TUGAS GUI OPERATOR LOGIKA ALGORITMA PEMROGRAMAN

Dosen pengampu: Dr. Wahyudi S.T, M.T

Disusun oleh:
Muhammad Althaf Mulya
NIM: 2511533018

FAKULTAS TEKNOLOGI INFORMASI
DEPARTEMEN INFORMATIKA
UNIVERSITAS ANDALAS
2025

Program Operator Logika GUI dibuat dengan Java Swing (dibuat pakai WindowBuilder/Eclipse).

Fungsinya: menerima dua input boolean (true / false), menerima pilihan operator logika (AND, OR, NOR, XOR, NOT), lalu menampilkan hasil operasi logika dalam bentuk true/false.

Fitur penting: validasi input (case-insensitive), penonaktifan input kedua saat operator NOT, pesan peringatan/error via JOptionPane, dan penanganan event pada tombol *Hitung*.

Kode Program:

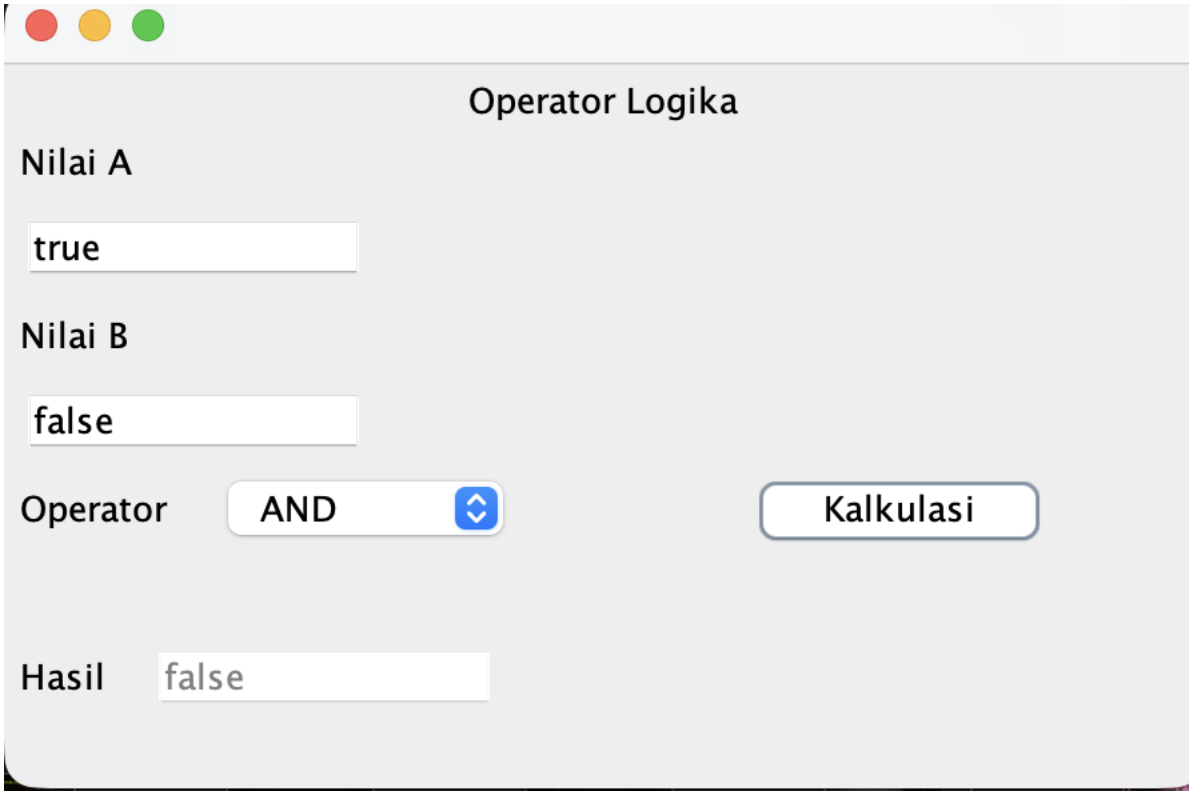
```java
package pekan8_2511533018;

import java.awt.EventQueue;

public class OperatorLogika_2511533018 extends JFrame {

    private static final long serialVersionUID = 1L;
    private JPanel contentPane;
    private JTextField text1;
    private JTextField text2;
    private JTextField textHasil;

    private void pesanPeringatan(String pesan) {
        JOptionPane.showMessageDialog(this, pesan, "Peringatan", JOptionPane.WARNING_MESSAGE);
    }

    private void pesanError(String pesan) {
        JOptionPane.showConfirmDialog(this, pesan, "Kesalahan", JOptionPane.ERROR_MESSAGE);
    }

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    OperatorLogika_2511533018 frame = new OperatorLogika_2511533018();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
```

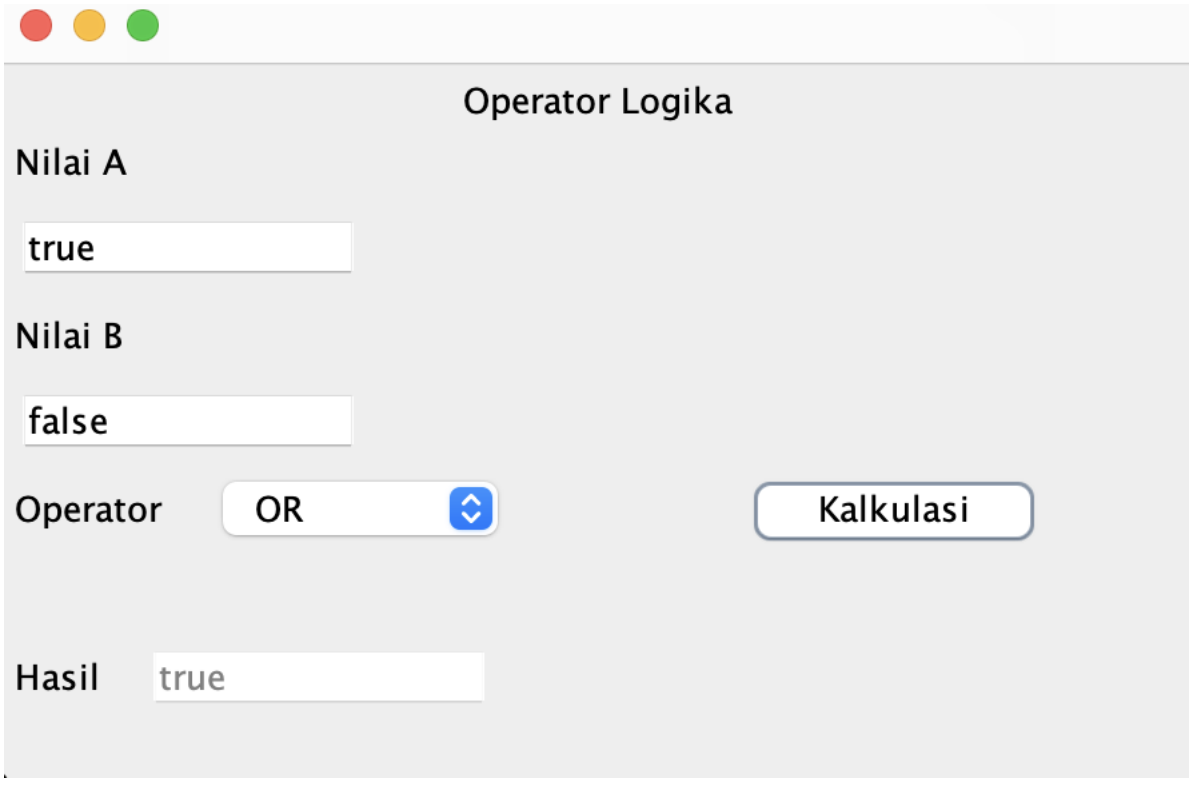
```java
    /**
     * Create the frame.
     */
    public OperatorLogika_2511533018() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 450, 300);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JLabel lblNewLabel = new JLabel("Operator Logika");
        lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
        lblNewLabel.setBounds(153, 6, 143, 16);
        contentPane.add(lblNewLabel);

        JLabel lblNewLabel_1 = new JLabel("Nilai A");
        lblNewLabel_1.setBounds(6, 29, 61, 16);
        contentPane.add(lblNewLabel_1);

        JLabel lblNewLabel_2 = new JLabel("Nilai B");
        lblNewLabel_2.setBounds(6, 94, 61, 16);
        contentPane.add(lblNewLabel_2);

        JLabel lblNewLabel_3 = new JLabel("Operator");
        lblNewLabel_3.setBounds(6, 159, 61, 16);
        contentPane.add(lblNewLabel_3);

        JLabel lblNewLabel_4 = new JLabel("Hasil");
        lblNewLabel_4.setBounds(6, 222, 61, 16);
        contentPane.add(lblNewLabel_4);

        JComboBox comboBox = new JComboBox();
        comboBox.setModel(new DefaultComboBoxModel(new String[] {
```

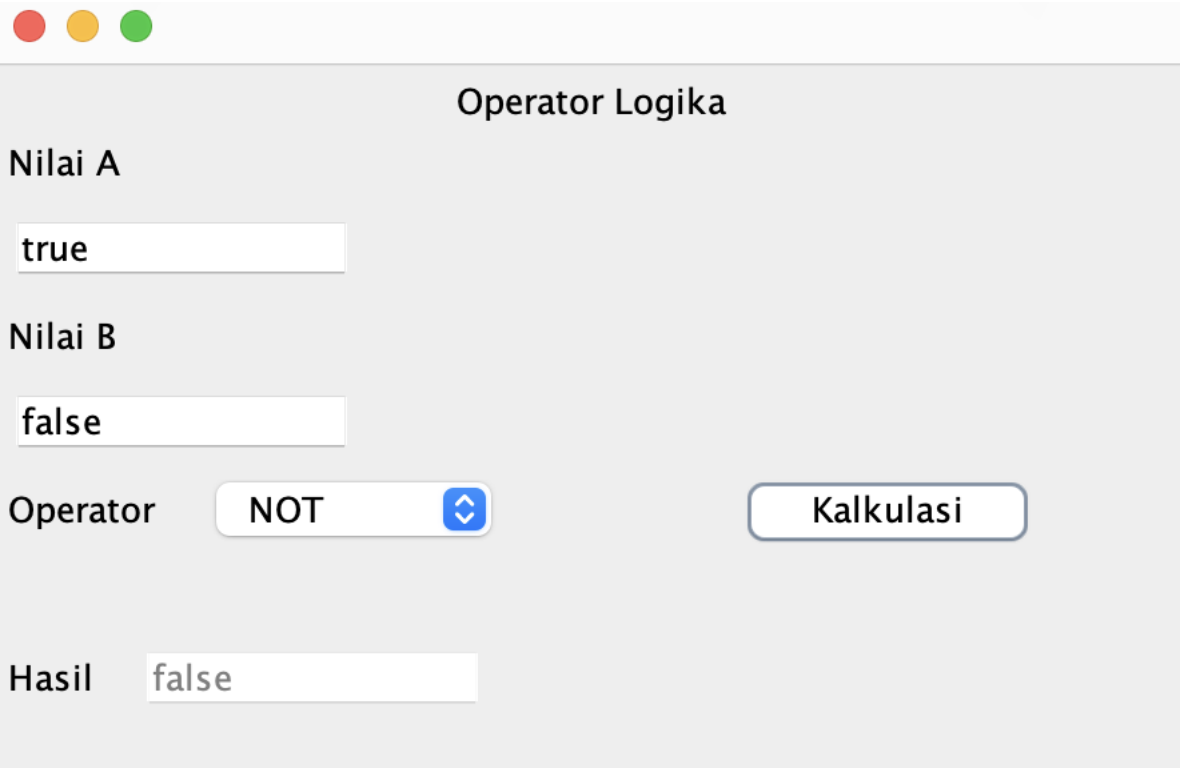
```
        comboBox.setModel(new DefaultComboBoxModel(new String[] {
                "AND",
                "OR",
                "NOR",
                "XOR",
                "NOT"
            }));
        comboBox.setBounds(79, 155, 52, 27);
        contentPane.add(comboBox);

        text1 = new JTextField();
        text1.setBounds(6, 56, 130, 26);
        contentPane.add(text1);
        text1.setColumns(10);

        text2 = new JTextField();
        text2.setBounds(6, 121, 130, 26);
        contentPane.add(text2);
        text2.setColumns(10);

        textHasil = new JTextField();
        textHasil.setEditable(false);
        textHasil.setEnabled(false);
        textHasil.setBounds(55, 217, 130, 26);
        contentPane.add(textHasil);
        textHasil.setColumns(10);

        JButton btnCalc = new JButton("Kalkulasi");
        btnCalc.addActionListener(new ActionListener() {

        public void actionPerformed(ActionEvent e) {

            String s1 = text1.getText().trim().toLowerCase();
            String s2 = text2.getText().trim().toLowerCase();
            String op = comboBox.getSelectedItem().toString();
```

```
        public void actionPerformed(ActionEvent e) {

            String s1 = text1.getText().trim().toLowerCase();
            String s2 = text2.getText().trim().toLowerCase();
            String op = comboBox.getSelectedItem().toString();

            if (op.equals("NOT")) {
                if (!s1.equals("true") && !s1.equals("false")) {
                    pesanError("Input harus true atau false.");
                    return;
                }
            } else {
                if (!s1.equals("true") && !s1.equals("false")) {
                    pesanError("Input 1 harus true atau false.");
                    return;
                }
                if (!s2.equals("true") && !s2.equals("false")) {
                    pesanError("Input 2 harus true atau false.");
                    return;
                }
            }

            boolean a = Boolean.parseBoolean(s1);
            boolean b = Boolean.parseBoolean(s2);
            boolean hasil = false;

            switch (op) {
                case "AND": hasil = a && b; break;
                case "OR": hasil = a || b; break;
                case "NOR": hasil = !(a || b); break;
                case "XOR": hasil = a ^ b; break;
                case "NOT": hasil = !a; break;
            }

            textHasil.setText(String.valueOf(hasil));
        }
```

Output Program:


Operator Logika
Nilai A
true
Nilai B
false
Operator
AND
Kalkulasi
Hasil
false


Operator Logika
Nilai A
true
Nilai B
false
Operator
OR
Kalkulasi
Hasil
true



Psedocode:

START

Read s1 = trim(lowercase(textField1))

Read s2 = trim(lowercase(textField2))

op = comboBox.selectedItem

IF op == "NOT" THEN

    IF s1 not in {"true","false"} THEN

        showError("Input harus true atau false")

        STOP

    END IF

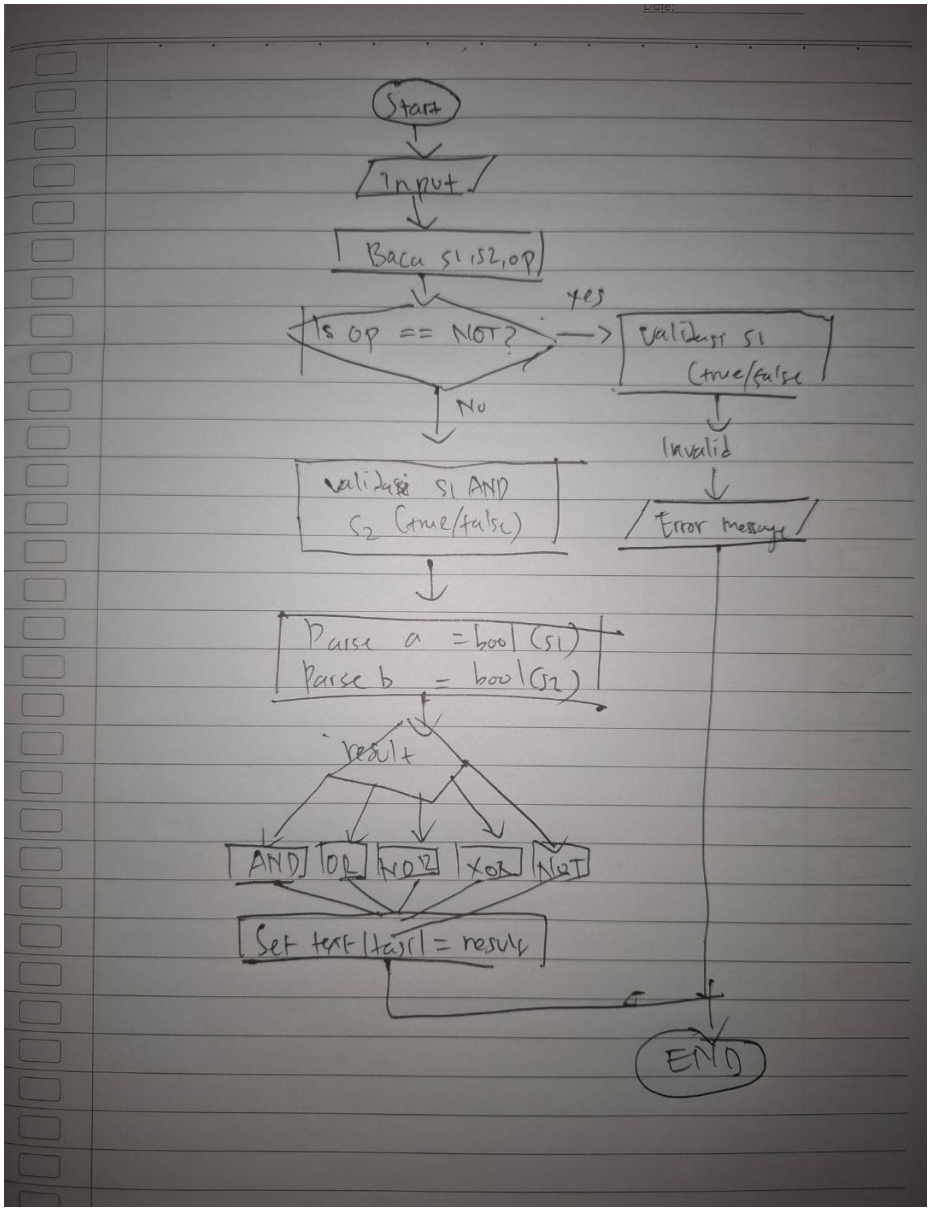
ELSE

    IF s1 not in {"true","false"} OR s2 not in {"true","false"} THEN

        showError("Input 1/2 harus true atau false")

```
        STOP
    END IF
END IF

a = parseBoolean(s1)
b = parseBoolean(s2)

SWITCH op
    CASE "AND":  result = a && b
    CASE "OR":   result = a || b
    CASE "NOR":  result = !(a || b)
    CASE "XOR":  result = a ^ b
    CASE "NOT":  result = !a
END SWITCH
 textResult.setText(stringify(result))

END
```

Flowchart:


Start
Input
Baca s1, s2, op
Is op == NOT?
yes
Validasi s1 (true/false
No
Invalid
Validasi s1 AND s2 (true/false)
Error message
Parse a = bool(s1)
Parse b = bool(s2)
result
AND
OR
NOR
XOR
NOT
Set text label = result
END