Windows Forms Application – Project Report

Project Title: School Management System

Submitted by: Mohamed Fazal **UT010621**

Introduction

This project is a basic Windows Forms Application created using C# and Visual Studio. It is designed to perform simple tasks like [adding records, saving data, displaying information, etc.].
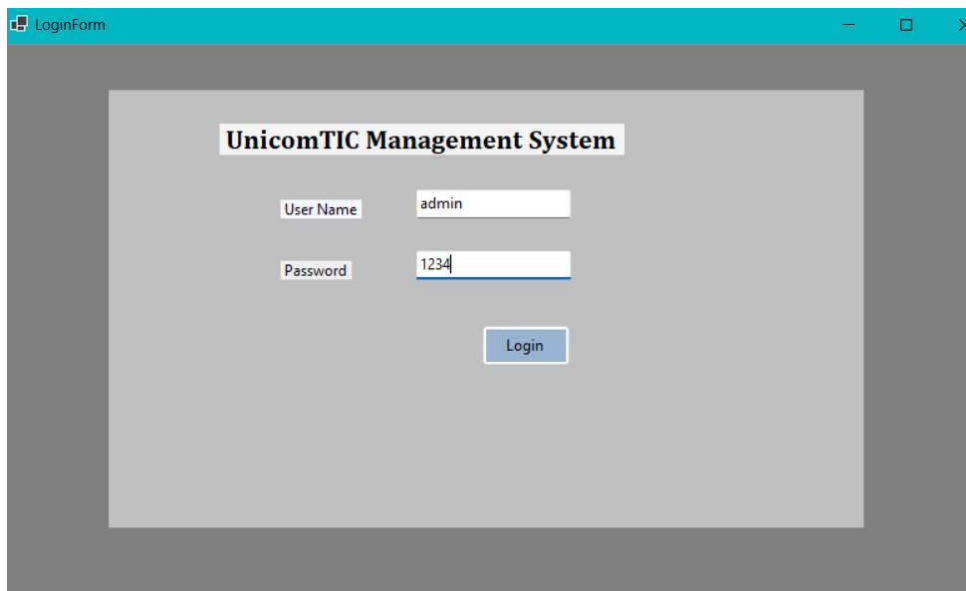
Tools & Technologies Used

Tool	Description
Visual Studio	IDE used to develop the project
C#	Programming Language
.NET Framework	Platform for building the app
Windows Forms	For designing the user interface

Screenshots

Login Form

Hard-coded value (U. Name, Password) **=> To Login First**

Course Adding Form

Subject adding form

Staffs adding form

Lecture adding form

Student adding form

Room creation form

Timetable Form

Exams Form

Marks Form

User Password Setting Form
(Admin, Staff, Lecture, Student)

Code Samples

```csharp
public static List<Lecture> GetAllLectures()
{
    List<Lecture> lectures = new List<Lecture>();

    using (var cmd = DataCon.GetConnection().CreateCommand())
    {
        cmd.CommandText = @"
            SELECT l.LectureID, l.Name, l.SubjectID, l.Qualification, l.Email
            FROM Lectures l";

        using (var reader = cmd.ExecuteReader())
        {
            while (reader.Read())
            {
                lectures.Add(new Lecture
                {
                    LectureID = Convert.ToInt32(reader["LectureID"]),
                    Name = reader["Name"].ToString(),
                    SubjectID = Convert.ToInt32(reader["SubjectID"]),
                    Qualification = reader["Qualification"].ToString(),
                    Email = reader["Email"].ToString()
                });
```

```csharp
1 reference | Mohamed Fazal, 7 hours ago | 1 author, 1 change
public static void AddLecture(string name, int subjectId, string qualification, string email)
{
    using (var cmd = DataCon.GetConnection().CreateCommand())
    {
        cmd.CommandText = "INSERT INTO Lectures (Name, SubjectID, Qualification, Email) VALUES (@n, @s, @q, @e)";
        cmd.Parameters.AddWithValue("@n", name);
        cmd.Parameters.AddWithValue("@s", subjectId);
        cmd.Parameters.AddWithValue("@q", qualification);
        cmd.Parameters.AddWithValue("@e", email);
        cmd.ExecuteNonQuery();
    }
}
```

```csharp
1 reference | Mohamed Fazal, 7 hours ago | 1 author, 1 change
public static void UpdateLecture(int id, string name, int subjectId, string qualification, string email)
{
    using (var cmd = DataCon.GetConnection().CreateCommand())
    {
        cmd.CommandText = "UPDATE Lectures SET Name = @n, SubjectID = @s, Qualification = @q, Email = @e WHERE LectureID = @id";
        cmd.Parameters.AddWithValue("@n", name);
        cmd.Parameters.AddWithValue("@s", subjectId);
        cmd.Parameters.AddWithValue("@q", qualification);
        cmd.Parameters.AddWithValue("@e", email);
        cmd.Parameters.AddWithValue("@id", id);
        cmd.ExecuteNonQuery();
    }
}
```

```csharp
1 reference | UT010621, 4 days ago | 1 author, 1 change
public static void DeleteLecture(int id)
{
    using (var cmd = DataCon.GetConnection().CreateCommand())
    {
        cmd.CommandText = "DELETE FROM Lectures WHERE LectureID = @id";
        cmd.Parameters.AddWithValue("@id", id);
        cmd.ExecuteNonQuery();
    }
}
}
```

```csharp
private void MainForm_Load(object sender, EventArgs e)
{
    lblWelcom.Text = $"Welcome {loggedInUser.Username} ({loggedInUser.Role})";

    if (loggedInUser.Role == "Admin")
    {
        EnableAll();
    }
    else if (loggedInUser.Role == "Staff")
    {
        btnUser.Visible = false;
        btnLecture.Visible = false;
    }
    else if (loggedInUser.Role == "Lecturer")
    {
        btnUser.Visible = false;
        btnCourse.Visible = false;
        btnStudent.Visible = false;
    }
    else if (loggedInUser.Role == "Student")
    {
        btnUser.Visible = false;
        btnStudent.Visible = false;
        btnLecture.Visible = false;
        btnMark.Visible = false;
    }
}
```

```csharp
private async void btnLogin_Click(object sender, EventArgs e)
{
    string username = txtUsername.Text.Trim();
    string password = txtPassword.Text.Trim();

    var user = await LoginController.LoginUserAsync(username, password);

    if (user != null)
    {
        this.Hide();
        new MainForm(user).Show();
    }
    else
    {
        MessageBox.Show("Invalid username or password.");
    }
}
```