



KONSTRUKSI PERANGKAT LUNAK

MODUL 15

D

I

S

U

S

U

N

OLEH :

TIURMA GRACE ANGELINA – 2311104042

S1-SE07-02

Dosen :

YIS

Prodi S1 Rekaya Perangkat Lunak

Direktorat Kampus Purwokerto

1. CODE

User.cs

```
namespace modul15_2311104042
{
    public class User
    {
        public string Username { get; set; }
        public string PasswordHash { get; set; }
    }
}
```

Program.cs

```
using System;

namespace modul15_2311104042
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("=== Aplikasi Login/Register ===");
            Console.WriteLine("1. Register");
            Console.WriteLine("2. Login");
            Console.Write("Pilih menu (1/2): ");
            string pilihan = Console.ReadLine();

            switch (pilihan)
            {
                case "1":
                    Auth.Register();
                    break;
                case "2":
                    Auth.Login();
                    break;
                default:
                    Console.WriteLine("Pilihan tidak valid.");
                    break;
            }

            Console.ReadKey();
        }
    }
}
```

Auth.cs

```
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
```

```

using System.Text.Json;
using System.Text.RegularExpressions;

namespace modul15_2311104042
{
    public class Auth
    {
        private static string filePath = "users.json";

        public static void Register()
        {
            Console.WriteLine("Masukkan username: ");
            string username = Console.ReadLine();

            if (username.Length < 5 || username.Length > 20 ||
                !Regex.IsMatch(username, @"^[a-zA-Z0-9]+$"))
            {
                Console.WriteLine("Username harus 5-20 karakter dan  
hanya boleh huruf/angka.");
                return;
            }

            Console.WriteLine("Masukkan password: ");
            string password = Console.ReadLine();

            if (password.Length < 8 || !Regex.IsMatch(password,
                @"[!@#%$^&*]") || password.Contains(username))
            {
                Console.WriteLine("Password minimal 8 karakter, harus  
mengandung 1 karakter spesial (!@#%$^&*), dan tidak boleh  
mengandung username.");
                return;
            }

            string hashPassword = ComputeSha256Hash(password);

            var users = LoadUsers();

            if (users.Any(u => u.Username == username))
            {
                Console.WriteLine("Username sudah digunakan.");
                return;
            }

            users.Add(new User { Username = username,
                PasswordHash = hashPassword });
            SaveUsers(users);

            Console.WriteLine("Registrasi berhasil.");
        }
    }
}

```

```

public static void Login()
{
    Console.Write("Username: ");
    string username = Console.ReadLine();

    Console.Write("Password: ");
    string password = Console.ReadLine();

    string hashPassword = ComputeSha256Hash(password);
    var users = LoadUsers();

    var user = users.FirstOrDefault(u => u.Username ==
username && u.PasswordHash == hashPassword);
    if (user != null)
    {
        Console.WriteLine("Login berhasil!");
    }
    else
    {
        Console.WriteLine("Username atau password salah.");
    }
}

private static List<User> LoadUsers()
{
    if (!File.Exists(filePath))
    {
        return new List<User>();
    }

    string json = File.ReadAllText(filePath);
    return JsonSerializer.Deserialize<List<User>>(json);
}

private static void SaveUsers(List<User> users)
{
    string json = JsonSerializer.Serialize(users, new
JsonSerializerOptions { WriteIndented = true });
    File.WriteAllText(filePath, json);
}

private static string ComputeSha256Hash(string rawData)
{
    using (SHA256 sha256 = SHA256.Create())
    {
        byte[] bytes =
sha256.ComputeHash(Encoding.UTF8.GetBytes(rawData));
        return Convert.ToHexString(bytes);
    }
}

```

```
}
```

Output :

```
Microsoft Visual Studio Debug Console
=== Aplikasi Login/Register ===
1. Register
2. Login
Pilih menu (1/2): 1
Masukkan username: tiurma123
Masukkan password: Grace@2025
Registrasi berhasil.

C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_SISE-07-02\15_Review_Tugas_Besar\tjmod15_2311104042\tjmod15_2311104042\bin\Debug\net8.0\tjmod15_2311104042.exe (process 9676) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

```
Microsoft Visual Studio Debug Console
=== Aplikasi Login/Register ===
1. Register
2. Login
Pilih menu (1/2): 2
Username: tiurma123
Password: Grace@2025
Login berhasil!

C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_SISE-07-02\15_Review_Tugas_Besar\tjmod15_2311104042\tjmod15_2311104042\bin\Debug\net8.0\tjmod15_2311104042.exe (process 29896) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

user.json

```
users.json X
C:\Users\USER> source > repos > KPL_TiurmaGraceA_2311104042_SISE-07-02 > 15_Review_Tugas_Besar > tjmod15_2311104042 > tjmod15_2311104042 > bin > Debug > net8.0 > users.json > ...

1 {
2   {
3     "Username": "tiurma123",
4     "PasswordHash": "7812DC644D30654ED555D65461D377D32706CCF130527EBF83AC3FE246A74D63"
5   }
6 }
```

Penjelasan :

Program yang dikembangkan dalam modul 15 berdasarkan jurnal modul 05 dimana program ini merupakan aplikasi sederhana berbasis console yang memiliki fitur **registrasi dan login user**, serta mengimplementasikan prinsip **secure coding practices** sesuai dengan ketentuan tugas. Program terdiri dari tiga file utama, yaitu Program.cs, Auth.cs, dan User.cs.

Pada saat program dijalankan, pengguna akan disuguhkan dengan menu untuk memilih antara registrasi atau login. Jika pengguna memilih registrasi, maka program akan meminta input username dan password. Pada bagian ini, dilakukan validasi secara ketat terhadap input tersebut. Username harus memiliki panjang antara 5 hingga 20 karakter, serta hanya boleh terdiri dari huruf dan angka. Sementara itu, password harus memiliki panjang minimal 8 karakter, mengandung minimal satu karakter spesial seperti !@#\$%^&*, dan tidak boleh mengandung kata yang sama dengan username. Jika input tidak memenuhi syarat, maka sistem akan menolak input tersebut dan menampilkan pesan kesalahan yang sesuai.

Setelah input valid, password akan diproses menggunakan algoritma SHA256 untuk diubah menjadi bentuk hash. Hal ini dilakukan agar password tidak disimpan dalam bentuk teks asli (plain text) demi menjaga keamanan data pengguna. Username dan password yang telah di-hash kemudian disimpan dalam sebuah file bernama users.json. File ini akan otomatis dibuat di dalam folder output program (biasanya di folder bin/Debug/net6.0/), dan dapat berisi beberapa user yang telah teregistrasi.

Jika pengguna memilih login, sistem akan meminta input username dan password, lalu melakukan hashing terhadap password tersebut dan mencocokkannya dengan data yang tersimpan di dalam file users.json. Jika kombinasi username dan password yang di-hash cocok dengan salah satu data yang ada, maka sistem akan menampilkan pesan bahwa login berhasil. Sebaliknya, jika data tidak cocok, maka sistem akan menampilkan pesan bahwa login gagal.

Seluruh proses login dan registrasi telah menerapkan praktik secure coding, termasuk validasi input untuk mencegah masukan tidak valid atau berbahaya, serta penggunaan hash untuk menjaga keamanan penyimpanan password. Dengan pendekatan ini, program tidak hanya memenuhi fungsionalitas yang diminta, tetapi juga telah dibangun dengan prinsip keamanan yang baik, sehingga dapat menjadi dasar untuk sistem login yang aman pada aplikasi yang lebih kompleks.