Unicom TIC Management System

A.I.M.Thinukshini Derose

UT No:- UT 010618

1. User Authentication Overview

The system requires all users to log in before accessing any functionality. There are four types of users in the system:

Admin

Staff

Student

Teacher

Each user type has a unique username and password combination to ensure secure access and role-based functionality.

2. Sample Login Credentials

Admin

Username: Admin

Password: admin007

Role: Admin

Staff

Username: Staff

Password: staff007

Role: Staff

Student & Teacher

Similar to Admin and Staff, students and teachers will also have their own unique usernames, passwords, and assigned roles.

3. Role-Based Access Functionality

Admin View

Upon logging in, admins are directed to the admin dashboard.

The admin has full access to the system.

Admins can add, update, and delete details of:

Students

Teachers

Staff

Student Form

After logging in, students can:

View their timetable

Check their marks

Access their course details

Teacher Form

When teachers log in, they can:

Access student details

View their timetable

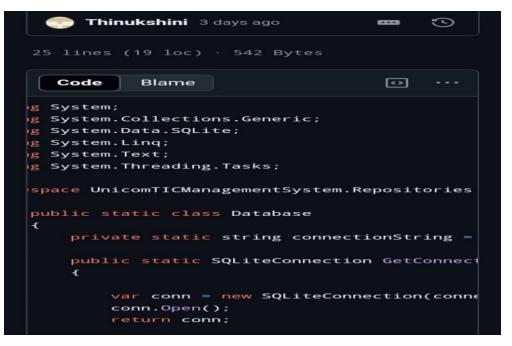
Manage course information

View and update marks

Staff Form

Staff members, upon logging in, can access all system data except adminspecific information. I thought these could serve as important screenshots for my project.

```
oublic static void CreateTable()
   using (var conn = Database.GetConnection(
   -{
       //conn.Open(); // Ensure the connecti
       string createTables = @"
   CREATE TABLE IF NOT EXISTS Users (
       UserID INTEGER PRIMARY KEY AUTOINCREM
       Username TEXT NOT NULL,
       Password TEXT NOT NULL,
       role TEXT
   );
   CREATE TABLE IF NOT EXISTS Courses (
       course_id INTEGER PRIMARY KEY AUTOINC
       course_name TEXT NOT NULL,
       department TEXT
   CREATE TABLE IF NOT EXISTS Subjects (
       subject_id INTEGER PRIMARY KEY AUTOIN
       subject_name TEXT NOT NULL,
       course_id INTEGER,
       FOREIGN KEY(course_id) REFERENCES Cou
   );
   CREATE TABLE IF NOT EXISTS Students (
       Id TEXT PRIMARY KEY,
       Name TEXT NOT NULL,
       NIC_Number TEXT NOT NULL
   );
   CREATE TABLE IF NOT EXISTS Staffs (
       ID TEXT PRIMARY KEY,
       Name TEXT NOT NULL
```



```
vate void Login_Click(object sender, EventArg
string username = UserName.Text.Trim();
string password = Password.Text.Trim();
//Safely handle null SelectedItem by providi
string userType = comboBox1.SelectedItem?.To
Dictionary<string, (string user, string pass
{ "Admin", ("Admin", "admin007") },
{ "Staff", ("Staff", "staff007") },
{ "Student", ("Student", "student007") },
{ "Teacher", ("Teacher", "teacher007") }
if (credentials.ContainsKey(userType))
{
    var (validUsername, validPassword) = cre
    if (username == validUsername && passwor
        MessageBox.Show($"{userType} Login S
        if (userType == "Admin")
             Adminview adminview = new Adminv
             adminview.Show();
         else if (userType == "Staff")
         {
             Staffform staffform = new Stafff
             staffform.Show();
         }
         else if (userType == "Student")
```

```
vate void coursecombobox_SelectedIndexChanged
subjectcombobox.Items.Clear();
switch (coursecombobox.SelectedItem.ToString
{
     case "Python":
         subjectcombobox.Items.AddRange(new s
         lectcombobox.Items.Add("Mr.Kathir");
         break;
     case "C#":
         subjectcombobox.Items.AddRange(new s
         lectcombobox.Items.Add("Mr.Kathir");
         break;
     case "C++":
         subjectcombobox.Items.AddRange(new s
         lectcombobox.Items.Add("Mr.Mayooran"
         break;
     case "Java":
         subjectcombobox.Items.AddRange(new s
         lectcombobox.Items.Add("Dr.Neil Anth
         break;
     case "HTML":
         subjectcombobox.Items.AddRange(new s
         lectcombobox.Items.Add("Ms.S.Dharsik
```

```
nicNumtxt.Text = "";
vate void DataGridView1_SelectionChanged(obj
 if (DataGridView1.SelectedRows.Count > 0)
  {
      var row = DataGridView1.SelectedRows[0]
      Student selectedStudent = (Student)row.
      // if (student != null)
      //if (int.TryParse(student.Id, out int
      // selectedStudentID = student.id;
      stIDtxt.Text = selectedStudent.Id.ToStr
      stnametxt.Text = selectedStudent.Name;
      nicNumtxt.Text = selectedStudent.NIC_Nu
      selectedStudentID = selectedStudent.Id;
 else
  {
      ClearInputs();
      selectedStudentID = "";
  }
ivate void coursebtn_Click(object sender, Eve
 Courseform courseform = new Courseform();
```

4. Conclusion

This login system ensures secure, role-based access control for different users. Each role is granted specific permissions tailored to their responsibilities, allowing smooth and organized system management.

Thank you