D3 Teknik Informatika (TI)
Jurusan Teknik Informatika
Politeknik Negeri Indramayu

UI Swing Responsif

Fachrul Pralienka Bani Muhamad, S.ST., M.Kom.
fachrul.pbm@polindra.ac.id

2025

Tujuan

Setelah materi ini disampaikan, mahasiswa diharapkan mampu:

- **Menyajikan** tampilan aplikasi yang responsif berdasarkan ukuran jendela saat ini melalui penerapan event handling

Pengantar UI Swing Responsif

- Aplikasi desktop yang tidak modern seringkali didesain dengan ukuran jendela yang tetap (kaku)
- Aplikasi desktop yang modern, fleksibel, dan ramah pengguna, memastikan pengalaman yang konsisten terlepas dari lingkungan tampilan pengguna
- Swing memiliki kemampuan untuk mengubah tata letak (layout), ukuran, dan visibilitas komponennya secara dinamis sebagai respons terhadap perubahan dimensi jendela aplikasi

ComponentListener

- Event handler yang mendengarkan perubahan ukuran jendela secara real-time
- Penghubung antara tindakan pengguna (mengubah ukuran jendela) dan logika responsifitas aplikasi
- Memicu logika perubahan tata letak
- Setiap kali lebar atau tinggi jendela berubah, metode componentResized() dipanggil
- Dapat diperoleh ukuran jendela Saat Ini
- Memvalidasi ulang tata letak (revalidate() dan repaint())

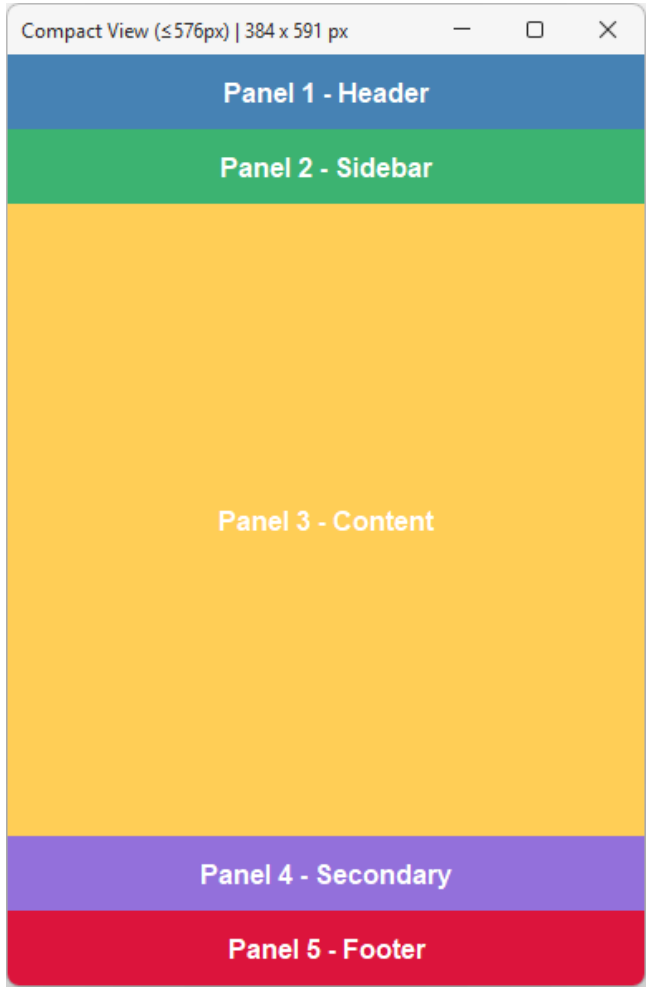
Threshold Lebar (Width) Jendela Aplikasi

No	Kategori	Treshold (px)
1	Compact View	≤ 576 px
2	Split View	576 - 768 px
3	Intermediate View	768 - 992 px
4	Desktop Small	992 - 1200 px
5	Desktop Standard	1200 - 1400 px)
6	Desktop Large	≥ 1400 px

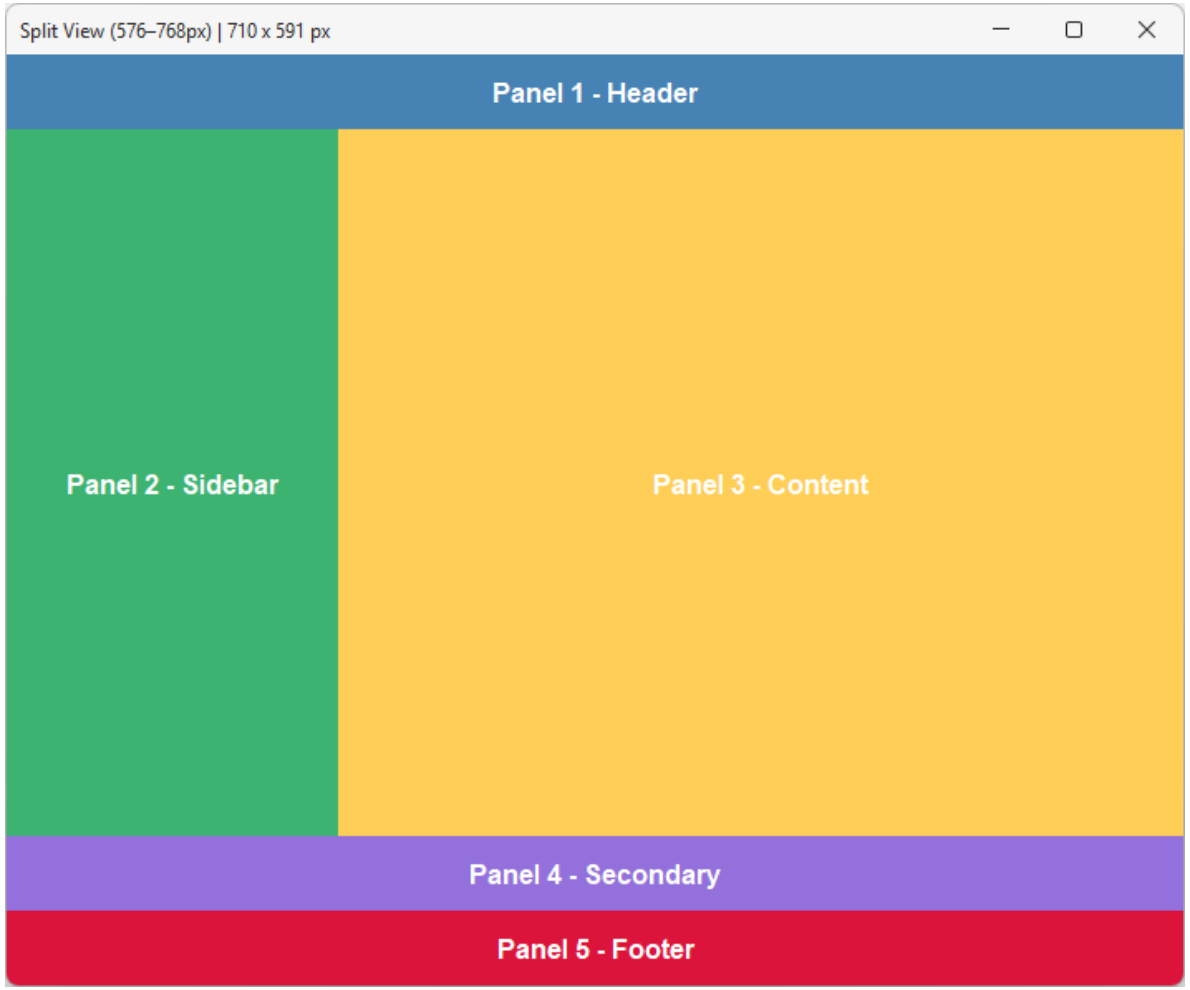
Langkah Penerapan

1. Merancang tata letak (layout) awal
2. Menambahkan event listener (componentListener) dengan Adapter componentResized
3. Menerapkan logika threshold ukuran jendela
4. Mengubah tata letak (layout) berdasarkan threshold

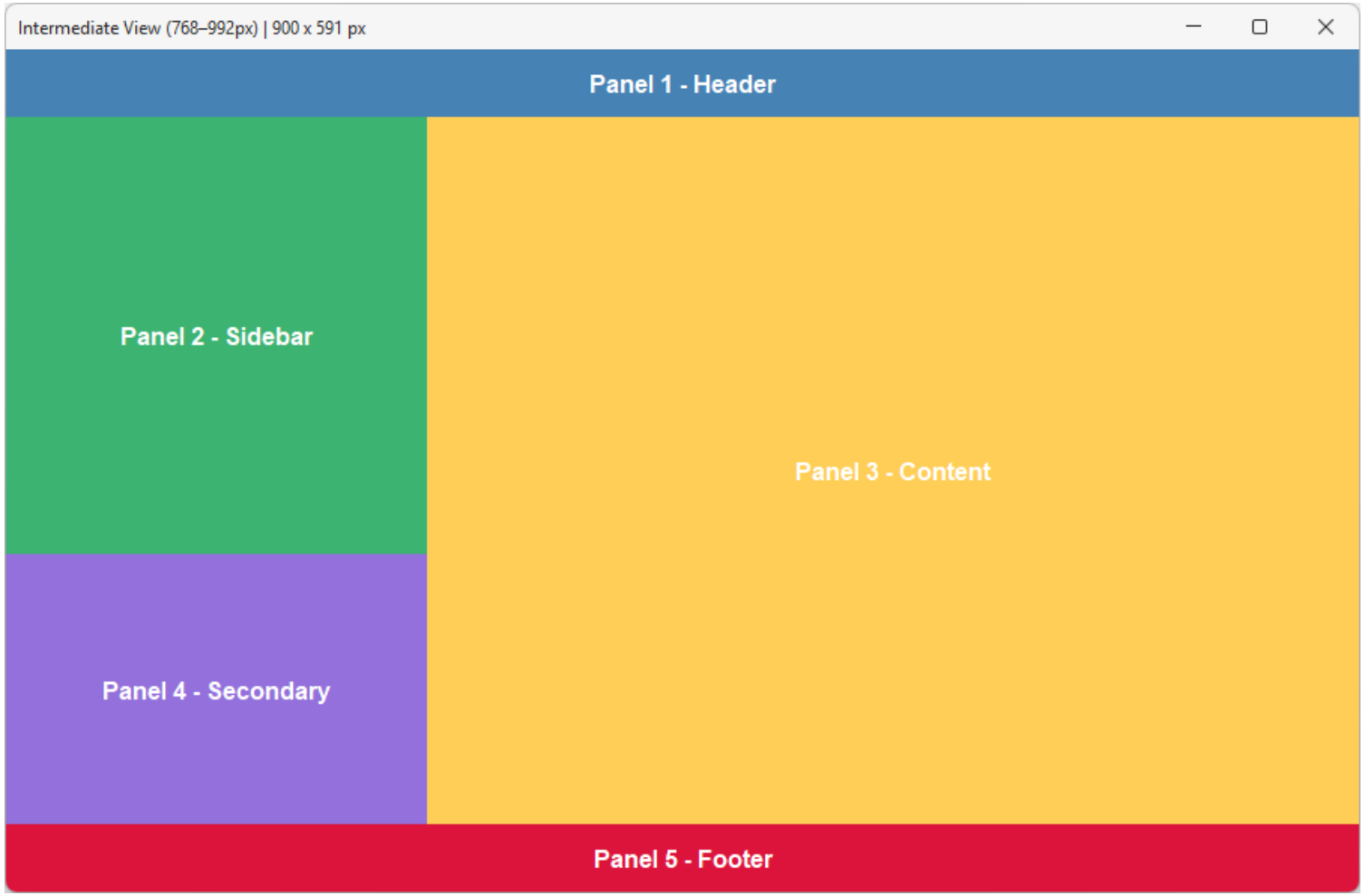
Compact View (≤ 576 px)



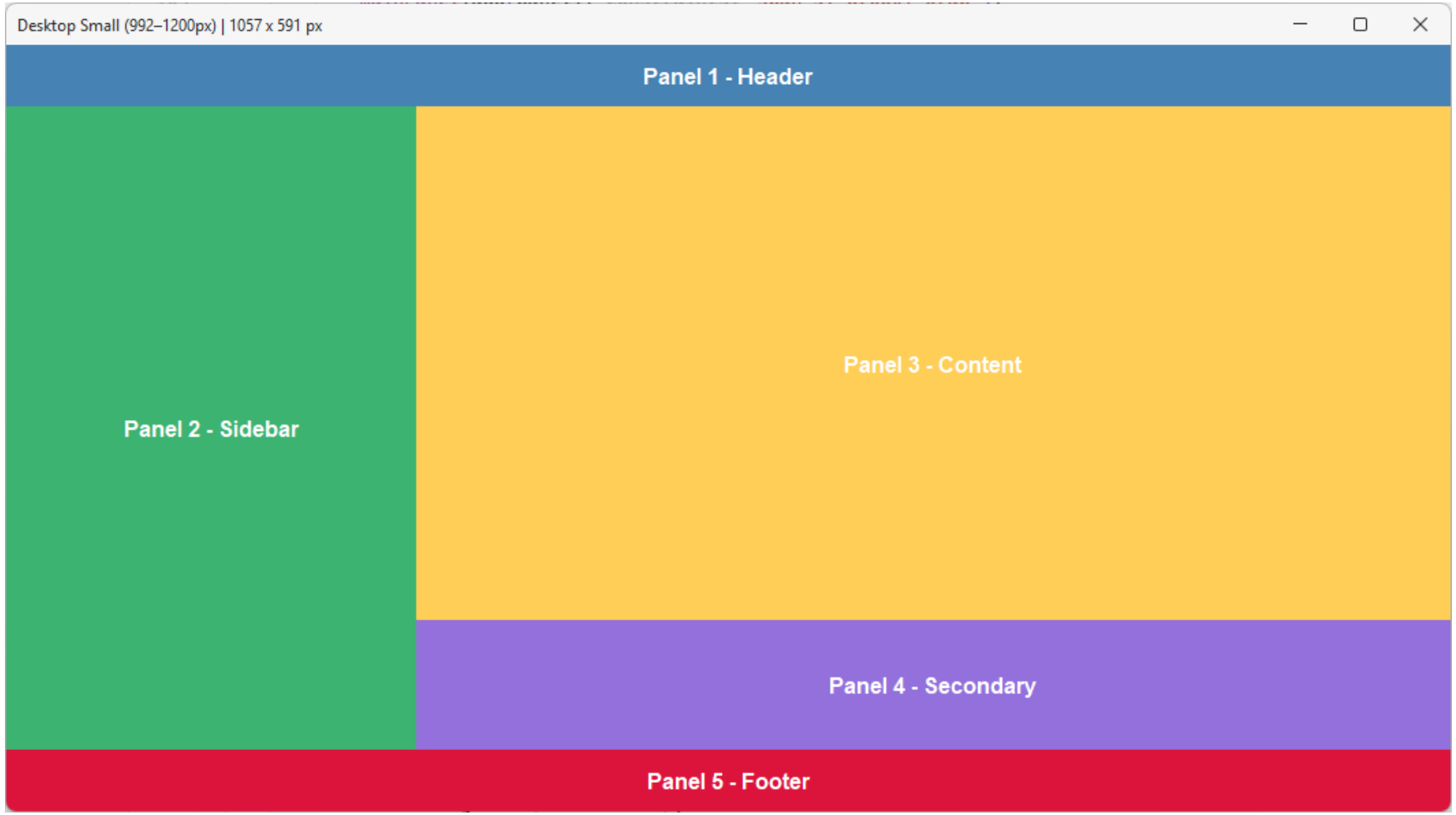
Split View (576 - 768 px)



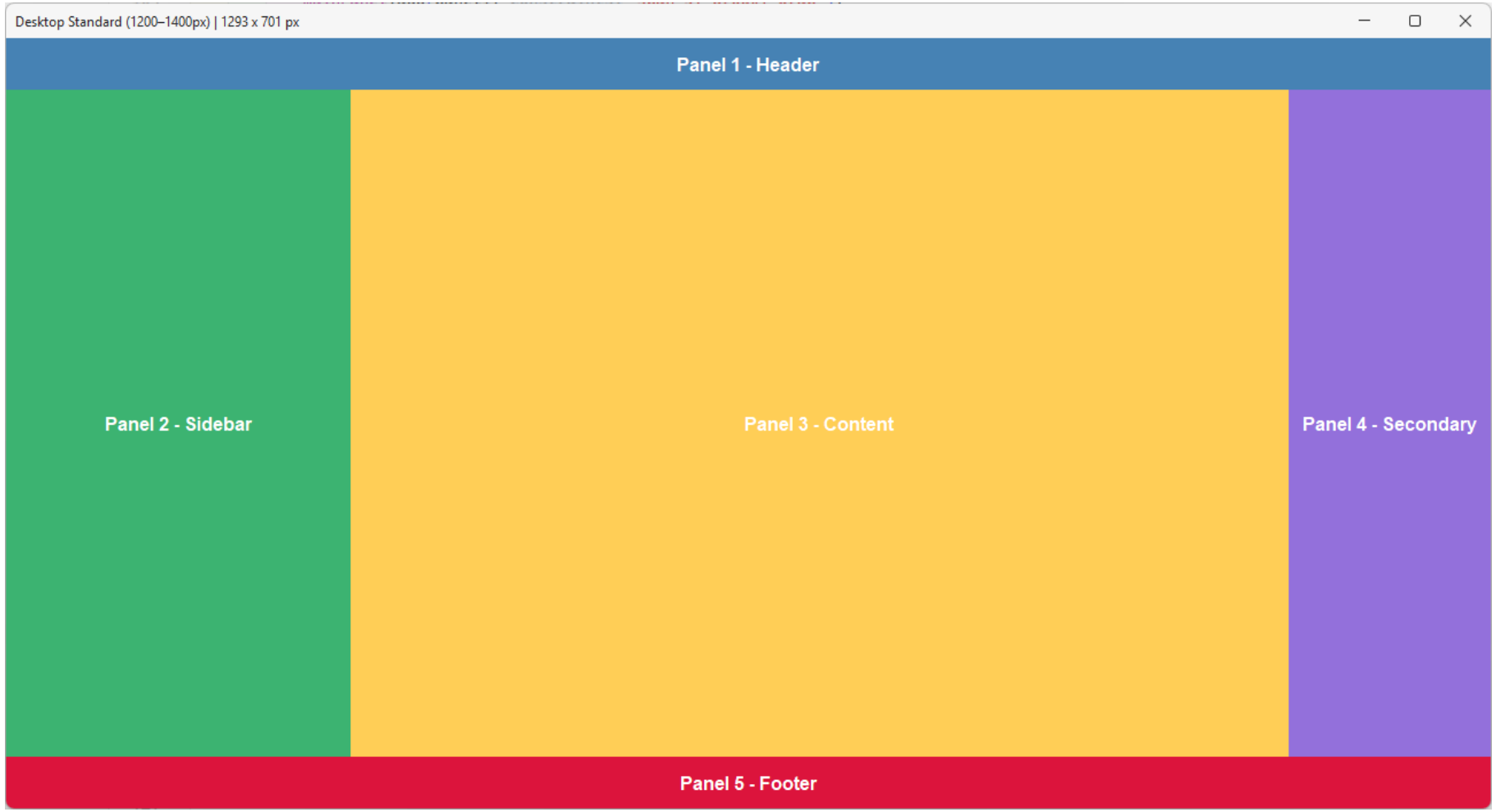
Intermediate View (768 - 992 px)



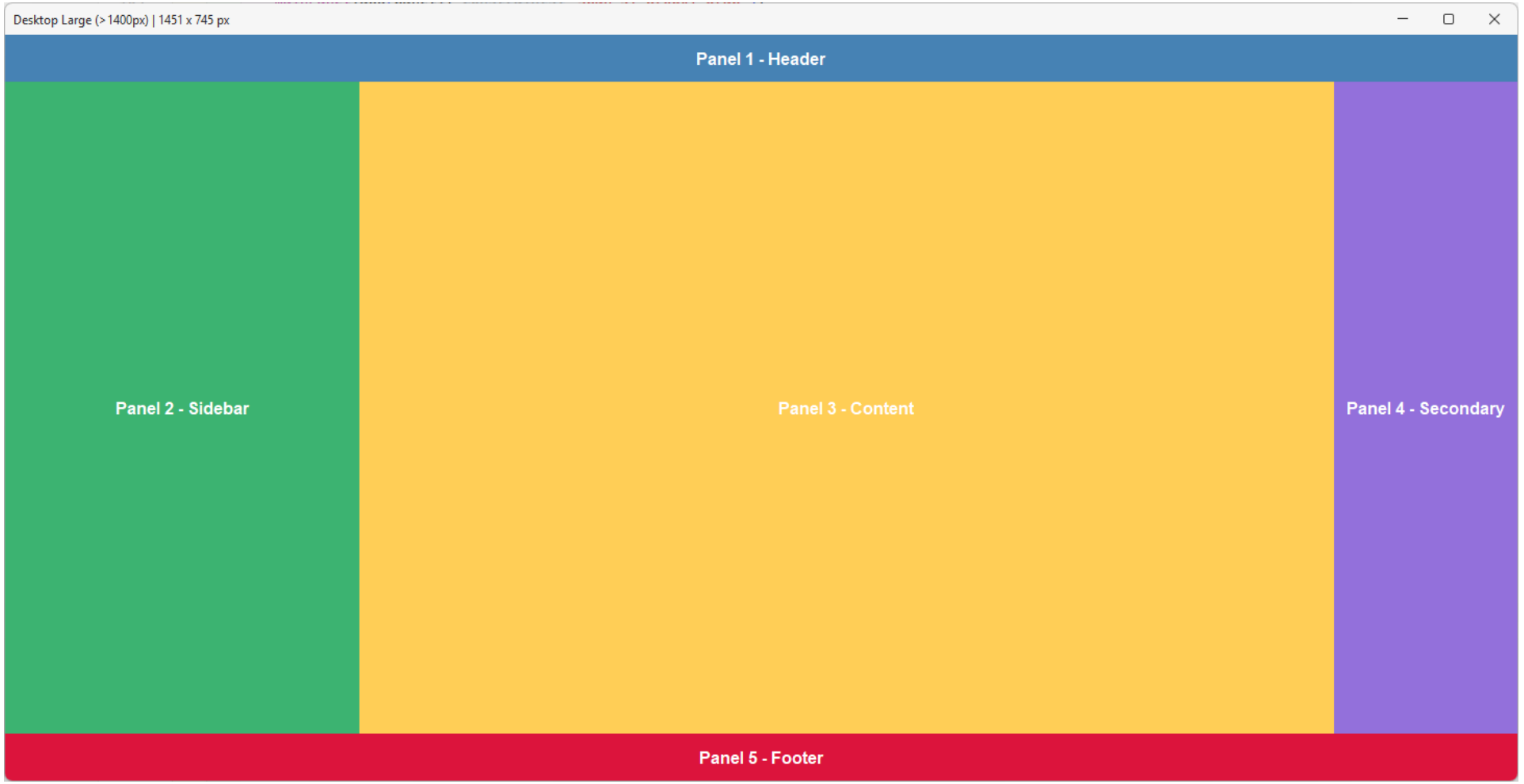
Desktop Small (992 - 1200 px)



Desktop Standard (1200 - 1400 px)



Desktop Large (≥ 1400 px)



```java
public class UISwingResponsif extends JFrame {

    private JPanel mainPanel;
    private JPanel panel1;
    private JPanel panel2;
    private JPanel panel3;
    private JPanel panel4;
    private JPanel panel5;

    private static final int COMPACT_VIEW_MAX = 576;
    private static final int SPLIT_VIEW_MAX = 768;
    private static final int INTERMEDIATE_VIEW_MAX = 992;
    private static final int DESKTOP_SMALL_MAX = 1200;
    private static final int DESKTOP_STANDARD_MAX = 1400;

    public UISwingResponsif() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setMinimumSize(new Dimension(400, 600));

        panel1 = createColoredPanel("Panel 1 - Header", new Color(70, 130, 180));
        panel2 = createColoredPanel("Panel 2 - Sidebar", new Color(60, 179, 113));
        panel3 = createColoredPanel("Panel 3 - Content", new Color(255, 206, 86));
        panel4 = createColoredPanel("Panel 4 - Secondary", new Color(147, 112, 219));
        panel5 = createColoredPanel("Panel 5 - Footer", new Color(220, 20, 60));

        mainPanel = new JPanel();
        add(mainPanel);
```

```
38          addComponentListener(new ComponentAdapter() {
39              @Override
40              public void componentResized(ComponentEvent e) {
41                  updateLayoutAndTitle();
42              }
43
44              @Override
45              public void componentMoved(ComponentEvent e) {
46                  updateLayoutAndTitle();
47              }
48          });
49
50          updateLayoutAndTitle();
51
52          pack();
53          setExtendedState(JFrame.MAXIMIZED_BOTH);
54          setLocationRelativeTo(null);
55      }
56
57      private void updateLayoutAndTitle() {
58          Dimension size = getSize();  // ukuran aktual termasuk border
59          Insets insets = getInsets(); // border + title bar
60          int innerWidth  = size.width  - insets.left - insets.right;
61          int innerHeight = size.height - insets.top  - insets.bottom;
62
63          String category = getDeviceCategory(innerWidth);
64          setTitle(category + " | " + innerWidth + " x " + innerHeight + " px");
65
66          rebuildLayout(innerWidth);
67      }
```

```
68
69      private String getDeviceCategory(int width) {
70          if (width <= COMPACT_VIEW_MAX) return "Compact View (≤576px)";
71          else if (width <= SPLIT_VIEW_MAX) return "Split View (576-768px)";
72          else if (width <= INTERMEDIATE_VIEW_MAX) return "Intermediate View (768-992px)";
73          else if (width <= DESKTOP_SMALL_MAX) return "Desktop Small (992-1200px)";
74          else if (width <= DESKTOP_STANDARD_MAX) return "Desktop Standard (1200-1400px)";
75          else return "Desktop Large (>1400px)";
76      }
77
78      private void rebuildLayout(int width) {
79          mainPanel.removeAll();
80          mainPanel.setLayout(new MigLayout("fill, insets 0, gap 0"));
81
82          if (width <= COMPACT_VIEW_MAX) {
83              mainPanel.add(panel1, "grow, wrap");
84              mainPanel.add(panel2, "grow, wrap");
85              mainPanel.add(panel3, "grow, push, h 300::, wrap");
86              mainPanel.add(panel4, "grow, wrap");
87              mainPanel.add(panel5, "grow, wrap");
88
89          } else if (width <= SPLIT_VIEW_MAX) {
90              mainPanel.add(panel1, "span 2, growx, wrap");
91              mainPanel.add(panel2, "w 200!, growy");
92              mainPanel.add(panel3, "grow, push, wrap");
93              mainPanel.add(panel4, "span 2, growx, wrap");
94              mainPanel.add(panel5, "span 2, growx, wrap");
```

```
96          } else if (width <= INTERMEDIATE_VIEW_MAX) {
97              mainPanel.add(panel1, "span 2, growx, wrap");
98              mainPanel.add(panel2, "w 280!, growy");
99              mainPanel.add(panel3, "grow, push, wrap, spany 2");
100             mainPanel.add(panel4, "growx, h 180!, wrap");
101             mainPanel.add(panel5, "span 2, growx, wrap");
102
103         } else if (width <= DESKTOP_SMALL_MAX) {
104             mainPanel.add(panel1, "span 3, growx, wrap");
105             mainPanel.add(panel2, "growy, span 1 2, w 300!, h 100%"); // span 1 kolom × 2 baris
106             mainPanel.add(panel3, "grow, push, wrap");
107             mainPanel.add(panel4, "h 25%, growx, wrap");
108             mainPanel.add(panel5, "span 3, growx");
109
110         } else if (width <= DESKTOP_STANDARD_MAX) {
111             mainPanel.add(panel1, "span 3, growx, wrap");
112             mainPanel.add(panel2, "w 300!, growy");
113             mainPanel.add(panel3, "grow, pushx 2, pushy");
114             mainPanel.add(panel4, "grow, wrap");
115             mainPanel.add(panel5, "span 3, growx, wrap");
116
117         } else {
118             mainPanel.add(panel1, "span 3, growx, wrap");
119             mainPanel.add(panel2, "w 340!, growy");
120             mainPanel.add(panel3, "grow, pushx 3, pushy");
121             mainPanel.add(panel4, "grow, wrap");
122             mainPanel.add(panel5, "span 3, growx, wrap");
123         }
124
125         mainPanel.revalidate();
126         mainPanel.repaint();
127     }
```

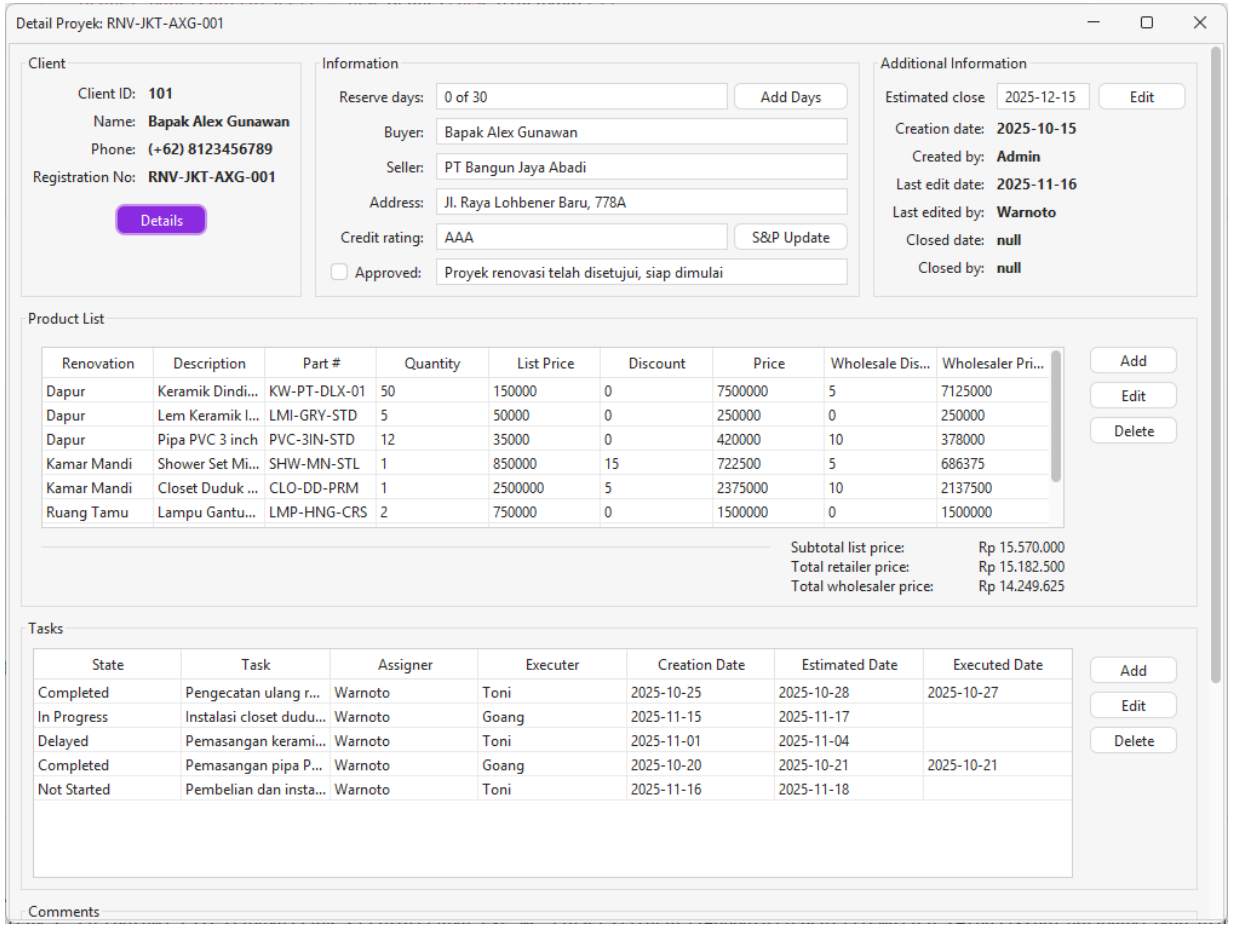
```
    private JPanel createColoredPanel(String title, Color background) {
        JPanel panel = new JPanel(new MigLayout("fill, insets 0"));
        panel.setBackground(background);
        panel.setBorder(BorderFactory.createEmptyBorder(12, 12, 12, 12));
        JLabel label = new JLabel(title, SwingConstants.CENTER);
        label.setFont(new Font("Inter", Font.BOLD, 16));
        label.setForeground(Color.WHITE);
        label.setOpaque(false);
        panel.add(label, "grow, align center");

        return panel;
    }

    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(new FlatMacLightLaf());
        } catch (UnsupportedLookAndFeelException e) {
            e.printStackTrace();
        }
        SwingUtilities.invokeLater(() -> {
            new UISwingResponsif().setVisible(true);
        });
    }
}
```

Latihan

- **Split View** (576 - 768 px)
  (semua panel melebar kiri ke kanan)
- **Desktop Small** (992 - 1200 px)
  (panel client dan panel additional information bersebelahan, panel information memanjang dan berada di bawah kedua panel tersebut)
- **Desktop Standard** (1200 - 1400 px)
  (normal sesuai layout awal)



Selesai

Terima kasih 🙏