

LAPORAN PRATIKUM
PEMROGRAMAN ALGORITMA DAN PEMROGRAMAN
“OPERATOR ARITMATIKA MENGUUNAKAN GUI PADA JAVA”
disusun Oleh:
AFIF RAHMAN SALEH
NIM 251153105
Dosen Pengampu: Dr. WAHYUDI, S.T, M.T
Asisten Pratikum: AUFAN TAUFIQURRAHAMAN



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
TAHUN
2025

DAFTAR ISI

DAFTAR ISI	i
KATA PENGANTAR.....	ii
BAB I	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan.....	1
1.3 Manfaat.....	2
BAB II.....	3
PEMBAHASAN	3
2.1 Penjelasan Langkah-Langkah	3
2.2 Analisis Hasil	4
BAB III.....	7
KSIMPULAN	7
3.1 Kesimpulan	7
3.2 Saran.....	7
DAFTAR PUSTAKA.....	8

KATA PENGANTAR

Puji syukur kami panjatkan atas kehadirat Tuhan Yang Maha Esa atas rahmat dan karunia-Nya laporan praktikum ini dapat diselesaikan dengan baik. Laporan ini disusun sebagai bentuk penugasan dalam kegiatan praktikum mata kuliah Algoritma dan Pemograman khususnya pada materi “Operator Aritmatika Menggunakan GUI Pada Java “.

Dalam penyusunan laporan ini, penulis berusaha untuk menjelaskan secara sistematis mulai dari latar belakang, tujuan, dan manfaat praktikum, hingga pembahasan kode program, analisis hasil, serta kesimpulan dan saran untuk pengembangan lebih lanjut. Materi yang dibahas diharapkan dapat menjadi referensi bagi mahasiswa atau praktikan dalam memahami dan mengimplementasikan algoritma perulangan secara efektif.

Padang, 2025

Penulis

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini User Interface (UI) sangat banyak digunakan dalam aplikasi komputer. Aplikasi yang memiliki User Interface cenderung lebih disukai oleh pengguna, karena dapat memudahkan pembuatan aplikasi tanpa menghabiskan banyak waktu menulis dalam kode. Salah satu UI pada bahasa pemrograman yang kita pelajari saat ini adalah Window Builder.

Window Builder adalah sebuah plugin untuk IDE Eclipse yang memungkinkan pengembang untuk merancang antarmuka grafis secara drag-and-drop. Window Builder menyediakan editor visual yang memungkinkan pengguna untuk menempatkan komponen GUI (seperti tombol, text field, combo box, dan label) pada form secara langsung, sambil secara otomatis menghasilkan kode Java yang sesuai di belakang layar. Hal ini sangat membantu dalam mempercepat proses pengembangan, mengurangi kesalahan penulisan kode, dan memudahkan pemahaman struktur hubungan antar komponen.

Dalam praktikum ini, Window Builder dimanfaatkan untuk membangun aplikasi kalkulator aritmatika sederhana yang mampu melakukan operasi dasar seperti penjumlahan, pengurangan, perkalian, pembagian, dan modulo (+, -, *, /, %). Aplikasi ini dirancang sebagai simulasi dari sistem input-output sederhana yang umum ditemukan dalam aplikasi desktop.

1.2 Tujuan

Tujuan dari praktikum ini adalah:

1. Mengenal dan memahami komponen GUI dasar di Java Swing, seperti JLabel, JTextField, JButton, dan JComboBox.

2. Menguasai cara penempatan dan pengaturan tampilan komponen GUI menggunakan Window Builder atau secara manual di IDE.
3. Membuat aplikasi GUI sederhana yang mampu menerima input dari pengguna melalui JTextField dan JComboBox.
4. Melakukan validasi input pengguna, seperti memastikan bahwa JTextField bilangan 1 dan bilangan 2 tidak kosong, serta menampilkan pesan peringatan jika input kosong.
5. Mengimplementasikan logika dasar untuk memproses input berdasarkan pilihan operator dari JComboBox.

1.3 Manfaat

Manfaat dari praktikum ini adalah :

1. Meningkatkan kemampuan dalam memilih dan mengatur komponen GUI dasar seperti JLabel, JTextField, JButton, dan JComboBox secara visual maupun melalui kode.
2. Memahami pentingnya validasi input pengguna dalam aplikasi desktop.
3. Melatih logika pemrograman dasar dalam mengecek kondisi (if-else) untuk menentukan apakah data input telah diisi.
4. Menjadi dasar untuk pengembangan aplikasi GUI yang lebih kompleks untuk kedepannya, seperti sistem login, form pendaftaran, atau aplikasi perhitungan lainnya yang memerlukan validasi input serupa.

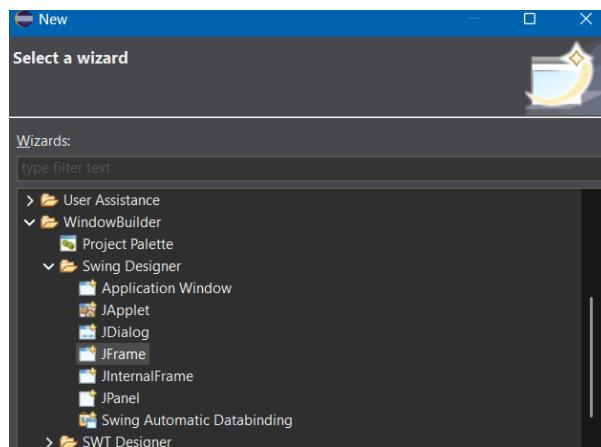
BAB II

PEMBAHASAN

2.1 Penjelasan Langkah-Langkah

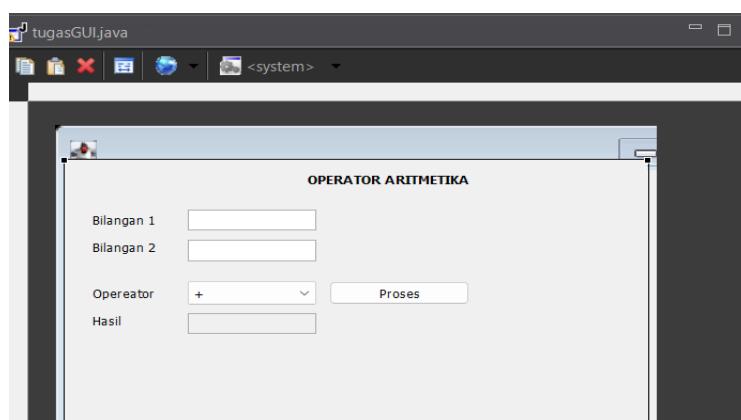
Pembuatan program GUI operator aritmatika sederhana ini melibatkan beberapa tahapan penting, mulai dari pembuatan kelas JFrame, mendesain bentuk aplikasi, serta mengkoding programnya. Berikut langkah-langkahnya :

1. Buat kelas JFrame pada windowbuilder, lalu beri nama program tersebut.



Gambar 2.1

2. Setelah itu bikin desain aplikasi menggunakan beberapa tools seperti JLabel, JTextField, JButton, dan JComboBox seperti contoh di bawah ini.



Gambar 2.2

3. Bikin logika di dalam ActionListener seperti kode dibawah ini.

```
btnNewButton.addActionListener(new ActionListener() {
    int Hasil;
    public void actionPerformed(ActionEvent e) {
        if (textbil1.getText().trim().isEmpty()) {
            pesanperingatan("silahkan masukkan bilangan 1");
        }
        else if(textbil2.getText().trim().isEmpty()) {
            pesanperingatan("silahkan masukkan bilangan 2");
        }
        else {
            try {
                int a =Integer.parseInt(textbil1.getText());
                int b =Integer.parseInt(textbil2.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;
                }
            }
            catch (NumberFormatException ex) {
                pesaneror("bilangan 1 dan bilangan 2 harus angka");
            }
        }
        txthasil.setText(String.valueOf(Hasil));
    }
})
```

Kode Program 2.1

4. Bikin method untuk WARNING_MESSAGE dan EROR_MESSAGE.

```
private void pesanperingatan(String pesan) {
    JOptionPane.showConfirmDialog(this, pesan, "peringatan", JOptionPane.WARNING_MESSAGE);
}
private void pesaneror(String pesan) {
    JOptionPane.showConfirmDialog(this, pesan, "kesalahan", JOptionPane.ERROR_MESSAGE);
}
```

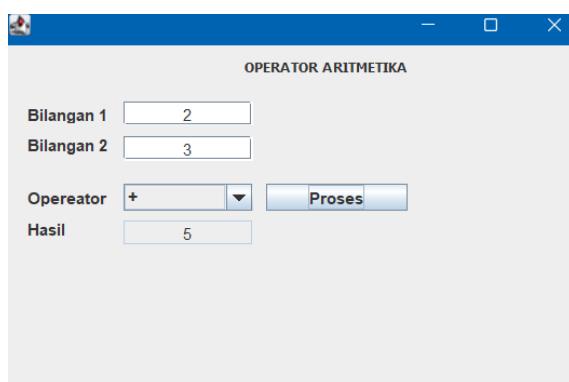
Kode Program 2.2

2.2 Analisis Hasil

Aplikasi yang dihasilkan dapat berfungsi sesuai dengan tujuan awal. Berikut adalah analisis dari berbagai skenario:

1. Jika Input Valid:

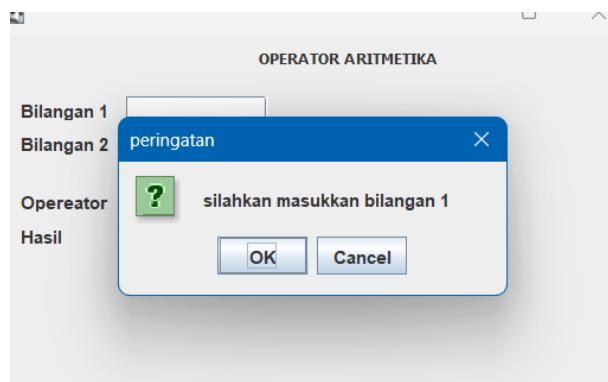
Jika pengguna memasukkan dua bilangan bulat dan memilih operator, maka tombol "Proses" akan menghitung hasilnya dan menampilkannya di txthasil.



Output 2.1

2. Jika Input Kosong:

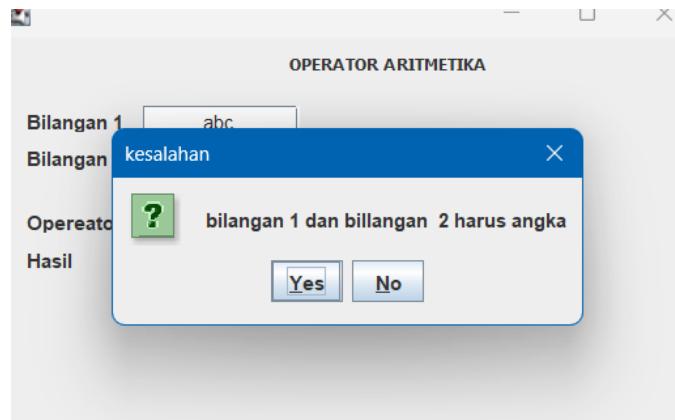
- Jika textbil1 kosong dan tombol "Proses" diklik, aplikasi akan menampilkan pesan peringatan "silahkan masukkan bilangan 1".
- Jika textbil1 terisi, tetapi textbil2 kosong, aplikasi akan menampilkan pesan "silahkan masukkan bilangan 2".



Output 2.2

3. Jika Input Bukan angka:

Jika pengguna memasukkan huruf atau simbol di salah satu atau kedua JTextField, aplikasi akan menangkap NumberFormatException dan menampilkan pesan error "bilangan 1 dan bilangan 2 harus angka".



Output 2.3

BAB III

KSIMPULAN

3.1 Kesimpulan

Dari praktikum ini, dapat disimpulkan bahwa pembuatan aplikasi GUI sederhana dengan Java Swing menggunakan komponen seperti JLabel, JTextField, JComboBox, dan JButton dapat dilakukan dengan cukup mudah. Aplikasi yang dibuat berhasil menampilkan interface yang sederhana, di mana pengguna dapat memasukkan dua bilangan, memilih operator aritmatika, dan menampilkan hasil perhitungan setelah menekan tombol Proses. Validasi input juga berhasil diterapkan, sehingga jika ada bilangan yang kosong, maka aplikasi akan memberikan peringatan kepada pengguna untuk mengisi data terlebih dahulu. Selain itu, ketika pengguna memasukkan huruf atau simbol bukan angka, aplikasi akan menampilkan pesan kesalahan pada inputan.

3.2 Saran

Aplikasi ini belum sepenuhnya sempurna karena belum menangani kasus pembagian atau modulo dengan angka nol, yang akan menyebabkan program berhenti bekerja. Untuk perbaikan kedepannya, sebaiknya tambahkan penanganan khusus untuk kasus tersebut agar aplikasi lebih stabil. Selain itu, penggunaan layout seperti BorderLayout atau GridLayout lebih disarankan daripada layout null agar tampilan tidak berantakan saat jendela diubah ukurannya. Penggunaan JOptionPane.showMessageDialog juga lebih tepat daripada showConfirmDialog untuk pesan peringatan dan error, karena tidak memerlukan konfirmasi dari pengguna. Secara keseluruhan, praktikum ini memberikan pemahaman dasar yang sangat berharga tentang bagaimana membuat aplikasi desktop interaktif di Java, serta pentingnya memperhatikan pengalaman pengguna dan keamanan program.

DAFTAR PUSTAKA

- [1] Oracle Corporation, “Java Platform, Standard Edition Documentation,” 2023. [Online]. Available: <https://docs.oracle.com/en/java/javase/index.html>
- [2] Oracle Corporation, “How to Use Combo Boxes,” The Java Tutorials, 2023. [Online]. Available: <https://docs.oracle.com/javase/tutorial/uiswing/components/combobox.html>
- [3] Eclipse Foundation, "WindowBuilder," *Eclipse Foundation*. [Online]. Available: <https://eclipse.dev/windowbuilder/>
- [4] D. Liang, *Introduction to Java Programming and Data Structures*, 12th ed. Pearson Education, 2019.
- [5] B. Eckel, *Thinking in Java*, 4th ed. Prentice Hall, 2006.
- [6] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1994.