Unicom TIC Management System

Developed Using: C#, WinForms, SQLite
Submitted by: P. Anjela
UT Number: UT010302

Introduction

The **Unicom TIC Management System (UMS)** is a basic yet comprehensive desktop application designed to help manage daily academic operations within an educational institution. This beginner-friendly project aims to help understand essential concepts in C#, Windows Forms, MVC architecture, database management using SQLite, and role-based access systems.

The Unicom TIC Management System is a desktop application designed to manage basic operations of a school. This project aims to provide a simple and efficient way to handle courses, subjects, students, exams, marks, and timetables.

Acknowledgement

I would like to thank Unicom TIC and My Lecturers,Mentors and my friends whose guidance and encouragement helped me complete this project. I also appreciate the support from my peers and the resources provided by Microsoft and SQLite documentation that greatly aided the development process.

Objective of the Project

- To build a user-friendly, responsive desktop application using **C# WinForms**.
- To understand and apply the **MVC design pattern** for better organization of application logic.
- To manage data using **SQLite** with create, read, update, and delete (CRUD) operations.
- To implement **role-based login access** for Admin, Staff, Lecturers, and Students.
- To allow basic scheduling through **timetable management** with room allocation (labs/halls).
- To demonstrate form design and data-binding techniques with proper validation and error handling.
- **Extra Features**

Role-Based Login System

- A secure login system with different roles

Admin – Full access: can add/edit/delete all data including courses, students, exams, marks, timetables, and attendance.

Staff – Can manage exams and marks and view timetables.

Lecturer – Can view timetable and mark/edit student attendance and marks.

Student – Can view their own profile, timetable, marks, and attendance.

- Role-based dashboards hide or show features depending on the user's access level.

Room Allocation (Lab/Hall) in Timetable

- Admin can assign a **Room (Lab or Hall)** when adding a timetable entry.
- Rooms are stored in a dedicated table with type (`Lab`, `Hall`).
- ComboBox is used to select room.

Form View and Hide (Navigation)

- After login, the **LoginForm is hidden**, and the **MainForm is shown**.
- Forms are shown or hidden dynamically using `Show()` and `Hide()` methods.
- On logout, current form closes and **LoginForm is reopened**.

Pop-up Messages

- Important alerts and confirmations using `MessageBox.Show()`:
    - Login success/failure.
    - Data saved or updated (e.g., "Timetable saved!").
    - Validation errors (e.g., "Please select a room").

Error Logging

- Errors such as failed logins, database issues, or form validation failures are **logged to a text file** (`errorlog.txt`).

Technical Details

Feature	**Technology Used**
GUI	Windows Forms (WinForms)
Language	C# (.NET Framework)
Database	SQLite (Local database)

Feature	**Technology Used**
	`System.Data.SQLite` via NuGet
Architecture	MVC (Model-View-Controller)
UI Elements	DataGridView, ComboBox, Buttons, TextBoxes
Data Storage	`.db` file for SQLite

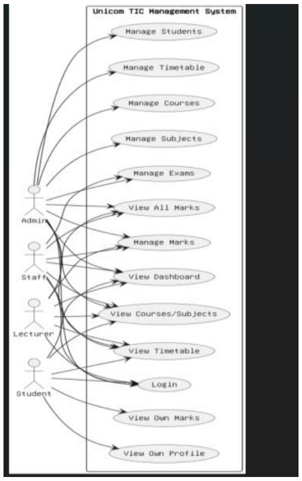
Budget Plan

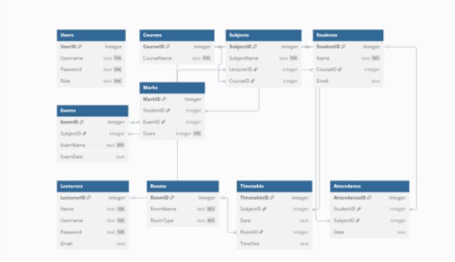
Item	**Estimated Cost (USD)**
Visual Studio Community	$0 (Free version used)
SQLite	$0 (Open-source)
System.Data.SQLite Library	$0 (Installed via NuGet)
Hardware (Laptop/PC)	$0 (Personal use)
Total	**$0 (No cost incurred)**

Design

Use Case Diagram

ER Diagram





References

- Microsoft Docs
- SQLite Official Site
- C# Tutorials: Code with Karthik ,Tamil Programmer C#
- System.Data.SQLite: https://system.data.sqlite.org/
- YouTube Tutorials on WinForms & MVC
- Stack Overflow for debugging and best practices
- Get help from Chatgpt

Challenges

- Table query is not written correctly
- Runtime error
- Role base Login
- Exam Marks management
- Timetable Management Time slot
- Data is not getting saved in DataGridView
- Repository (Database Manager)

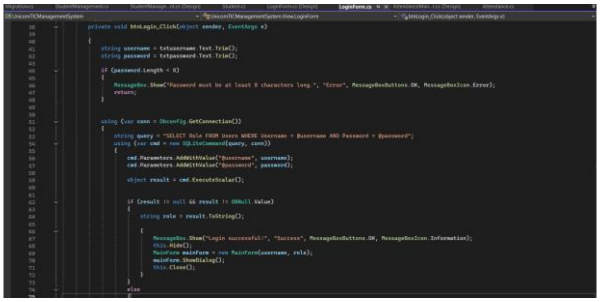
Overcome

- I ran each form and noted down the errors or missing parts...
- I corrected each form accordingly, and then checked whether the data was being saved in the database...
- I also took help from ChatGPT.

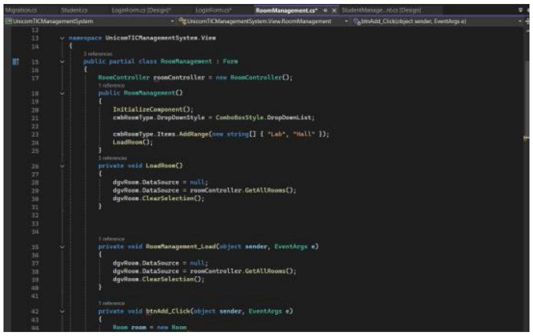
2. Code Samples (Screenshots)

- Role base Login and validation
- Login Main Form Run

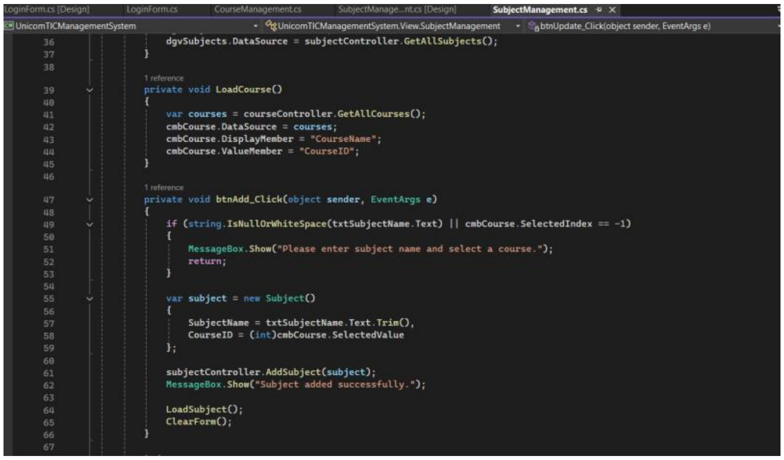
- Room allocation and dropdownlist
- Course Management Add/Update
- Subject Management Add and ClearForm


LoginForm.cs
UnicomTICManagementSystem
UnicomTICManagementSystem.View.LoginForm
btnLogin_Click(object sender, EventArgs e)

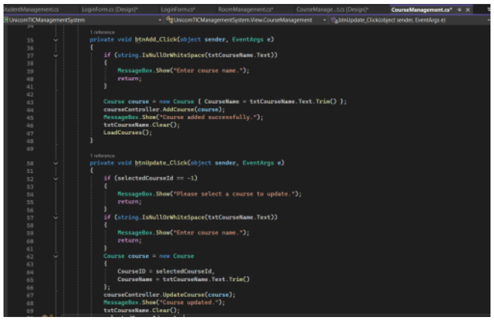
```
private void btnLogin_Click(object sender, EventArgs e)

{
    string username = txtusername.Text.Trim();
    string password = txtpassword.Text.Trim();

    if (password.Length < 8)
    {
        MessageBox.Show("Password must be at least 8 characters long.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
        return;
    }


    using (var conn = Dbconfig.GetConnection())
    {
        string query = "SELECT Role FROM Users WHERE Username = @username AND Password = @password";
        using (var cmd = new SQLiteCommand(query, conn))
        {
            cmd.Parameters.AddWithValue("@username", username);
            cmd.Parameters.AddWithValue("@password", password);

            object result = cmd.ExecuteScalar();


            if (result != null && result != DBNull.Value)
            {
                string role = result.ToString();

                {
                    MessageBox.Show("Login successful!", "Success", MessageBoxButtons.OK, MessageBoxIcon.Information);
                    this.Hide();
                    MainForm mainForm = new MainForm(username, role);
                    mainForm.ShowDialog();
                    this.Close();
                }
            }
            else
```


Migration.cs
Student.cs
LoginForm.cs [Design]*
LoginForm.cs*
RoomManagement.cs*
StudentManage...nt.cs [Design]
UnicomTICManagementSystem
UnicomTICManagementSystem.View.RoomManagement
btnAdd_Click(object sender, EventArgs e)

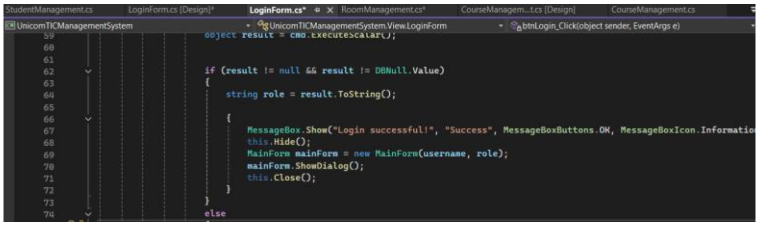
```
namespace UnicomTICManagementSystem.View
{
    public partial class RoomManagement : Form
    {
        RoomController roomController = new RoomController();
        public RoomManagement()
        {
            InitializeComponent();
            cmbRoomType.DropDownStyle = ComboBoxStyle.DropDownList;

            cmbRoomType.Items.AddRange(new string[] { "Lab", "Hall" });
            LoadRoom();
        }
        private void LoadRoom()
        {
            dgvRoom.DataSource = null;
            dgvRoom.DataSource = roomController.GetAllRooms();
            dgvRoom.ClearSelection();
        }



        private void RoomManagement_Load(object sender, EventArgs e)
        {
            dgvRoom.DataSource = null;
            dgvRoom.DataSource = roomController.GetAllRooms();
            dgvRoom.ClearSelection();
        }

        private void btnAdd_Click(object sender, EventArgs e)
        {
            Room room = new Room
```


LoginForm.cs [Design]
LoginForm.cs
CourseManagement.cs
SubjectManage...nt.cs [Design]
SubjectManagement.cs
UnicomTICManagementSystem
UnicomTICManagementSystem.View.SubjectManagement
btnUpdate_Click(object sender, EventArgs e)

```
        dgvSubjects.DataSource = subjectController.GetAllSubjects();
    }

    private void LoadCourse()
    {
        var courses = courseController.GetAllCourses();
        cmbCourse.DataSource = courses;
        cmbCourse.DisplayMember = "CourseName";
        cmbCourse.ValueMember = "CourseID";
    }

    private void btnAdd_Click(object sender, EventArgs e)
    {
        if (string.IsNullOrWhiteSpace(txtSubjectName.Text) || cmbCourse.SelectedIndex == -1)
        {
            MessageBox.Show("Please enter subject name and select a course.");
            return;
        }

        var subject = new Subject()
        {
            SubjectName = txtSubjectName.Text.Trim(),
            CourseID = (int)cmbCourse.SelectedValue
        };

        subjectController.AddSubject(subject);
        MessageBox.Show("Subject added successfully.");

        LoadSubject();
        ClearForm();
    }
```


StudentManagement.cs
LoginForm.cs [Design]*
LoginForm.cs*
RoomManagement.cs*
CourseManage...t.cs [Design]*
CourseManagement.cs*
UnicomTICManagementSystem
UnicomTICManagementSystem.View.CourseManagement
btnUpdate_Click(object sender, EventArgs e)

```
private void btnAdd_Click(object sender, EventArgs e)
{
    if (string.IsNullOrWhiteSpace(txtCourseName.Text))
    {
        MessageBox.Show("Enter course name.");
        return;
    }

    Course course = new Course { CourseName = txtCourseName.Text.Trim() };
    courseController.AddCourse(course);
    MessageBox.Show("Course added successfully.");
    txtCourseName.Clear();
    LoadCourses();
}

private void btnUpdate_Click(object sender, EventArgs e)
{
    if (selectedCourseId == -1)
    {
        MessageBox.Show("Please select a course to update.");
        return;
    }
    if (string.IsNullOrWhiteSpace(txtCourseName.Text))
    {
        MessageBox.Show("Enter course name.");
        return;
    }
    Course course = new Course
    {
        CourseID = selectedCourseId,
        CourseName = txtCourseName.Text.Trim()
    };
    courseController.UpdateCourse(course);
    MessageBox.Show("Course updated.");
    txtCourseName.Clear();
```


StudentManagement.cs
LoginForm.cs [Design]*
LoginForm.cs*
RoomManagement.cs*
CourseManagem...t.cs [Design]
CourseManagement.cs
UnicomTICManagementSystem
UnicomTICManagementSystem.View.LoginForm
btnLogin_Click(object sender, EventArgs e)

```
        object result = cmd.ExecuteScalar();


        if (result != null && result != DBNull.Value)
        {
            string role = result.ToString();

            {
                MessageBox.Show("Login successful!", "Success", MessageBoxButtons.OK, MessageBoxIcon.Informatio
                this.Hide();
                MainForm mainForm = new MainForm(username, role);
                mainForm.ShowDialog();
                this.Close();
            }
        }
        else
```