

LAPORAN PRATIKUM ALGORITMA PEMROGRAMAN

KALKULATOR GUI

DI SUSUN OLEH :

ABDUL KARIM ALGAZALI

NIM 2511532029

DOSEN PENGAMPU : Dr.WAHYUDI, S.T, M.T

ASISTEN LABORATORIUM: AUFAN TAUFIQURRAHMAN



DEPARTEMEN INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS ANDALAS

PADANG

2025

KATA PENGANTAR

Sebelumnya saya Panjatkan Puji syukur atas kehadiran Tuhan Yang Maha Esa, berkat rahmat dan izinnya juga laporan praktikum “KALKULATOR GUI” ini dapat diselesaikan dengan baik. Saya ucapkan terima kasih yang sebesar besarnya kepada Dr. Wahyudi, S.T, M.T selaku dosen pengampu yang telah membimbing Mata kuliah Algoritma dan Pemrograman. Dan tidak lupa saya ucapkan terima kasih kepada Uda Aufan Tafiqurrahman selaku asisten labor yang telah membimbing praktikum Kalkulator GUI

Laporan ini disusun sebagai Bentuk hasil kegiatan praktikum yang bertujuan untuk memahami konsep dasar GUI Pada java dalam pemrograman. Penulis menyadari bahwa pemahaman mendalam terhadap kedua topik ini merupakan fondasi essential bagi pengembangan keterampilan pemrograman yang lebih advanced.

Saya menyadari bahwa penulisan laporan praktikum ini masih jauh dari kata sempurna baik dari segi pembahasan dan penulisannya. namun dengan demikian saya telah berupaya dengan segala kemampuan dan pengetahuan yang dimiliki supaya laporan ini dapat selesai dengan baik dan oleh karenanya saya dengan rendah hati menerima saran, masukan dan kritikan guna penyempurnaan laporan ini.

Padang, 28 November 2025

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
BAB I PENDAHULUAN	3
1.1 Latar Belakang	3
1.2 Tujuan.....	3
1.3 Manfaat	3
BAB II PEMBAHASAN	4
2.1 Langkah pengerjaan	4
BAB III KESIMPULAN	7
3.1 Kesimpulan	7
3.2 Saran.....	7
DAFTAR PUSTAKA	8

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi perangkat lunak menuntut hadirnya aplikasi dengan antarmuka yang mudah digunakan (user-friendly). Bahasa pemrograman Java, dengan dukungan library Swing untuk pembuatan GUI, merupakan platform yang ideal untuk mempelajari pengembangan aplikasi semacam itu. Kalkulator digital dipilih sebagai proyek karena merupakan contoh yang fundamental dan universal.

Pembuatan kalkulator GUI ini tidak hanya bertujuan untuk menciptakan alat bantu hitung, tetapi juga sebagai media praktik untuk menerapkan konsep pemrograman berorientasi objek dan mempelajari integrasi antara logika program (back-end) dengan desain antarmuka (front-end) dalam satu aplikasi yang utuh.

1.2 Tujuan

1. Untuk mengimplementasikan konsep pemrograman berorientasi objek (OOP) dalam Java
2. Untuk mempelajari dan memahami penggunaan komponen-komponen GUI Swing dalam membangun antarmuka pengguna yang interaktif
3. Untuk mengembangkan logika pemrograman dalam menangani operasi aritmatika dasar dan event handling.
4. Untuk menghasilkan sebuah aplikasi kalkulator fungsional dengan antarmuka yang intuitif.

1.3 Manfaat

1. Memperoleh keterampilan praktis dalam pembangunan aplikasi desktop
2. Mempersiapkan diri untuk mengembangkan aplikasi dengan kebutuhan antarmuka yang lebih kompleks

BAB II

PEMBAHASAN

2.1 Langkah pengerjaan

Struktur Program KalkulatorGUI

a) Deklarasi Class dan Variabel

- Class KalkulatorGUI_2511532029 dengan sistem GUI
- Variabel penyimpanan:
 - first, second, result (double) - untuk operasi matematika
 - operation (String) - menyimpan jenis operasi
 - answer (String) - menyimpan hasil format string

b) Main Method dan Inisialisasi

- main() method menggunakan EventQueue.invokeLater() untuk thread safety
- Membuat instance window dan menampilkan frame

Konfigurasi GUI dan Komponen

a) Frame Setup

- Window size: 299x396 pixels
- Default close operation: EXIT_ON_CLOSE
- Layout: Absolute positioning (setBounds)

b) Komponen Utama

- TextField: display kalkulator dengan font bold 18pt
- Button Layout:
 - Baris 1: Backspace(B), Clear(C), Double Zero(00), Plus(+)
 - Baris 2: Angka 7, 8, 9, Minus(-)
 - Baris 3: Angka 4, 5, 6, Multiply(*)
 - Baris 4: Angka 1, 2, 3, Divide(/)
 - Baris 5: Angka 0, Decimal(.), Equals(=), Modulus(%)

Fungsi dan Event Handling

a) Tombol Angka dan Input

- Tombol 0-9: Menambahkan digit ke textField
- Tombol 00: Menambahkan double zero
- Tombol .: Menambahkan titik desimal
- Mekanisme: Concatenation string ke content textField

b) Tombol Control

- Backspace (B): Menghapus karakter terakhir menggunakan `StringBuilder`
- Clear (C): Mengosongkan seluruh `textField`

c) Tombol Operator

- +, -, *, /, %:
 - Menyimpan bilangan pertama (first)
 - Mengosongkan `textField` untuk input kedua
 - Menyimpan jenis operasi di variabel `operation`

d) Tombol Calculation (=)

- Mengambil bilangan kedua (second)
- Melakukan operasi berdasarkan `operation`:
 - Penjumlahan (`first + second`)
 - Pengurangan (`first - second`)
 - Perkalian (`first * second`)
 - Pembagian (`first / second`)
 - Modulus (`first % second`)
- Format output: 2 decimal places (`String.format("%.2f", result)`)

Alur program

Input Angka → Simpan first → Pilih Operator →

Input Angka → Simpan second → Calculate →

Format Result → Display Output

Gambar 2.2

```

2840 public void actionPerformed(ActionEvent e) {
2841     firstDouble.parseDouble(textField.getText());
2842     textField.setText("");
2843     operation = "+";
2844 }
2845
2846 }
2847
2848 bttnid_23151017.setAction(new Action("Istana", font_BOLD, 20));
2849 bttnid_23151017.setBounds(198, 58, 82);
2850 frame.getContentPane().add(bttnid_23151017);
2851
2852 }
2853
2854 bttnid_23151017.setAction(new Action("*/"),
2855 bttnid_23151017.addAction(new ActionListener() {
2856     public void actionPerformed(ActionEvent e) {
2857         firstDouble.parseDouble(textField.getText());
2858         textField.setText("");
2859         operation = "/";
2860     }
2861 });
2862
2863 }
2864
2865 bttnid_23151017.setAction(new Action("Istana", font_BOLD, 20));
2866 bttnid_23151017.setBounds(198, 98, 82);
2867 frame.getContentPane().add(bttnid_23151017);
2868
2869 }
2870
2871 bttnid_23151017.setAction(new Action("*/"),
2872 bttnid_23151017.addAction(new ActionListener() {
2873     public void actionPerformed(ActionEvent e) {
2874         firstDouble.parseDouble(textField.getText());
2875         textField.setText("");
2876         operation = "*";
2877     }
2878 });
2879
2880 }
2881
2882 bttnid_23151017.setAction(new Action("Istana", font_BOLD, 20));
2883 bttnid_23151017.setBounds(198, 297, 82);
2884 frame.getContentPane().add(bttnid_23151017);
2885
2886 }
2887
2888 }
2889
2890 }

```

Gambar 2.3

BAB III KESIMPULAN

3.1 Kesimpulan

Berdasarkan serangkaian praktikum pemrograman Java yang telah dilaksanakan, dapat disimpulkan bahwa seluruh materi berhasil dipahami dan diimplementasikan dengan baik. Praktikum ini berhasil membangun fondasi yang kuat dalam pemrograman berorientasi objek, dimulai dari konsep dasar class dan object pada program Mahasiswa, hingga pengembangan aplikasi GUI yang interaktif. Implementasi encapsulation, method setter-getter, dan mekanisme input/output menunjukkan pemahaman yang komprehensif terhadap prinsip-prinsip fundamental Java.

Pada tingkat lanjut, praktikum berhasil mendemonstrasikan kemampuan dalam mengintegrasikan berbagai konsep pemrograman menjadi aplikasi yang fungsional. Pengembangan kalkulator GUI dengan event handling, validasi input, dan operasi aritmatika lengkap membuktikan kemampuan dalam menyelesaikan masalah nyata menggunakan Java Swing. Demikian pula, penguasaan string manipulation dan control flow statements telah diaplikasikan dalam berbagai skenario, menunjukkan kedalaman pemahaman terhadap bahasa pemrograman Java secara keseluruhan.

3.2 Saran

1. Akan lebih baik bila dosen menyelenggarakan sesi pra-praktikum di kelas agar mahasiswa dapat memperoleh pemahaman awal yang lebih memadai, sehingga dapat menghindari kepanikan atau kesalahan saat praktikum
2. Sebaiknya dosen membagikan materi praktikum terlebih dahulu melalui iLearn agar mahasiswa bisa mempersiapkan diri sebelum pelaksanaan praktikum

DAFTAR PUSTAKA

- [1] Budi Raharjo. 2012. *Pemrograman Java untuk Aplikasi Desktop dan Android*. Bandung: Informatika
- [2] Hendarin, D. 2022. *Pengembangan Aplikasi Desktop dengan Java*. Yogyakarta: Deepublish.