

**LAPORAN PRAKTIKUM
AGORITMA PEMROGRAMAN
“PEMOGRAMAN GUI 2”**

DISUSUN OLEH:

NOFRI ILHAM

2511531013

DOSEN PENGAMPU:

Dr. WAHYUDI, S.T, M.T

ASISTEN PRAKTIKUM:

MUHAMMAD ZAKI AL HAFIZ



**DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS**

2025

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Puji syukur kehadiran Allah SWT atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Laporan Praktikum Pekan 9 dengan judul “Pemrograman GUI 2” tepat pada waktunya. Laporan ini disusun sebagai bentuk pertanggungjawaban dan dokumentasi proses praktikum pada mata kuliah Algoritma dan Pemrograman / Pemrograman Berbasis Objek.

Pada praktikum ini, penulis mempelajari implementasi lanjutan Graphical User Interface (GUI) menggunakan bahasa pemrograman Java dengan memanfaatkan library Swing. Materi yang dipraktikkan mencakup penggunaan komponen antarmuka seperti JButton, JTextField, dan JLabel, serta penerapan Event Handling melalui ActionListener untuk menangani interaksi pengguna.

Sebagai output dari praktikum, penulis berhasil membuat aplikasi Kalkulator Sederhana yang dapat melakukan operasi aritmatika dasar seperti penjumlahan, pengurangan, perkalian, dan pembagian. Hasil ini diharapkan menjadi dasar pemahaman dalam pengembangan aplikasi desktop yang lebih kompleks serta meningkatkan kemampuan integrasi logika program dengan antarmuka visual.

Penulis menyadari bahwa laporan ini masih memiliki kekurangan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat penulis harapkan demi penyempurnaan di masa mendatang. Penulis juga mengucapkan terima kasih kepada dosen pengampu dan asisten praktikum atas bimbingan dan dukungan yang diberikan selama proses pembelajaran berlangsung.

Padang, 28 November 2025

Nofri Ilham

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI	ii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan.....	1
1.3 Manfaat	2
BAB II PEMBAHASAN	3
2.1 Pengertian GUI.....	3
2.2 Desain GUI pada Aplikasi Kalkulator	3
2.3 Kode Program pada Aplikasi Kalkulator	5
2.4 Kode Program pada Aplikasi Kalkulator	6
2.5 Kode Program pada Aplikasi Kalkulator	8
2.6 Kode Program pada Aplikasi Kalkulator	9
2.7 Kode program pada Aplikasi Kalkulator	10
BAB III PENUTUP	12
3.1 Kesimpulan	12
3.2 Saran.....	12
DAFTAR PUSTAKA	14

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam dunia pemrograman, pembuatan aplikasi tidak hanya berfokus pada logika perhitungannya saja, tetapi juga bagaimana pengguna dapat berinteraksi dengan aplikasi tersebut. Karena itu, dibutuhkan tampilan yang mudah dipakai dan dipahami oleh pengguna. Salah satu cara membuat tampilan tersebut adalah dengan menggunakan GUI (Graphical User Interface), yaitu antarmuka visual yang berisi tombol, kotak teks, dan komponen lain yang bisa diklik atau diisi oleh pengguna.

Bahasa pemrograman Java menyediakan fitur untuk membuat GUI menggunakan library Swing. Dengan Swing, programmer dapat menambahkan komponen seperti JButton, JTextField, dan JLabel untuk membangun tampilan aplikasi. Selain itu, dibutuhkan juga Event Handling, seperti ActionListener, agar program bisa merespon aksi ketika pengguna menekan tombol.

Pada praktikum Pemrograman GUI 2, mahasiswa belajar bagaimana menggabungkan komponen GUI dengan logika program. Salah satu latihan yang dilakukan adalah membuat program Kalkulator Sederhana yang dapat melakukan operasi dasar seperti tambah, kurang, kali, dan bagi. Program ini dibuat menggunakan komponen Swing dan pengaturan event untuk membaca input dan menampilkan hasil.

Dengan adanya praktikum ini, mahasiswa diharapkan memahami cara membuat aplikasi sederhana berbasis GUI serta mampu mengembangkan aplikasi desktop yang interaktif. Pengetahuan dasar ini sangat penting sebelum masuk ke pembuatan aplikasi yang lebih besar dan kompleks.

1.2 Tujuan

1. Memahami konsep dasar GUI (Graphical User Interface) dalam pemrograman Java menggunakan library Swing.
2. Mempelajari penggunaan komponen GUI seperti JButton, JTextField, dan JLabel dalam membangun tampilan aplikasi.
3. Menerapkan Event Handling menggunakan ActionListener agar program dapat merespon aksi pengguna.

4. Mengimplementasikan pembuatan program Kalkulator Sederhana yang menjalankan operasi aritmatika dasar.
5. Melatih kemampuan menggabungkan logika program dengan antarmuka visual sehingga menghasilkan aplikasi interaktif.

1.3 Manfaat

1. Menambah pemahaman mahasiswa mengenai pembuatan aplikasi berbasis GUI pada bahasa pemrograman Java.
2. Melatih kemampuan menggunakan komponen Swing dalam membangun tampilan aplikasi.
3. Membantu mahasiswa memahami cara kerja Event Handling dan respon program terhadap aksi pengguna.
4. Meningkatkan keterampilan dalam menggabungkan logika perhitungan dengan antarmuka visual.
5. Menjadi dasar untuk mengembangkan aplikasi desktop yang lebih kompleks dan fungsional.

BAB II

PEMBAHASAN

2.1 Pengertian GUI

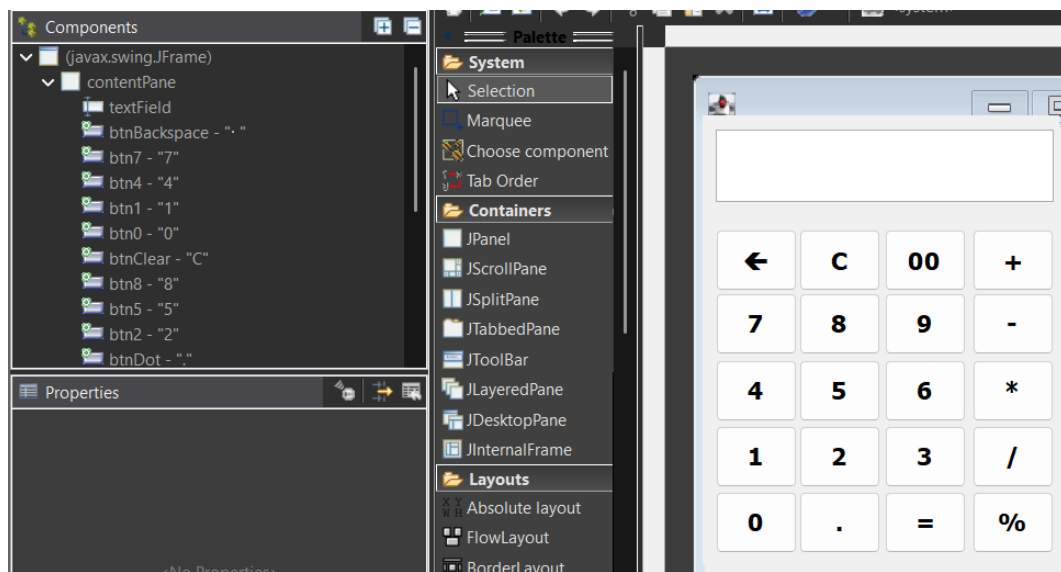
Graphical User Interface adalah sebuah sistem desain visual yang interaktif untuk sebuah software komputer. GUI sendiri menampilkan objek dari sebuah desain yang dapat menyampaikan informasi yang dibutuhkan dan mempresentasikan kerja dari pengguna. Melalui GUI, kita dapat mengetahui apakah input yang dilakukan pengguna telah diterima dan diberikan respon dalam bentuk visual.

Tampilan dari sebuah GUI dapat dilihat dari perubahan warna, visibilitas, ukuran, dan sejenisnya ketika terjadi sebuah interaksi dari pengguna terhadap aplikasi. Xerox Palo Alto adalah pengembang pertama dari GUI tepatnya pada tahun 1970. Selanjutnya, sistem GUI diterapkan ke dalam sistem Machintoshnya Apple dan juga sistem operasi Windows milik Microsoft.

Dengan ditemukannya GUI, dapat memberikan respon dari permasalahan yang dirasakan oleh pengguna. Jenis GUI yang bisa kalian lihat pada umumnya dapat dilihat pada interface sistem operasi yang biasa kalian gunakan adalah tombol, menu, scroll bar, kursor, dan masih banyak lagi jenisnya. Dan tak lupa juga GUI selalu berubah-ubah mengikuti perkembangan zaman.

2.2 Desain GUI pada Aplikasi Kalkulator

Berikut merupakan desain yang saya gunakan pada aplikasi Kalkulator sederhana Ketika praktikum pekan ke 9



Gambar 2.1

Pada pembuatan aplikasi kalkulator sederhana berbasis Java Swing, digunakan beberapa komponen GUI yang berfungsi untuk membangun tampilan dan memungkinkan terjadinya interaksi antara pengguna dan program. Setiap komponen memiliki peran penting dalam proses input, pemilihan operasi, dan penampilan hasil perhitungan. Dengan penggunaan komponen tersebut, aplikasi kalkulator dapat berjalan secara interaktif dan mudah digunakan oleh pengguna.

Adapun komponen GUI yang digunakan pada aplikasi ini adalah sebagai berikut:

1. JFrame, berfungsi sebagai jendela utama aplikasi yang menjadi tempat atau wadah bagi semua komponen GUI yang ditampilkan pada layar.
2. JTextField, digunakan sebagai area untuk menampilkan input angka dari pengguna sekaligus menampilkan hasil perhitungan. Komponen ini juga berfungsi sebagai display utama pada tampilan kalkulator.
3. JButton, digunakan sebagai tombol interaksi untuk mengeksekusi perintah tertentu, seperti memasukkan angka, memilih operator (+, -, x, ÷), dan menjalankan proses perhitungan melalui tombol sama dengan ("=").

2.3 Kode Program pada Aplikasi Kalkulator

```

101      JButton btn1 = new JButton("1");
102      btn1.addActionListener(new ActionListener() {
103          public void actionPerformed(ActionEvent e) {
104              String number=textField.getText()+btn1.getText();
105              textField.setText(number);
106          }
107      });
108      btn1.setFont(new Font("Tahoma", Font.BOLD, 18));
109      btn1.setBounds(10, 226, 59, 45);
110      contentPane.add(btn1);
111

```

Kode Program 2.1

Pada program kalkulator ini terdapat komponen JButton dengan nama btn1 yang berfungsi sebagai tombol input angka 1. Ketika tombol btn1 ditekan oleh pengguna, program akan menjalankan proses yang terdapat di dalam ActionListener melalui metode actionPerformed().

Di dalam event tersebut, program terlebih dahulu mengambil nilai teks yang sudah ada pada komponen textField menggunakan perintah getText(). Setelah itu, nilai teks tersebut digabungkan dengan teks dari tombol (btn1.getText()), sehingga angka 1 akan ditambahkan ke tampilan input. Hasil penggabungan tersebut kemudian dikembalikan dan ditampilkan kembali ke layar kalkulator melalui perintah textField.setText(number).

Dengan demikian, setiap kali tombol btn1 ditekan, angka 1 akan muncul pada layar input kalkulator sesuai urutan penekanannya oleh pengguna.

Selain pada komponen btn1, penggunaan struktur kode program yang sama juga diterapkan pada tombol angka lainnya, yaitu btn0, btn2, btn3, btn4, btn5, btn6, btn7, btn8, btn9, serta tombol btn00. Setiap tombol angka tersebut memiliki fungsi utama sebagai input nilai numerik ke dalam kalkulator.

Ketika salah satu tombol tersebut ditekan, sistem akan menjalankan mekanisme `ActionListener` yang bertugas mengeksekusi event `actionPerformed()`. Prosesnya sama seperti pada `btn1`, yaitu mengambil nilai yang sudah terdapat pada `textField`, menambahkan karakter angka sesuai tombol yang ditekan, kemudian menampilkan kembali hasilnya pada layar kalkulator melalui perintah `setText()`.

2.4 Kode Program pada Aplikasi Kalkulator

```

60      JButton btnBackspace = new JButton("\u2190");
61      btnBackspace.addActionListener(new ActionListener() {
62          public void actionPerformed(ActionEvent e) {
63              String backSpace=null;
64              if(textField.getText().length()>0)
65              {
66                  StringBuilder str= new StringBuilder(textField.getText());
67                  str.deleteCharAt(textField.getText().length()-1);
68                  backSpace=str.toString();
69                  textField.setText(backSpace);
70              }
71          }
72      });
73      btnBackspace.setFont(new Font("Wingdings", Font.BOLD, 18));
74      btnBackspace.setBounds(10, 83, 59, 45);
75      contentPane.add(btnBackspace);
76
77      JButton btnClear = new JButton("C");
78      btnClear.addActionListener(new ActionListener() {
79          public void actionPerformed(ActionEvent e) {
80              textField.setText(null);
81          }
82      });
83      btnClear.setFont(new Font("Tahoma", Font.BOLD, 18));
84      btnClear.setBounds(71, 83, 59, 45);
85      contentPane.add(btnClear);

```

Kode Program 2.2

Pada gambar diatas terdapat 2 kode program yaitu pada `btnBackSpace` dan `btnClear`

Pada `btnBackSpace` berfungsi untuk menghapus satu karakter terakhir pada input yang sedang tampil pada `textField`. Ketika tombol Backspace ditekan, event `actionPerformed()` akan dijalankan melalui `ActionListener`.

Di dalam event tersebut, program akan mengecek terlebih dahulu apakah panjang teks pada `textField` lebih dari nol

(textField.getText().length() > 0). Jika tersedia karakter, maka program akan membuat objek `StringBuilder` dari teks tersebut dan menghapus satu karakter terakhir menggunakan perintah `deleteCharAt()`. Setelah proses penghapusan selesai, hasilnya dikonversi kembali menjadi string dan ditampilkan pada layar menggunakan `setText()`. Dengan mekanisme ini, pengguna dapat menghapus input angka secara bertahap tanpa harus menghapus seluruh angka sekaligus.

Selain itu, terdapat komponen `JButton` bernama `btnClear` dengan fungsi untuk menghapus seluruh isi input pada `textField`. Ketika tombol `btnClear` ditekan, event `actionPerformed()` akan dijalankan dan program akan langsung menjalankan perintah `textField.setText(null)`. Perintah tersebut bertujuan mengosongkan seluruh teks yang sedang ditampilkan, sehingga area input menjadi kosong kembali. Tombol ini sangat berguna apabila pengguna ingin memulai perhitungan baru tanpa perlu menghapus angka satu per satu.

2.5 Kode Program pada Aplikasi Kalkulator

```

231         JButton btnEqual = new JButton("=");
232         btnEqual.addActionListener(new ActionListener() {
233             public void actionPerformed(ActionEvent e) {
234                 String answer;
235                 second=Double.parseDouble(textField.getText());
236                 if(operation=="+")
237                 {
238                     result=first+second;
239                     answer=String.format("%.2f", result);
240                     textField.setText(answer);
241                 }
242                 else if(operation=="-")
243                 {
244                     result=first-second;
245                     answer=String.format("%.2f", result);
246                     textField.setText(answer);
247                 }
248                 else if(operation=="*")
249                 {
250                     result=first*second;
251                     answer=String.format("%.2f", result);
252                     textField.setText(answer);
253                 }
254                 else if(operation=="/")
255                 {
256                     result=first/second;
257                     answer=String.format("%.2f", result);
258                     textField.setText(answer);
259                 }
260                 else if(operation=="%")
261                 {
262                     result=first%second;
263                     answer=String.format("%.2f", result);
264                     textField.setText(answer);
265                 }

```

Kode Program 2.3

Selanjutnya terdapat komponen JButton bernama btnEqual yang berfungsi untuk menampilkan hasil perhitungan dari operasi aritmatika yang dipilih pengguna. Ketika tombol = ditekan, event actionPerformed() akan dipanggil melalui ActionListener, dan program akan mulai memproses operasi matematika berdasarkan nilai variabel operation yang telah ditentukan sebelumnya.

Di dalam event tersebut, nilai angka kedua (second) diambil dari input yang berada pada textField dan dikonversi menjadi tipe data Double

menggunakan method `Double.parseDouble()`. Selanjutnya, program akan membandingkan nilai variabel `operation` untuk mengetahui jenis operasi yang sedang dijalankan, seperti penjumlahan (+), pengurangan (-), perkalian (*), pembagian (/), ataupun modulus (%).

Setelah jenis operasi ditentukan, program akan melakukan proses perhitungan menggunakan variabel `first` dan `second`, lalu menyimpan hasilnya ke variabel `result`. Hasil perhitungan tersebut kemudian diformat menggunakan `String.format("%.2f", result)` agar tampil dengan dua angka di belakang koma, dan ditampilkan kembali ke layar kalkulator melalui perintah `textField.setText(answer)`.

Dengan adanya tombol `btnEqual`, pengguna dapat melihat hasil akhir dari operasi yang telah dilakukan berdasarkan input angka dan operator yang dipilih.

2.6 Kode Program pada Aplikasi Kalkulator

```

        btn3.setFont(new Font("Tahoma", Font.BOLD, 18));
        btn3.setBounds(133, 226, 59, 45);
        contentPane.add(btn3);

        JButton btnDot = new JButton(".");
        btnDot.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String number=textField.getText()+btnDot.getText();
                textField.setText(number);
            }
        });
    });

```

Kode program 2.4

Selanjutnya terdapat komponen `JButton` bernama `btnDot` yang berfungsi sebagai tombol untuk memasukkan tanda titik desimal (.) pada nilai input. Tombol ini digunakan ketika pengguna ingin melakukan perhitungan angka pecahan atau bilangan desimal.

Ketika `btnDot` ditekan, event `ActionListener` akan dijalankan dan method `actionPerformed()` akan dipanggil. Di dalam event tersebut, program mengambil isi teks yang telah ada pada `textField`, kemudian

menambahkan karakter titik (.) menggunakan perintah `btnDot.getText()`. Setelah itu, hasil penggabungan ditampilkan kembali ke input kalkulator menggunakan perintah `textField.setText(number)`.

2.7 Kode program pada Aplikasi Kalkulator

```

280         JButton btnSub = new JButton("-");
281         btnSub.addActionListener(new ActionListener() {
282             public void actionPerformed(ActionEvent e) {
283                 first=Double.parseDouble(textField.getText());
284                 textField.setText("");
285                 operation="-";
286             }
287         });
288         btnSub.setFont(new Font("Tahoma", Font.BOLD, 18));
289         btnSub.setBounds(197, 129, 59, 45);
290         contentPane.add(btnSub);
291
292         JButton btnPlus = new JButton("+");
293         btnPlus.addActionListener(new ActionListener() {
294             public void actionPerformed(ActionEvent e) {
295                 first=Double.parseDouble(textField.getText());
296                 textField.setText("");
297                 operation="+";
298             }
299         });
300         btnPlus.setFont(new Font("Tahoma", Font.BOLD, 18));
301         btnPlus.setBounds(197, 83, 59, 45);
302         contentPane.add(btnPlus);
303
304         JButton btnMul = new JButton("*");
305         btnMul.addActionListener(new ActionListener() {
306             public void actionPerformed(ActionEvent e) {
307                 first=Double.parseDouble(textField.getText());
308                 textField.setText("");
309                 operation="*";
310             }
311         });
312         btnMul.setFont(new Font("Tahoma", Font.BOLD, 18));
313         btnMul.setBounds(197, 178, 59, 45);
314         contentPane.add(btnMul);

```

Kode program 2.5

Selanjutnya terdapat tombol operator `btnSub` yang digunakan untuk melakukan operasi penjumlahan. Ketika tombol `-` ditekan, event `ActionListener` akan dijalankan dan program akan mengambil nilai angka pertama yang telah diinput pada `textField` dan menyimpannya ke dalam variabel `first` menggunakan perintah

`Double.parseDouble(textField.getText())`. Setelah itu, `textField` dikosongkan (`setText("")`) agar pengguna dapat memasukkan angka selanjutnya. Selanjutnya operator penjumlahan disimpan ke dalam variabel `operation` dengan nilai "-", sehingga ketika tombol `=` ditekan, program mengetahui bahwa operasi yang harus dilakukan adalah penjumlahan.

Struktur kode dan cara kerja tombol operator lainnya seperti `btnPlus` (+), `btnMul` (*), `btnDivide` (/), dan `btnPercent` (%) mengikuti pola yang sama dengan tombol `btnPlus`. Perbedaannya hanya pada simbol operator yang disimpan pada variabel `operation` untuk menentukan jenis operasi aritmatika yang akan dijalankan.

BAB III

PENUTUP

3.1 Kesimpulan

Berdasarkan praktikum pekan ke-9 mengenai “Pemrograman GUI 2”, dapat disimpulkan bahwa mahasiswa telah mampu memahami serta mengimplementasikan konsep dasar pengembangan *Graphical User Interface* (GUI) pada bahasa pemrograman Java menggunakan library Swing. Melalui praktikum ini, mahasiswa berhasil memanfaatkan berbagai komponen GUI seperti JFrame, JButton, JTextField, dan JLabel untuk membangun antarmuka aplikasi. Selain itu, mahasiswa juga mampu menerapkan mekanisme Event Handling menggunakan ActionListener sehingga setiap aksi yang dilakukan pengguna dapat diproses dan ditampilkan dengan tepat.

Sebagai output praktikum, aplikasi Kalkulator Sederhana berhasil dibuat dan berfungsi dengan baik untuk menjalankan operasi aritmatika dasar seperti penjumlahan, pengurangan, perkalian, dan pembagian. Fitur tambahan seperti clear, backspace, dan input desimal juga telah diimplementasikan. Dengan tercapainya hasil tersebut, praktikum ini memberikan pemahaman dasar yang penting dalam pengembangan aplikasi desktop berbasis GUI dan menjadi pondasi untuk pembuatan aplikasi yang lebih kompleks.

3.2 Saran

1. Mengembangkan fitur lanjutan seperti operasi matematika tambahan agar aplikasi kalkulator lebih lengkap dan bervariasi.
2. Melakukan optimalisasi pada struktur kode dengan mengurangi pengulangan logika, terutama pada tombol angka.

3. Menambahkan mekanisme penanganan kesalahan (error handling) untuk mencegah terjadinya error akibat input tidak valid, termasuk pembagian dengan nol.
4. Meningkatkan tampilan dan kerapian antarmuka menggunakan layout manager yang lebih terstruktur.
5. Memberikan komentar dan dokumentasi tambahan pada kode agar lebih mudah dipahami dan dikembangkan pada pertemuan atau proyek selanjutnya.

DAFTAR PUSTAKA

- [1] Purwadhika, “Graphical User Interface Adalah: Fungsi dan Manfaatnya.” [Daring]. Tersedia pada: <https://purwadhika.com/blog/graphical-user-interface-adalah-fungsi-dan-manfaatnya>. [Diakses: 25-Nov-2025].