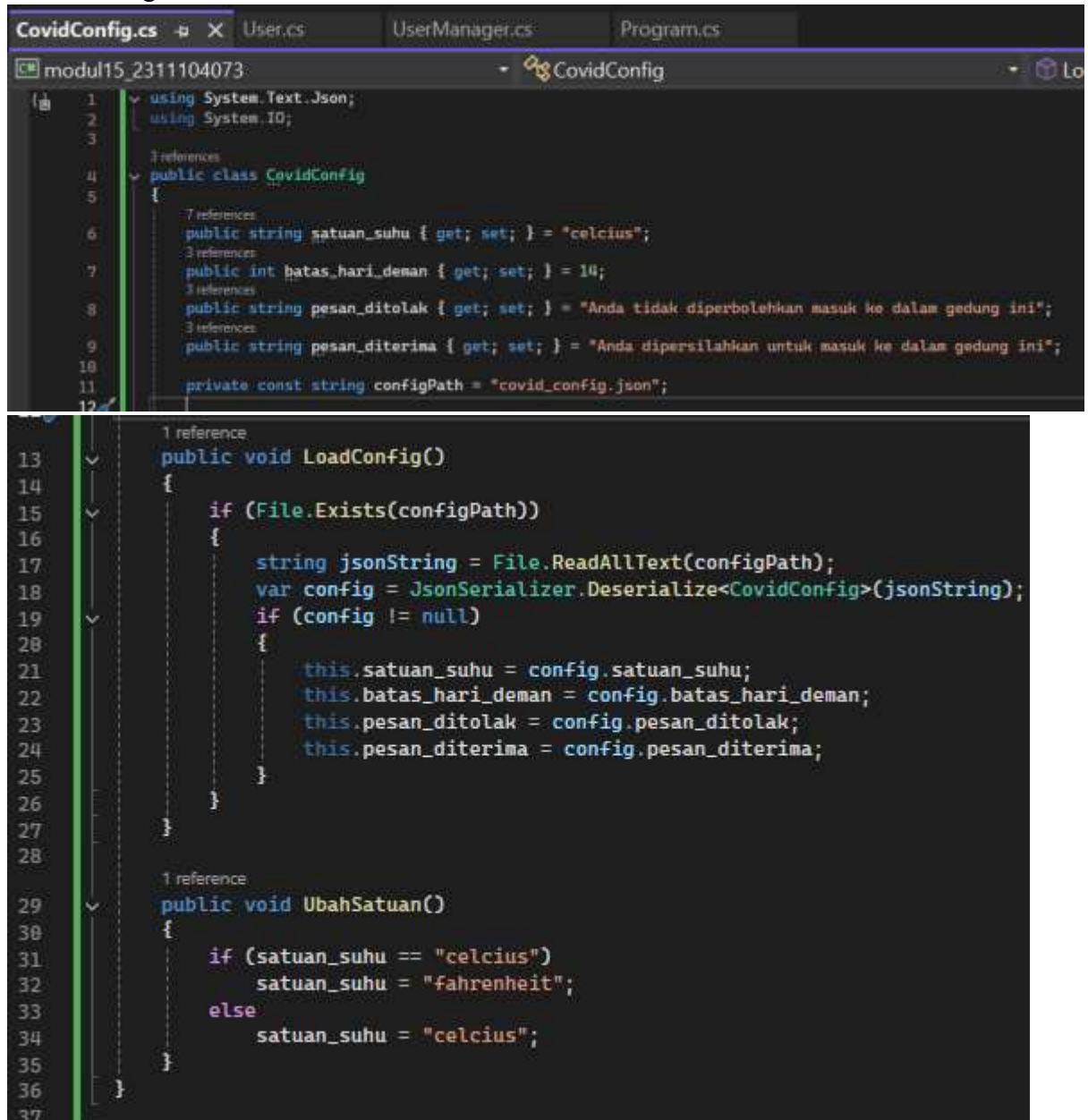


Nama: Farhan Kurniawan

NIM: 2311104073

Kelas: SE-07-02

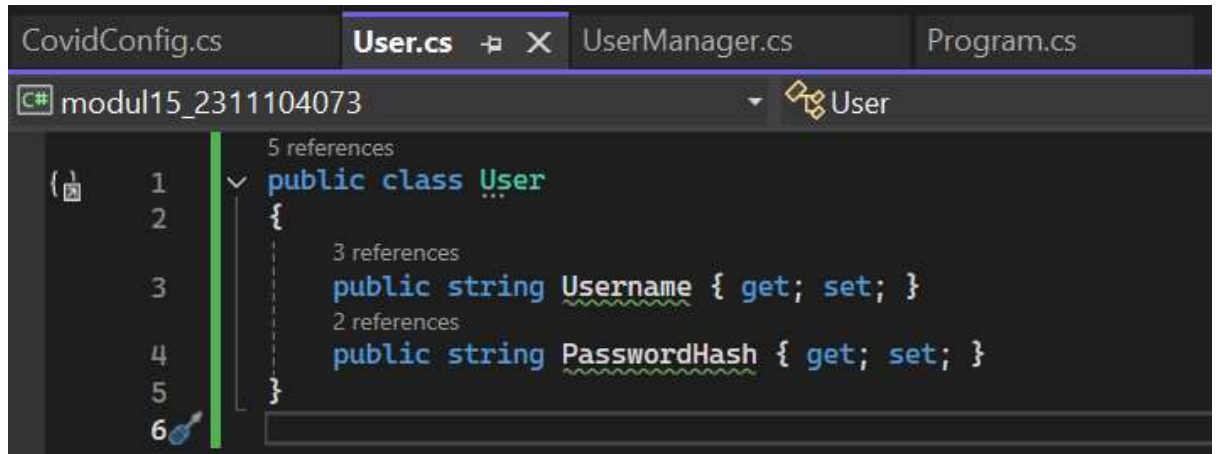
1. CovidConfig.cs



```
1 using System.Text.Json;
2 using System.IO;
3
4 // references
5 public class CovidConfig
6 {
7     // 7 references
8     public string satuan_suhu { get; set; } = "celcius";
9     // 3 references
10    public int batas_hari_deman { get; set; } = 14;
11    // 3 references
12    public string pesan_ditolak { get; set; } = "Anda tidak diperbolehkan masuk ke dalam gedung ini";
13    // 3 references
14    public string pesan_diterima { get; set; } = "Anda dipersilahkan untuk masuk ke dalam gedung ini";
15    private const string configPath = "covid_config.json";
16
17    // 1 reference
18    public void LoadConfig()
19    {
20        if (File.Exists(configPath))
21        {
22            string jsonString = File.ReadAllText(configPath);
23            var config = JsonSerializer.Deserialize<CovidConfig>(jsonString);
24            if (config != null)
25            {
26                this.satuan_suhu = config.satuan_suhu;
27                this.batas_hari_deman = config.batas_hari_deman;
28                this.pesan_ditolak = config.pesan_ditolak;
29                this.pesan_diterima = config.pesan_diterima;
30            }
31        }
32    }
33
34    // 1 reference
35    public void UbahSatuan()
36    {
37        if (satuan_suhu == "celcius")
38            satuan_suhu = "fahrenheit";
39        else
40            satuan_suhu = "celcius";
41    }
42 }
```

File CovidConfig.cs berisi class CovidConfig yang bertugas memuat konfigurasi protokol kesehatan seperti satuan suhu, batas hari demam, serta pesan diterima atau ditolak. Data ini dibaca dari file JSON eksternal bernama covid_config.json menggunakan System.Text.Json. Jika file konfigurasi ada, maka nilai-nilai properti akan diperbarui sesuai isi file. Selain itu, terdapat metode UbahSatuan() yang mengubah satuan suhu dari Celcius ke Fahrenheit atau sebaliknya.

2. User.cs



File User.cs mendefinisikan class User yang merepresentasikan akun pengguna. Setiap objek User menyimpan dua properti penting: Username dan PasswordHash. Password disimpan dalam bentuk hash (SHA256) untuk keamanan sesuai dengan prinsip Secure Coding Practices. File ini juga menyediakan metode statis HashPassword(string password) untuk melakukan hashing terhadap input password yang akan disimpan atau dibandingkan saat login.

3. UserManager.cs

```
using System.Text.Json;
using System.Security.Cryptography;
using System.Text;
using System.Text.RegularExpressions;

public class UserManager
{
    private const string filePath = "user_data.json";
    private List<User> users = new List<User>();

    public UserManager()
    {
        LoadUsers();
    }

    public bool Register(string username, string password, out string errorMsg)
    {
        errorMsg = "";

        // Validasi panjang username
        if (username.Length < 4 || username.Length > 20 ||
            !Regex.IsMatch(username, @"^[a-zA-Z]+$"))
        {
            errorMsg = "Username harus terdiri dari 4-20 huruf alfabet ASCII saja.";
            return false;
        }

        // Validasi panjang password
        if (password.Length < 8 || password.Length > 20)
        {
            errorMsg = "Password harus 8-20 karakter.";
            return false;
        }
    }
}
```

```

    }

    // Password tidak boleh mengandung username
    if (password.ToLower().Contains(username.ToLower()))
    {
        errorMsg = "Password tidak boleh mengandung username.";
        return false;
    }

    // Password harus mengandung angka dan karakter unik
    if (!Regex.IsMatch(password, @"^d") || !Regex.IsMatch(password,
@"[!@#$$%^&*]"))
    {
        errorMsg = "Password harus mengandung setidaknya 1 angka dan
1 karakter unik (!@#$$%^&*).";
        return false;
    }

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

    string hash = ComputeSha256Hash(password);
    users.Add(new User { Username = username, PasswordHash = hash
});

    SaveUsers();
    return true;
}

public bool Login(string username, string password)
{
    string hash = ComputeSha256Hash(password);
    return users.Any(u => u.Username == username && u.PasswordHash
== hash);
}

private void LoadUsers()
{
    if (File.Exists(filePath))
    {
        string json = File.ReadAllText(filePath);
        users = JsonSerializer.Deserialize<List<User>>(json) ?? new
List<User>();
    }
}

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

private string ComputeSha256Hash(string rawData)
{
    using (SHA256 sha256 = SHA256.Create())
    {
        byte[] bytes =
sha256.ComputeHash(Encoding.UTF8.GetBytes(rawData));
        return Convert.ToHexString(bytes); // .NET 5+
    }
}

```

```
}  
}
```

File `UserManager.cs` mendefinisikan class `userManager` yang bertanggung jawab mengelola seluruh proses autentikasi pengguna, yaitu registrasi dan login. Di dalamnya, `userManager` menyimpan daftar pengguna (`List<User>`) yang dimuat dari atau disimpan ke file JSON `user_data.json`. Proses registrasi mencakup validasi panjang dan karakter username, serta validasi password agar memenuhi aturan keamanan (panjang minimal, karakter unik, dan tidak mengandung username). Password yang valid kemudian di-hash menggunakan algoritma SHA256 sebelum disimpan. Saat login, `userManager` mencocokkan username dan password yang telah di-hash dengan data yang tersimpan. File ini menerapkan prinsip Secure Coding Practices seperti validasi input, penanganan data tidak valid, dan pengelolaan password secara aman. Fungsinya sangat penting sebagai lapisan penghubung antara interaksi user di `Program.cs` dan objek data `User`.

4. Program.cs

```
using System;  
  
class Program  
{  
    static void Main(string[] args)  
    {  
        userManager userManager = new userManager();  
  
        while (true)  
        {  
            Console.WriteLine("=== Menu ===");  
            Console.WriteLine("1. Registrasi");  
            Console.WriteLine("2. Login");  
            Console.WriteLine("3. Keluar");  
            Console.Write("Pilih opsi: ");  
            string option = Console.ReadLine();  
  
            if (option == "1")  
            {  
                Console.Write("Masukkan username: ");  
                string username = Console.ReadLine();  
  
                Console.Write("Masukkan password: ");  
                string password = Console.ReadLine();  
  
                if (userManager.Register(username, password, out string  
errorMsg))  
                    Console.WriteLine("Registrasi berhasil!");  
                else  
                    Console.WriteLine($"Registrasi gagal: {errorMsg}");  
            }  
            else if (option == "2")  
            {  
                Console.Write("Masukkan username: ");  
                string username = Console.ReadLine();  
  
                Console.Write("Masukkan password: ");
```

```

        string password = Console.ReadLine();

        if (userManager.Login(username, password))
        {
            Console.WriteLine("Login berhasil!");

            CovidConfig config = new CovidConfig();
            config.LoadConfig();
            config.UbahSatuan();

            Console.Write($"Berapa suhu badan anda saat ini?
Dalam nilai {config.satuan_suhu}: ");
            if (!double.TryParse(Console.ReadLine(), out double
suhu))
            {
                Console.WriteLine("Input suhu tidak valid.");
                continue;
            }

            Console.Write("Berapa hari yang lalu (perkiraan)
anda terakhir memiliki gejala demam? ");
            if (!int.TryParse(Console.ReadLine(), out int
hariDemam))
            {
                Console.WriteLine("Input hari demam tidak
valid.");
                continue;
            }

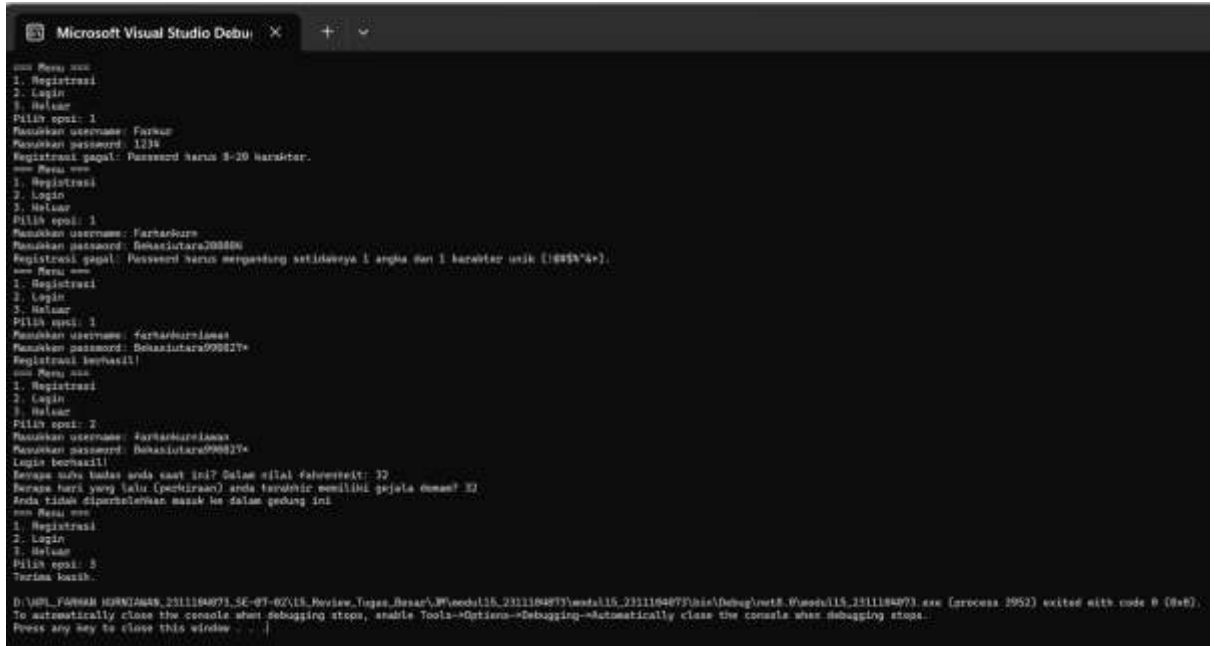
            bool suhuNormal = false;
            if (config.satuan_suhu == "celcius")
                suhuNormal = suhu >= 36.5 && suhu <= 37.5;
            else
                suhuNormal = suhu >= 97.7 && suhu <= 99.5;

            Console.WriteLine(suhuNormal && hariDemam <
config.batas_hari_deman
                ? config.pesan_diterima
                : config.pesan_ditolak);
        }
        else
        {
            Console.WriteLine("Login gagal: Username atau
password salah.");
        }
    }
    else if (option == "3")
    {
        Console.WriteLine("Terima kasih.");
        break;
    }
    else
    {
        Console.WriteLine("Ops! tidak valid.");
    }
}
}
}

```

File Program.cs adalah file utama tempat logika program dijalankan. Program akan memuat konfigurasi dari CovidConfig, lalu meminta input suhu tubuh dan hari terakhir mengalami demam dari pengguna. Berdasarkan konfigurasi tersebut, program memeriksa apakah suhu tubuh dalam batas normal dan apakah hari demam berada di bawah batas. Selain itu, file ini mengatur menu interaktif yang memungkinkan pengguna melakukan registrasi, login, dan pengecekan suhu — semuanya dikelola oleh class Manager.

5. Output:



```
Microsoft Visual Studio Debug Console
>
1. Registrasi
2. Login
3. Keluar
Pilih opsi: 1
Masukkan username: Farhan
Masukkan password: 1234
Registrasi gagal: Password harus 8-20 karakter.
>
1. Registrasi
2. Login
3. Keluar
Pilih opsi: 1
Masukkan username: Farhan@ura
Masukkan password: Bhanjutara200406
Registrasi gagal: Password harus mengandung setidaknya 1 angka dan 1 karakter unik ([a-zA-Z*]).
>
1. Registrasi
2. Login
3. Keluar
Pilih opsi: 1
Masukkan username: farhan@ura@mas
Masukkan password: Bhanjutara990812*
Registrasi berhasil!
>
1. Registrasi
2. Login
3. Keluar
Pilih opsi: 2
Masukkan username: farhan@ura@mas
Masukkan password: Bhanjutara990812*
Login berhasil!
Seperti suhu tubuh anda saat ini? Silah pilih Fahrenheit: 32
Berapa hari yang lalu (penderitaan) anda terakhir memiliki gejala demam? 32
Anda tidak dipaparkan masuk ke dalam gedung ini.
>
1. Registrasi
2. Login
3. Keluar
Pilih opsi: 3
Terima kasih.
D:\VIRUS_FANAN_HURNIANGAS_2311184073_5C-07-02\15.Review_Tugas_Besar\B\modul15_2311184073\modul15_2311184073\bin\Debug\net8.0\modul15_2311184073.exe (process 2952) 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 . . .
```