

# **LAPORAN PRATIKUM**

## ***“PEKAN 1”***

***Disusun Untuk Memenuhi Tugas Mata Kuliah PBO***

**DOSEN PENGAMPU:**

**Nurfiah, S.ST. M.Kom.**



**DISUSUN OLEH:**

**Karimah Irsyadiyah (2411533018)**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS  
T.A 2024/2025**

## **Daftar Pustaka**

BAB I PENDAHULUAN.....	3
1.1    Latar Belakang.....	3
1.2    Tujuan .....	3
1.3    Alat dan Bahan.....	3
BAB II PEMBAHASAN .....	5
2.1    Langkah-langkah Praktikum dan Pembahasan Program .....	5
BAB III PENUTUP .....	13
3.1    Kesimpulan .....	13

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Pemrograman Berorientasi Objek (PBO) merupakan salah satu paradigma pemrograman yang banyak digunakan dalam pengembangan perangkat lunak modern. Konsep utama PBO seperti enkapsulasi, pewarisan (inheritance), dan polimorfisme memungkinkan kode program lebih terstruktur, mudah dipelihara, serta dapat digunakan kembali.

Dalam implementasinya, PBO sering dikombinasikan dengan Graphical User Interface (GUI) untuk membuat aplikasi yang interaktif dan mudah digunakan. Salah satu pustaka GUI yang populer di Java adalah **Swing**, yang menyediakan berbagai komponen seperti tombol, teks, label, hingga jendela aplikasi.

Pada praktikum ini, dibuat sebuah aplikasi sederhana bernama **Laundry Apps**. Aplikasi ini memiliki fitur login menggunakan username dan password, serta menampilkan menu utama dengan berbagai tombol navigasi seperti **Pesanan, Layanan, Pelanggan, Pengguna, Laporan, Profile**, dan **Keluar**. Program ini bertujuan melatih penerapan konsep PBO serta pemanfaatan komponen GUI pada bahasa pemrograman Java.

### 1.2 Tujuan

- Menerapkan konsep dasar Pemrograman Berorientasi Objek (PBO) dalam pembuatan aplikasi.
- Memahami penggunaan package untuk pemisahan model (User) dan antarmuka (LoginFrame, MainFrame).
- Mengimplementasikan GUI menggunakan Swing pada Java.
- Membuat fitur autentikasi login sederhana dengan validasi username dan password.
- Menampilkan menu utama aplikasi dengan beberapa tombol navigasi.

### 1.3 Alat dan Bahan

Alat dan bahan yang digunakan dalam praktikum ini adalah:

- **Perangkat Keras (Hardware):**

- Laptop/PC dengan sistem operasi Windows/Linux/macOS.
- RAM minimal 4 GB.
- **Perangkat Lunak (Software):**
  - Java Development Kit (JDK) versi 8 atau lebih baru.
  - IDE Eclipse untuk coding dan desain GUI.
  - Library bawaan Java Swing dan AWT untuk membangun GUI.

## **BAB II**

### **PEMBAHASAN**

#### **2.1 Langkah-langkah Praktikum dan Pembahasan Program**

##### **A. Package model: Class User.java**

###### **a) Tujuan Program:**

Class User digunakan untuk merepresentasikan data pengguna (user) pada sistem Laundry Apps. Class ini berfungsi sebagai model data dengan atribut dasar user (id, nama, username, password) serta method login untuk memvalidasi username dan password.

###### **b) Struktur dan Penjelasan Komponen Program:**

###### **1. Package model;**

Menyatakan bahwa class User berada di dalam package model.

###### **2. public class Costumer**

Mendefinisikan class dengan nama User.

###### **3. Atribut**

- String id → menyimpan ID pengguna.
- String nama → menyimpan nama pengguna.
- String username → menyimpan username untuk login.
- String password → menyimpan password untuk login.

###### **4. Getter & Setter:**

- getId(), setId(String id)
- getNama(), setNama(String nama)
- getUsername(), setUsername(String username)
- getPassword(), setPassword(String password). Method ini digunakan untuk mengakses dan mengubah nilai atribut secara aman.

###### **5. Method login(String username, String password):**

- Membuat objek User dengan data default (username = fulan, password = 12345).
- Memvalidasi input login dari GUI. Jika sesuai, return true, jika salah return false.

**c) Fitur Utama Program:**

- Menyediakan model data user.
- Memungkinkan validasi login sederhana tanpa database

```
package model;

public class User {
    String id, nama, username, password;

    public String getId() {
        return id;
    }

    public void setId(String id) {
        this.id = id;
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }
}
```

```

    }

    public static boolean login(String username, String password) {
        boolean isLoggin = false;
        User user = new User();
        user.setId("1");
        user.setNama("ima");
        user.setUsername("ima");
        user.setPassword("12345");

        if (user.getUsername().equalsIgnoreCase(username)
            && user.getPassword().equalsIgnoreCase(password)) {
            isLoggin = true;
        } else {
            isLoggin = false;
        }
        return isLoggin;
    }
}

```

## B. Package ui: Class LoginFrame.java

### a) Tujuan Program:

Class LoginFrame berfungsi sebagai tampilan awal (GUI) untuk login ke aplikasi Laundry Apps.

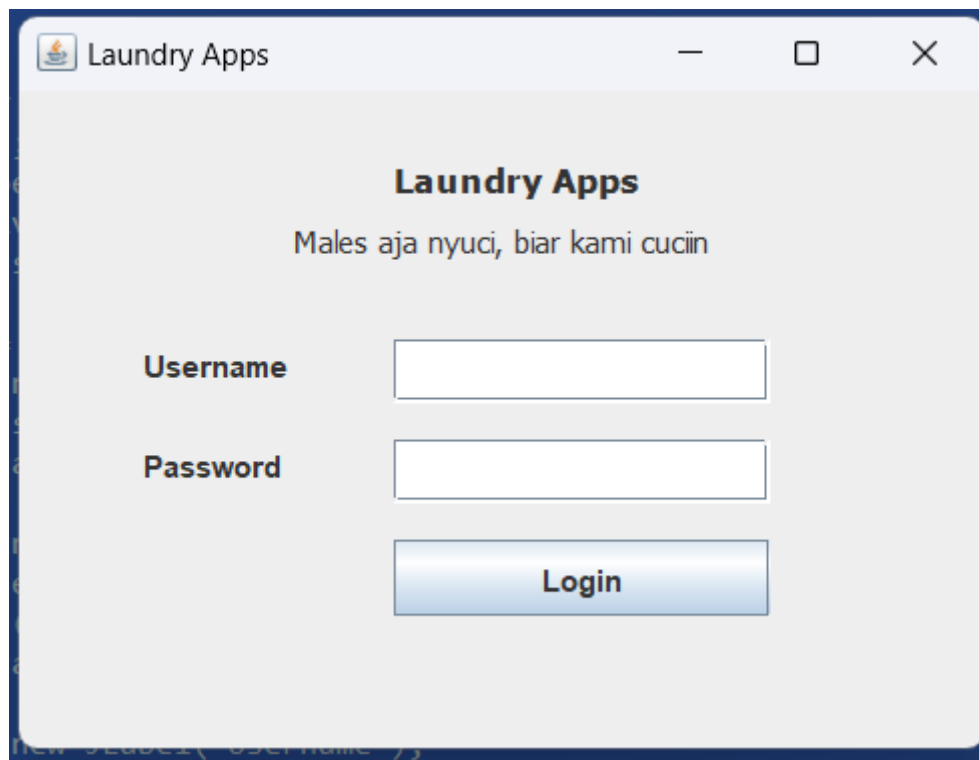
### b) Struktur dan Penjelasan Komponen Program:

1. Public class LoginFrame extends JFrame: Membuat GUI berbasis JFrame.
2. Komponen GUI:
  - JLabel lblTitle, lblDesc, lblUser, lblPass → Label judul, deskripsi, username, password.
  - JTextField txtUsername → Input untuk username.
  - JPasswordField txtPassword → Input untuk password.
  - JButton btnLogin → Tombol login.
3. Event Handling:
  - btnLogin.addActionListener(new ActionListener() {...})
  - → Saat tombol login ditekan, input username & password dibandingkan dengan data di User.login().
  - Jika benar → membuka MainFrame (halaman utama).

- Jika salah → menampilkan JOptionPane "Login Gagal!".

**c) Fitur Utama Program:**

- Tampilan login sederhana dengan validasi username dan password.
- Jika login sukses → masuk ke menu utama (MainFrame).



```
package ui;

import model.User;
import javax.swing.*;
import java.awt.event.*;
import java.awt.Font;

public class LoginFrame extends JFrame {
    private JTextField txtUsername;
    private JPasswordField txtPassword;
    private JButton btnLogin;

    public LoginFrame() {
        setTitle("Laundry Apps");
        setSize(400, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        getContentPane().setLayout(null);
    }
}
```



```

JLabel lblTitle = new JLabel("Laundry Apps");
lblTitle.setFont(new Font("Tahoma", Font.BOLD, 14));
lblTitle.setBounds(150, 20, 98, 30);
getContentPane().add(lblTitle);

JLabel lblDesc = new JLabel("Males aja nyuci, biar kami cuciin");
lblDesc.setFont(new Font("Tahoma", Font.PLAIN, 12));
lblDesc.setBounds(110, 50, 173, 20);
getContentPane().add(lblDesc);

JLabel lblUser = new JLabel("Username");
lblUser.setBounds(50, 100, 100, 20);
getContentPane().add(lblUser);

txtUsername = new JTextField();
txtUsername.setBounds(150, 100, 150, 25);
getContentPane().add(txtUsername);

JLabel lblPass = new JLabel("Password");
lblPass.setBounds(50, 140, 100, 20);
getContentPane().add(lblPass);

txtPassword = new JPasswordField();
txtPassword.setBounds(150, 140, 150, 25);
getContentPane().add(txtPassword);

btnLogin = new JButton("Login");
btnLogin.setBounds(150, 180, 150, 30);
getContentPane().add(btnLogin);

// Action Login
btnLogin.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String username = txtUsername.getText();
        String password = new String(txtPassword.getPassword());

        if (User.login(username, password)) {
            new MainFrame().setVisible(true);
            dispose(); // menutup login frame
        } else {
            JOptionPane.showMessageDialog(null, "Login Gagal!");
        }
    }
});
}

public static void main(String[] args) {
    new LoginFrame().setVisible(true);
}

```

```
}  
}
```

### **C. Package ui: Class MainFrame.java**

#### **a) Pemodelan Objek:**

Program ini menerapkan konsep **Object-Oriented Programming (OOP)** dengan pemisahan model (data) dan ui (tampilan/antarmuka pengguna).

#### **b) Konsep OOP yang digunakan:**

- Encapsulation: dengan penggunaan getter dan setter pada class User.
- Modularity: pemisahan kode menjadi package model dan ui.
- Inheritance: GUI dibuat dengan extends JFrame.

#### **c) Konsep GUI:**

- Menggunakan library Swing (JFrame, JLabel, JTextField, JPasswordField, JButton).
- Event handling dengan ActionListener untuk interaksi tombol.

#### **d) Alur Program:**

- User membuka aplikasi → tampil LoginFrame.
- User memasukkan username & password → validasi User.login().
- Jika valid → masuk MainFrame.
- Jika gagal → pesan "Login Gagal".
- Di MainFrame, user dapat memilih menu atau keluar dari aplikasi.



```
package ui;

import javax.swing.*.*;

public class MainFrame extends JFrame {

    public MainFrame() {
        setTitle("Laundry Apps - Halaman Utama");
        setSize(600, 400);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        setLayout(null);

        JLabel lblTitle = new JLabel("Laundry Apps");
        lblTitle.setBounds(230, 20, 200, 30);
        add(lblTitle);

        // Tombol menu
        JButton btnPesanan = new JButton("PESANAN");
        btnPesanan.setBounds(100, 80, 150, 40);
        add(btnPesanan);

        JButton btnLayanan = new JButton("LAYANAN");
        btnLayanan.setBounds(300, 80, 150, 40);
        add(btnLayanan);

        JButton btnPelanggan = new JButton("PELANGGAN");
        btnPelanggan.setBounds(100, 140, 150, 40);
        add(btnPelanggan);
    }
}
```

```
        JButton btnPengguna = new JButton("PENGGUNA");
        btnPengguna.setBounds(300, 140, 150, 40);
        add(btnPengguna);

        JButton btnLaporan = new JButton("LAPORAN");
        btnLaporan.setBounds(100, 200, 150, 40);
        add(btnLaporan);

        JButton btnProfile = new JButton("PROFILE");
        btnProfile.setBounds(300, 200, 150, 40);
        add(btnProfile);

        JButton btnKeluar = new JButton("KELUAR");
        btnKeluar.setBounds(200, 260, 150, 40);
        add(btnKeluar);

        btnKeluar.addActionListener(e -> {
            dispose(); // menutup aplikasi
        });
    }
}
```

## **BAB III**

### **PENUTUP**

#### **3.1 Kesimpulan**

Berdasarkan praktikum yang telah dilakukan, dapat disimpulkan bahwa:

1. Pemrograman Berorientasi Objek (PBO) dapat diaplikasikan secara nyata dalam pembuatan aplikasi sederhana, seperti Laundry Apps.
2. Pembuatan class dengan atribut dan method setter-getter memungkinkan data disimpan dalam bentuk object yang lebih terstruktur dan mudah diakses.
3. Implementasi GUI menggunakan Java Swing dengan JTabbedPane memberikan antarmuka yang lebih terorganisir, di mana data Customer, Service, dan Order dapat dikelola pada tab masing-masing.
4. Tombol **Simpan** pada GUI berhasil menghubungkan input dari pengguna dengan pembuatan object (Customer, Service, dan Order), sementara tombol **Reset** memudahkan penghapusan data input.
5. Praktikum ini memberikan pemahaman menyeluruh tentang bagaimana konsep OOP tidak hanya dipelajari secara teori, tetapi juga dapat diintegrasikan dengan tampilan antarmuka pengguna untuk aplikasi sehari-hari.