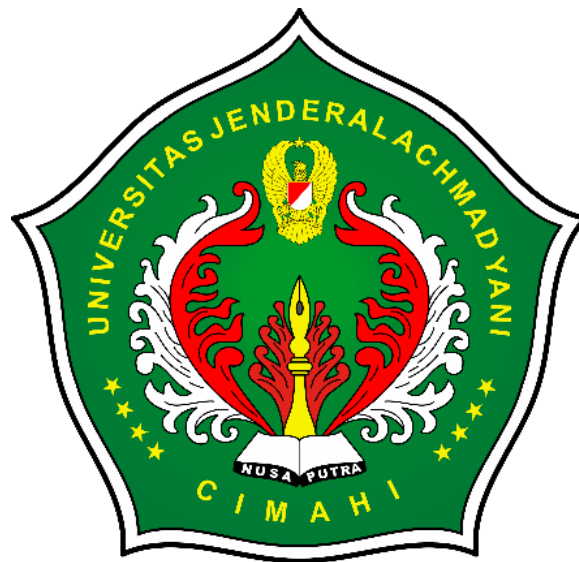


**LAPORAN PRAKTIKUM
PEMROGRAMAN OBJEK 2**

**MODUL 4
MEMBANGUN APLIKASI BERBASIS GUI**

**DISUSUN OLEH :
FATHIR AHMAD NURPADLI - 2250081132**



**PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN INFORMATIKA
UNIVERSITAS JENDERAL ACHMAD YANI
TAHUN 2024**

DAFTAR ISI

DAFTAR ISI.....	i
DAFTAR GAMBAR.....	iv
BAB I. HASIL PRAKTIKUM.....	1
I.1 Program Membuat Frame.....	1
I.1.A. Source Code	1
I.1.B. Screenshot.....	1
I.1.C. Analisa	2
I.2 Program Menambahkan Panel	2
I.2.A. Source Code	2
I.2.B. Screenshot.....	3
I.2.C. Analisa	4
I.3 Program Membuat Graphic Panel	4
I.3.A. Source Code	4
I.3.B. Screenshot.....	5
I.3.C. Analisa	5
I.4 Program Membuat FrameWControl.....	6
I.4.A. Source Code	6
I.4.B. Screenshot.....	7
I.4.C. Analisa	7
I.5 Membuat Flow Layout	8
I.5.A. Source Code	8
I.5.B. Screenshot.....	9
I.5.C. Analisa	9
I.6 Program BorderLayout.....	9
I.6.A. Source Code	9

I.6.B.	Screenshot.....	10
I.6.C.	Analisa	11
I.7	Membuat Tampilan Panel Kompleks.....	11
I.7.A.	Source Code	11
I.7.B.	Screenshot.....	12
I.7.C.	Analisa	13
I.8	Program Pacakage Swing.....	13
I.8.A.	Source Code	13
I.8.B.	Screenshot.....	14
I.8.C.	Analisa	14
I.9	Program JFrame	14
I.9.A.	Source Code	14
I.9.B.	Screenshot.....	16
I.9.C.	Analisa	16
BAB II.	TUGAS PRAKTIKUM	18
II.1	Tugas 4-1.....	18
II.2	Tugas 1-2.....	18
II.2.A.	Source Code	18
II.2.B.	Screenshot	20
II.2.C.	Analisa.....	20
II.3	Tugas 4-3.....	20
II.3.A.	Source Code	20
II.3.B.	Screenshot	20
II.3.C.	Analisa.....	20
II.4	Tugas 4-4.....	21
II.4.A.	Source Code	21
II.4.B.	Screenshot	22

II.5	Tugas 4-5.....	22
II.5.A.	Source Code	22
II.5.B.	Screenshot	23
BAB III.	KESIMPULAN	25

DAFTAR GAMBAR

Gambar 1 Hasil Output dari Program Membuat Frame.....	2
Gambar 2 Hasil Output dari Program Menambahkan Panel	4
Gambar 3 Hasil Output dari Program Membuat Graphic Panel	5
Gambar 4 Hasil Output dari Program FrameWControl	7
Gambar 5 Hasil Output dari Program Flow Layout.....	9
Gambar 6 Hasil Output dari Program BorderLayout.....	10
Gambar 7 Hasil Output dari Membuat Tampilan Panel Kompleks	13
Gambar 8 Hasil Output dari Program Package Swing.....	14
Gambar 9 Hasil Output dari Program JFrame	16
Gambar 10 Hasil Setelah diubah beberapa tulisan	20
Gambar 11 Hasil Membuat Layout sesuai dengan Tugas	20
Gambar 12 Hasil Membuat Tampilan Layout seperti pada Tugas	22
Gambar 13 Tampilan TicTacToe	24

BAB I. HASIL PRAKTIKUM

I.1 Program Membuat Frame

I.1.A. Source Code

```
import java.awt.*;

public class FrameExample {

    private Frame f;

    public FrameExample(){

        f = new Frame("Hello Out There!");

    }

    public void launchFrame(){

        f.setSize(170, 170);

        f.setBackground(Color.blue);

        f.setVisible(true);

    }

    public static void main(String args[]){

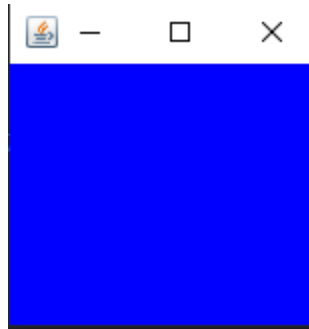
        FrameExample guiWindow = new FrameExample();

        guiWindow.launchFrame();

    }

}
```

I.1.B. Screenshot



Gambar 1 Hasil Output dari Program Membuat Frame

I.1.C. Analisa

Program Java di atas mendemonstrasikan cara membuat dan menampilkan sebuah window grafis sederhana menggunakan AWT (Abstract Window Toolkit). Kelas `FrameExample` memiliki sebuah atribut `Frame` yang diinisialisasi dalam konstruktor dengan judul "Hello Out There!". Metode `launchFrame` mengatur ukuran window menjadi 170x170 piksel, menetapkan latar belakang berwarna biru, dan menampilkan window tersebut dengan memanggil `setVisible(true)`. Dalam metode `main`, sebuah instance dari `FrameExample` dibuat dan metode `launchFrame` dipanggil untuk menampilkan window.

I.2 Program Menambahkan Panel

I.2.A. Source Code

```
import java.awt.*;

public class FrameWithPanel {
    private Frame f;

    public FrameWithPanel(String title){
        f = new Frame(title);
    }
}
```



```
public void launchFrame(){
    f.setSize(200,200);
    f.setBackground(Color.blue);
    f.setLayout(null); //override default layout mgr

    Panel pan = new Panel();
    pan.setSize(100, 100);
    pan.setBackground(Color.yellow);

    f.add(pan);
    f.setVisible(true);
}

public static void main(String args[]){
    FrameWithPanel guiWindow = new FrameWithPanel("Frame
With Panel");
    guiWindow.launchFrame();
}
}
```

I.2.B. Screenshot



Gambar 2 Hasil Output dari Program Menambahkan Panel

I.2.C. Analisa

Program Java di atas mendemonstrasikan cara membuat window grafis dengan panel menggunakan AWT (Abstract Window Toolkit). Kelas `FrameWithPanel` memiliki sebuah atribut `Frame` yang diinisialisasi dalam konstruktor dengan judul yang diberikan sebagai argumen. Metode `launchFrame` mengatur ukuran window menjadi 200x200 piksel, menetapkan latar belakang berwarna biru, dan mengatur layout menjadi null untuk menonaktifkan manajer tata letak default. Sebuah `Panel` dengan ukuran 100x100 piksel dan latar belakang berwarna kuning ditambahkan ke dalam window. Setelah itu, window ditampilkan dengan memanggil `setVisible(true)`. Dalam metode `main`, sebuah instance dari `FrameWithPanel` dibuat dengan judul "Frame With Panel" dan metode `launchFrame` dipanggil untuk menampilkan window beserta panel di dalamnya.

I.3 Program Membuat Graphic Panel

I.3.A. Source Code

```
import java.awt.*;

public class GraphicPanel extends Panel {
    public GraphicPanel(){
        setBackground(Color.black);
    }
}
```

```

public void paint(Graphics g){
    g.setColor(new Color(0, 255, 0)); //green
    g.setFont(new Font("Helvetica",Font.PLAIN,16));
    g.drawString("Hello GUI World", 30, 100);
    g.setColor(new Color(1.0f,0,0)); //red
    g.fillRect(30, 100, 150, 10);
}

public static void main(String args[]){
    Frame f = new Frame("Testing Graphics Panel");
    GraphicPanel gp = new GraphicPanel();
    f.add(gp);
    f.setSize(600, 300);
    f.setVisible(true);
}
}

```

I.3.B. Screenshot



Gambar 3 Hasil Output dari Program Membuat Graphic Panel

I.3.C. Analisa

Program Java di atas mendemonstrasikan cara menggambar grafik dasar di dalam sebuah panel menggunakan AWT (Abstract Window Toolkit). Kelas `GraphicPanel` yang merupakan subclass dari `Panel`, mengatur latar belakang menjadi hitam dalam konstruktor.

Metode `paint` digunakan untuk menggambar teks dan bentuk grafik. Pada metode ini, warna teks diatur menjadi hijau dan teks "Hello GUI World" digambar pada koordinat (30, 100) dengan font Helvetica ukuran 16. Setelah itu, warna diubah menjadi merah dan sebuah persegi panjang berukuran 150x10 piksel diisi di bawah teks. Metode `main` membuat sebuah `Frame` dengan judul "Testing Graphics Panel", menambahkan instance `GraphicPanel` ke dalamnya, mengatur ukuran frame menjadi 600x300 piksel, dan menampilkan frame tersebut.

I.4 Program Membuat FrameWControl

I.4.A. Source Code

```
import java.awt.*;

public class FrameWControls extends Frame {
    public FrameWControls(){

    }

    public static void main(String args[]){
        FrameWControls fwc = new FrameWControls();
        fwc.setLayout(new FlowLayout()); //more on this later
        fwc.setSize(600, 600);
        fwc.add(new Button("Test Me!"));
        fwc.add(new Label("Labe"));
        fwc.add(new TextField());
        CheckboxGroup cbg = new CheckboxGroup();
        fwc.add(new Checkbox("chk1", cbg, true));
        fwc.add(new Checkbox("chk2", cbg, false));
    }
}
```

```

        fwc.add(new Checkbox("chk3", cbg, false));

        List list = new List();
        list.add("MTV");
        list.add("V");
        fwc.add(list);

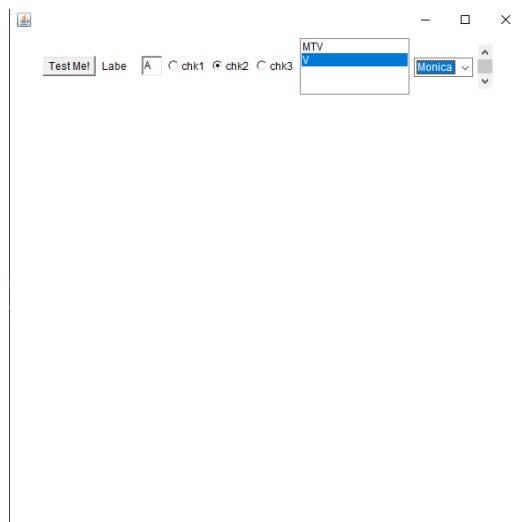
        Choice chooser = new Choice();
        chooser.add("Avril");
        chooser.add("Monica");
        chooser.add("Britney");
        fwc.add(chooser);

        fwc.add(new Scrollbar());

        fwc.setVisible(true);
    }
}

```

I.4.B. Screenshot



Gambar 4 Hasil Output dari Program FrameWControl

I.4.C. Analisa

Program Java di atas mendemonstrasikan cara membuat window grafis dengan berbagai kontrol menggunakan AWT (Abstract Window Toolkit). Kelas `FrameWControls` merupakan subclass dari `Frame`. Pada metode `main`, sebuah instance dari `FrameWControls` dibuat dan layout-nya diatur menggunakan `FlowLayout`, yang mengatur komponen secara berurutan sesuai urutan penambahannya. Ukuran frame diatur menjadi 600x600 piksel. Program ini kemudian menambahkan berbagai kontrol GUI ke frame, termasuk tombol (`Button`), label (`Label`), field teks (`TextField`), grup checkbox dengan tiga pilihan (`CheckboxGroup`), daftar pilihan (`List`), pilihan dropdown (`Choice`), dan scrollbar (`Scrollbar`). Setelah semua kontrol ditambahkan, frame ditampilkan dengan memanggil `setVisible(true)`.

I.5 Membuat Flow Layout

I.5.A. Source Code

```
import java.awt.*;

public class FlowLayoutDemo extends Frame {

    public FlowLayoutDemo(){

    }

    public static void main(String args[]){

        FlowLayoutDemo fld = new FlowLayoutDemo();

        fld.setLayout(new FlowLayout(FlowLayout.RIGHT, 10,
10));

        fld.add(new Button("ONE"));

        fld.add(new Button("TWO"));
```

```

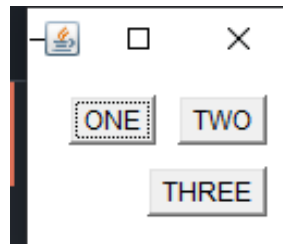
        fld.add(new Button("THREE"));

        fld.setSize(100, 100);

        fld.setVisible(true);
    }
}

```

I.5.B. Screenshot



Gambar 5 Hasil Output dari Program Flow Layout

I.5.C. Analisa

Program Java di atas mendemonstrasikan penggunaan layout manager `FlowLayout` dengan AWT (Abstract Window Toolkit) untuk mengatur tata letak komponen dalam sebuah frame. Kelas `FlowLayoutDemo` merupakan subclass dari `Frame`. Pada metode `main`, sebuah instance dari `FlowLayoutDemo` dibuat dan layout-nya diatur menggunakan `FlowLayout` dengan pengaturan alignment ke kanan (`FlowLayout.RIGHT`) dan jarak horizontal serta vertikal antara komponen sebesar 10 piksel. Tiga tombol (`Button`) dengan label "ONE", "TWO", dan "THREE" ditambahkan ke dalam frame. Ukuran frame diatur menjadi 100x100 piksel, dan kemudian frame ditampilkan dengan memanggil `setVisible(true)`.

I.6 Program BorderLayout

I.6.A. Source Code

```

import java.awt.*;

```

```

public class BorderLayoutDemo extends Frame {

    public BorderLayoutDemo(){

    }

    public static void main(String args[]){

        BorderLayoutDemo bld = new BorderLayoutDemo();

        bld.setLayout(new BorderLayout(10, 10)); //may remove

        bld.add(new Button("NORTH"), BorderLayout.NORTH);

        bld.add(new Button("SOUTH"), BorderLayout.SOUTH);

        bld.add(new Button("EAST"), BorderLayout.EAST);

        bld.add(new Button("WEST"), BorderLayout.WEST);

        bld.add(new Button("CENTER"), BorderLayout.CENTER);

        bld.setSize(200, 200);

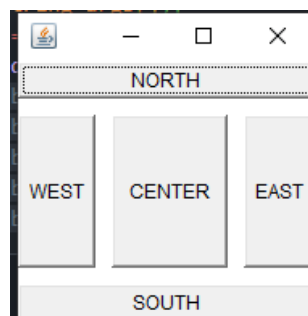
        bld.setVisible(true);

    }

}

```

I.6.B. Screenshot



Gambar 6 Hasil Output dari Program BorderLayout

I.6.C. Analisa

Program Java di atas mendemonstrasikan penggunaan layout manager `BorderLayout` dengan AWT (Abstract Window Toolkit) untuk mengatur tata letak komponen dalam sebuah frame. Kelas `BorderLayoutDemo` merupakan subclass dari `Frame`. Pada metode `main`, sebuah instance dari `BorderLayoutDemo` dibuat dan layout-nya diatur menggunakan `BorderLayout` dengan jarak horizontal dan vertikal antara komponen sebesar 10 piksel. Program ini menambahkan lima tombol (`Button`) ke dalam frame, masing-masing ditempatkan di lima area utama `BorderLayout`: "NORTH", "SOUTH", "EAST", "WEST", dan "CENTER". Ukuran frame diatur menjadi 200x200 piksel, dan frame ditampilkan dengan memanggil `setVisible(true)`.

I.7 Membuat Tampilan Panel Kompleks

I.7.A. Source Code

```
import java.awt.*;

public class ComplexLayout extends Frame {

    public ComplexLayout(){

    }

    public static void main(String args[]){

        ComplexLayout cl = new ComplexLayout();

        Panel panelNorth = new Panel();

        Panel panelCenter = new Panel();

        Panel panelSouth = new Panel();

        panelNorth.add(new Button("ONE"));
```

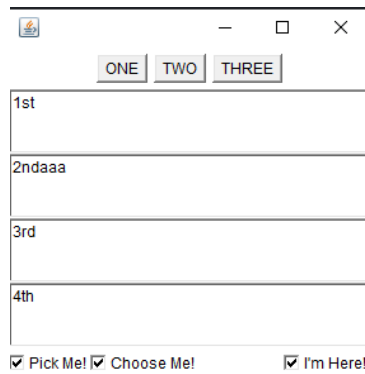
```

        panelNorth.add(new Button("TWO"));
        panelNorth.add(new Button("THREE"));
        panelCenter.setLayout(new GridLayout(4, 4));
        panelCenter.add(new TextField("1st"));
        panelCenter.add(new TextField("2nd"));
        panelCenter.add(new TextField("3rd"));
        panelCenter.add(new TextField("4th"));
        panelSouth.setLayout(new BorderLayout());
        panelSouth.add(new      Checkbox("Choose      Me!"),
BorderLayout.CENTER);
        panelSouth.add(new      Checkbox("I'm      Here!"),
BorderLayout.EAST);
        panelSouth.add(new      Checkbox("Pick      Me!"),
BorderLayout.WEST);

        cl.add(panelNorth, BorderLayout.NORTH);
        cl.add(panelCenter, BorderLayout.CENTER);
        cl.add(panelSouth, BorderLayout.SOUTH);
        cl.setSize(300,300);
        cl.setVisible(true);
    }
}

```

I.7.B. Screenshot



Gambar 7 Hasil Output dari Membuat Tampilan Panel Kompleks

I.7.C. Analisa

Program Java di atas mendemonstrasikan penggunaan berbagai layout manager dalam sebuah frame dengan AWT (Abstract Window Toolkit). Kelas `ComplexLayout` merupakan subclass dari `Frame`. Pada metode `main`, sebuah instance dari `ComplexLayout` dibuat. Tiga panel (`Panel`) dibuat dan diatur dalam tiga area berbeda dari frame menggunakan `BorderLayout`. `panelNorth` ditempatkan di area utara (`BorderLayout.NORTH`) dan berisi tiga tombol (`Button`) yang ditambahkan secara berurutan. `panelCenter` ditempatkan di area tengah (`BorderLayout.CENTER`) dan menggunakan `GridLayout` 4x4 untuk mengatur empat `TextField`. `panelSouth` ditempatkan di area selatan (`BorderLayout.SOUTH`) dan menggunakan `BorderLayout` untuk mengatur tiga `Checkbox` di bagian tengah, timur, dan barat. Ukuran frame diatur menjadi 300x300 piksel, dan frame ditampilkan dengan memanggil `setVisible(true)`.

I.8 Program Pacakage Swing

I.8.A. Source Code

```
import javax.swing.*;

public class MyContainer {

    public static void main(String args[]){

        JFrame.setDefaultLookAndFeelDecorated(true);
```

```
        JFrame container = new JFrame();

container.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        container.setSize(300, 100);

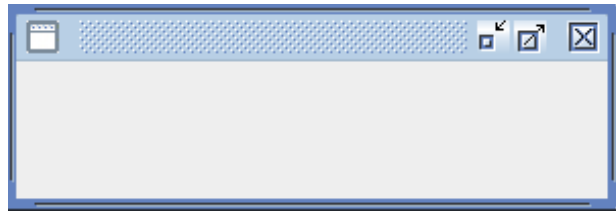
        container.setLocation(200, 200);

        container.setVisible(true);

    }

}
```

I.8.B. Screenshot



Gambar 8 Hasil Output dari Program Package Swing

I.8.C. Analisa

Program Java di atas mendemonstrasikan pembuatan jendela grafis menggunakan Swing, yang merupakan bagian dari Java Foundation Classes (JFC) untuk membangun antarmuka pengguna. Kelas `MyContainer` memiliki metode `main` yang mengatur tampilan default JFrame agar dihiasi dengan metode `setDefaultLookAndFeelDecorated(true)`. Sebuah `JFrame` baru bernama `container` dibuat, diatur agar operasi keluar program dilakukan saat window ditutup dengan `setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE)`. Ukuran frame diatur menjadi 300x100 piksel, dan lokasi awal window diatur pada koordinat (200, 200) di layar. Akhirnya, frame ditampilkan dengan memanggil `setVisible(true)`.

I.9 Program JFrame

I.9.A. Source Code

```
import java.awt.*;

import javax.swing.*;

public class SwingDemo {

    JFrame frame;

    JPanel panel;

    JTextField textField;

    JButton button;

    Container contentPane;

    public SwingDemo(){

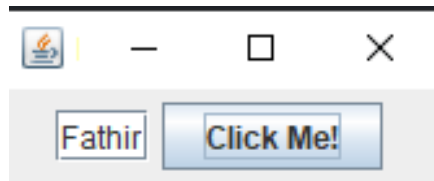
    }

    void launchFrame(){

        frame = new JFrame("My First Swing Application");
        panel = new JPanel();
        textField = new JTextField("Default Text");
        button = new JButton("Click Me!");
        contentPane = frame.getContentPane();
        panel.add(textField);
        panel.add(button);
        contentPane.add(panel, BorderLayout.CENTER);
        frame.pack();
        frame.setVisible(true);
    }
}
```

```
public static void main(String args[]){  
    SwingDemo sd = new SwingDemo();  
    sd.launchFrame();  
}  
}
```

I.9.B. Screenshot



Gambar 9 Hasil Output dari Program JFrame

I.9.C. Analisa

Program Java di atas mendemonstrasikan pembuatan dan pengaturan komponen GUI menggunakan Swing, yang merupakan bagian dari Java Foundation Classes (JFC). Kelas `SwingDemo` memiliki beberapa atribut: `JFrame`, `JPanel`, `JTextField`, `JButton`, dan `Container`. Konstruktor `SwingDemo` tidak melakukan inisialisasi apapun, namun metode `launchFrame` melakukan inisialisasi dan pengaturan komponen GUI. Metode ini membuat sebuah `JFrame` dengan judul "My First Swing Application", sebuah `JPanel`, sebuah `JTextField` dengan teks default "Default Text", dan sebuah `JButton` dengan label "Click Me!". `JPanel` ditambahkan ke `contentPane` dari `JFrame` di area tengah menggunakan `BorderLayout.CENTER`. Metode `frame.pack()` digunakan untuk mengatur ukuran frame berdasarkan komponen yang ada di dalamnya, dan `frame.setVisible(true)` digunakan untuk menampilkan frame. Metode `main` membuat sebuah instance dari `SwingDemo` dan memanggil `launchFrame` untuk menampilkan antarmuka pengguna.

BAB II. TUGAS PRAKTIKUM

II.1 Tugas 4-1

1. Program sudah di compile di BAB Hasil Praktikum Program Membuat Frame, dan Program Menambahkan Panel
2. Frame adalah jendela utama yang biasanya memiliki elemen-elemen seperti baris judul, tombol untuk menutup, memperkecil, dan memperbesar jendela, serta dapat berdiri sendiri. Panel, di sisi lain, adalah wadah sederhana yang digunakan untuk mengelompokkan komponen GUI lainnya seperti tombol, label, dan teks di dalam Frame atau wadah lainnya.
3. Program sudah di tambahkan beberapa objek
4. Program sudah di Analisa di BAB Hasil Praktikum Program Membuat Frame dan Program Menambahkan Panel

II.2 Tugas 1-2

II.2.A. Source Code

```
import java.awt.*;

public class GraphicPanel extends Panel {
    public GraphicPanel() {
        setBackground(Color.black);
    }

    public void paint(Graphics g) {
        // Teks hijau dengan font Helvetica
        g.setColor(new Color(0, 255, 0)); // green
        g.setFont(new Font("Helvetica", Font.PLAIN, 16));
        g.drawString("Hello GUI World", 30, 100);
    }
}
```



```

// Menggambar kotak merah
g.setColor(new Color(1.0f, 0, 0)); // red
g.fillRect(30, 110, 150, 10);

// Teks biru dengan font Arial Tebal
g.setColor(new Color(0, 0, 255)); // blue
g.setFont(new Font("Arial", Font.BOLD, 20));
g.drawString("Blue Bold Text", 30, 150);

// Teks oranye dengan font Courier Miring
g.setColor(new Color(255, 165, 0)); // orange
g.setFont(new Font("Courier", Font.ITALIC, 18));
g.drawString("Orange Italic Text", 30, 180);

// Teks ungu dengan font Times New Roman Tebal Miring
g.setColor(new Color(128, 0, 128)); // purple
g.setFont(new Font("Times New Roman", Font.BOLD |
Font.ITALIC, 22));
g.drawString("Purple Bold Italic Text", 30, 210);
}

public static void main(String args[]) {
    Frame f = new Frame("Testing Graphics Panel");
    GraphicPanel gp = new GraphicPanel();
    f.add(gp);
}

```

```

        f.setSize(600, 300);

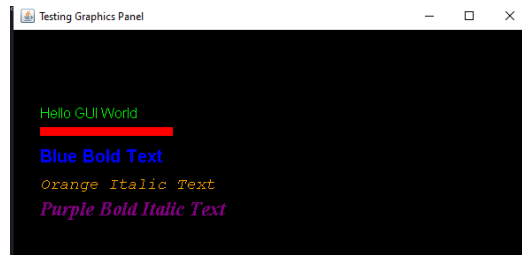
        f.setVisible(true);

    }

}

```

II.2.B. Screenshot



Gambar 10 Hasil Setelah diubah beberapa tulisan

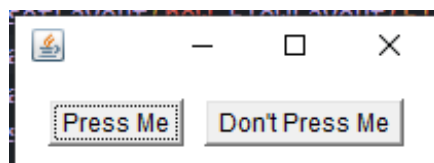
II.2.C. Analisa

Program Java di atas mendemonstrasikan penggunaan berbagai gaya teks dan warna menggunakan AWT (Abstract Window Toolkit) dalam sebuah panel grafis. Kelas `GraphicPanel` adalah subclass dari `Panel` yang mengatur latar belakang menjadi hitam dalam konstruktor. Dalam metode `paint`, beberapa teks dengan berbagai warna dan font digambar: teks hijau dengan font Helvetica, teks biru dengan font Arial tebal, teks oranye dengan font Courier miring, dan teks ungu dengan font Times New Roman tebal miring. Selain itu, program juga menggambar sebuah kotak merah di bawah teks "Hello GUI World". Dalam metode `main`, sebuah frame berjudul "Testing Graphics Panel" dibuat, panel `GraphicPanel` ditambahkan ke dalamnya, dan frame ditampilkan dengan ukuran 600x300 piksel.

II.3 Tugas 4-3

II.3.A. Source Code

II.3.B. Screenshot



Gambar 11 Hasil Membuat Layout sesuai dengan Tugas

II.3.C. Analisa

Layout manager di Java mengatur tata letak komponen dalam wadah seperti Frame atau Panel. FlowLayout adalah salah satu layout manager yang mengatur komponen secara berurutan dari kiri ke kanan, seperti kata-kata dalam sebuah paragraf. Jika tidak ada cukup ruang dalam satu baris, FlowLayout akan melanjutkan ke baris berikutnya. Anda dapat menentukan alignment (penyelarasan) dan jarak antara komponen (horizontal dan vertikal) saat membuat objek FlowLayout.

Program Java di atas mendemonstrasikan penggunaan FlowLayout untuk mengatur tata letak komponen dalam sebuah Frame. Kelas FlowLayoutDemo adalah subclass dari Frame. Pada metode main, sebuah instance dari FlowLayoutDemo dibuat dan layout-nya diatur menggunakan FlowLayout dengan penyelarasan tengah (FlowLayout.CENTER), serta jarak horizontal dan vertikal antara komponen masing-masing sebesar 10 piksel. Program menambahkan dua tombol (Button) ke dalam frame dengan label "Press Me" dan "Don't Press Me". Ukuran frame diatur menjadi 100x100 piksel dan frame ditampilkan dengan memanggil setVisible(true).

II.4 Tugas 4-4

II.4.A. Source Code

```
import java.awt.*;

public class ComplexLayout extends Frame {
    public ComplexLayout(){

    }

    public static void main(String args[]){
        ComplexLayout cl = new ComplexLayout();
        Panel panelNorth = new Panel();
        Panel panelCenter = new Panel();
        Panel panelSouth = new Panel();
```

```

        panelNorth.add(new Button("File"));

        panelNorth.add(new Button("Help"));

        panelSouth.setLayout(new BorderLayout());

        panelSouth.add(new Button("West"), BorderLayout.WEST);

        panelSouth.add(new Button("Work Space region"),
BorderLayout.CENTER);

        cl.add(panelNorth, BorderLayout.NORTH);

        cl.add(panelSouth, BorderLayout.SOUTH);

        cl.setSize(200,200);

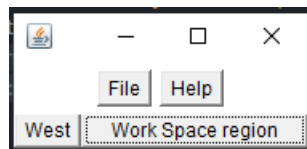
        cl.setVisible(true);

    }

}

```

II.4.B. Screenshot



Gambar 12 Hasil Membuat Tampilan Layout seperti pada Tugas

II.5 Tugas 4-5

II.5.A. Source Code

```

import java.awt.*;

import javax.swing.*;

public class TicTacToeGUI extends JFrame {

    private JButton[][] buttons = new JButton[3][3];

    public TicTacToeGUI() {

        setTitle("Tic Tac Toe");
    }
}

```

```

        setLayout(new GridLayout(3, 3));

        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                buttons[i][j] = new JButton("");
                buttons[i][j].setFont(new Font("Arial",
Font.PLAIN, 60));

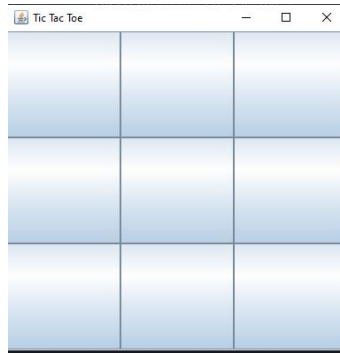
                buttons[i][j].setFocusPainted(false);
                add(buttons[i][j]);
            }
        }

        setSize(400, 400);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    public static void main(String[] args) {
        new TicTacToeGUI();
    }
}

```

II.5.B. Screenshot



Gambar 13 Tampilan TicTacToe

BAB III. KESIMPULAN

Membangun aplikasi berbasis GUI (Graphical User Interface) dalam Java melibatkan penggunaan komponen dan layout manager dari pustaka AWT (Abstract Window Toolkit) atau Swing untuk menciptakan antarmuka yang interaktif dan mudah digunakan. Proses ini mencakup inisialisasi komponen seperti tombol, label, panel, dan frame, serta pengaturan tata letak menggunakan layout manager seperti `FlowLayout`, `BorderLayout`, atau `GridLayout`. Selain itu, penanganan acara (event handling) merupakan bagian penting untuk menangani interaksi pengguna, seperti klik tombol atau input teks. Dengan menggabungkan elemen-elemen ini, pengembang dapat membangun aplikasi yang tidak hanya fungsional tetapi juga intuitif dan menarik secara visual, memenuhi kebutuhan pengguna yang beragam. Melalui pemahaman dan penerapan konsep-konsep ini, pengembang dapat membuat aplikasi GUI yang efektif dan efisien, seperti permainan sederhana, alat produktivitas, atau aplikasi bisnis.