**EduLity**
**Elevate Education, Embrace Quality**

# REPORT BOOK

## Tugas Akhir Pemrograman Berorientasi Obyek

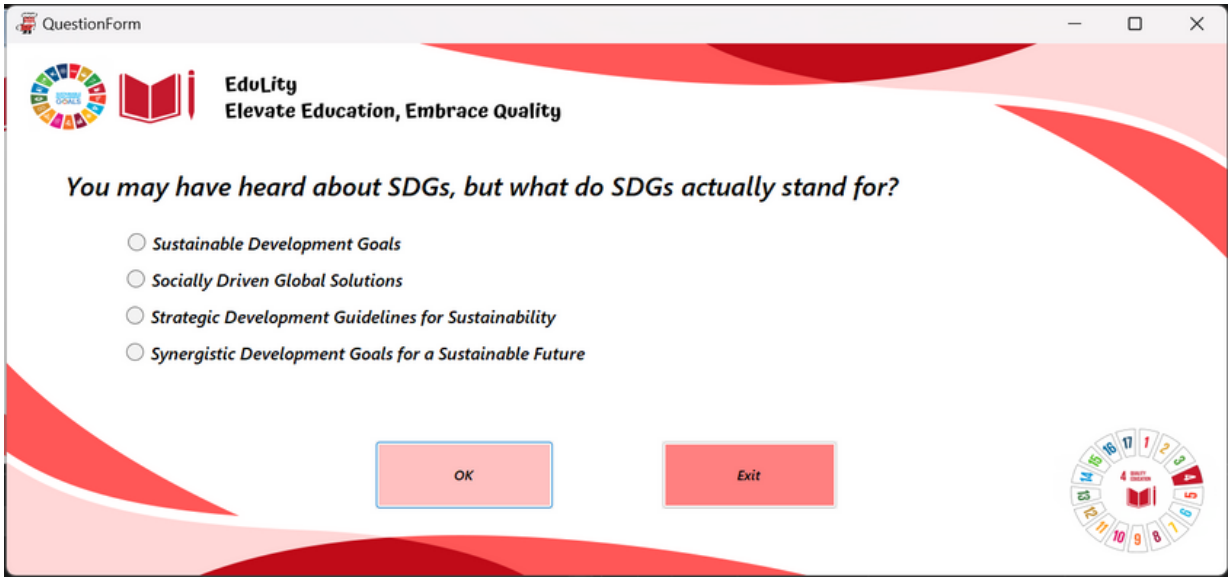
**Program Studi Teknologi Informasi 2023**

Kelompok **ASTERIA**
Barbara Neanake A. (22/494495/TK/54238)
Christella Jesslyn D. (22/493149/TK/54003)

# Apa itu EduLity?

EduLity adalah aplikasi berbasis C# Windows Forms yang memberikan pengalaman pembelajaran inovatif dengan fokus pada Pendidikan Berkelanjutan (SDGs ke-4). Aplikasi ini dibentuk sebagai sarana peningkatan pengetahuan yang memberikan kesempatan pada pengguna untuk belajar melalui serangkaian kuis yang tidak hanya informatif tetapi juga menghibur.

EduLity, dikembangkan dengan manajemen data kuat melalui PostgreSQL, dirancang cermat dengan Microsoft Form dan diimplementasikan lancar melalui Visual Studio 2022. Dengan bahasa pemrograman C#, kami memastikan fungsionalitas efisien dan manajemen database tanpa hambatan. Sebagai representasi perpaduan teknologi dan pendidikan, EduLity bertujuan memfasilitasi lingkungan pembelajaran dinamis dan interaktif.



# Tujuan Hadirnya EduLity

## Empowering Quality Education Through Adaptive Quizzes

EduLity berkomitmen mendukung SDG 4, Pendidikan Berkualitas, melalui kuis adaptif yang informatif. Dengan fitur uniknya, EduLity bertujuan meningkatkan pemahaman pengguna terhadap tantangan dan peluang pendidikan, menjadikannya kontributor kunci dalam mencapai SDG4.


## Igniting Profound Awareness of SDGs

EduLity tidak hanya menyajikan kuis menghibur; tetapi juga menggali pemahaman tentang SDGs, khususnya SDG 4. EduLity mengajak pengguna berpartisipasi aktif & mendukung inisiatif peningkatan mutu pendidikan. EduLity bukan sekadar penyedia quiz, tapi kami penggerak perubahan.


## Innovative Program for OOP Final Project

Kami menciptakan EduLity untuk memenuhi tugas akhir Pemrograman Berorientasi Objek. Dengan pendekatan adaptif, desain kode yang unik, dan teknologi terkini, EduLity adalah jalan kami bereksplorasi di pemrograman berorientasi objek bersama pembelajaran global yang dinamis.

# Fitur Unggulan EduLity

## Quiz Berkualitas mengenai SDGs

- Kuis EduLity menawarkan pertanyaan berkualitas tinggi yang menyentuh inti Pendidikan Berkelanjutan (SDGs ke-4).
- Setiap pertanyaan didesain untuk merangsang pemikiran mendalam tentang isu-isu terkait SDGs.

## Feedback Interaktif

- Umpan balik instan dalam setiap pertanyaan, sehingga dapat meningkatkan pengalaman belajar.
- Memberikan motivasi dan bimbingan, mendorong pemahaman mendalam.

## Tampilan Rapi dan Menarik

- *Interface* yang rapi dan menarik untuk pengalaman pembelajaran yang menyenangkan.
- Desain intuitif untuk mempermudah navigasi dan memastikan pengguna dapat dengan mudah mengakses fitur pembelajaran.

## Recap, Scoring, & Explanation

- Fitur recap memberikan gambaran hasil kuis secara menyeluruh.
- Menampilkan peringkat pengguna, menambah elemen kompetitif.

## Kode Terstruktur & Mudah Dimodifikasi

- EduLity memprioritaskan kode yang terstruktur dan fleksibel, memudahkan pengembang dalam modifikasi tanpa mengurangi stabilitas aplikasi.

PROGRAMMING JOBDESC

# Asteria Group

## Tim Dibalik Lahirnya EduLity

**BARBARA** Neanake Ajiesti
22/494495/TK/54238

**1. Creating UI/UX Design**
- Crafting the complete UI/UX design for EduLity, ensuring a user-friendly experience.

**2. Managing Database**
- Overseeing the database operations using PostgreSQL for efficient data management.

**3. Refactoring with Techniques**
- Enhancing EduLity's codebase with advanced refactoring methods for improved structure.

**4. Implementing Design Patterns**
- Incorporating design patterns like Singleton, Observer, and Strategy for optimized functionality.

**5. Developing Coding Entire Application**
- (since the application is only available on Barbara's laptop). Creating FirstForm, RegistForm, LoginForm, MainForm, RecapForm, and ExplanationForm.

**6. Creating User Control Elements**
- Designing and implementing User Control elements for enhanced application behavior (sidebar, etc).

**7. Creating Application Report**
- Generating a comprehensive report with use cases, flowcharts, and detailed explanations.

**8. Creating Presentation Video**
- Crafting an engaging video showcasing EduLity's features and design approach.

**9. Researching EduLity Question & Explanation**
- Thoroughly researching and addressing questions to deepen understanding.

# Asteria Group

## Tim dibalik lahirnya EduLity

Christella **JESSLYN** Dewantara
22/493149/TK/54003

1. **Coding Quiz Functionality (QuestionForm)**
    - Developing the codebase for EduLity's quiz module, focusing on the QuestionForm to enable seamless quiz execution.
2. **Creating Group Name**
    - Generating meaningful group names, like "Asteria," inspired by themes or meanings relevant to each group's identity.
3. **Creating Application Report**
    - Creating a concise report detailing EduLity's functionalities and technical aspects for user understanding and developer insights.

# EduLity

# Use Case Diagram

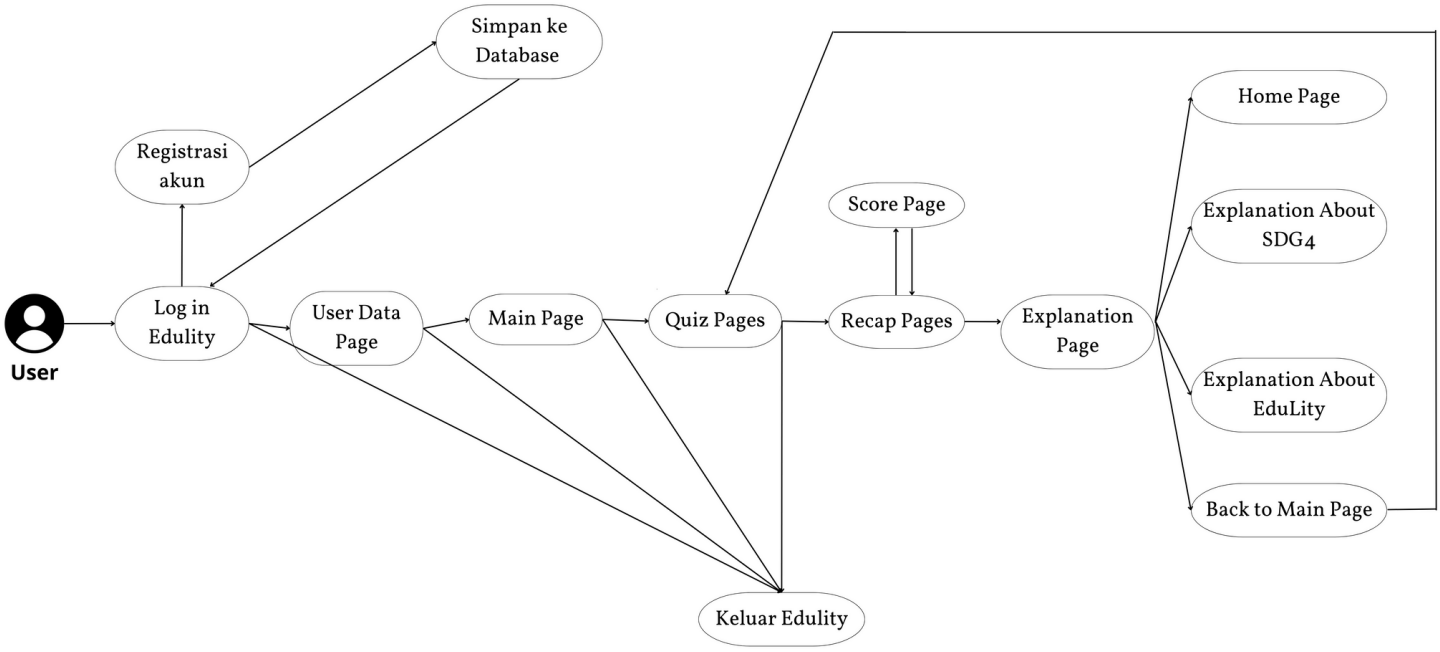
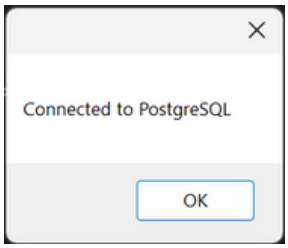


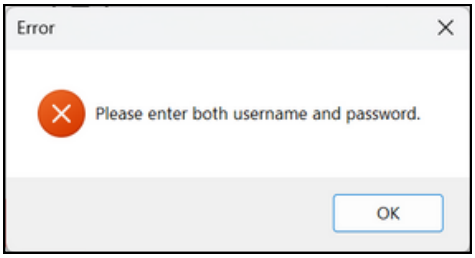

Diagram use case di atas menampilkan hubungan antara FirstForm (Login), RegistForm (Registrasi akun), LoginForm (User Data Page), MainForm (Main Page), QuestionForm (Quiz Pages), RecapForm (Recap Page), hingga ExplanationForm (Explanation Page), dan hubungan mereka dengan database. Diagram ini mencakup hubungan fungsional antar elemen-elemen tersebut dan kaitannya dengan basis data.

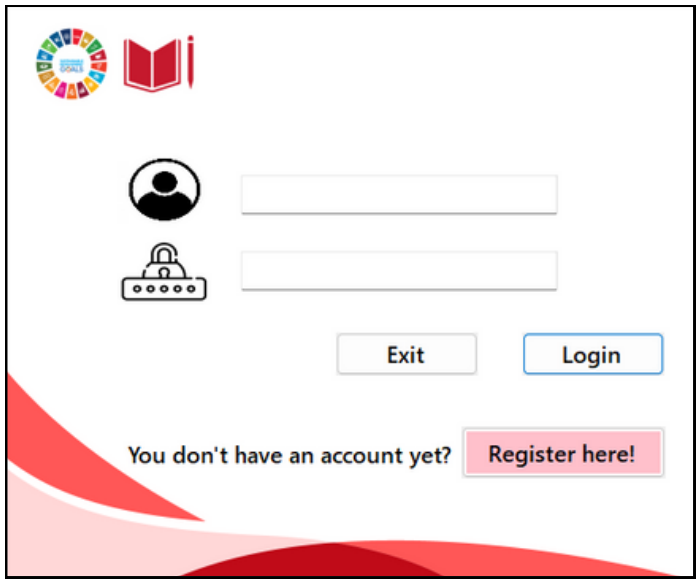
# EduLity

# User Scenario

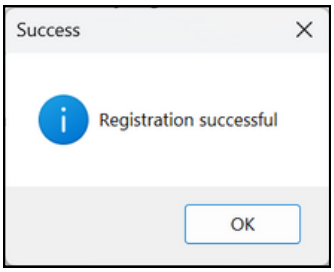
1. Ketika code dijalankan, program menampilkan pop up konfirmasi telah terhubung ke PostgreSQL dan user dapat mengakses FirstForm untuk login dengan memasukkan username serta password.

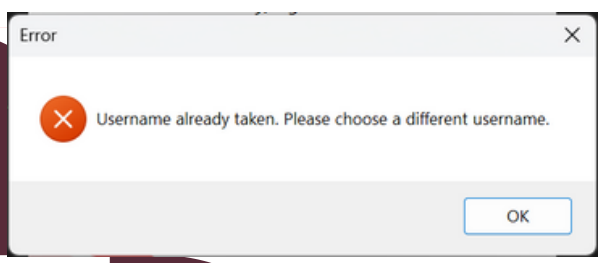


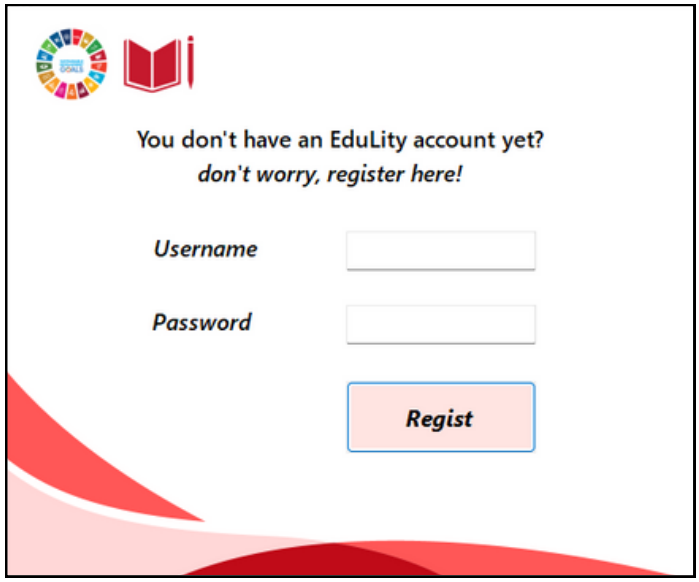




2. Jika belum memiliki akun, user dapat melakukan registrasi melalui RegistForm dan kembali login.



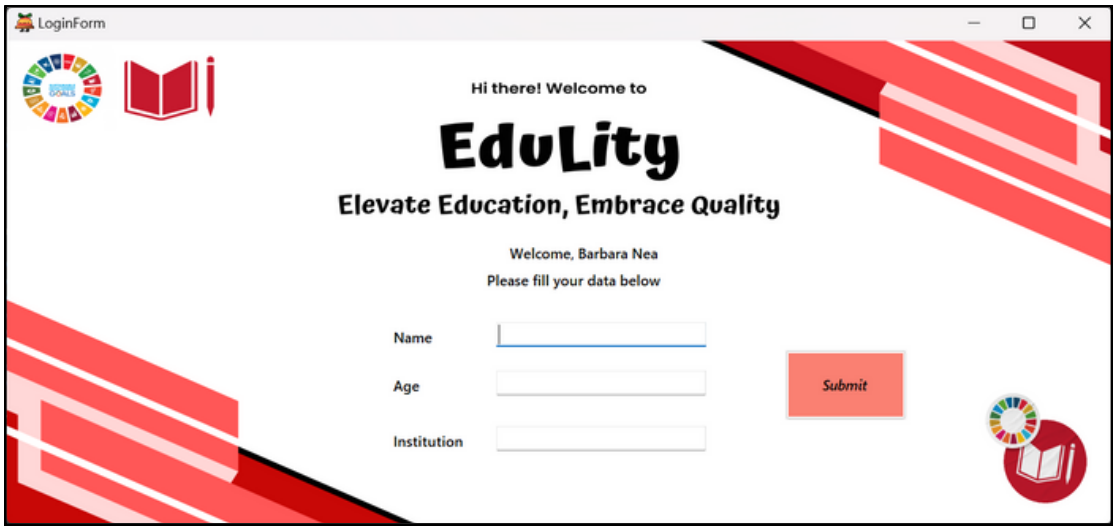



# EduLity

# User Scenario

3. Setelah berhasil login, user diarahkan ke LoginForm untuk mengisi data pribadi seperti nama, usia, dan instansi.

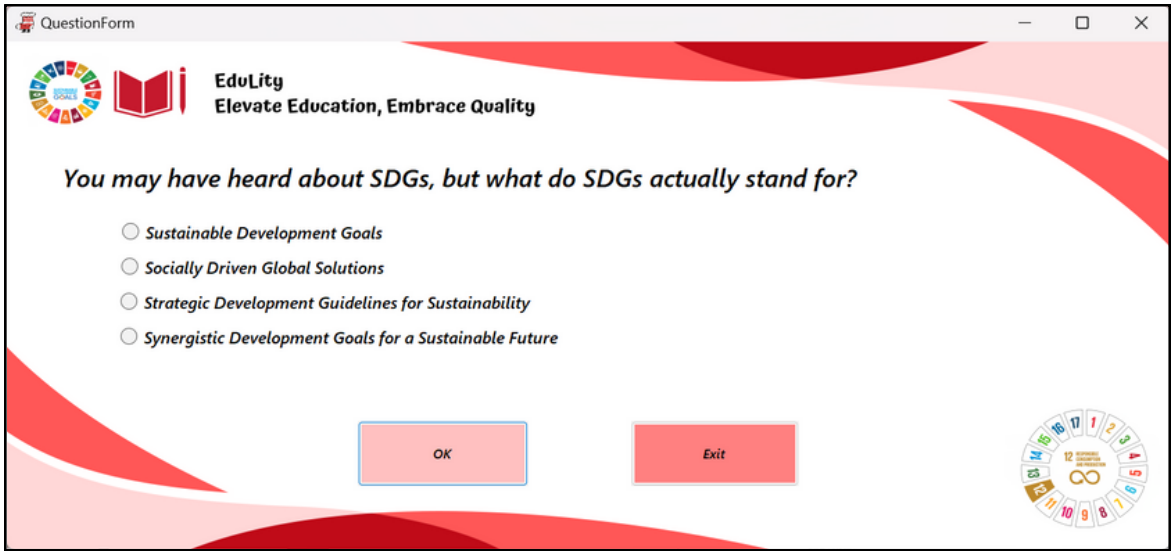


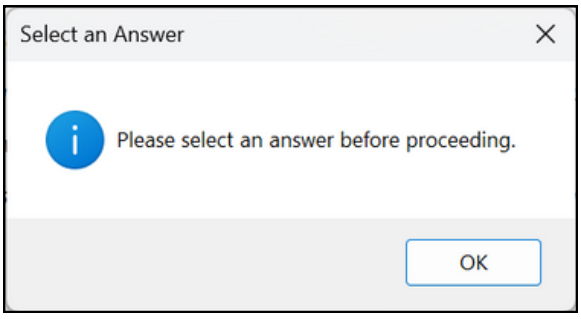

4. Saat menekan submit, MainForm muncul, meminta user untuk memastikan kesiapan mengikuti kuis.

# EduLity

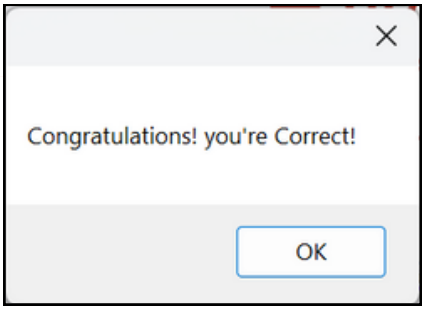
# User Scenario

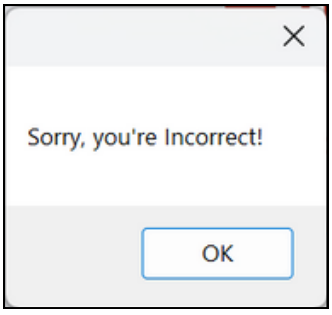
5. Jika siap, user dapat menekan "Ready" untuk beralih ke QuestionForm dan menjawab pertanyaan kuis.




6. Feedback pop-up akan memberikan informasi setiap kali user memberikan jawaban.

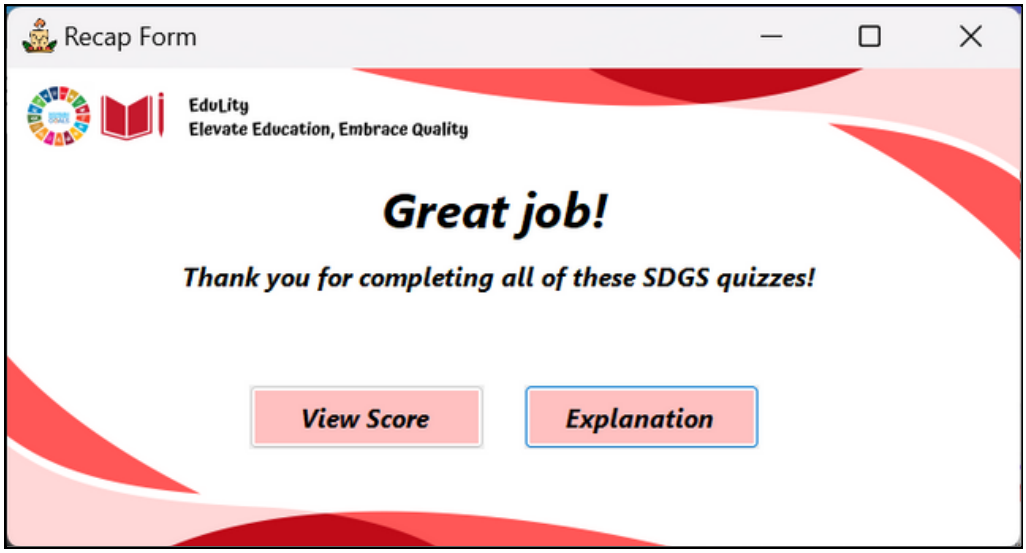


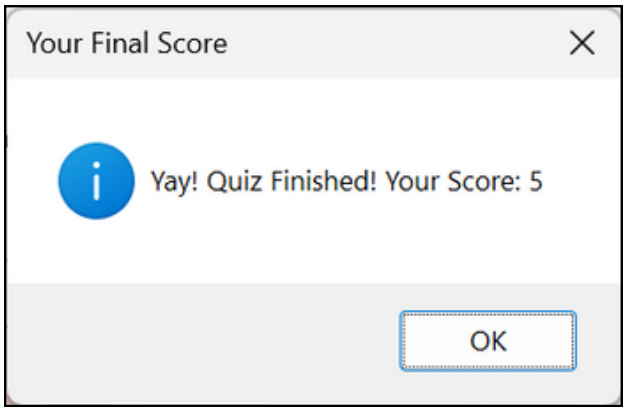
# EduLity

# User Scenario

7. Setelah menyelesaikan QuestionForm, user otomatis dialihkan ke RecapForm untuk melihat skor quiz. Dari RecapForm, user dapat melihat skor akhir dengan menekan "View Score" dan membuka penjelasan dalam “Explanation”.



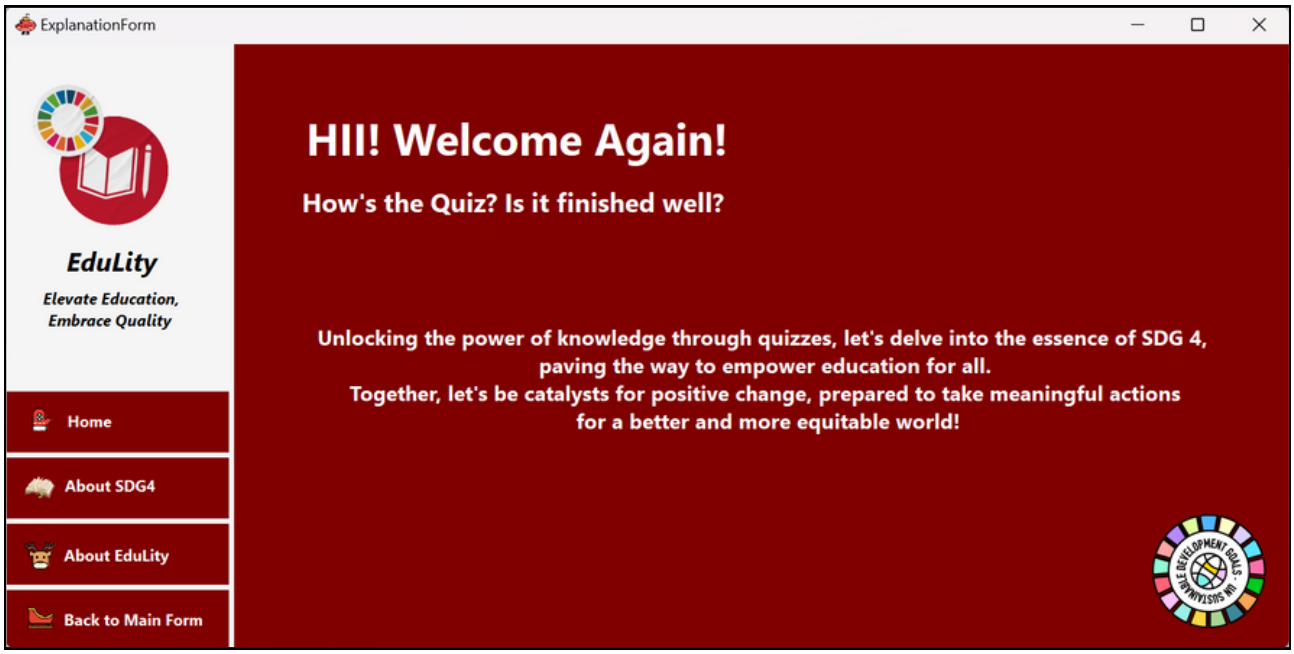

.



# EduLity

# User Scenario

8. ExplanationForm menyajikan sidebar dengan opsi untuk menjelajahi berbagai fitur seperti halaman utama, penjelasan SDG 4, penjelasan Edulity, dan opsi Quit untuk kembali ke MainForm.



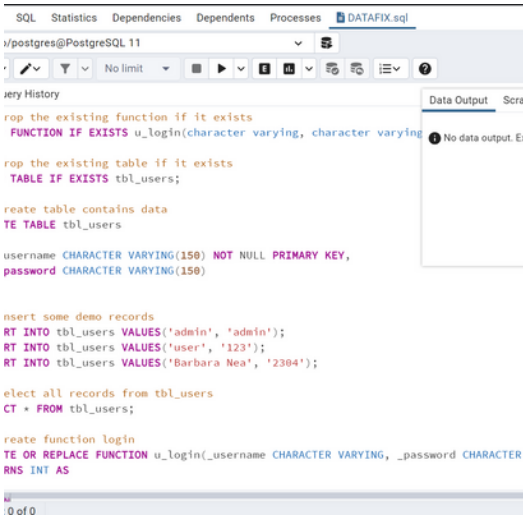

9. Panel "Back to Main Form" pada ExplanationForm memungkinkan user untuk kembali ke MainForm, dan user dapat **keluar dari aplikasi** atau **mengulang kuis.**

# 3 Concept OOP

## that we apply in EduLity

## 1 DATABASE

EduLity menggunakan PostgreSQL untuk menyimpan data pengguna, seperti username dan password. Struktur database melibatkan tabel pengguna, fungsi login, dan registrasi.

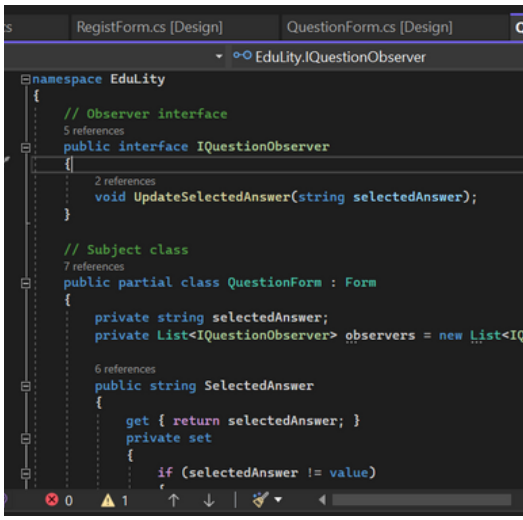



## 2 REFACTORING

Kode EduLity disempurnakan dengan ekstraksi metode, properti, dan manajemen exception. Pemeliharaan dan kejelasan kode ditingkatkan melalui prinsip OOP.

## 3 DESIGN PATTERNS

Dalam EduLity, Design Pattern Observer digunakan pada QuestionForm untuk memisahkan subjek (form pertanyaan) dan pengamat (contoh pengamat), meningkatkan fleksibilitas dan pemeliharaan kode.



# 1 Database using PostgreSQL

## Implementasi Database pada EduLity menggunakan PostgreSQL

Dalam aplikasi EduLity, penggunaan database PostgreSQL diimplementasikan untuk menyimpan data pengguna, terutama informasi seperti username dan password. Berikut adalah penjelasan teknis mengenai sistem database PostgreSQL yang diimplementasikan:

### Struktur Tabel (tbl_users)

```
7   -- Create table contains data
8   CREATE TABLE tbl_users
9   (
10      username CHARACTER VARYING(150) NOT NULL PRIMARY KEY,
11      password CHARACTER VARYING(150)
12  );
13
```

- username: kolom untuk menyimpan nama pengguna. Dijadikan PRIMARY KEY untuk memastikan keunikan setiap pengguna.
- password: kolom untuk menyimpan kata sandi / password pengguna.

### Fungsi Login (u_login)

```
22  -- Create function login
23  CREATE OR REPLACE FUNCTION u_login(_username CHARACTER VARYING, _password CHARACTER VARYING)
24  RETURNS INT AS
25  $$
26  BEGIN
27      RETURN EXISTS (SELECT 1 FROM tbl_users WHERE username = _username AND password = _password)::INT;
28  END
29  $$
30  LANGUAGE plpgsql;
31
```

- Fungsi: Digunakan untuk memeriksa apakah kombinasi username dan password yang dimasukkan oleh pengguna valid.
- Return Value: Mengembalikan 1 jika login berhasil dan 0 jika gagal.

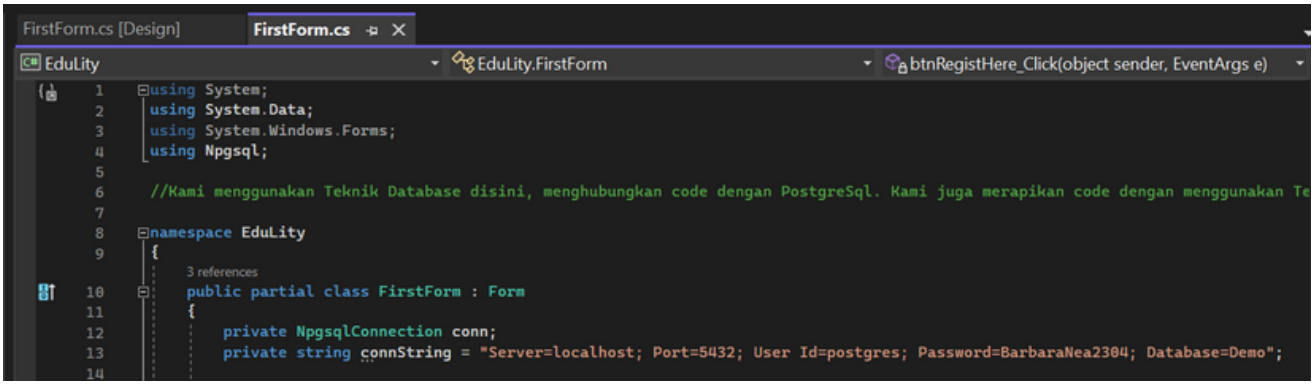
## Fungsi Registrasi (u_register)

```
-- Create function for user registration
CREATE OR REPLACE FUNCTION u_register(_username CHARACTER VARYING, _password CHARACTER VARYING)
RETURNS VOID AS
$$
BEGIN
    -- Check if the username already exists
    IF NOT EXISTS (SELECT 1 FROM tbl_users WHERE username = _username) THEN
        -- If username is available, insert the new user
        INSERT INTO tbl_users(username, password) VALUES (_username, _password);
    END IF;
END
$$
LANGUAGE plpgsql;

```

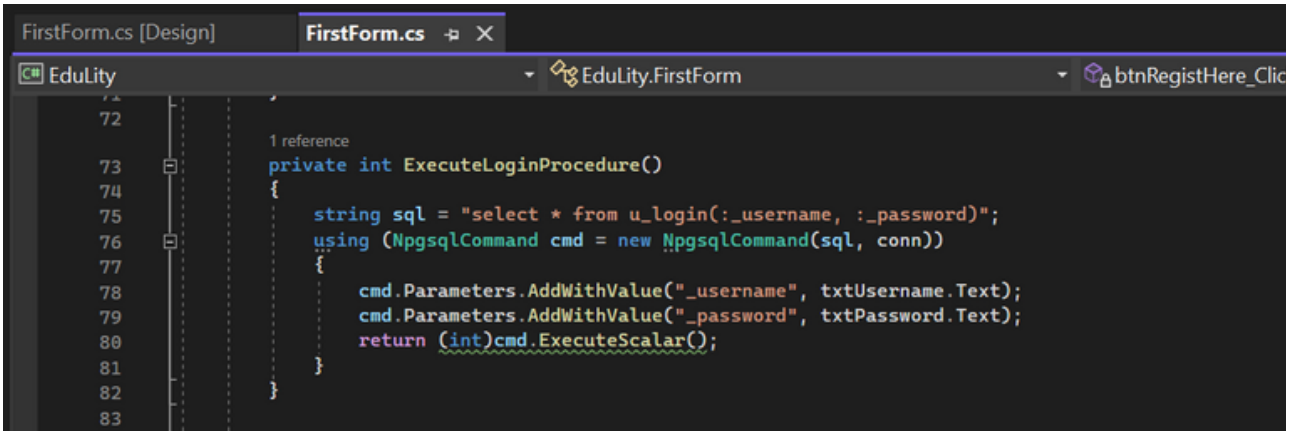
- Fungsi: Digunakan untuk mendaftarkan pengguna baru ke dalam tabel tbl_users.
- Proses: Memeriksa apakah username sudah terdaftar. Jika belum, melakukan penyisipan data baru.

## Koneksi ke Database dalam FirstForm



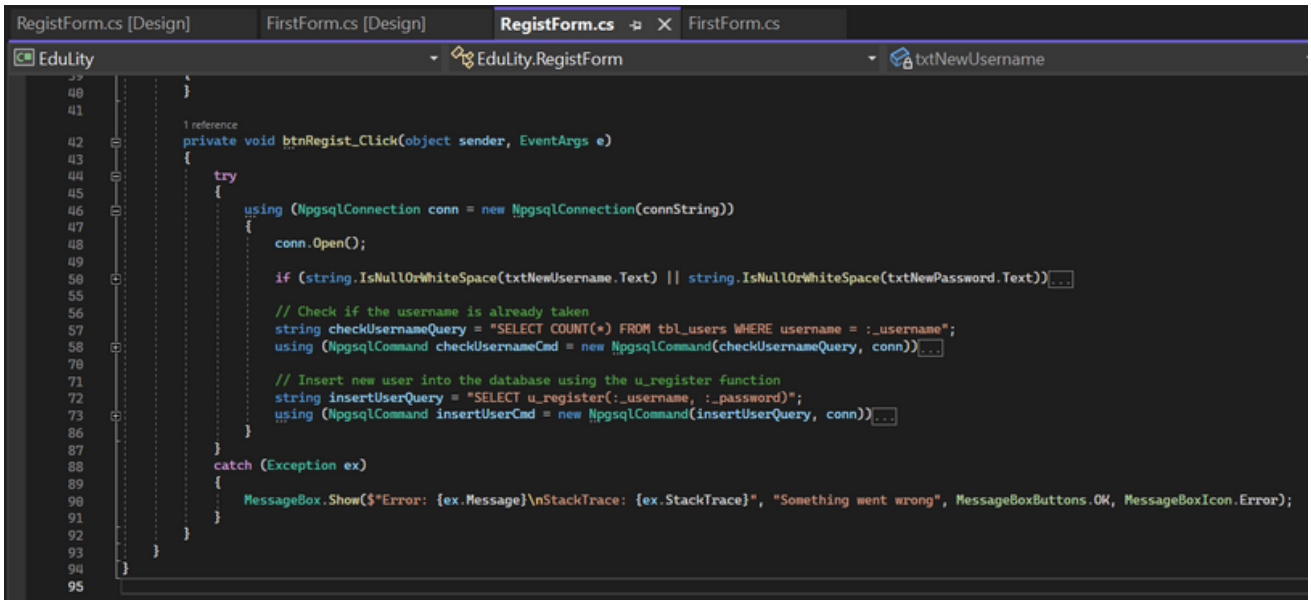

- Inisialisasi Koneksi: Membuat koneksi ke database PostgreSQL dengan parameter seperti server, port, user, password, dan nama database.

## Login Procedure dalam FirstForm




- Eksekusi Login Procedure: Memanggil fungsi login pada database untuk memeriksa kevalidan login.

## Registrasi dalam RegistForm




- Registrasi Pengguna Baru: Memastikan koneksi dibuka sebelum melakukan proses registrasi pengguna baru menggunakan fungsi u_register di database.

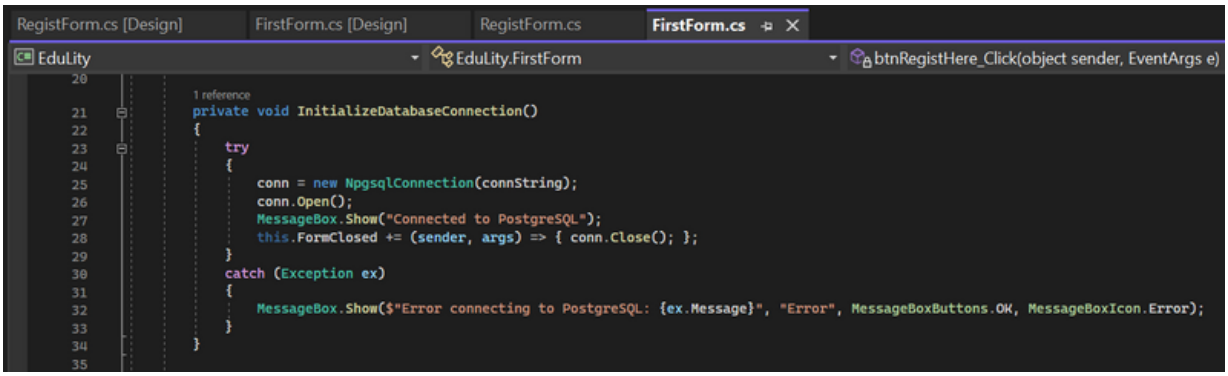
## Kesimpulan Implementasi Database dalam EduLity

- Implementasi ini memastikan bahwa data pengguna tersimpan dengan aman dalam tabel PostgreSQL dan proses login/registrasi dapat dilakukan dengan baik melalui fungsi yang telah dibuat di database.

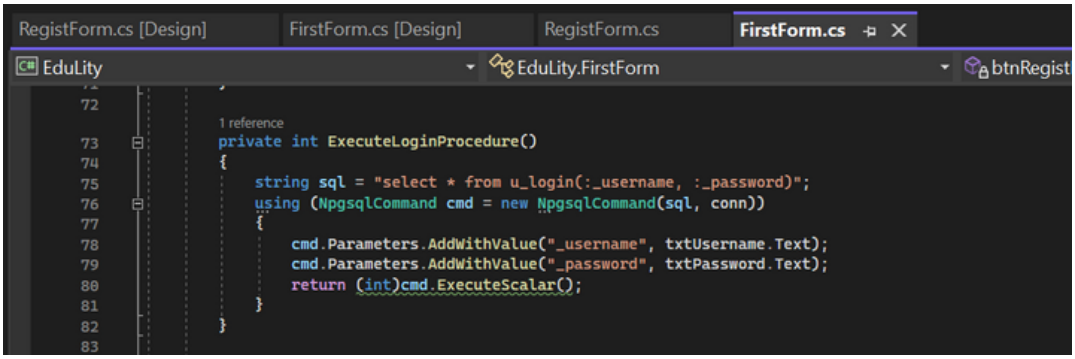
# Refactoring Implementations 2

## Refactoring dalam FirstForm

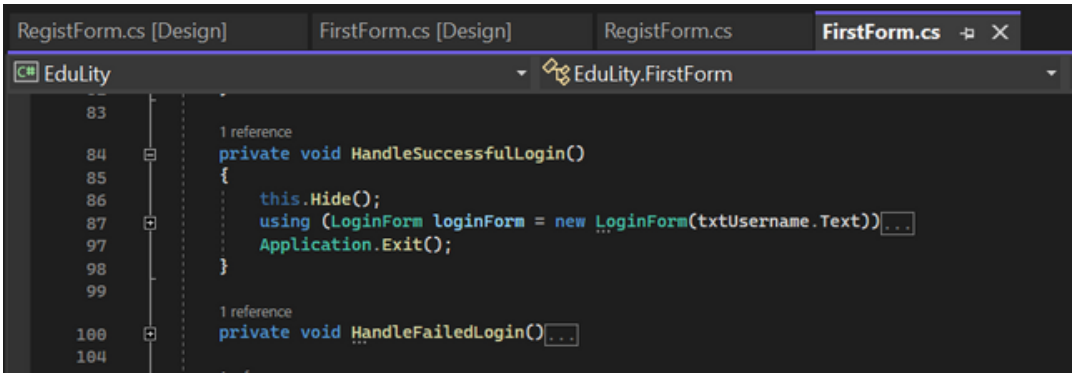
**Manajemen Koneksi Database:**
Meningkatkan keterbacaan kode dengan mengenkapsulasi pengaturan dan penutupan koneksi database dalam metode terpisah (**InitializeDatabaseConnection**).




**Eksekusi Login:** Mengekstrak logika eksekusi login ke dalam metode terpisah (**ExecuteLoginProcedure**) untuk kemudahan pemeliharaan.

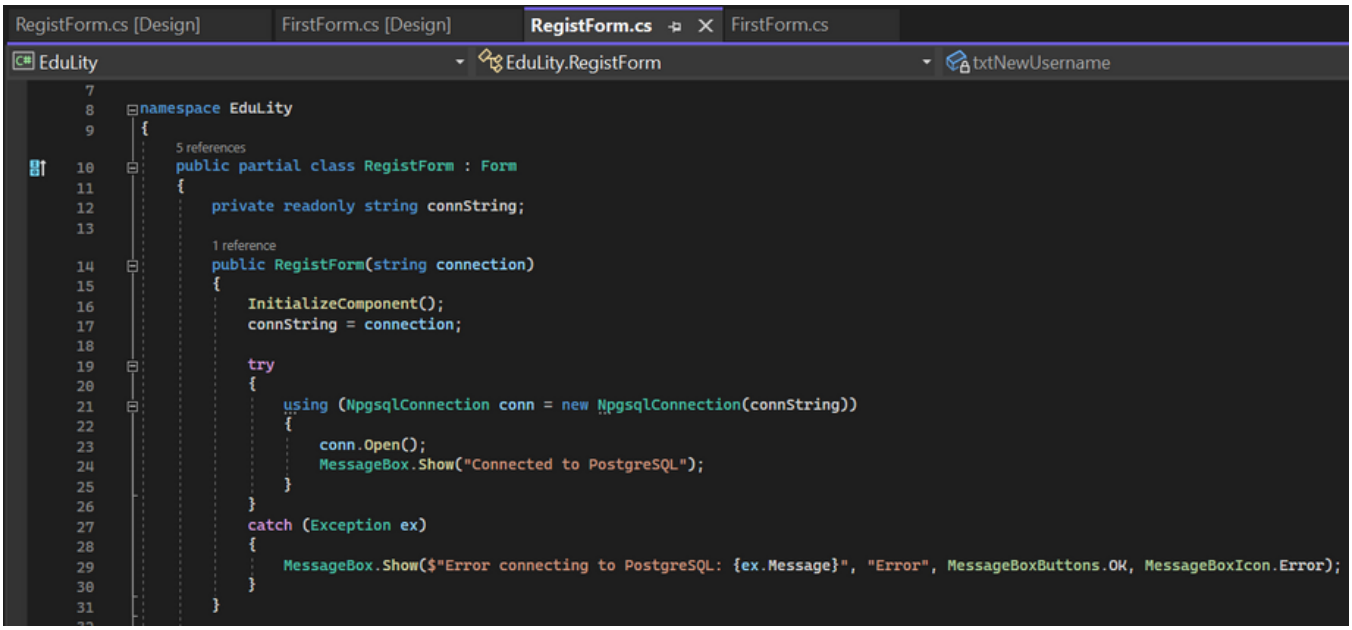



**Penanganan Login Sukses dan Gagal:** Modularisasi penanganan login sukses dan gagal ke dalam metode terpisah (**HandleSuccessfulLogin** dan **HandleFailedLogin**).



## Refactoring dalam RegistForm

**Penanganan Koneksi Database:** Meningkatkan struktur kode dengan menggunakan pernyataan **using** untuk setup koneksi database, mengurangi redundansi, dan meningkatkan keterbacaan.

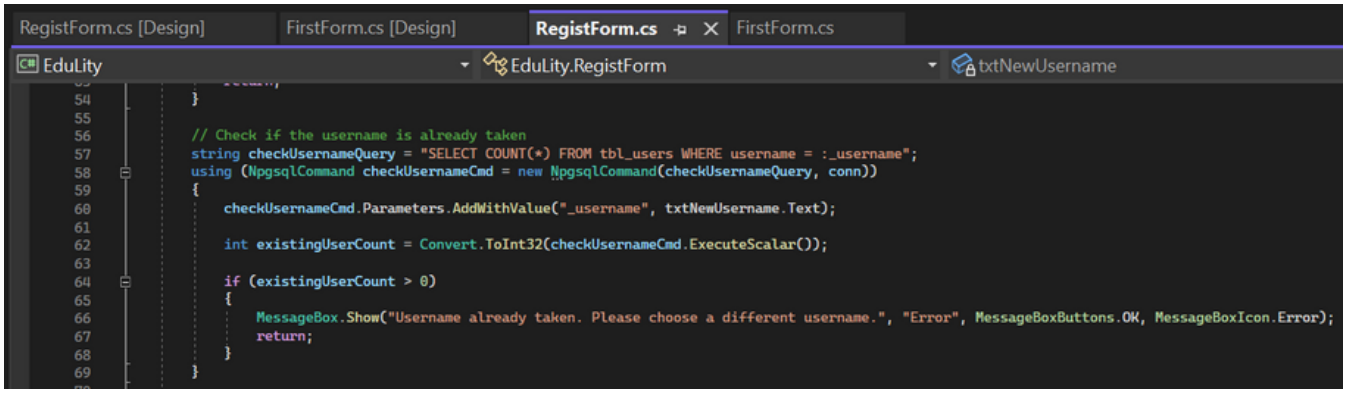



```
namespace EduLity
{
    5 references
    public partial class RegistForm : Form
    {
        private readonly string connString;

        1 reference
        public RegistForm(string connection)
        {
            InitializeComponent();
            connString = connection;

            try
            {
                using (NpgsqlConnection conn = new NpgsqlConnection(connString))
                {
                    conn.Open();
                    MessageBox.Show("Connected to PostgreSQL");
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show($"Error connecting to PostgreSQL: {ex.Message}", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
            }
        }
```

**Validasi Username:** Mengekstrak logika validasi username ke dalam query terpisah untuk meningkatkan keterbacaan.



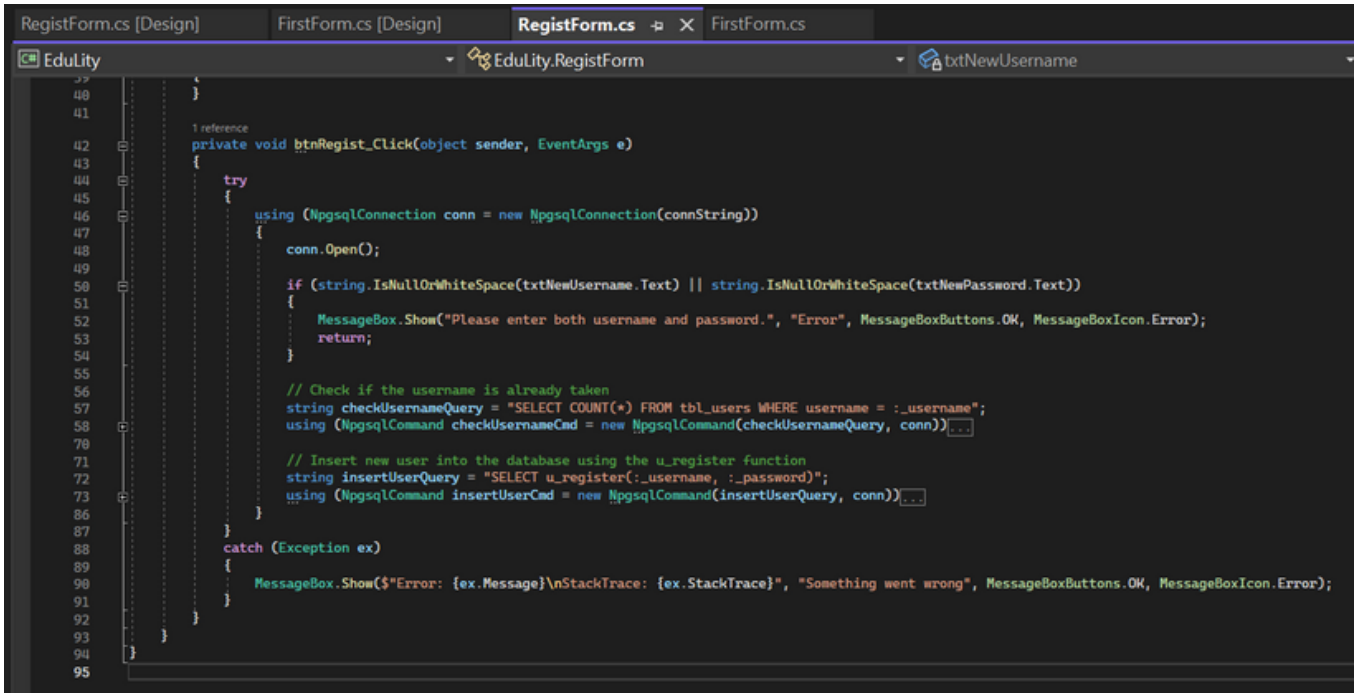

```
        }

        // Check if the username is already taken
        string checkUsernameQuery = "SELECT COUNT(*) FROM tbl_users WHERE username = :_username";
        using (NpgsqlCommand checkUsernameCmd = new NpgsqlCommand(checkUsernameQuery, conn))
        {
            checkUsernameCmd.Parameters.AddWithValue("_username", txtNewUsername.Text);

            int existingUserCount = Convert.ToInt32(checkUsernameCmd.ExecuteScalar());

            if (existingUserCount > 0)
            {
                MessageBox.Show("Username already taken. Please choose a different username.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
                return;
            }
        }
```

**Logika Registrasi Pengguna:** Modularisasi logika registrasi pengguna ke dalam metode terpisah (**btnRegist_Click**) untuk organisasi kode yang lebih baik.



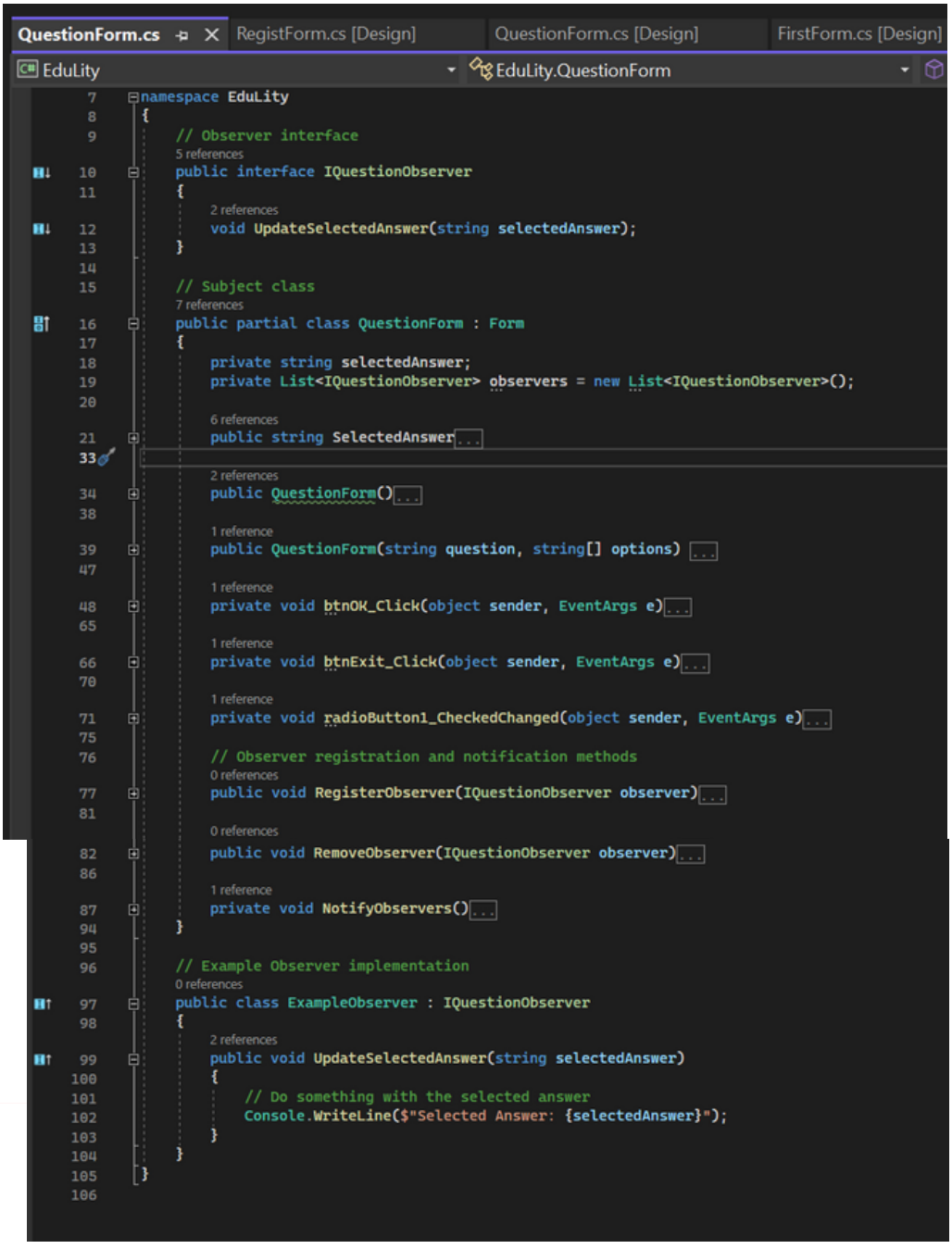

## Kesimpulan Implementasi Refactoring dalam EduLity

Refactoring pada FirstForm dan RegistForm di EduLity membawa sejumlah perbaikan signifikan. Refactoring pada FirstForm meningkatkan penanganan kesalahan koneksi database, sementara di RegistForm membawa perbaikan dalam manajemen sumber daya dengan penggunaan blok "using". Keduanya menghasilkan kode yang lebih bersih, mudah dibaca, dan mematuhi prinsip-prinsip desain.

# 3 Design Patterns Implementations

## Design Pattern pada QuestionForm: Observer

Pola Observer: Diterapkan untuk meningkatkan fleksibilitas dan keterbacaan dengan memisahkan subjek (QuestionForm) dan observer (ExampleObserver).



```
QuestionForm.cs | RegistForm.cs [Design] | QuestionForm.cs [Design] | FirstForm.cs [Design]
EduLity | EduLity.QuestionForm

namespace EduLity
{
    // Observer interface
    5 references
    public interface IQuestionObserver
    {
        2 references
        void UpdateSelectedAnswer(string selectedAnswer);
    }

    // Subject class
    7 references
    public partial class QuestionForm : Form
    {
        private string selectedAnswer;
        private List<IQuestionObserver> observers = new List<IQuestionObserver>();

        6 references
        public string SelectedAnswer...

        2 references
        public QuestionForm()...

        1 reference
        public QuestionForm(string question, string[] options) ...

        1 reference
        private void btnOK_Click(object sender, EventArgs e)...

        1 reference
        private void btnExit_Click(object sender, EventArgs e)...

        1 reference
        private void radioButton1_CheckedChanged(object sender, EventArgs e)...

        // Observer registration and notification methods
        0 references
        public void RegisterObserver(IQuestionObserver observer)...

        0 references
        public void RemoveObserver(IQuestionObserver observer)...

        1 reference
        private void NotifyObservers()...
    }

    // Example Observer implementation
    0 references
    public class ExampleObserver : IQuestionObserver
    {
        2 references
        public void UpdateSelectedAnswer(string selectedAnswer)
        {
            // Do something with the selected answer
            Console.WriteLine($"Selected Answer: {selectedAnswer}");
        }
    }
}
```

## Design Pattern pada QuestionForm: Observer

Subjek dan Pengamat: Subjek (QuestionForm) memberikan notifikasi kepada pengamat (ExampleObserver) setiap kali ada perubahan pada jawaban yang dipilih.

Registrasi dan Pemberitahuan: Metode RegisterObserver, RemoveObserver, dan NotifyObservers memungkinkan dinamika pengelolaan pengamat.

Pengamatan Contoh: Melalui implementasi ExampleObserver, kita dapat menyesuaikan tindakan yang diambil setelah pembaruan jawaban terpilih.

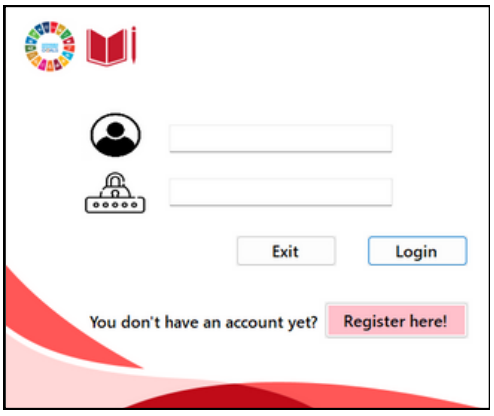
## Kesimpulan Implementasi Design Pattern dalam QuestionForm EduLity

Penerapan pola observer ini memberikan kemampuan untuk memperluas fungsionalitas dalam menanggapi perubahan pada subjek tanpa mengubah struktur subjek itu sendiri. Misalnya, kita dapat dengan mudah menambahkan pengamat tambahan dengan menerapkan antarmuka IQuestionObserver dan mendaftarkannya pada QuestionForm.

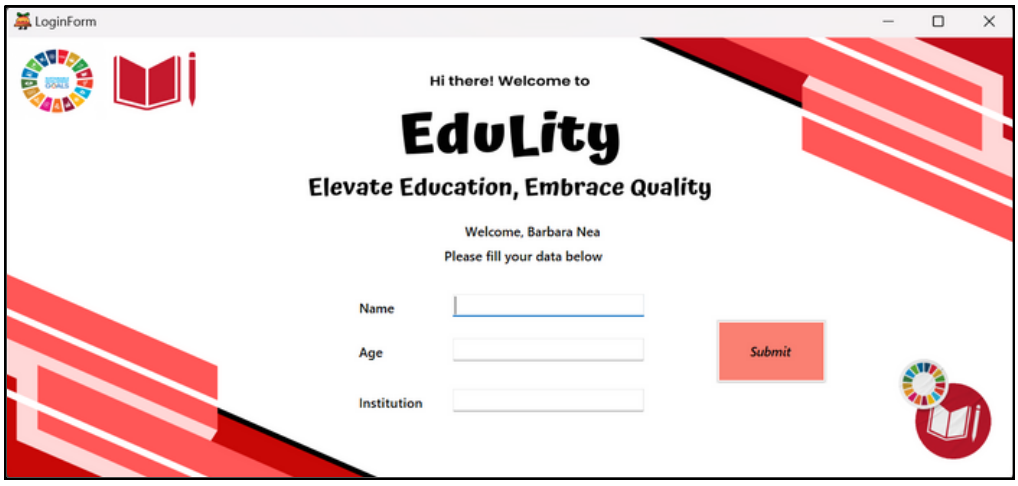
# UI/UX DESIGN

## Design & Elements that we use in EduLity

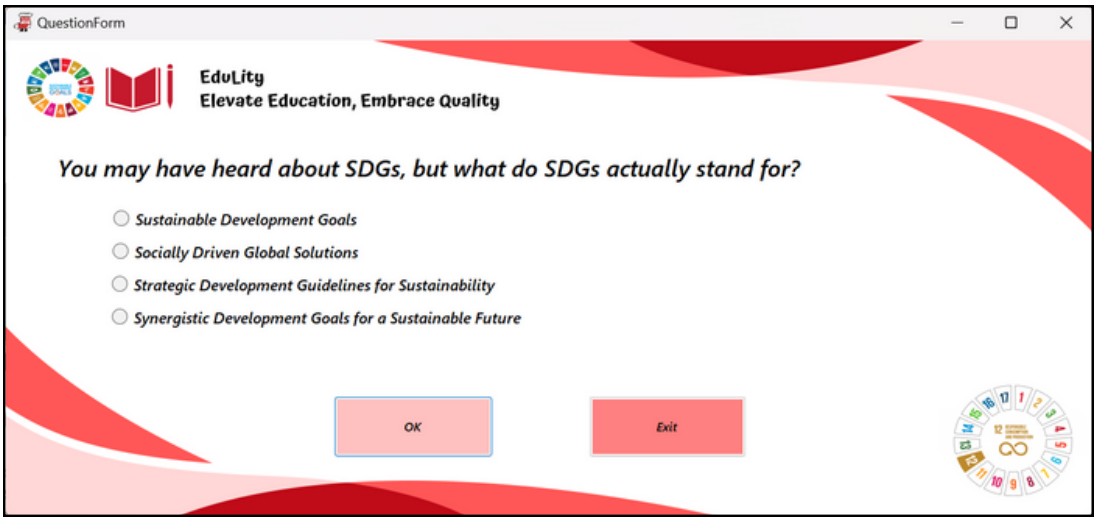
## Forms Design

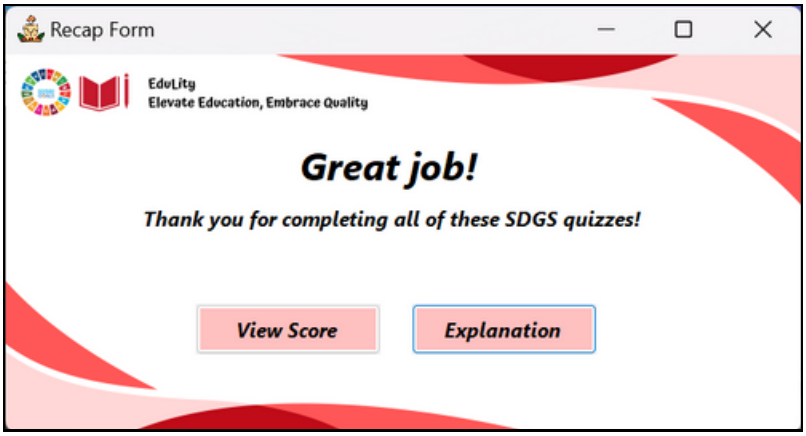


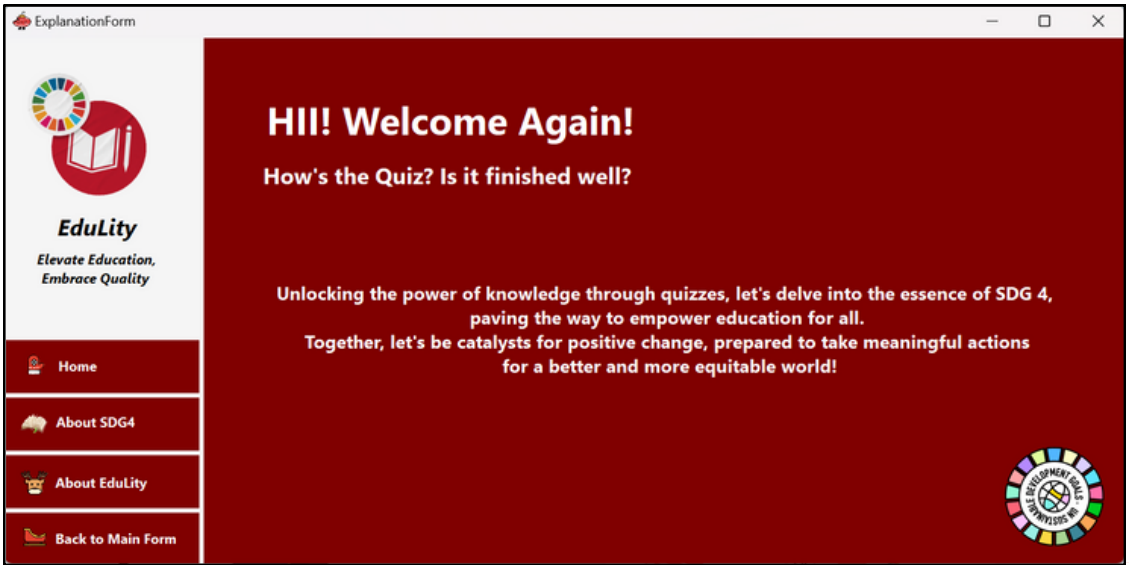




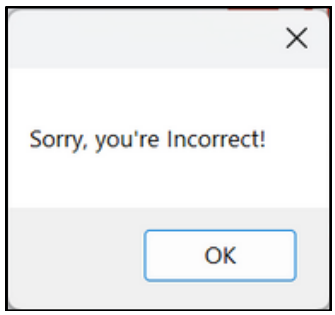

# Forms Design

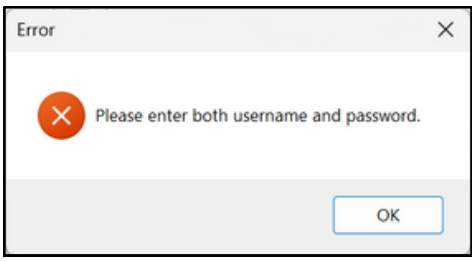


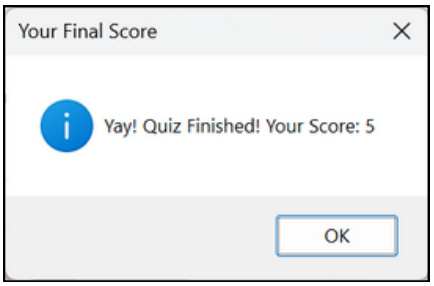



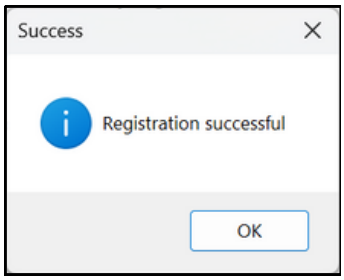



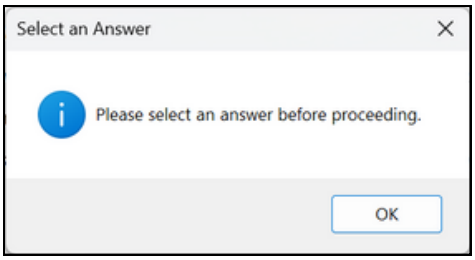
# Pop Up Notification

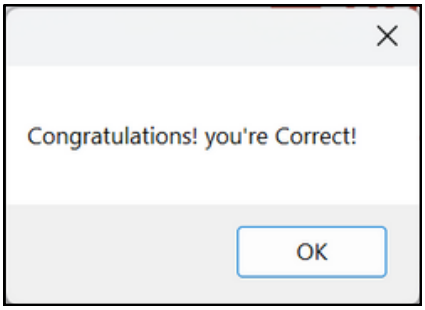


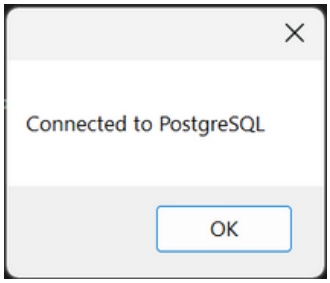



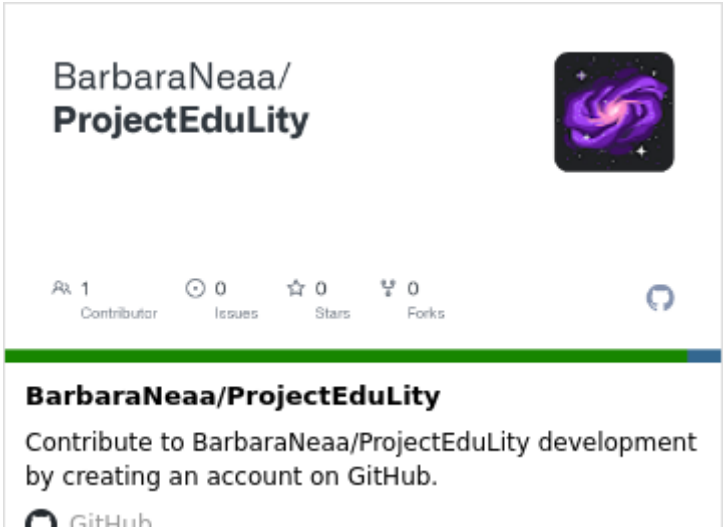

# Elements in EduLity

I added Christmas themed elements for decoration because EduLity was created during the Christmas season!


Our Repository in

# GITHUB

https://github.com/BarbaraNeaa/ProjectEduLity.git



Our Presentation on

# YouTube

https://youtu.be/T75qnlH7v1I

# Thank You!

**EDULITY**
**ELEVATE EDUCATION, EMBRACE QUALITY**

PROJECT AKHIR PEMROGRAMAN BERORIENTASI OBJEK
**KELOMPOK ASTERIA**

# Member Information

## ASTERIA GROUP

### BARBARA Neanake Ajiesti

barbaraneanakeajiesti@mail.ugm.ac.id
instagram.com/neanakee
wa.me/6285875891040

### Christella JESSLYN Dewantara

christellajesslyndewantara@mail.ugm.ac.id
instagram.com/jzz.jesslyn
wa.me/6285803747376