Introduction to Unicom Management System

The Unicom Management System is a considerable software solution designed to efficiently manage and streamline the day-to-day administrative and academic operations of a college. This system supports the organization in handling essential functions related to students, lecturers, and staff, and integrates several key modules such as:

- Course and Subject Management
- Class, Examination and Marks Management
- Timetable and Hall Allocation
- Attendance Tracking
- Student Information Management

Each module is interconnected to ensure seamless data flow and operational efficiency, reducing manual workload and improving accuracy across departments. With user-friendly interfaces and secure role-based access.

Functioning of System

The system includes a centralized dashboard accessible by the Admin, which provides access to all system features. Students have restricted access, allowing them to view only their personal details within the Student Management module. Lecturers can access modules related to Exams and Marks, Attendance Management, and Student Information for all students. Staff members have access to Timetable and Hall Management as well as Attendance Management.

The system operation begins with the Admin, who is responsible for creating and managing user accounts by assigning specific roles. This ensures that only registered users can access the system. When a new user is created with an assigned role by the Admin, they can then register their account and receive a username and password. Upon logging in, the system automatically detects the user's role and grants access permissions accordingly, ensuring that each user can only access features relevant to their role.

Steps of Function in the Unicom Management System

1. **User Registration by Role**
   The Admin registers users by assigning roles (Student, Lecturer, Staff), ensuring role-based access to system features.
2. **Course and Subject Creation**
   Courses are created and linked with relevant subjects to organize academic structure.
3. **Add Exams and Classes**
   Exams and classes are added and assigned to subjects for academic planning and assessment.
4. **Hall Allocation**
   Halls are scheduled and allocated based on sessions and course requirements.
5. **Marks Entry for Students**
   Lecturers input student marks for each exam under the appropriate subject.
6. **Timetable Creation**
   Timetables are generated based on lecturers, halls, dates, and time slots to manage class schedules efficiently.
7. **Attendance Management**
   Attendance is recorded for students, lecturers, and staff for each session.
8. **Student Access to Information**
   Students can securely view their **timetable**, **exam marks**, and **attendance records** from their dashboard.

Key Features Implemented

Inserted Role based notification when User logged in to the System.

I have implemented role-based notifications that appear when a user logs into the system, displaying a personalized welcome message and informing them of their available features and access level based on their assigned role (Admin, Lecturer, Student, or Staff).
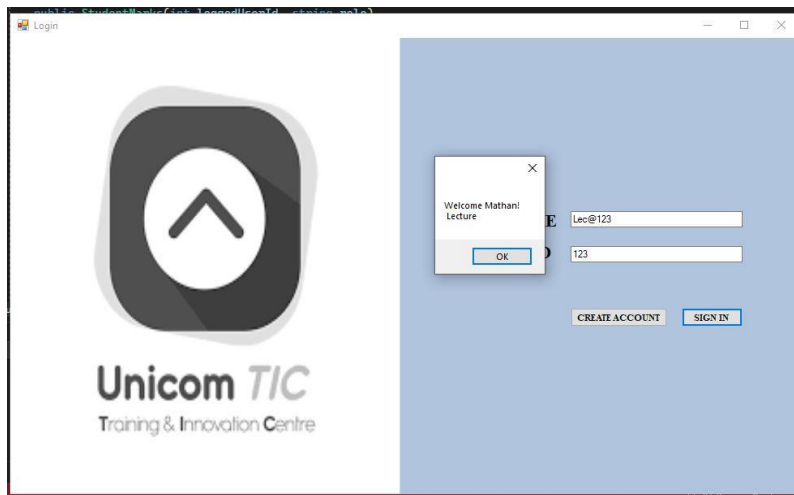
Given User based Access. It is Allowing Modules Based On the Features
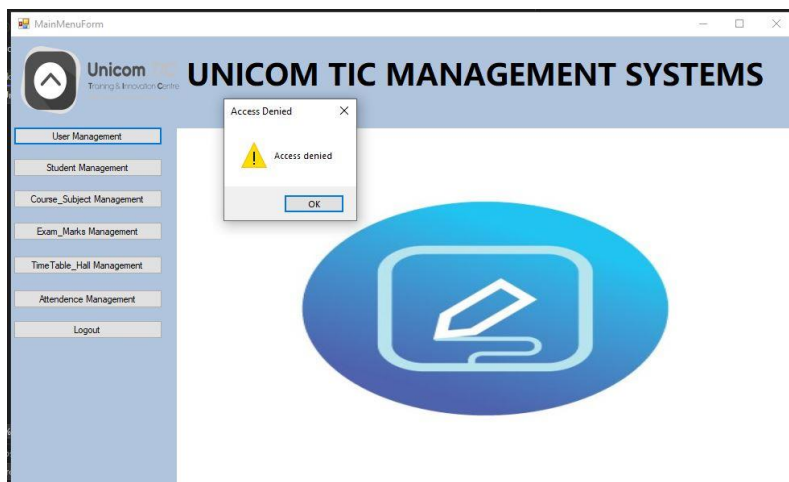
Given CRUD operation features for all Forms(Add,Edit,Delete,Search and Clear)

I have given Add features for subjects based on the Course Selection:

System will Allocate Class room according to the required Exam or Class Session

Add Attendance based on the Student, Subject and Date With Status

Add Marks based on Student name ,Subject and Mark.When Enter marks, it will
Show grade and store.

**Add Timetable lecture Name, Course name, Subject name,Room ,day and
Time.**

When allocate a time in particular class room for a lecture ,it can not repeat.

Allocate Class With Topic name

Allocate Exam For Specific Subject

View Timetable for Admin,Staff and Lecture by enter Student Id. It is For View Marks,Attendence too

View Timetable for Student by enter username and password.**It is For View Marks,Attendence too**
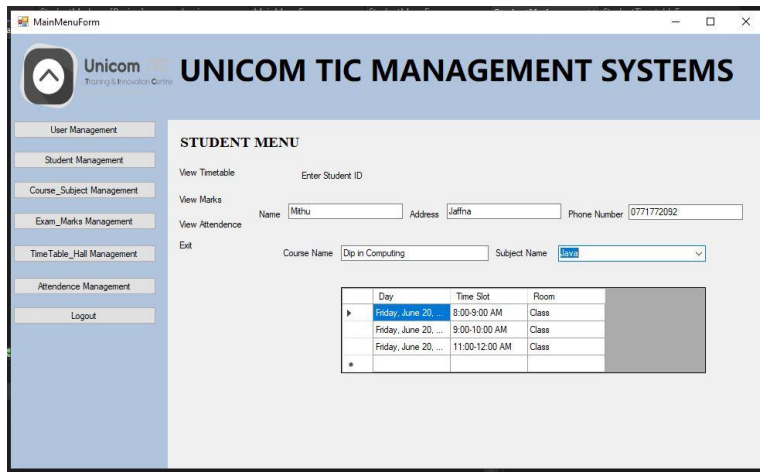
Technologies used

Environment:

Visual Studio with C# WinForms.

Database:

SQLite, using System.Data.SQLite.Core (install via NuGet).

User Interface:

WinForms with Form Designer, using Button, TextBox, ComboBox, DataGridView,Date timePicker.

OOP:

Encapsulation: Use private fields and public properties (e.g., private string name; with public string Name { get; set; }).

Inheritance (optional): Use a Person class for Student if you want to share Name.

Design Pattern:

MVC: Keep data, forms, and logic separate.

Error Handling:

Check inputs (e.g., Phone number is 10 digits).

Show messages (e.g., "Wrong User name and password" or "Select Subject").

Handle database errors (e.g., "Database is locked , No such column name m.score ").

Challenges Faced and Methods to Solved