

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN
GRAPHICAL USER INTERFACE (GUI)

Disusun Oleh:

Rahmat Ananda Nazar

2511532008

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

Asisten Praktikum : Muhammad Zaki Al Hafiz



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
TAHUN 2025

KATA PENGANTAR

Puji syukur atas kehadiran Tuhan Yang Maha Esa karena berkat karunia-nya penulis dapat menyelesaikan laporan praktikum dari mata kuliah algoritma dan pemrograman dengan judul “Graphical User Interface (GUI)” ini dengan baik. Laporan ini disusun untuk memenuhi kegiatan praktikum pada pekan 8 yang membahas tentang (GUI) menggunakan *tools* “Window Builder”.

Penulis menyadari bahwa laporan praktikum ini tidak terlepas dari kesalahan dan kekurangan baik dari segi penulisannya maupun pembahasannya. Oleh karena itu penulis ingin mengucapkan terimakasih kepada bapak dosen pengampu, Bapak Dr. Wahyudi S.T M.T yang telah membimbing dan memberi arahan selama praktikum. Penulis berharap laporan ini bisa bermanfaat bagi pembaca dalam hal memahami GUI ini.

DAFTAR ISI

T

KATA PENGANTAR.....	i
DAFTAR ISI	ii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan Praktikum.....	1
1.3 Manfaat Praktikum.....	1
BAB II PEMBAHASAN	2
2.1 Pengoperasian GUI	2
2.2 Program GUI.....	2
2.2.1 Struktur Design	2
2.2.2 Pesan Peringatan	3
2.2.3 Operator Aritmatika	4
2.3 Output Program.....	5
BAB III PENUTUP	6
3.1 Kesimpulan	6
DAFTAR PUSTAKA	7

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam era modern sekarang pengaruh dunia digital berkat perkembangan aplikasi dan website yang menggunakan antarmuka dan sistem yang inovatif, Hal ini membuat adopsi besar terhadap dunia digital. Salah satu dasar dari pengembangan antarmuka dan sistem yang inovatif itu adalah GUI (*Graphical User Interface*).

GUI merupakan antarmuka pengguna grafis yang memungkinkan pengguna berinteraksi dengan komputer melalui elemen visual seperti ikon, tombol, dan menu. Hal ini memudahkan pengguna dalam menggunakan sebuah sistem karena adanya tampilan visual. GUI ini penting dipelajari dan dipahami agar mahasiswa bisa membuat tampilan dan sistem yang menarik serta mudah digunakan untuk pengguna.

1.2 Tujuan Praktikum

Tujuan utama dari praktikum ini adalah memberikan pemahaman kepada mahasiswa tentang penggunaan GUI yang dispesifikan pada bahasa java, mulai dari awal cara mengunduh *tools* untuk menjalankan GUI nya dan bagaimana cara penggunaan dari *tools* yang akan membuat GUI dalam program java ini.

1.3 Manfaat Praktikum

Praktikum ini mahasiswa diharapkan dapat memahami bagaimana proses pembuatan GUI sebuah sistem yang kemudian bisa bermanfaat dan akan digunakan untuk membuat berbagai sistem yang inovatif dan menarik

BAB II PEMBAHASAN

2.1 Pengoperasian GUI

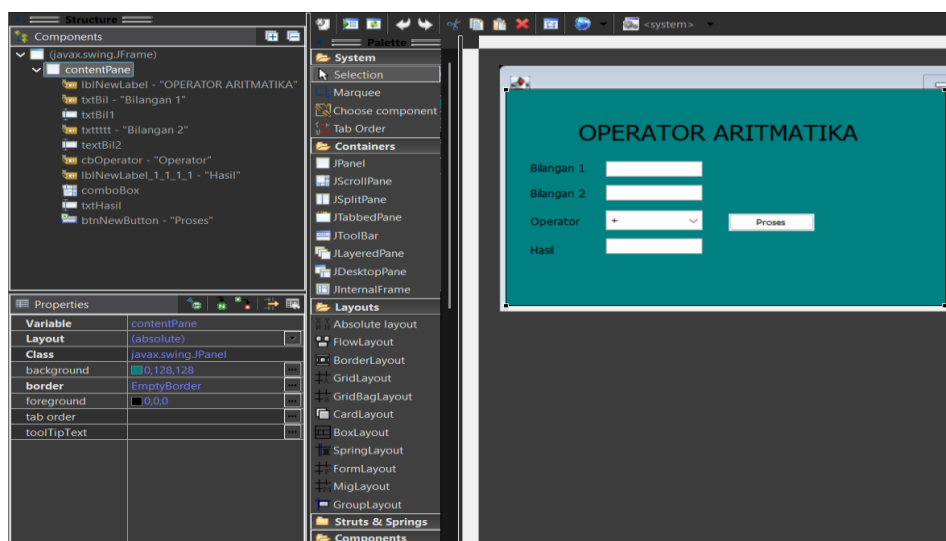
Untuk membuat sebuah GUI dari program java, perlu diunduh terlebih dahulu *tools* bernama WindowBuilder. Setelah itu kita buat package baru yang didalamnya terdapat sebuah wizard JFrame untuk mengoperasikan GUI ini.

2.2 Program GUI

Pada praktikum ini kita membuat sebuah program perhitungan aritmatika sederhana menggunakan logika if else dalam java, dan dibuat bentuk GUI nya. Program aplikasi operator aritmatika ini dikembangkan menggunakan bahasa pemrograman Java dengan memanfaatkan pustaka Java Swing untuk antarmuka grafis (GUI) dan Java AWT untuk manajemen *event*.

2.2.1 Struktur Design

Struktur utama program ini terdiri dari satu kelas turunan JFrame yang menampung seluruh komponen visual seperti JLabel, JTextField, JComboBox, dan JButton. Dalam inisialisasi program, tata letak komponen diatur menggunakan *Absolute Layout*, yang memungkinkan penempatan elemen secara presisi. Hal ini mempermudah penyusunan formulir input untuk bilangan pertama, bilangan kedua, pemilihan operator, serta kolom hasil.



Gambar 2.1

Dalam Design kita berikan JLabel nya untuk menampilkan text pada program yang diberi nama txtbil dan txttttt. 2 Buah box input JTextField untuk menerima input dari user dengan nama txtBil1, textBil2 dan txtHasil untuk menampilkan output. ComboBox yang berisi operator aritmatika yang akan dioperasikan. Serta JButton untuk mengeksekusi kode.

2.2.2 Pesan Peringatan

Salah satu aspek penting dalam program ini adalah penanganan interaksi pengguna melalui method pendukung untuk menampilkan pesan dialog. Terdapat dua *method private* yang dibuat secara manual, yaitu pesanPeringatan dan pesanError. Kedua method ini memanfaatkan kelas JOptionPane untuk memunculkan jendela *pop-up*. Method pesanPeringatan digunakan untuk memberikan notifikasi, sedangkan pesanError digunakan untuk menangani kesalahan fatal seperti kesalahan format input. Pembuatan method ini bertujuan agar kode dapat dipanggil berulang kali tanpa perlu menulis ulang kode JOptionPane yang panjang.

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

Gambar 2.2.



Gambar 2.3

Gambar 2.2 menunjukkan kode program yang digunakan untuk memberikan peringatan apabila user salah melakukan input (selain angka) atau tidak memasukan input salah satu angka, maka program tidak akan di eskeskusi dan akan mengeluarkan pesan error seperti yang ada pada gambar 2.3.

2.2.3 Operator Aritmatika

```
JButton btnNewButton = new JButton("Proses");
btnNewButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        int hasil = 0;
        if (txtBil1.getText().trim().isEmpty()) {
            pesanPeringatan ("Bilangan 1 Harus Di Isi");
        } else if (txtBil2.getText().trim().isEmpty()) {
            pesanPeringatan ("Bilangan 2 Harus diisi ");
        } else
        {
            try {
                int a = Integer.parseInt(txtBil1.getText());
                int b = Integer.parseInt(txtBil2.getText());
                int c = comboBox.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) {
                pesanError ("Bilangan 1 dan Bilangan 2 harus diisi angka");
            }
            txtHasil.setText(String.valueOf(hasil));
        }
    }
});
btnNewButton.setBounds(221, 151, 85, 21);
contentPane.add(btnNewButton);
```

Gambar 2.4

Setelah data berhasil, program menentukan jenis operasi matematika berdasarkan pilihan pengguna pada komponen comboBox. Logika pemilihan ini menggunakan struktur kontrol if yang mengecek indeks yang dipilih (getSelectedIndex). Indeks ke-0 merepresentasikan penjumlahan, indeks ke-1 pengurangan, dan seterusnya hingga operasi modulus. Hasil perhitungan disimpan dalam variabel integer, yang kemudian dikonversi kembali menjadi String agar dapat ditampilkan pada komponen txtHasil. Dengan alur ini, program menjamin bahwa output yang dihasilkan sesuai dengan operator yang dipilih dan data yang diinputkan valid.

2.3 Output Program

Setelah data berhasil dikonversi menjadi angka, program menentukan jenis operasi matematika berdasarkan pilihan pengguna pada komponen comboBox. Logika pemilihan ini menggunakan struktur kontrol if yang mengecek indeks yang dipilih (`getSelectedIndex`). Indeks ke 0 merepresentasikan penjumlahan, indeks ke 1 pengurangan, dan seterusnya hingga operasi modulus. Hasil perhitungan disimpan dalam variabel integer, yang kemudian ditampilkan pada komponen `txtHasil`.

The image displays five separate windows of a Java Swing application titled "OPERATOR ARITMATIKA". Each window contains the following elements:

- Bilangan 1:** A text input field.
- Bilangan 2:** A text input field.
- Operator:** A dropdown menu with a downward arrow.
- Proses:** A button to execute the calculation.
- Hasil:** A text input field for the result.

The data shown in the five windows is as follows:

Window	Bilangan 1	Bilangan 2	Operator	Hasil
Top Left	7	5	+	12
Top Right	56	36	%	20
Middle Left	7	8	-	-1
Middle Right	18	8	*	144
Bottom	81	9	/	9

Gambar 2.5

Pada Gambar 2.5 kita bisa melihat bagaimana program bekerja dengan meminta input 2 angka integer dari user di dalam fieldbox lalu pengguna memilih operasi aritmatika yang akan digunakan dan menekan tombol proses. Program langsung terkekekusi dan ouput hasil langsung ditampilkan

BAB III PENUTUP

3.1 Kesimpulan

Berdasarkan hasil praktikum pembuatan aplikasi GUI Operator Aritmatika ini, dapat disimpulkan bahwa penggunaan pustaka Java Swing dan AWT sangat memudahkan dalam pembangunan antarmuka pengguna yang. Penerapan blok try-catch terbukti krusial untuk menjaga stabilitas aplikasi. Hal ini menunjukkan sesuai yang ada pada latar belakang, tentang bagaimana antarmuka (*User Experience*) bisa mempengaruhi pengguna.

Selain itu, dari praktikum ini bisa dilihat bahwa pemahaman tentang GUI (Graphical User Interface) dalam Java juga sangat penting. Dari hasil yang ada, kita bisa mengubah kode logika menjadi tampilan visual yang interaktif. Dengan demikian, tujuan praktikum untuk memahami GUI yang ada dalam Java, untuk mengombinasikan *tools* yang menampilkan tampilan dan operator aritmatika untuk mengesekusi program telah tercapai.

DAFTAR PUSTAKA

W3Schools, “Java Tutorial,” 2025. [Daring]. Tersedia pada:
<https://www.w3schools.com/java/default.asp>.

Oracle, The Java® Tutorials — Primitive Data Types. [Online]. Available:
<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html>