



**Department of Information and  
Communication  
Technology Faculty of Technology  
University of Ruhuna**

**Database Management Systems Practicum (ICT 1222)**

**Assignment 02 – Mini Project Group 14**

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

Submitted by:

TG/2021/1087 M.N.F Ashfa

TG/2021/1045 M.R.F Sajeeya

TG/2021/1042 S.A.M Sukry

TG/2021/1067 N.U.M Hewage

## **Table of Contents**

### **1. Introduction**

1.1 Purpose

1.2 Scope

1.3 Document Conventions

1.4 Reference

### **2. User Roles and Permission**

### **3. Data Requirement**

### **4. Functional Requirement**

4.1. User Authentication and Authorization

4.2. Student Management:

4.3. Course Unit Management

4.4. Attendance Recording

4.5. Marks Recording

4.6. Eligibility Checking

4.7. Grading

4.5. Report Generation

4.9. Medical Leave Handling

4.10. Data Security and Backup

### **5. Normal Flow of Functional Requirements**

5.1. Student Entity

5.2. Course Unit Entity

5.3. User entity

5.4 Attendance Entity

5.5. Marks Entity

5.6. Lecturer Entity

5.7. Assignment Entity

5.8. Technical Officer Entity

### **6. Documentation**

## 1. Introduction

The Student Information Management System (SIMS) is designed to manage student details, attendance, marks, and results for the Faculty of Technology. This system will provide various user roles with different levels of access to ensure efficient data management and reporting.

### 1.1 Purpose

The purpose of this document is to outline the functional and non-functional requirements of the SIMS, including user roles, data management, attendance, marks recording, eligibility checking, grading, and reporting.

### 1.2 Scope

The scope of SIMS includes:

- User accounts: Admin, Dean, Lecturer, Technical Officer, Student.
- Management of student, course unit, and user data.
- Attendance tracking for Level 01 Semester 02 courses.
- Recording of various types of exam marks.
- Eligibility checks based on attendance and CA marks.
- Grading according to faculty by-laws.
- Reporting features for marks, grades, SGPA, and CGPA.

### 1.3 Document Conventions

- All section headings follow a hierarchical numbering format (e.g., 2.1, 2.2) for easy reference.
- Technical terms and acronyms are spelled out upon first use and then abbreviated.

### 1.4 References

- Lecture Notes of Fundamental of Database Management Systems
- <https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document>
- [https://amirsmvt.github.io/Database/Static\\_files/Fundamental\\_of\\_Database\\_Systems.pdf](https://amirsmvt.github.io/Database/Static_files/Fundamental_of_Database_Systems.pdf)
- FUNDAMENTALS OF Database Systems SEVENTH EDITION by Ramez Elmasri
- ER was Drawn Using [https://www.figma.com/file/FL3zTbGFiNQRrTarCVZ979/ER-SMIS-\(DBMS-Mini-Project\)?type=whitebo](https://www.figma.com/file/FL3zTbGFiNQRrTarCVZ979/ER-SMIS-(DBMS-Mini-Project)?type=whitebo)

## 2. User Roles and Permissions

- Admin: Full privileges with grant option.
- Dean: Full privileges without grant.
- Lecturer: All privileges without grant and user creation.
- Technical Officer: Read, write, and update permissions for attendance-related tables/views.
- Student: Read permission for final attendance and final marks/grades tables/views.

### **3. Data Requirements**

The database must store information for at least:

- 10 proper students, 5 repeat students.
- 5 lecturers, 5 technical officers.
- Basic data for these roles, including at least 6 different attributes.

### **4. Functional Requirements**

#### **4.1. User Authentication and Authorization:**

- The system shall provide user authentication for Admin, Dean, Lecturer, Technical Officer, and Student.
- Each user role shall have specific access privileges and permissions as defined in the system.

#### **4.2. Student Management:**

- The system shall allow administrators to add, edit, and delete student records.
- Students shall be categorized as proper, repeat, or suspended according to their status.
- Administrators and relevant personnel shall be able to update student medical status.

#### **4.3. Course Unit Management:**

- Admins shall be able to manage course units, including creating, editing, and deleting records.
- Each course unit shall include attributes like course code, name, credits, theory/practical indicator, and lecturer in charge.

#### **4.4. Attendance Recording:**

- Lecturers shall have access to record student attendance for theory and practical sessions.
- Attendance records shall include student ID, course code, date, and attendance percentages.
- The system shall support the recording of medical leave for individual students on specific dates.

#### **4.5. Marks Recording:**

- Lecturers shall be able to record marks for various assessment types, such as quizzes, mid-semester, and final exams.
- Marks shall be recorded on a scale of 0 to 100.
- The system shall calculate the CA (Continuous Assessment) part of the final marks based on predefined weightage.

#### **4.6. Eligibility Checking:**

- The system shall check if students are eligible to sit for final exams based on their attendance percentages and CA marks.
- Students who meet the eligibility criteria shall be marked as eligible.

#### 4.7. Grading:

- The system shall automatically assign grades to students based on their final marks and faculty by-laws.
- For suspended students, all results shall be marked as "WH" (Withheld).
- Repeat students shall have a maximum grade of "C" for any course unit.

#### 4.8. Report Generation:

- The system shall provide the capability to generate various reports, including:
- Attendance summaries for the whole batch and by course code.
- CA marks summaries for the whole batch and individually.
- Final marks and grades for individuals and the entire batch.
- SGPA and CGPA calculations with subject and grade details.

#### 4.9. Medical Leave Handling:

- The system shall allow students to submit medical leave requests.
- Administrators and technical officers shall have access to review and approve/reject medical leave requests.

#### 4.10. Data Security and Backup:

- The system shall ensure data security, including user authentication and authorization.
- Regular data backups and recovery procedures shall be implemented to safeguard data.

#### 4.11 Users

- Admin
- Dean.
- Lecturer.
- Technical Officer

### **5. Normal Flow of Functional Requirements**

#### 5.1. Student Entity:

##### Attributes:

- Student ID (Primary Key)
- Registration Number (Unique)
- Name
- Status (Proper, Repeat, Suspended)
- Medical Status (e.g., Medical Certificate)

##### Privileges:

- Admin: Full CRUD (Create, Read, Update, Delete) privileges.
- Dean: Read-only privileges.

- Lecturer: No access.
- Technical Officer: Read and Update privileges for medical status.
- Student: Read-only privileges for own data.

Constraints:

- Registration Number must be unique.
- Status should be one of Proper, Repeat, or Suspended.

## 5.2. Course Unit Entity:

Attributes:

- Course Code (Primary Key)
- Course Name
- Credits
- Type (Theory, Practical)
- Lecturer In Charge

Privileges:

- Admin: Full CRUD privileges.
- Dean: Read-only privileges.
- Lecturer: Read-only privileges for assigned courses.
- Technical Officer: No access.
- Student: No access.

Constraints:

- Course Code must be unique.
- Type should be one of Theory or Practical.

## 3. User Entity:

Attributes:

- User ID (Primary Key)
- Username
- Password
- User Role (Admin, Dean, Lecturer, Technical Officer, Student)

Privileges:

- Admin: Full CRUD privileges.
- Dean: Read-only privileges for user management.
- Lecturer: No access.
- Technical Officer: No access.
- Student: No access.

Constraints:

- Username must be unique.
- User Role should be one of the defined roles.

#### 4. Attendance Entity:

Attributes:

- Attendance ID (Primary Key)
- Student ID (Foreign Key)
- Course Code (Foreign Key)
- Date
- Theory Attendance
- Practical Attendance
- Medical Leave (Boolean)

Privileges:

- Admin: Full CRUD privileges.
- Dean: Read-only privileges.
- Lecturer: Full CRUD privileges for attendance records related to their courses.
- Technical Officer: Read and Update privileges for medical leave.
- Student: Read-only privileges for own attendance records.

Constraints:

- Student ID and Course Code must reference valid students and courses.
- Date should be within the semester.
- Theory and Practical Attendance values should be within a valid range.

#### 5.5. Marks Entity:

Attributes:

- Marks ID (Primary Key)
- Student ID (Foreign Key)
- Course Code (Foreign Key)
- Exam Type (Quiz, Assessment, Mid-Semester, Final)
- Marks (out of 100)

Privileges:

- Admin: Full CRUD privileges.
- Dean: Read-only privileges.
- Lecturer: Full CRUD privileges for marks related to their courses.
- Technical Officer: No access.
- Student: Read-only privileges for own marks.

Constraints:

- Student ID and Course Code must reference valid students and courses.
- Exam Type should be one of the defined types.
- Marks should be within the valid range (0-100).

5.6. Lecturer Entity:

Attributes:

- Lecturer ID (Primary Key)
- First Name
- Last Name
- Email
- Phone Number
- Department

Privileges:

- Admin: Full CRUD (Create, Read, Update, Delete) privileges for lecturer records.
- Dean: Read-only privileges for viewing lecturer records.
- Lecturer: Read-only privileges for their own profile.
- Technical Officer: No access to lecturer records.
- Student: No access to lecturer records.

Constraints:

- Lecturer ID must be unique.
- Email should follow a valid email format.
- Phone Number should be in a valid format.

5.7. Assignments Entity:

Attributes:

- Assignment ID (Primary Key)
- Course Code (Foreign Key)
- Assignment Name
- Due Date
- Description
- Marks (out of 100)

Privileges:

- Admin: Full CRUD privileges for assignments.
- Dean: Read-only privileges for viewing assignment details.
- Lecturer: Full CRUD privileges for assignments related to their courses.
- Technical Officer: No access to assignment records.



- Student: Read-only privileges for viewing assignment details.

Constraints:

- Course Code must reference a valid course.
- Due Date should be within a valid range.
- Marks should be within the valid range (0-100).

#### 5.8. Technical Officer Entity:

Attributes:

- Technical Officer ID (Primary Key)
- First Name
- Last Name
- Email
- Phone Number
- Department

Privileges:

- Admin: Full CRUD privileges for technical officer records.
- Dean: Read-only privileges for viewing technical officer records.
- Lecturer: No access to technical officer records.
- Technical Officer: Read-only privileges for their own profile.
- Student: No access to technical officer records.

Constraints:

- Technical Officer ID must be unique.
- Email should follow a valid email format.
- Phone Number should be in a valid format.

## 5. Documentation

The development process must follow the DDLC (Database Development Life Cycle) and produce related documents, including Data Requirement Document, ER diagrams, and Relational Mapping diagrams.