UNICOM TIC MANAGEMENT SYSTEM

# Final Assignment Report

Student Name: Sanjeevan

Project Type: Management System

Submitted To: UnicomTIC

Date: 24.6.2025

## 1. Introduction

The Unicom TIC Management System (UMS) is a simple desktop application designed for beginners to manage basic operations of a school. It handles courses, subjects, students, exams, marks, and timetables, with a login system for Admin, Staff, Students, and Lecturers. The system includes computer labs and lecture halls allocation to assign places (labs for computer classes, halls for regular lessons) to timetable entries. You’ll build it using C# WinForms with a Model-View-Controller (MVC) structure and a SQLite database to save data. This assignment helps you learn basic C# programming, create forms, use a database, and build a login system. It’s designed to be easy, focusing on simple tasks to help you understand coding for a school system.

## 2. Objectives

• Build a simple desktop app using C# WinForms and a basic MVC design.  
• Create modules: Course & Subject Management, Student Management, Exam & Marks Management, and Timetable Management with Computer Labs and Lecture Halls Allocation.  
• Make a basic login system for Admin, Staff, Students, and Lecturers with role-based access.  
• Use MVC to organize data (Model), forms (View), and logic (Controller).  
• Save data in a SQLite database with simple add, view, edit, and delete actions.  
• Build an easy WinForms interface with buttons and lists.  
• Check inputs and show error messages.  
• Use C# async/await to keep the app smooth.

## 3. System Scope and Features

The UMS has a few simple modules, with data saved in SQLite and access controlled by a login system. Computer labs and lecture halls are part of the Timetable module.

## 3.1 Login System

• Login for Admin, Staff, Students, and Lecturers.  
• Role-based permissions.  
• Dashboard shows buttons based on role.

## 3.2 Course & Subject Management

• Admin can add/edit/delete courses and subjects.  
• All users can view.

## 3.3 Student Management

• Admin can manage student data.  
• Students can view their own details.

## 3.4 Exam & Marks Management

• Admin can add exams.  
• Admin/Lecturer/Staff can add/edit/view marks.  
• Students can view their own marks.

## 3.5 Timetable Management with Room Allocation

• Assign subject, time slot, and room (lab/hall).  
• Admin adds/edit timetables.  
• All roles can view timetable.

## 4. Architecture and Design

• MVC architecture: Model-View-Controller  
• SQLite used for persistent storage  
• Role-based UI behavior

## 5. Technical Requirements

• Visual Studio, C# WinForms  
• SQLite DB  
• Controls: ComboBox, TextBox, Button, DataGridView

## 6. Folder Structure

• Models, Views, Controllers, Repositories  
• Organized in a clean folder structure

## 7. Development Process

• Setup project  
• Create DB tables  
• Create forms  
• Connect forms to DB  
• Test roles and modules

## 8. Challenges

• Role filtering  
• Room assignment  
• Navigation between forms

## 9. Bonus Features

• Room type filters  
• Top marks display  
• Pop-up alerts

## 10. Deliverables

• Source Code with unicomtic.db  
• Documentation and demo

## 11. Resources

• Microsoft Docs, C# and WinForms tutorials  
• SQLite documentation

## 12. Notes

• Start early  
• Focus on role-based features  
• Keep UI simple and clean

## 13. Conclusion

The Unicom TIC Management System project provided a complete learning experience in developing a real-world desktop application using C# and SQLite. It allowed for a practical understanding of MVC architecture, role-based access, and database integration. Through managing modules such as students, courses, subjects, exams, marks, and timetables, the system demonstrates a functional and structured approach to software development. The addition of computer labs and lecture halls enhances the timetabling feature, giving it practical classroom relevance. This project strengthens beginner-level programming skills and sets a solid foundation for more advanced application development.