**Unicom TIC Management System**

Name: Disalini

Index Number: UT010735

Date: June 21, 2025

# **Table of Contents**

1. Introduction  
2. System Architecture  
3. Functional Requirements  
4. Implementation Details  
5. Database Design  
6. Challenges Faced  
7. Conclusion  
8. Appendices

# **1. Introduction**

The Unicom TIC Management System is designed to manage academic operations such as student registration, course and subject management, examination marks entry, and timetable scheduling. It supports multiple user roles with separate permissions for admins, instructors, and students.

# **2. System Architecture**

This project uses a layered MVC architecture with the following technologies:  
- Language: C#  
- UI Framework: WinForms  
- Architecture: Model-View-Controller (MVC)  
- Database: SQLite

A simplified architecture flow:

LoginForm → MainForm → Panel Loads UserControls  
 ↓ ↓  
Controller → Service → Repository → SQLite DB

# **3. Functional Requirements**

|  |  |
| --- | --- |
| Feature | Description |
| User Login | Role-based authentication |
| Course Management | Add/edit/delete courses |
| Subject Management | Subjects assigned to courses and instructors |
| Student Management | CRUD operations on student data |
| Exam Marks | Enter and view student marks |
| Timetable | Schedule subjects to time slots and rooms |
| Profile & Password | View/update profile, change password |

# **4. Implementation Details**

## **4.1 Login Module**

Handles authentication and redirects to respective role dashboards.

## **4.2 Admin Dashboard**

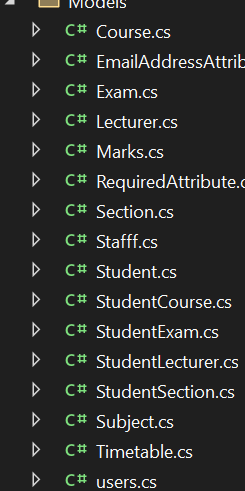
Manages users, courses, and subjects.

## **4.3 Instructor Dashboard**

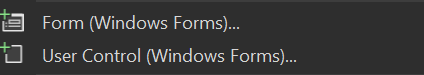
Enters exam marks and views timetables.

## **4.4 Student Dashboard**

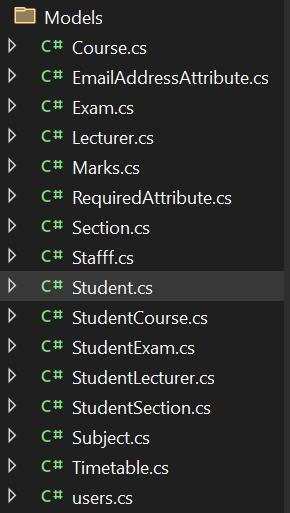
Views profile, enrolled courses, marks, and timetable.



*Figure 1: Project Folder Structure*



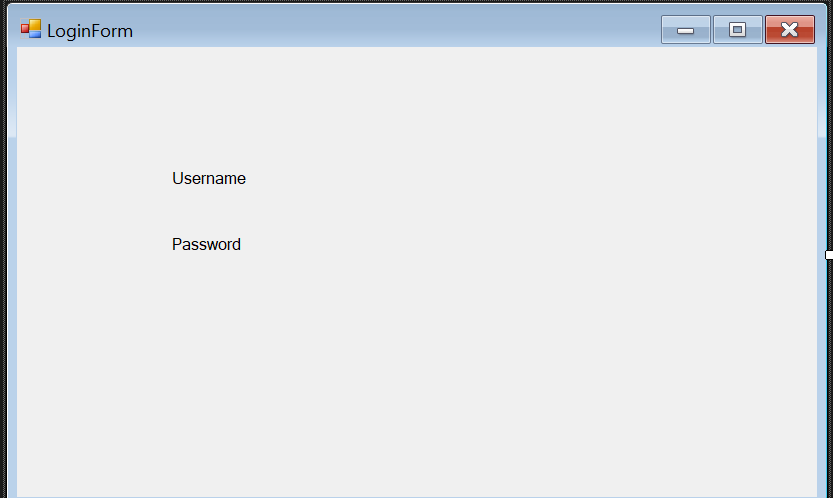
*Figure 2: Form and UserControl Options*



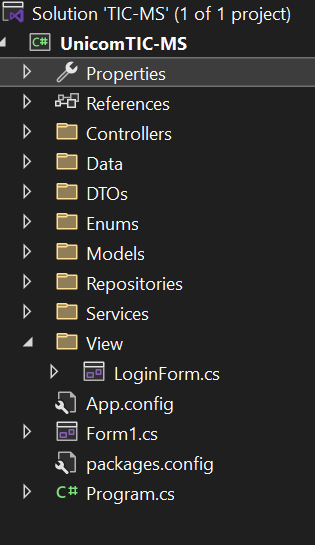
*Figure 3: Models Folder with Classes*



*Figure 4: InitializeTables() Method*



*Figure 5: Login Form Design*



*Figure 6: Full Solution Structure*

# **5. Database Design**

|  |  |
| --- | --- |
| Table Name | Description |
| Users | Stores name, email, role |
| Login | Stores username, password, userId |
| Courses | Course name, duration |
| Subjects | Subject name, courseId, instructorId |
| Students | Name, email, courseId |
| Marks | StudentId, SubjectId, Marks, ExamDate |
| Timetable | SubjectId, Room, Day, Time |

# **6. Challenges Faced**

- Handling role-based access  
- UI navigation using panel and UserControl  
- Storing relational data in SQLite

# **7. Conclusion**

The system helped reinforce understanding of C#, WinForms, and MVC. It successfully demonstrates a working management system for a TIC. Future improvements could include online access, email notifications, and report generation.

# **8. Appendices**

Additional screenshots, code snippets, and references may be added here.