

**TUGAS JURNAL  
KONSTRUKSI PERANGKAT LUNAK**

**MODUL XV**



**Disusun Oleh :**

Rizky Hanifa Afania

2211104017

SE-06-01

**Dosen Pengampu :**

Yudha Islami Sulistya, S.Kom., M.Cs.

**PROGRAM STUDI S1 SOFTWARE ENGINEERING  
TELKOM UNIVERSITY PURWOKERTO**

## **TUGAS JURNAL XV**

### **1. MEMBUAT PROJECT MODUL**

Buka IDE misalnya dengan Visual Studio

- A. Copy salah satu folder tugas jurnal 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:

1. Registrasi user dengan input username dan password
2. Penyimpanan data user pada file json
3. Login user

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

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

- a. Validasi range data

Range data input harus dibatasi dan ditetapkan. Contoh:

- Hanya boleh huruf alfabet ASCII
- Harus mengandung angka

- b. Validasi panjang data

- Panjang atau ukuran data harus dibatasi dan ditetapkan. Contoh:
- Minimal 8 karakter
- Maksimal 20 karakter

- c. Handling data invalid

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

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

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

a. 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

b. Password rules

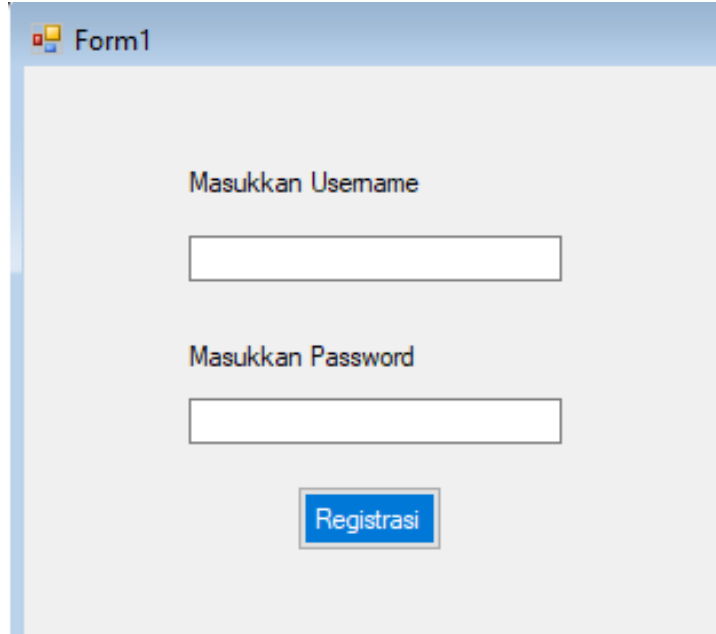
Ketika sistem harus menerima inputan password, harus ada aturan keamanan untuk password tersebut. Contoh:

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

**Jawab:**

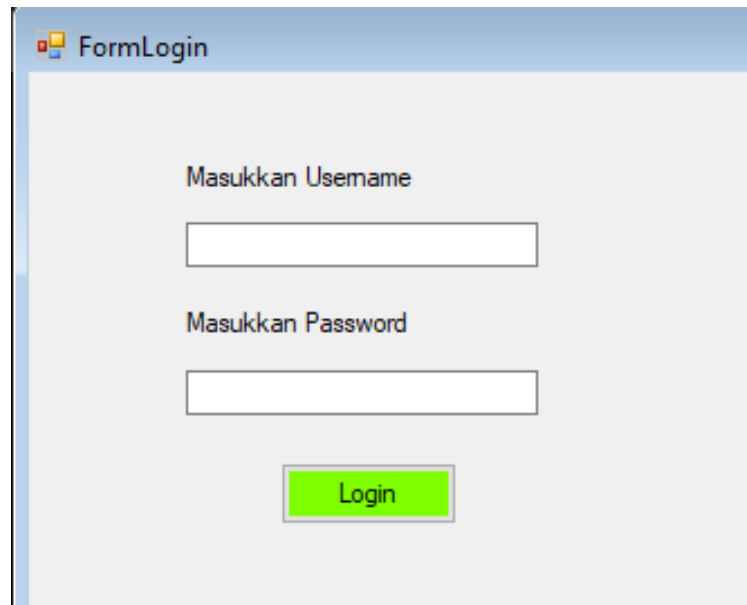
**Source Code**

- a. File [FormRegister.cs](#)[Design]



The image shows a Windows Form titled "Form1". It has a light gray background. At the top, there is a label "Masukkan Username" above a white text box. Below that is another label "Masukkan Password" above another white text box. At the bottom center, there is a blue button with the text "Registrasi" in white.

b. File [FormLogin.cs](#)[Design]



FormLogin

Masukkan Username

Masukkan Password

Login

c. File [FormRegister.cs](#)

```
modul15_2211104017 modul15_2211104017.FormRegister
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Text;
8  using System.Threading.Tasks;
9  using System.Windows.Forms;
10
11  namespace modul15_2211104017
12  {
13      public partial class FormRegister: Form
14      {
15          public FormRegister()
16          {
17              InitializeComponent();
18          }
19
20          private void FormRegister_Load(object sender, EventArgs e)
21          {
22          }
23
24          private void btnRegister_Click(object sender, EventArgs e)
25          {
26              string username = txtUsername.Text;
27              string password = txtPassword.Text;
28
29              if (!AuthHelper.IsValidInput(username, password, out string error))
30              {
31                  MessageBox.Show(error, "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
32                  return;
33              }
34
35              string hashed = AuthHelper.HashPassword(password);
36              AuthHelper.SaveUser(new User { Username = username, PasswordHash = hashed });
37
38              MessageBox.Show("Registrasi berhasil!", "Info", MessageBoxButtons.OK, MessageBoxIcon.Information);
39              this.Hide();
40              FormLogin loginForm = new FormLogin();
41              loginForm.Show();
42          }
43
44          private void label1_Click(object sender, EventArgs e)
45          {
46          }
47      }
48  }
```

d. File [FormLogin.cs](#)

```
modul15_2211104017 modul15_2211104017.FormLogin
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Text;
8  using System.Threading.Tasks;
9  using System.Windows.Forms;
10
11  namespace modul15_2211104017
12  {
13      public partial class FormLogin: Form
14      {
15          public FormLogin()
16          {
17              InitializeComponent();
18          }
19
20          private void btnLogin_Click(object sender, EventArgs e)
21          {
22              string username = txtUsernameLogin.Text;
23              string password = txtPasswordLogin.Text;
24
25              if (AuthHelper.CheckLogin(username, password))
26              {
27                  MessageBox.Show("Login berhasil!", "Sukses", MessageBoxButtons.OK, MessageBoxIcon.Information);
28                  // TODO: buka form utama aplikasi
29              }
30              else
31              {
32                  MessageBox.Show("Username atau password salah!", "Gagal", MessageBoxButtons.OK, MessageBoxIcon.Error);
33              }
34          }
35
36          private void FormLogin_Load(object sender, EventArgs e)
37          {
38          }
39
40          private void label1_Click(object sender, EventArgs e)
41          {
42          }
43
44          private void label1_Click_1(object sender, EventArgs e)
45          {
46          }
47
48          }
49  }
```

e. File [User.cs](#)

```
modul15_2211104017 modul15_2211104017.User
1  namespace modul15_2211104017
2  {
3      public class User
4      {
5          public string Username { get; set; }
6          public string PasswordHash { get; set; }
7      }
8  }
```

## Screenshoot Output

### a. Registrasi

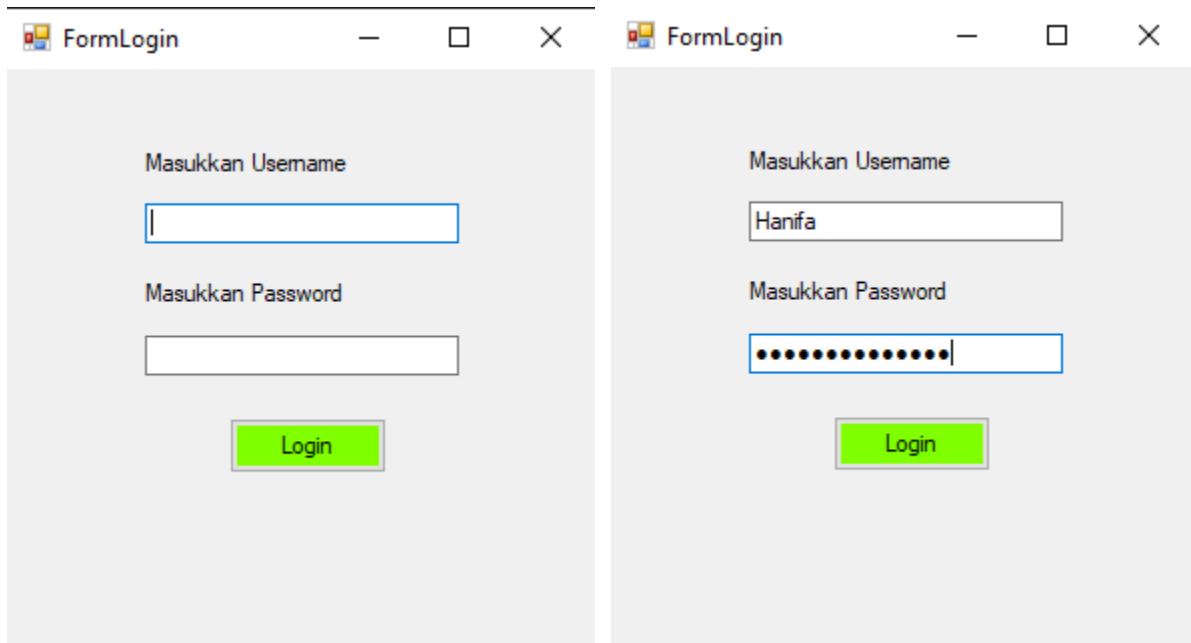
The image displays three sequential screenshots of a registration form titled "Form1".

**Screenshot 1 (Top Left):** The form contains two input fields: "Masukkan Ustreame" (empty) and "Masukkan Password" (empty). A blue "Registrasi" button is at the bottom.

**Screenshot 2 (Top Right):** The "Masukkan Ustreame" field now contains the text "Hanifa". The "Masukkan Password" field contains masked characters (dots). The "Registrasi" button remains at the bottom.

**Screenshot 3 (Bottom):** This screenshot shows the same form with the fields filled. An "Info" dialog box is overlaid on the form, displaying a blue information icon, the text "Registrasi berhasil!", and an "OK" button.

b. Login



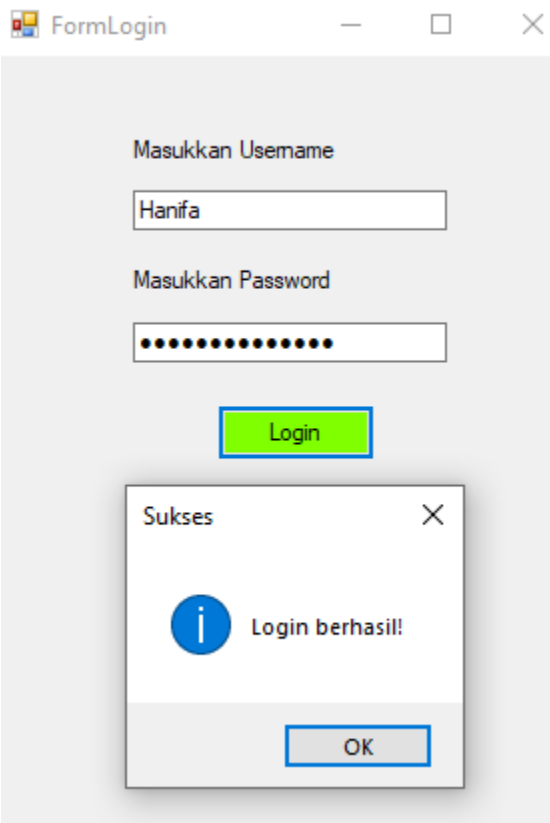
The image displays two side-by-side screenshots of a Windows application window titled "FormLogin".

The left screenshot shows the login form with the following elements:

- Label: "Masukkan Ustreame" (Note the typo in the image)
- Input field: Empty text box.
- Label: "Masukkan Password"
- Input field: Empty password box.
- Button: Green "Login" button.

The right screenshot shows the same form after input:

- Label: "Masukkan Ustreame"
- Input field: Contains the text "Hanifa".
- Label: "Masukkan Password"
- Input field: Contains 12 black dots, representing a masked password.
- Button: Green "Login" button.



The image shows the "FormLogin" application window with the following elements:

- Label: "Masukkan Ustreame"
- Input field: Contains the text "Hanifa".
- Label: "Masukkan Password"
- Input field: Contains 12 black dots.
- Button: Green "Login" button.

A modal dialog box is overlaid on top of the main window:

- Title: "Sukses"
- Icon: Information icon (blue circle with a white 'i').
- Text: "Login berhasil!"
- Button: "OK"



## Penjelasan

File [FormRegister.cs](#), [FormLogin.cs](#), dan [User.cs](#) bekerja sama untuk membuat fitur registrasi dan login sederhana dalam aplikasi berbasis Windows Forms. File `FormRegister.cs`, digunakan untuk mendaftarkan akun baru. Saat pengguna klik tombol Register, data username dan password akan diperiksa dulu. Sistem mengecek apakah username hanya huruf alfabet, apakah password cukup panjang (minimal 8, maksimal 20 karakter), mengandung angka, dan punya karakter unik seperti `!@#$%^&*`. Bahkan dicek juga apakah password-nya terlalu mirip dengan username. Kalau ada kesalahan, akan muncul pesan kesalahan yang jelas. Jika semua valid, password akan dihash dengan metode SHA256 supaya aman (jadi tidak disimpan dalam bentuk asli), lalu disimpan ke dalam sistem.

Sementara file `FormLogin.cs`, digunakan untuk masuk ke dalam aplikasi setelah pengguna berhasil mendaftar. Sistem akan mencocokkan username dan password yang dimasukkan dengan data yang tersimpan. Karena password disimpan dalam bentuk hash, password yang dimasukkan juga diubah ke bentuk hash terlebih dahulu, lalu dibandingkan. Kalau cocok, muncul pesan login sukses, kalau tidak, ditampilkan pesan kesalahan yang jelas. File `User.cs` sendiri hanya membuat struktur data user sederhana, yang menyimpan username dan password yang sudah dihash.