+------+-----+-------------------+-----------------------------------------------+
| Field           | Type        | Null | Key | Default           | Extra                                         |
+-----------------+-------------+------+-----+-------------------+-----------------------------------------------+
| enrollment_id   | int         | NO   | PRI | NULL              |                                               |
| reg_num         | varchar(20) | YES  | MUL | NULL              |                                               |
| level           | int         | YES  |     | NULL              |                                               |
| semester        | int         | YES  |     | NULL              |                                               |
| enrollment_date | date        | YES  |     | NULL              |                                               |
| updated_at      | timestamp   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at      | timestamp   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+-----------------+-------------+------+-----+-------------------+-----------------------------------------------+
7 rows in set (0.01 sec)
```

Attendance_session

```
mysql> desc attendance_session;
+--------------+-----------------------------+------+-----+-------------------+-----------------------------------------------+
| Field        | Type                        | Null | Key | Default           | Extra                                         |
+--------------+-----------------------------+------+-----+-------------------+-----------------------------------------------+
| session_id   | int                         | NO   | PRI | NULL              | auto_increment                                |
| course_code  | varchar(20)                 | NO   | MUL | NULL              |                                               |
| intake_batch | int                         | YES  |     | NULL              |                                               |
| session_date | date                        | YES  |     | NULL              |                                               |
| session_type | enum('practical','lecture') | YES  |     | NULL              |                                               |
| updated_at   | timestamp                   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at   | timestamp                   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+--------------+-----------------------------+------+-----+-------------------+-----------------------------------------------+
7 rows in set (0.02 sec)
```

Attendance_record

```
mysql> desc attendance_records;
+-------------+-----------------------------------+------+-----+-------------------+-----------------------------------------------+
| Field       | Type                              | Null | Key | Default           | Extra                                         |
+-------------+-----------------------------------+------+-----+-------------------+-----------------------------------------------+
| record_id   | int                               | NO   | PRI | NULL              | auto_increment                                |
| reg_num     | varchar(20)                       | YES  |     | NULL              |                                               |
| session_id  | int                               | YES  | MUL | NULL              |                                               |
| status      | enum('Present','Absent','Medical') | YES  |     | NULL              |                                               |
| medical_id  | int                               | YES  |     | -1                |                                               |
| updated_at  | timestamp                         | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at  | timestamp                         | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+-------------+-----------------------------------+------+-----+-------------------+-----------------------------------------------+
7 rows in set (0.02 sec)
```

Medical

```
mysql> desc medical;
+--------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
| Field        | Type                                    | Null | Key | Default           | Extra                                         |
+--------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
| medical_id   | int                                     | NO   | PRI | NULL              | auto_increment                                |
| reg_num      | varchar(20)                             | YES  | MUL | NULL              |                                               |
| source_name  | varchar(100)                            | YES  |     | NULL              |                                               |
| source_type  | varchar(50)                             | YES  |     | NULL              |                                               |
| contact_info | varchar(100)                            | YES  |     | NULL              |                                               |
| location     | varchar(100)                            | YES  |     | NULL              |                                               |
| valid_from   | date                                    | YES  |     | NULL              |                                               |
| valid_to     | date                                    | YES  |     | NULL              |                                               |
| status       | enum('Pending','Approved','Rejected')   | YES  |     | Pending           |                                               |
| updated_at   | timestamp                               | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
| created_at   | timestamp                               | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
+--------------+-----------------------------------------+------+-----+-------------------+-----------------------------------------------+
11 rows in set (0.01 sec)
```

Assessment_type

```
mysql> desc assessment_type;
+---------------------+-----------------------------------------------------------------------------+------+-----+-------------------+-----------------------------------------------+
| Field               | Type                                                                        | Null | Key | Default           | Extra                                         |
+---------------------+-----------------------------------------------------------------------------+------+-----+-------------------+-----------------------------------------------+
| assessment_id       | int                                                                         | NO   | PRI | NULL              | auto_increment                                |
| assessment_category | enum('Quiz 1','Quiz 2','Quiz 3','Assignment','Project','Mid Exam','End Exam') | NO   |     | NULL              |                                               |
| weightage           | decimal(5,2)                                                                | NO   |     | NULL              |                                               |
| course_id           | int                                                                         | YES  | MUL | NULL              |                                               |
| created_at          | timestamp                                                                   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
| updated_at          | timestamp                                                                   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
+---------------------+-----------------------------------------------------------------------------+------+-----+-------------------+-----------------------------------------------+
6 rows in set (0.01 sec)
```

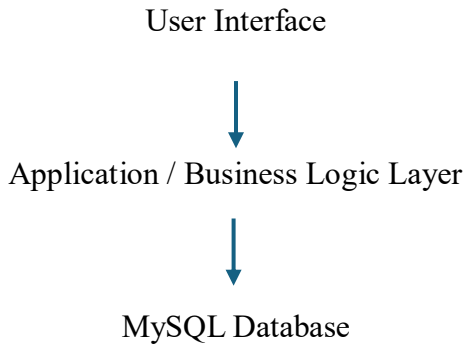
Marks

```
mysql> desc marks;
+-----------------+-------------------------+------+-----+-------------------+-----------------------------------------------+
| Field           | Type                    | Null | Key | Default           | Extra                                         |
+-----------------+-------------------------+------+-----+-------------------+-----------------------------------------------+
| mark_id         | int                     | NO   | PRI | NULL              | auto_increment                                |
| assessment_id   | int                     | NO   | MUL | NULL              |                                               |
| reg_num         | varchar(20)             | NO   | MUL | NULL              |                                               |
| marked_obtained | decimal(5,2)            | YES  |     | 0.00              |                                               |
| status          | enum('Present','Absent') | YES  |     | Present           |                                               |
| created_at      | timestamp               | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
| updated_at      | timestamp               | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
+-----------------+-------------------------+------+-----+-------------------+-----------------------------------------------+
7 rows in set (0.01 sec)
```

Result_summary

```
mysql> desc result_summary;
+----------------+-------------+------+-----+-------------------+-----------------------------------------------+
| Field          | Type        | Null | Key | Default           | Extra                                         |
+----------------+-------------+------+-----+-------------------+-----------------------------------------------+
| result_id      | int         | NO   | PRI | NULL              | auto_increment                                |
| reg_num        | varchar(20) | NO   | MUL | NULL              |                                               |
| course_id      | int         | NO   | MUL | NULL              |                                               |
| f_quiz         | decimal(5,2)| YES  |     | 0.00              |                                               |
| project        | decimal(5,2)| YES  |     | 0.00              |                                               |
| assignment     | decimal(5,2)| YES  |     | 0.00              |                                               |
| mid_exam_marks | decimal(5,2)| YES  |     | 0.00              |                                               |
| ca_marks       | decimal(5,2)| YES  |     | NULL              | STORED GENERATED                              |
| ca_pass        | varchar(6)  | YES  |     | NULL              |                                               |
| end_mark       | decimal(5,2)| YES  |     | 0.00              |                                               |
| end_pass       | varchar(6)  | YES  |     | NULL              |                                               |
| final_mark     | decimal(5,2)| YES  |     | NULL              | STORED GENERATED                              |
| grade          | varchar(2)  | YES  |     | NULL              |                                               |
| created_at     | timestamp   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED                             |
| updated_at     | timestamp   | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
+----------------+-------------+------+-----+-------------------+-----------------------------------------------+
15 rows in set (0.02 sec)
```

Architecture of the Database



- The user (Studentt/Lecturer/Technical_officer/Admin/Dean) logs into the system.
- The Application Layer handles data validation, processing and communication with the MySQL backend.
- The MySQL database layer stores all students, lecturers, marks and attendance data and results.

Tools & Technologies

- ✓ MySQL Server → Create, manage and query the relational database.
- ✓ Draw.io → Design the EER diagram and the Relational Mapping Diagram
- ✓ Notepad++ → Utilizing the writing and editing SQL queries, scripts, and code snippets efficiently.
- ✓ GitHub → Version Control and Collaboration, allowing the team to manage code updates, track changes, and share progress seamlessly.

Security Measures

Database Security is essential to protect sensitive academic information such as student marks, grades and attendance records from unauthorized access or modification. We have taken few steps to solve the security issues that can be raised related to the database.

1. Role-based Access Control
2. Privilege Limitation
3. Use of Views for secure data access
4. Authentication and password protection
5. Data validation and consistency

Users in the Database

User Role	Privileges	Description
Admin	ALL PRIVILEGES on all tables with GRANT OPTION	Responsible for database maintenance, user creation, backups, and overall system management.
Dean	ALL PRIVILEGES (without GRANT OPTION)	Oversees academic operations, manages course details, and reviews marks and attendance reports.
Lecturer	SELECT, INSERT, UPDATE on marks and attendance tables; SELECT on student and course tables	Enters and updates student marks and attendance; reviews reports.
Technical Officer (TO)	SELECT, INSERT, UPDATE on attendance-related tables and views	Manages attendance for practical sessions and verifies records.
Student	SELECT on attendance and results views only	Can view personal attendance percentage, eligibility, and final grades.

Code Snippets Used

- Stored Procedures (for calculating eligibility or grades)

```
DELIMITER //

CREATE PROCEDURE get_attendance_by_regnum_course(
    IN p_reg_num VARCHAR(20),
    IN p_course_code VARCHAR(20)

)
BEGIN
    SELECT
    ar.reg_num,
        s.course_code,
        e.level,
        e.semester,
        s.session_date,
        s.session_type,
        ar.status
    FROM attendance_records ar
    JOIN attendance_session s ON ar.session_id = s.session_id
    JOIN enrollment e ON ar.reg_num = e.reg_num
    WHERE s.course_code = p_course_code
    AND ar.reg_num = p_reg_num
    ORDER BY s.session_date, ar.reg_num;
END //

DELIMITER ;
```

```
mysql> call get_attendance_by_regnum_course("TG-2023-0004","ICT1253");
+--------------+-------------+-------+----------+--------------+--------------+---------+
| reg_num      | course_code | level | semester | session_date | session_type | status  |
+--------------+-------------+-------+----------+--------------+--------------+---------+
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-02-14   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-02-21   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-02-28   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-03-07   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-03-14   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-03-21   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-03-28   | lecture      | Absent  |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-04-04   | practical    | Absent  |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-04-11   | lecture      | Absent  |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-04-18   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-04-25   | lecture      | Absent  |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-05-02   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-05-09   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-05-16   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-05-23   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-05-30   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-06-06   | lecture      | Absent  |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-06-13   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-06-20   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-06-27   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-07-04   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-07-11   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-07-18   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-07-25   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-08-01   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-08-08   | practical    | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-08-15   | lecture      | Absent  |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-08-22   | practical    | Absent  |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-08-29   | lecture      | Present |
| TG-2023-0004 | ICT1253     |     1 |        1 | 2025-09-05   | practical    | Absent  |
+--------------+-------------+-------+----------+--------------+--------------+---------+
30 rows in set (0.01 sec)
```

```
DELIMITER //

CREATE PROCEDURE get_attendance_by_course(
    IN p_course_code VARCHAR(20)
)
BEGIN
    SELECT
    ar.reg_num,
        s.course_code,
        e.level,
        e.semester,
        s.session_date,
        s.session_type,
        ar.status
    FROM attendance_records ar
    JOIN attendance_session s ON ar.session_id = s.session_id
    JOIN enrollment e ON ar.reg_num = e.reg_num
    WHERE s.course_code = p_course_code
    ORDER BY s.session_date, ar.reg_num;
END //

DELIMITER ;
```

```
mysql> call get_attendance_by_type("ICT1212","theory");
+--------------+-------------+----------------+---------------+--------------+---------------+-----------------------+
| reg_num      | course_code | total_sessions | present_count | absent_count | medical_count | attendance_percentage |
+--------------+-------------+----------------+---------------+--------------+---------------+-----------------------+
| TG-2023-0001 | ICT1212     |             15 |            13 |            2 |             0 |                 86.67 |
| TG-2023-0002 | ICT1212     |             15 |            12 |            3 |             0 |                 80.00 |
| TG-2023-0003 | ICT1212     |             15 |            12 |            3 |             0 |                 80.00 |
| TG-2023-0004 | ICT1212     |             15 |            15 |            0 |             0 |                100.00 |
| TG-2023-0005 | ICT1212     |             15 |             8 |            7 |             0 |                 53.33 |
| TG-2023-0006 | ICT1212     |             15 |            11 |            4 |             0 |                 73.33 |
| TG-2023-0007 | ICT1212     |             15 |            13 |            2 |             0 |                 86.67 |
| TG-2023-0008 | ICT1212     |             15 |            12 |            3 |             0 |                 80.00 |
| TG-2023-0009 | ICT1212     |             15 |            10 |            5 |             0 |                 66.67 |
| TG-2023-0010 | ICT1212     |             15 |             6 |            9 |             0 |                 40.00 |
+--------------+-------------+----------------+---------------+--------------+---------------+-----------------------+
10 rows in set (0.00 sec)

Query OK, 0 rows affected (0.04 sec)
```

```
mysql> call get_attendance_by_type("ICT1253","combined");
+--------------+-------------+----------------+---------------+--------------+---------------+-----------------------+
| reg_num      | course_code | total_sessions | present_count | absent_count | medical_count | attendance_percentage |
+--------------+-------------+----------------+---------------+--------------+---------------+-----------------------+
| TG-2023-0001 | ICT1253     |             30 |            24 |            6 |             0 |                 80.00 |
| TG-2023-0002 | ICT1253     |             30 |            28 |            2 |             0 |                 93.33 |
| TG-2023-0003 | ICT1253     |             30 |            26 |            4 |             0 |                 86.67 |
| TG-2023-0004 | ICT1253     |             30 |            22 |            8 |             0 |                 73.33 |
| TG-2023-0005 | ICT1253     |             30 |            25 |            5 |             0 |                 83.33 |
| TG-2023-0006 | ICT1253     |             30 |            21 |            9 |             0 |                 70.00 |
| TG-2023-0007 | ICT1253     |             30 |            22 |            8 |             0 |                 73.33 |
| TG-2023-0008 | ICT1253     |             30 |            24 |            6 |             0 |                 80.00 |
| TG-2023-0009 | ICT1253     |             30 |            19 |           11 |             0 |                 63.33 |
| TG-2023-0010 | ICT1253     |             30 |            21 |            9 |             0 |                 70.00 |
+--------------+-------------+----------------+---------------+--------------+---------------+-----------------------+
10 rows in set (0.00 sec)

Query OK, 0 rows affected (0.04 sec)
```

```
DELIMITER //

CREATE PROCEDURE get_attendance_by_type(
    IN p_course_code VARCHAR(20),
    IN p_type ENUM('theory','practical','combined')
)
BEGIN
    SELECT
        ar.reg_num,
        s.course_code,
        COUNT(ar.record_id) AS total_sessions,
        SUM(CASE WHEN ar.status = 'Present' THEN 1 ELSE 0 END) AS present_count,
        SUM(CASE WHEN ar.status = 'Absent' THEN 1 ELSE 0 END) AS absent_count,
        SUM(CASE WHEN ar.status = 'Medical' THEN 1 ELSE 0 END) AS medical_count,
        ROUND(
            (SUM(CASE WHEN ar.status IN ('Present','Medical') THEN 1 ELSE 0 END) / COUNT(ar.record_id)) * 100,
            2
        ) AS attendance_percentage
    FROM attendance_records ar
    JOIN attendance_session s ON ar.session_id = s.session_id
    WHERE s.course_code = p_course_code
      AND (
          (p_type = 'theory' AND s.session_type = 'lecture') OR
          (p_type = 'practical' AND s.session_type = 'practical') OR
          (p_type = 'combined')
      )
    GROUP BY ar.reg_num, s.course_code
    ORDER BY ar.reg_num;
END //

DELIMITER ;
```

```
mysql> call get_eligibilities_by_regnum("TG-2023-0001");
+--------------+-------------+----------+---------+----------+----------+----------------------+-------------+
| reg_num      | course_code | ca_marks | ca_pass | end_mark | end_pass | attendance_percentage | eligibility |
+--------------+-------------+----------+---------+----------+----------+----------------------+-------------+
| TG-2023-0001 | ENG1222     |    51.41 | Pass    |   117.25 | Pass     |                80.00 | Eligible    |
| TG-2023-0001 | ICT1212     |    59.85 | Pass    |   100.15 | Pass     |                86.67 | Eligible    |
| TG-2023-0001 | ICT1222     |    43.79 | Pass    |    65.16 | Pass     |                86.67 | Eligible    |
| TG-2023-0001 | ICT1233     |    43.48 | Pass    |    88.24 | Pass     |                80.00 | Eligible    |
| TG-2023-0001 | ICT1242     |    17.14 | Pass    |    69.66 | Pass     |                93.33 | Eligible    |
| TG-2023-0001 | ICT1253     |    37.50 | Pass    |    57.79 | Pass     |                80.00 | Eligible    |
| TG-2023-0001 | TCS1212     |    46.18 | Pass    |    91.48 | Pass     |                86.67 | Eligible    |
| TG-2023-0001 | TMS1233     |    60.09 | Pass    |    97.80 | Pass     |                80.00 | Eligible    |
+--------------+-------------+----------+---------+----------+----------+----------------------+-------------+
8 rows in set (0.01 sec)

Query OK, 0 rows affected (0.04 sec)
```

```
mysql> call get_eligibilities_by_regnum("TG-2023-0003");
+--------------+-------------+----------+---------+----------+----------+----------------------+--------------+
| reg_num      | course_code | ca_marks | ca_pass | end_mark | end_pass | attendance_percentage | eligibility  |
+--------------+-------------+----------+---------+----------+----------+----------------------+--------------+
| TG-2023-0003 | ENG1222     |    59.83 | Pass    |    65.68 | Pass     |                80.00 | Eligible     |
| TG-2023-0003 | ICT1212     |    21.79 | Pass    |     0.00 | Repeat   |                80.00 | Eligible     |
| TG-2023-0003 | ICT1222     |    25.33 | Pass    |    89.78 | Pass     |                93.33 | Eligible     |
| TG-2023-0003 | ICT1233     |    19.84 | Pass    |    50.26 | Pass     |                83.33 | Eligible     |
| TG-2023-0003 | ICT1242     |    23.80 | Pass    |    64.42 | Pass     |                86.67 | Eligible     |
| TG-2023-0003 | ICT1253     |    62.43 | Pass    |    58.02 | Pass     |                86.67 | Eligible     |
| TG-2023-0003 | TCS1212     |    48.00 | Pass    |    88.94 | Pass     |                93.33 | Eligible     |
| TG-2023-0003 | TMS1233     |    41.86 | Pass    |     0.00 | Repeat   |                73.33 | Not Eligible |
+--------------+-------------+----------+---------+----------+----------+----------------------+--------------+
8 rows in set (0.02 sec)

Query OK, 0 rows affected (0.05 sec)
```

```
DELIMITER //

CREATE PROCEDURE get_eligibilities_by_regnum(IN p_reg_num VARCHAR(20))
BEGIN

    SELECT
        rs.reg_num,
        c.course_code,
        rs.ca_marks,
        rs.ca_pass,
        rs.end_mark,
        rs.end_pass,
        COALESCE(ea.attendance_percentage, nea.attendance_percentage, 0) AS attendance_percentage,
        COALESCE(ea.eligibility, nea.eligibility, 'Not Eligible') AS eligibility
    FROM result_summary rs
    JOIN course c ON rs.course_id = c.course_id
    LEFT JOIN eligible_students_attendance ea
           ON rs.reg_num = ea.reg_num AND c.course_code = ea.course_code
    LEFT JOIN not_eligible_students_attendance nea
           ON rs.reg_num = nea.reg_num AND c.course_code = nea.course_code
    WHERE rs.reg_num = p_reg_num
    ORDER BY c.course_code;

END //

DELIMITER ;
```

```
DELIMITER //

CREATE PROCEDURE get_eligibility_by_course(
    IN p_course_code VARCHAR(20)
)
BEGIN

    SELECT
        rs.reg_num,
        c.course_code,
        rs.ca_marks,
        rs.ca_pass,
        rs.end_mark,
        rs.end_pass,
        COALESCE(ea.attendance_percentage, nea.attendance_percentage, 0) AS attendance_percentage,
        COALESCE(ea.eligibility, nea.eligibility, 'Not Eligible') AS eligibility
    FROM result_summary rs
    JOIN course c ON rs.course_id = c.course_id
    LEFT JOIN eligible_students_attendance ea
           ON rs.reg_num = ea.reg_num AND c.course_code = ea.course_code
    LEFT JOIN not_eligible_students_attendance nea
           ON rs.reg_num = nea.reg_num AND c.course_code = nea.course_code
    WHERE c.course_code = p_course_code
    ORDER BY rs.reg_num;

END //

DELIMITER ;
```

```
mysql> call get_eligibility_by_course('ICT1212');
+--------------+-------------+----------+---------+----------+----------+-----------------------+--------------+
| reg_num      | course_code | ca_marks | ca_pass | end_mark | end_pass | attendance_percentage | eligibility  |
+--------------+-------------+----------+---------+----------+----------+-----------------------+--------------+
| TG-2023-0001 | ICT1212     |    59.85 | Pass    |   100.15 | Pass     |                 86.67 | Eligible     |
| TG-2023-0002 | ICT1212     |    18.50 | Pass    |    14.40 | Repeat   |                 80.00 | Eligible     |
| TG-2023-0003 | ICT1212     |    21.79 | Pass    |     0.00 | Repeat   |                 80.00 | Eligible     |
| TG-2023-0004 | ICT1212     |    34.66 | Pass    |     0.00 | Repeat   |                100.00 | Eligible     |
| TG-2023-0005 | ICT1212     |    53.85 | Pass    |    97.31 | Pass     |                 53.33 | Not Eligible |
| TG-2023-0006 | ICT1212     |    43.12 | Pass    |    74.70 | Pass     |                 73.33 | Not Eligible |
| TG-2023-0007 | ICT1212     |    63.01 | Pass    |     0.00 | Repeat   |                 86.67 | Eligible     |
| TG-2023-0008 | ICT1212     |    17.74 | Pass    |    75.92 | Pass     |                 80.00 | Eligible     |
| TG-2023-0009 | ICT1212     |    49.41 | Pass    |   111.65 | Pass     |                 66.67 | Not Eligible |
| TG-2023-0010 | ICT1212     |    22.23 | Pass    |    98.22 | Pass     |                 40.00 | Not Eligible |
+--------------+-------------+----------+---------+----------+----------+-----------------------+--------------+
10 rows in set (0.01 sec)

Query OK, 0 rows affected (0.04 sec)
```

```
DELIMITER //

CREATE PROCEDURE get_eligibility_by_regnum_and_course(
    IN p_reg_num VARCHAR(20),
    IN p_course_code VARCHAR(20)
)
BEGIN

    SELECT
        rs.reg_num,
        c.course_code,
        rs.ca_marks,
        rs.ca_pass,
        rs.end_mark,
        rs.end_pass,
        COALESCE(ea.attendance_percentage, nea.attendance_percentage, 0) AS attendance_percentage,
        COALESCE(ea.eligibility, nea.eligibility, 'Not Eligible') AS eligibility
    FROM result_summary rs
    JOIN course c ON rs.course_id = c.course_id
    LEFT JOIN eligible_students_attendance ea
           ON rs.reg_num = ea.reg_num AND c.course_code = ea.course_code
    LEFT JOIN not_eligible_students_attendance nea
           ON rs.reg_num = nea.reg_num AND c.course_code = nea.course_code
    WHERE rs.reg_num = p_reg_num
      AND c.course_code = p_course_code;

END //

DELIMITER ;
```

```
mysql> call get_eligibility_by_regnum_and_course("TG-2023-0003","ICT1253");
+--------------+-------------+-----------+---------+-----------+----------+----------------------+-------------+
| reg_num      | course_code | ca_marks | ca_pass | end_mark | end_pass | attendance_percentage | eligibility |
+--------------+-------------+-----------+---------+-----------+----------+----------------------+-------------+
| TG-2023-0003 | ICT1253     |    62.43 | Pass    |    58.02 | Pass     |                86.67 | Eligible    |
+--------------+-------------+-----------+---------+-----------+----------+----------------------+-------------+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.02 sec)
```

```
DELIMITER //

CREATE PROCEDURE get_student_course_result(
    IN p_reg_num VARCHAR(20),
    IN p_course_code VARCHAR(20)
)
BEGIN
    SELECT
    m.reg_num,
    c.course_code,
        a.assessment_category,
        m.marked_obtained,
    a.weightage
    FROM marks m
    JOIN assessment_type a ON m.assessment_id = a.assessment_id
    JOIN course c ON a.course_id = c.course_id
    WHERE m.reg_num = p_reg_num
      AND c.course_code = p_course_code
    ORDER BY a.assessment_id;
END //

DELIMITER ;


example call .............

call get_student_course_result("TG-2023-0001","ICT1212");
```

```
mysql> call get_student_course_result("TG-2023-0001","ICT1212");
+--------------+-------------+----------------------+-----------------+-----------+
| reg_num      | course_code | assessment_category  | marked_obtained | weightage |
+--------------+-------------+----------------------+-----------------+-----------+
| TG-2023-0001 | ICT1212     | Quiz 1               |           44.88 |      0.05 |
| TG-2023-0001 | ICT1212     | Quiz 1               |           44.88 |      0.05 |
| TG-2023-0001 | ICT1212     | Quiz 2               |           73.45 |      0.05 |
| TG-2023-0001 | ICT1212     | Quiz 2               |           73.45 |      0.05 |
| TG-2023-0001 | ICT1212     | Quiz 3               |           64.54 |      0.05 |
| TG-2023-0001 | ICT1212     | Quiz 3               |           64.54 |      0.05 |
| TG-2023-0001 | ICT1212     | Assignment           |           83.78 |      0.05 |
| TG-2023-0001 | ICT1212     | Assignment           |           83.78 |      0.05 |
| TG-2023-0001 | ICT1212     | Project              |           99.04 |      0.05 |
| TG-2023-0001 | ICT1212     | Project              |           99.04 |      0.05 |
| TG-2023-0001 | ICT1212     | Mid Exam             |           85.54 |      0.20 |
| TG-2023-0001 | ICT1212     | Mid Exam             |           85.54 |      0.20 |
| TG-2023-0001 | ICT1212     | End Exam             |           83.46 |      0.60 |
| TG-2023-0001 | ICT1212     | End Exam             |           83.46 |      0.60 |
+--------------+-------------+----------------------+-----------------+-----------+
14 rows in set (0.00 sec)

Query OK, 0 rows affected (0.04 sec)
```

```sql
DELIMITER //

CREATE PROCEDURE get_result_by_coursecode(
    IN p_course_code VARCHAR(20)
)
BEGIN
    SELECT
        rs.reg_num,
        c.course_code,
        rs.f_quiz,
        rs.project,
        rs.assignment,
        rs.mid_exam_marks,
        rs.ca_marks,
        rs.end_mark,
        rs.final_mark,
        rs.grade
    FROM result_summary rs
    JOIN course c ON rs.course_id = c.course_id
    WHERE c.course_code = p_course_code
    ORDER BY rs.reg_num;
END //

DELIMITER ;
```

```
mysql> call get_result_by_coursecode("ICT1212");
+---------------+-------------+---------+---------+------------+-----------------+----------+----------+------------+-------+
| reg_num       | course_code | f_quiz  | project | assignment | mid_exam_marks  | ca_marks | end_mark | final_mark | grade |
+---------------+-------------+---------+---------+------------+-----------------+----------+----------+------------+-------+
| TG-2023-0001  | ICT1212     |    7.35 |    9.90 |       8.38 |           34.22 |    59.85 |   100.15 |     160.00 | A+    |
| TG-2023-0002  | ICT1212     |    1.00 |    1.50 |       2.00 |           14.00 |    18.50 |    14.40 |      32.90 | F     |
| TG-2023-0003  | ICT1212     |    6.86 |    8.73 |       6.20 |            0.00 |    21.79 |     0.00 |      21.79 | F     |
| TG-2023-0004  | ICT1212     |    0.00 |    0.00 |       0.00 |           34.66 |    34.66 |     0.00 |      34.66 | F     |
| TG-2023-0005  | ICT1212     |    8.18 |    6.29 |       5.59 |           33.79 |    53.85 |    97.31 |     151.16 | A+    |
| TG-2023-0006  | ICT1212     |   10.00 |    6.75 |       9.25 |           17.12 |    43.12 |    74.70 |     117.82 | A+    |
| TG-2023-0007  | ICT1212     |    9.05 |    7.80 |       7.28 |           38.88 |    63.01 |     0.00 |      63.01 | B-    |
| TG-2023-0008  | ICT1212     |    6.43 |    4.58 |       6.73 |            0.00 |    17.74 |    75.92 |      93.66 | A+    |
| TG-2023-0009  | ICT1212     |    9.39 |    4.03 |       4.48 |           31.51 |    49.41 |   111.65 |     161.06 | A+    |
| TG-2023-0010  | ICT1212     |    6.12 |    8.54 |       7.57 |            0.00 |    22.23 |    98.22 |     120.45 | A+    |
+---------------+-------------+---------+---------+------------+-----------------+----------+----------+------------+-------+
10 rows in set (0.00 sec)

Query OK, 0 rows affected (0.03 sec)
```

```
DELIMITER //

CREATE PROCEDURE get_result_by_regnum(IN p_reg_num VARCHAR(20))
BEGIN
    SELECT *
    FROM result_summary
    WHERE reg_num = p_reg_num;
END //

DELIMITER ;
```

```
mysql> call get_result_by_regnum("TG-2023-0003");
+-----------+-------------+-----------+---------+---------+------------+-----------------+----------+---------+----------+----------+------------+-------+---------------------+---------------------+
| result_id | reg_num     | course_id | f_quiz  | project | assignment | mid_exam_marks  | ca_marks | ca_pass | end_mark | end_pass | final_mark | grade | created_at          | updated_at          |
+-----------+-------------+-----------+---------+---------+------------+-----------------+----------+---------+----------+----------+------------+-------+---------------------+---------------------+
|         3 | TG-2023-0003 |         1 |    6.86 |    8.73 |       6.20 |            0.00 |    21.79 | Pass    |     0.00 | Repeat   |      21.79 | F     | 2025-10-26 12:24:36 | 2025-10-26 12:30:40 |
|        13 | TG-2023-0003 |         2 |    9.68 |    9.02 |       6.63 |            0.00 |    25.33 | Pass    |    89.78 | Pass     |     115.11 | A+    | 2025-10-26 12:24:36 | 2025-10-26 12:30:40 |
|        23 | TG-2023-0003 |         3 |    7.97 |    6.58 |       5.29 |            0.00 |    19.84 | Pass    |    50.26 | Pass     |      70.10 | B+    | 2025-10-26 12:24:36 | 2025-10-26 12:30:40 |
|        33 | TG-2023-0003 |         4 |    8.91 |    6.90 |       7.99 |            0.00 |    23.80 | Pass    |    64.42 | Pass     |      88.22 | A+    | 2025-10-26 12:24:36 | 2025-10-26 12:30:40 |
|        43 | TG-2023-0003 |         5 |    9.25 |    9.13 |       7.58 |           36.47 |    62.43 | Pass    |    58.62 | Pass     |     120.45 | A+    | 2025-10-26 12:24:36 | 2025-10-26 12:30:40 |
|        53 | TG-2023-0003 |         6 |    7.16 |    6.00 |       6.89 |           39.78 |    59.83 | Pass    |    65.68 | Pass     |     125.51 | A+    | 2025-10-26 12:24:36 | 2025-10-26 12:30:40 |
|        63 | TG-2023-0003 |         7 |    5.75 |    5.92 |       4.07 |           32.26 |    48.00 | Pass    |    88.94 | Pass     |     136.94 | A+    | 2025-10-26 12:24:36 | 2025-10-26 12:30:40 |
|        73 | TG-2023-0003 |         8 |    9.20 |    5.48 |       5.22 |           21.96 |    41.86 | Pass    |     0.00 | Repeat   |      41.86 | F     | 2025-10-26 12:24:36 | 2025-10-26 12:30:40 |
+-----------+-------------+-----------+---------+---------+------------+-----------------+----------+---------+----------+----------+------------+-------+---------------------+---------------------+
8 rows in set (0.00 sec)

Query OK, 0 rows affected (0.04 sec)
```

```
DELIMITER //

CREATE PROCEDURE get_student_gpa(
    IN p_reg_num VARCHAR(20)
)
BEGIN
    SELECT
        reg_num,
        level,
        semester,
        sgpa,
        cgpa
    FROM student_grades_summary
    WHERE reg_num = p_reg_num
    ORDER BY level, semester;
END //

DELIMITER ;
```

```
mysql> call get_student_gpa("TG-2023-0001");
+--------------+-------+----------+------+------+
| reg_num      | level | semester | sgpa | cgpa |
+--------------+-------+----------+------+------+
| TG-2023-0001 |     1 |        2 | 4.00 | 4.00 |
+--------------+-------+----------+------+------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)

mysql> call get_student_gpa("TG-2023-0003");
+--------------+-------+----------+------+------+
| reg_num      | level | semester | sgpa | cgpa |
+--------------+-------+----------+------+------+
| TG-2023-0003 |     1 |        2 | 2.84 | 2.84 |
+--------------+-------+----------+------+------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)
```

```sql
DELIMITER //

CREATE PROCEDURE get_final_marks_by_regnum(
    IN p_regnum VARCHAR(20)
)
BEGIN
    SELECT
        rs.reg_num,
        c.course_code,
        rs.final_mark,
        rs.grade
    FROM result_summary rs
    JOIN course c ON rs.course_id = c.course_id
    WHERE rs.reg_num = p_regnum
    ORDER BY rs.reg_num;
END //

DELIMITER ;
```

```
mysql> call get_final_marks_by_regnum("TG-2023-0001");
+--------------+-------------+------------+-------+
| reg_num      | course_code | final_mark | grade |
+--------------+-------------+------------+-------+
| TG-2023-0001 | ICT1212     |     160.00 | A+    |
| TG-2023-0001 | ICT1222     |     108.95 | A+    |
| TG-2023-0001 | ICT1233     |     131.72 | A+    |
| TG-2023-0001 | ICT1242     |      86.80 | A+    |
| TG-2023-0001 | ICT1253     |      95.29 | A+    |
| TG-2023-0001 | ENG1222     |     168.66 | A+    |
| TG-2023-0001 | TCS1212     |     137.66 | A+    |
| TG-2023-0001 | TMS1233     |     157.89 | A+    |
+--------------+-------------+------------+-------+
8 rows in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)

mysql> call get_final_marks_by_regnum("TG-2023-0003");
+--------------+-------------+------------+-------+
| reg_num      | course_code | final_mark | grade |
+--------------+-------------+------------+-------+
| TG-2023-0003 | ICT1212     |      21.79 | F     |
| TG-2023-0003 | ICT1222     |     115.11 | A+    |
| TG-2023-0003 | ICT1233     |      70.10 | B+    |
| TG-2023-0003 | ICT1242     |      88.22 | A+    |
| TG-2023-0003 | ICT1253     |     120.45 | A+    |
| TG-2023-0003 | ENG1222     |     125.51 | A+    |
| TG-2023-0003 | TCS1212     |     136.94 | A+    |
| TG-2023-0003 | TMS1233     |      41.86 | F     |
+--------------+-------------+------------+-------+
8 rows in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)
```

- Views (for displaying marks/attendance summaries)

```
mysql>
mysql> CREATE VIEW not_eligible_students_attendance AS
    -> SELECT
    ->     ar.reg_num,
    ->     s.course_code,
    ->     COUNT(ar.record_id) AS total_sessions,
    ->     SUM(CASE WHEN ar.status = 'Present' THEN 1 ELSE 0 END) AS present_count,
    ->     SUM(CASE WHEN ar.status = 'Absent' THEN 1 ELSE 0 END) AS absent_count,
    ->     SUM(CASE WHEN ar.status = 'Medical' THEN 1 ELSE 0 END) AS medical_count,
    ->     ROUND(
    ->         (SUM(CASE WHEN ar.status IN ('Present','Medical') THEN 1 ELSE 0 END) / COUNT(ar.record_id)) * 100,
    ->         2
    ->     ) AS attendance_percentage,
    ->     'Not Eligible' AS eligibility
    -> FROM attendance_records ar
    -> JOIN attendance_session s ON ar.session_id = s.session_id
    -> GROUP BY ar.reg_num, s.course_code
    -> HAVING attendance_percentage < 80
    -> ORDER BY ar.reg_num, s.course_code;
Query OK, 0 rows affected (0.01 sec)
```

```
mysql>
mysql> CREATE VIEW eligible_students_attendance AS
    -> SELECT
    ->     ar.reg_num,
    ->     s.course_code,
    ->     COUNT(ar.record_id) AS total_sessions,
    ->     SUM(CASE WHEN ar.status = 'Present' THEN 1 ELSE 0 END) AS present_count,
    ->     SUM(CASE WHEN ar.status = 'Absent' THEN 1 ELSE 0 END) AS absent_count,
    ->     SUM(CASE WHEN ar.status = 'Medical' THEN 1 ELSE 0 END) AS medical_count,
    ->     ROUND(
    ->         (SUM(CASE WHEN ar.status IN ('Present','Medical') THEN 1 ELSE 0 END) / COUNT(ar.record_id)) * 100,
    ->         2
    ->     ) AS attendance_percentage,
    ->     'Eligible' AS eligibility
    -> FROM attendance_records ar
    -> JOIN attendance_session s ON ar.session_id = s.session_id
    -> GROUP BY ar.reg_num, s.course_code
    -> HAVING attendance_percentage >= 80
    -> ORDER BY ar.reg_num, s.course_code;
Query OK, 0 rows affected (0.01 sec)
```

```
mysql>
mysql> CREATE VIEW student_course_attendance_summary AS
    -> SELECT
    ->     ar.reg_num,
    ->     s.course_code,
    ->     COUNT(ar.record_id) AS total_sessions,
    ->     SUM(CASE WHEN ar.status = 'Present' THEN 1 ELSE 0 END) AS present_count,
    ->     SUM(CASE WHEN ar.status = 'Absent' THEN 1 ELSE 0 END) AS absent_count,
    ->     SUM(CASE WHEN ar.status = 'Medical' THEN 1 ELSE 0 END) AS medical_count,
    ->     ROUND(
    ->         (SUM(CASE WHEN ar.status IN ('Present','Medical') THEN 1 ELSE 0 END) / COUNT(ar.record_id)) * 100,
    ->         2
    ->     ) AS attendance_percentage,
    ->     CASE
    ->         WHEN ROUND(
    ->             (SUM(CASE WHEN ar.status IN ('Present','Medical') THEN 1 ELSE 0 END) / COUNT(ar.record_id)) * 100,
    ->             2
    ->         ) >= 80 THEN 'Eligible'
    ->         ELSE 'Not Eligible'
    ->     END AS eligibility
    -> FROM attendance_records ar
    -> JOIN attendance_session s ON ar.session_id = s.session_id
    -> GROUP BY ar.reg_num, s.course_code
    -> ORDER BY ar.reg_num, s.course_code;
Query OK, 0 rows affected (0.01 sec)
```

- Triggers (optional, for automatic updates)

```
mysql> DELIMITER //
mysql>
mysql> CREATE TRIGGER trg_update_cgpa
    -> AFTER UPDATE ON result_summary
    -> FOR EACH ROW
    -> BEGIN
    ->     DECLARE total_points DECIMAL(10,2) DEFAULT 0.0;
    ->     DECLARE total_credits DECIMAL(10,2) DEFAULT 0.0;
    ->     DECLARE cgpa_val DECIMAL(4,2);
    ->
    ->
    ->     SELECT SUM(
    ->                 CASE
    ->                     WHEN final_mark >= 85 THEN 4.0
    ->                     WHEN final_mark >= 80 THEN 4.0
    ->                     WHEN final_mark >= 75 THEN 3.7
    ->                     WHEN final_mark >= 70 THEN 3.3
    ->                     WHEN final_mark >= 65 THEN 3.0
    ->                     WHEN final_mark >= 60 THEN 2.7
    ->                     WHEN final_mark >= 55 THEN 2.3
    ->                     WHEN final_mark >= 50 THEN 2.0
    ->                     WHEN final_mark >= 45 THEN 1.7
    ->                     ELSE 0
    ->                 END * c.credits
    ->             ),
    ->             SUM(c.credits)
    ->     INTO total_points, total_credits
    ->     FROM result_summary r
    ->     JOIN course c ON r.course_id = c.course_id
    ->     WHERE r.reg_num = NEW.reg_num;
    ->
    ->     IF total_credits > 0 THEN
    ->         SET cgpa_val = ROUND(total_points / total_credits, 2);
    ->
    ->         UPDATE student
    ->         SET cgpa = cgpa_val,
    ->             updated_at = CURRENT_TIMESTAMP
    ->         WHERE reg_num = NEW.reg_num;
    ->     END IF;
    -> END //
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> DELIMITER //
mysql>
mysql> CREATE TRIGGER trg_medical_approved
    -> AFTER UPDATE ON medical
    -> FOR EACH ROW
    -> BEGIN
    ->     IF NEW.status = 'Approved' THEN
    ->         UPDATE result_summary r
    ->         JOIN assessment_type a
    ->           ON a.course_id = r.course_id
    ->         JOIN marks m
    ->           ON m.assessment_id = a.assessment_id
    ->           AND m.reg_num = r.reg_num
    ->         SET r.grade = 'MC'
    ->         WHERE r.reg_num = NEW.reg_num
    ->           AND m.status = 'Absent'
    ->           AND a.assessment_category IN ('Mid Exam', 'End Exam')
    ->           AND a.exam_date BETWEEN NEW.valid_from AND NEW.valid_to;
    ->     END IF;
    -> END //
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> CREATE TRIGGER trg_result_pass
    -> BEFORE UPDATE ON result_summary
    -> FOR EACH ROW
    -> BEGIN
    ->
    ->     IF NEW.ca_marks < 16 THEN
    ->         SET NEW.ca_pass = 'Fail';
    ->     ELSE
    ->         SET NEW.ca_pass = 'Pass';
    ->     END IF;
    ->
    ->     IF NEW.end_mark < 21 THEN
    ->         SET NEW.end_pass = 'Repeat';
    ->     ELSE
    ->         SET NEW.end_pass = 'Pass';
    ->     END IF;
    -> END //
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> CREATE TRIGGER trg_update_result_summary
    -> AFTER INSERT ON marks
    -> FOR EACH ROW
    -> BEGIN
    ->     DECLARE total_quiz DECIMAL(5,2) DEFAULT 0.00;
    ->     DECLARE total_project DECIMAL(5,2) DEFAULT 0.00;
    ->     DECLARE total_assignment DECIMAL(5,2) DEFAULT 0.00;
    ->     DECLARE mid_marks DECIMAL(5,2) DEFAULT 0.00;
    ->     DECLARE end_marks DECIMAL(5,2) DEFAULT 0.00;
    ->     DECLARE course INT;
    ->
    ->
    ->
    ->     SELECT course_id INTO course FROM assessment_type WHERE assessment_id = NEW.assessment_id;
    ->
    ->
    ->     SELECT SUM(marked_obtained * weightage) INTO total_quiz
    ->     FROM (
    ->         SELECT m.marked_obtained, at.weightage
    ->         FROM marks m
    ->         JOIN assessment_type at ON m.assessment_id = at.assessment_id
    ->         WHERE m.reg_num = NEW.reg_num
    ->           AND at.course_id = course
    ->           AND at.assessment_category LIKE 'Quiz%'
    ->         ORDER BY marked_obtained DESC
    ->         LIMIT 2
    ->     ) AS top2_quizzes;
    ->
    ->
    ->     SELECT
    ->         SUM(CASE WHEN at.assessment_category = 'Project' THEN m.marked_obtained * at.weightage ELSE 0 END),
    ->         SUM(CASE WHEN at.assessment_category = 'Assignment' THEN m.marked_obtained * at.weightage ELSE 0 END),
    ->         SUM(CASE WHEN at.assessment_category = 'Mid Exam' THEN m.marked_obtained * at.weightage ELSE 0 END),
    ->         SUM(CASE WHEN at.assessment_category = 'End Exam' THEN m.marked_obtained * at.weightage ELSE 0 END)
    ->     INTO total_project, total_assignment, mid_marks, end_marks
    ->     FROM marks m
    ->     JOIN assessment_type at ON m.assessment_id = at.assessment_id
    ->     WHERE m.reg_num = NEW.reg_num
    ->       AND at.course_id = course;
    ->
    ->
    ->     IF EXISTS (SELECT 1 FROM result_summary WHERE reg_num = NEW.reg_num AND course_id = course) THEN
    ->         UPDATE result_summary
    ->         SET f_quiz = total_quiz,
    ->             project = total_project,
    ->             assignment = total_assignment,
    ->             mid_exam_marks = mid_marks,
    ->             end_mark = end_marks,
    ->             updated_at = CURRENT_TIMESTAMP
    ->         WHERE reg_num = NEW.reg_num AND course_id = course;
    ->     ELSE
    ->         INSERT INTO result_summary(reg_num, course_id, f_quiz, project, assignment, mid_exam_marks, end_mark)
    ->         VALUES(NEW.reg_num, course, total_quiz, total_project, total_assignment, mid_marks, end_marks);
    ->     END IF;
    ->
```

```
mysql> CREATE TRIGGER trg_update_grade_after_update
    -> BEFORE UPDATE ON result_summary
    -> FOR EACH ROW
    -> BEGIN
    ->     IF NEW.grade IS NULL OR NEW.grade <> 'MC' THEN
    ->         IF NEW.final_mark >= 85 THEN
    ->             SET NEW.grade = 'A+';
    ->         ELSEIF NEW.final_mark >= 80 THEN
    ->             SET NEW.grade = 'A';
    ->         ELSEIF NEW.final_mark >= 75 THEN
    ->             SET NEW.grade = 'A-';
    ->         ELSEIF NEW.final_mark >= 70 THEN
    ->             SET NEW.grade = 'B+';
    ->         ELSEIF NEW.final_mark >= 65 THEN
    ->             SET NEW.grade = 'B';
    ->         ELSEIF NEW.final_mark >= 60 THEN
    ->             SET NEW.grade = 'B-';
    ->         ELSEIF NEW.final_mark >= 55 THEN
    ->             SET NEW.grade = 'C+';
    ->         ELSEIF NEW.final_mark >= 50 THEN
    ->             SET NEW.grade = 'C';
    ->         ELSEIF NEW.final_mark >= 45 THEN
    ->             SET NEW.grade = 'C-';
    ->         ELSE
    ->             SET NEW.grade = 'F';
    ->         END IF;
    ->     END IF;
    -> END$$
Query OK, 0 rows affected (0.01 sec)
```

Problems faced during the development

During the development of Student Management System, following are the challenges we encountered.

- Database Normalization

Achieving the correct level of normalization while maintaining easy data retrieval without data redundancy

- Privilege Management

Creating users with the right access and prevent privilege escalation or loss of access.

- Handling Special Student cases

Cases like medical, repeat and suspended students were complex.

- Data consistency and relationships

Maintaining referential integrity among related tables.

- Time Management

Testing all functionalities within a limited timeframe.

Solutions to the problems encountered

Above challenges were addressed using different solutions:

- Query Optimization
- Stored Procedures and Views
- Normalization & Relational Mapping
- Systematic Privilege Testing
- Continuous Debugging & Validation

Backend Hosting

The database was hosted locally using MySQL Server through XAMPP for development and testing.

Hosting Choice:

Local hosting was selected because it provided easy access, full administrative control, and quick testing without internet dependency.
For future scalability, the system can easily be migrated to a cloud-based MySQL environment

Benefits:

- Faster query execution and debugging during development.
- No external connectivity delays.
- Full database control for user creation and privilege testing.

Cloud Hosting

If the system is moved to a cloud environment such as AWS or Azure, a few changes are needed to make it secure and accessible:

- IP Access Control:
  Only allow trusted computers or networks to connect to the cloud database.
- User Login Security:
  Use strong passwords and create separate accounts for each user role (Admin, Lecturer, etc.).
- Automatic Backups:
  Set up daily or weekly backups to protect against data loss.
- Scalability:
  The system can be easily upgraded if more users or storage are needed in the future.

Individual Contribution- I

TG/2023/1753- U.G. Harshana Prabhath

GitHub Acc- HarshanaPrabhath (Harshana Prabhath)

Gmail- harshanaprabhath147@gmail.com

Contribution Overview:

Member 1 played a vital role in the implementation and functional development of the Student Management System. Their main responsibilities included creating database tables, inserting sample data, and developing views, stored procedures, and triggers that automated key database functions.

Detailed Contribution:

Using MySQL Server and Notepad++, Member 1 created the complete set of database tables and ensured that all fields, constraints, and relationships matched the logical model. They entered realistic data for students, lecturers, courses, and marks.

This member developed SQL views for simplified access to attendance and marks data, ensuring user-specific data retrieval. They also wrote stored procedures to calculate eligibility and grades automatically and implemented triggers to maintain real-time updates when data changed.

Version control and progress tracking were managed through GitHub, which helped maintain consistent development and collaboration among group members. Throughout the process, the member frequently referred to MySQL documentation, W3Schools tutorials, and ChatGPT for query optimization and syntax clarification.

Reflection:

This member gained strong hands-on experience with SQL scripting, data manipulation, and automation in MySQL. Their contribution enhanced the system's efficiency and reliability while ensuring that database operations adhered to normalization and consistency rules.

Individual Contribution- II

TG/2023/1759- U. Shonali Galpihilla

GitHub Acc- ShonaUG (Shonali Galpihilla)

Gmail- shona20ug@gmail.com

Contribution Overview:

Member 2 was responsible for the database design and documentation stages of the project. Their main tasks included designing the Enhanced Entity Relationship (EER) Diagram, assisting with table creation, inserting data, and compiling the final project report.

Detailed Contribution:

Using Draw.io, Member 2 designed the EER diagram to represent entities, attributes, and relationships across the database. This diagram provided a clear visual reference for the database structure and guided the implementation phase.

They also worked in MySQL Server and Notepad++ to create tables based on the EER design, insert data, and validate relationships. Member 2 led the creation of the project report, combining all sections—including introduction, DRD, ER and relational mapping diagrams, tools, security, and hosting details—into a cohesive and professional academic document.

For version tracking and document sharing, GitHub was used to maintain consistency among the group members. Member 2 also consulted AI tools (ChatGPT), W3Schools, and university-provided study materials to verify syntax, understand best practices, and ensure academic accuracy in documentation.

Reflection:

Through this contribution, the member developed strong analytical, design, and technical writing skills. Their use of both design tools and documentation platforms ensured that the project met academic and technical standards.

Individual Contribution- III

TG/2023/1737- K. L. Yasiru Nimsara

GitHub Acc- klynimsara

Gmail- kulasinliyanageyasirunimsara@gmail.com

Contribution Overview:

Member 3's primary role was to design the Relational Mapping Diagram and help implement the database structure and sample data in MySQL Server. Their work focused on transforming the EER diagram into a fully normalized relational schema.

Detailed Contribution:

Using Draw.io, Member 3 created the Relational Mapping Diagram, defining how entities from the EER model would translate into relational tables. They identified primary and foreign keys, resolved many-to-many relationships, and ensured normalization up to Third Normal Form (3NF).

This member used MySQL Server and Notepad++ to create and populate the database tables, then tested relationships using join queries to verify data accuracy. They also helped document the relational mapping process and contributed to testing and validation.

Collaboration and version control were managed through GitHub, ensuring smooth integration of all team contributions. Member 3 used W3Schools, MySQL tutorials, and ChatGPT to refine SQL syntax, understand complex joins, and troubleshoot errors during testing.

Reflection:

Member 3 strengthened their understanding of database normalization, relational design, and SQL query testing. Their contribution was crucial in ensuring that the database structure was logically sound, efficient, and aligned with the academic requirements of the project.

References

We have used different sources to gather information.

- OpenAI
- Lecture Materials
- W3Schools
- MySQL Documentations