

**LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN**  
**“Pembuatan Aplikasi Operator Aritmatika menggunakan WindowBuilder di**  
**Java”**

**disusun Oleh:**

**Az Zahrand Solichul Tajussalathin**

**NIM 2511532001**

**Dosen Pengampu: Dr. Wahyudi S.T M.T**

**Asisten Praktikum: Aufan Taufiqurrahman**



**FAKULTAS TEKNOLOGI INFORMASI**

**DEPARTEMEN INFORMATIKA**

**UNIVERSITAS ANDALAS**

**2025**

## DAFTAR ISI

<b>DAFTAR ISI.....</b>	<b>i</b>
<b>KATA PENGANTAR .....</b>	<b>ii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
<b>    1.1 Latar Belakang.....</b>	<b>1</b>
<b>    1.2 Tujuan Praktikum.....</b>	<b>1</b>
<b>    1.3 Manfaat Praktikum .....</b>	<b>2</b>
<b>BAB II PEMBAHASAN.....</b>	<b>3</b>
<b>    2.1 Program OperatorAritmatika .....</b>	<b>3</b>
<b>BAB III KESIMPULAN.....</b>	<b>6</b>
<b>    3.1 Ringkasan Hasil Praktikum.....</b>	<b>6</b>
<b>DAFTAR PUSTAKA.....</b>	<b>iii</b>

## KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas limpahan rahmat dan karunia-Nya sehingga laporan praktikum dengan judul “Pembuatan Aplikasi Operator Aritmatika menggunakan WindowBuilder di Java” ini dapat disusun dan diselesaikan dengan baik. Laporan ini disusun sebagai salah satu bentuk pelaksanaan kegiatan praktikum pada mata kuliah Algoritma dan Pemrograman.

Ucapan terima kasih disampaikan kepada Bapak Dr. Wahyudi, S.T., M.T. selaku dosen pengampu, serta kepada asisten laboratorium yang telah memberikan bimbingan selama kegiatan praktikum berlangsung.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, baik dari segi penulisan maupun isi. Oleh karena itu, saran dan kritik yang bersifat membangun sangat diharapkan demi perbaikan di masa yang akan datang. Harapannya, laporan ini dapat memberikan manfaat dan menjadi referensi bagi pembelajaran konsep perulangan dalam bahasa pemrograman Java.

## BAB 1

### PENDAHULUAN

#### 1.1 Latar Belakang

Perkembangan aplikasi berbasis Graphical User Interface (GUI) telah menjadi bagian penting dalam dunia pemrograman modern. Tidak hanya digunakan untuk aplikasi besar, konsep GUI juga diterapkan pada program sederhana untuk membuat interaksi pengguna menjadi lebih mudah. Dalam bahasa pemrograman Java, pembuatan GUI dapat dilakukan menggunakan WindowBuilder, yaitu plugin visual yang memungkinkan pengembang merancang tampilan aplikasi secara langsung tanpa menuliskan seluruh kode secara manual.

Pada praktikum ini, diterapkan penggunaan WindowBuilder untuk membuat aplikasi sederhana berupa kalkulator operator aritmatika dasar. Program ini menggabungkan konsep GUI dengan logika pemrograman dasar melalui penggunaan text field, combo box, tombol aksi, serta metode event handling.

#### 1.2 Tujuan Praktikum

Tujuan dari praktikum ini adalah:

1. Memahami dasar penggunaan Java Swing untuk membuat antarmuka grafis (GUI).
2. Menggunakan WindowBuilder sebagai alat bantu dalam merancang tampilan aplikasi secara visual.
3. Mengimplementasikan komponen GUI seperti label, text field, combo box, dan button ke dalam sebuah program Java.
4. Menerapkan konsep event handling melalui penggunaan ActionListener pada tombol aksi.
5. Menerapkan teknik validasi input untuk mencegah kesalahan saat program dijalankan.

### **1.3 Manfaat Praktikum**

Manfaat yang diperoleh dari praktikum ini antara lain:

1. Menambah pemahaman mengenai cara membuat aplikasi desktop sederhana menggunakan Java Swing.
2. Melatih kemampuan dalam merancang tampilan aplikasi menggunakan WindowBuilder.
3. Memberikan pengalaman dalam menerapkan logika program pada antarmuka grafis.

## BAB 2

### PEMBAHASAN

#### 2.1 Program Bilangan Prima

```

1 package pekan8_2511532001;
2
3 import java.awt.EventQueue;
4
5 import javax.swing.JFrame;
6 import javax.swing.JPanel;
7 import javax.swing.border.EmptyBorder;
8 import javax.swing.JLabel;
9 import javax.swing.JOptionPane;
10
11 import java.awt.Font;
12 import javax.swing.JTextField;
13 import javax.swing.JComboBox;
14 import javax.swing.JButton;
15 import javax.swing.DefaultComboBoxModel;
16 import java.awt.event.ActionListener;
17 import java.awt.event.ActionEvent;
18
19 public class OperatorAritmatika_2511532001 extends JFrame {
20
21     private static final long serialVersionUID = 1L;
22     private JPanel contentPane;
23     private JTextField txtBil1;
24     private JTextField txtBil2;
25     private JTextField txtHasil;
26
27     private void pesanPeringatan(String pesan) {
28         JOptionPane.showMessageDialog(this, pesan, "Peringatan", JOptionPane.WARNING_MESSAGE);
29     }
30     private void pesanError(String pesan) {
31         JOptionPane.showMessageDialog(this, pesan, "Kesalahan", JOptionPane.WARNING_MESSAGE);
32     }
33     /**
34      * Launch the application.
35      */
36     public static void main(String[] args) {
37         EventQueue.invokeLater(new Runnable() {
38             public void run() {
39                 try {
40                     OperatorAritmatika_2511532001 frame = new OperatorAritmatika_2511532001();
41                     frame.setVisible(true);
42                 } catch (Exception e) {
43                     e.printStackTrace();
44                 }
45             }
46         });
47     }
48
49     /**
50      * Create the frame.
51      */
52     public OperatorAritmatika_2511532001() {
53         setResizable(false);
54         setTitle("Operator Aritmatika");
55         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
56         setBounds(100, 100, 497, 416);
57         contentPane = new JPanel();
58         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
59         setContentPane(contentPane);
60         contentPane.setLayout(null);
61
62         JLabel lblNewLabel = new JLabel("OPERATOR ARITMATIKA");
63         lblNewLabel.setFont(new Font("Segoe UI Black", Font.PLAIN, 14));
64         lblNewLabel.setBounds(158, 10, 188, 55);
65         contentPane.add(lblNewLabel);
66
67         JLabel lblNewLabel_1 = new JLabel("Bilangan 1");
68         lblNewLabel_1.setBounds(55, 62, 76, 30);
69         contentPane.add(lblNewLabel_1);
70
71         JLabel lblNewLabel_1_1 = new JLabel("Bilangan 2");
72         lblNewLabel_1_1.setBounds(55, 103, 76, 30);
73         contentPane.add(lblNewLabel_1_1);
74
75         JLabel lblNewLabel_1_1_1 = new JLabel("Operator");
76         lblNewLabel_1_1_1.setBounds(55, 152, 76, 30);
77         contentPane.add(lblNewLabel_1_1_1);
78
79         JLabel lblNewLabel_1_1_1_1 = new JLabel("Hasil");
80         lblNewLabel_1_1_1_1.setBounds(55, 192, 76, 30);
81         contentPane.add(lblNewLabel_1_1_1_1);
82
83         txtBil1 = new JTextField();
84         txtBil1.setBounds(130, 68, 64, 19);
85         contentPane.add(txtBil1);
86         txtBil1.setColumns(10);
87
88         txtBil2 = new JTextField();
89         txtBil2.setColumns(10);
90         txtBil2.setBounds(130, 109, 64, 19);
91         contentPane.add(txtBil2);
92
93         txtHasil = new JTextField();
94         txtHasil.setEditable(false);
95         txtHasil.setColumns(10);
96         txtHasil.setBounds(130, 198, 64, 19);
97         contentPane.add(txtHasil);
98
99         JComboBox cbOperator = new JComboBox();
100         cbOperator.setModel(new DefaultComboBoxModel(new String[] {"+", "-", "*", "/", "%"}));
101         cbOperator.setBounds(130, 157, 64, 21);
102         contentPane.add(cbOperator);
103
104         JButton btnNewButton = new JButton("Proses");
105         btnNewButton.addActionListener(new ActionListener() {
106             int hasil;
107             public void actionPerformed(ActionEvent arg0) {
108                 if(txtBil1.getText().trim().isEmpty()) {
109                     pesanPeringatan ("Silahkan Input Bilangan 1");
110                 } else if(txtBil2.getText().trim().isEmpty()) {
111                     pesanPeringatan ("Silahkan Input Bilangan 2");
112                 } else {
113                     try {
114                         int a = Integer.parseInt(txtBil1.getText());
115                         int b = Integer.parseInt(txtBil2.getText());
116                         int c = 0;
117                         if(cbOperator.getSelectedItem().equals("+")) {
118                             c = a + b;
119                         } else if(cbOperator.getSelectedItem().equals("-")) {
120                             c = a - b;
121                         } else if(cbOperator.getSelectedItem().equals("*")) {
122                             c = a * b;
123                         } else if(cbOperator.getSelectedItem().equals("/")) {
124                             c = a / b;
125                         } else if(cbOperator.getSelectedItem().equals("%")) {
126                             c = a % b;
127                         }
128                         txtHasil.setText(Integer.toString(c));
129                     } catch (Exception e) {
130                         pesanError("Input yang anda masukkan salah");
131                     }
132                 }
133             }
134         });
135     }
136
137     JButton btnNewButton_1 = new JButton("Batal");
138     btnNewButton_1.addActionListener(new ActionListener() {
139         public void actionPerformed(ActionEvent arg0) {
140             dispose();
141         }
142     });
143
144     JButton btnNewButton_2 = new JButton("Exit");
145     btnNewButton_2.addActionListener(new ActionListener() {
146         public void actionPerformed(ActionEvent arg0) {
147             System.exit(0);
148         }
149     });
150
151     JPanel panel = new JPanel();
152     panel.setLayout(null);
153     panel.add(btnNewButton);
154     panel.add(btnNewButton_1);
155     panel.add(btnNewButton_2);
156     contentPane.add(panel);
157
158     pack();
159     setLocationRelativeTo(null);
160     setVisible(true);
161
162 }

```

```
58 setContentPane(contentPane);
59 contentPane.setLayout(null);
60
61 JLabel lblNewLabel = new JLabel("OPERATOR ARITMATIKA");
62 lblNewLabel.setFont(new Font("Segoe UI Black", Font.PLAIN, 14));
63 lblNewLabel.setBounds(158, 10, 188, 55);
64 contentPane.add(lblNewLabel);
65
66
67 JLabel lblNewLabel_1 = new JLabel("Bilangan 1");
68 lblNewLabel_1.setBounds(55, 62, 76, 30);
69 contentPane.add(lblNewLabel_1);
70
71 JLabel lblNewLabel_1_1 = new JLabel("Bilangan 2");
72 lblNewLabel_1_1.setBounds(55, 103, 76, 30);
73 contentPane.add(lblNewLabel_1_1);
74
75 JLabel lblNewLabel_1_1_1 = new JLabel("Operator");
76 lblNewLabel_1_1_1.setBounds(55, 152, 76, 30);
77 contentPane.add(lblNewLabel_1_1_1);
78
79 JLabel lblNewLabel_1_1_1_1 = new JLabel("Hasil");
80 lblNewLabel_1_1_1_1.setBounds(55, 192, 76, 30);
81 contentPane.add(lblNewLabel_1_1_1_1);
82
83 JTextField txtBill1 = new JTextField();
84 txtBill1.setBounds(130, 68, 64, 19);
85 contentPane.add(txtBill1);
86 txtBill1.setColumns(10);
87
88 JTextField txtBill2 = new JTextField();
89 txtBill2.setColumns(10);
90 txtBill2.setBounds(130, 109, 64, 19);
91 contentPane.add(txtBill2);
92
93 JTextField txtHasil = new JTextField();
94 txtHasil.setEditable(false);
95 txtHasil.setColumns(10);
96 txtHasil.setBounds(130, 198, 64, 19);
97 contentPane.add(txtHasil);
98
99 JComboBox cbOperator = new JComboBox();
100 cbOperator.setModel(new DefaultComboBoxModel(new String[] {"+", "-", "*", "/", "%"}));
101 cbOperator.setBounds(130, 157, 64, 21);
102 contentPane.add(cbOperator);
103
104 JButton btnNewButton = new JButton("Proses");
105 btnNewButton.addActionListener(new ActionListener() {
106     int hasil;
107     public void actionPerformed(ActionEvent arg0) {
108         if((txtBill1.getText().trim()).isEmpty()) {
109             pesanPeringatan ("Silahkan Input Bilangan 1");
110         } else if((txtBill2.getText().trim()).isEmpty()) {
111             pesanPeringatan ("Silahkan Input Bilangan 2");
112         } else {
113             try {
114                 int a = Integer.parseInt(txtBill1.getText());
115                 int b = Integer.parseInt(txtBill2.getText());
116
117                 if(cbOperator.getSelectedItem().equals("+")) {
118                     hasil = a + b;
119                 } else if(cbOperator.getSelectedItem().equals("-")) {
120                     hasil = a - b;
121                 } else if(cbOperator.getSelectedItem().equals("*")) {
122                     hasil = a * b;
123                 } else if(cbOperator.getSelectedItem().equals("/")) {
124                     hasil = a / b;
125                 } else if(cbOperator.getSelectedItem().equals("%")) {
126                     hasil = a % b;
127                 }
128
129                 txtHasil.setText(String.valueOf(hasil));
130             } catch (Exception e) {
131                 pesanPeringatan ("Input yang anda masukkan salah");
132             }
133         }
134     }
135 }
```

Program ini merupakan sebuah aplikasi berbasis GUI (Graphical User Interface) yang dibuat menggunakan Java Swing melalui WindowBuilder. Program ini dibuat untuk melakukan operasi aritmatika sederhana antara dua bilangan yang dimasukkan pengguna. Tampilan program yang digunakan terdiri dari beberapa komponen seperti label, text field, combo box, dan tombol proses yang diletakkan di dalam sebuah JFrame. Cara kerja program ini cukup mudah. Pengguna pertama tama memasukkan dua bilangan pada text field yang tersedia, memilih jenis operator melalui combo box, kemudian menekan tombol “Proses” untuk mendapatkan hasil perhitungan.

Pada bagian inisialisasi program, program mengatur seluruh komponen GUI dan menentukan posisi serta fungsinya. Dua text field, yaitu txtBil1 dan txtBil2, digunakan untuk menerima input angka dari pengguna. Sementara itu, txtHasil digunakan untuk menampilkan hasil perhitungan dan dibuat non-editable untuk mencegah pengguna mengubah hasil secara manual. Combo box cbOperator berisi lima operator aritmatika, yaitu penjumlahan, pengurangan, perkalian, pembagian, dan

modulus. Pilihan yang dipilih dari dropdown akan menentukan jenis perhitungan yang akan dijalankan ketika tombol proses diklik.

Ketika tombol “Proses” ditekan, program akan melakukan pengecekan terlebih dahulu untuk memastikan bahwa kedua input bilangan tidak kosong. Jika ada text field yang belum terisi, program menampilkan pesan peringatan melalui method pesanPeringatan(). Selanjutnya, program akan mengonversi nilai input menjadi integer menggunakan Integer.parseInt(). Jika input tidak dapat dikonversi karena berisi karakter non-angka, program menampilkan pesan kesalahan melalui method pesanError(). Jika kedua input valid, program membaca indeks operator yang dipilih dari dropdown, melakukan perhitungan sesuai operator tersebut, dan hasilnya ditampilkan pada kolom txtHasil.

## BAB 3

### KESIMPULAN

#### 3.1 Ringkasan Hasil Praktikum

Berdasarkan praktikum yang telah dilakukan, dapat disimpulkan bahwa penggunaan Java Swing melalui WindowBuilder sangat kemudahan dalam membuat antarmuka grafis yang interaktif. Penerapan berbagai komponen GUI seperti label, text field, combo box, dan button memungkinkan pembuatan aplikasi operator aritmatika yang dapat menerima input, memproses data, serta menampilkan hasil secara langsung. Melalui praktikum ini juga dipahami bahwa mekanisme event handling, khususnya penggunaan ActionListener, memiliki peran penting dalam menghubungkan antarmuka dengan logika program. Selain itu, proses validasi input melalui pesan peringatan dan pesan kesalahan meningkatkan kualitas program saat digunakan.

Sebagai pengembangan lebih lanjut, aplikasi masih bisa ditingkatkan dengan menambahkan fitur tambahan seperti tombol reset untuk mengosongkan input, kemampuan menghitung angka desimal atau pecahan, maupun tata letak antarmuka yang lebih responsif agar tampilan terlihat lebih profesional. Pembuatan aplikasi berbasis desktop menggunakan Java Swing dapat menjadi keterampilan yang bermanfaat dalam proses pembelajaran maupun pengembangan software di masa depan.

## **DAFTAR PUSTAKA**

[1] Oracle, “Creating a GUI With Swing,” *The Java™ Tutorials*, 2023. [Daring].

Tersedia pada: <https://docs.oracle.com/javase/tutorial/uiswing/>.

[Diakses: 18 Nov 2025].

[2] Oracle, “Class JOptionPane,” *Java Platform SE Documentation*, 2023. [Daring].

Tersedia pada:

<https://docs.oracle.com/en/java/javase/17/docs/api/java.desktop/javax/swing/JOptionPane.html>.

[Diakses: 18 Nov 2025].