

**LAPORAN RESMI
MODUL III
LAYOUT
PEMROGRAMAN VISUAL**



NAMA	: ANISYAFAAH
N.R.P	: 22041100105
DOSEN	: Ir. ACH. DAFID, S.T., M.T.
ASISTEN	: NURI HIDAYATULOH
TGL PRAKTIKUM	: 11 OKTOBER 2023

Disetujui : 17 Oktober 2023
Asisten

NURI HIDAYATULOH
21.04.411.00100



**LABORATORIUM BISNIS INTELIJEN SISTEM
PRODI SISTEM INFORMASI
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS TRUNOJOYO MADURA**

BAB I

PENDAHULUAN

1.1 Latar Belakang

Layout adalah komponen kunci dalam pengembangan aplikasi berbasis Java. Ketika Anda menciptakan antarmuka pengguna untuk program Java, pengaturan dan tata letak elemen-elemen grafisnya sangat penting. Latar belakang layout dalam Java seringkali berkaitan dengan masalah tata letak komponen seperti tombol, teks, panel, dan area gambar dalam suatu frame atau jendela. Penggunaan layout yang tepat dapat memengaruhi pengalaman pengguna dan fungsionalitas aplikasi.

Pentingnya layout dalam pengembangan Java terletak pada fleksibilitas dan responsifnya terhadap berbagai ukuran layar dan perangkat. Java menyediakan berbagai jenis layout manager, seperti BorderLayout, FlowLayout, GridLayout, dan lainnya, yang memungkinkan pengembang untuk dengan mudah mengatur elemen-elemen antarmuka pengguna dalam suatu frame. Ini memungkinkan aplikasi Java untuk lebih mudah beradaptasi dengan perubahan ukuran jendela atau layar pengguna.

Selain itu, layout yang baik dalam Java juga dapat meningkatkan estetika dan kejelasan antarmuka pengguna. Dengan mengatur elemen-elemen secara logis dan terstruktur, pengguna dapat dengan mudah memahami bagaimana berinteraksi dengan aplikasi. Layout juga dapat membantu menghindari tumpang tindih elemen-elemen dan meminimalkan potensi kesalahan dalam tampilan antarmuka.

1.2 Tujuan

- Mampu memahami konsep pemrograman Swing
- Mampu membuat halaman sederhana menggunakan komponen Swing dibantu tool GUI Builder
- Mampu membuat halaman sederhana menggunakan salah satu layout dari komponen swing (Border, Grid, GridBag) atau kombinasinya

BAB II

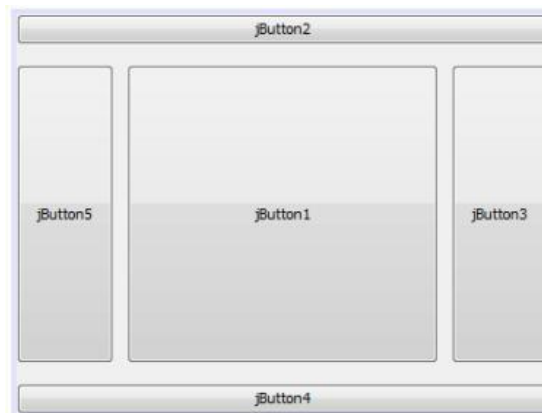
DASAR TEORI

2.1 Layout

Layout merupakan pengaturan peletakan komponen swing pada container-nya (container bisa panel atau frame atau apapun). Cara mengatur layout pada gui builder yang disediakan NetBeans adalah dengan klik kanan di bagian kosong pada container, lalu pilih “set Layout”. Ada 8 layout yang dapat dipilih ditambah 1 layout default dari gui builder. Tapi pada modul ini hanya dibahas 3 modul saja, Border Layout, Grid Layout dan GridBagLayout.

2.2.1. Border Layout

Layout merupakan pengaturan peletakan komponen swing pada container-nya (container bisa panel atau frame atau apapun). Cara mengatur layout pada gui builder yang disediakan NetBeans adalah dengan klik kanan di bagian kosong pada container, lalu pilih “set Layout”. Ada 8 layout yang dapat dipilih ditambah 1 layout default dari gui builder. Tapi pada modul ini hanya dibahas 3 modul saja, Border Layout, Grid Layout dan GridBagLayout.



2.2.2. Grid Layout

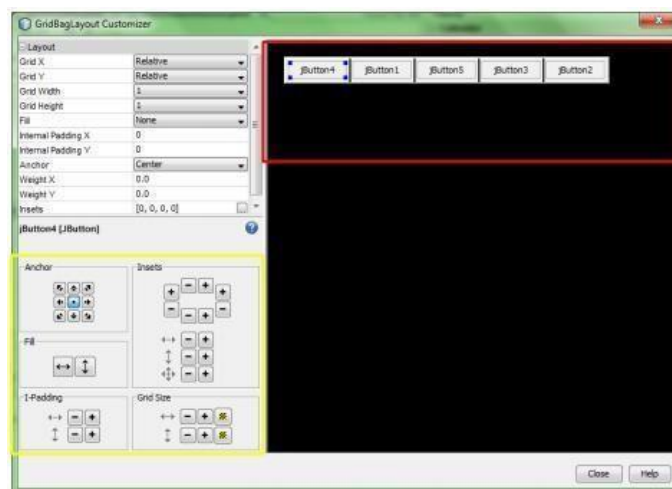
Grid layout akan menempatkan komponen ke dalam cell yang terkoordinat pada n kolom, dan m baris. 1 komponen menempati 1 cell dengan ukuran yang seragam. Pada properties dapat diatur jumlah baris dan kolom maksimal yang bisa ditempati oleh komponen. Selain itu dapat diatur jarak vertical dan horizontal antar

komponen. Pengaturan jumlah rows akan lebih dominan dibandingkan jumlah kolom. Dengan pengaturan seperti di atas, menggunakan 5 komponen button, maka letak komponen yang dihasilkan adalah sebagai berikut:



2.2.3. GridBag Layout

GridBag, sama seperti halnya Grid Layout, tapi dengan menggunakan GridBag Layout, 1 komponen bisa mengambil space penempatan lebih dari 1 cell. Tidak seragam seperti Grid Layout. Pengaturan peletakan komponen dapat dilakukan dengan klik kanan pada container yang dikenai layout, dan pilih “Customize Layout”. Dengan window yang muncul, letak komponen dapat diatur sesuai keinginan. Kotak merah memperlihatkan peletakan komponen di frame, dan kotak kuning memperlihatkan control komponen terhadap layout grid bag (padding, inset dll). Peletakan komponen dapat diubah menjadi n baris dan m kolom dengan menarik komponen yang ada.



MS
Nur H.

BAB III
TUGAS PENDAHULUAN

3.1 Soal

1. Jelaskan tentang apa itu layout dalam pemrograman Swing Java sedetail mungkin!
2. Apa perbedaan antara BorderLayout, FlowLayout, GridLayout, dan GridBagLayout dalam Java Swing?
3. Berikan contoh program mengenai layout! Gambarkan tampilan visualnya dan beri penjelasan gambar!

3.2 Jawab

1. Layout merupakan pengaturan penempatan komponen swing pada container-nya (bisa panel, frame, atau apapun). Layout sangat penting dalam pengembangan aplikasi GUI karena memungkinkan pengguna mengatur tampilan komponen dengan cara yang terorganisir dan sesuai dengan kebutuhan pengguna. Cara mengatur layout pada GUI builder yang disediakan NetBeans adalah dengan klik kanan dibagian kosong pada container, lalu pilih "set Layout". Terdapat beberapa jenis layout yang umum digunakan dalam Swing Java, antara lain:

- Flow Layout
- Box Layout
- Border Layout
- GridBag Layout
- Grid Layout
- Group Layout
- Card Layout
- Spring Layout.

2. Perbedaan antara Border Layout, Flow Layout, Grid Layout, dan GridBag Layout :

• Border Layout

- a. Membagi container menjadi lima wilayah
- b. Hanya satu komponen yang dapat ditempatkan setiap wilayah
- c. Komponen pada bagian tengah akan mengisi sisa ruang yang ada
- d. Cocok untuk tampilan yang memiliki satu komponen sentral utama.

Handwritten signature and name: NURAH

- Flow Layout
 - a. Flow Layout mengatur komponen dalam urutan horizontal dari kiri ke kanan, dan jika ruang habis, komponen baru akan pindah ke baris berikutnya.
 - b. Cocok untuk mengatur komponen dalam baris yang rapi, seperti menu bar atau toolbar.
- Grid Layout
 - a. Membagi container menjadi sel grid dengan jumlah baris dan kolom yang telah ditentukan.
 - b. Setiap sel dalam grid dapat menampung satu komponen.
 - c. Cocok untuk membuat tata letak yang terorganisir
- GridBag Layout
 - a. Dapat mengatur komponen dalam sel-sel grid dengan properti proporsi yang berbeda
 - b. Cocok untuk tampilan yang memerlukan tata letak yang lebih kompleks dan fleksibel.

3.

Luas Persegi Panjang	
Panjang :	<input style="width: 100%;" type="text"/>
Lebar :	<input style="width: 100%;" type="text"/>
Hasil :	<input style="width: 100%;" type="text"/>
HITUNG	

Gambar di atas adalah contoh pembuatan program menggunakan layout. Saya menggunakan border layout dan membaginya menjadi tiga bagian, yaitu bagian atas, tengah, dan selatan. Program di atas digunakan untuk menghitung luas persegi panjang. Pada bagian atas dan tengah saya berikan panel dan komponen-komponen lainnya seperti label dan textfield. Sedangkan pada bagian selatan saya berikan button Hitung yang didalamnya terdapat aksi menghitung luas persegi panjang ketika pengguna telah mengisi inputan panjang dan lebarnya.

BAB IV

IMPLEMENTASI

4.1 Source Code

1. Program Perpustakaan Sederhana

```
package modul3;

import javax.swing.JOptionPane;

public class Nomor1 extends javax.swing.JFrame {

    String judul, penulis;
    int tahun, jumlah;

    public Nomor1() {
        initComponents();

        JudulField.setText(judul);
        PenulisField.setText(penulis);
        JmlField.setText(Integer.toString(jumlah));
        TahunField.setText(Integer.toString(tahun));
    }

    @SuppressWarnings("unchecked")
    //Generated Code
    private void LihatButtonActionPerformed(java.awt.event.ActionEvent evt) {
        if (Radio1.isSelected()){
            judul = "PEMBELAJARAN SEJARAH";
            penulis = "abd. Rahman Hamid";
            tahun = 2014;
            jumlah = 198;
        }else if (Radio2.isSelected()){
            judul = "FLAMBOYAN SENJA";
            penulis = "Aliya Nurlela";
            tahun = 2012;
            jumlah = 139;
        }else if (Radio3.isSelected()){
            judul = "HUJAN";
            penulis = "Tere Liye";
            tahun = 2012;
            jumlah = 320;
        }else if (Radio4.isSelected()){
            judul = "SEJARAH YANG DISEMBUNYIKAN";
            penulis = "Jonathan Black";
            tahun = 2015;
            jumlah = 636;
        }else if (Radio5.isSelected()){
            judul = "LAUT BERCERITA";
            penulis = "Leila Salikha Chudori";
            tahun = 2017;
            jumlah = 394;
        }else if (Radio6.isSelected()){
            judul = "MARIPOSA";
            penulis = "Luluk HF";
            tahun = 2018;
            jumlah = 482;
        }else if (Radio7.isSelected()){
            judul = "KAMUS LENGKAP";
            penulis = "Niswatin Nurul";
            tahun = 2020;
```

```

        jumlah = 200;
    }else if (Radio8.isSelected()){
        judul = "PERAHU KERTAS";
        penulis = "Dee Lestari";
        tahun = 2008;
        jumlah = 444;
    }else if (Radio9.isSelected()){
        judul = "SISTEM INFORMASI MANAJEMEN";
        penulis = "Dr. Deni Darmawan";
        tahun = 2013;
        jumlah = 269;

    }else if (!Radio1.isSelected() && !Radio2.isSelected() && !Radio3.isSelected() &&
!Radio4.isSelected() &&
        !Radio5.isSelected() && !Radio6.isSelected()&& !Radio7.isSelected() &&
!Radio8.isSelected() &&
        !Radio9.isSelected()) {
        JOptionPane.showMessageDialog(this, "Silahkan Pilih Buku!");
        return;
    }

    JudulField.setText(judul);
    PenulisField.setText(penulis);
    String TahunStr = Integer.toString(tahun);
    TahunField.setText(TahunStr);
    String JumlahStr = Integer.toString(jumlah);
    JmlField.setText(JumlahStr + " Halaman");
}

```

2. Program Perhitungan Pajak

```

package modul3;

import javax.swing.*;

public class Nomor2 extends javax.swing.JFrame {

    public Nomor2() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    //Generated Code
    private void hitungButtonActionPerformed(java.awt.event.ActionEvent evt) {
        double gaji = Double.parseDouble(gajiField.getText());
        int kendaraan = Integer.parseInt(kendaraanField.getText());
        double pajak = 0;

        if (gaji >= 10000000 && kendaraan == 3) {
            pajak = 0.15 * gaji;
        } else if (gaji >= 5000000 && kendaraan == 2) {
            pajak = 0.10 * gaji;
        } else if (gaji >= 5 && kendaraan == 1) {
            pajak = 0.07 * gaji;
        } else if (gaji < 5000000 && kendaraan == 1) {
            pajak = 0.05 * gaji;
        } else if (gaji < 5000000 && kendaraan == 0) {
            pajak = 0.025 * gaji;
        }

        String kendaraanString = Integer.toString (kendaraan);
        kendaraanLabel.setText(kendaraanString + "\n Kendaraan");
    }
}

```



```

String gajiString = String.format("%.2f", gaji);
gajiLabel.setText(gajiString);
String pajakString = Double.toString(pajak);
hasilLabel.setText(pajakString);
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    gajiField.setText("");
    kendaraanField.setText("");
    gajiLabel.setText("");
    kendaraanLabel.setText("");
    hasilLabel.setText("");
}

```

4.2 Hasil

1. Program Perpustakaan Sederhana





Program di atas adalah program untuk melihat daftar dan detail buku. Disini saya menggunakan radiobutton untuk bagian daftar buku sehingga nanti user dapat memilih salah satu buku dan melihat detail bukunya. Adapun detail buku yang terdiri dari judul, penulis, tahun terbit, dan jumlah halaman buku. Untuk design, saya menggunakan layout yaitu border layout yang saya bagi menjadi tiga bagian, yaitu bagian utara, tengah, dan timur. Ketiga bagian ini saya berikan panel dan komponen" lainnya seperti button, radiobutton, label, dan textfield. Saya atur sour code setiap komponen sesuai dengan sistem programnya seperti gambar di atas.

2. Program Perhitungan Pajak

Program Perhitungan Pajak

Gaji :

Kendaraan :

Gaji Kendaraan Pajak

 CLEAR

Program Perhitungan Pajak

Gaji :

Kendaraan :

Gaji Kendaraan Pajak

 CLEAR

Program di atas adalah program untuk menghitung pajak. Disini saya menggunakan layout yaitu border layout yang saya bagi menjadi 4 bagian bagian, antara lain utara, tengah, timur, dan selatan. Bagian dan bagian ini saya

berikan panel dan bagian yang lain saya menggunakan button untuk menghitung pajak dan mengkosongkan inputan. Panel - panel ini saya isi dengan beberapa komponen seperti label dan textfield. Saya atur source code button menghitung untuk menghitung pajak sesuai dengan sistem programnya dan button clear untuk mengkosongkan inputan.

BAB V

PENUTUP

5.1 Analisa

Penggunaan layout dalam pengembangan aplikasi Java adalah aspek yang sangat penting karena memengaruhi tampilan dan fungsionalitas antarmuka pengguna. Layout manager dalam Java, seperti BorderLayout, GridLayout, dan lainnya, memungkinkan pengembang untuk mengorganisasi komponen-komponen grafis dalam suatu frame atau panel. Dengan menggunakan layout manager yang sesuai, pengembang dapat menciptakan tata letak yang responsif dan fleksibel, memungkinkan antarmuka pengguna beradaptasi dengan baik terhadap berbagai ukuran layar dan perangkat.

Selain itu, penggunaan layout juga berdampak pada pengalaman pengguna. Sebuah tampilan yang rapi dan terstruktur dapat membuat pengguna lebih mudah memahami bagaimana berinteraksi dengan aplikasi. Pengaturan elemen-elemen dengan baik juga membantu dalam menghindari tumpang tindih komponen atau masalah penataan. Dengan kata lain, layout yang tepat dapat mengurangi potensi kesalahan dan meningkatkan kenyamanan pengguna. Oleh karena itu, penggunaan layout yang bijak dalam pengembangan Java adalah suatu keterampilan yang penting dan dapat berdampak besar pada kesuksesan dan penerimaan aplikasi yang Anda kembangkan.

5.2 Kesimpulan

1. Layout merupakan pengaturan peletakan komponen swing pada container-nya(container bisa panel atau frame atau apapun).
2. Terdapat 8 layout yang dapat dipilih ditambah 1 layout default dari gui builder, namun pada modul ini hanya dibahas 3 modul saja, Border Layout, Grid Layout dan GridBagLayout.
3. Pada program yang telah diimplementasikan di atas menggunakan layout dari beberapa tipe layout yang telah disebutkan di atas.