

LAMPIRAN

Koding Program

```
/*  
 * Sistem_DMNaiveBayesView.java  
 */  
  
package sistem_dmnaivebayes;  
  
import org.jdesktop.application.Action;  
import org.jdesktop.application.ResourceMap;  
import org.jdesktop.application.SingleFrameApplication;  
import org.jdesktop.application.FrameView;  
import org.jdesktop.application.TaskMonitor;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import javax.swing.Timer;  
import javax.swing.Icon;  
import javax.swing.JDialog;  
import javax.swing.JFrame;  
  
/**  
 * The application's main frame.  
 */  
public class Sistem_DMNaiveBayesView extends FrameView {  
  
    public Sistem_DMNaiveBayesView(SingleFrameApplication app) {  
        super(app);  
  
        initComponents();  
    }  
}
```

```

// status bar initialization - message timeout, idle icon and busy animation, etc
ResourceMap resourceMap = getResourceMap();
int messageTimeout = resourceMap.getInteger("StatusBar.messageTimeout");
messageTimer = new Timer(messageTimeout, new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        statusMessageLabel.setText("");
    }
});
messageTimer.setRepeats(false);
int busyAnimationRate = resourceMap.getInteger("StatusBar.busyAnimationRate");
for (int i = 0; i < busyIcons.length; i++) {
    busyIcons[i] = resourceMap.getIcon("StatusBar.busyIcons[" + i + "]");
}
busyIconTimer = new Timer(busyAnimationRate, new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        busyIconIndex = (busyIconIndex + 1) % busyIcons.length;
        statusAnimationLabel.setIcon(busyIcons[busyIconIndex]);
    }
});
idleIcon = resourceMap.getIcon("StatusBar.idleIcon");
statusAnimationLabel.setIcon(idleIcon);
progressBar.setVisible(false);

// connecting action tasks to status bar via TaskMonitor
TaskMonitor taskMonitor = new TaskMonitor(getApplication().getContext());
taskMonitor.addPropertyChangeListener(new java.beans.PropertyChangeListener() {
    public void propertyChange(java.beans.PropertyChangeEvent evt) {
        String propertyName = evt.getPropertyName();
        if ("started".equals(propertyName)) {
            if (!busyIconTimer.isRunning()) {
                statusAnimationLabel.setIcon(busyIcons[0]);
            }
        }
    }
});

```

```

        busyIconIndex = 0;
        busyIconTimer.start();
    }
    progressBar.setVisible(true);
    progressBar.setIndeterminate(true);
} else if ("done".equals(propertyName)) {
    busyIconTimer.stop();
    statusAnimationLabel.setIcon(idleIcon);
    progressBar.setVisible(false);
    progressBar.setValue(0);
} else if ("message".equals(propertyName)) {
    String text = (String)(evt.getNewValue());
    statusMessageLabel.setText((text == null) ? "" : text);
    messageTimer.restart();
} else if ("progress".equals(propertyName)) {
    int value = (Integer)(evt.getNewValue());
    progressBar.setVisible(true);
    progressBar.setIndeterminate(false);
    progressBar.setValue(value);
}
}
});
}

```

@Action

```

public void showAboutBox() {
    if (aboutBox == null) {
        JFrame mainFrame = Sistem_DMNaiveBayesApp.getApplication().getMainFrame();
        aboutBox = new Sistem_DMNaiveBayesAboutBox(mainFrame);
        aboutBox.setLocationRelativeTo(mainFrame);
    }
}

```

```

    Sistem_DMNaiveBayesApp.getApplication().show(aboutBox);
}

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    mainPanel = new javax.swing.JPanel();
    menampilkan = new PanelBackground();
    menuBar = new javax.swing.JMenuBar();
    javax.swing.JMenu fileMenu = new javax.swing.JMenu();
    JMenuItem1 = new javax.swing.JMenuItem();
    JMenuItem2 = new javax.swing.JMenuItem();
    JMenuItem3 = new javax.swing.JMenuItem();
    javax.swing.JMenuItem exitMenuItem = new javax.swing.JMenuItem();
    statusPanel = new javax.swing.JPanel();
    javax.swing.JSeparator statusPanelSeparator = new javax.swing.JSeparator();
    statusMessageLabel = new javax.swing.JLabel();
    statusAnimationLabel = new javax.swing.JLabel();
    progressBar = new javax.swing.JProgressBar();

    mainPanel.setName("mainPanel"); // NOI18N

    menampilkan.setName("menampilkan"); // NOI18N

    javax.swing.GroupLayout mainPanelLayout = new javax.swing.GroupLayout(mainPanel);
    mainPanel.setLayout(mainPanelLayout);

```

```

mainPanelLayout.setHorizontalGroup(
    mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(menampilkan, javax.swing.GroupLayout.DEFAULT_SIZE, 393,
Short.MAX_VALUE)
);
mainPanelLayout.setVerticalGroup(
    mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(menampilkan, javax.swing.GroupLayout.DEFAULT_SIZE, 239,
Short.MAX_VALUE)
);

menuBar.setName("menuBar"); // NOI18N

org.jdesktop.application.ResourceMap resourceMap =
org.jdesktop.application.Application.getInstance(sistem_dmnaivebayes.Sistem_DMNaiveBayesApp.class
).getContext().getResourceMap(Sistem_DMNaiveBayesView.class);
fileMenu.setText(resourceMap.getString("fileMenu.text")); // NOI18N
fileMenu.setName("fileMenu"); // NOI18N

jMenuItem1.setText(resourceMap.getString("jMenuItem1.text")); // NOI18N
jMenuItem1.setName("jMenuItem1"); // NOI18N
jMenuItem1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem1ActionPerformed(evt);
    }
});
fileMenu.add(jMenuItem1);

jMenuItem2.setText(resourceMap.getString("jMenuItem2.text")); // NOI18N
jMenuItem2.setName("jMenuItem2"); // NOI18N
jMenuItem2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem2ActionPerformed(evt);
    }
});
fileMenu.add(jMenuItem2);

```

```
    }  
});  
fileMenu.add(jMenuItem2);  
  
jMenuItem3.setText(resourceMap.getString("jMenuItem3.text")); // NOI18N  
jMenuItem3.setName("jMenuItem3"); // NOI18N  
jMenuItem3.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        jMenuItem3ActionPerformed(evt);  
    }  
});  
fileMenu.add(jMenuItem3);  
  
javax.swing.ActionMap actionMap =  
org.jdesktop.application.Application.getInstance(sistem_dmnaivebayes.Sistem_DMNaiveBayesApp.class  
).getContext().getActionMap(Sistem_DMNaiveBayesView.class, this);  
  
exitMenuItem.setAction(actionMap.get("quit")); // NOI18N  
exitMenuItem.setName("exitMenuItem"); // NOI18N  
fileMenu.add(exitMenuItem);  
  
menuBar.add(fileMenu);  
  
statusPanel.setName("statusPanel"); // NOI18N  
  
statusPanelSeparator.setName("statusPanelSeparator"); // NOI18N  
  
statusMessageLabel.setName("statusMessageLabel"); // NOI18N  
  
statusAnimationLabel.setHorizontalAlignment(javax.swing.SwingConstants.LEFT);  
statusAnimationLabel.setName("statusAnimationLabel"); // NOI18N  
  
progressBar.setName("progressBar"); // NOI18N
```

```

javax.swing.GroupLayout statusPanelLayout = new javax.swing.GroupLayout(statusPanel);
statusPanel.setLayout(statusPanelLayout);
statusPanelLayout.setHorizontalGroup(
    statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(statusPanelSeparator, javax.swing.GroupLayout.DEFAULT_SIZE, 393,
Short.MAX_VALUE)
        .addGroup(statusPanelLayout.createSequentialGroup()
            .addComponent(statusMessageLabel)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 223,
Short.MAX_VALUE)
            .addComponent(progressBar, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(statusAnimationLabel)
            .addComponent(statusAnimationLabel)
        );
statusPanelLayout.setVerticalGroup(
    statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(statusPanelLayout.createSequentialGroup()
            .addComponent(statusPanelSeparator, javax.swing.GroupLayout.PREFERRED_SIZE, 2,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addGroup(statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(statusMessageLabel)
                .addComponent(statusAnimationLabel)
                .addComponent(progressBar, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            )
            .addGap(3, 3, 3)
        );

```

```
    setComponent(mainPanel);
    setMenuBar(menuBar);
    setStatusBar(statusPanel);
} // </editor-fold>

private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {
    Pola_Naivebayes x= new Pola_Naivebayes();
    menampilkan.add(x);
    x.setVisible(true);
}

private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {
    PrediksiNB x= new PrediksiNB();
    x.setVisible(true);
}

private void jMenuItem3ActionPerformed(java.awt.event.ActionEvent evt) {
    Data_siswa x= new Data_siswa();
    menampilkan.add(x);
    x.setVisible(true);
}

// Variables declaration - do not modify
private javax.swing.JMenuItem jMenuItem1;
private javax.swing.JMenuItem jMenuItem2;
private javax.swing.JMenuItem jMenuItem3;
private javax.swing.JPanel mainPanel;
private javax.swing.JDesktopPane menampilkan;
private javax.swing.JMenuBar menuBar;
private javax.swing.JProgressBar progressBar;
private javax.swing.JLabel statusAnimationLabel;
```



```
private javax.swing.JLabel statusMessageLabel;
private javax.swing.JPanel statusPanel;
// End of variables declaration

private final Timer messageTimer;
private final Timer busyIconTimer;
private final Icon idleIcon;
private final Icon[] busyIcons = new Icon[15];
private int busyIconIndex = 0;

private JDialog aboutBox;
}
```