

LAPORAN PRAKTIKUM (Java GUI)

Diajukan untuk memenuhi salah satu tugas praktikum Mata kuliah Pemrograman Berorientasi Objek



Disusun Oleh:
Nalendra Praja Bredtyopati Yudo (231511056)
Jurusan Teknik Komputer dan Informatika

Program Studi D-3 Teknik Informatika
Politeknik Negeri Bandung
2024

<https://github.com/Cobalttt2311/OOP-Praktik>

1. Source Code Program

FoodOrderApp.java	<pre>import javax.swing.*; import java.awt.*; import java.awt.event.ActionEvent; import java.awt.event.ActionListener; public class FoodOrderApp { public static void main(String[] args) { // Frame Utama JFrame frame = new JFrame("Aplikasi Pemesanan Makanan"); frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); frame.setSize(600, 500); frame.setLayout(new BorderLayout()); // Panel Data Customer dengan warna latar belakang JPanel customerPanel = new JPanel(new GridLayout(3, 2, 5, 5)); customerPanel.setBorder(BorderFactory.createTitledBorder(BorderFactory.createLineBorder(Color.BLACK), "Data Customer")); customerPanel.setBackground(new Color(240, 240, 240)); customerPanel.add(new JLabel("Nama: ")); JTextField nameField = new JTextField(10); customerPanel.add(nameField); customerPanel.add(new JLabel("Alamat: ")); JTextField addressField = new JTextField(10); customerPanel.add(addressField); customerPanel.add(new JLabel("No Telp: ")); JTextField phoneField = new JTextField(10); customerPanel.add(phoneField); // Panel Pilih Menu JPanel menuPanel = new JPanel(new GridLayout(3, 1, 5, 5)); menuPanel.setBorder(BorderFactory.createTitledBorder(BorderFactory.createLineBorder(Color.BLACK), "Pilih Menu")); menuPanel.setBackground(new Color(240, 240, 240)); JCheckBox steakCheckBox = new JCheckBox("Steak"); JCheckBox spaghettiCheckBox = new JCheckBox("Spageti"); JCheckBox pizzaCheckBox = new JCheckBox("Pizza"); menuPanel.add(steakCheckBox); menuPanel.add(spaghettiCheckBox); menuPanel.add(pizzaCheckBox); // Panel Total Bayar dengan warna teks JPanel totalPanel = new JPanel(new GridLayout(1, 2)); JLabel totalLabel = new JLabel("TOTAL BAYAR"); totalLabel.setFont(new Font("Arial", Font.BOLD, 14)); totalPanel.add(totalLabel); JTextField totalField = new JTextField("0", 10); totalField.setEditable(false); totalField.setBackground(Color.BLACK); totalField.setForeground(Color.YELLOW); totalField.setFont(new Font("Arial", Font.BOLD, 16)); totalField.setHorizontalAlignment(JTextField.CENTER); totalPanel.add(totalField); // Panel Pesanan JTextArea orderArea = new JTextArea(10, 30); orderArea.setEditable(false); orderArea.setBorder(BorderFactory.createLineBorder(Color.GRAY)); orderArea.setFont(new Font("Consolas", Font.PLAIN, 12)); JScrollPane scrollPane = new JScrollPane(orderArea); // Tombol Tambah dengan warna tombol</pre>
-------------------	---

```

        JButton addButton = new JButton("TAMBAH");
        addButton.setBackground(new Color(60, 179, 113));
        addButton.setForeground(Color.WHITE);
        addButton.setFont(new Font("Arial", Font.BOLD, 14));
        addButton.setFocusPainted(false);
        addButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                String name = nameField.getText();
                String address = addressField.getText();
                String phone = phoneField.getText();
                StringBuilder order = new StringBuilder();
                int total = 0;

                order.append("Nama :
") .append(name) .append("\n");
                order.append("Alamat :
") .append(address) .append("\n");
                order.append("Telp :
") .append(phone) .append("\n");
                order.append("-----
\n");

                order.append("Pesanan:\n");

                if (steakCheckBox.isSelected()) {
                    order.append("~ Steak (50000)\n");
                    total += 50000;
                }
                if (spaghettiCheckBox.isSelected()) {
                    order.append("~ Spageti (40000)\n");
                    total += 40000;
                }
                if (pizzaCheckBox.isSelected()) {
                    order.append("~ Pizza (80000)\n");
                    total += 80000;
                }

                order.append("-----
\n");

                order.append("Total Bayar : Rp.
") .append(total) .append("\n");

                totalField.setText(String.valueOf(total));
                orderArea.setText(order.toString());
            }
        });

        // Mengatur Layout Frame
        JPanel topPanel = new JPanel(new GridLayout(1, 2, 10,
10));
        topPanel.add(customerPanel);
        topPanel.add(menuPanel);
        topPanel.setBorder(BorderFactory.createEmptyBorder(10,
10, 10, 10));

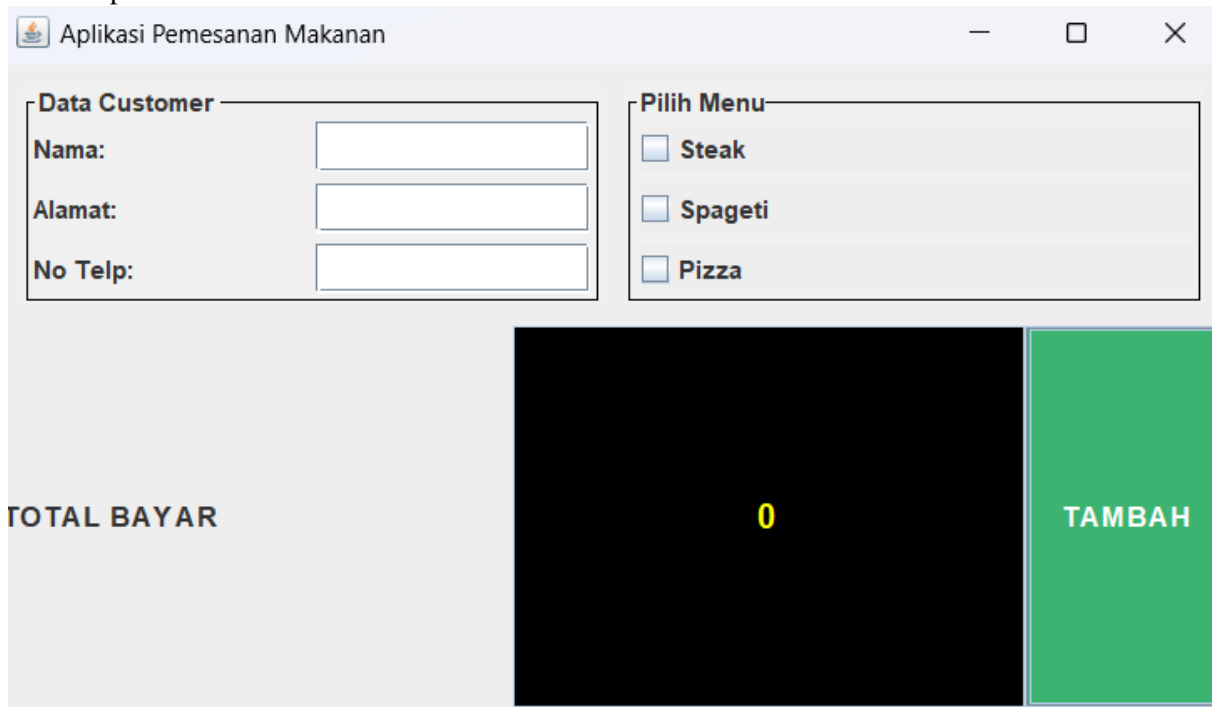
        frame.add(topPanel, BorderLayout.NORTH);
        frame.add(totalPanel, BorderLayout.CENTER);
        frame.add(addButton, BorderLayout.EAST);
        frame.add(scrollPane, BorderLayout.SOUTH);

        // Menempatkan Frame di Tengah
        frame.setLocationRelativeTo(null);

        // Menampilkan Frame
        frame.setVisible(true);
    }
}

```

2. Hasil Implementasi



3. Penjelasan

Aplikasi pemesanan makanan ini dibangun dengan Java Swing dan metode scripting untuk membuat antarmuka grafis (GUI). Aplikasi ini memiliki frame utama 600 x 500 piksel yang diatur dengan layout BorderLayout. Di bagian atas, terdapat panel yang mengumpulkan data pelanggan, seperti nama, alamat, dan nomor telepon, yang dimasukkan melalui JTextField. Panel lain berisi pilihan menu makanan yang dapat dipilih dengan JCheckBox, seperti pizza, steak, dan spaghetti. Di bawah panel menu, terdapat panel total bayar, yang menampilkan harga total berdasarkan pilihan makanan pengguna. Jumlah total pembayaran ditampilkan di sebuah JTextField dengan latar belakang hitam dan teks berwarna kuning untuk pengguna melihatnya.

Aplikasi ini menggunakan JTextArea yang dilengkapi dengan JScrollPane untuk menampilkan detail pesanan. Ini memungkinkan data digulir jika melebihi batas tampilan. Untuk menyelesaikan pesanan, klik tombol "TAMBAH". Ketika tombol ini ditekan, aplikasi akan mengumpulkan data pelanggan dan memeriksa pilihan menu. Setelah itu, harga total akan dihitung dan ditampilkan di area pesanan dan panel total.

bayar. Saya mendapatkan banyak pengetahuan tentang cara mengelola event, mengatur layout, dan mengubah komponen GUI seperti `TextField`, `CheckBox`, dan `TextArea` dengan aplikasi ini. Pengalaman ini meningkatkan pemahaman Anda tentang membuat aplikasi desktop interaktif dengan Java Swing.