

**LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN  
PEMROGRAMAN GUI 1**



**Oleh:**

**Naufal Arkan Octyandi**

**2511533005**

**MATA KULIAH ALGORITMA PEMROGRAMAN**

**DOSEN PENGAMPU: DR. WAHYUDI, S.T, M.T**

**ASISTEN LABOR: AUFAN TAUFIQURRAHMAN**

**FAKULTAS TEKNOLOGI INFORMASI**

**DEPARTEMEN INFORMATIKA**

**UNIVERSITAS ANDALAS**

**PADANG, 23 SEPTEMBER 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, 2025**

**Tim Penyusun**

**Naufal Arkan Octyandi**

## DAFTAR PUSTAKA

<b>KATA PENGANTAR .....</b>	<b>i</b>
<b>DAFTAR PUSTAKA.....</b>	<b>ii</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang Praktikum.....	1
1.2 Tujuan.....	1
<b>BAB II PENULISAN LAPORAN PRAKTIKUM .....</b>	<b>2</b>
2.1 Pengenalan Windows Builder .....	2
2.2 Kode Program .....	2
<b>BAB III KESIMPULAN.....</b>	<b>5</b>
3.1 kesimpulan .....	5
<b>DAFTAR PUSTAKA.....</b>	<b>6</b>

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang Praktikum**

Dalam kehidupan sehari-hari kita seringkali menggunakan aplikasi aplikasi yang memiliki pengaruh besar terhadap hidup kita. Tidak jarang pula kita memperhatikan perubahan-perubahan tata letak tombol-tombol yang kita gunakan pada setiap pembaharuan (update) aplikasi. Tata letak yang kita perhatikan setiap hari tersebut dapat disebut dengan Graphical User Interface (GUI). GUI mendeskripsikan bagaimana proses tatap muka pengguna dengan suatu perangkat lunak, tentu saja dengan disajikan dengan tampilan yang semenarik mungkin. Pada Java, kita bisa mengembangkan GUI untuk membuat aplikasi kita menjadi lebih menarik, karena program kita memiliki output yang tidak hanya berbentuk tulisan melainkan terdapat visual bentuk yang interaktif. Untuk melakukan ini maka kita harus mengunduh sebuah GUI Designer untuk Java. Pada Eclipse terdapat sebuah plugin bernama WindowBuilder yang berguna untuk membuat GUI dengan menggunakan bahasa pemrograman Java.

#### **1.2 Tujuan**

1. Menjelaskan mengenai penggunaan plugin WindowBuilder pada perangkat IDE seperti Eclipse.
2. Menjelaskan cara mengembangkan sebuah aplikasi yang dapat memproses algoritma dasar dengan plugin WindowBuilder.
3. Mempelajari cara menambahkan aksi pada tombol menggunakan *ActionListener*.
4. Mengetahui hubungan antara tampilan GUI dengan logika program di dalam Java.
5. Mengembangkan kemampuan mahasiswa dalam pembuatan program desktop sederhana berbasis Java Swing.

## BAB II

### PENULISAN LAPORAN PRAKTIKUM

#### 2.1 Pengenalan Windows Builder

Windows Builder adalah plugin yang digunakan pada Eclipse atau IntelliJ IDEA untuk membuat GUI berbasis Swing dan SWT. Dengan Windows Builder, programmer dapat menyusun tampilan secara visual layaknya membuat layout di Android Studio.

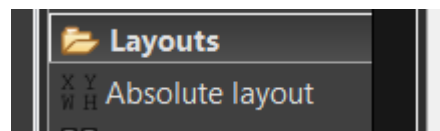
Tampilan dapat dibuat melalui dua mode:

1. Design View — tempat komponen di-drag ke dalam frame
2. Source View — tempat kode Java ditampilkan dan diubah secara manual

Keunggulan Windows Builder adalah hasil desain selalu disinkronkan dengan kode Java yang dihasilkan secara otomatis.

#### 2.2 Kode Program

Pada langkah pertama, penulis menerapkan Absolute Layout pada windows. Penggunaan absolute layout bertujuan agar seluruh komponen yang dipasang berada pada posisi dan ukuran yang tetap, walaupun jendela aplikasi digeser, diperbesar, atau diregangkan saat program dijalankan.



Setelah layout diterapkan, penulis mulai menambahkan beberapa komponen dasar, yaitu:

- JLabel

JLabel digunakan untuk menampilkan teks statis pada window.

Biasanya dipakai sebagai penanda atau keterangan untuk komponen-komponen lain, seperti label Nama, Alamat, Judul, dan sebagainya.

- JTextField

digunakan sebagai input teks untuk nama.

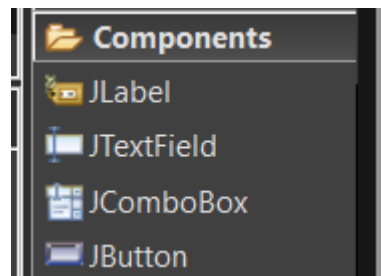
- JButton

JButton adalah tombol yang dapat ditekan oleh pengguna untuk menjalankan suatu aksi. Setiap tombol biasanya diberikan ActionListener, yang akan mengeksekusi kode tertentu saat tombol diclick.

- JComboBox

JComboBox adalah komponen dropdown/list pilihan yang digunakan ketika pengguna harus memilih salah satu dari beberapa opsi yang telah disediakan.

Komponen dapat ditambahkan dengan cara menarik (*drag*) dari panel Components ke area desain utama. Penulis juga melakukan pengaturan properti seperti ukuran font, alignment teks, dan posisi label agar tampilan terlihat rapi.



Langkah selanjutnya adalah memastikan program source code memiliki variabel yang sesuai dengan nama nama komponen yang ada pada sub-tab design.

```
JButton btnNewButton = new JButton("Hitung");
btnNewButton.addActionListener(new ActionListener() {
    int hasil;
    public void actionPerformed(ActionEvent e) {
        if (txtBil1.getText ().trim().isEmpty()) {
            pesanPeringatan("Bilangan 1 harus diisi");
        } else if (txtBil2.getText().trim ().isEmpty()) {
            pesanPeringatan("Bilangan 2 harus diisi");
        } else
        {
            try {
                int a=Integer.valueOf(txtBil1.getText ());
                int b=Integer.valueOf(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;}
                txtHasil.setText(String.valueOf(hasil));
            } catch(NumberFormatException ex) {
                PesanError("Bilangan 1 dan bilangan 2 harus angka");
            }
        }
    }
});
```

Kita juga memerlukan barisan program berikut untuk menjalankan fitur pesan error dan pesan peringatan mengenai input yang dibaca.

```
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);  
}
```

## **BAB III**

### **KESIMPULAN**

#### **3.1 kesimpulan**

Berdasarkan hasil praktikum pembuatan GUI menggunakan Windows Builder pada Java Swing, dapat disimpulkan bahwa penggunaan Windows Builder sangat membantu dalam proses pengembangan antarmuka aplikasi. Dibandingkan dengan pembuatan GUI secara manual melalui penulisan kode, Windows Builder memberikan efisiensi yang jauh lebih tinggi karena memungkinkan pengguna menempatkan komponen melalui metode *drag-and-drop*, serta secara otomatis menghasilkan kode Java yang terstruktur. Melalui praktikum ini, penulis memahami bahwa setiap komponen dalam Java Swing memiliki fungsi dan karakteristik tersendiri, mulai dari komponen input seperti JTextField, komponen interaksi seperti JButton, hingga komponen tampilan seperti JLabel. Windows Builder juga memudahkan pengaturan layout, terutama ketika menggunakan Absolute Layout, sehingga posisi setiap komponen dapat ditentukan secara presisi. Melalui tahapan-tahapan yang telah dilakukan, dapat disimpulkan bahwa penggunaan Windows Builder bukan hanya mempermudah pembuatan tampilan, tetapi juga meningkatkan pemahaman mahasiswa terhadap struktur GUI, interaksi komponen, serta proses penyusunan event dalam aplikasi Java. Dengan demikian, Windows Builder menjadi alat yang sangat efektif dalam pembelajaran dasar-dasar pengembangan aplikasi desktop.



## DAFTAR PUSTAKA

Coderanch. (2025). Absolute Layout GUI. coderanch.com .Diakses dari <https://coderanch.com/t/717654/java/Absolute-Layout>  
GUI#:~:text=The%20Sample%20Output%20uses%206,even%20on  
%20the%20same%20display.

Coursera. (2025). What Is a GUI. coursera.org. Diakses dari [https://www.coursera.org/articles/gui#:~:text=A%20graphical%20user%20interface%20\(GUI\)%20is%20an,cursors%2C%20and%20buttons%2C%20to%20make%20computing%20commands](https://www.coursera.org/articles/gui#:~:text=A%20graphical%20user%20interface%20(GUI)%20is%20an,cursors%2C%20and%20buttons%2C%20to%20make%20computing%20commands)

Eclipse Foundation. WindowBuilder | A powerful, easy-to-use, bi directional Java GUI designer. eclipse.dev. Diakses dari <https://eclipse.dev/windowbuilder/>

Oracle. (2024). *Cara Menggunakan ActionListener di Java*. oracle.com.  
Diakses dari <https://docs.oracle.com/javase/tutorial/uiswing/events/actionlistener.html>