

**RAKTIKUM KONSTRUKSI PERANGKAT LUNAK**  
**TUGAS JURNAL 15**



**Telkom  
University**

disusun Oleh:  
Nita Fitrotul Mar'ah  
2211104005  
SE0601

**Dosen Pengampu :**  
Yudha Islami Sulistya, S.Kom., M.Cs.

**S1 REKAYASA PERANGKAT LUNAK**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2025**

## **1. MEMBUAT PROJECT MODUL**

Buka IDE misalnya dengan Visual Studio

- a. Copy salah satu folder tugas pendahuluan yang dimiliki sebelumnya (dari modul 2 sampai modul 13), kemudian rename folder hasil copy-paste tersebut dengan modul15\_NIM (coba pilih tugas pendahuluan yang paling sederhana)
- b. Misalnya menggunakan Visual Studio, bukalah project/folder yang di-copy sebelumnya.

## **2. PENGEMBANGAN DENGAN SECURE CODING PRACTICES**

Buatlah aplikasi desktop dengan fitur:

- a. Registrasi user dengan input username dan password
- b. Penyimpanan data user pada file json
- c. Login user
- d. Comments

Dengan mengikuti Secure Coding Practices yang memenuhi faktor-faktor berikut:

- a. Input Validation (wajib mengimplementasikan salah satu, diizinkan lebih)

1. Validasi range data

Range data input harus dibatasi dan ditetapkan. Contoh:

- Hanya boleh huruf alfabet ASCII
- Harus mengandung angka

2. Validasi panjang data

Panjang atau ukuran data harus dibatasi dan ditetapkan. Contoh:

- Minimal 8 karakter
- Maksimal 20 karakter

3. Handling data invalid

Data yang tidak valid harus ditolak atau di handle dengan jelas (jangan dibiarkan menjadi runtime error yang tidak di handle). Contoh:

- Jika terdapat aturan minimal 8 karakter input, input di bawah 8 karakter harus ditolak atau di handle dengan spesifik
- Jika terdapat aturan hanya boleh huruf alfabet ASCII, input dengan karakter selain alfabet ASCII harus ditolak atau di handle dengan spesifik

- b. Password Management (wajib mengimplementasikan salah satu, diizinkan lebih)

1. Password hashing

Ketika sistem menyimpan password, password harus dienkripsi atau dihash.

Contoh:

- Sistem mengenkripsi password dengan metode hash SHA256
- Sistem mengubah password dengan konsisten supaya tidak sama persis dengan inputan user

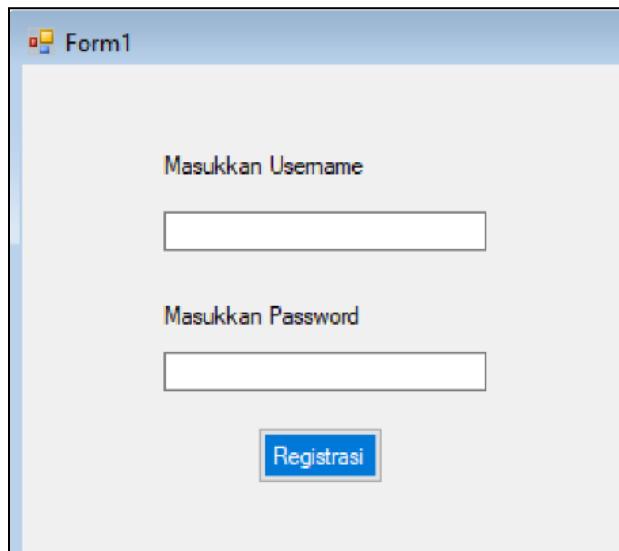
2. Password rules

Ketika sistem harus menerima inputan password, harus ada aturan keamanan untuk password tersebut. Contoh:

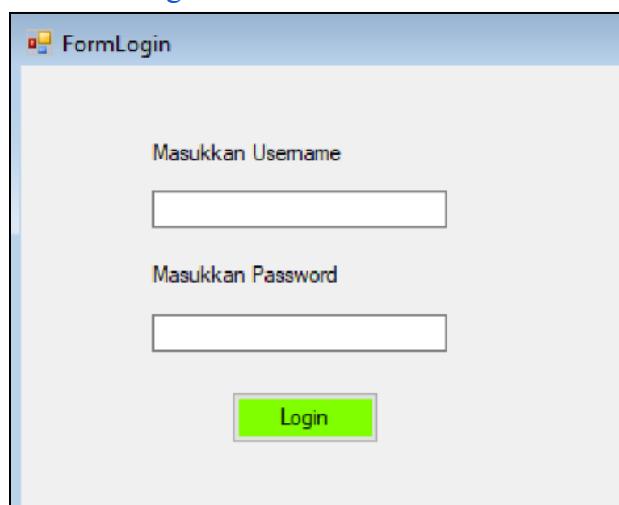
- Password harus mengandung minimal 1 karakter unik (!@#\$%^&\*)
- Password tidak boleh mengandung kata dari username

**Jawab:**

1. File [FormRegister.cs](#)



2. File [FormLogin.cs](#)



3. File [FormRegister.cs](#)

```
namespace modulis_2211104005
{
    1 reference
    public partial class FormRegister: Form
    {
        0 references
        public FormRegister()
        {
            InitializeComponent();
        }

        0 references
        private void FormRegister_Load(object sender, EventArgs e)
        {
        }

        0 references
        private void btnRegister_Click(object sender, EventArgs e)
        {
            string username = txtUsername.Text;
            string password = txtPassword.Text;

            if (!AuthHelper.IsValidInput(username, password, out string error))
            {
                MessageBox.Show(error, "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
                return;
            }

            string hashed = AuthHelper.HashPassword(password);
            AuthHelper.SaveUser(new User { Username = username, PasswordHash = hashed });

            MessageBox.Show("Registrasi berhasil!", "Info", MessageBoxButtons.OK, MessageBoxIcon.Information);
            this.Hide();
            FormLogin loginForm = new FormLogin();
            loginForm.Show();
        }

        0 references
        private void label1_Click(object sender, EventArgs e)
        {
        }
    }
}
```

#### 4. File FormLogin.cs

```
namespace modulis_2211104005
{
    1 reference
    public partial class FormLogin: Form
    {
        0 references
        public FormLogin()
        {
            InitializeComponent();
        }

        0 references
        private void btnLogin_Click(object sender, EventArgs e)
        {
            string username = txtUsernameLogin.Text;
            string password = txtPasswordLogin.Text;

            if (AuthHelper.CheckLogin(username, password))
            {
                MessageBox.Show("Login berhasil!", "Sukses", MessageBoxButtons.OK, MessageBoxIcon.Information);
                // TODO: buka Form utama aplikasi
            }
            else
            {
                MessageBox.Show("Username atau password salah!", "Gagal", MessageBoxButtons.OK, MessageBoxIcon.Error);
            }
        }

        0 references
        private void FormLogin_Load(object sender, EventArgs e)
        {
        }

        0 references
        private void label1_Click(object sender, EventArgs e)
        {
        }

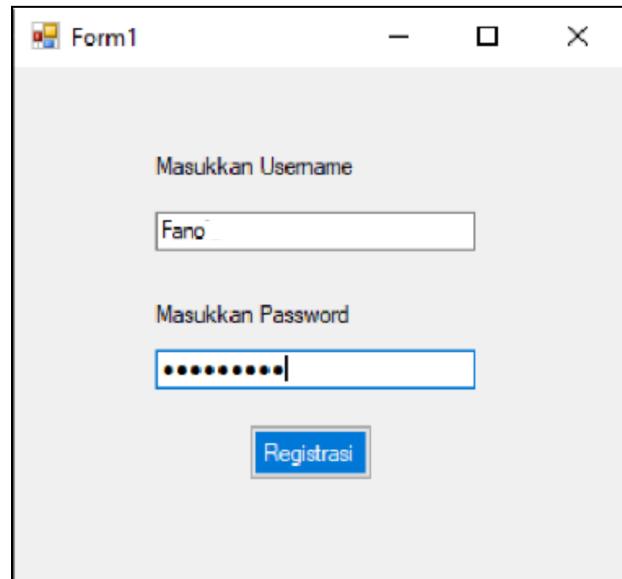
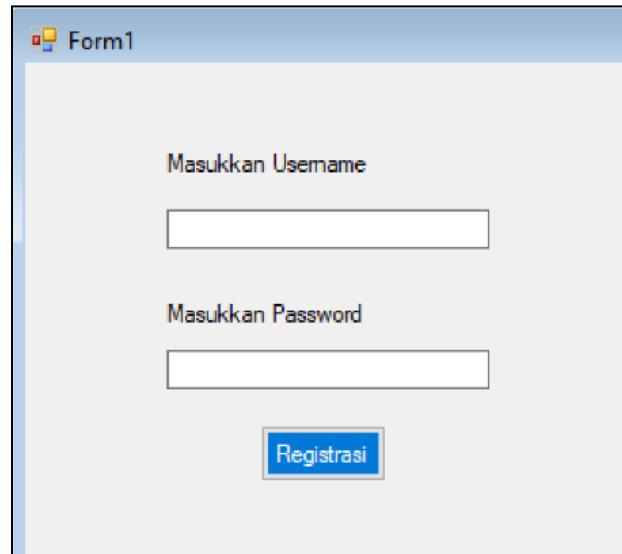
        0 references
        private void label1_Click_1(object sender, EventArgs e)
        {
        }
    }
}
```

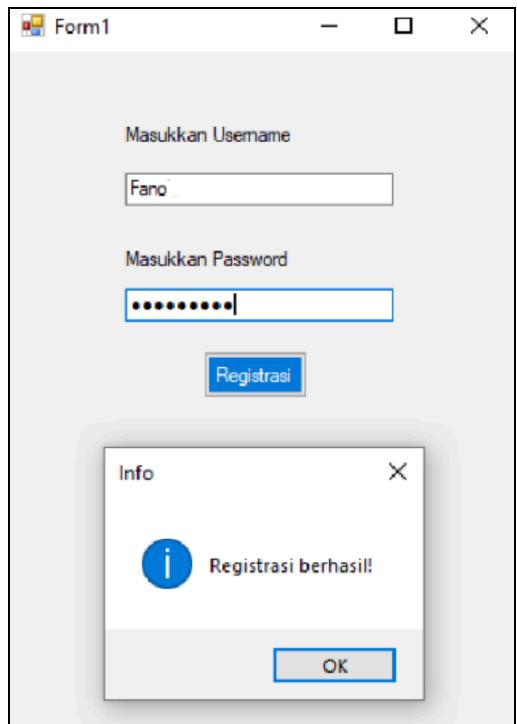
#### 5. File User.cs

```
namespace modul15_2211104005
{
    public class User
    {
        public string Username { get; set; }
        public string PasswordHash { get; set; }
    }
}
```

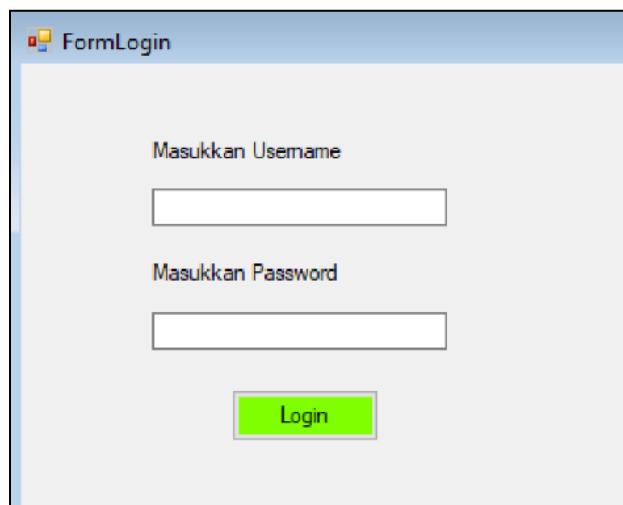
## Screenshot Output

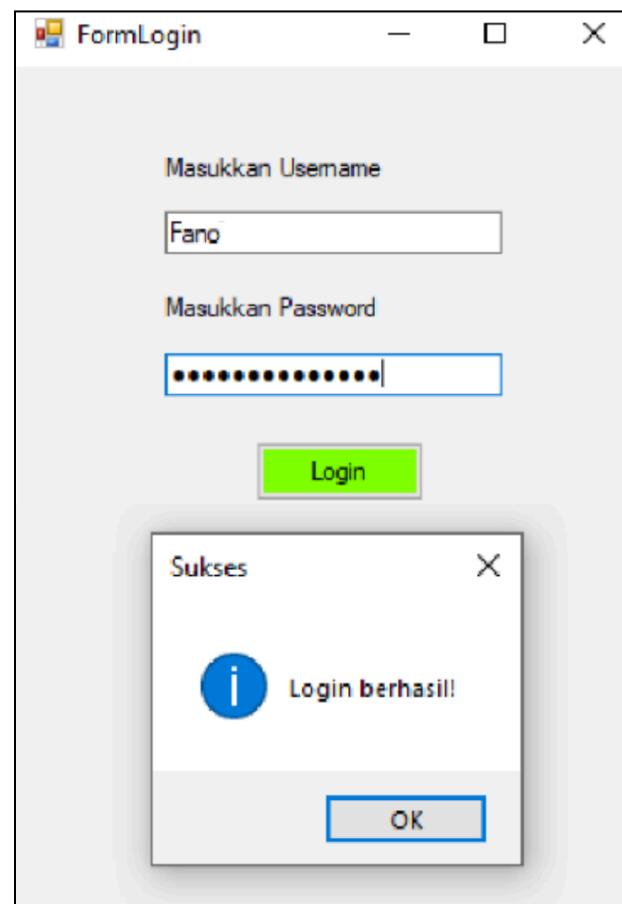
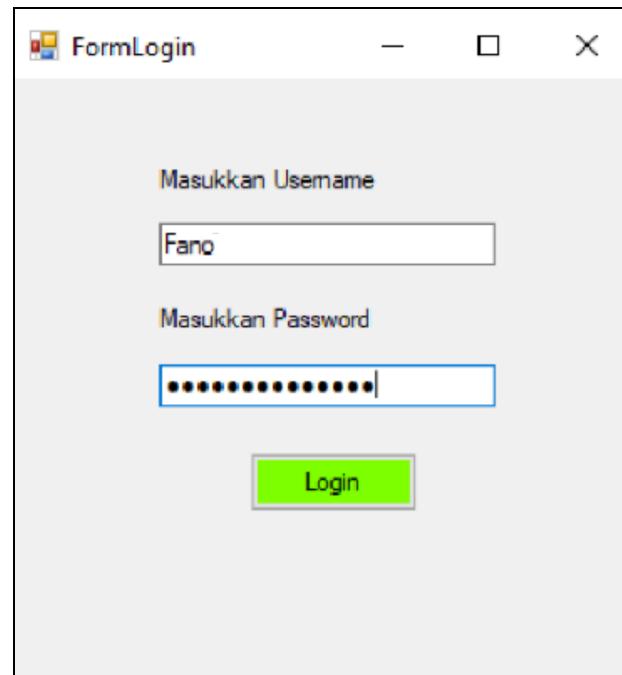
### 1. Registrasi





## 2. Login





## **Penjelasan**

Sistem ini memiliki fitur registrasi dan login pengguna yang terintegrasi dalam aplikasi berbasis Windows Forms. Proses pembuatan akun baru dikelola melalui FormRegister.cs, di mana dilakukan validasi ketat terhadap username dan password. Username wajib terdiri dari huruf alfabet, sedangkan password harus memiliki panjang 8 hingga 20 karakter, mengandung angka, serta setidaknya satu karakter khusus seperti !@#\$%^&\*. Selain itu, password tidak boleh terlalu mirip dengan username. Jika ada syarat yang tidak terpenuhi, sistem akan menampilkan pesan kesalahan yang jelas.

Apabila seluruh kriteria validasi dipenuhi, password akan di-hash menggunakan algoritma SHA256 untuk menjaga keamanannya sebelum disimpan.

Pada saat login, FormLogin.cs bertanggung jawab memeriksa kredensial yang diinputkan dengan data yang sudah disimpan. Karena password sudah di-hash ketika registrasi, password yang diketik pengguna saat login juga akan di-hash lebih dulu untuk kemudian dibandingkan. Jika hash cocok, pengguna akan mendapatkan notifikasi login berhasil, sedangkan jika tidak cocok akan muncul pesan error.

Struktur data pengguna, yang hanya memuat username dan password dalam bentuk hash, didefinisikan pada file User.cs.