

**LAPORAN PRAKTIKUM ALGORITMA PEMROGRAMAN
WINDOW BUILDER**



disusun Oleh:

Tiara Amalia Insani

NIM 2511532019

Dosen Pengampu : DR. Wahyudi, S.T, M.T

Asisten Praktikum : Jovantri Immanuel Gulo

**DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
PADANG
2025**

KATA PENGANTAR

Pedoman ini disusun sebagai rujukan resmi bagi mahasiswa Departemen Informatika dalam penyusunan laporan praktikum pada mata kuliah Pemrograman Dasar dengan Java. Dokumen ini tidak hanya memberikan gambaran umum mengenai format penulisan, tetapi juga menguraikan secara rinci sistematika laporan, tata cara penyajian isi, serta contoh penulisan kode program yang dilengkapi dengan referensi ilmiah. Melalui panduan ini, mahasiswa diharapkan mampu menyusun laporan yang tidak sekadar memenuhi aspek administratif, tetapi juga mencerminkan ketelitian, keteraturan, dan penerapan kaidah penulisan akademik pada tingkat dasar. Dengan demikian, laporan praktikum yang dihasilkan dapat berfungsi sebagai media pembelajaran, dokumentasi kegiatan, sekaligus sarana untuk melatih keterampilan menulis ilmiah yang akan bermanfaat dalam jenjang studi selanjutnya.

Padang, 19 November 2025

Penyusun

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR.....	iii
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Tujuan Praktikum.....	1
1.3 Manfaat.....	1
BAB II	2
PEMBAHASAN	2
2.1 WindowBuilder	2
2.2 Praktikum.....	2
2.3.1 Alat dan Bahan	2
2.3.2 Program.....	3
BAB III.....	8
KESIMPULAN DAN SARAN	8
3.1 Kesimpulan	8
3.2 Saran	8
DAFTAR PUSTAKA	9

DAFTAR GAMBAR

Gambar 1 3

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam pemrograman, WindowBuilder dipelajari karena dapat membantu proses pembuatan antarmuka grafis (GUI) pada java. Dengan menggunakan window builder pembuatan tampilan menjadi lebih mudah dan cepat karna tidak perlu menulis seluruh kode GUI secara manual. Selain itu WindowBuilder memberikan pratinjau terhadap hasil desain sehingga proses pengembangan dan pengecekan tampilan menjadi jauh lebih mudah.

1.2 Tujuan Praktikum

Tujuan dari pelaksanaan praktikum antara lain sebagai berikut:

1. Membantu mahasiswa memahami konsep dasar pemrograman Java melalui penerapan langsung.
2. Melatih kemampuan menulis, mengompilasi, dan mengeksekusi program dengan mengikuti aturan *sintaksis* Java
3. Membiasakan mahasiswa bekerja sistematis dalam menyusun laporan yang memuat analisis hasil praktikum.
4. Menanamkan sikap teliti, disiplin, serta tanggung jawab dalam melaksanakan kegiatan laboratorium.
5. Memahami cara menggunakan WindowBuilder sebagai alat bantu dalam merancang antarmuka grafis (GUI) di java.
6. Mengenali kode yang dihasilkan secara otomatis oleh WindowBuilder.

1.3 Manfaat

Mahasiswa diharapkan memahami cara penggunaan window builder dan lebih cepat memahami cara memposisikan komponen GUI, meningkatkan ketelitian, dan memudahkan mahasiswa saat membuat program yang membutuhkan antarmuka grafis.

BAB II

PEMBAHASAN

2.1 WindowBuilder

WindowBuilder adalah sebuah alat bantu yang digunakan untuk membuat tampilan antarmuka grafis (GUI) pada java dengan lebih mudah dan lebih cepat. WindowBuilder membuat GUI melalui proses drag and drop sehingga pengguna tidak perlu menulis seluruh kode secara manual. Saat menggunakan WindowBuilder komponen komponen seperti tombol, label, text field, panel dan lainnya dapat dipilih langsung. Komponen tersebut kemudian dapat ditempatkan ke dalam form dan diatur ukurannya sesuai kebutuhan. Setelah pengguna melakukan perubahan pada tampilan, WindowBuilder secara otomatis menghasilkan kode java

Keunggulan dari windowBuilder lainnya adalah mahasiswa bisa langsung melihat perubahan tampilan tanpa harus terlebih dahulu menjalankan program, hal ini dapat meminimalkan kesalahan dalam pengaturan layout.

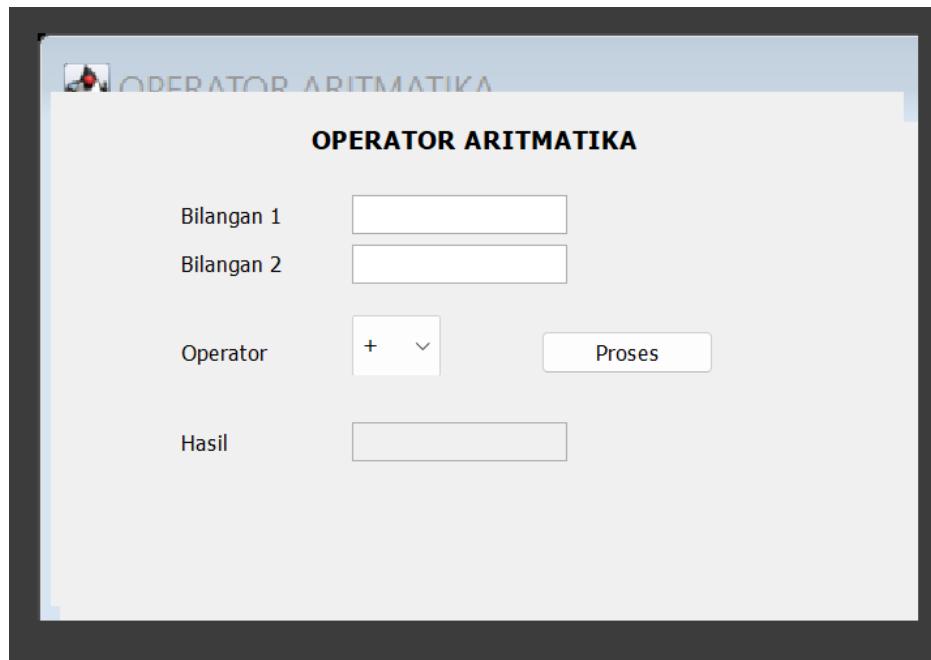
2.2 Praktikum

2.3.1 Alat dan Bahan

1. Laptop/computer dengan eclipse dan WindowBuilder terinstal.
2. Materi praktikum mengenai WindowBuilder

2.3.2 Program

1. BilanganPrima



```
package pekan8_2511532019;

import java.awt.BorderLayout;

public class OperatorAritmatikaGUI_2511532019 extends JFrame {

    private static final long serialVersionUID = 1L;
    private JPanel contentPane;
    private JTextField txtBil1;
    private JTextField txtBil2;
    private JTextField txtHasil;

    private void pesanPeringatan(String pesan) {
        JOptionPane.showMessageDialog(this, pesan, "Peringatan", JOptionPane.WARNING_MESSAGE);
    }
    private void pesanError(String pesan) {
        JOptionPane.showMessageDialog(this, pesan, "Kesalahan", JOptionPane.ERROR_MESSAGE);
    }

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    OperatorAritmatikaGUI_2511532019 frame = new OperatorAritmatikaGUI_2511532019();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public OperatorAritmatikaGUI_2511532019() {
        setResizable(false);
        setTitle("OPERATOR ARITMATIKA");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 450, 300);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);
    }
}
```

```

JLabel lblNewLabel_1 = new JLabel("");
lblNewLabel_1.setBounds(154, 17, 0, 0);
contentPane.add(lblNewLabel_1);

JLabel lblNewLabel = new JLabel("OPERATOR ARITMATIKA");
lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 13));
lblNewLabel.setBounds(133, 17, 184, 14);
contentPane.add(lblNewLabel);

JLabel lblBilangan = new JLabel("Bilangan 1");
lblBilangan.setBounds(66, 56, 127, 14);
contentPane.add(lblBilangan);

JLabel lblBilangan_2 = new JLabel("Bilangan 2");
lblBilangan_2.setBounds(66, 81, 127, 14);
contentPane.add(lblBilangan_2);

JLabel lblBilangan_2_1 = new JLabel("Operator");
lblBilangan_2_1.setBounds(66, 126, 71, 14);
contentPane.add(lblBilangan_2_1);

JLabel lblBilangan_2_1_1 = new JLabel("Hasil");
lblBilangan_2_1_1.setBounds(66, 172, 71, 14);
contentPane.add(lblBilangan_2_1_1);

txtBill = new JTextField();
txtBill.setBounds(154, 53, 110, 20);
contentPane.add(txtBill);
txtBill.setColumns(10);

JComboBox cbOperator = new JComboBox();
JButton Proses = new JButton("Proses");
Proses.addActionListener(new ActionListener() {
    int hasil;
    public void actionPerformed(ActionEvent e) {
        if(txtBill.getText().trim().isEmpty()) {
            pesanPeringatan ("Silahkan Input Bilangan 1");
        }else if (txtBill2.getText().trim().isEmpty()) {
            pesanPeringatan("Silahkan Input Bilangan 2");
        }else if(txtBill2.getText().trim().startsWith("0")) {
            pesanPeringatan("Bilangan 2 tidak boleh angka 0");
        }else {
            try {
                int a = Integer.parseInt(txtBill.getText());
                int b = Integer.parseInt(txtBill2.getText());
                int c = cbOperator.getSelectedIndex();
                if (c==0) {hasil = a+b;}
                if (c==1) {hasil = a-b;}
                if (c==2) {hasil = a*b;}
                if (c==3) {hasil= a/b;}
                if (c==4) {hasil = a%b;}
            }
            txtHasil.setText(String.valueOf(hasil));
        }
    }
});

```

```

        }catch (NumberFormatException ex) {
            pesanError ("Bilangan 1 dan Bilangan 2 harus Angka");
        }
    }
    int a = Integer.parseInt(txtBil1.getText());
    int b = Integer.parseInt(txtBil2.getText());
    int c = cbOperator.getSelectedIndex();
    if (c==0) {hasil = a+b;}
    if (c==1) {hasil = a-b;}
    if (c==2) {hasil = a*b;}
    if (c==3) {hasil = a/b;}
    if (c==4) {hasil = a%b;}

    txtHasil.setText(String.valueOf(hasil));
}
});  

Proses.setBounds(250, 122, 88, 22);
contentPane.add(Proses);

txtBil2 = new JTextField();
txtBil2.setColumns(10);
txtBil2.setBounds(154, 78, 110, 20);
contentPane.add(txtBil2);

cbOperator.setModel(new DefaultComboBoxModel(new String[] {"+", "-", "*", "/", "%"}));
cbOperator.setBounds(154, 114, 45, 31);
contentPane.add(cbOperator);

txtHasil = new JTextField();
txtHasil.setEditable(false);
txtHasil.setColumns(10);
txtHasil.setBounds(154, 169, 110, 20);
contentPane.add(txtHasil);

}
}

```

Gambar 1

Langkah Langkah

1. Pilih desain
2. Klik (javax.swing.JFrame)-Operator aritmatika
3. Di properties beri title “OPERATOR ARITMATIKA”
4. Pada pallete, klik JLabel lalu di properties beri text “Bilangan 1”, font Tahoma size 13, bold.
5. Pada pallete, klik JTextField untuk nantinya digunakan menerima input pengguna, di properties ubah variabel menjadi “txtBil1”
6. Lalu ulangi Langkah 5 dan 6 untuk bilangan 2, nama variabel diganti menjadi “txtBil2”
7. Tambahkan JLabel dan beri text “Operator”
8. Klik JComboBox, dan isi model (+,-,*,/,%)
9. Klik JButton dan beri text “Proses”

10. Buat JLabel baru, beri text “hasil”
11. Tambahkan JTextField dan ubah variabel menjadi “txtHasil”
12. Klik 2 kali pada proses maka secara otomatis code akan dibuat
13. Tambahkan kode dibawah ini sebagai methode yang bisa dipanggil kapan saja, berfungsi menampilkan pesan peringatan dan error

```
private void pesanPeringatan(String pesan) {
    JOptionPane.showMessageDialog(this, pesan,
    "Peringatan", JOptionPane.WARNING_MESSAGE);
}
private void pesanError(String pesan) {
    JOptionPane.showMessageDialog(this, pesan,
    "Kesalahan", JOptionPane.ERROR_MESSAGE);}
```

14. Kode dibawah ini berguna untuk

```
JComboBox cbOperator = new JComboBox();
JButton Proses = new JButton("Proses");
Proses.addActionListener(new ActionListener() {
    int hasil;
    public void actionPerformed(ActionEvent e) {
        if(txtBil1.getText().trim().isEmpty()) {
            pesanPeringatan ("Silahkan Input Bilangan 1");
        }else if (txtBil2.getText().trim().isEmpty()) {
            pesanPeringatan("Silahkan Input Bilangan 2");
        }else if(txtBil2.getText().trim().startsWith("0")) {
            pesanPeringatan("Bilangan 2 tidak boleh angka 0");
        }else {
            try {
                int a = Integer.parseInt(txtBil1.getText());
                int b = Integer.parseInt(txtBil2.getText());
                int c = cbOperator.getSelectedIndex();
                if (c==0) {hasil = a+b;}
                if (c==1) {hasil = a-b;}
                if (c==2) {hasil = a*b;}
                if (c==3) {hasil= a/b;}
                if (c==4) {hasil = a%b;}

                txtHasil.setText(String.valueOf(hasil));
            }catch (NumberFormatException ex) {
                pesanError ("Bilangan 1 dan Bilangan 2 harus Angka");
            }
            int a = Integer.parseInt(txtBil1.getText());
            int b = Integer.parseInt(txtBil2.getText());
            int c = cbOperator.getSelectedIndex();
            if (c==0) {hasil = a+b;}
            if (c==1) {hasil = a-b;}
            if (c==2) {hasil = a*b;}
            if (c==3) {hasil= a/b;}
            if (c==4) {hasil = a%b;}}
```

```
        txtHasil.setText(String.valueOf(hasil));
    }
JComboBox cbOperator = new JComboBox();
 JButton Proses = new JButton("Proses");
cbOperator = tempat memilih operator (+,-,*,/,%)
```

proses = untuk menjalankan perhitungan.

```
Proses.addActionListener(new ActionListener()) {
ActionListener= kode yang dijalankan saat tombol "proses" di klik
```

```
if(txtBil1.getText().trim().isEmpty() {
    pesanPeringatan ("Silahkan Input Bilangan 1");
}else if (txtBil2.getText().trim().isEmpty()) {
    pesanPeringatan("Silahkan Input Bilangan 2");}
```

mengecek apakah bilangan 1 atau bilangan 2 masih kosong jika ya maka

tampilkan pesan peringatan

```
else if(txtBil2.getText().trim().startsWith("0")) {
    pesanPeringatan("Bilangan 2 tidak boleh angka 0");}
```

mencegah input yang dimulai dengan angka 0

```
try {
    int a = Integer.parseInt(txtBil1.getText());
    int b = Integer.parseInt(txtBil2.getText());
    int c = cbOperator.getSelectedIndex();
```

mengubah input text menjadi angka (integer.parseInt) dan mengambil operator yang dipilih dari JComboBox

```
if (c==0) {hasil = a+b;}
if (c==1) {hasil = a-b;}
if (c==2) {hasil = a*b;}
if (c==3) {hasil= a/b;}
if (c==4) {hasil = a%b;}
```

jika operator pilihan adalah 0 (+), 1(-), 2(*), 3(/), 4(%) hasil akan sesuai operasi yang dipilih

```
txtHasil.setText(String.valueOf(hasil));
```

mengubah hasil (angka) menjadi String lalu menampilkan ke txtHasil

```
}catch (NumberFormatException ex) {
    pesanError ("Bilangan 1 dan Bilangan 2 harus Angka");
}
```

Jika input berisi huruf atau simbol maka pesan error akan ditampilkan

BAB III

KESIMPULAN DAN SARAN

3.1 Kesimpulan

Dari praktikum ini dapat disimpulkan bahwa penggunaan WindowBuilder sangat membantu mahasiswa membuat GUI pada java karena proses desain dilakukan secara langsung melalui fitur drag and drop. WindowBuilder juga Memberikan pratinjau sehingga mempermudah pengaturan layout dan membantu memahami kode otomatis yang dihasilkan

3.2 Saran

1. Praktikum berikutnya sebaiknya dilengkapi dengan lebih banyak contoh program nyata (kasus sehari-hari) supaya lebih mudah memahami penerapan tipe data dalam menyelesaikan masalah.
2. Materi pembelajaran sebaiknya diberikan sebelum praktikum dimulai beserta flowchart dan pseudocode nya, agar dapat dipahami terlebih dahulu, sehingga pada saat praktikum lebih mudah mengerti.
3. Latihan mandiri akan membantu mahasiswa memahami cara program secara mendalam

DAFTAR PUSTAKA

- Dokumen Institusi
 - [1] Departemen Informatika, *Pedoman Penulisan Laporan Praktikum Java*. Padang: Universitas Andalas,2025.
- Buku
 - [2] P. Deitel and H. Deitel, *Java: How to Program*, 10th ed.
 - [3] F. Sujarwadi, *Pemrograman GUI dengan Java Swing*.
- Sumber daring
 - [4] Eclipse Foundation, “Eclipse WindowBuilder.” [Online]. Available: <https://eclipse.dev/windowbuilder/>
Accessed: 20-Nov-2025.